

BA76.76
I582.3



UNIVERSIDAD DE LOS ANDES FACULTAD DE INGENIERÍA
DOCTORADO EN CIENCIAS APLICADAS

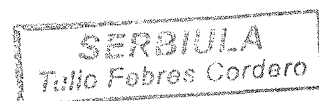
Sistema de Control Inteligente con Alto Grado de Autonomía para Procesos Industriales Complejos

Tesis doctoral presentada ante la ilustre Universidad de Los Andes
como requisito parcial para obtener el título de:
Doctor en Ciencias Aplicadas

Autor: Miguel Angel Ramírez Canelón

Director:
Dr. Eliezer Colina Morles

Merida, Mayo del 2013



Dedicatoria

A Mi Hijo Anyelo

A Mi Madre Gladys

A Mi Padre Carmelo

A Mi Hermana Adriana y Mi Hermano Carmelo

A todas mis tías, tíos, primas y primos

A la memoria de Mi Primo Hermano Juvenal

A la memoria de los Profesores Federico Martín y Edilio Escalona

www.bdigital.ula.ve

Agradecimientos

*A nuestro Señor Jesucristo...“Dad gracias en todo, porque esta es la voluntad de Dios para con Vosotros en Cristo Jesús”.
Tesalonicenses 5,18*

*“El temor de Jehová es el principio de la sabiduría, y el conocimiento del Santísimo es la inteligencia”.
Proverbios 9,10*

Quiero expresar mi más profundo agradecimiento a mi Director de Tesis, el Profesor Eliezer Colina Morles, por su apoyo, tiempo y dedicación y quien ha sabido darme suficiente libertad para desarrollar este trabajo y al mismo tiempo aconsejarme para tomar decisiones y ayudarme a terminarlo de la mejor manera. Además, es importante resaltar que sus consejos y directrices durante el desarrollo de las publicaciones presentadas en congresos internacionales en el período 2007 - 2011, fueron de vital importancia.

Agradezco de manera especial a los Profesores Robert Babuska de la Universidad de Delft en Holanda y José Luis Díez de la Universidad Politécnica de Valencia en España por la asesoría brindada sobre modelación difusa y control basado en modelos difusos. Igualmente, deseo expresar mi agradecimiento al Profesor Janos Abonyi de la Universidad de Veszprém en Hungría por la facilitación del “toolbox” de modelación difusa, el cual fue de gran utilidad.

Deseo expresar mi agradecimiento al Profesor Jacinto Dávila de la Universidad de Los Andes por toda su colaboración brindada en el área de sistemas multi-agente. De igual forma, agradezco a los Profesores Vicente Julian y Vicente Botti Navarro de la Universidad Politécnica de Valencia España por la facilitación de tesis doctorales sobre este tema.

También quiero agradecer al Profesor Edgar Chacón de la Universidad de los Andes por la facilitación de varias publicaciones y tesis doctorales sobre sistemas supervisores inteligentes y gracias a sus importantes sugerencias se logró mejorar

de manera significativa todo lo concerniente al tema de supervisión de la presente tesis. Igualmente, en este mismo sentido, estoy sumamente agradecido con la Profesora Mariela Cerrada de la Universidad de los Andes por todo su tiempo invertido y sus múltiples sugerencias realizadas que permitieron mejorar notablemente la versión final de la presente tesis.

Durante esta etapa, he contado con palabras de apoyo y motivación de parte del Profesor Jesús Rodríguez de la Universidad de Los Andes, el Profesor Félix García Padilla de la Universidad de Oriente y de la Señora Milagros Mejías, secretaria del Doctorado en Ciencias Aplicada.

Finalmente, no puedo terminar sin expresar mi más profundo agradecimiento a dos personas que han hecho posible que me supere día a día en lo personal y en lo académico, a mi madre Gladys y mi Padre Carmelo, por enseñarme los valores de honestidad, esfuerzo, perseverancia y humildad. Definitivamente no tengo palabras para expresar la inmensa ayuda que me han brindado durante el desarrollo de la tesis, especialmente en estos tres últimos años. También agradezco a mi hermana Adriana por sus palabras de apoyo en la etapa de culminación de este trabajo. Del mismo modo no olvido su preocupación en todo el recorrido.

www.bdigital.ula.ve

Resumen

En los grandes complejos industriales que procesan productos a gran escala, es necesario cumplir con los volúmenes de producción planificados y con ciertos niveles de calidad adecuados. Tales complejos industriales se caracterizan por tener múltiples sistemas de control distribuidos en cada una de las distintas fases del procesamiento del producto. Para poder cumplir con el objetivo de producción, se requiere que cada sistema de control local funcione de una manera autónoma, coherente y sub-óptima.

Esta tesis tiene como objetivo establecer una propuesta metodológica para el diseño un sistema de supervisión local que pueda ser aplicado en sistemas complejos con la finalidad de detectar eventos que generen transiciones entre estados de operación para luego realizar cambios en la consigna de operación y la respectiva adaptación en línea del controlador del proceso.

Es importante resaltar, la evaluación de varias alternativas para la identificación de modelos difusos invertibles y de alta interpretabilidad y exactitud a partir de datos de entrada y salida de los procesos de separación gas líquido y calentamiento de crudo; procesos del área del procesamiento de petróleo que han sido usados como referencia en los casos de estudios a lo largo del desarrollo de esta tesis. Dentro de las alternativas para la identificación de modelos difusos de alta interpretabilidad se evaluaron las técnicas de agrupamiento difuso Fuzzy C Means y Gustafson-Kessel, técnicas neurodifusas, específicamente el algoritmo ANFIS (Adaptive Network Based Fuzzy Inference System) y optimización de modelos difusos a través de algoritmos genéticos.

Se aplicó el método de inversión analítico para modelos difusos y tras la revisión de varias alternativas para la implementación de un sistema de control basado en modelos, el controlador difuso obtenido se implemento bajo el esquema de control por modelo interno. Los resultados de las simulaciones computacionales muestran de manera empírica que este tipo de controlador es estable y se garantiza un error de estado estático nulo.

Finalmente, se propuso una arquitectura de supervisión local conformada por un detector de eventos, un modelo cualitativo del proceso basado en un autómata de estado finito y un sistema de toma de decisión que cambia la consigna de operación y ajusta de manera iterativa los parámetros escalares del controlador para minimizar el error de estado estático. La propuesta metodológica fue evaluada en un separador de producción de petróleo y los resultados de las simulaciones computacionales muestran el buen desempeño del sistema supervisor.

www.bdigital.ula.ve

Abstract

In production complexes, where a considerable amount of products are processed, it is necessary to fulfill both with the scheduled production volumes and with the adequate level of quality. Such production complexes are characterized by the presence of multiple distributed control systems in each of the different phases of the processing of the products. In order to fulfill with the production objective, it is required that each local control system works in an autonomous, coherent and sub-optimum manner.

The purpose of this thesis is to establish a methodological proposal for the design of a local supervision system that can be applied to complex systems so that it can detect events that generates transitions between operation states and thereby changing the setpoint and adapting the controller of the process on line.

It is important to highlight some previous evaluation of different alternatives for identifying invertible, high interpretability an exact fuzzy models based on input-output data applied to a production separator and a fired heater; these particular processes have been used to validate the design of the supervisory system. Fuzzy clustering techniques such as Fuzzy C Means and Gustaffson - Kessel algorithm, neurofuzzy techniques, specifically the algorithm ANFIS (Adaptive Network Based Fuzzy Inference System) and the optimization of fuzzy models through the use of genetic algorithms were the alternatives evaluated for the identification of fuzzy models of high interpretability.

The analytical inversion method for fuzzy models was applied and after some exhaustive revision of different strategies related to model based control system, the fuzzy controller obtained was implemented under the internal model control (ICM) scheme. The results of the computational simulation show in an empirical manner that this sort of controller is stable and has a minimum static state error.

Finally, it was proposed a local supervision architecture composed of an event detector, a qualitative process model based on a finite state machine and the deci-

sion system whose task is to adjust the setpoint and adapt the scalar parameters of the controller in order to minimize the static state error. The methodological proposal was evaluated in a production separator and the results of the computational simulations show the overall good performance of the supervisor system.

www.bdigital.ula.ve

Índice general

1. INTRODUCCIÓN	1
1.1. Motivación	1
1.2. Objetivos de la tesis	3
1.3. Estructura y contenido de la tesis	3
2. MODELACIÓN DIFUSA Y SISTEMAS MULTIAGENTE	7
2.1. Introducción	7
2.2. Modelos difusos basados en reglas	8
2.2.1. Modelo difuso lingüístico (Mamdani)	9
2.2.2. Modelo difuso relacional	11
2.2.3. Modelo difuso unitario (singleton)	13
2.2.4. Modelo difuso Takagi-Sugeno	15
2.3. Modelación difusa de sistemas dinámicos	17
2.4. Construcción de modelos difusos	18
2.4.1. Elementos a identificar	19
2.4.2. Preprocesamiento y extracción de datos	19
2.4.3. Métodos de aprendizaje	20
2.4.3.1. Métodos de mínimos cuadrados	21
2.4.3.2. Métodos del gradiente	22
2.5. Sistemas multiagente	23
2.5.1. Agente y sistemas multiagente	23
2.5.2. Interacciones entre agentes en un sistema multiagente	25
2.5.2.1. Técnicas de interacción entre agentes	26
2.5.2.2. Colaboración, coordinación y comunicación entre agentes	26
2.5.2.3. Protocolos de comunicación	29
2.5.3. Aprendizaje y razonamiento	30
2.5.4. Modelado de un sistema multiagente	30
2.5.5. Ontología y gestión del conocimiento	31
2.5.6. Metodologías orientadas a simular sistemas multiagentes	32
2.6. Conclusiones	33

3. IDENTIFICACIÓN DE MODELOS DIFUSOS A PARTIR DE DATOS DE ENTRADA-SALIDA	35
3.1. Introducción	35
3.2. Técnicas de agrupamiento	36
3.2.1. Algoritmo de agrupamiento difuso Fuzzy C-Means (FCM)	40
3.2.2. Algoritmo de Gustafson-Kessel (GK)	45
3.3. Consideraciones teóricas para la obtención de modelos difusos usando técnicas de agrupamiento	46
3.3.1. Diseño de experimentos	48
3.3.2. Selección de la estructura	48
3.3.3. Agrupación de datos	49
3.3.4. Selección del número de clases	50
3.3.5. Modelo difuso inicial	52
3.3.6. Índices de desempeño	55
3.4. Modelación neurodifusa	56
3.4.1. Generalidades	56
3.4.2. Modelo ANFIS	58
3.5. Casos de estudios	61
3.5.1. Separador de producción	61
3.5.1.1. Resultados de la aplicación del algoritmo GK	62
3.5.1.2. Resultados de la aplicación del algoritmo FCM	65
3.5.1.3. Resultados de la aplicación del algoritmo ANFIS	67
3.5.2. Análisis de interpretabilidad de los modelos difusos identificados del separador	68
3.5.3. Hornos de fuego directo	70
3.5.3.1. Resultados de la aplicación del algoritmo GK	71
3.5.3.2. Resultados de la aplicación del algoritmo FCM	73
3.5.3.3. Resultados de la aplicación del algoritmo ANFIS	76
3.5.4. Análisis de interpretabilidad de los modelos difusos identificados del horno	78
3.6. Propuesta metodológica para la identificación de modelos difusos de alta interpretabilidad	78
3.6.1. Determinación del modelo difuso unitario para el separador de producción	80
3.6.2. Determinación del modelo difuso unitario para el horno de producción	83
3.7. Algoritmo ANFIS y algoritmos genéticos: análisis comparativo	85
3.8. Conclusiones	86

4. CONTROL BASADO EN MODELOS DIFUSOS	89
4.1. Introducción	89
4.2. Estrategias de control basados en modelos difusos	90
4.2.1. Control por modelo inverso	91
4.2.2. Control adaptativo por modelo de referencia	92
4.2.3. Control por modelo interno (CMI)	93
4.2.4. Control predictivo basado en modelo (MPC)	95
4.3. Inversión de modelos difusos	100
4.3.1. Inversión de modelos difusos unitarios	102
4.3.2. Inversión de modelos Takagi - Sugeno	106
4.4. Modelación matemática de procesos de petróleo	107
4.4.1. Modelación matemática del separador horizontal de producción	108
4.4.2. Modelación matemática del horno de fuego directo	111
4.5. Implementación del control basado en modelos difusos usando CMI para el separador de producción	115
4.6. Implementación del control basado en modelos difusos usando CMI para el horno de producción	118
4.7. Conclusiones	122
5. PROPUESTA METODOLÓGICA DE UN SISTEMA DE SUPERVISIÓN	123
5.1. Introducción	123
5.2. Funciones básicas de los sistemas supervisores	124
5.2.1. Control supervisorio difuso	125
5.2.2. Detección y diagnóstico de fallas	127
5.2.3. Tolerancia a fallas	128
5.3. Enfoque de sistemas holónicos para fabricación	129
5.4. Propuesta metodológica para el diseño de un sistema de supervisión	131
5.4.1. Módulo de medición	133
5.4.2. Detector de eventos	133
5.4.3. Modelo cualitativo del proceso	136
5.4.4. Sistema de toma de decisión	137
5.4.5. Módulo de coordinación	141
5.5. Validación de la propuesta	142
5.6. Conclusiones	150
6. Conclusiones, Trabajos futuros y Publicaciones	153
6.1. Sumario y Conclusiones	153
6.2. Trabajos futuros	154
6.3. Publicaciones relacionadas a esta tesis	155

ÍNDICE GENERAL

A. Métodos de mínimos cuadrados	159
B. Aplicación de técnicas neurodifusas para la identificación de modelos difusos	165
C. Negociación entre agentes	169
C.1. Introducción	169
C.2. Negociación entre agentes	169
C.2.1. Fundamentos de un protocolo de negociación desde el punto de vista de la teoría de juegos	171
C.2.2. El Entorno de negociación	172
C.2.3. Las reglas de encuentro	172
C.2.4. La Situación de conflicto	175
C.2.5. Estrategia e información sobre la negociación	175
C.2.6. Protocolos de negociación basados en la teoría de juegos	176
C.3. Propuesta de un Protocolo de Negociación Para un Entorno Multiagente en Procesos Continuos	177
C.4. Entorno multiagente para la supervisión de procesos continuos	180
C.4.1. Modelo de la organización	181
C.4.2. Modelo del entorno	183
C.4.3. Modelo de tareas y objetivos	184
C.4.4. Modelo del agente	185
C.4.5. Modelo de interacciones	187
C.5. Implementación del sistema multiagente	188
D. Implementación del algoritmo de negociación en controladores lógicos programables	191

Índice de figuras

2.1. Modelo difuso relacional	12
2.2. Modelo unitario con funciones de membresías triangulares y trapezoidales	14
2.3. Modelo difuso Takagi-Sugeno	15
2.4. Modelo difuso afín Takagi-Sugeno	16
2.5. Sistema difuso genérico con unidades de difusificación y desdifusificación con filtros dinámicos externos	18
2.6. Algoritmo de coordinación de agentes	28
3.1. Diferentes posibilidades de clases en R^2	37
3.2. Normas usadas en agrupamiento difuso	44
3.3. Esquema sistematizado de identificación de modelos difusos	47
3.4. Función de pertenencia proyectada y aproximaciones	53
3.5. Arquitectura del sistema ANFIS	59
3.6. Componentes principales de un separador de producción horizontal	62
3.7. Datos medidos de entrada-salida del separador de producción	63
3.8. Agrupación de datos del proceso de separación usando el algoritmo Gustaffson-Kessel	63
3.9. Proceso de separación: funciones de pertenencia de $y(\tau)$ y $u(\tau)$ (algoritmo GK)	64
3.10. Modelos locales detectados (algoritmo GK).	65
3.11. Validación del modelo difuso del separador de producción (algoritmo GK).	65
3.12. Agrupación de datos usando el algoritmo Fuzzy C Means	66
3.13. Proceso de separación: funciones de pertenencia de $y(\tau)$ y $u(\tau)$ (algoritmo FCM)	66
3.14. Validación del modelo difuso del separador de producción (Algoritmo FCM).	67
3.15. Comportamiento de la señal de Error (modelación ANFIS del separador)	68

ÍNDICE DE FIGURAS

3.16. Funciones de membresía del antecedente (modelación ANFIS del separador)	69
3.17. Validación del modelo difuso del separador de producción (técnicas neurodifusas)	69
3.18. Horno de fuego directo	71
3.19. Datos medidos de entrada-salida del horno de fuego directo	72
3.20. Agrupación de datos del proceso de calentamiento usando el algoritmo Gustaffson-Kessel	73
3.21. Proceso de calentamiento: funciones de pertenencia de $y(\tau)$ y $u(\tau)$ (algoritmo GK)	73
3.22. Validación del modelo difuso del horno de producción (algoritmo GK).	74
3.23. Agrupación de datos del proceso de calentamiento usando el algoritmo Fuzzy C Means	74
3.24. Proceso de calentamiento: funciones de pertenencia de $y(\tau)$ y $u(\tau)$ (Algoritmo FCM)	75
3.25. Validación del modelo difuso del horno de producción (algoritmo FCM)	75
3.26. Comportamiento de la señal de Error (modelación ANFIS del horno) 76	
3.27. Funciones de membresía del antecedente (modelación ANFIS del horno)	77
3.28. Validación del modelo difuso del horno de producción (técnicas neurodifusas)	77
3.29. Estructura del cromosoma, funciones de membresía y consecuentes unitarios	79
3.30. Parámetros a optimizar del modelo difuso unitario (separador)	81
3.31. Parámetros optimizados del modelo difuso unitario del separador	83
3.32. Validación del modelo difuso unitario del separador	84
3.33. Parámetros del modelo difuso unitario del horno	84
3.34. Validación del modelo difuso unitario del horno	85
4.1. Esquema de control por modelo inverso	91
4.2. Esquema de control adaptativo por modelo de referencia	92
4.3. Esquema de control por modelo interno	93
4.4. Esquema de control por modelo interno adaptativo	96
4.5. Estrategia de los controladores predictivos basado en modelos	97
4.6. Controlador MPC con observador de estado	99
4.7. Esquema de control lazo abierto y lazo cerrado	100
4.8. Aprendizaje inverso directo	101
4.9. Aprendizaje inverso especializado	102
4.10. Modelo difuso y su inversa	103

4.11. Ejemplo de condiciones no cumplidas de invertibilidad	105
4.12. Particiones de los conjuntos difusos $C_j(\tau)$	105
4.13. Dimensiones del separador horizontal de producción	110
4.14. Modelo matemático aproximado del separador horizontal de producción	111
4.15. Esquemático del funcionamiento de un horno de fuego directo	111
4.16. Configuración del circuito eléctrico	112
4.17. Desempeño del sistema de control ICM para el separador	117
4.18. Comportamiento de la válvula de control del separador	117
4.19. Desempeño del sistema de control ICM para el horno	120
4.20. Comportamiento de la válvula de control de flujo de gas del horno	121
5.1. Esquema de control supervisorio difuso	126
5.2. Arquitectura del sistema supervisor para procesos continuos	132
5.3. Ejemplo de mapeo directo de variable continua a discreta	134
5.4. Detección de eventos	135
5.5. Modelo cualitativo general de un proceso	137
5.6. Sistema de toma de decisión	138
5.7. Estrategias para mejorar el error de estado estático	140
5.8. Modelo cualitativo del proceso	143
5.9. Comportamiento del sistema supervisor local	146
5.10. Salida del detector de eventos	147
5.11. Salida de la máquina de estados finitos	147
5.12. Salida del controlador	148
5.13. Flujo de entrada del separador	148
5.14. Presión del separador	148
5.15. Índice de desempeño del controlador	149
5.16. Modificación de los núcleos de las funciones de membresía de $u(\tau)$	149
A.1. Mapeo de Función con tres pares de datos	159
C.1. Funciones para el calculo de $\phi_r^a(t)$	174
C.2. Protocolo de negociación	178
C.3. Modelo de la organización de una estación de descarga de petróleo	182
C.4. Arquitectura del sistema multiagente propuesto	183
C.5. Entorno del sistema multiagente	184
C.6. Modelo de objetivos agente especializado	185
C.7. Salida agente separador 1 y 2	189
C.8. Salida agente horno 1	189
C.9. Simulación de un proceso de negociación	190
D.1. Rung que activa el proceso de negociación (Agente A)	192

ÍNDICE DE FIGURAS

D.2. Rungs que establece las condiciones de negociación y envía primera oferta	192
D.3. Rungs que determina si la oferta es aceptada o si evalúa una contraoferta	193
D.4. Rungs que determina si acepta contraoferta o envía una nueva oferta	193

www.bdigital.ula.ve

Índice de tablas

3.1. Índices de desempeño de los modelos difusos del separador	70
3.2. Índices de desempeño de los modelos difusos obtenidos para el horno de fuego directo	76
4.1. Comparación de los parámetros de un circuito eléctrico y un horno	113
4.2. Variaciones de los núcleos $c_j(\tau)$ de C_j (separador)	118
4.3. Variaciones de los núcleos $c_j(\tau)$ de C_j (horno)	121

www.bdigital.ula.ve

Capítulo 1

INTRODUCCIÓN

En este capítulo se describen las motivaciones que han conducido al desarrollo de esta tesis doctoral, para luego enunciar los objetivos planteados y presentar la estructura con el respectivo resumen de cada capítulo.

1.1. Motivación

Las grandes corporaciones o empresas más competitivas a nivel mundial vinculadas con el área de procesamiento del petróleo, el gas, la ferrominera, la petroquímica, etc., por citar tan sólo algunos ejemplos, han logrado realizar inversiones muy significativas en sistemas automáticos de control en cada una de las cadenas de sus respectivos procesos de producción, con la finalidad de incrementar por una parte los niveles de seguridad dentro de sus operaciones y, por otra parte, lograr procesar o producir sus respectivos productos con la mayor calidad posible, minimizando en lo posible el uso de recursos en términos de consumo de energía y reduciendo las labores de mantenimiento no programadas.

La actual realidad en muchas de estas empresas que tienen sus procesos automatizados se puede interpretar en términos de la presencia e implantación de múltiples lazos de control, que normalmente interactúan unos con otros, en donde cada controlador opera en un contexto local bien definido. Adicionalmente, los operadores de las salas de control que acceden a la información sobre sus procesos a través de un software especializado denominado SCADA (Sistema de Control y Adquisición de Datos), toman las decisiones en el corto plazo que no son las más idóneas. En esencia, los operadores se basan en su experiencia y conocimientos, expresados casi siempre en términos cualitativos, para llevar a cabo sus decisiones de realizar cambios en las consignas en algunos de los procesos o detectar algún tipo de anomalía en el funcionamiento de un determinado sensor, algún controlador o en el proceso mismo.

La supervisión inteligente de procesos industriales es un área de intensos estudios y se ha convertido en una de las principales aplicaciones de la inteligencia artificial que mayor trascendencia ha logrado. De hecho, se han realizado una gran variedad de esfuerzos para tratar de emular las funciones de supervisión de los operadores de la sala de control, es decir, se han orientado estrategias para concebir, diseñar e implementar sistemas de control con el mayor grado de autonomía posible. En este sentido, existen al menos tres claras orientaciones: algunos esquemas de supervisión buscan como objetivo la detección y el diagnóstico de fallas de los sensores de campo, otros esquemas contemplan realizar adaptaciones en línea sobre el controlador del proceso e inclusive acciones específicas sobre el proceso directamente y un tercer esquema involucra el hecho de mejorar su base de conocimiento o simplemente tener la capacidad de aprender. Entre las alternativas que han enmarcado habitualmente el desarrollo de sistemas de control supervisorio tenemos: la lógica difusa, los sistemas expertos, los sistemas multiagentes como tecnología de implementación de sistemas distribuidos, sistemas de eventos discretos, las redes de Petri, etc. En definitiva, la hibridización de algunas de las alternativas mencionadas, es lo que ha permitido desarrollar múltiples propuestas de esquemas de supervisión aplicado a procesos continuos.

En el marco de la presente de tesis, se plantea una propuesta metodológica que permita diseñar un sistema de supervisión para procesos complejos caracterizados por la presencia de incertidumbre, no linealidades y distintos estados de operación. Para ello se contempla, en primer lugar, la identificación de modelos difusos de alta interpretabilidad basado en datos de entrada - salida tomando en cuenta una serie de restricciones que garanticen el cálculo de la inversa para la obtención de controladores basado en modelos difusos. Luego se evalúa el desempeño del controlador bajo el esquema de control por modelo interno aplicado a sistemas complejos.

El nivel de supervisión contempla un detector de eventos para la captura de eventos relacionados con incremento de la cantidad o calidad de producción, el modelo cualitativo del proceso representado a través de una máquina de estados finitos y el sistema de toma de decisión que se encarga de ajustar la consigna de operación y de realizar adaptaciones en línea sobre los controladores de tal manera de garantizar en el proceso una operación continua, coherente y sub-óptima.

Una opción factible que puede conducir a una solución idónea al planteamiento anterior, estaría orientada hacia el área de los sistemas de inferencia difusa. La lógica difusa permite entre otras cosas tratar la imprecisión en cuanto a la información obtenida de los sensores de campo, que siempre se encuentra presente

en la mayoría de los procesos industriales, y es por ello que se hará énfasis en la aplicación de la modelación difusa, los esquemas de control basados en modelos difusos y la adaptación en línea de controladores difusos. Adicionalmente, el enfoque de sistemas de eventos discretos y algunas ideas claves sobre la modelación multirresolucional permiten consolidar los elementos necesarios para establecer una propuesta metodológica de un sistema de supervisión.

1.2. Objetivos de la tesis

El objetivo principal de esta tesis es establecer una propuesta metodológica para el diseño de un sistema de supervisión aplicado a sistemas complejos caracterizados por la presencia de no linealidades y distintos estados de operación que permita mantener desempeños idóneos sobre el controlador del proceso.

Se propone como alternativa metodológica plantear el uso de la modelación difusa y de las técnicas de control basado en modelos difusos en el nivel de regulación y los principios de los sistemas de eventos discretos y los sistemas multirresolucionales en el nivel de supervisión. En general, para lograr este objetivo, será necesario cumplir con los siguientes objetivos específicos:

- Identificar modelos difusos de alta interpretabilidad y exactitud en base a datos de entrada y salida. Se evaluarán tres alternativas al respecto: las técnicas de agrupamiento difuso, la modelación neurodifusa ANFIS y el uso de algoritmos genéticos.
- Diseñar controladores basados en modelos difusos.
- Establecer una propuesta metodológica para el diseño de un sistema de supervisión aplicado a sistemas complejos que permita detectar eventos, cambios de estados de operación, ajuste de consigna de operación y adaptación en línea del controlador del proceso.
- Validar la propuesta metodológica de supervisión.

1.3. Estructura y contenido de la tesis

Después de la introducción que se desarrolla en este primer capítulo, para cumplir con los objetivos establecidos en la presente tesis doctoral, se ha dividido

la misma en seis capítulos, los cuales son descritos brevemente a continuación.

Capítulo 2: Modelación difusa y sistemas multiagente

Este capítulo contiene el marco teórico de la presente tesis y se revisa por una parte los diferentes tipos de modelos difusos basados en reglas que pueden ser usados para modelar un sistema dinámico no lineal; resaltando sus características de interpretabilidad y haciendo énfasis general en los modelos de tipo Takagi-Sugeno (TS). Se realiza una previa revisión de los sistemas difusos lingüísticos tipo Mamdani y de tipo relacional y el sistema difuso singleton. Con respecto a los métodos de aprendizaje que pueden ser aplicados a un modelo difuso TS para realizar adaptaciones en línea, se revisarán el método del gradiente y la técnica de mínimos cuadrados.

Por otra parte, en cuanto a la teoría de sistemas multiagente, se revisará por una parte el concepto y las características básicas de un agente y por otra parte los términos asociados a este tema tales como la colaboración, la cooperación, la coordinación, protocolos de comunicación, ontología, entre otros.

Capítulo 3: Identificación de modelos difusos a partir de datos de entrada-salida

En este capítulo se revisa el estado del arte de las técnicas de agrupamiento difuso clásicas como lo son el algoritmo Fuzzy C Means (Media de las clases difusas) y el algoritmo Gustafson Kessel (GK). Seguidamente se presenta de manera formal, un método sistematizado de identificación difusa usando técnicas de agrupamiento mostrando aspectos tales como la selección de las variables entrada - salida, la determinación del número de clases, evaluación de índices de desempeño y el modelo difuso inicial. Por otra parte, se describen los sistemas neurodifusos (con especial énfasis en los sistemas ANFIS) como una alternativa para identificar modelos difusos Takagi-Sugeno de primer orden a partir de datos entrada-salida. Posteriormente, las técnicas de agrupamiento difuso y los sistemas ANFIS se aplican en la identificación de modelos difusos usando datos reales de entrada y salida de un separador de producción y un horno para calentamiento de crudo asociado a los procesos de producción de petróleo. Finalmente, se presenta una tercera alternativa para identificar modelos difusos singleton (Takagi-Sugeno de orden cero) invertibles, de alta interpretabilidad y exactitud usando algoritmos genéticos en base a los mismos casos de estudio mencionados.

Capítulo 4: Control basado en modelos difusos

En este capítulo se estudia todo lo relacionado a la inversión de modelos difusos con la finalidad de diseñar controladores. Luego se realiza una revisión exhaustiva sobre las estrategias sobre control basado en modelos difusos. En particular, se hará énfasis en el esquema CMI (Control por Modelo Interno) en donde, basado en el cálculo analítico de la inversa de un modelo difuso Takagi-Sugeno de orden cero, se determinarán los controladores para los casos de estudios del capítulo anterior. Previamente se hará un estudio fenomenológico tanto del separador de producción como del horno de calentamiento con la finalidad de obtener modelos matemáticos que permitan implementar y analizar el desempeño del controlador en el esquema de control por modelo interno.

Capítulo 5: Propuesta metodológica de un sistema de supervisión

En este capítulo se plantea una propuesta metodológica que permita diseñar sistema de supervisión para procesos complejos. Se propone una arquitectura de supervisión local en la cual se explicará cada uno de los componentes tales como detector de eventos, modelo cualitativo del proceso. En particular, se hará una propuesta de ajuste de controladores difusos unitarios en línea que permite mejorar el error de estado estático.

Capítulo 6: Conclusiones, Futuros Trabajos y Publicaciones

En el capítulo que cierra la tesis, se resumirán las conclusiones que se han obtenido en el desarrollo del trabajo, resaltando las diferentes aportaciones. El capítulo concluye con una portada de preguntas abiertas donde se concretan una serie de líneas de trabajos que sugiere la línea de investigación de la tesis. Finalmente, se presenta un resumen de las publicaciones presentadas tanto en congresos internacionales y nacionales así como las publicaciones en revistas arbitradas relacionadas con esta tesis.

Capítulo 2

MODELACIÓN DIFUSA Y SISTEMAS MULTIAGENTE

2.1. Introducción

El desarrollo de modelos matemáticos es fundamental en muchas disciplinas de la ingeniería y la ciencia. Los modelos son usados para múltiples propósitos tales como simulación, diseño y análisis de sistemas, la monitorización y supervisión entre otras aplicaciones. El esquema tradicional de modelación de procesos físicos se basa en el conocimiento preciso de un determinado proceso a través de la aplicación de leyes físicas y / o químicas, que bajo un tratamiento matemático adecuado, permite la obtención de un modelo para un determinado propósito.

Sin embargo, existen procesos para los cuales la obtención de modelos matemáticos es sumamente difícil y laborioso de lograr. Más aún, es importante resaltar que existe un número significativo de procesos cuyo conocimiento puede ser expresado en términos cualitativos e imprecisos y como tales no pueden ser traducidos en un esquema matemático basado en ecuaciones diferenciales o algebraicas. Métodos formales para incorporar tal información han sido desarrollados para obtención de modelos y controladores. Dichos métodos se basan en el uso del lenguaje natural, reglas, redes semánticas y modelos cualitativos como tal (Pedrycz 1990)(Sugeno and Yasukawa 1993)(Yager and Filev 1994).

Entre las técnicas de modelación pertenecientes a la inteligencia computacional, la modelación difusa es una de las alternativas más usadas en virtud de su flexibilidad al permitir la incorporación de una manera estructurada del conocimiento experto en forma de reglas de producción y la posibilidad de incorporar conocimiento proveniente de datos del proceso. De hecho, una de las ventajas de la modelación difusa es su capacidad de representar sistemas dinámi-

cos no lineales, por ende constituye una alternativa útil para aquellos procesos que son difíciles de modelar de manera fenomenológica y específicamente, donde la experiencia humana esté disponible con la suficiente calidad aportando conocimiento sobre el sistema, como en el caso de los bioreactores (Rodriguez, Abreu, Gomez, and Martinez 2007). En términos computacionales, es importante destacar que los modelos difusos son estructuras matemáticas flexibles, que en analogía con las redes neuronales y las funciones de base radial, son conocidos como aproximadores funcionales universales (Ferreiro 2005)(Sousa and Kaymak 2002)(Fernandez 2008).

Por otra parte, el paradigma de agentes y sistemas multiagente constituye actualmente un área de creciente interés dentro de la inteligencia artificial, entre otras razones por ser aplicable a la resolución de problemas complejos no resueltos de manera satisfactoria mediante técnicas clásicas. Numerosas aplicaciones basadas en este nuevo paradigma vienen siendo empleadas en infinidad de áreas, tales como control de procesos, procesos de producción, control de tráfico aéreo, aplicaciones comerciales, gestión de la producción, comercio electrónico, etc. (Jennings and Woolridge 1998)

En cuanto al contenido de este capítulo, en primer lugar, se revisarán los diversos tipos de modelos difusos basados en reglas. Luego se hará una descripción concerniente a la modelación difusa de sistemas dinámicos. Seguidamente se presentará un análisis de los aspectos a considerar para la construcción de modelos difusos como son los elementos a identificar, el procesamiento y la extracción de datos y los métodos de aprendizaje. En cuanto al contenido teórico de sistemas multiagente, se describirá el concepto de agentes y sistemas multiagente, sus características, las interacciones entre agentes, el concepto de colaboración, coordinación y comunicación, la ontología de agentes, el modelado de un sistema multiagente y las metodologías de diseño de sistemas multiagentes. Finalmente, el capítulo cierra con las conclusiones.

2.2. Modelos difusos basados en reglas

Basado en la teoría de conjuntos difusos existen diversos tipos de modelos difusos. Entre ellos, cabría destacar los sistemas difusos de regresión lineal (Pasha, Razzaghnia, and Allaviranloo 2007)(Tanaka, Hayashi, and Watada 1989)(Tanaka 1987), los sistemas difusos basados en estructura de celdas (Smith and Corner 1991) (Smith and Johansen 1997) y los sistemas difusos basados en reglas(Babuska 1996). Estos últimos serán revisados en esta sección.

En los sistemas difusos basados en reglas, la relación entre las variables se

presenta por medio de reglas Si-Entonces, de la siguiente forma: Si Proposición antecedente Entonces Proposición Consecuente. La proposición del antecedente es siempre una proposición difusa del tipo \tilde{x} es A , en donde \tilde{x} es una variable lingüística y A es un término lingüístico constante. El grado de veracidad de la proposición depende de la similitud entre \tilde{x} y A . Dependiendo de la forma del consecuente existen cuatro tipos de modelos difusos basados en reglas, que se describen a continuación:

- Modelo difuso lingüístico: tanto el antecedente como el consecuente son proposiciones difusas
- Modelo difuso relacional: es una generalización del modelo difuso lingüístico, en donde la relación entre el antecedente y el consecuente es difusa.
- Modelo difuso unitario: es considerado como un caso particular de los modelos difusos lingüísticos y Takagi-Sugeno, en donde el consecuente es reducido a un conjunto difuso singleton o número real.
- Modelo difuso Takagi-Sugeno: el antecedente es una proposición difusa y el consecuente es una función real (por lo general una función lineal).

A continuación se explicará cada uno de estos tipos de modelos.

2.2.1. Modelo difuso lingüístico (Mamdani)

Este tipo de modelo es útil para capturar información cualitativa disponible en la forma de reglas Si-Entonces, expresado de la siguiente forma:

$$R_k : \text{Si } \tilde{x} \text{ es } A_k \text{ Entonces } y \text{ es } B_k, k = 1, 2, \dots, K \quad (2.1)$$

En donde, R_k denota la k -ésima regla y K es el número total de reglas. El antecedente viene expresado por $\tilde{x} \in X \subset \mathfrak{R}^n$ y representa la entrada del sistema difuso. Similarmente, $y \in Y \subset \mathfrak{R}^p$ es la variable del consecuente, que representa la salida del sistema difuso. A_k y B_k son conjuntos difusos descritos a través de las funciones de membresía $\mu_{A_k}(x) \rightarrow [0, 1]$ y $\mu_{B_k}(x) \rightarrow [0, 1]$, respectivamente. Tanto los conjuntos difusos A_k como B_k definen regiones en el espacio del antecedente X y del consecuente Y respectivamente. Las variables del antecedente normalmente se definen como un conjunto de proposiciones difusas simples para cada $x_j, j = 1, \dots, n$ del vector x y de la misma manera las variables del consecuente son divididas en proposiciones difusas $y_i, i = 1, \dots, p$. Las proposiciones difusas tienen atributos lingüísticos tales como alta temperatura, baja velocidad, alta presión, etc. Diferentes operadores lógicos difusos tales como los operadores

CAPÍTULO 2. MODELACIÓN DIFUSA Y SISTEMAS MULTIAGENTE

de conjunción, disyunción, complemento, etc., pueden ser usados para combinar las proposiciones del antecedente. El operador de conjunción es el más usado y se expresa de la siguiente forma:

$$R_k : \text{Si } x_1 \text{ es } A_{1k} \text{ y...y } x_n \text{ es } A_{nk} \text{ Entonces } y_1 \text{ es } B_{1k} \text{ y...y } y_p \text{ es } B_{pk} \quad (2.2)$$

La base de reglas $R = \{R_k | k = 1, 2, \dots, K\}$ y los conjuntos A y B constituyen la base de conocimiento del modelo lingüístico y se requiere un algoritmo que permita calcular un valor de salida dado un valor de entrada. Este algoritmo se denomina mecanismo de inferencia. Por ende, dada la base de reglas y conocidas las entradas, el mecanismo de inferencia calcula las salidas del modelo difuso. La regla composicional de inferencia permite aplicar la inferencia difusa para modelos lingüísticos (Zadeh 1973). Cada regla $R_k : X * Y \rightarrow [0, 1]$, es una relación difusa que se calcula de la siguiente forma:

$$\mu_{R_k}(x, y) = I(\mu_{A_k}(x), \mu_{B_k}(x)) \quad (2.3)$$

En donde el operador I puede ser una implicación difusa o un operador de conjunción (t-norma). La base de reglas entera es representada a través de la combinación de las K relaciones de cada regla individual R_k en una relación global R . Si, por ejemplo I , es un operador de implicación, R se obtiene al realizar una conjunción de todas las relaciones individuales R_k . Dada una entrada de la forma \tilde{x} es A y la relación R , el conjunto difuso de salida correspondiente es expresado de la siguiente forma:

$$B' = A' \circ R \quad (2.4)$$

En donde \circ denota la composición supremo estrella (sup-t). La t-norma del mínimo es la más usada para este tipo de composición y por ende la ecuación (2.4) se puede expresar de la siguiente forma:

$$\mu_{B'}(y) = \max_x \min_{x,y} (\mu_{A'}(x), \mu_R(x, y)) \quad (2.5)$$

Cuando la implicación I se selecciona como el operador de conjunción mínimo, μ_R se convierte en el mínimo de μ_{A_k} y μ_{B_k} y la regla composicional de inferencia se convierte en la inferencia de Mamdani Max-Min (Driankov, Hellendoorn, and Reinfrank 1993), la cual se puede resumir en los siguientes pasos:

- El grado de cumplimiento del antecedente es calculado para cada k regla como:

$$\beta_k = \mu_{A_{1k}}(x_1) \wedge \mu_{A_{2k}}(x_2) \wedge \dots \wedge \mu_{A_{nk}}(x_n), k = 1, \dots, K \quad (2.6)$$

- Para cada regla, la salida del conjunto difuso B_k se calcula usando la t-norma del mínimo:

$$\mu_{B'_k}(y) = \beta_k \wedge \mu_{B_k}(y) \quad (2.7)$$

- Se agregan los conjuntos difusos de salida a través de la operación del máximo:

$$\mu_{B'}(y) = \max_{k=1, \dots, K} (\mu_{B'_k}(y)) \quad (2.8)$$

En muchos casos, se requiere de una salida numérica y por ende la salida difusa debe ser desdifusificada. Existen varios métodos de desdifusificación y los más usados son el centro de gravedad (COG) y la media de máximo (MOM). En particular, para dominios de tiempo discreto de Y , el método COG para cada coordenada $y_i, i = 1, \dots, p$, calcula el centro de gravedad para el conjunto difuso B' como una suma ponderada expresada de la siguiente forma:

$$Z_{y_i}^{cog}(B') = \frac{\sum_{i=1}^{N_q} \mu_{B'}(y_q) y_{i,q}}{\sum_{i=1}^{N_q} \mu_{B'}(y_q)} \quad (2.9)$$

En donde N_q es la cardinalidad del dominio y discretizado y y_q es el q -ésimo punto discreto en la cuantización de Y . En (Sousa and Kaymak 2002) se hace una revisión exhaustiva sobre los métodos de desdifusificación.

2.2.2. Modelo difuso relacional

Un modelo difuso relacional es considerado como una extensión del modelo lingüístico difuso en donde el mapeo entre los conjuntos difusos de entrada y salida es representado a través de una relación difusa (Pedrycz 1990)(Pedrycz 1993). Si se denota a A_j y B como los terminos lingüísticos para las variables del antecedente y el consecuente respectivamente:

$$\begin{aligned} A_j &= \{A_{jk} | k = 1, 2, \dots, N_j\}, j = 1, 2, \dots, p, \\ B &= \{B_k | k = 1, 2, \dots, M\} \end{aligned} \quad (2.10)$$

La base de reglas (2.2) puede ser representada como una relación ordinaria R entre los términos lingüísticos en los antecedentes y consecuentes tal que:

$$R : A_1 \times A_2 \times \dots \times A_n \times B \rightarrow 0, 1 \quad (2.11)$$

Además, al denotar $A = A_1 \times A_2 \times \dots \times A_n$ el espacio cartesiano de los términos lingüísticos del antecedente, la expresión anterior puede ser simplificada a $R : A \times B \rightarrow \{0, 1\}$. Un modelo difuso relacional se obtiene al generalizar R como

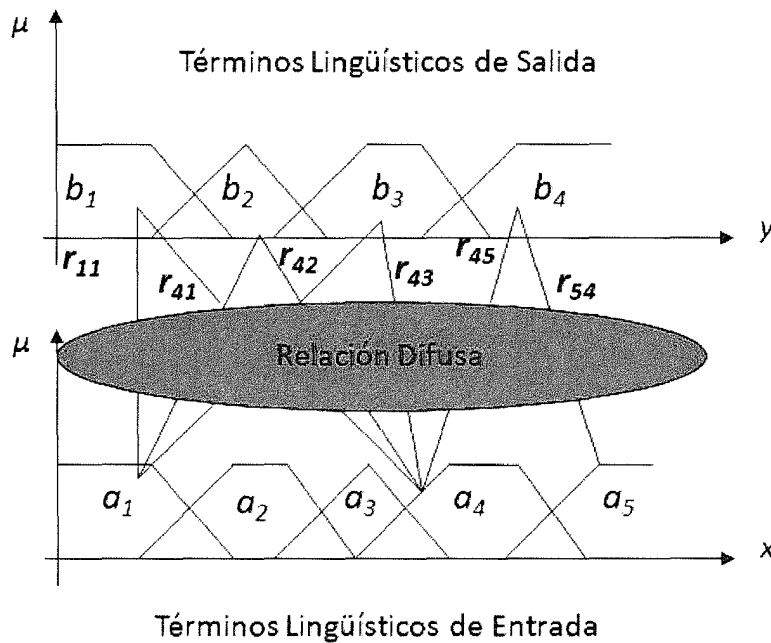


Figura 2.1: Modelo difuso relacional

una relación difusa:

$$R : A \times B \rightarrow [0, 1] \quad (2.12)$$

En este modelo, cada regla contiene todos los términos consecuentes posibles, cada uno con un factor de ponderación diferente, dado por los respectivos elementos de la relación difusa. Con esta ponderación, se puede fácilmente ajustar o sintonizar el modelo, i.e, para ajustar algún dato. En la figura 2.1 se muestra un modelo relacional difuso de una entrada y una salida.

La principal ventaja del modelo difuso relacional es que el mapeo entrada-salida puede ser fácilmente sintonizado sin tener que cambiar los conjuntos difusos del consecuente. El algoritmo de inferencia del modelo relacional se muestra a continuación:

- Se calcula el grado de cumplimiento de cada regla:

$$\beta_i = \mu_{A_{i1}}(x_1) \wedge \dots \wedge \mu_{A_{ip}}(x_p), i = 1, 2, \dots, M \quad (2.13)$$

- Se aplica la composición relacional: $w = \beta R$, dada por :

$$w_j = \max_{1 \leq i \leq k} (\beta_i \wedge r_{ij}) \quad (2.14)$$

- Se calcula la desfusificación como:

$$y = \frac{\sum_{l=1}^M w_l b_l}{\sum_{l=1}^M w_l}, w_l = \text{cog}(b_l) \quad (2.15)$$

2.2.3. Modelo difuso unitario (singleton)

Un caso especial del modelo lingüístico se obtiene cuando los consecuentes B_i son reducidos a conjuntos unitarios. Los conjuntos unitarios se representan a través de números reales b_i , generandose una base de reglas de la siguiente forma:

$$R_i : \text{Si } x \text{ es } A_i \text{ Entonces } y = b_i, i = 1, 2, \dots, K \quad (2.16)$$

Este es el modelo difuso unitario conocido en la literatura inglesa como modelo difuso "singleton". El método de desfusificación COG para este tipo de modelo, se reduce al método de la media difusa expresado de la siguiente forma:

$$y = \frac{\sum_{i=1}^K \beta_i b_i}{\sum_{i=1}^K \beta_i} \quad (2.17)$$

En donde β_i es el grado de cumplimiento de la i -ésima regla y b_i es un número real. Es importante destacar que la desfusificación depende del número de reglas K y no del número de conjuntos difusos para una cierta salida $y_i, i = 1, \dots, p$. En contraste con el modelo lingüístico Mamdani, los parámetros consecuentes b_i pueden ser fácilmente estimados a partir de la medición de datos usando técnicas de mínimos cuadrados.

Además, el modelo difuso unitario pertenece a una clase de aproximadores de funciones, denominados expansión de funciones base, que puede ser expresado de la siguiente forma:

$$y = \sum_{i=1}^K \Phi^i(x) b_i, \Phi^i(x) = \frac{\beta_i}{\sum_{i=1}^K \beta_i} \quad (2.18)$$

En donde $\Phi^i(x)$ es la función de base radial dado por el grado de cumplimiento normalizado de cada regla i (Sousa and Kaymak 2002). Vale la pena destacar que la mayoría de las estructuras usadas en sistemas no lineales tales como redes neuronales y funciones de base radial pertenecen a esta clase de sistemas (Freidman 1991).

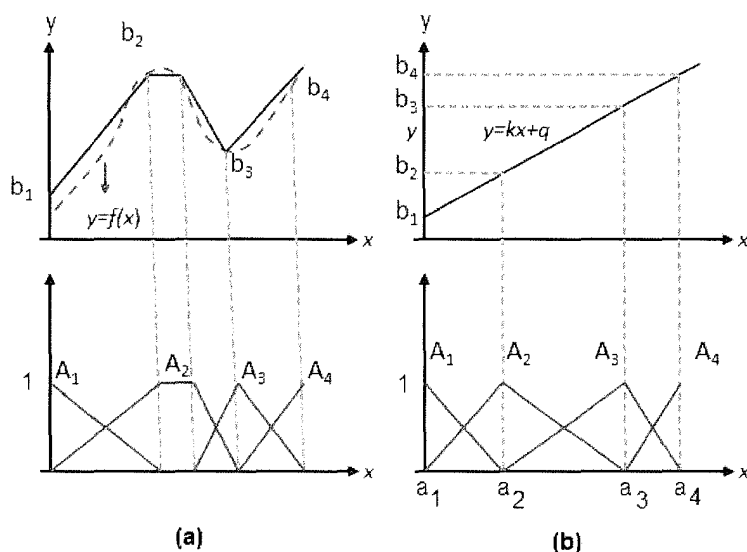


Figura 2.2: Modelo unitario con funciones de membresías triangulares y trapezoidales

En particular, los modelos difusos unitarios tienen una característica de gran importancia. Cuando las funciones de membresía de los antecedentes son de tipo triangular y trapezoidal tal y como se muestra en la figura 2.2a y para cada variable del antecedente la suma del grado de activación del conjunto difuso o los conjuntos difusos implicados es igual a 1 y adicionalmente la t-norma del producto es usada para representar el conectivo “y” lógico en los antecedentes de las reglas, entonces se puede obtener una interpolación multilineal entre los consecuentes de las reglas (Sousa and Kaymak 2002)(Babuska 1996). Bajo ciertas condiciones muy particulares, el modelo difuso unitario puede ser exactamente invertido y usado como controlador (Babuska 1996).

El mapeo de entrada-salida del modelo unitario es multi-lineal tal y como se puede apreciar en la figura 2.2a. Claramente, un modelo difuso unitario puede representar cualquier mapeo lineal de la forma:

$$y = K^T x + q = \sum_{k=1}^p k_j x_j + q \quad (2.19)$$

En este caso particular, de acuerdo a la figura 2.2b, todas las funciones de membresías deben ser triangulares, en donde p es la cantidad de salida del modelo difuso unitario, k representa la pendiente de la recta usada para la interpolación

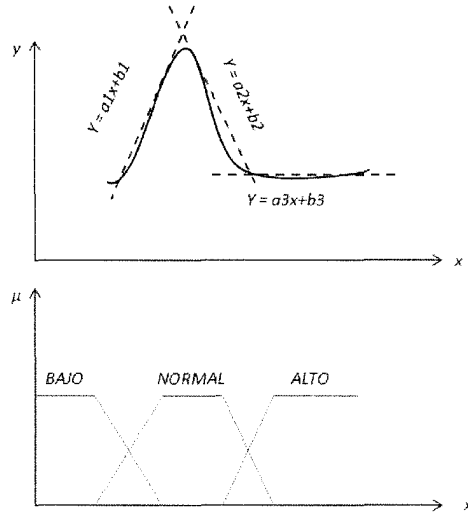


Figura 2.3: Modelo difuso Takagi-Sugeno

y q es el término de “offset” de dicha recta. Los consecuentes unitarios se calculan al evaluar el mapeo deseado por los núcleos a_{ij} de los conjuntos difusos del antecedente A_{ij} a través de la siguiente expresión:

$$b_i = \sum_{j=1}^p k_j a_{ij} + q \quad (2.20)$$

2.2.4. Modelo difuso Takagi-Sugeno

El modelo difuso Takagi-Sugeno (TS) se diferencia del modelo lingüístico Mamdani en que presenta funciones reales en su consecuente en lugar de una proposición difusa (Abonji 2007). Su base de reglas se puede expresar de la siguiente forma:

$$R_i : \text{Si } x_1 \text{ es } A_{i1} \text{ y } \dots \text{ y } x_s \text{ es } A_{is} \text{ Entonces } y_i = f_i(x_1, x_2, \dots, x_s); i = 1, 2, \dots, c \quad (2.21)$$

Por ende, este tipo de modelo puede ser visto como una combinación de la modelación lingüística y la regresión matemática, en el sentido de que los antecedentes describen regiones difusas en el espacio de entrada cuyas funciones consecuentes sean válidas.

Un ejemplo sencillo y útil de este modelo es el caso en el que se toma la función consecuente lineal en sus parámetros tal y como se observa en la figura 2.3, obteniéndose la siguiente base de reglas:

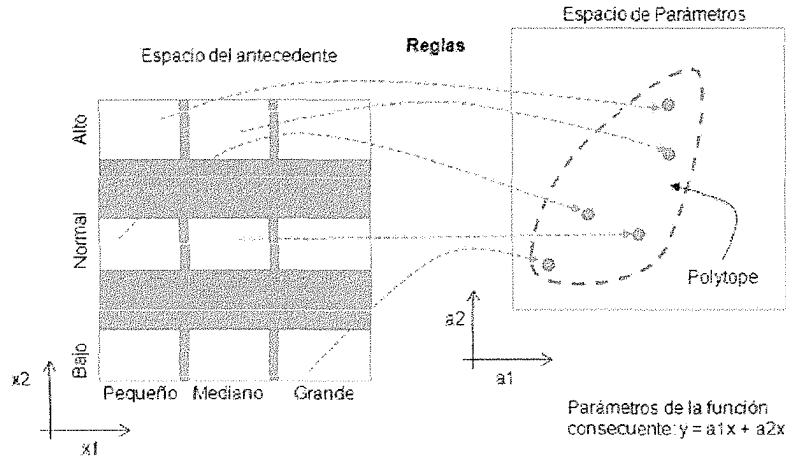


Figura 2.4: Modelo difuso afin Takagi-Sugeno

$$R_i : \text{Si } x \text{ es } A_i \text{ Entonces } y_i = a_i^T x + b_i \quad (2.22)$$

En donde a_i es un vector de parámetros y b_i es un escalar. Este modelo se le denomina modelo afin de Takagi-Sugeno y proporciona el marco adecuado para la descomposición de un sistema no lineal en modelos lineales localmente válidos, muy apropiado para aplicar las técnicas de control clásicas (Diez 2003).

La inferencia y desdifusificación de los modelos TS, una vez determinado el grado de cumplimiento $\beta_i(x)$ de cada regla y dado los datos x , se reduce a la expresión algebraica:

$$f(x) = \frac{\sum_{i=1}^c \beta_i(x) f_i(x)}{\sum_{i=1}^c \beta_i(x)} = \frac{\sum_{i=1}^c \beta_i(x) (a_i^T x + b_i)}{\sum_{i=1}^c \beta_i(x)} \quad (2.23)$$

Cuando los conjuntos difusos del antecedente definen regiones que se solapan en el espacio del antecedente y los parámetros a_i y b_i corresponden a una linealización local de una función no lineal, el modelo TS puede ser considerado como una aproximación lineal a trozos (piece-wise) de tal función (Babuska 1996). Esta interpretación se puede apreciar en la figura 2.4.

Por ende el modelo difuso TS puede ser considerado como un sistema cuasi-lineal (sistema lineal con parámetros de entrada dependiente). Para apreciar esto mejor, denotemos el grado de cumplimiento normalizado como:

$$\gamma_i(x) = \frac{\beta_i(x)}{\sum_{j=1}^k \beta_j(x)} \quad (2.24)$$

A partir de esta ecuación se escribe $\beta_i(x)$ explícitamente como una función de x para poder expresar al modelo TS como un modelo cuasi-lineal de la siguiente forma:

$$y = \left(\sum_{i=1}^K \gamma_i(\mathbf{x}) a_i^T \right) \mathbf{x} + \sum_{i=1}^K \gamma_i(x) b_i = a^T(\mathbf{x}) \mathbf{x} + b(x) \quad (2.25)$$

Los parámetros $a(x)$, $b(x)$ son combinaciones lineales convexas de los parámetros consecuentes a_i y b_i que se expresan como:

$$a(x) = \sum_{i=1}^K \gamma_i(\mathbf{x}) \mathbf{a}_i, b(x) = \sum_{i=1}^K \gamma_i(\mathbf{x}) b_i \quad (2.26)$$

2.3. Modelación difusa de sistemas dinámicos

Es bien sabido que los sistemas dinámicos invariantes en el tiempo son en general modelados a través de funciones estáticas usando el concepto de estado del sistema. Dado el estado de un sistema y dadas sus entradas, se puede determinar cual será el siguiente estado. En el dominio del tiempo discreto se puede definir la siguiente ecuación:

$$x(k+1) = f(x(k), u(k)) \quad (2.27)$$

En donde $x(k)$ y $u(k)$ son los estados y las entradas en el tiempo k y f es una función estática, denominada función de transición de estado. Diversos tipos de modelos difusos pueden ser usados para aproximar la función de transición de estados. Dado que el estado de un proceso por lo general no es medible, la modelación entrada-salida es aplicada como alternativa viable de solución. El método más usado se llama NARX (Modelo No Lineal Autoregresivo con entradas exógenas) definido de la siguiente forma:

$$y(k+1) = f(y(k), y(k-1), \dots, y(k-n_y+1), u(k), u(k-1), \dots, u(k-n_u+1)) \quad (2.28)$$

En donde $y(k), \dots, y(k-n_y+1)$ es la salida actual y las salidas anteriores y $u(k), \dots, u(k-n_u+1)$ denota la entrada actual y las entradas anteriores y n_y y n_u son enteros que denotan el orden del modelo. Por ejemplo, un modelo difuso singleton puede consistir de reglas de la siguiente forma:

R_i : Si $y(k)$ es A_{i1} y $y(k-1)$ es A_{i2} y...y $y(k-n+1)$ es A_{in} y $u(k)$ es B_{i1} y

$u(k-1)$ es B_{i2} y...y $u(k-m+1)$ es B_{in} Entonces $\hat{y}(k+1)$ es c_i (2.29)

En este sentido, se puede decir que el comportamiento dinámico es llevado a cabo por filtros dinámicos agregados al sistema difuso (ver figura 2.5). Por ejemplo en

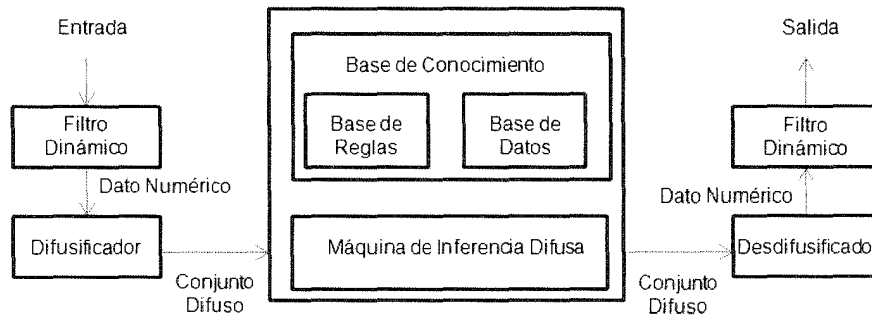


Figura 2.5: Sistema difuso genérico con unidades de difusificación y desdifusificación con filtros dinámicos externos

la base reglas anterior, el filtro dinámico de entrada es un generador de entradas y salidas en retardo. Dado que los modelos difusos pueden aproximar cualquier función continua con cualquier grado de exactitud (Wang 1997), entonces modelos del tipo (2.23) pueden aproximar bien sea en modo observable y/o controlable una gran cantidad de sistemas no lineales de tiempo discreto (Leonaritis and Billings 1985).

Para la identificación de modelos difusos de los procesos dinámicos que serán explicados en el capítulo 3, se tomará como referencia la estructura del modelo NARX dada por $y(\tau + 1) = f(y(\tau), u(\tau))$, la cual es similar a la ecuación (2.27). En (Barragan 2009) y (Aznarte 2008) se explican algunos criterios para seleccionar la estructura de modelos NARX. En general, la identificación de modelos difusos partiendo de la estructura del modelo NARX de la ecuación (2.27) con la premisa de diseñar posteriormente controladores, es una alternativa que ha dado resultados satisfactorios como se demuestra en (Babuska 1996)(Abonji 2007).

2.4. Construcción de modelos difusos

En el diseño de modelos difusos hay que distinguir dos partes fundamentales: la estructura y los parámetros. La estructura determina la flexibilidad del modelo. Una vez seleccionada, se estiman los valores de los parámetros para que el modelo se asemeje a los datos experimentales que existan.

2.4.1. Elementos a identificar

En los modelos difusos, la selección de la estructura conlleva a la elección de:

- Variables de entrada y salida: No siempre está claro que variables deben tomarse como entradas al modelo del sistema que se pretende identificar; además, también es importante estimar el orden del proceso. Por lo general, se hace uso del comportamiento del sistema, aunque también es posible definir algún índice que nos permita comparar elecciones de forma automática, basándose en datos experimentales.
- Estructura de las reglas: hay que elegir el tipo de modelo (Takagi-Sugeno, lingüístico, relacional, etc) y la forma del antecedente de las reglas teniendo en cuenta el propósito del modelo y el conocimiento del que se dispone.
- Número y tipo de funciones de pertenencias para cada variable: con esta elección se determina la granularidad del modelo. En este caso, la clave en la elección es el propósito del modelo, aunque también se pueden usar métodos automáticos que, haciendo uso de datos experimentales, añadan o quiten funciones de pertenencia del modelo.
- Mecanismo de inferencia, operadores conectivos y métodos de desfusificación: aunque estas elecciones vienen restringidas por el tipo de modelo elegido, aunque queda algo de libertad. Generalmente, si se pretende hacer uso de algún método de optimización del modelo usando datos experimentales es preferible usar los operadores diferenciales (producto, suma) a los habituales de mínimo o máximo.

Una vez que la estructura ha sido fijada, los parámetros deben ser ajustados. Estos parámetros ajustables son: las reglas y las funciones de pertenencia del antecedente y el consecuente.

2.4.2. Preprocesamiento y extracción de datos

Se debe tener cierta precaución durante la recolección de datos de históricos para la construcción de un modelo difuso. En general, se debe tratar de recolectar datos con el mayor rango de variación posible. Obviamente, es muy probable que en un proceso real no sea posible perturbar el sistema para recolectar los datos requeridos porque esto pudiera causar dificultades en la operación. Técnicas de data mining de los datos sobre un gran período de tiempo pudiera ser útil cuando los datos necesarios estén disponibles a lo largo de todo el rango de operación.

En realidad es difícil establecer un procedimiento formal para obtener la información más valiosa de los datos históricos disponibles y cada aplicación es

diferente dependiendo de la cantidad de datos, la utilidad, el rango en el cual los datos residen, etc. No obstante se recomiendan los siguientes pasos:

- Preprocesamiento de los datos (Reconciliación): esto consiste en remover aquellos datos que han sido registrados durante la falla de algún sensor, falla de equipos o sencillamente rangos fuera del punto normal de operación. Se debe verificar la desviación estándar de los datos y la media. La filtración de los datos juega un papel crucial a la hora de obtener datos con la mayor calidad posible.
- Reducción de datos: en muchos casos, no es necesario recolectar una gran cantidad de datos ya que es posible encontrar redundancia entre éstos. Se pueden usar técnicas de correlación para determinar la influencia de las entradas sobre las salidas.

2.4.3. Métodos de aprendizaje

En base a las características de la información disponible, las técnicas de optimización existentes para el diseño de modelos desde datos experimentales son tradicionalmente divididos en dos grandes grupos (Jang, Sun, and Mizutani 1997). El primero de ellos es el que incluye los llamados métodos de aprendizaje supervisados, basados o no en el método del gradiente, en los cuales se dispone de información sobre las respuestas deseadas. El otro gran grupo es el método de aprendizaje no supervisados en los que no se tiene ninguna información sobre las respuestas deseadas. Por supuesto, el límite entre ambos métodos no está nada definido y existen métodos intermedios como el aprendizaje reforzado, en los que se tiene sólo una información parcial de las salidas deseadas (Aprendizaje por premio y castigo).

Entre los métodos de aprendizaje supervisado se encuentran las conocidas técnicas de mínimos cuadrados o propagación hacia atrás (backpropagation) (Wang 1994). Entre los métodos no supervisados se encuentran los que utilizan redes neuronales (redes con aprendizaje competitivo, redes autoorganizadas de Kohonen y aprendizaje del Hebbiano), las técnicas de agrupamiento y algunas mixtas como LVQ (Learning vector quantization), donde se inicializa con agrupamiento y se continua con aprendizaje competitivo (Chan, Narasbadi, and Mirelli 1996).

En cuanto a la aplicabilidad de las diferentes técnicas de identificación para la construcción de un modelo difuso existen diferentes opciones. Por ejemplo, las plantillas constituyen un método heurístico simple y eficaz para diseñar sistemas

difusos desde datos entrada-salida que se desarrolla en cinco pasos (Wang 1994).

Una opción muy útil e inspirada en el método de propagación del error hacia atrás para redes neuronales (Wang 1994), consiste en especificar la estructura del sistema (con la ayuda de un experto) y ajustar sus parámetros con el método del gradiente de forma iterativa hasta que el error entre la salida estimada y la real para cada dato no supere un error máximo permitido.

El método de mínimos cuadrados recursivo también puede ser empleado para definir los parámetros del sistema difuso de forma similar al caso anterior pero minimizando en este caso un error global (suma de todos los errores de predicción de los pares entrada-salida). En general, los métodos que se han descrito no permiten establecer un marco de referencia formal para la identificación, dado que hay numerosos parámetros (sobre todo en la determinación de la estructura y el número de reglas) que exigen conocimiento del experto y la habilidad del ingeniero. En particular, las técnicas de agrupamiento difuso como herramienta para la determinación de modelos difusos de sistemas no lineales (ésto será descrito con suficiente detalle en el capítulo 3) es una opción que minimiza de manera notable este problema, ya que permite (dado que es un método de aprendizaje no supervisado) construir un modelo difuso basado en reglas únicamente a partir de datos experimentales incluyendo tanto la selección de la estructura como el ajuste de los parámetros, con la ventaja de que la identificación puede quedar automatizada. En cuanto a la estimación de los parámetros del antecedente y consecuente con esta técnica, se suele aplicar técnicas de mínimos cuadrados que minimicen el error de predicción (Babuska 1996). También es posible plantear problemas de optimización que son resueltos usando algoritmos genéticos como en (Delgado, Gomez, and F. 1997) (Kim, Minkee, and Park 1997) (Kinzel, Klawonn, and Kruse 1994).

2.4.3.1. Métodos de mínimos cuadrados

Los métodos de mínimos cuadrados, en general, permiten calcular los parámetros consecuentes de un sistema difuso Takagi-Sugeno, luego de la aplicación de algún algoritmo de agrupamiento, que serán revisados en detalle en el próximo capítulo. Los métodos de mínimos cuadrados son relativamente sencillos y tienen clara conexión con los métodos de estimación. En el apéndice A se deducen los métodos de mínimos cuadrados tanto de tipo batch como de tipo recursivo.

2.4.3.2. Métodos del gradiente

Mientras en la sección anterior se propuso entrenar un modelo difuso que pudiera de manera apropiada aproximar la función g que es inherentemente representada por los datos de entrenamiento G . En esta sección se plantea el uso del método de optimización del gradiente para calcular el vector de parámetros θ que desarrolle la mejor aproximación posible. Desafortunadamente, el método del gradiente no garantiza que sea exitoso en la búsqueda de la mejor aproximación. Comparado con los métodos de mínimos cuadrados, a través de este método es posible sintonizar los parámetros de un sistema difuso, como por ejemplo, las funciones de membresía de las variables de entrada. Para ello se define, por ejemplo, un modelo difuso del tipo Takagi-Sugeno que puede ser expresado de manera compacta a través de la siguiente ecuación:

$$f(x|\theta(k)) = \frac{\sum_{i=1}^C g_i(x, k)\beta_i(x, k)}{\sum_{i=1}^C \beta_i(x, k)} \quad (2.30)$$

Si se parte del hecho de que las funciones de membresía de las variables de entrada son de tipo Gaussianas entonces:

$$\beta_i(x^m, k) = \prod_{j=1}^n \exp\left(-\frac{1}{2}\left(\frac{x_j^m - c_j^i(k)}{\sigma_j^i(k)}\right)^2\right) \quad (2.31)$$

En donde σ_j^i representa el ancho de las funciones de membresía de las variables de entrada y c_j^i son los centros de dichas funciones. Por otro lado $x = [x_1, x_2, \dots, x_n]^T$ y finalmente, $g_i(x, k)$ se define como:

$$g_i(x, k) = a_{i,0}(k) + a_{i,1}(k)x_1 + a_{i,2}(k)x_2 + \dots + a_{i,n}(k)x_n \quad (2.32)$$

Se ha incorporado el índice $k > 0$ dado que se requiere actualizar los parámetros $a_{i,j}$ en cada instante de tiempo k . Para ello usaremos la siguiente expresión:

$$a_{i,j}(k+1) = a_{i,j}(k) - \lambda_1 \frac{\partial e_m}{\partial a_{i,j}} \Big|_k \quad (2.33)$$

En donde $\lambda_1 > 0$ es el tamaño de cada paso. Defínase el error en la iteración k como $\epsilon_m(k)$ expresado de la siguiente forma:

$$\epsilon_m(k) = f(x^m|\theta(k)) - y^m \quad (2.34)$$

En base a esto se tiene que:

$$\frac{\partial e_m}{\partial a_{i,j}} = \epsilon_m(k) \frac{\partial f(x^m|\theta(k))}{\partial g_i(x^m, k)} \frac{\partial g_i(x^m, k)}{\partial a_{i,j}(k)} \quad (2.35)$$

para todo $i = 1, 2, \dots, C, j = 1, 2, \dots, n$. Por otro lado:

$$\frac{\partial f(x^m | \theta(k))}{\partial g_i(x^m, k)} = \frac{\beta_i(x^m, k)}{\sum_{i=1}^C \beta_i(x^m, k)} \quad (2.36)$$

para todo $i = 1, 2, \dots, C$. También:

$$\frac{\partial g_i(x^m, k)}{\partial a_{i,0}(k)} = 1 \quad (2.37)$$

Y finalmente:

$$\frac{\partial g_i(x^m, k)}{\partial a_{i,j}(k)} = x_j \quad (2.38)$$

para todo $j = 1, 2, \dots, n$ e $i = 1, 2, \dots, C$. Con ésto se actualiza cada uno de los parámetros de la función lineal de cada consecuente del modelo Takagi-Sugeno. Con respecto al método del gradiente por descenso, en general, hay que tener mucha cautela a la hora de seleccionar el valor de λ y las consideraciones con respecto a la convergencia.

2.5. Sistemas multiagente

En esta sección, dada la gran variedad de información que se maneja en la literatura, se presentarán los conceptos más relevantes relacionado a los los sistemas multiagente.

2.5.1. Agente y sistemas multiagente

Un agente inteligente es un proceso computacional capaz de realizar tareas de forma autónoma y que se comunica con otros agentes para resolver problemas mediante cooperación, coordinación y negociación. Habita en un entorno complejo y dinámico con el cual interactúa para conseguir un conjunto de objetivos (Wooldridge 2002). En otras palabras, un agente es una entidad activa que lleva a cabo procesos cognitivos como respuesta a las percepciones que recibe del entorno donde se encuentra, de tal manera que responde actuando sobre el mismo.

Al igual que con el concepto de agente, para la definición de sistemas multiagente (SMA) se encuentran algunas variantes tales como:

- Un sistema multiagente es diseñado e implementado como varios agentes interactuando entre sí, para así lograr una funcionalidad deseada.

- Su aspecto clave reside en que realiza operaciones muy simples que permiten resolver problemas complejos, el intercambio de información entre agentes es el factor que hace posible obtener buenas soluciones (Perez, Serrano, and Trigue 2005).
- Los sistemas multiagente son sistemas distribuidos con un número de agentes heterogéneos autónomos individuales que interactúan y operan en un entorno. Los agentes pueden ser físicos o lógicos (Amigoni and Solmalvico 2003).
- Un sistema multiagente se concibe como una sociedad, similar a una sociedad humana, consistente de entidades que poseen características similares a los humanos, tal como movilidad, inteligencia, habilidad de comunicación, etc. (Mouratidis, Giorgini, and Manson 2003)(Alexandres 2007).

Es importante destacar que para soportar SMA es necesario disponer de un entorno adecuado para su desarrollo (Giret, Infrán, Pastor, and Cernuzzi 2000). Este tipo de entorno debe disponer al menos de las siguientes características:

- Contener agentes con las capacidades de autonomía, adaptabilidad, interacción y coordinación.
- Partir de un diseño abierto donde se abandone el diseño centralizado.
- Ofrecer una infraestructura donde se pueda especificar claramente la comunicación y los protocolos de interacción entre los agentes .

La implementación de un SMA suele plantear una serie de problemas como la elección del tipo de comunicación entre los agentes, el tipo de plataforma o de la metodología de desarrollo, de los criterios para la seguridad del sistema, entre otros. Sin embargo, un SMA tiene grandes ventajas con respecto a un sistema centralizado o monolítico como son:

- Solución de problemas con mayor rapidez, debido al aprovechamiento de procesamiento en paralelo.
- Comunicación mínima, pues se transmite soluciones parciales de alto nivel a otros agentes en vez de tener que enviar datos básicos a un sistema central.
- Mayor flexibilidad pues se tienen agentes con diferentes habilidades que en forma dinámica cooperan entre sí para resolver problemas.
- Mayor confiabilidad, pues otros agentes pueden tomar las responsabilidades de los agentes que llegasen a fallar en su operación.

2.5.2. Interacciones entre agentes en un sistema multiagente

En un SMA las interacciones entre agentes son necesarias para poder lograr un objetivo común. El intercambio de información tiene la finalidad de coordinar los elementos que trabajan juntos por una meta común (coordinación), otra interacción es la sincronización de acciones antes de iniciar una actividad particular o resolver conflictos (negociación) (Posada 2003). Esta interacción se lleva a cabo intercambiando información (pase de mensajes) de modo síncrono o asíncrono y pueden ocurrir entre agentes similares o diferentes dentro del mismo ambiente o en ambientes heterogéneos. Ya que las interacciones en SMA son dinámicas, es decir, pueden crearse y terminarse en el tiempo, se han definidos protocolos de interacción y lenguajes de comunicación de agentes (Agent Communication Language ACL) para poder llevar a cabo la comunicación (Grosz and Sidner 1990). La interacción entre agentes se da por medio del intercambio de información, que puede ser de alguna de las siguientes formas:

- Comunicación mediante mensajes que se deben establecer entre los agentes.
- Percepción de los cambios en el medio ambiente debidos a la acción de otro(s) agente(s).

Si un agente adquiere conocimiento sobre otros agentes, puede ser capaz de reconocer las acciones y de inferir las decisiones de dichos agentes. Es posible distinguir dos tipos de intercambio entre agentes, de acuerdo al tipo de información que se intercambia: intercambio de conocimiento e intercambio de soluciones.

Intercambio de conocimiento: Se debe a las diferencias y a la percepción incompleta del medio ambiente que rodea a dos agentes debido al conocimiento parcial sobre la realidad. De acuerdo a lo anterior, sus conocimientos podrían ser complementarios o conflictivos acerca de una situación común. En el caso de conocimiento complementario, es útil que cada agente pueda razonar acerca del conocimiento de otros agentes para poder hacer el intercambio de información.

Intercambio de posibles soluciones: A partir del conjunto de posibles soluciones, se requiere que los agentes se pongan de acuerdo para seleccionar una solución mutua cuando éstos se encuentran cooperando entre sí. Una opción para esta situación, consiste en ponerse de acuerdo al optar por una primera solución que se encuentre en la intersección de las soluciones halladas por cada agente. Sin embargo, puede ser que la solución que sea común para ambos agentes, tenga diferente calificación para cada uno. Puede ser que para un agente una solución

tenga una buena calificación, mientras que para el otro, la solución común no es interesante. Bajo esta situación, es necesario que se establezca una negociación entre agentes (Sundermeyer 1996).

2.5.2.1. Técnicas de interacción entre agentes

Una técnica de interacción entre agentes ampliamente utilizada es la comunicación directa. Una forma de éste tipo de comunicación es el paso de mensaje punto-a-punto. Los agentes que intervienen en la comunicación tienen que conocerse entre ellos. Este enfoque es bastante difícil cuando no se conocen los diferentes agentes con los que se desea interactuar o el número de ellos es muy grande. Otra manera de comunicación directa es el pase de mensajes denominado comúnmente “*broadcast*”, en la cual, un mensaje es enviado simultáneamente a varios agentes con comportamiento similar (entienden el mismo lenguaje) siendo el más utilizado en los sistemas distribuidos. Esta forma de comunicación tiene la desventaja del alto costo de comunicación cuando se envía información a un grupo muy numeroso de agentes o se tienen agentes con diferentes comportamientos.

Para evitar los problemas de la comunicación directa se usa la comunicación indirecta. En esta forma de comunicación los agentes se organizan en grupos, y no se comunican directamente sino por medio de un *facilitador*. Los agentes de un grupo que entienden un lenguaje pueden comunicarse con otros agentes que entienden el lenguaje por medio del *facilitador* pero se conserva el problema del costo de comunicación cuando hay numerosos grupos de agentes con diferentes lenguajes como es el caso de Internet.

2.5.2.2. Colaboración, coordinación y comunicación entre agentes

Los problemas que intentan solucionar los SMA se resuelven en función del grado de cooperación o colaboración entre los agentes individuales. Es importante enfatizar que en el diseño de esta faceta en los agentes, tradicionalmente se han presentado tres estrategias de interacción básicas:

- *Agentes cooperativos*: trabajan conjuntamente con la intención de resolver juntos un problema y pueden verse como un modelo adecuado para la gestión de redes. En general, esta estrategia es útil para el control de sistemas críticos en donde se debe conocer el estado de equilibrio del sistema. Este tipo de agentes está pensado para sacrificar la utilidad individual a cambio del bien general de todo el sistema.

- *Agentes auto-interesados*: son aquellos que intentan maximizar su propio beneficio sin preocuparse del interés general del sistema. En general, este tipo de coordinación es apropiado cuando los agentes van a competir en entornos abiertos.
- *Agentes hostiles*: Este tipo de agentes no tiene una aplicación directa en los agentes que integran al sistema propuesto; ya que se basan en la idea de una utilidad que se ve incrementada con su propia ganancia y con la pérdida de otros agentes competidores.

Los agentes que integran a un SMA deben colaborar entre ellos para desempeñar sus tareas, esta colaboración suele realizarse mediante un lenguaje de comunicación comprensible por todos los agentes que integran el sistema, y también por otros programas si es necesario. La diferencia entre un agente y un objeto (dentro la programación orientada a objetos) es que el último ejecuta métodos de otro objeto siempre que tenga permiso, un agente puede rechazar una petición de otro agente, por lo que deben ser capaces de comunicarse entre sí, para decidir qué acción ejecutar o qué datos obtener. El mayor inconveniente para conseguir la comunicación está en el hecho de que existen múltiples lenguajes propios de cada fabricante, lo que no facilita la compatibilidad en sistemas heterogéneos.

Los agentes que integran un SMA deben coordinar sus actividades para alcanzar sus objetivos y ser capaces de negociar con otros agentes cuando aparecen problemas como la escasez de recursos. La coordinación es necesaria para determinar la organización estructural entre un grupo de agentes. Una de las técnicas más usadas para la asignación de recursos y tareas entre una sociedad de agentes es "*Contract-Net Protocol (CNP)*", adoptado en las especificaciones de "*Foundation for Intelligent Physical Agents (FIPA)*". En CNP se aplica una estructura de mercado donde los agentes toman dos roles principales, el de director y el de contratista (Weiming, Douglas, and Barthes 2001).

El principio básico en la coordinación es el siguiente: si un agente no puede resolver la acción que tiene asignada usando su conocimiento local, entonces lo que requiere hacer es descomponer el problema en sub-problemas, y buscar a otros agentes con recursos suficientes que sean capaces de solucionar los sub-problemas. La asignación de los sub-problemas se soluciona mediante un mecanismo de contratación donde un agente director crea un contrato que luego difunde a otros agentes del sistema que pueden aceptarlo o no. El CNP aporta las ventajas de permitir la asignación dinámica de tareas con la participación de otros agentes.

La comunicación entre los agentes y la forma de realizarla es otra de las características importantes en los SMA. Los agentes pueden pasar información a otros

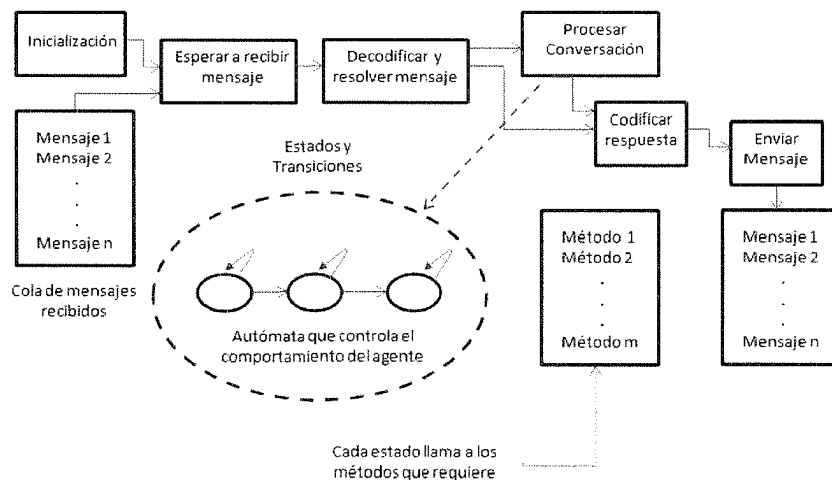


Figura 2.6: Algoritmo de coordinación de agentes

agentes para compartir entre ellos los cambios producidos en el sistema o bien para informarse de los cambios previstos en el estado del sistema. Esto generalmente se realiza en los SMA en los niveles de abstracción muy altos, en lugar de usar los tradicionales métodos remotos de paso de mensajes (Giampapa and Sycara 2003).

Normalmente, la comunicación entre agentes implica el paso de información a nivel de conocimiento, y la comunicación puede realizarse a través de mecanismos dinámicos como es la memoria compartida, los “Inter Process Communication (IPC)” o a través de mecanismos estáticos como es el uso de archivos con bloques. De este modo, la literatura nos presenta dos categorías de comunicación entre agentes; por un lado, el uso del ACL propietarios y por otro lado, el uso de ACL normalizados.

La figura 2.6 muestra el algoritmo general de coordinación entre agentes, y el mecanismo para controlar el comportamiento de cada agente (Vasquez 2003). El núcleo del agente se basa en un algoritmo de coordinación, el cual se encarga de manipular el comportamiento del agente de acuerdo a los mensajes que recibe y el conocimiento que tiene.

En la figura 2.6, se puede visualizar un primer paso que indica la inicialización del agente, quedando en estado de espera de los mensajes provenientes de otros agentes. Cuando un mensaje llega a la cola de mensajes recibidos, se debe decodificar el mensaje (esto lo realiza el módulo codificador/decodificador de mensajes programado en cada agente). Una vez decodificado el mensaje si no puede ser en-

tendido por el módulo de razonamiento, entonces se envía una respuesta al agente que originó el mensaje indicando que hubo un error. Por el contrario, si el mensaje recibido lo maneja correctamente el agente receptor, entonces el módulo de razonamiento procesa la conversación.

En este punto el comportamiento del agente es modelado de acuerdo a una determinada conversación que tiene con otros agentes. Para responder al mensaje, se expresa en términos de estados y transiciones. En cada estado de la conversación, el agente puede hacer llamadas a métodos que le permiten hacer inferencias sobre el conocimiento que tiene para dar una respuesta, o generar un mensaje de pregunta, o solicitar alguna acción a otro agente, o utiliza la interfaz de usuario para comunicarse con él. Los mensajes de salida, son producto de cada estado de la conversación, y deben ser codificados por el módulo codificador/decodificador para posteriormente enviarlos al módulo de comunicaciones en la cola de mensajes por enviar. Para que dos o más agentes puedan comunicarse entre sí, es necesario contar con un lenguaje de comunicación universal. Existen actualmente dos puntos de vista en el diseño del lenguaje que son: Estilo procedural y Estilo declarativo (Bedia, J., Corchado, and Fyfe 2002).

2.5.2.3. Protocolos de comunicación

Los protocolos de interacción definen la estructura de la secuencia de intercambio de información (mensajes) entre agentes. Entre los protocolos más conocidos para lograr la interacción entre los agentes se mencionan los siguientes:

- Protocolos de coordinación: en el cual los agentes se coordinan de tal manera que llegan a un acuerdo entre ellos, para alcanzar sus metas. “ContractNet” es un protocolo de este tipo, donde se ofrece una propuesta a los agentes y ellos deben de ofrecer soluciones a dicha propuesta eligiendo la mejor solución, se le debe informar al agente que ganó para que realice su propuesta basándose en reglamentos previamente definidos, mientras que los otros agentes se deslindan de la propuesta (Conry 1998).
- Protocolos de negociación: se utilizan cuando los agentes necesitan desempeñar una tarea con la mejor propuesta para resolver conflictos. Uno de los protocolos de negociación es Workteam. Este protocolo permite al agente recibir la especificación de un problema. El agente puede entonces generar una propuesta la cual otros agentes pueden revisar y responder; las respuestas son revisadas y el proceso se repite.

- Protocolos de colaboración: son utilizados con la finalidad de que los agentes puedan pedir ayuda a otros agentes y así resolver el problema. De tal manera, los protocolos de colaboración se utilizan para resolver problemas de gran tamaño, donde un agente individual no puede resolver (Murilo and Melo 2002).

Existe el protocolo de colaboración en donde si un tipo de agente ofrece asistencia (a otros agentes que integran el SMA) se tiene registrar dentro de un agente tipo “board” que debe ser conocido por todos los agentes. De tal manera que si un determinado agente llegase a estar ante una situación problemática puede consultar al agente board para obtener información de los agentes que le pueden dar asistencia. La colaboración entre agentes se da a manera de solicitud y réplica.

2.5.3. Aprendizaje y razonamiento

En virtud de que el aprendizaje en un SMA está relacionado con varias áreas de investigación, se han creado de una gran variedad de modelos de razonamiento y aprendizaje. Debido a esto se han diseñado diferentes tipos de agentes que pueden aprender en base a órdenes, a ejemplos, por observación, debido a la comunicación con otros agentes, por refuerzo, por clasificación de información, por construcción de modelos y por formas especializadas de comparación. En la actualidad es muy frecuente desarrollar el aprendizaje y razonamiento en los agentes utilizando nuevas técnicas de Inteligencia Artificial que les permite aprender directamente de su entorno, como son: redes neuronales, razonamiento basado en casos, algoritmos genéticos y algoritmos difusos.

2.5.4. Modelado de un sistema multiagente

En un SMA general, cada agente tiene un comportamiento autónomo, ya que puede iniciar concurrentemente acciones por sí mismo y también puede tener interacciones concurrentes entre diferentes agentes del sistema. Por lo tanto, esto sugiere que modelar un SMA requiere un formalismo que exprese de manera adecuada, entre otras cosas, la ejecución concurrente de las acciones internas y de las interacciones de cada agente.

Los diferentes formalismos que existen actualmente para especificar agentes deben ser capaces de representar los siguientes aspectos :

- La información de los agentes acerca del ambiente que lo rodea.

- Las metas que tienen un agente por alcanzar.
- Las acciones que desempeña el agente para alcanzar sus metas.
- Las interacciones de un agente con su ambiente y otros agentes.

Por ejemplo, el agente puede recibir, procesar y enviar varios mensajes al mismo tiempo sin bloquearse (varias acciones concurrentemente). Entre los formalismos más importantes para especificar un SMA se encuentran la lógica, el álgebra de procesos, los lenguajes formales, los autómatas y las Redes de Petri. Cabe mencionar que hay formalismos que se adaptan mejor que otros a ciertos tipos de agentes.

El caso particular de las Redes de Petri (Lakos 1995) sobre otras herramientas de modelado es su poder de visualización y facilidad para el modelado de concurrencia, causalidad y sincronización de procesos; además tienen una base matemática para el análisis de propiedades de los modelos obtenidos. Las Redes de Petri son un buen formalismo para el modelado del comportamiento de agentes autónomos e interacciones entre ellos, ya que modelan fácilmente procesos con acciones (transiciones) concurrentes a partir de un estado (marcado) determinado, además de que implican una interpretación computacional o flujo de ejecución.

El álgebra de procesos temporizada básica, se trata de un lenguaje recursivo, secuencial no determinista. Con el álgebra de procesos se pueden modelar diferentes protocolos de comunicación entre agentes con el propósito de intercambiar información a través de un determinado medio.

2.5.5. Ontología y gestión del conocimiento

La necesidad de los sistemas distribuidos de utilizar y compartir información implica la utilización de vocabularios comunes que permitan describir y representar de forma unívoca, para todos los componentes del sistema, el conocimiento que se tiene del mismo.

La representación de este conocimiento se enfoca mediante el diseño de formalismos computacionalmente apropiados que permiten expresar y representar la información en un área particular. Este conocimiento se ha representado en los sistemas de información utilizando diversos formalismos: en las bases de datos se han utilizado diagramas entidad-relación para definir los conceptos y sus relaciones en un determinado universo; en programación se han utilizado gramáticas y estructuras de datos como clases y objetos, en Ingeniería del Software se ha

propuesto el uso de lenguajes de modelado, como UML, donde es posible definir clases y sus relaciones.

Otro de los formalismos utilizados son las ontologías que proporcionan una forma de representar y compartir conocimiento haciendo uso de un vocabulario común. Mediante esta representación permiten utilizar un formato de intercambio de este conocimiento, permitiendo la posibilidad de ampliar, integrar o reutilizar otras ontologías (Samper 2005).

En el campo de los agentes, las ontologías son utilizadas como base del conocimiento, lo que permite que los agentes puedan comunicarse y comprender el contenido de los mensajes. FIPA define y especifica el concepto de servicio ontológico. Esta especificación trata únicamente los aspectos relacionados con el formato de comunicación de la ontología, mientras que la implementación interna y las capacidades de la misma son realizadas por los desarrolladores (Gruber 2006).

2.5.6. Metodologías orientadas a simular sistemas multiagentes

El desarrollo de cualquier tipo de software requiere de métodos y herramientas que faciliten la obtención del producto final. En esa línea, en los últimos años han aparecido diversos trabajos de investigación que proponen procesos para el desarrollo de SMA. En esta sección se presenta los métodos más utilizados para el análisis y diseño de un SMA. Las actuales metodologías de desarrollo para SMA toman algunos de los conceptos de la Ingeniería de Software Orientada a Objetos y de la Inteligencia Artificial. Desde el punto de vista de las mencionadas tecnologías, los sistemas distribuidos pasan tomar el nombre de SMA. Las tecnología Multiagente representan algo nuevo y excitante para el análisis, diseño y desarrollo de software complejo, a continuación se presenta la definición de lo que es una metodología para modelar software que aplica para los SMA.

Una metodología se presenta normalmente como una serie de pasos, con técnicas y notaciones asociadas a cada paso. Los pasos de la producción del software se organizan normalmente en un ciclo de vida consistente en varias fases de desarrollo (Rumbau and Blaha 1991). Ahora bien, todo los conceptos antes mencionados relacionados con los agentes y SMA, aterrizan en poder aplicarlos a un sistema que resuelva un problema determinado para ello es requisito indispensable desarrollarlo creando un modelo del SMA basado en un análisis de tal manera que su implementación sea relativamente fácil.

2.6. Conclusiones

En este capítulo se revisaron cuatro tipos de modelos difusos basados en reglas: el modelo lingüístico (Mamdani), el modelo relacional, el modelo unitario (singleton) y el modelo Takagi-Sugeno. La diferencia principal entre ellos es que mientras los modelos lingüísticos (el modelo relacional es una extensión del modelo lingüístico) tienen tanto en el antecedente como en el consecuente conjuntos difusos, el modelo Takagi-Sugeno tiene funciones reales en su consecuente y específicamente en el modelo difuso unitario, los consecuentes son números reales. En general, estos modelos pueden ser usados para modelar sistemas estáticos y dinámicos, y en particular, los modelos Takagi-Sugeno brindan el marco ideal para la construcción de un modelo no lineal conformado por modelos lineales locales.

Se verificaron las consideraciones para la selección de los datos de entrada - salida para determinar el modelo difuso de una planta, haciendo énfasis en los modelos de tipo NARX, que servirán de base para la modelación de procesos de petróleo los cuales se estudiarán en detalles en el capítulo 3. En cuanto a los métodos de aprendizaje, se destacan los métodos de mínimos cuadrados y el método del gradiente; ambos permiten adaptar modelos difusos tanto fuera de línea como en línea, lo cual representa una de las funciones que requiere el sistema supervisor que se desea diseñar en esta tesis y en particular el método de mínimos cuadrados es usado para estimar los consecuentes de los modelos difusos Takagi-Sugeno.

Por otra parte, como resultado de la revisión de los principios de los paradigmas de agentes y sistemas multiagentes, es posible concluir que la aplicación de la teoría de sistemas multiagente es de gran utilidad para el diseño del sistema supervisor inteligente sistemas complejos ya que actualmente en los complejos industriales existe una tendencia orientada a la instalación de sistemas de control distribuido en donde un conjunto de entes autónomos requieren tener la capacidad de coordinación y cooperación para lograr los objetivos de la producción.

Capítulo 3

IDENTIFICACIÓN DE MODELOS DIFUSOS A PARTIR DE DATOS DE ENTRADA-SALIDA

3.1. Introducción

La identificación de modelos difusos usando datos de entrada y salida es una alternativa que ha despertado un gran interés dado que en principio no se requiere del conocimiento experto de un operador sobre el proceso que se requiere modelar. Existen dos enfoques clásicos que permiten obtener modelos difusos y en esencia se basan en el uso de redes neuronales artificiales y computación evolutiva, los cuales, a grandes rasgos, buscan optimizar los parámetros escalares de un modelo difuso (Nauck, Klawonn, and Kruse 1996)(Jang 1993)(Jin 2000)(Roubos and Setnes 2010). Una alternativa distinta e interesante consiste en el uso de técnicas de agrupamiento difuso del espacio producto de los datos de entrada - salida. Tales técnicas se consideran métodos muy versátiles para la identificación de este tipo de modelos en virtud de la clara conexión que existe entre una determinada clase o grupo de datos y una regla difusa.

En este capítulo se van a evaluar tres metodologías que permiten identificar modelos difusos a partir de datos de entrada - salida. La primera metodología trata sobre las técnicas de agrupamiento difuso y específicamente se hará una revisión exhaustiva de los algoritmos Fuzzy C-Means o media de la clase difusa y Gustafson-Kessel. En particular, estos algoritmos se aplican en la identificación de modelos difusos para los procesos de separación gas-líquido y calentamiento del petróleo . En consecuencia, se pretende obtener un modelo difuso de tipo Takagi-Sugeno de primer orden para los procesos mencionados usando cada uno de los algoritmos de agrupamiento difuso evaluando por una parte la exactitud de cada

CAPÍTULO 3. IDENTIFICACIÓN DE MODELOS DIFUSOS A PARTIR DE DATOS DE ENTRADA-SALIDA

modelo en base a la raíz cuadrada del error cuadrático medio y por otra parte, en términos de la convexidad de las funciones de pertenencia del antecedente, se aplicará tanto el índice de solapamiento como el índice de compacidad que son de gran aplicabilidad para el análisis de la interpretabilidad (Diez 2003).

La segunda metodología consiste en el uso de técnicas neurodifusas, específicamente la técnica denominada sistema de inferencia difuso basado en redes adaptativas (ANFIS), en donde previamente se hará una revisión sobre sus fundamentos teóricos y su aplicación a los casos de estudios mencionados. Se establecerán las comparaciones con respecto a las técnicas de agrupamiento.

Finalmente, se propondrá una alternativa para la obtención de modelos difusos de alta interpretabilidad, específicamente modelos difusos unitarios (singleton) en base a datos de entrada - salida, a través del uso de algoritmos genéticos. Se presentará un análisis desde el punto de vista de la interpretabilidad y se determinará la alternativa más idónea que tenga el mejor equilibrio exactitud e interpretabilidad en comparación con las técnicas de agrupamiento y con la técnica de modelación neurodifusa.

El capítulo está organizado de la siguiente forma. En primer lugar, se presentarán los fundamentos de las técnicas de agrupamiento difuso y sus respectivas consideraciones teóricas que se deben tomar en cuenta durante el proceso de identificación. En segundo lugar se revisarán los fundamentos teóricos de la modelación neurodifusa, específicamente el modelo ANFIS. Luego, se mostrarán los resultados de la identificación de los modelos difusos del proceso de separación gas - líquido y del de calentamiento de crudo en base a las técnicas mencionadas con sus respectivos análisis. Luego, en función de mejorar la interpretabilidad, se propone una metodología de obtención de modelos difusos unitarios a través de algoritmos genéticos. Finalmente, el capítulo cierra con las conclusiones.

3.2. Técnicas de agrupamiento

El objetivo fundamental de las técnicas de agrupamiento (clustering) es la clasificación de objetos de acuerdo con sus similitudes y la organización de datos en grupos. La propiedad más importante es su facilidad para dividir un conjunto de datos dado en varios, de tal manera que la similitud entre los elementos dentro de un grupo sea mayor comparada con la de otros grupos. Por ende, un cluster (grupo o clase) no es más que un grupo de objetos que son más similares entre ellos que a miembros de otros grupos. En particular, el término similitud debe ser

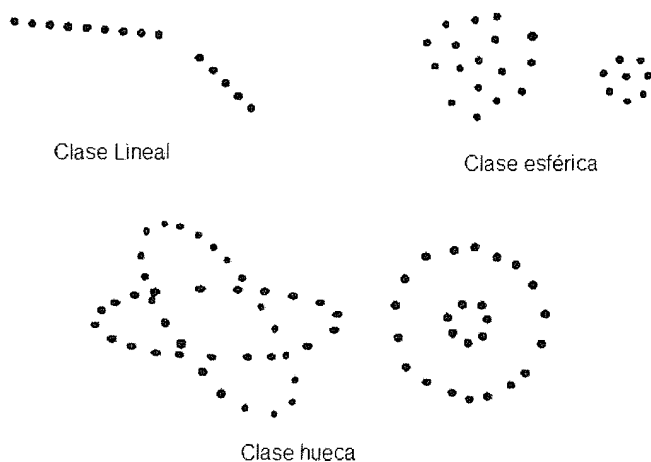


Figura 3.1: Diferentes posibilidades de clases en R^2

entendido en términos matemáticos de tal forma que sea mensurable de manera exacta. En los espacios métricos, la similitud suele ser definida mediante una norma o distancia. Esta distancia puede estar medida entre vectores de datos, o como una distancia a unos vectores de datos prototipo de la clase. Estos prototipos no son conocidos de antemano, sino que los propios algoritmos de agrupamiento los van definiendo a medida que realizan la partición de los datos (Kruse, Doring, and Lesot 1998).

El aspecto geométrico de las clases puede ser muy variado en función a las formas, tamaños y densidades tal y como lo muestra la figura 3.1 para R^2 . Tal y como se aprecia en esta figura, las clases suelen ser lineales, esféricas y huecas. El mejor o peor comportamiento de la mayoría de los algoritmos de agrupamiento es influenciado por la forma y densidad de la clase; al igual que por la relación espacial y la distancia entre ellas.

En la literatura existe una gran variedad de métodos de agrupamiento (Klawonn and Keller 1998)(Klawonn and R. 1995). Dado que las clases pueden ser vistas como subconjuntos del conjunto de datos, una primera clasificación podría ser la que tuviese en cuenta si los subconjuntos son ordinarios o difusos. Los métodos de agrupamiento clásicos se fundamentan en la teoría clásica de conjuntos y requieren que un objeto pertenezca o no a un conjunto. Al aplicar esta teoría a un conjunto dado de datos Z lo que se obtiene es una división de estos datos en un número c determinado de subconjuntos mutuamente excluyentes. En contraste, los métodos de agrupamiento difusos permiten a los objetos pertenecer a

CAPÍTULO 3. IDENTIFICACIÓN DE MODELOS DIFUSOS A PARTIR DE DATOS DE ENTRADA-SALIDA

diferentes clases simultáneamente, con diferentes grados de pertenencia, siendo c el número total de subconjuntos difusos definidos (Kruse, Doring, and Lesot 1998).

Otra posible clasificación de los métodos de agrupamiento podría establecerse con respecto al tipo de algoritmo que el método proponga. Los métodos jerárquicos utilizan técnicas de aglomeración y división de los subconjuntos creados, atendiendo a alguna medida de similitud. La mayor ventaja de estos métodos radica en su sencillez y rapidez. En los métodos basados en la teoría de grafos, el conjunto de datos Z se toma como un conjunto de nodos en donde los enlaces indican similitud entre nodos y existe la posibilidad de dividir el grafo en un conjunto de subgrafos. El tercer tipo de algoritmos de agrupamiento usa una función objetivo para medir la adecuación de las particiones, minimizando una función objetivo global (Diez 2003).

El objetivo del agrupamiento es dividir el conjunto de datos $Z = \{z_1, z_2, \dots, z_N\}$ (cada dato puede ser de dimensión n) en c clases. Supongase que c es conocido; por ejemplo en base a la experiencia. Una partición ordinaria de Z se define como la familia de los subconjuntos $\{A_i | 1 \leq i \leq c\} \subset P(Z)$ con las siguientes propiedades (Bezdek 1987):

$$A_i \cap A_j = \emptyset, 1 \leq i \neq j \leq c \quad (3.1)$$

$$\emptyset \subset A_i \subset Z, 1 \leq i \leq c \quad (3.2)$$

$$\bigcup_{i=1}^c A_i = Z \quad (3.3)$$

La ecuación (3.3) indica que la unión de los subconjuntos A_i contiene todos los datos de Z . Los subconjuntos han de ser disjuntos como se indica en (3.1) y ninguno de ellos es el vacío ni contiene los datos de Z , tal como se expresa en (3.2). En términos de funciones de pertenencia, las ecuaciones anteriores se pueden expresar como:

$$\mu_{A_i} \wedge \mu_{A_j} = 0, 1 \leq i \neq j \leq c \quad (3.4)$$

$$0 < \mu_{A_i} < 1, 1 \leq i \leq c \quad (3.5)$$

$$\bigvee_{i=1}^c \mu_{A_i} = 1 \quad (3.6)$$

Los valores 0 y 1 indican las funciones cero y uno, respectivamente, y μ_{A_i} es la función característica de A_i (con el fin de simplificar la notación se usará μ_i en lugar de μ_{A_i}) definida como:

$$\mu_{ik} \equiv \mu_i(z_k) = \begin{cases} 1, & z_k \in A_i \\ 0, & z_k \notin A_i \end{cases} \quad (3.7)$$

Una matriz de $U = [\mu_{ik}]$ de dimensión $c \times N$, representa una partición ordinaria si y solo si sus elementos satisfacen las condiciones:

$$\mu_{ik} \in \{0, 1\}, 1 \leq i \leq c, 1 \leq k \leq N \quad (3.8)$$

$$\sum_{i=1}^c \mu_{ik} = 1, 1 \leq k \leq N \quad (3.9)$$

$$0 < \sum_{k=1}^N \mu_{ik} < N, 1 \leq i \leq c \quad (3.10)$$

La i -ésima fila de la matriz de partición U contiene los valores de la función característica del subconjunto A_i de Z . Todo lo visto con respecto a la partición ordinaria puede resumirse en la siguiente definición (Kruse, Doring, and Lesot 1998).

Definición 3.1 Espacio de particiones ordinarias. Sea $Z = \{z_1, z_2, \dots, z_N\}$ un conjunto finito de datos y $2 \leq c \leq N$ un entero. El espacio de particiones ordinarias de Z es el conjunto:

$$M_{hc} = \{U \in R^{c \times N} \mid \mu_{ik} \in \{0, 1\}, \forall i, k; \sum_{i=1}^c \mu_{ik} = 1, \forall k; 0 < \sum_{k=1}^N \mu_{ik} < N, \forall i\} \quad (3.11)$$

La generalización de la partición ordinaria a la partición difusa de acuerdo a (Bezdek 1973) es tan sencilla como permitir que μ_{ik} tome valores reales en el intervalo $[0,1]$. Las condiciones que debe cumplir la matriz de partición difusa, análogas a (3.8), (3.9) y (3.10) son:

$$\mu_{ik} \in [0, 1], 1 \leq i \leq c, 1 \leq k \leq N \quad (3.12)$$

$$\sum_{i=1}^c \mu_{ik} = 1, 1 \leq k \leq N \quad (3.13)$$

$$0 < \sum_{k=1}^N \mu_{ik} < N, 1 \leq i \leq c \quad (3.14)$$

La siguiente definición hace un resumen con respecto a la partición difusa (Kruse, Doring, and Lesot 1998).

Definición 3.2. Espacio de particiones difusas. Sea $Z = \{z_1, z_2, \dots, z_N\}$ un conjunto finito de datos y $2 \leq c \leq N$ un entero. El espacio de particiones

difusas de Z es el conjunto:

$$M_{hc} = \{U \in R^{c \times N} \mid \mu_{ik} \in [0, 1], \forall i, k; \sum_{i=1}^c \mu_{ik} = 1, \forall k; 0 < \sum_{k=1}^N \mu_{ik} < N, \forall i\} \quad (3.15)$$

La i -ésima fila de la matriz de partición difusa U contiene los valores de la función de pertenencia del subconjunto difuso A_i de Z . La ecuación (3.13) hace que la suma de cada columna sea 1, con lo que la pertenencia total de cada z_k es 1. La distribución de las funciones de pertenencia entre los c conjuntos difusos no está restringida de ningún modo.

Existe la posibilidad de flexibilizar la condición (3.13) no obligando a que la suma de funciones de pertenencia sea la unidad para cada objeto. En (Krishnapuram and Freg 1992) se propone sustituir dicha condición por:

$$\exists i, \mu_{ik} > 0, \forall k \quad (3.16)$$

con lo que la partición difusa pasaría a ser una partición posibilística que a continuación se describe.

Definición 3.3. Espacio de particiones posibilísticas. Sea $Z = \{z_1, z_2, \dots, z_N\}$ un conjunto finito de datos y $2 \leq c \leq N$ un entero. El espacio de particiones posibilística de Z es el conjunto:

$$M_{hc} = \{U \in R^{c \times N} \mid \mu_{ik} \in [0, 1], \forall i, k; \mu_{ik} > 0, \forall k; 0 < \sum_{k=1}^N \mu_{ik} < N, \forall i\} \quad (3.17)$$

La i -ésima fila de la matriz de partición posibilística U contiene los valores de la función de pertenencia del subconjunto difuso A_i de Z . La ecuación (3.16) hace que la suma de cada columna de como resultado un número entre 0 y 1, pero en este caso la pertenencia total de cada z_k no necesariamente debe ser 1.

3.2.1. Algoritmo de agrupamiento difuso Fuzzy C-Means (FCM)

En los algoritmos de agrupamiento ordinario cada dato debe ser asignado a una sola clase. El análisis con agrupamiento difuso relaja este requerimiento permitiendo pertenencias graduales y ofreciendo la posibilidad de que un conjunto de datos pueda pertenecer simultáneamente a varias clases. La mayoría de las técnicas analíticas de agrupamiento difuso se basan en la optimización de una función objetivo c-means o alguna modificación de ésta (Bezdek 1973).

La función objetivo base de una gran familia de algoritmos de agrupamiento difuso es la siguiente:

$$J(Z; U, C) = \sum_{i=1}^c \sum_{k=1}^N (\mu_{ik})^m \|z_k - c_i\|_A^2 \quad (3.18)$$

donde:

$$Z = \{z_1, z_2, \dots, z_N\} \quad (3.19)$$

son los datos a ser clasificados, N es la cantidad total de datos y c es el número de clases (clusters) el cual es conocido de antemano o pudiera ser validado usando una medida de validez de cluster,

$$U = [\mu_{ik}] \in M_{fc} \quad (3.20)$$

es una matriz de partición difusa tal y como se definió en (3.15) de Z que contiene los grados de membresía de cada dato z_k con respecto al i -ésimo cluster,

$$C = [c_1, c_2, \dots, c_c], c_i \in R^n \quad (3.21)$$

es el vector de centros (centroides, prototipos) a determinar,

$$D_{ikA}^2 = \|z_k - c_i\|_B^2 = (z_k - c_i)^T B (z_k - c_i) \quad (3.22)$$

es una norma que representa la distancia de cada dato z_k con respecto al prototipo c_i y finalmente,

$$m \in [1, \infty) \quad (3.23)$$

es un exponente que determina la difusividad de las clases resultantes. A medida que m crece entonces las clases dectadas serán más difusas. La ecuación (3.18) esta sujeta a las siguientes restricciones:

$$\sum_{k=1}^N \mu_{ik} > 0 \forall i \in \{1, \dots, c\} \quad (3.24)$$

$$\sum_{i=1}^c \mu_{ik} = 1 \forall k \in \{1, \dots, N\} \quad (3.25)$$

La restricción (3.24) garantiza que ningún cluster esta completamente vacío y la restricción (3.25) que para cada dato su clasificación puede estar distribuida sobre diferentes clusters pero la suma de los grados de membresia a todos los cluster debe dar 1 (Yang and C. 1997).

CAPÍTULO 3. IDENTIFICACIÓN DE MODELOS DIFUSOS A PARTIR DE DATOS DE ENTRADA-SALIDA

La minimización de la función objetivo (3.18) es un problema de optimización no lineal que puede ser resuelto usando varias alternativas. Tomando en cuenta la restricción (3.25), una forma de determinar los puntos estacionarios de la ecuación (3.18) es a través del uso de los multiplicadores de Lagrange por lo que (3.18) se convierte en:

$$J(Z; U, C, \lambda) = \sum_{i=1}^c \sum_{k=1}^N (\mu_{ik})^m \|z_k - c_i\|_A^2 + \sum_{k=1}^N \lambda_k \left(\sum_{i=1}^c \mu_{ik} - 1 \right) \quad (3.26)$$

Los puntos estacionarios se calculan igualando a cero las derivadas parciales de J con respecto a U , C y λ . Se puede demostrar que dichos puntos se corresponden con las dos siguientes expresiones:

$$\mu_{ik} = \frac{1}{\sum_{j=1}^c \left(\frac{D_{ikA}}{D_{jkA}} \right)^{\frac{2}{m-1}}} \quad (3.27)$$

$$c_i = \frac{\sum_{k=1}^N (\mu_{ik})^m z_k}{\sum_{k=1}^N (\mu_{ik})^m} \quad (3.28)$$

Los pasos que debe seguir el algoritmo c-means, dado un conjunto de datos como en (3.19), un número c de clases, un exponente de ponderación $m > 1$, una tolerancia $\epsilon > 0$ (normalmente entre 0.01 y 0.001) y una norma A , son los siguientes:

- Inicializar la matriz de partición difusa con valores aleatorios $U^{(0)}$, haciendo $l = 1$ (índice de iteración)
- Calcular los centros de las clases:

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

- Calcular las distancias:

$$D_{ikA}^2 = (z_k - c_i^{(l)})^T A (z_k - c_i^{(l)}), 1 \leq i \leq c, 1 \leq k \leq N$$

- Recalcular la matriz de partición.
 - Si $D_{ikA}^2 > 0$ para todo $1 \leq i \leq c, 1 \leq k \leq N$, entonces

$$\mu_{ik}^{(l)} = \frac{1}{\sum_{j=1}^c \left(\frac{D_{ikA}}{D_{jkA}} \right)^{\frac{2}{m-1}}}$$

- En cualquier otro caso:

$$\mu_{ik}^{(l)} = 1$$

- Si $\|U^{(l)} - U^{(l-1)}\| < \epsilon$, parar. De lo contrario, volver al segundo paso del algoritmo.

La forma de las clases se determina por la elección de la matriz A en (3.22). La opción más simple es seleccionar a $A = I$ (matriz identidad) lo que genera la norma Euclídea:

$$D_{ik}^2 = (z_k - c_i)^T (z_k - c_i) \quad (3.29)$$

Esta elección induce clase hiperesféricas como la que se muestra en la figura 3.2. Otra alternativa es que A sea una matriz diagonal de dimensiones $N \times N$ que tome diferentes varianzas en cada una de las direcciones de los ejes coordenados de Z :

$$A = \begin{bmatrix} \frac{1}{\sigma_1^2} & 0 & \cdots & 0 \\ 0 & \frac{1}{\sigma_2^2} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \frac{1}{\sigma_N^2} \end{bmatrix} \quad (3.30)$$

Esta matriz induce una norma diagonal en R^N y una forma hiper-elipsoidal en las clases como lo muestra la figura 3.2. Otra opción consiste en definir a A como la inversa de la matriz de covarianzas de las muestras de Z :

$$A = R^{-1} \text{ con } R = \frac{1}{N} \sum_{k=1}^N (z_k - \bar{z})(z_k - \bar{z})^T \quad (3.31)$$

En donde \bar{z} es la media de los datos y en este caso particular se induce la norma Mahalanobis (Babuska 1996), que también da lugar a clases hiper-elipsoidales, pero con una orientación particular de sus ejes (figura 3.2).

Es importante destacar que algunas normas son sensibles a los rangos numéricos de los datos, con lo que una normalización previa puede ser una buena opción. La norma Euclídea, por ejemplo, asigna un mayor peso a aquellas características con rangos amplios que a las de rangos estrechos. Una simple elección de unidades (metros o centímetros) puede influenciar negativamente los resultados del agrupamiento. Sin embargo, la normalización algunas veces no es deseable ya que puede alterar la separación entre las clases y afectar de manera negativa los resultados del agrupamiento. Afortunadamente, los algoritmos basados en medidas de distancia adaptativas no son sensibles al escalado de los datos, ya que la adaptación de la medida de distancia compensa automáticamente las diferencias de escala.

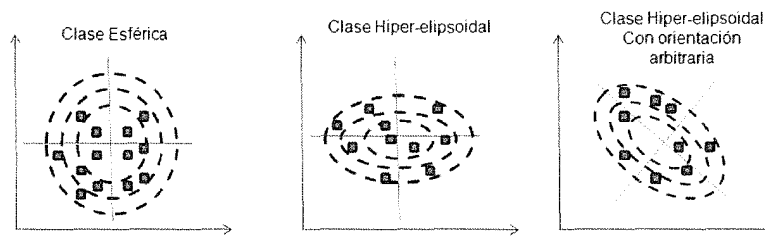


Figura 3.2: Normas usadas en agrupamiento difuso

En algunas aplicaciones, el algoritmo FCM no da buenos resultados dado que todas las clases que detectan presentan dimensiones y orientaciones similares (clases esféricas o hiper-elipsoidales) (Babuska 1996).

Sin embargo, existen numerosas extensiones y modificaciones del algoritmo básico fuzzy c-means que adaptan la ecuación (3.22) con el fin de detectar clases de diferentes formas geométricas y tamaños en un mismo conjunto de datos. Algunas de ellas pueden hacer que sus resultados sean más adecuados para otros fines, como por ejemplo el modelado y control por modelos locales. Los métodos de posible interés para aplicaciones en las que la detección de espacios lineales sea de importancia se clasifican en dos grandes grupos:

- Algoritmos que usan una medida de la distancia adaptativa, tomando una norma diferente para cada clase.
- Algoritmos basados en prototipos lineales, dejando la norma constante y haciendo los prototipos variables.

El algoritmo de Gustafson-Kessel (GK) es uno de los que hace uso de la distancia adaptativa, el cual genera clases hiper-elipsoidales que detectan de forma correcta los comportamientos quasi-lineales de los diversos regímenes de funcionamiento que pueden existir en un conjunto de datos (Gustafson and Kessel 1979). Otros algoritmos adaptativos con los que se obtienen buenos resultados son el Algoritmo de estimación de la máxima probabilidad (FMLE) y el algoritmo Gath y Geva (GG) (Abonyi, Babuska, and Szeifert 2001).

Por otra parte, los algoritmos basados en prototipos lineales mantienen la norma constante pero definen prototipos lineales r -dimensionales ($0 \leq r \leq n - 1$) en subespacios del espacio de datos. Los algoritmos más importantes en esta área son el fuzzy c-regression models (FCMR) (Hathaway and Bezdek 1991) y el fuzzy c-varieties (FCV) (Bezdek 1973).

3.2.2. Algoritmo de Gustafson-Kessel (GK)

Este algoritmo extiende el algoritmo básico fuzzy c-means eligiendo una norma diferente B_i para cada clase con lo que la ecuación 3.22 se convierte en:

$$D_{ik}^2 = \|z_k - c_i\|_{A_i}^2 = (z_k - c_i)^T A_i (z_k - c_i) \quad (3.32)$$

Estas matrices son ahora tomadas como posibles variables para la optimización de la función (3.18) con lo que se adaptará la norma a cada clase según sus características. Sea $B = \{B_1, B_2, \dots, B_c\}$ el vector que contiene las c normas. La nueva función a minimizar será:

$$J(Z; U, C, B) = \sum_{i=1}^c \sum_{k=1}^N (\mu_{ik})^m D_{ikA_i}^2 \quad (3.33)$$

cumpliendo con (3.20), (3.21) y (3.23). Para obtener una solución viable para cada norma A_i , ésta se debe limitar de alguna manera. La forma más habitual es fijar el determinante de A_i y con esto se logra optimizar la forma de la clase manteniendo su volumen constante:

$$|A_i| = \rho_i, \rho_i > 0 \quad (3.34)$$

con ρ_i constante para cada clase. La expresión que se obtiene para la minimización de A_i con la restricción (3.34) de volumen constante es:

$$A_i = [\rho_i \det(F_i)]^{1/n} F_i^{-1} \quad (3.35)$$

siendo F_i la matriz de covarianzas de clase i definida por:

$$F_i = \frac{\sum_{k=1}^N (\mu_{ik})^m (z_k - c_i)(z_k - c_i)^T}{(\mu_{ik})^m} \quad (3.36)$$

Dado un conjunto de datos Z , un número c de clases, un exponente de ponderación $m > 1$, un valor para cada determinante ρ_i (En principio si no se tiene información alguna $\rho_i = 1$) y una tolerancia $\epsilon > 0$ para la terminación, los pasos del algoritmo GK se describen a continuación:

- Inicializar la matriz de partición difusa con valores aleatorios $U^{(0)}$, haciendo $l = 1$ (índice de iteración)
- Calcular los centros de las clases:

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

- Calcular la matriz de covarianzas de cada clase:

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

- Calcular las distancias:

$$D_{ikA_i}^2 = (z_k - c_i^{(l)})^T [[\rho_i \det(F_i)]^{1/n} F_i^{-1}] (z_k - c_i^{(l)}), 1 \leq i \leq c, 1 \leq k \leq N$$

- Recalcular la matriz de partición.

- Si $D_{ikA_i}^2 > 0$ para todo $1 \leq i \leq c, 1 \leq k \leq N$, entonces

$$\mu_{ik}^{(l)} = \frac{1}{\sum_{j=1}^c \left(\frac{D_{ikA_i}}{D_{jkA_i}}\right)^{\frac{2}{m-1}}}$$

- En cualquier otro caso:

$$\mu_{ik}^{(l)} = 1$$

- Si $\|U^{(l)} - U^{(l-1)}\| < \epsilon$, parar. De lo contrario, volver al segundo paso del algoritmo.

3.3. Consideraciones teóricas para la obtención de modelos difusos usando técnicas de agrupamiento

En esta sección se hará una revisión detallada de los distintos pasos a seguir para la identificación del modelo de un sistema no lineal mediante modelos difusos, identificado su estructura y parámetros con técnicas de agrupamiento.

El proceso de identificación de modelos difusos con las técnicas vistas en la sección 2.4 es iterativo por naturaleza y consiste en la repetición, para diferentes elecciones de diversos parámetros de los pasos que se muestran en la figura 3.3 (Diez and Navarro 1999). En este sentido, podría ser útil aplicar la etapa de agrupación de los datos (mostrado en la figura 3.3), con el fin de determinar diferentes comportamientos del sistema en estudio y dar una idea de cuantas reglas son necesarias. Sin embargo, la aplicación del agrupamiento difuso proporciona ventajas adicionales en la identificación difusa ya que la pertenencia de los datos experimentales a las clases puede servir para obtener las funciones de validez.

3.3. CONSIDERACIONES TEÓRICAS PARA LA OBTENCIÓN DE MODELOS DIFUSOS USANDO TÉCNICAS DE AGRUPAMIENTO

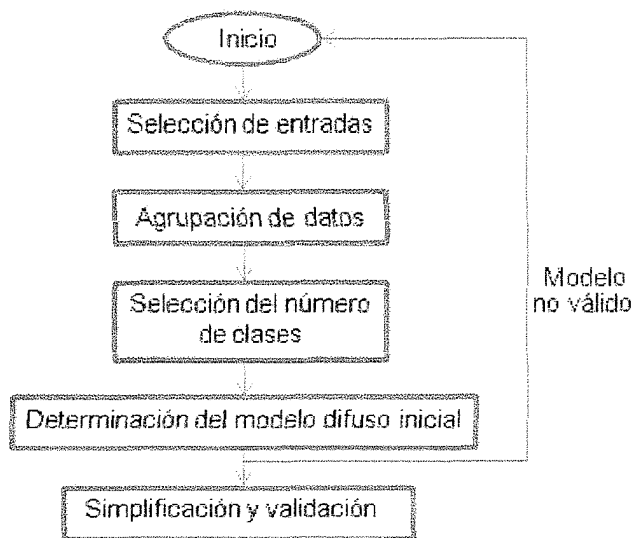


Figura 3.3: Esquema sistematizado de identificación de modelos difusos

www.bdigital.ula.ve

En base a lo anteriormente descrito, pareciera que la construcción de un modelo difuso aproximado desde datos experimentales es una tarea simple y pudiera ser resuelta de manera automática. No obstante, tareas como la selección de entradas y la simplificación y validación de modelos no están completamente resueltas. Además, una vez elegido el tipo de modelo difuso y el algoritmo de agrupamiento apropiado al sistema a identificar, se han de definir ciertos parámetros como: el número de clases c , el parámetro de difusidad m , la tolerancia ϵ , la norma A a emplear, etc.

El primer intento de sistematización de la identificación de modelos difusos mediante técnicas de agrupamiento que ha sido utilizado por la mayoría de los autores es ampliado en dos pasos en (Babuska 1996): la simplificación del modelo y la validación del mismo. Aunque no explicitados en (Sugeno and Yasukawa 1993), estos dos nuevos pasos parecen supuestos, ya que su fin es el de conseguir un modelo cualitativo y altamente interpretable, cuya lectura de parte del experto sea de por sí una forma de validación del modelo. A continuación se detallan cada uno de los pasos en los que consiste la identificación del modelo, tal y como lo proponen en (Babuska 1996) (Sugeno and Yasukawa 1993).

3.3.1. Diseño de experimentos

Este paso inicial es importante en cualquier método de identificación. Los métodos de excitación mediante señales binarias pseudoaleatorias no son apropiados para los sistemas no lineales que serán objeto de estudio en esta trabajo.

Según (Ljung 1999), lo conveniente es excitar al sistema a identificar por un rango completo de señales, tanto en amplitud como en frecuencia, y por tal motivo señales de dos valores de amplitud no son apropiadas. Una señal de excitación con componentes senoidales variables en amplitud y frecuencia o escalones de amplitud y anchura aleatoria puede ser una buena solución. A la señal de excitación elegida es conveniente agregar un ruido blanco de pequeña amplitud para garantizar la adecuada excitación de la dinámica del proceso. Es importante considerar la elección de un período de muestreo adecuado y el tiempo que debe durar el experimento, así como las posibles ventajas que ofrece la normalización de los datos.

3.3.2. Selección de la estructura

Además de la forma de tratar la dinámica del proceso y la granularidad a emplear en el modelo, en este paso se identifican cuales son las variables de entrada y salida relevantes para el problema de modelado. Aunque el experto en el sistema puede dar las claves en este paso, se puede automatizar la elección de las variables comparando varias propuestas desde el punto de vista de algún índice. Su resolución no es directa sino que requiere técnicas iterativas, lo que lleva a que sus valores definitivos no sean determinados hasta el último paso de construcción del modelo difuso, con lo que todo el proceso de obtención del modelo se convierte en un gran problema de iteración, con sus múltiples partes relacionadas entre sí y sin ser posible la determinación aislada y precisa de los parámetros referentes a la estructura.

Elección de Variables de Entrada: aunque muchos de los métodos revisados en la bibliografía ((Nakamori and Ryoke 1994) (Klawonn and Keller 1998) (Klawonn and R. 1997) (Delgado, Gomez, and F. 1997)) suponen, para la construcción del modelo difuso de un sistema, que las entradas y salidas son conocidas, generalmente no está claro en los sistemas multivariables qué variables han de ser elegidas como entradas al sistema (Sugeno and Yasukawa 1993). El conocimiento de los expertos es fundamental en este tema, aunque los métodos estadísticos de correlación también pueden jugar un papel importante.

En las soluciones heurísticas a este problema, se parte de un número N de

3.3. CONSIDERACIONES TEÓRICAS PARA LA OBTENCIÓN DE MODELOS DIFUSOS USANDO TÉCNICAS DE AGRUPAMIENTO

posibles candidatos a ser entradas del sistema y se estudia el efecto de las distintas combinaciones de variables de entrada. Con la finalidad de no tener que usar todas las posibles combinaciones, se parte de la selección de algunas entradas de entre las candidatas y se van incrementando el número de entradas una a una, teniendo algún criterio. En (Diez 2003) se emplea un índice de no-significatividad de las variables. Otra opción al problema de la selección de entradas, de mayor carga computacional pero mas preciso, es el llamado criterio de regularidad que se estudia en (Delgado, Gomez, and F. 1997).

Representación de la dinámica del proceso: en este paso se transforma el problema de identificación de un sistema dinámico en un problema de regresión estática. La idea es realizar alguna transformación desde el dominio temporal al de las variables de estado, de manera que éstas determinen el estado del sistema sin necesidad del tiempo. En el caso de tratar con sistemas discretos, una solución sencilla y que da buenos resultados, consiste en tratar todas las muestras disponibles de una misma variable u como variables independientes ($u(t), u(t-1), u(t-2), \dots$) con lo que todo el sistema se convierte en estático.

Granularidad del modelo difuso: este parámetro está relacionado tanto con el número de reglas como con el número de términos lingüísticos para cada variable. Es el paso que en la literatura se le llama normalmente identificación de la estructura. El valor que inicializa el proceso de determinación de estos valores, si se usan métodos de agrupamiento, es el número de clases inicial. Con los métodos de simplificación de modelos difusos que serán estudiados más adelante se disminuirá, si es posible, el número de conjuntos difusos en las reglas mediante medidas de similaridad y el número de reglas con técnicas de detección de redundancia.

3.3.3. Agrupación de datos

Una vez se tienen identificadas cuales son las entradas y salidas de importancia para el sistema, se realiza una partición de los datos mediante las técnicas de agrupamiento, con el fin de identificar posibles regímenes de funcionamiento en el espacio de datos elegido. Cada uno de los grupos de datos detectados tiene, generalmente, un comportamiento similar y se podrá obtener en un paso posterior un modelo común a todos ellos, lineal o no.

La primera limitación puede presentarse en base a la disposición de los datos con los que se va a trabajar. La recolección de datos debe tratar de agrupar la mayor cantidad de datos cubriendo todo el espacio que se desea analizar.

Otro posible problema surge con elección del algoritmo de agrupamiento. La diferencia entre la mayoría de los algoritmos de agrupamiento difuso estriba en el tipo de norma elegida y la forma de la clase, con lo que el mismo grupo de datos puede dar lugar a diferentes agrupaciones dependiendo del algoritmo seleccionado. Ninguno de ellos en principio es más conveniente que otros, sino que cada uno será más apropiado en ciertas circunstancias. La información que a priori se puede tener sobre la forma de las clases puede ser fundamental, ya que se han visto métodos de agrupamiento que buscan elipsoides de tamaños similares, elipsoides de tamaños variables, prototipos lineales, etc. Inclusive hay autores que desarrollan su propio algoritmo de agrupamiento para la identificación de modelos difusos, en su aplicación concreta como el agrupamiento hiperelipsoidal (Nakamori and Ryoke 1994) o modifican los algoritmos existentes como las variaciones del GK y GG en (Klawonn and Keller 1998)(Klawonn and R. 1995). En particular, en el caso de modelado de sistemas no lineales para su posterior aplicación al control de sistema, los métodos de agrupamiento más idóneos, se requiere aplicar técnicas de control lineal a cada grupo de datos, serán aquellos que poseen consecuentes lineales (clases lineales o hiperplanos) (Krishnapuram and Freg 1992).

3.3.4. Selección del número de clases

Excepto un número reducido de algoritmos de agrupamiento, la gran mayoría necesita saber, con mayor o menor certeza, el número de clases para poder iniciar el proceso de iteración. Una solución posible al problema del número de clases inicial es realizar un estudio previo de los datos con algún algoritmo capaz de detectarlos como Chain-map, max-min o agrupación por eliminación (Gracia, Moreno, and Zapatero 2004). El valor obtenido en esta etapa de inicialización se introducirá en el algoritmo que se desee emplear. Sin embargo, existen dos metodologías más formales que permiten obtener un número óptimo de clases:

- Comenzar con un número de clases suficientemente elevado y realizar uniones entre clases compatibles con el algoritmo de unión de clases compatibles (CCM) es una posibilidad a tener en cuenta.
- Realizar la agrupación de los datos para diferentes c , y usar posteriormente medidas de validez para determinar la bondad de las particiones obtenidas.

La metodología CCM usa las propiedades de cualquier algoritmo para la detección de líneas y las mejora uniendo clases compatibles. Dos clases se consideran compatibles si ambas están en el mismo plano y la distancia entre ellas es pequeña comparada con su tamaño. Los pasos que deben seguirse se muestran a continuación:

- Elegir un número c_{max} de clases.

3.3. CONSIDERACIONES TEÓRICAS PARA LA OBTENCIÓN DE MODELOS DIFUSOS USANDO TÉCNICAS DE AGRUPAMIENTO

- Aplicar el método de agrupamiento (Por ejemplo GK) hasta su convergencia.
- Determinar los autovalores $\{\lambda_{i1}, \dots, \lambda_{in}\}$ y autovectores $\{\Phi_{i1}, \dots, \Phi_{in}\}$ correspondiente de cada clase i .
- Aplicar las condiciones de compatibilidad, por parejas i y j , a las clases obtenidas con el fin de obtener conjuntos de clases compatibles. Estas condiciones de compatibilidad se muestran a continuación:
 - Si las clases son paralelas entonces:

$$|\Phi_{in}^T \Phi_{jn}| \geq k_1, k_1 \cong 1 \quad (3.37)$$

- Las normales a los hiperplanos definidos por las clases deben ser ortogonales a la línea que une a ambas clases:

$$\left| \frac{(\Phi_{in}^T + \Phi_{jn})(c_i - c_j)}{2} \right| \leq k_2, k_2 \cong 0 \quad (3.38)$$

- Si las clases están cercanas entonces:

$$\frac{\|c_i - c_j\|}{\sqrt{\lambda_i} + \sqrt{\lambda_j}} \leq k_3, k_3 \in [2, 4] \quad (3.39)$$

- Unir cada uno de los grupos compatibles en una única clase.
- Recalcular los centros y las matrices de covarianza de cada clase.
- Repetir el proceso desde el paso 2 hasta que no existan más clases compatibles.

Existen modificaciones al algoritmo básico de CCM presentado y que consisten en la variación del criterio de compatibilidad de agrupamiento. Por ejemplo, en (Babuska 1996) se relajan las condiciones de compatibilidad eliminando la segunda condición establecida en la ecuación (3.38), con lo que se permite la unión de clases con datos contaminados por el ruido.

Con respecto a las medidas de validez, un buen estudio sobre este tema, se puede encontrar en (Bezdek 1973). Dentro de los índices más importantes se destacan el criterio de varianza y el de dispersión (Sugeno and Yasukawa 1993). El criterio de varianza se define como:

$$S(C) = \sum_{i=1}^c \sum_{k=1}^N (\mu_{ik})^m (\|z_k - c_i\|^2 - \|c_i - \bar{z}\|^2) \quad (3.40)$$

Y el índice de dispersión se define a través de las tres siguientes ecuaciones:

$$S(C) = \sum_{i=1}^c \sum_{k=1}^N (\mu_{ik})^m (\|z_k - c_i\|^2 - \|c_i - \bar{c}\|^2) \quad (3.41)$$

$$\bar{c} = \left(\frac{1}{\sum_{i=1}^c \sum_{k=1}^N (\mu_{ik})^m} \right) \left(\sum_{i=1}^c \sum_{k=1}^N (\mu_{ik})^m z_k \right) \quad (3.42)$$

$$c_i = \left(\frac{1}{\sum_{k=1}^N (\mu_{ik})^m} \right) \left(\sum_{k=1}^N (\mu_{ik})^m z_k \right) \quad (3.43)$$

En donde N es el número de datos, c es el número de clases, z_k es el k -ésimo vector de los datos, \bar{z} es la media de los datos y c_i es el centro de la i -ésima clase. En ambos casos, el número de clases se determina incrementando c hasta que $S(c)$ alcanza un mínimo local. La diferencia entre ambos índices es la extensión del vector de medias \bar{z} al vector difuso de medias \bar{c} . La diferencia entre el resultado de los dos índices se hace considerable para grandes valores de m .

3.3.5. Modelo difuso inicial

La clasificación de los datos realizada con los métodos descritos anteriormente permite determinar un modelo lineal para cada una de las clases detectadas con lo que se obtendría una buena aproximación del modelo no lineal complejo bajo estudio. Esta determinación de modelos difusos basados en particiones ha sido de mucha utilidad (Bezdek 1973). Es importante destacar que también existe la posibilidad de, por ejemplo, hacer un modelo basado en redes neuronales para cada clase (Babuska 1996) siendo la salida total del sistema una ponderación del resultado obtenido por cada red.

En cuanto a la determinación de modelos difusos, en base a los resultados del proceso de agrupamiento de los datos, existen dos opciones de obtener las funciones de pertenencia del antecedente:

- Proyectar los conjuntos difusos definidos punto a punto en la matriz de partición difusa resultante sobre cada una de las variables del antecedente tal y como se lleva a cabo en (Klawonn and R. 1997)(Sugeno and Yasukawa 1993).
- Calcular los grados de pertenencia directamente en el espacio de las variables de entrada (Delgado, Gomez, and F. 1997), usando la norma empleada en el algoritmo de agrupamiento.

3.3. CONSIDERACIONES TEÓRICAS PARA LA OBTENCIÓN DE MODELOS DIFUSOS USANDO TÉCNICAS DE AGRUPAMIENTO

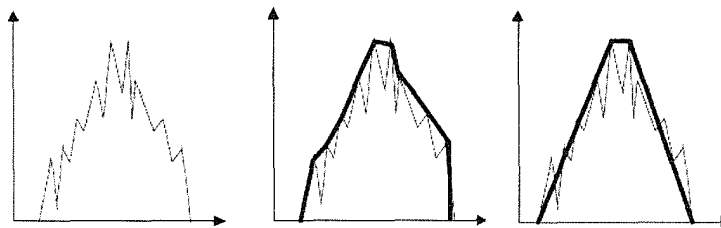


Figura 3.4: Función de pertenencia proyectada y aproximaciones

El primer caso se denomina generación de funciones de pertenencia del antecedente por proyección y básicamente consiste en proyectar la matriz de partición difusa obtenida luego de finalizar el proceso de iteración sobre las variables del antecedente de las reglas a construir. Para el caso de identificación de un sistema (SISO) de una sola entrada y una sola salida, cada columna de la matriz de partición difusa resultante representa el conjunto difuso unidimensional la variable de entrada. Ahora bien, en el caso de sistemas (MISO) cada columna de la matriz de partición difusa contiene el conjunto difuso multidimensional que debe ser proyectado sobre cada una de las variables del antecedente. En ambos casos, por lo general, se obtienen errores en el sentido de pérdida de información sobre todo en el caso de la proyección del conjunto difuso multidimensional. Normalmente es necesario suavizar los conjuntos difusos obtenidos para lograr obtener conjuntos difusos convexos previamente definidos (funciones triangulares, trapezoidales, etc.) tal y como la muestra la figura 3.4. Esta pérdida de información durante el proceso de proyección es lo que se denomina error de reconstrucción (Klawonn and R. 1997).

El segundo caso trata sobre la generación de las funciones de pertenencia del antecedente en forma directa en el espacio de entradas en donde se plantea como objetivo la medición de la distancia de cada punto x_k al prototipo de cada clase y transformar dicha distancia en un grado de pertenencia aplicando alguna expresión similar a la definida para el algoritmo FCM en la ecuación 3.27 o expresiones similares en otros algoritmos de agrupamiento, con lo que se da un valor de pertenencia de cada punto a cada clase. Esto conlleva a la desventaja de su alta carga computacional pues, para evaluar la pertenencia de nuevos datos, cada vez hay que buscar entre los valores de las funciones de pertenencias identificadas el punto más cercano al cual se pretende evaluar la pertenencia, asignando esa pertenencia o bien interpolando entre varias pertenencias si hay más de un punto a la misma distancia.

CAPÍTULO 3. IDENTIFICACIÓN DE MODELOS DIFUSOS A PARTIR DE DATOS DE ENTRADA-SALIDA

Con respecto a la estimación de los parámetros del consecuente del modelo difuso TS, la solución habitual es aplicar a los resultados del agrupamiento técnicas de mínimos cuadrados que minimicen el error de predicción del modelo global o de cada uno de los subespacios aproximados por cada clase. Otra opción bastante común es plantear problemas de optimización que sean resueltos usando algoritmos genéticos como en (Delgado, Gomez, and F. 1997).

Cuando se usa para la agregación de los resultados de los consecuentes de las reglas difusas la fórmula de la media difusa (2.17) (definida en el capítulo 2), la estimación de los parámetros del consecuente se puede obtener resolviendo un problema lineal de mínimos cuadrados (Diez 2003). El grado de cumplimiento de la i -ésima regla $\beta_i(x)$ mediante la matriz de partición difusa U . Ahora bien, para expresar (2.17) en forma matricial, se define una matriz diagonal Γ_i en el espacio $N \times N$:

$$\Gamma_i = \begin{bmatrix} \Upsilon_{i1} & 0 & \cdots & 0 \\ 0 & \Upsilon_{i2} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \Upsilon_{iN} \end{bmatrix} \quad (3.44)$$

en donde Υ_{ik} es un grado de pertenencia normalizado según:

$$\Upsilon_{ik} = \frac{\beta_{ik}}{\sum_{j=1}^K \beta_{jk}} \quad (3.45)$$

Llamando X' a la matriz compuesta por las matrices Γ_i y X_e

$$X' = [[\Gamma_1 X_e]; [\Gamma_2 X_e]; \dots; [\Gamma_c X_e]] \quad (3.46)$$

con X_e la matriz regresora (vectores de entrada) extendida por un columna unitaria:

$$X_e = \begin{bmatrix} x_1^T & 1 \\ x_2^T & 1 \\ \vdots & \vdots \\ x_N^T & 1 \end{bmatrix} \quad (3.47)$$

y denotando a θ' como:

$$\theta' = [\theta_1^T; \theta_2^T; \dots; \theta_c^T]^T \quad (3.48)$$

en donde $\theta_i = [a_i, b_i]$, donde $1 \leq i \leq c$. El problema resultante de mínimos cuadrado es:

$$y = X' \theta' + \epsilon \quad (3.49)$$

Y la solución como se demostró en el capítulo 2 es:

$$\theta' = [(X')^T X']^{-1} (X')^T y \quad (3.50)$$

3.3. CONSIDERACIONES TEÓRICAS PARA LA OBTENCIÓN DE MODELOS DIFUSOS USANDO TÉCNICAS DE AGRUPAMIENTO

Los parámetros a_i y b_i se obtienen de 3.48 mediante:

$$a_i = [\theta'_{q+1}; \theta'_{q+2}; \dots; \theta'_{q+p}]^T, b_{i+1} = \theta_{q+p+1} \quad (3.51)$$

En donde $q = (p + 1)(i - 1)$.

3.3.6. Índices de desempeño

La medida del rendimiento global para los modelos difusos identificados usando los algoritmos de agrupamiento identificados es una tarea de gran importancia, no siendo tan común la validación de la interpretabilidad de los modelos identificados. En esta sección se presentan dos índices idóneos para la validación de los modelos locales, con la intención de poder realizar posteriormente un análisis no sólo cualitativo, sino cuantitativo de los algoritmos que se desee. El primero de ellos es el índice de compacidad el cual sirve para estudiar la forma de las funciones de pertenencia en lo que respecta a su compacidad (Ahvenlampi and Diez 2003).

Se entiende que una función de pertenencia tiene una forma compacta si las altas pertenencias se dan en el entorno del prototipo y las bajas pertenencias lejos del prototipo, sin extenderse demasiado las zonas de pertenencias no nula en el universo de discurso. El índice calcula entre la pertenencia asociada a cada dato y su distancia al centro (prototipo) de la clase, realizando la suma de todos ellos, mediante la ecuación:

$$IC_i = \sum_{k=1}^N \mu_{ik}(z_k - c_i)^2 \quad (3.52)$$

donde IC_i es el índice de compacidad para la clase i , N es el número de datos, z_k es un dato y c_i es el prototipo. La determinación de este índice equivale a calcular el momento de inercia del contorno de la función de pertenencia en torno al eje perpendicular al hiperplano (n-1) dimensional del universo de discurso que pasa por el prototipo.

El segundo índice es el de solapamiento a través del cual se busca determinar si las funciones de pertenencias están poco solapadas o no. Para ello, en primer lugar, se reordena la matriz de partición μ_{ik} de forma que cada una de las columnas tenga las pertenencias ordenadas de forma descendente en una matriz μ_{jk}^{IS} , siendo la primera fila de la columna j la mayor pertenencia a una clase de dicho dato, la segunda fila la segunda mayor pertenencia del dato k a una clase y así sucesivamente hasta llegar a la menor pertenencia del dato k a una clase que se encontrará en la fila j . Evidentemente, los índices i y j no tienen relación alguna,

ya que j indica el orden de importancia de una pertenencia e i es una clase concreta (Ahvenlampi and Diez 2003).

Una vez obtenida la matriz μ_{jk}^{IS} se procede a calcular el j -ésimo índice de solapamiento como la media de los valores de la fila correspondiente j con:

$$IS_j = \frac{\sum_{k=1}^N \mu_{jk}^{IS}}{N} \quad (3.53)$$

quedando toda la información almacenada en la matriz columna IS con c filas:

$$IS = [IS_1, IS_2, \dots, IS_c]^T \quad (3.54)$$

El análisis de este índice matricial se realiza por comparación de las diversas entradas de IS . Un elevado valor de IS_1 y bajos valores en el resto de los términos indica poco solapamiento. En cambio, valores similares en todas las entradas de IS indican un elevado solapamiento.

3.4. Modelación neurodifusa

3.4.1. Generalidades

Los sistemas neurodifusos son una integración de las ventajas inherentes a los enfoques de las redes neuronales y lógica difusa lo que permite construir sistemas de toma de decisiones aun más inteligentes. Los sistemas neurodifusos surgen de la mano de J.S. Yang a principio de los noventa combinando las ventajas de los enfoques mencionados (Jang 1993). A continuación se muestra una de las definiciones más aceptadas de un sistema neurodifuso (Jang, Sun, and Mizutani 1997):

Un sistema neurodifuso es aquel sistema difuso cuyas reglas Si-Entonces se encuentran representadas en una estructura de red a través de la cual se aplican algoritmos de aprendizaje derivados del área de las redes neuronales artificiales.

A finales de la década de los noventa ya se habían desarrollado varios sistemas híbridos neurodifusos que se clasificaban en dos grandes grupos: redes neuronales dotadas con la capacidad de manejar información difusa (FNN, Fuzzy - Neural Network) y sistemas difusos combinados con redes neuronales para mejorar algunas de sus características (NFS, Neural - Fuzzy Systems).

En un sistema FNN, bien las señales de entradas, los pesos de las conexiones y / o salidas son subconjuntos difusos o conjuntos de valores de pertenencia de conjuntos difusos como sucede por ejemplo en (Mitra and Pal 1995)(Lee and Lee

1975)(Pal and Mitra 1992). En general, estas variables se modelizan empleando valores lingüísticos tales como "bajo", "medio", etc. o números difusos o intervalos. Las redes neuronales con neuronas difusas también se denominan FNN y son capaces de procesar información difusa.

Por otra parte, un sistema neurodifuso del tipo NFS está diseñado para realizar el proceso de razonamiento difuso y, en él, los pesos de conexión de la red se corresponden con los parámetros de razonamiento difuso. Un ejemplo de estos sistemas se puede encontrar en (Nauck, Klawonn, and Kruse 1997)(Keller and Tahani 1992)(Takagi, Suzuki, Koda, and Kojima 1992)(Wang and Mendel 1992). El uso de algoritmos de aprendizaje del tipo propagación hacia atrás hace que los sistemas NFS puedan identificar reglas difusas y aprender funciones de pertenencia del razonamiento difuso. En resumen, un sistema de tipo NFS debe ser capaz de aprender reglas lingüísticas y/o funciones de pertenencia u optimizar las existentes y para ello hay tres posibilidades(Nauck, Klawonn, and Kruse 1997):

- El sistema se inicia sin reglas, creándose nuevas reglas hasta que el problema de aprendizaje queda resuelto. La creación de una nueva regla se producirá por la entrada al sistema de un patrón de entrenamiento que no está suficientemente cubierto por la base de reglas actual.
- El sistema se inicia con todas las reglas, las cuales pueden ser creadas por partición de variables y eliminadas de la base de reglas, gracias a una evaluación del desempeño de las mismas.
- El sistema se inicia con una base de reglas con un número fijo de reglas. Durante el aprendizaje, las reglas se sustituyen a través de un proceso de optimización.

Diferentes esquemas híbridos neurodifusos han sido propuestos por varios autores y cada día se están desarrollando nuevos tipos de hibridaciones que implican síntesis a distintos niveles. Sin embargo, todas estas metodologías pueden clasificarse dentro de algunos de los grupos que se encuentran a continuación:

- Incorporación de incertidumbre y borrosidad en el marco de una red neuronal: emborronado de los datos de entrada, asignación de etiquetas difusas para las muestras de entrenamiento, posible emborronado en el procedimiento de aprendizaje, obtención de productos de redes neuronales en términos de conjuntos difusos.
- Diseño de redes neuronales guiadas por el formalismo de lógica difusa: diseño de redes neuronales para implementar lógica difusa y toma de decisiones

difusas, creación de funciones de pertenencia que representen conjuntos difusos.

- Cambio en las características básicas de las neuronas: neuronas diseñadas para realizar diversas acciones utilizando la teoría de conjuntos difusos tales como unión, intersección, etc. en lugar de operaciones estándar (multiplicación, suma, etc.).

Finalmente, es importante destacar que existe una tendencia bien interesante a desarrollar sistemas neurodifusos evolutivos (Evolving Neuro-Fuzzy Systems). En este tipo de sistemas neurodifusos, la estructura de los mismos va cambiando a lo largo del tiempo, a diferencia de los sistemas neurodifusos anteriores en los cuales solamente se modificaban los parámetros a lo largo del tiempo. Ejemplo de estos sistemas son los trabajos (Kasabov and Song 2002)(Song and Kasabov 2006)(Song and Kasabov 2005)(Lin and Chen 2008). No hay que confundir esta clase de sistemas neurodifusos con aquellos que modifican sus parámetros a través de computación evolutiva (algoritmos genéticos), como por ejemplo en (Farag, Quintana, and G. 1998)(Nearchou 1999).

3.4.2. Modelo ANFIS

La mayoría de los modelos de aprendizaje existentes en los sistemas de inteligencia artificial desarrollados y aplicados hasta ahora, especialmente en el ámbito del softcomputing y particularmente en los sistemas de razonamiento neurodifusos, están basados en métodos de inferencia inductivos. Estos métodos de inferencia obtienen un modelo general a partir del conjunto de datos que representa el espacio del problema, aplicando posteriormente este modelo a los nuevos datos a procesar o utilizando dicho modelo como base para el diseño del sistema de control. En otras palabras el aprendizaje inductivo realiza una extrapolación a partir de un determinado conjunto de ejemplos, para que se puedan hacer predicciones exactas sobre ejemplos futuros.

El sistema neurodifuso tomado como ejemplo de sistema neurodifuso inductivo es el conocido como Adaptive Network Based Fuzzy Inference System (ANFIS), en el que las reglas difusas se obtienen a partir de datos del propio proceso (Jang 1993). El sistema ANFIS es uno de los primeros sistemas neurodifusos y su principio se basa en la extracción de reglas difusas a través de una estrategia descomposicional, donde las reglas son extraídas en cada nivel de una red neuronal. Una vez obtenidas las reglas, éstas deben proporcionar la información necesaria del comportamiento global del proceso.

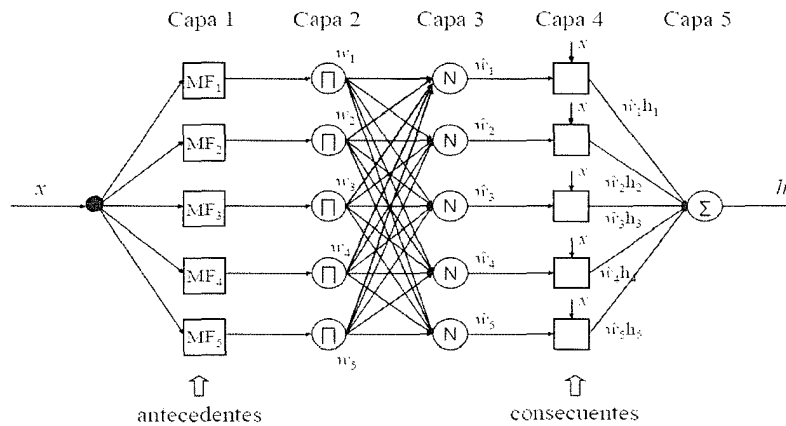


Figura 3.5: Arquitectura del sistema ANFIS

Como todo sistema de inferencia difuso, el sistema neurodifuso ANFIS desarrollará los siguientes pasos: emborronado de las entradas en forma de variables físicas del sistema, computación del grado de satisfacción de cada regla lingüística, conjunción de las diferentes premisas o antecedentes y desemborronado. Todo este proceso se realiza en las distintas capas secuenciales de la red neuronal.

ANFIS implementa el modelo de Takagi-Sugeno para la estructura de las reglas Si-Entonces del sistema neurodifuso. La arquitectura de ANFIS dispone de cinco capas, tal y como se muestra en la figura anterior. En esta arquitectura, todos los nodos de una misma capa tienen la misma función. Los nodos representados con cuadros son nodos cuyos parámetros son ajustables, mientras que los nodos representados por círculos son nodos fijos. Las funciones de cada capa son las siguientes:

- La primera capa representa las funciones de pertenencia del sistema difuso.
- La segunda y tercera capa contienen los nodos que forman el antecedente de cada regla difusa.
- La cuarta capa calcula la función de Takagi - Sugeno para cada regla difusa.
- La quinta capa es la de salida y calcula el peso global de la salida del sistema.

A continuación se presenta ANFIS para el caso particular de un sistema de una entrada y una salida (Denai and Palis 2007). En la primera capa se produce la difusificación y la salida de cada nodo se representa por $O_{l,i}$ donde i es el i -ésimo nodo de la capa l :

$$O_{1,i} = \mu_{A_i}(x) \tag{3.55}$$

donde x es la variable de entrada al nodo y A_i es el conjunto difuso asociado al nodo. Si se utiliza una función *Gaussiana* como función de pertenencia difusa, se obtendría la siguiente expresión, donde a_i , b_i y d_i son los parámetros antecedentes ajustables:

$$\mu_{A_i}(x) = \frac{1}{1 + [((x - d_i)/a_i)^2]^{b_i}} \quad (3.56)$$

En la segunda capa se multiplican las señales de entrada. La salida de esta capa es el resultado de aplicar la regla del máximo. La tercera capa normaliza la importancia de cada regla.

$$O_{2,i} = w_i = \mu_{A_i}(x) \quad (3.57)$$

$$O_{3,i} = \bar{w}_i = \frac{w_i}{w_1 + w_2} \quad (3.58)$$

La cuarta capa calcula el consecuente, es decir, la función de Takagi - Sugeno para cada regla difusa, donde m_i y c_i son los parámetros consecuentes.

$$O_{4,i} = \bar{w}_i h_i = \bar{w}_i (m_i x + c_i) \quad (3.59)$$

Por último la quinta capa realiza la desfusificación como sigue:

$$O_{5,i} = \sum_i \bar{w}_i h_i = \frac{\sum_i w_i h_i}{\sum_i w_i} \quad (3.60)$$

ANFIS utiliza como estrategia de aprendizaje la retropropagación hacia atrás de los errores para determinar el antecedente de las reglas. El consecuente de la regla se estima por medio del método de los mínimos cuadrados. En el primer paso o “paso hacia adelante”, los datos de entrada son propagados y los consecuentes óptimos son estimados por un procedimiento iterativo de mínimos cuadrados, mientras que los antecedentes permanecen constantes. En el segundo paso o “paso hacia atrás” se utiliza el procedimiento de retropropagación de errores para modificar los antecedentes mientras los consecuentes permanecen constantes. Este criterio se repite hasta que se alcanza la condición de parada (Criterio de error).

Cuando los valores de los antecedentes son fijos, la salida general del sistema puede expresarse como una combinación lineal de los consecuentes (3.61). Por otra parte, los antecedentes son actualizados por un criterio de “gradiente descendente” (3.63), siendo η la tasa de aprendizaje para a_{ij} .

$$h = [\bar{w}_1 x \quad \bar{w}_1 \quad \bar{w}_2 x \quad \bar{w}_2], \quad \begin{bmatrix} m_1 \\ c_1 \\ m_2 \\ c_2 \end{bmatrix} = \bar{X} \cdot \bar{W} \quad (3.61)$$

Si \bar{X} es una matriz invertible entonces:

$$\bar{W} = \bar{X}^{-1} \cdot h \quad (3.62)$$

$$a_{ij}(t+1) = a_{ij} - \eta \frac{\partial E}{\partial a_{i,j}} \quad (3.63)$$

3.5. Casos de estudios

Los dos casos de estudios de esta tesis tratan sobre los procesos de separación gas-líquido a través de equipos denominados separadores de producción y el calentamiento para el procesamiento del crudo a través de hornos de fuego directo. El crudo desde su formación contiene tres elementos básicos: agua, gas y petróleo como tal. En la fase de separación gas-líquido se busca separar completamente el gas del crudo para que posteriormente en la fase de calentamiento se incremente la temperatura de la fase líquida disminuyendo así su viscosidad. Una vez calentada la fase líquida (agua+petróleo), este pasa a una tercera fase que consiste en separar el agua del petróleo como tal a través de un equipo denominado tanque de lavado.

Para propósitos de venta, el petróleo no debe contener un corte de agua mayor que 5% de su volumen, de ahí radica la importancia de obtener modelos de gran exactitud para posteriormente poder diseñar controladores que cumplan con las especificaciones de funcionamiento requeridas por el diseñador. A continuación se mostrarán los resultados de la aplicación de las técnicas de agrupamiento difuso sobre los procesos mencionados tal y como se desarrolló en (Ramirez and Colina 2008a).

3.5.1. Separador de producción

Los separadores de producción en las empresas de petróleo tienen como función separar la fase gaseosa de la fase líquida. El crudo normalmente está conformado por tres componentes: gas, agua y petróleo. La fase líquida está conformada por el agua y el petróleo. Específicamente, existen algunos separadores especiales denominados trifásicos que separan los tres componentes de manera individual. Los separadores se clasifican de acuerdo a su configuración en separadores verticales, horizontales y esféricos y de acuerdo a su función en separadores de producción y separadores de prueba.

El fluido entra en el separador de producción y se contacta con un desviador de ingreso causando un cambio repentino en la separación bruta inicial de líquido

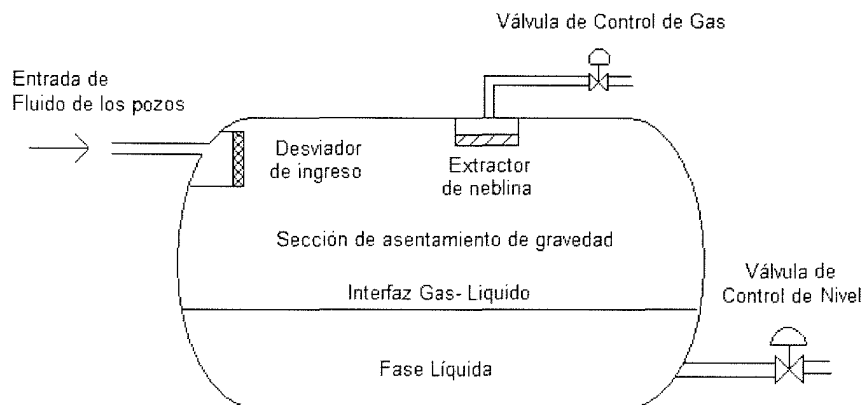


Figura 3.6: Componentes principales de un separador de producción horizontal

vapor tal y como se muestra en la figura 3.6. La gravedad hace que las gotas de líquido caigan de la corriente de gas al fondo del recipiente de recolección. Esta sección de recolección de líquido provee el tiempo de retención necesario para que el gas arrastrado evolucione del petróleo y suba al espacio de vapor. También esta sección provee volumen de oleada, si fuera necesario, para manejar los sobrepesos intermitentes de líquidos. Luego la fase líquida sale del recipiente a través de una válvula de control reguladora de nivel. En los separadores automatizados, el actuador de la válvula de control normalmente de tipo neumático con su convertidor de I/P es gobernado por un controlador, el cual, a través de la señal de nivel de líquido abre o cierra de manera continua la válvula para mantener un nivel fijado por el operador.

Los factores que afectan la separación de las fases son: el tiempo de residencia del separador, la densidad y viscosidad de los fluidos (la cual depende a su vez de la temperatura), la distribución de los tamaños de gotas de agua y petróleo en la entrada del equipo, la velocidad del gas y la presión de operación.

3.5.1.1. Resultados de la aplicación del algoritmo GK

Los datos operacionales de un separador de producción son monitorizados por un software especial llamado SCADA (Sistema de Control y Adquisición de Datos). Entre estos datos operacionales se tienen la temperatura del crudo en el separador, la presión del separador, el nivel instantáneo de crudi, el porcentaje de apertura de la válvula de control y los conmutadores de seguridad como son la detección de alto nivel de crudo, muy alta presión o muy bajo nivel de crudo.

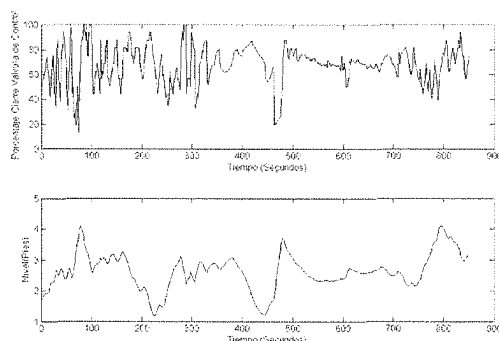


Figura 3.7: Datos medidos de entrada-salida del separador de producción

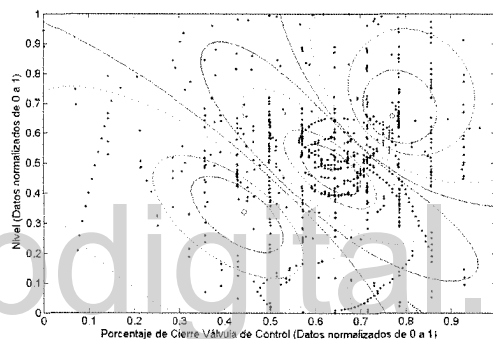


Figura 3.8: Agrupación de datos del proceso de separación usando el algoritmo Gustaffson-Kessel

El modelo seleccionado es de tipo regresión y tiene la forma $y(\tau + 1) = f(y(\tau), u(\tau))$, en donde $y(\tau)$ es el nivel instantáneo de crudo y el porcentaje de cierre actual $u(\tau)$ son las variables de entrada y la variable de salida es el nivel de crudo en la siguiente muestra $\hat{y}(\tau + 1)$. Se recolectaron un total de 856 datos para la identificación del modelo difuso del separador, los cuales se pueden apreciar en las gráficas de la figura 3.7.

Los datos de entrada $u(\tau)$ y $y(\tau)$ se agruparon usando el algoritmo Gustaffson-Kessel. Para ello se seleccionó el número de clases $c = 3$, $\epsilon = 0,01$, el índice de borrosidad $m = 2,2$ y el volumen de las clases $\rho = 1$. Se usó el toolbox Fuzzy Clustering And Data Analysis Toolbox de Matlab y los datos quedaron agrupados tal y como lo muestra la gráfica 3.8 y se requirió de 261 iteraciones para obtener la matriz de partición difusa definitiva. Previamente se habían agrupado los datos

CAPÍTULO 3. IDENTIFICACIÓN DE MODELOS DIFUSOS A PARTIR DE DATOS DE ENTRADA-SALIDA

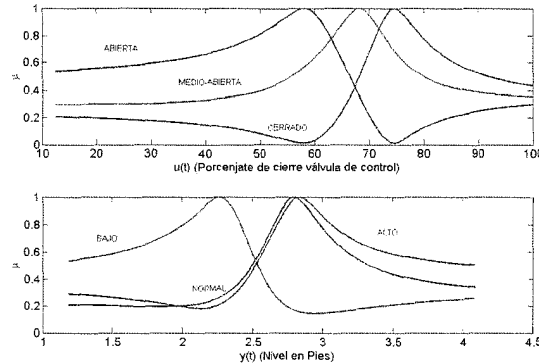


Figura 3.9: Proceso de separación: funciones de pertenencia de $y(\tau)$ y $u(\tau)$ (algoritmo GK)

usando cuatro clase pero aplicando el algoritmo CCM se detectó la compatibilidad de dos clases.

Para la determinación del modelo difuso, se usó el método de proyección de la matriz de partición difusa sobre cada una de las variables del antecedente. La gráfica 3.9 muestra las funciones de pertenencia del antecedente obtenidas de la proyección de la matriz de partición difusa resultante de la aplicación del algoritmo GK sobre los ejes de $y(\tau)$ y $u(\tau)$.

La determinación de los parámetros lineales del consecuente se realizó a través de técnicas de mínimos cuadrados usando la versión recursiva. La base de reglas obtenidas se muestra a continuación:

- Si $u(\tau)$ está Cerrado y $y(\tau)$ es Bajo Entonces $\hat{y}(\tau + 1) = -0,0025u(\tau) + 1,0029y(\tau) + 0,1649$
- Si $u(\tau)$ está Medio Abierto y $y(\tau)$ es Normal Entonces $\hat{y}(\tau+1) = -0,0029u(\tau) + 0,99y(\tau) + 0,2019$
- Si $u(\tau)$ está Abierto y $y(\tau)$ es Alto Entonces $\hat{y}(\tau + 1) = -0,0023u(\tau) + 0,9814y(\tau) + 0,2134$

La figura 3.10 muestra la gráfica de los modelos locales obtenidos. El modelo difuso TS obtenido fue validado con 450 datos entrada-salida distintos de los usados para crear el modelo. En la gráfica de la figura 3.11 la curva de color morado representa la salida real del proceso y la curva de rosado es la salida del modelo. La raíz cuadrada del error cuadrático medio es de 0.0701 y el índice de compacidad es

3.456 y los índices de solapamiento de cada modelo local (IS_1, IS_2, IS_3) son de 0.0234, 0.0567 y 0.0984.

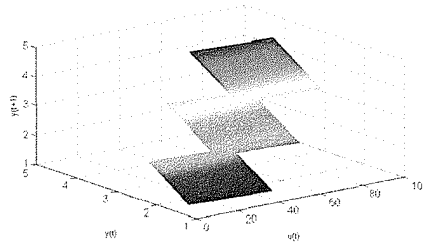


Figura 3.10: Modelos locales detectados (algoritmo GK).

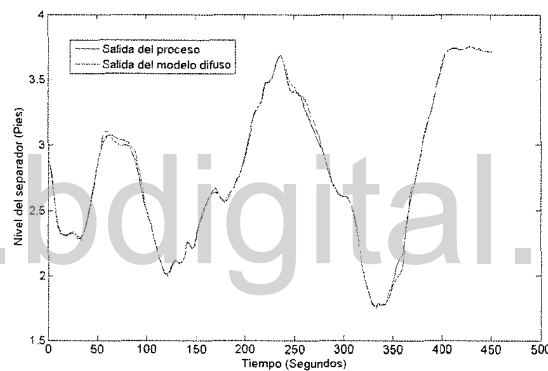


Figura 3.11: Validación del modelo difuso del separador de producción (algoritmo GK).

3.5.1.2. Resultados de la aplicación del algoritmo FCM

Análogamente a los mismos datos operacionales recolectados del separador de producción (representados en la figura 3.7) se les aplicó el algoritmo Fuzzy C Means. Para ello se seleccionó el número de clases $c = 3$, $\epsilon = 0,01$, el índice de borrosidad $m = 2,2$. Los datos quedaron agrupados tal y como lo muestra la figura 3.12.

Se usó el método de la proyección de la matriz de partición difusa sobre cada una de las variables del antecedente $y(\tau)$ y $u(\tau)$ para la determinación del modelo difuso TS. La figura 3.13 muestra las tres funciones de membresía para cada una de las variables de entrada. Luego al aplicar las técnicas de mínimos cuadrados para estimar los parámetros de los consecuentes, se obtuvo el siguiente resultado:

CAPÍTULO 3. IDENTIFICACIÓN DE MODELOS DIFUSOS A PARTIR DE DATOS DE ENTRADA-SALIDA

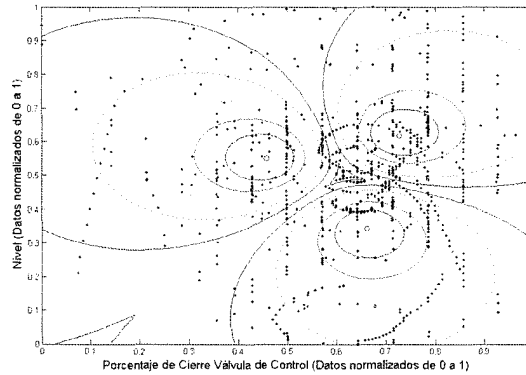


Figura 3.12: Agrupación de datos usando el algoritmo Fuzzy C Means

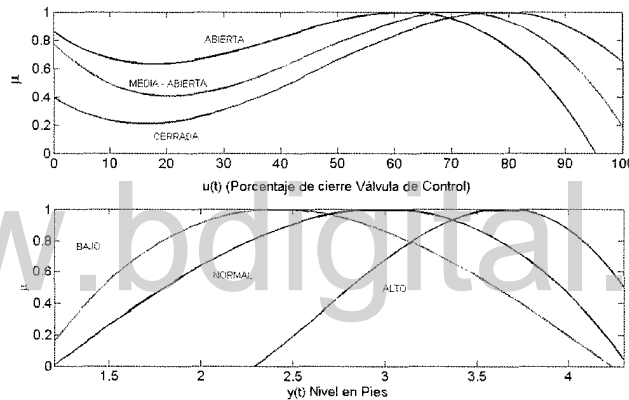


Figura 3.13: Proceso de separación: funciones de pertenencia de $y(\tau)$ y $u(\tau)$ (algoritmo FCM)

- Si $u(\tau)$ está Cerrado y $y(\tau)$ es Bajo Entonces $\hat{y}(\tau + 1) = -0,0045u(\tau) + 0,0056y(\tau) + 0,3345$
- Si $u(\tau)$ está Medio Abierto y $y(\tau)$ es Normal Entonces $\hat{y}(\tau+1) = -0,0018u(\tau) + 1,134y(\tau) + 0,1123$
- Si $u(\tau)$ está Abierto y $y(\tau)$ es Alto Entonces $\hat{y}(\tau + 1) = -0,0045u(\tau) + 1,1232y(\tau) + 0,3456$

La validación del modelo difuso TS obtenido se hizo con 450 datos de entrada-salida distintos a los usados para crear el modelo. La gráfica de la figura 3.14 muestra una muy buena capacidad de generalización del modelo. La raíz cuadrada del error cuadrático medio es de 0.0932 y el índice de compacidad es 8.756

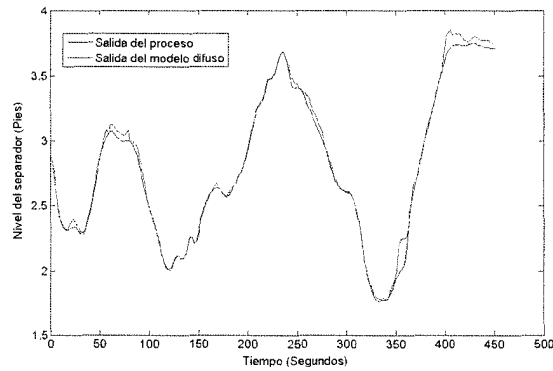


Figura 3.14: Validación del modelo difuso del separador de producción (Algoritmo FCM).

y los índices de solapamiento de cada modelo local (IS_1, IS_2, IS_3) es de 0.1032, 0.09876 y 0.230.

www.bdigital.ula.ve

3.5.1.3. Resultados de la aplicación del algoritmo ANFIS

Nuevamente sobre el mismo conjunto de datos de la figura 3.7, se determinó un modelo de tipo ANFIS. Para ello se definieron los siguientes parámetros: el número de épocas fue de 3000 y se usó el algoritmo de retropropagación de errores hacia atrás partiendo como premisa previa el uso de funciones de membresía de tipo gaussiana, campana y triangular. La siguiente tabla muestra las diversas configuraciones usadas y la más óptima fue el modelo ANFIS con cinco funciones de pertenencia de tipo triangular para las variables $y(\tau)$ y $u(\tau)$. La gráfica de la figura 3.15 muestra el comportamiento de la curva de error obtenida tras aplicar el algoritmo de retropropagación de errores hacia atrás y la raíz cuadrada del error cuadrático medio obtenida fue de 0.07 con un total de 1880 épocas de entrenamiento. Las funciones de pertenencias se muestran en la figura 3.16 y la base de reglas difusas que en total son 16 se encuentran en el apéndice B. En la figura 3.17 se muestra la validación del modelo difuso obtenido usando ANFIS se aprecia la muy buena generalización y exactitud del modelo obtenido.

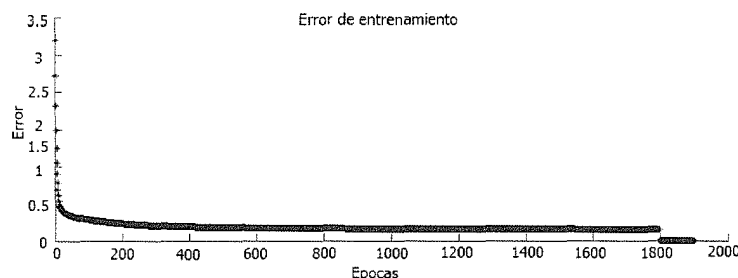


Figura 3.15: Comportamiento de la señal de Error (modelación ANFIS del separador)

3.5.2. Análisis de interpretabilidad de los modelos difusos identificados del separador

Para analizar la interpretabilidad, es conveniente revisar de manera breve los aspectos más resaltantes sobre este tema. Bodenhofer y Bauer (Bodenhofer and Bauer 2003)(Bodenhofer and Bauer 2000) plantean de manera formal la siguiente definición: “La interpretabilidad de un sistema es la posibilidad de estimar su comportamiento a partir de la lectura y comprensión de la descripción de su base de reglas”. Puesto que la legibilidad de las reglas depende a su vez de la legibilidad de las expresiones lingüísticas que estas incluyen, Bodenhofer considera que el análisis se debe realizar a nivel de particiones garantizando que estas respeten las relaciones de orden e inclusión entre conjuntos difusos.

En (Alonso and Magdalena 2008) se plantea que para garantizar una buena interpretabilidad, las características que deberían tener tales conjuntos difusos son las siguientes:

- Definir un número moderado de etiquetas, términos lingüísticos, por variable. Los psicólogos establecen 7 ± 2 como límite en el número de conceptos simultáneos que admite la capacidad de procesamiento humano.
- Asociar de manera clara un término lingüístico con cada función de pertenencia, de manera que los conjuntos difusos se distingan sin ambigüedades.
- Utilizar funciones de pertenencia que cubran todo el universo de discurso en que se define la variable, y que cumplan la propiedad de normalidad.
- Trabajar con conjuntos difusos convexos.

En base a estas premisas teóricas, en nuestro caso se tomará en cuenta la naturaleza de las funciones de membresía de las variables de los antecedentes y el

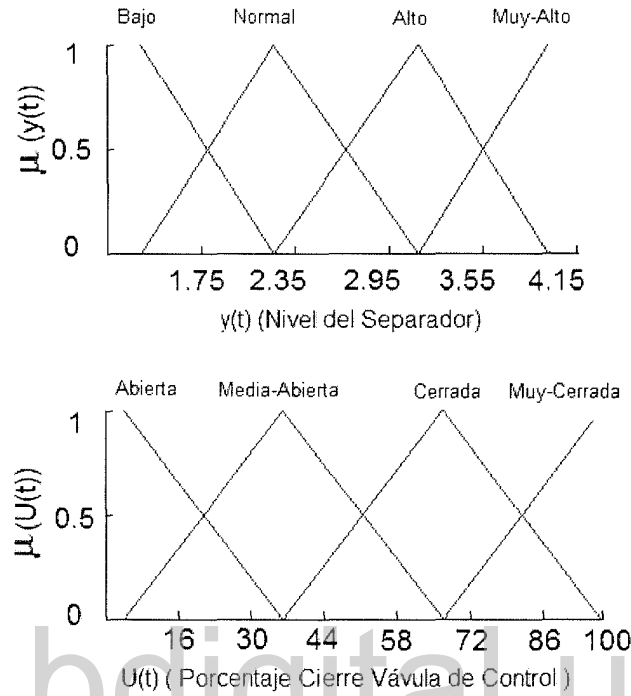


Figura 3.16: Funciones de membresia del antecedente (modelacin ANFIS del separador)

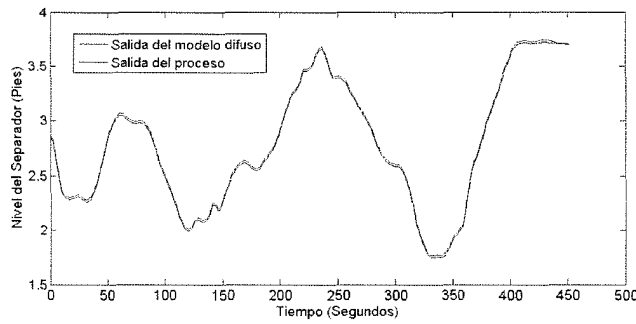


Figura 3.17: Validacin del modelo difuso del separador de produccin (tcnicas neurodifusas)

CAPÍTULO 3. IDENTIFICACIÓN DE MODELOS DIFUSOS A PARTIR DE DATOS DE ENTRADA-SALIDA

Tabla 3.1: Índices de desempeño de los modelos difusos del separador

Nombre de la Técnica	RSME	Índice de Solapamiento	Funciones de Membresía
GK	0.1103	$IS_1 = 0,0234, IS_2 = 0,0567, IS_3 = 0,0984$	No estándar (No convexas)
FCM	0.1406	$IS_1 = 0,1032, IS_2 = 0,09876, IS_3 = 0,230$	No estándar
ANFIS	0.094	No aplica	Triangular (estándar)

número total de reglas de los modelos difusos identificados tanto con las técnicas de agrupamiento como con el uso del algoritmo ANFIS. Claramente se observa que los términos lingüísticos de los antecedentes de los modelos difusos identificados usando las técnicas de agrupamiento presentan conjuntos difusos no estándares. De hecho, en el caso del algoritmo GK, se obtuvieron funciones de membresía no convexas.

La base de reglas de los modelos difusos obtenidos usando las técnicas de agrupamiento depende del número de clases seleccionadas. En ambos casos, se obtuvieron modelos con tres reglas que aunque resultan muy buenos aproximadores de funciones, no son idóneos para el diseño de controladores.

En la tabla 3.1 se muestran los índices de desempeño de los modelos difusos identificados para el separador de producción usando las técnicas de agrupamiento difuso y neurodifusas. Como se puede apreciar, los modelos difusos identificados usando ANFIS son un poco más exactos y los conjuntos difusos del antecedente están representados por funciones de membresías estándares de tipo triangular pero se obtuvo un modelo con 25 reglas. En cambio, los modelos difusos identificados con técnicas de agrupamiento a pesar de su sencillez y apenas tres reglas los antecedentes presentan funciones de membresía que no cumplen con la condición de normalidad. El índice de solapamiento en el caso del algoritmo FCM resultó menor en valor que el del algoritmo GK lo cual indica que las funciones de membresía están menos solapadas. La única ventaja del algoritmo GK sobre el FCM fue su exactitud.

3.5.3. Hornos de fuego directo

Los hornos de fuego directo son usados para calentar el crudo desde 90 grados Fahrenheit hasta 180 grados Fahrenheit. En términos formales, un horno de fuego directo o calentador por combustión es un intercambiador de calor en el que el fluido del proceso fluye dentro de los tubos y se calienta por radiación procedente de una llama de combustión y por convección desde los gases calientes que se forman en la reacción de combustión. Normalmente se dividen en tres partes:

- Sección radiante: en esta zona los tubos por donde fluye el líquido que se desea calentar están en presencia de la llama, la transmisión de calor es por radiación en un 80% aproximadamente y un 20% por convección de la circulación de gases calientes alrededor de los tubos.
- Sección de convección: los tubos están fuera del alcance de la llama. Los gases calientes de la combustión se dirigen por la parte externa del paquete de tubos. El calor transmitido es por convección cuando los gases calientes fluyen desde el hogar del horno hasta la chimenea. Los tubos están equipados con aletas para mejorar las condiciones de transmisión de calor.
- Sección de blindaje: está conformado por las primeras filas de tubos del área de convección que representan la zona de choque y en ella los tubos no tienen aletas.

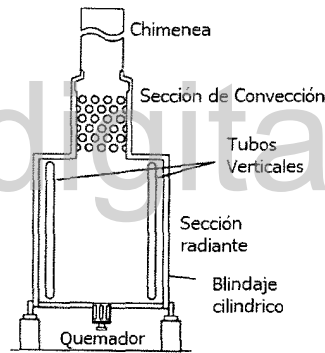


Figura 3.18: Horno de fuego directo

Un horno de fuego directo tipo cilindro vertical consiste en un casco cilíndrico aislado verticalmente con piso aislante y techo generalmente plano. Los quemadores se sitúan en la parte inferior mientras que los tubos se sitúan verticalmente con flujo ascendente y descendente tal y como se muestra en la figura 3.18.

3.5.3.1. Resultados de la aplicación del algoritmo GK

Los parámetros que normalmente se monitorizan en un horno de fuego directo son la temperatura de salida de crudo, la presión de entrada al horno, la presión de salida, la temperatura de la chimenea, el estado de operación de cada uno de los

CAPÍTULO 3. IDENTIFICACIÓN DE MODELOS DIFUSOS A PARTIR DE DATOS DE ENTRADA-SALIDA

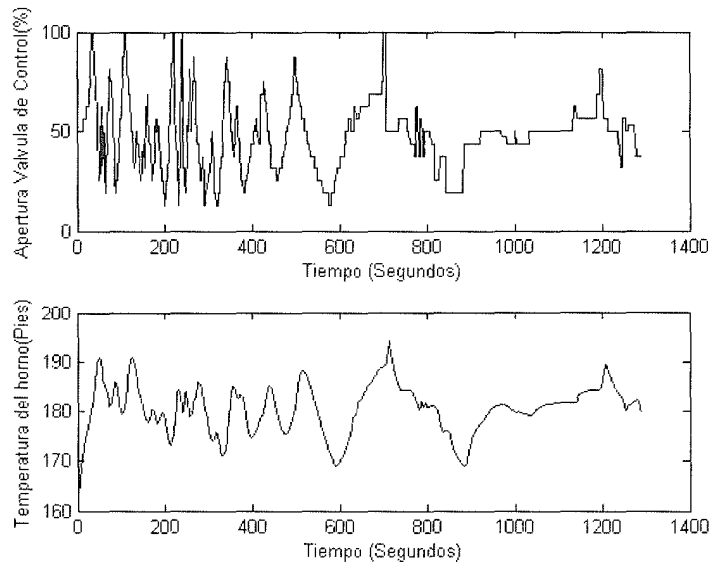


Figura 3.19: Datos medidos de entrada-salida del horno de fuego directo

quemadores y el flujo de entrada. Se recolectaron 1290 datos para la construcción del modelo difuso. Los datos obtenidos del SCADA se observan en la figura 3.19.

Para este caso y de manera análoga al proceso de separación se seleccionó un modelo de regresión de la forma $\hat{y}(\tau + 1) = f(y(\tau), u(\tau))$, en donde las variables de entrada son $y(\tau)$ y $u(\tau)$ que representan respectivamente la temperatura actual del crudo a la salida del horno y el porcentaje de apertura actual que controla el gas hacia los quemadores y la variable de salida es la temperatura del crudo a la salida del horno de la siguiente muestra $\hat{y}(\tau + 1)$. Sobre este conjunto de datos se aplicó el algoritmo GK obteniéndose como resultado la gráfica de la figura 3.20. Luego, al proyectar cada columna de la matriz de partición difusa sobre los datos de $y(\tau)$ y $u(\tau)$ se obtuvieron las funciones de membresía de la gráfica 3.21. La determinación de los parámetros lineales del consecuente se realizó a través de técnicas de mínimos cuadrados usando la versión recursiva. La base de reglas obtenidas se muestra a continuación:

- Si $u(\tau)$ esta Cerrada y $y(\tau)$ es Baja Entonces $\hat{y}(\tau + 1) = 0,0247u(\tau) + 0,9511y(\tau) + 7,6080$
- Si $u(\tau)$ esta Media Abierta y $y(\tau)$ es Normal Entonces $\hat{y}(\tau+1) = -0,0025u(\tau) + 0,9632y(\tau) + 6,8254$
- Si $u(\tau)$ esta Abierta y $y(\tau)$ es Alta Entonces $\hat{y}(\tau + 1) = 0,0209u(\tau) + 0,9990y(\tau) - 0,9149$

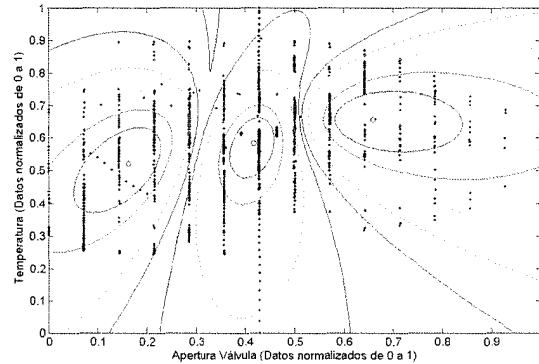


Figura 3.20: Agrupación de datos del proceso de calentamiento usando el algoritmo Gustaffson-Kessel

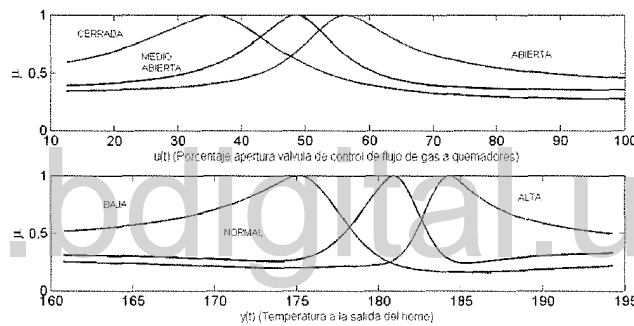


Figura 3.21: Proceso de calentamiento: funciones de pertenencia de $y(\tau)$ y $u(\tau)$ (algoritmo GK)

El modelo difuso TS obtenido para el horno fue validado con 750 datos entrada-salida distintos de los usados para crear el modelo. En la gráfica de la figura 3.22 la curva de color azul representa la salida real del proceso y la curva de rojo es la salida del modelo. La raíz cuadrada del error cuadrático medio es de 0.1201 y el índice de compacidad es 2.732 y los índices de solapamiento de cada modelo local (IS_1 , IS_2 , IS_3) son 0.1234, 0.1034 y 0.1127.

3.5.3.2. Resultados de la aplicación del algoritmo FCM

Tomando en cuenta las mismas premisas anteriores, se aplica a continuación el algoritmo Fuzzy C Means a los datos del proceso obteniéndose en la gráfica 3.23 la nueva agrupación de los datos. La gráfica 3.24 muestra las funciones de

CAPÍTULO 3. IDENTIFICACIÓN DE MODELOS DIFUSOS A PARTIR DE DATOS DE ENTRADA-SALIDA

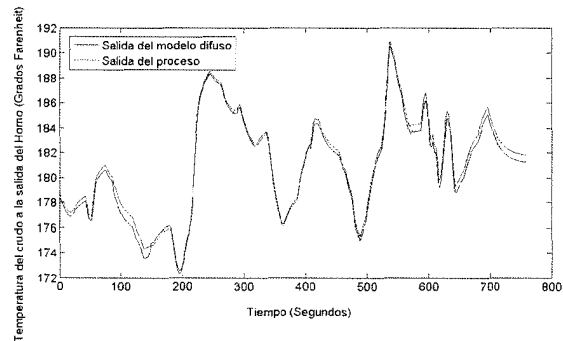


Figura 3.22: Validación del modelo difuso del horno de producción (algoritmo GK).

membresía de $y(\tau)$ y $u(\tau)$ obtenidas y la gráfica 3.25 muestra la validación del modelo difuso usando 740 datos distintos a los usados para la construcción del mismo. La determinación de los parámetros lineales del consecuente se realizó a

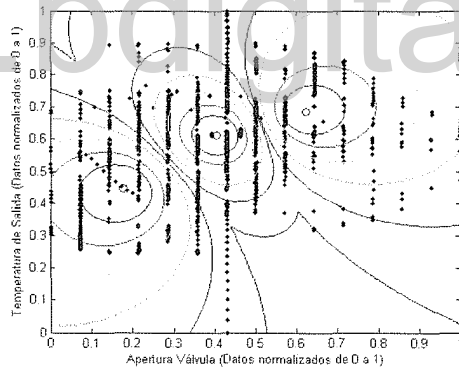


Figura 3.23: Agrupación de datos del proceso de calentamiento usando el algoritmo Fuzzy C Means

través de técnicas de mínimos cuadrados usando la versión recursiva. La base de reglas obtenidas se muestra a continuación:

- Si $u(\tau)$ está Cerrada y $y(\tau)$ es Baja Entonces $\hat{y}(\tau + 1) = 0,0147u(\tau) + 0,8511y(\tau) + 7,9345$
- Si $u(\tau)$ está Medio Abierto y $y(\tau)$ es Normal Entonces $\hat{y}(\tau+1) = -0,0045u(\tau) + 0,81y(\tau) + 8,3456$

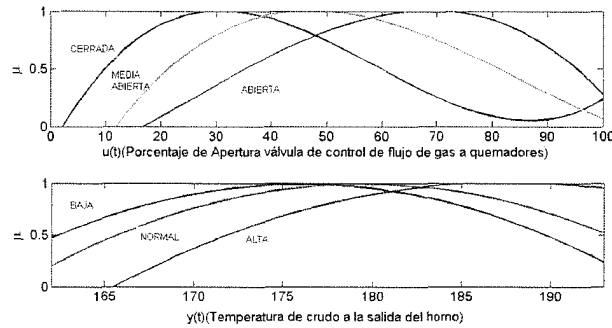


Figura 3.24: Proceso de calentamiento: funciones de pertenencia de $y(\tau)$ y $u(\tau)$ (Algoritmo FCM)

- Si $u(\tau)$ está Abierta y $y(\tau)$ es Alta Entonces $\hat{y}(\tau + 1) = 0,0356u(\tau) + 1,23y(\tau) - 0,99$

El modelo difuso TS obtenido para el horno fue validado con 750 datos entrada-salida distintos de los usados para crear el modelo. En La gráfica de la figura 3.25 la curva de color azul representa la salida real del proceso y la curva de rojo es la salida del modelo. La raíz cuadrada del error cuadrático medio es de 0.1483 y el índice de compacidad es 8.234 y los índices de solapamiento de cada modelo local (IS_1, IS_2, IS_3) son de 0.1684, 0.1912 y 0.2110.

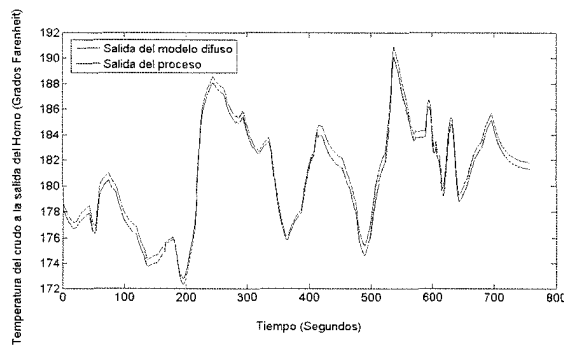


Figura 3.25: Validación del modelo difuso del horno de producción (algoritmo FCM)

3.5.3.3. Resultados de la aplicación del algoritmo ANFIS

Nuevamente sobre el mismo conjunto de datos de la figura 3.19, se determinó un modelo de tipo ANFIS. Para ello se definieron los siguientes parámetros: el número de épocas fue de 3500 y se usó el algoritmo de retropropagación de errores hacia atrás partiendo como premisa previa el uso de funciones de membresía de tipo gaussiana, campana y triangular.

La gráfica de la figura 3.26 muestra el comportamiento de la curva de error obtenida tras aplicar el algoritmo de retropropagación de errores hacia atrás y la raíz cuadrada del error cuadrático medio dio 0.1550. La gráfica 3.27 muestra las seis funciones de membresía obtenidas para las variables $y(\tau)$ y $u(\tau)$ y la base de reglas que arrojó un total de 36 reglas se encuentra en el anexo B. Finalmente la gráfica de la figura 3.28 muestra la validación del modelo difuso en donde se puede apreciar que el modelo tiene tanto una buena generalización como una buena precisión.

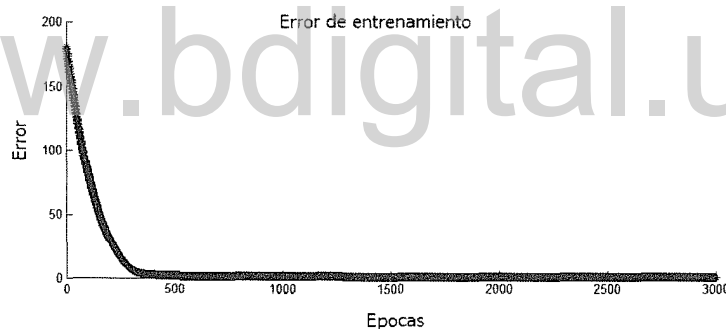


Figura 3.26: Comportamiento de la señal de Error (modelación ANFIS del horno)

Tabla 3.2: Índices de desempeño de los modelos difusos obtenidos para el horno de fuego directo

Nombre de la Técnica	RSME	Indice de Solapamiento	Funciones de Membresia
GK	0.1201	$IS_1 = 0,1234, IS_2 = 0,1034, IS_3 = 0,1127$	No estándar
FCM	0.1483	$IS_1 = 0,1684, IS_2 = 0,1912, IS_3 = 0,2110$	No estándar
ANFIS	0.1001	No aplica	Triangular (estándar)

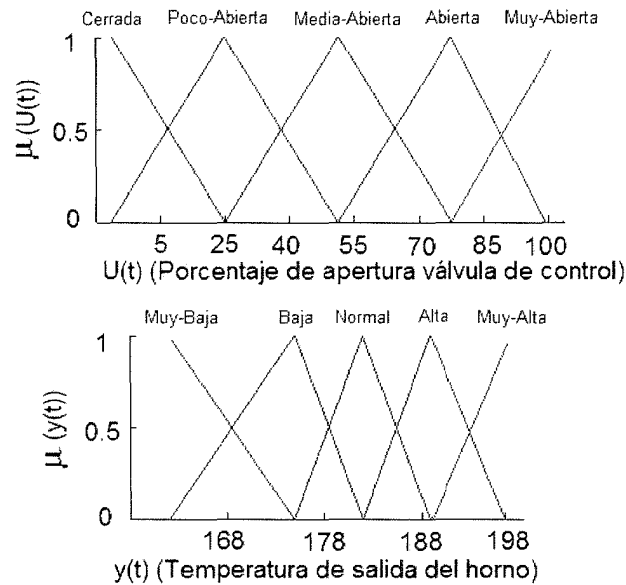


Figura 3.27: Funciones de membresía del antecedente (modelación ANFIS del horno)

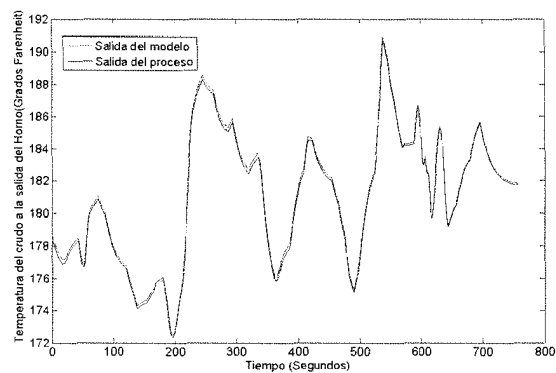


Figura 3.28: Validación del modelo difuso del horno de producción (técnicas neurodifusas)

3.5.4. Análisis de interpretabilidad de los modelos difusos identificados del horno

Análogamente al caso del separador, la interpretabilidad de los modelos difusos obtenidos para el horno se analizarán tomando en cuenta las características de los conjuntos difusos de los términos lingüísticos del antecedente y el número total de reglas .

En cuanto a los modelos difusos identificados usando técnicas de agrupamiento se puede deducir que como aproximadores de funciones son bastante exactos a pesar de contar con tan sólo tres reglas, sin embargo, los conjuntos difusos asociados a los términos lingüísticos del antecedente son no estándares. En forma general, como se puede apreciar en la tabla 3.2, el modelo difuso obtenido con el algoritmo GK presenta mejor desempeño que el modelo obtenido con el algoritmo FCM.

En cuanto al modelo difuso obtenido con el algoritmo ANFIS, se puede apreciar que los conjuntos difusos asociados a los términos lingüísticos del antecedente fueron de tipo triangular, no obstante, para que su exactitud fuera similar a los modelos difusos obtenidos usando las técnicas de agrupamiento, fue necesario que el total de reglas fuera de 36, es decir, cada término lingüístico del antecedente tiene asociado 6 conjuntos difusos.

3.6. Propuesta metodológica para la identificación de modelos difusos de alta interpretabilidad

Existen algunos métodos para la identificación de modelos difusos unitarios propuesto por varios autores. Un método muy interesante propuesto en (Contreras, Misa, and Urueta 2007)(Salas 1998) se denomina identificación difusa por error de inferencia y se fundamenta esencialmente en el uso de métodos de mínimos cuadrados. Otra alternativa interesante consiste en el uso de los algoritmos genéticos como método de optimización para ajustar los parámetros escalares de un modelo difuso tomando como referencia la definición adecuada de una función objetivo(Chang, Lee, and Leung 1997)(Cheng and Wong 1995)(Homaifar and McCormick 2002). Una tercera alternativa se basa en el uso de las técnicas neurodifusas ya estudiadas en la sección anterior.

Esta sección presenta un método que permite identificar modelos difusos de tipo unitario a partir de datos de entrada - salida usando algoritmos genéticos basado en el trabajo realizado en (Ramirez and Colina 2010), que cumpla con

3.6. PROPUESTA METODOLÓGICA PARA LA IDENTIFICACIÓN DE MODELOS DIFUSOS DE ALTA INTERPRETABILIDAD

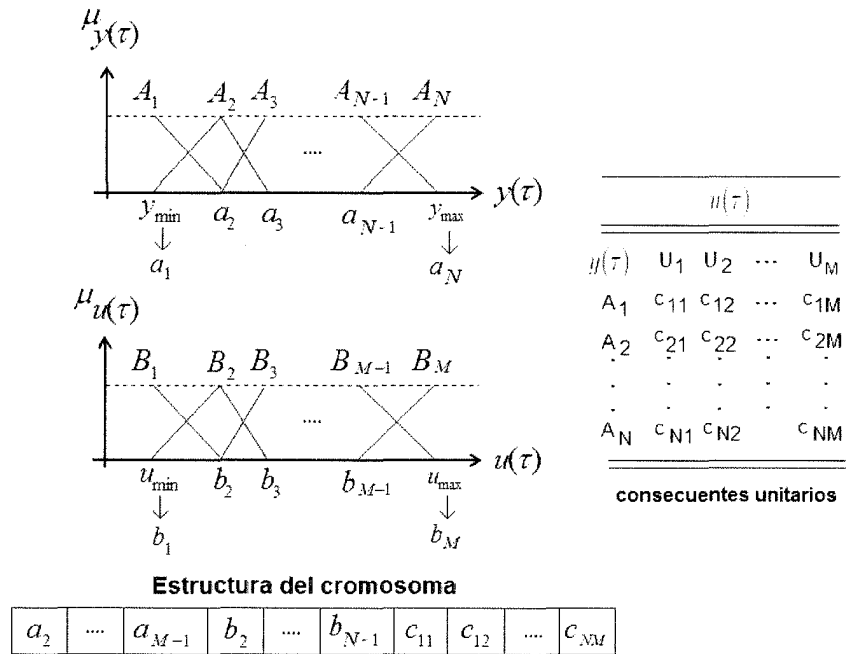


Figura 3.29: Estructura del cromosoma, funciones de membresía y consecuentes unitarios

las restricciones establecidas en (Sousa and Kaymak 2002) de manera de poder garantizar el cálculo analítico de la inversa del modelo a fin de obtener el controlador. Se trata de dos restricciones, la primera tiene que ver con que los conjuntos difusos deben ser de tipo triangular y la segunda está relacionada con la monotonicidad de los consecuentes unitarios. Este tema se explicará en detalle en el próximo capítulo.

El método que se propone a continuación parte del hecho de que los modelos difusos que se desean identificar son de dos entradas y una salida. Las variables de entrada tienen M y N términos lingüísticos representados por los conjuntos difusos triangulares tal y como se muestra en la figura 3.29. Si se denota a $y(\tau)$ y $u(\tau)$ como las entradas del modelo difuso unitario de un cierto proceso entonces los conjuntos difusos A_1, A_2, \dots, A_N pertenecen a la variable $y(\tau)$ y los conjuntos difusos B_1, B_2, \dots, B_M pertenecen a la variable $u(\tau)$.

El objetivo del algoritmo genético es optimizar las MXN salidas unitarias de la base de reglas del modelo difuso y los núcleos de cada conjunto difuso de las variables del antecedente. Sin embargo, como se puede observar en la figura 3.29,

los núcleos de los conjuntos difusos A_1, A_N son y_{min} y y_{max} respectivamente y de manera analoga los núcleos de los conjuntos difusos B_1 y B_M son u_{min} y u_{max} . Por ende, los núcleos que requieren ser optimizados por el algoritmo genético corresponden a los conjuntos difusos A_2, \dots, A_{N-1} y B_2, \dots, B_{M-1} . Denótese por a_2, a_3, \dots, a_{N-1} y b_2, b_3, \dots, b_{M-1} los núcleos de dichos conjuntos difusos de $y(\tau)$ y $u(\tau)$ tal y como se muestra en figura 3.29; entonces el total de variables a optimizar es de $(M * N) + (M - 2) + (N - 2)$.

Con respecto a los algoritmos genéticos, se puede decir, que estos son métodos adaptativos, aplicados en problemas de búsqueda y optimización de parámetros basados en la reproducción sexual y en el principio de la supervivencia del más apto (Davis 1991)(Goldberg 1989).

3.6.1. Determinación del modelo difuso unitario para el separador de producción

Basado en el mismo conjunto de datos que permitió la identificación de modelos difusos usando los métodos estudiados, se propone usar un algoritmo genético para optimizar un modelo difuso unitario que satisfaga las condiciones establecidas en (Sousa and Kaymak 2002) y cuyo índice de desempeño sea menor que 0,15 medido en base a la raíz cuadrada del error cuadrático medio.

El uso de algoritmos genéticos para optimizar un determinado problema requiere de la determinación de seis parámetros: la representación del cromosoma, la función de idoneidad, los operadores genéticos que ejecutan la función de reproducción, la creación de la población inicial, el criterio de terminación y la función de evaluación.

La representación del cromosoma es indispensable para describir a cada individuo de la población. Cada cromosoma se compone de una secuencia de genes de un determinado alfabeto. Un alfabeto puede consistir de dígitos binarios o de números en formato punto flotante o de números enteros, símbolos, etc. En la representación original de Holland, el alfabeto estuvo limitado a la representación de dígitos binarios. Una representación útil de un cromosoma para propósitos de optimización consiste en usar genes de un alfabeto de número flotantes acotadas en un rango determinado. Michalewicz en (Michalewicz 1994) realizó una cantidad significativa de pruebas comparando la representación binaria y la representación basada en números reales y demostró que esta última es mucho más eficiente en términos de ejecución. Por ende, se usará la representación del cromosoma basada en números reales.

3.6. PROPUESTA METODOLÓGICA PARA LA IDENTIFICACIÓN DE MODELOS DIFUSOS DE ALTA INTERPRETABILIDAD

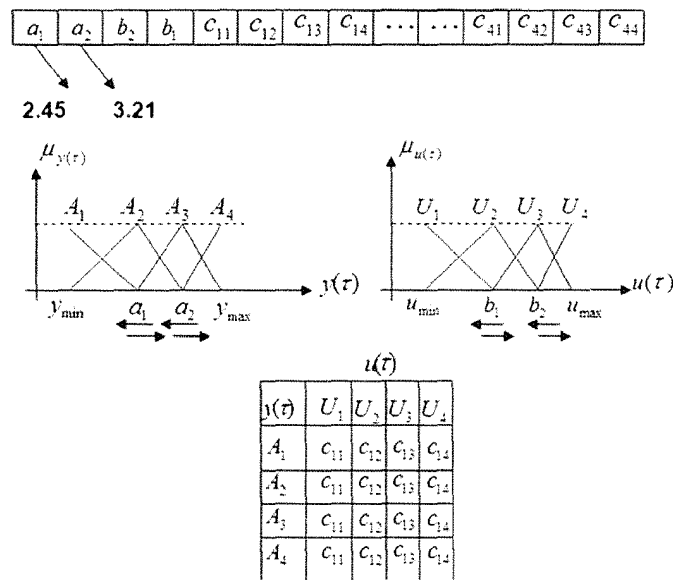


Figura 3.30: Parámetros a optimizar del modelo difuso unitario (separador)

Se hizo una primera prueba seleccionando de antemano tres términos lingüísticos para las variables de entrada $y(\tau)$ y $u(\tau)$ obteniéndose como mejor índice de desempeño el valor de 0,32. Luego se usaron cuatro términos lingüísticos quedando la estructura del cromosoma tal y como lo muestra la figura 3.30.

Dicha estructura consta de los siguientes parámetros: los núcleos a_2 y a_3 correspondientes a los conjuntos difusos A_2 y A_3 , los núcleos b_2 y b_3 correspondientes a los conjuntos difusos B_2 y B_3 y los 16 conjuntos difusos unitarios o salidas reales de la base de regla tal y como se observa en la figura 3.30. Cada parámetro es un número real con dos dígitos decimales de precisión.

Las restricciones relacionadas con la monotonía y los núcleos de las funciones de membresía para este caso específico son las siguientes:

- $a_1 \neq a_2$ y $b_1 \neq b_2$
- $c_{11} < c_{12} < c_{13} < c_{14}$
- $c_{21} < c_{22} < c_{23} < c_{24}$
- $c_{31} < c_{32} < c_{33} < c_{34}$

CAPÍTULO 3. IDENTIFICACIÓN DE MODELOS DIFUSOS A PARTIR DE DATOS DE ENTRADA-SALIDA

$$\blacksquare c_{41} < c_{42} < c_{43} < c_{44}$$

Se sabe que a_1, a_2, b_1 y b_2 están acotados en los rangos (y_{min}, y_{max}) y (u_{min}, u_{max}) respectivamente. La función objetivo que se desea minimizar es la raíz cuadrada del error cuadrático dado por:

$$RMES = \bar{J} = \sqrt{\frac{1}{N} \sum_{k=1}^N (y_k - \hat{y}_k)^2} \quad (3.64)$$

En donde y_k es la salida real y $\hat{y}_k = \hat{y}(\tau + 1)$ es la salida del modelo difuso unitario y N es el número total de muestra. Dado que los operadores genéticos maximizan la función de idoneidad, la función objetivo a minimizar debe ser de alguna manera maximizada. Al mismo tiempo, para garantizar las restricciones mencionadas, se definen las funciones F_m y C_{cm} asociada a la condición de monotonocidad y de los núcleos de las funciones de membresía tal que:

$$J = \frac{F_m}{1 + \bar{J}} * C_{cm} \quad (3.65)$$

En donde $F_m = 1$ si y solo si se cumple la condición de monotonocidad de lo contrario es cero. Análogamente $C_{cm} = 1$ si la condición de los núcleos se cumple de lo contrario es cero. En relación a los operadores genéticos, éstos son usados para crear nuevas soluciones basadas en soluciones existentes de la población inicial. Los operadores básicos son: cruce y mutación. El operador de cruce toma dos individuos y produce dos nuevos individuos mientras que el de mutación altera un individuo y produce una simple solución.

La aplicación de estos dos operadores y sus derivados dependen de la representación del cromosoma seleccionado. Algunos de los operadores que pueden ser usados son: mutación uniforme, mutación no uniforme, mutaciones multi-uniformes, mutación acotada, cruce simple y cruce heurístico. Para este caso particular, se seleccionó cruce simple y mutación uniforme; la tasa de cruce se estableció en 0,95 y la tasa de mutación en 0,05. Para seleccionar los mejores individuos en cada generación, se optó por el método de la ruleta.

En cuanto a la inicialización, terminación y evaluación, se debe proveer al algoritmo genético de una población inicial. El método más común consiste en generar soluciones aleatorias para la población entera. En cuanto al criterio de terminación, el más usado es un número máximo de generaciones o sencillamente el cumplimiento de la función de idoneidad. Durante la corrida del algoritmo genético, para la optimización del modelo difuso del separador, se fijó en 700 el máximo número de generaciones y se obtuvo que $J \geq 0,9524$, lo cual, equivale a

3.6. PROPUESTA METODOLÓGICA PARA LA IDENTIFICACIÓN DE MODELOS DIFUSOS DE ALTA INTERPRETABILIDAD

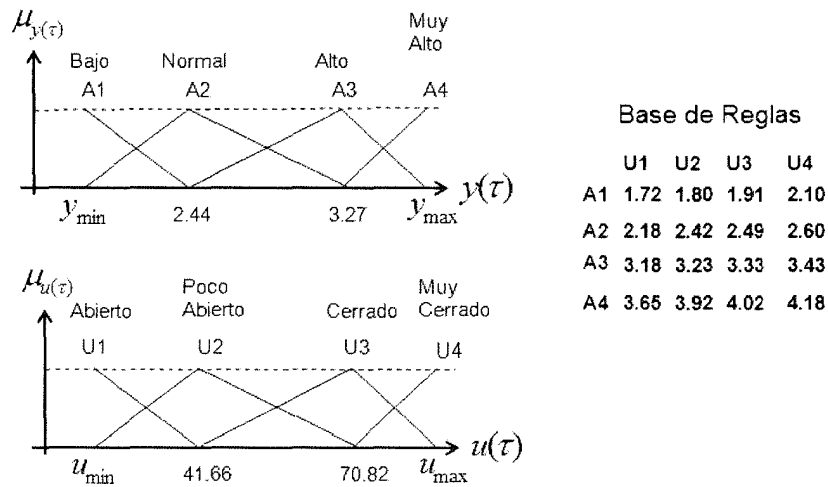


Figura 3.31: Parámetros optimizados del modelo difuso unitario del separador

$$\bar{J} \leq 0,15.$$

El toolbox Algoritmo genético para optimización del software Matlab fue usado para el desarrollo de las simulaciones computacionales. Se estableció la población inicial en 20 y luego de la corrida en la generación 557 se obtuvo que $J = 0.14998$ y el modelo difuso optimizado unitario contiene los parámetros que se muestran en la figura 3.31. La gráfica 3.32 muestra la capacidad de generalización del modelo difuso con 340 datos distintos de los usados para la optimización del modelo difuso.

3.6.2. Determinación del modelo difuso unitario para el horno de producción

Basado en un procedimiento similar al establecido en la sección anterior, se desea optimizar un modelo difuso unitario para el horno de producción en base a los mismos datos de entrada - salida usados para la identificación de los modelos difusos usando técnicas neurodifusas y técnicas de agrupamiento difuso.

Se usaron 5 términos lingüísticos para las variables $y(\tau)$ (temperatura actual) y $u(\tau)$ (Porcentaje de cierre de la válvula de control de flujo de gas) se fijó el índice de desempeño RSME en 0,17 y por consiguiente la estructura del cromosoma estuvo conformada por 6 registros (tres registros para los núcleos de $u(\tau)$ y tres registros para los núcleos de $y(\tau)$) más 25 registros correspondiente a las salidas unitarias de la base de reglas lo que daría un total de 31 parámetros a

CAPÍTULO 3. IDENTIFICACIÓN DE MODELOS DIFUSOS A PARTIR DE DATOS DE ENTRADA-SALIDA

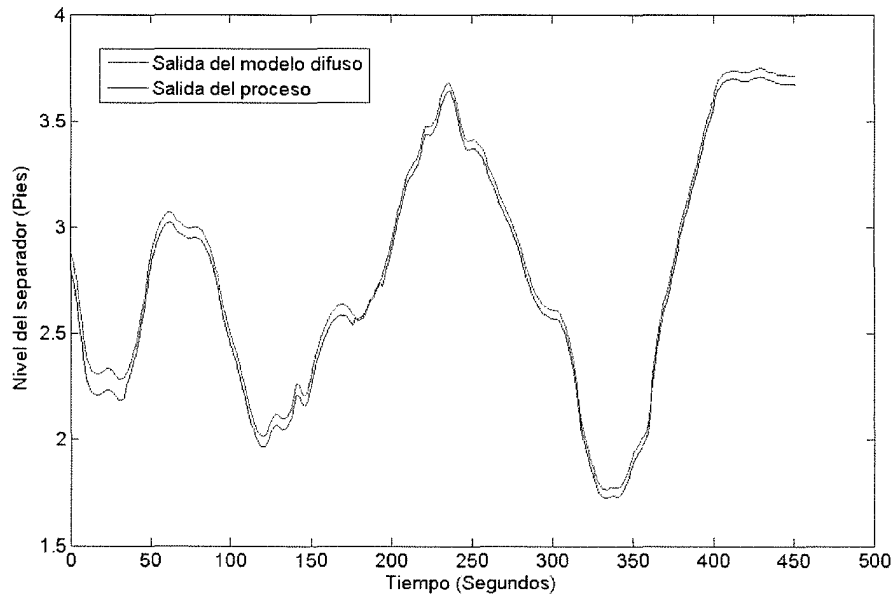


Figura 3.32: Validación del modelo difuso unitario del separador

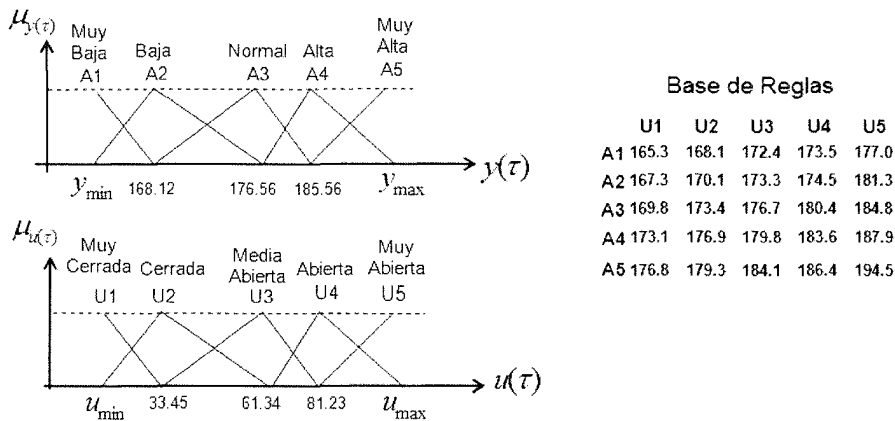


Figura 3.33: Parámetros del modelo difuso unitario del horno

3.7. ALGORITMO ANFIS Y ALGORITMOS GENÉTICOS: ANÁLISIS COMPARATIVO

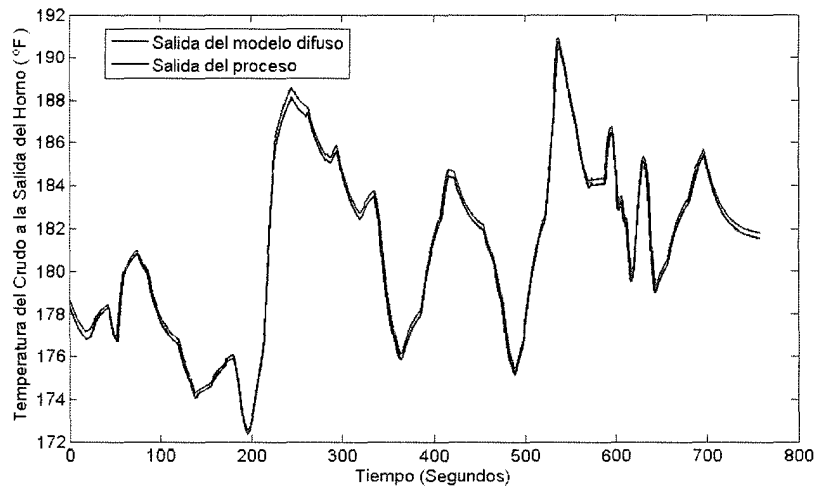


Figura 3.34: Validación del modelo difuso unitario del horno

optimizar.

La población inicial fue fijada en 30 y el máximo número de generaciones fue fijada en 1200. Luego de realizar la corrida del algoritmo genético, en la generación 924 se obtuvo que $J = 0.15$ y los parámetros del modelo difuso unitario del horno se encuentran en la figura 3.33. La figura 3.34 muestra la validación de dicho modelo.

3.7. Algoritmo ANFIS y algoritmos genéticos: análisis comparativo

La aplicación del algoritmo ANFIS y de los algoritmos genéticos en la identificación de los modelos difusos del separador de producción horizontal y del horno de fuego directo presentan una interpretabilidad muy buena desde el punto de vista de las funciones de membresía de los términos lingüísticos de las variables del antecedente. Sin embargo, es importante resaltar que a pesar de que los modelos difusos unitarios y los modelos difusos TS de primer orden tienen prácticamente la misma exactitud medida en base a la raíz cuadrada del error cuadrático medio, el modelo difuso unitario en ambos casos tuvo una base de reglas más pequeña.

Sin embargo, el hecho de que los consecuentes sean de tipo unitario mejora más aun la interpretabilidad y sencillez. Desde el punto de vista de diseño de controladores, resulta más sencillo diseñar controladores difusos unitarios que los

de tipo de TS. Esto se estudiará con más detalle en el capítulo 4.

En forma general, se puede afirmar que la búsqueda de soluciones óptimas de los algoritmos genéticos es mucho más eficiente que los métodos de descenso por gradiente como por ejemplo lo que se implementa en el algoritmo ANFIS.

3.8. Conclusiones

En este capítulo se hizo una completa revisión de tres métodos que permiten identificar modelos difusos de procesos complejos a partir de datos de entrada - salida tomando como base la estructura del modelo de regresión de la forma $\hat{y}(\tau + 1) = f(y(\tau), u(\tau))$. La primera metodología consistió en el uso de algoritmos de agrupamiento difuso, específicamente los algoritmos Gustafson-Kessel (GK) y Fuzzy C Means (FCM) usando datos reales de entrada - salida de los procesos de separación gas - líquido y calentamiento de crudo. En general, dado que las clases detectadas por el algoritmo GK son de tipo elipsoidal y variable, la agrupación de los datos resultó más eficiente que las clases esféricas detectadas por el algoritmo FCM. Se usó como índice de desempeño la raíz cuadrada del error cuadrático medio y el índice de solapamiento; siendo este último de gran utilidad cuando se emplean técnicas de agrupamiento.

El desempeño del algoritmo GK en cuanto a la exactitud y el índice de solapamiento supera al algoritmo FCM en los dos casos de estudios, sin embargo, la presencia de funciones de membresía no estándares y en algunos casos no convexas en ambos métodos, es una desventaja crucial cuando se requiera diseñar controladores. En general, los modelos difusos identificados con esta técnica son de muy buena exactitud a pesar del reducido número de reglas. Sin embargo, su interpretabilidad requiere ser mejorada notablemente.

La segunda metodología empleada consistió en la aplicación de la modelación neurodifusa de tipo ANFIS. Durante la identificación de los modelos difusos de tipo Takagi-Sugeno para los dos casos de estudios, se requirió en términos de simulaciones computacionales, realizar ajustes previos en cuanto al número de iteraciones, el tipo de funciones de membresía del antecedente y la cantidad de funciones de membresía. La exactitud de estos modelos difusos obtenidos fue ligeramente superior que los resultados obtenidos empleando técnicas de agrupamiento y para ello fue necesario entonar adecuadamente varios parámetros hasta lograr que el algoritmo de retropropagación lograra la convergencia. Sin embargo, la interpretabilidad de los modelos difusos ANFIS es notablemente mejor que la de los

modelos difusos obtenidos usando técnicas de agrupamiento, obteniéndose una base de reglas completa.

Finalmente, se empleó una metodología que consiste en identificar modelos difusos unitarios (singleton) a través del uso de algoritmos genéticos tomando en cuenta las restricciones necesarias que garanticen el cálculo analítico de su inversa. En este sentido, los modelos difusos identificados relacionado a los dos casos de estudios mencionados, presentaron una base de reglas más pequeña pero con aproximadamente la misma exactitud que los modelos difusos obtenidos a través de la técnica ANFIS; demostrando de esta forma la ventaja inherente de los algoritmos genéticos en la búsqueda del óptimo global comparado con el método del descenso por el gradiente usado en ANFIS. Desde luego, fue necesario seleccionar de manera adecuada la tasa de cruce, tasa de mutación, la función objetivo y el algoritmo de selección de tal manera de garantizar que el algoritmo genético converja.

Esta última estrategia de identificación de modelos difusos unitarios invertibles y de alta interpretabilidad representa el primer aporte de esta tesis y en el próximo capítulo se describirá el cálculo exacto de la inversa.

www.bdigital.ula.ve

Capítulo 4

CONTROL BASADO EN MODELOS DIFUSOS

4.1. Introducción

Los controladores difusos clásicos han sido diseñados a partir de la extracción del conocimiento de expertos directamente en forma de reglas difusas. Las reglas difusas generadas de esta forma suelen ser imprecisas e incompletas y hay situaciones en las que este conocimiento no necesariamente está disponible.

Otro enfoque para el diseño de controladores difusos consiste en disponer de un modelo difuso, por lo general de tipo Takagi - Sugeno de primer orden o unitario, obtenido a partir de un conjunto de datos de entrada - salida y aplicar algunas de las técnicas de diseño de controladores basados en la inversa del modelo como por ejemplo el control por modelo inverso y el control por modelo interno (Abonji 2007)(Abonyi, Andersen, Nagy, and Szeifert 1999). Es importante para ello garantizar previamente que el conjunto de datos de entrada-salida usado durante el proceso de identificación del modelo difuso permita en lo posible cubrir todos los estados de operación de la planta cuando el controlador se encuentre en funcionamiento (Tanaka and Sugeno 1992).

Vale la pena destacar que existen alternativas más complejas para el diseño de controladores de difusos que consisten en adaptar sus parámetros en línea según el comportamiento real de la planta sin la necesidad de intervención externa. Estos controladores son de tipo adaptativo y / o auto-organizados y en esencia son capaces de modificar automáticamente su base de reglas como resultado de evaluar la desviación de la salida del proceso con respecto a unos determinados índices de desempeño (Abonji 2007).

A lo largo de este capítulo se describen los principales esquemas estándares contemplados en el diseño de controladores basados en modelos difusos. Específicamente se revisarán los esquemas de control por modelo inverso, control adaptativo por modelo de referencia, control predictivo basado en modelos y control por modelo interno (CMI). En particular, se hará énfasis en el esquema CMI puesto que representa una alternativa distinta a la estructura de retroalimentación clásica y garantiza la estabilidad del sistema en lazo cerrado. Adicionalmente, se presentarán los aspectos concernientes a la inversión analítica de modelos difusos unitarios (Singleton) y Takagi-Sugeno y todas las restricciones inherentes a este tema.

En aras de poder implementar el esquema de control por modelo interno, es necesario disponer de los modelos matemáticos para los procesos de separación gas-líquido y el proceso de calentamiento de crudo los cuales servirán de marco de referencia para la evaluación de dicho esquema. En este sentido, se hará énfasis en la determinación de modelos matemáticos apropiados de los procesos mencionados. Finalmente, el capítulo cierra con las conclusiones.

4.2. Estrategias de control basados en modelos difusos

Un gran número de procesos industriales están caracterizados por un comportamiento no lineal y por la existencia de importantes perturbaciones no medidas que modifican las condiciones de operación. Para estos procesos se han desarrollado diversas técnicas de control adaptativo, control robusto, control inteligente, etc. Mediante la simulación de estas técnicas, se ha comprobado que muestran un buen comportamiento, sin embargo, solamente las soluciones con controladores PID, con controles expertos y con controles predictivo lineal han tenido una mayor repercusión en el campo de las realizaciones prácticas a gran escala.

Hoy en día, gracias al desarrollo computacional que han sufrido los dispositivos digitales, el control inteligente se ha convertido en una alternativa viable a las estructuras clásicas de control. La razón por la cual resultan más adecuada este tipo de técnicas se debe a su capacidad de tratar con la incertidumbre y / o con las desconocidas variaciones (perturbaciones) en los parámetros de las plantas y la estructura. En particular, los sistemas difusos desempeñan un papel fundamental dentro de los sistemas de control inteligente debido a que son capaces de modelar cualquier proceso complejo, tal y como se ha visto en el capítulo anterior. Una vez que se ha obtenido un modelo difuso de un proceso, este se puede usar para analizar el propio proceso, para mejorar el entendimiento sobre el mismo, para

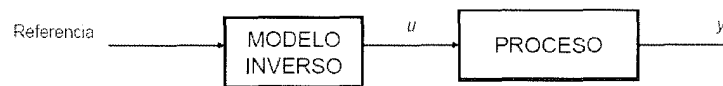


Figura 4.1: Esquema de control por modelo inverso

mejorar la operación, como herramienta de diagnóstico o bien se puede incorporar el modelo difuso obtenido a un sistema de control en tiempo real. Este es el principio de los sistemas de control basado en modelos difusos.

4.2.1. Control por modelo inverso

Una manera sencilla de diseñar un controlador para un proceso no lineal es invertir el modelo del proceso, cuando esto es posible. A esta técnica se le conoce con el nombre de control por modelo inverso o control basado en modelo inverso tal y como lo muestra la figura 4.1. Es el método más simple para diseñar un controlador basado en la adaptación de parámetros a partir de datos de entrada - salida, a pesar de ello, es muy utilizado para crear todo tipo de estructuras de control.

El empleo de modelos difusos en este tipo de control se debe fundamentalmente a dos razones. Por una parte, los modelos difusos son sistemas capaces de representar las no linealidades de los procesos, obteniéndose de este modo controladores no lineales (basados en modelos difusos)(Abonji 2007). En presencia de fuertes no linealidades, los controladores no lineales pueden ofrecer ventajas significativas sobre las técnicas de control lineal, y en muchas ocasiones pueden ser la única solución para desarrollar un control aceptable. La otra razón es que a través de los modelos difusos se puede modelar la dinámica inversa de los procesos (a través de datos entrada-salida) sin necesidad de realizar una inversión del modelo (tarea difícil y no siempre posible).

El control por modelo inverso puede ser aplicado a sistemas que son estables a lazo abierto (o que son estabilizados por realimentación) y cuyo modelo inverso también es estable, es decir, si el sistema no exhibe comportamientos de fase no mínima (Abonji 2007). En la teoría, el uso del control por modelo inverso posee las ventajas del control a lazo abierto: estabilidad inherente y control sin errores en las respuestas transitoria y permanente. Además, el uso de modelos difusos incorpora las saturaciones inherentes a las entradas del modelo y a otras variables del proceso, proporcionando un sistema de control capaz de controlar un sistema no lineal en todo el rango de operación. En la práctica, sin embargo, los sistemas

de control tienen que tratar con errores de modelado y con perturbaciones (no contempladas por este esquema de control) y, por tanto, se hace necesario incorporar nuevos elementos al esquema de control basado en modelo inverso para tratar con sistemas complejos (e.g., control por modelo interno).

4.2.2. Control adaptativo por modelo de referencia

Los sistemas adaptativos por modelo de referencia fueron diseñados primeramente para sistemas continuos con la finalidad de minimizar un índice de actuación, siendo dicho índice la integral del error al cuadrado (Hang and Parks 1973). Esta regla de diseño fue la propuesta inicialmente por Whitaker en el año 1958 (Whitaker, Yamron, and Kezer 1958), sin embargo, los sistemas fueron evolucionando creándose nuevos índices de referencia. En cuanto a las configuraciones posibles como modelo de referencia, la más usual es utilizar un modelo paralelo (Figura 4.2), aunque son posibles otras configuraciones como modelo serie, serie-paralelo, etc.

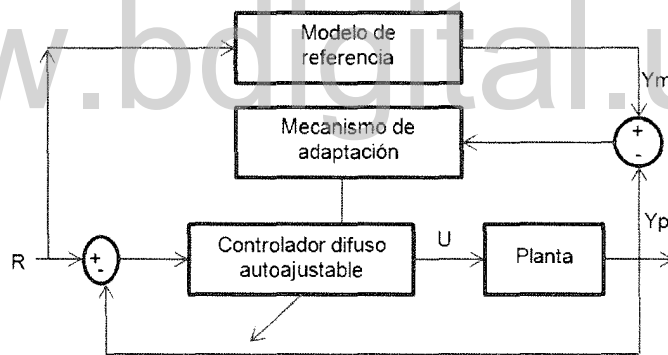


Figura 4.2: Esquema de control adaptativo por modelo de referencia

Existe una dualidad entre los sistemas de control adaptativo por modelo de referencia y el problema de identificación con un modelo ajustable, siendo en este caso el modelo de referencia la planta a identificar. Dado un modelo de referencia y un sistema ajustable, la minimización del error ha sido la regla más popular debido a su simplicidad. Sin embargo, para el caso de ajuste de varios parámetros, se requiere un número elevado de funciones de sensibilidad (tantas como parámetros).

La figura 4.2 muestra la configuración más común del control difuso adaptativo por modelo de referencia (CDMR) más usado (Pasino and Yurkovich 1998). El

4.2. ESTRATEGIAS DE CONTROL BASADOS EN MODELOS DIFUSOS

modelo de referencia cuantifica el desempeño deseado de la planta y está sujeto a criterio del diseñador. En general, el modelo de referencia puede ser de tiempo discreto o continuo, lineal o no lineal, invariante en el tiempo o variable en el tiempo, y así sucesivamente.

El mecanismo de adaptación es el elemento central del CDMR ya que se encarga de sintonizar la base de reglas original del controlador difuso con el propósito de que el sistema de lazo cerrado se comporte como el modelo de referencia. La modificación de las reglas bases se hace por la observación y comparación de los datos obtenidos en las salidas de la planta y la que se esperaría del modelo de referencia.

4.2.3. Control por modelo interno (CMI)

En esta sección, se revisará el control basado en modelo difuso usando la estrategia de Control por Modelo Interno (CMI). La estructura CMI consiste de un controlador C , un modelo del proceso M , un filtro F , y el proceso como tal P tal y como lo muestra la figura 4.3.

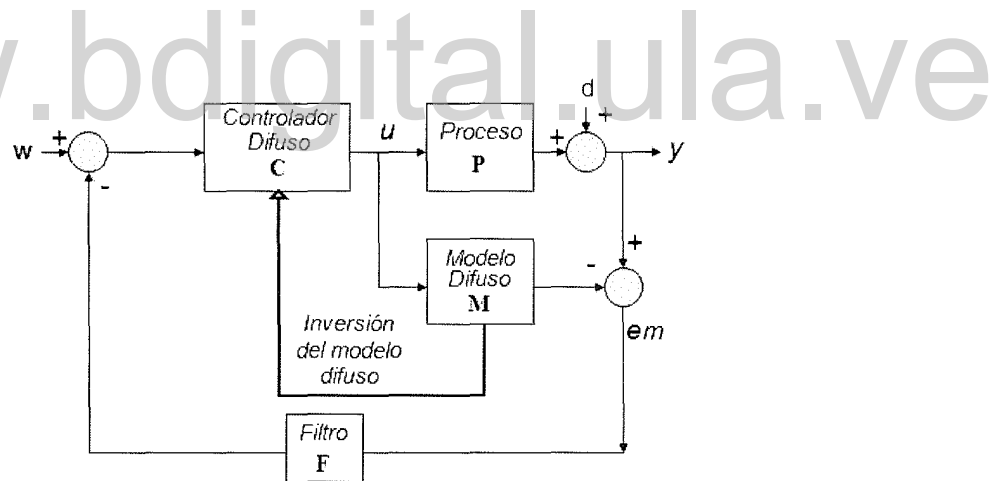


Figura 4.3: Esquema de control por modelo interno

En este esquema de control, el modelo del proceso M es colocado en paralelo con el sistema real. La diferencia entre la señal de salida del sistema y el modelo representa el error de modelación o perturbaciones del proceso no modeladas. Esta diferencia pasa por un filtro y es usada por el controlador para compensar los efectos del error de modelación y / o de las perturbaciones.

CAPÍTULO 4. CONTROL BASADO EN MODELOS DIFUSOS

Las características más importantes de este esquema de control pueden ser resumidas a continuación:

- Estabilidad: asumase que la planta P y el controlador C son de entrada - salida estable y que el modelo M es una representación perfecta de la planta $M = P$; entonces el sistema en lazo cerrado es de entrada - salida estable.
- Control Perfecto: si la inversa exacta del operador que describe a la planta M^{-1} existe, y esta inversa es usada como controlador $C = M^{-1}$, el sistema en lazo cerrado es de entrada-salida estable con este controlador, entonces el control será perfecto $y = w$, en donde w representa la salida deseada (setpoint) del controlador.
- Error de estado estático nulo: si la inversa exacta del operador que describe el modelo de la planta M existe, y esta inversa es usada como controlador $C = M^{-1}$, y el sistema en lazo cerrado es de entrada-salida estable con este controlador, entonces para entradas asintóticamente constantes, se garantiza un control con error de estado estático nulo.

La estructura CMI provee un método directo para el diseño de controladores retroalimentados. De acuerdo a las propiedades mencionadas si un buen modelo de la planta está disponible, el sistema en lazo cerrado exhibirá una salida con mínimo o nulo error de estado estático a pesar de la presencia de perturbaciones no medibles que actúan sobre la planta.

Discusiones posteriores han considerado el caso idealizado de un modelo perfecto, que conlleva a un control perfecto. En la práctica, un modelo perfecto de la planta nunca puede ser obtenido. Adicionalmente, la ganancia infinita requerida para el control perfecto crearía problemas de sensibilidad bajo la incertidumbre del modelo. El filtro F es introducido para aliviar estos problemas. Si se hace un diseño idóneo, el filtro puede ser seleccionado para reducir la ganancia del sistema retroalimentado e introducir robustez a la estructura CMI. Un tratamiento completo sobre la robustez y el diseño del filtro se encuentra en (Prett and Morari 1988). Los detalles de la estrategia de control CMI para sistemas lineales pueden ser conseguidos en la publicación de Garcia y Morari (Garcia and Morari 1982). Basado en esta publicación, para procesos estables en lazo abierto se puede conseguir un controlador perfecto, el cual es esencialmente la inversa de la fase mínima del sistema. Adicionalmente, se ha demostrado que bajo ciertas premisas, el sistema completo es estable y tiene un error de estado estático nulo. Este enfoque fue aplicado en sistemas no lineales por Economou (Economou, Morari, and Palsson 1986). Basado en este trabajo, estrategias de control directo e inverso fueron propuestos por Psychogios y Ungar (Psychogios and Ungar 1991).

4.2. ESTRATEGIAS DE CONTROL BASADOS EN MODELOS DIFUSOS

En el enfoque de control directo, la acción de control es explícitamente calculada como la salida del controlador (red neuronal o modelo difuso) que es entrenado para representar la dinámica del proceso inverso. El error de aproximación de la dinámica inversa afecta significativamente el desempeño del controlador. De aquí que, una alternativa distinta consiste en calcular las acciones de control indirectamente, basado en la inversión en línea del modelo que describe la dinámica del proceso. Esto trae algunos problemas, dado que es difícil manipular matemáticamente modelos de proceso bajo la forma de redes neuronales o modelos difusos relacionales. Por lo tanto, estos enfoques indirectos se basan en métodos de optimización numérica (Economou, Morari, and Palsson 1986)(Psychogios and Ungar 1991).

Desde el punto de vista de la aplicación, como en los esquemas de control convencional adaptativo (Astrom and Wittermark 1989), existen dos arquitecturas que pueden ser formuladas en el área del control difuso y neuronal: el enfoque adaptativo directo e indirecto. El esquema de control adaptativo directo adapta un controlador en base a alguna medida de desempeño relevante, mientras que el esquema indirecto identifica un modelo de la planta (Johnson 1988)(Ljung 1983) y usa esto como un cálculo de inversión predefinido para adaptar el controlador. La técnica de mapeo de entrada (Input matching technique) es una alternativa intermedia porque busca ajustar directamente los parámetros del controlador de acuerdo a la identificación en línea de la inversa del modelo del proceso, la cual es posteriormente usada como controlador (Abonyi, Andersen, Nagy, and Szeifert 1999)(Andersen, Lotfi, and Tsoi 1997)(Johnson and Tse 1978) .

De ahí que la adaptación en línea de modelos difusos puede ser usada para contrarrestar la salida del proceso con la salida del modelo difuso. En muchos casos, tal diferencia ocurre como consecuencia de un cambio temporal de los parámetros del proceso y de las variables ambientales. Para tratar este fenómeno, el modelo difuso puede ser adaptado en el lazo de control resultando en un controlador difuso adaptativo. En este sentido, el primer controlador difuso adaptativo fue propuesto por Procyk y Mamdani (Procyk and Mamdani 1979). Este algoritmo auto-organizado fue la base de los algoritmos de Arva y Nemet (Arva and Nemet 1991)(Renders, Saerens, and Hugues 1997). En la figura 4.4 se muestra el controlador CMI adaptativo resultante.

4.2.4. Control predictivo basado en modelo (MPC)

El control predictivo basado en modelo se refiere a una clase de algoritmos de control en la cual un determinado modelo de un proceso dinámico es usado para predecir y optimizar el desempeño del mismo. En esencia, el MPC se enmarca

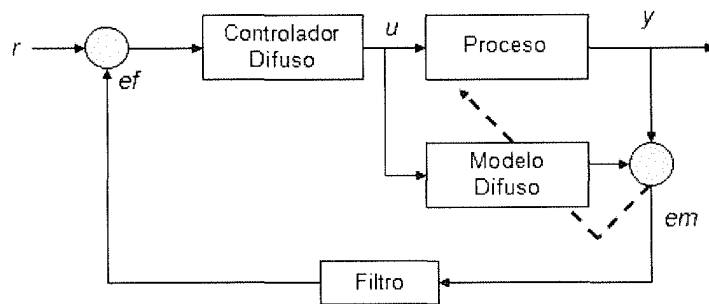


Figura 4.4: Esquema de control por modelo interno adaptativo

dentro de los controladores óptimos, es decir, aquellos en los que las actuaciones responden a la optimización de un criterio. El criterio a optimizar, o función de costo, está relacionado con el comportamiento futuro del sistema, que se predice gracias a un modelo dinámico del mismo. El intervalo de tiempo futuro que se considera en la optimización se denomina horizonte de predicción.

Dado que el comportamiento futuro del sistema depende de las actuaciones que se aplican a lo largo del horizonte de predicción, son estas las variables de decisión respecto a las que se optimiza el criterio. La aplicación de estas actuaciones sobre el sistema, conducen a un control en lazo abierto. La posible discrepancia entre el comportamiento predicho y el comportamiento real del sistema crean la necesidad de imponer cierta robustez al sistema, incorporando realimentación al mismo. Esta realimentación se consigue gracias a la técnica del horizonte deslizante, que consiste en aplicar las actuaciones obtenidas durante un periodo de tiempo, tras el cual se hace un muestreo al estado del sistema y se resuelve un nuevo problema de optimización. De esta manera, el horizonte de predicción se va deslizando a lo largo del tiempo (Camacho and Bordons 1995).

Una de las propiedades más atractivas del MPC es su formulación abierta, que permite la incorporación de distintos tipos de modelos de predicción, sean lineales o no lineales, monovariantes o multivariantes, a la vez que permiten incluir las restricciones sobre las señales del sistema. Esto hace que sea una estrategia utilizada en diversas áreas del control de procesos. Además, es una de las pocas técnicas que permiten controlar sistemas con restricciones, incorporándolas al propio diseño del controlador, aunque existen otras técnicas de control predictivo, como la basada en redes neuronales, que han demostrado su utilidad en el modelado de sistemas no lineales o el controlador predictivo por modelos no lineales (NMPC) que ayudan a solucionar el problema de la respuesta dinámica de los controladores lineales que resulta ser inaceptable cuando se aplica a procesos no lineales con distinto grado

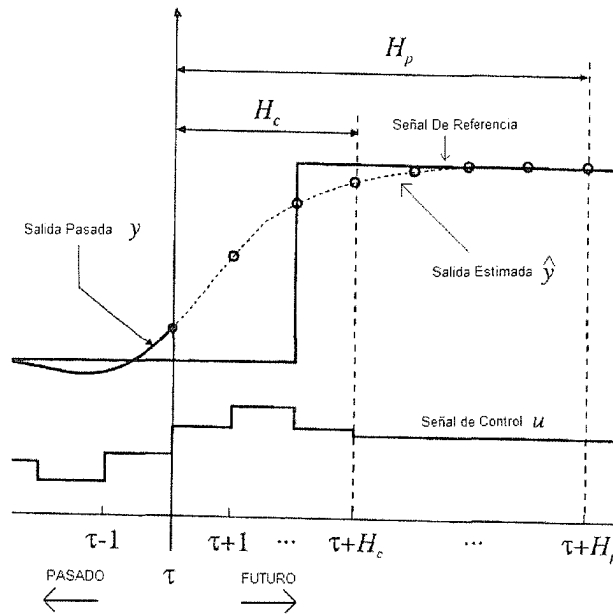


Figura 4.5: Estrategia de los controladores predictivos basado en modelos

de severidad (cuando la planta no opera en las cercanías del punto de operación) (García, Prett, and Morari 1989).

La metodología de controladores perteneciente a la familia MPC presenta las siguientes características:

- Como lo muestra la figura 4.5, las salidas futuras para un determinado horizonte de predicción H_p se predicen en cada instante τ usando un modelo de predicción. Estas futuras salidas $\hat{y}(\tau + j)$, $j = 1, \dots, H_p$ dependen del estado del modelo en el tiempo actual τ (dado, por ejemplo, por las entradas y salidas anteriores) y sobre las futuras señales de control $u(\tau + j)$.
- Las señales de control sólo cambian dentro del horizonte de control, H_c , permaneciendo constantes después, $u(\tau + j) = u(\tau + H_c - 1)$, $j = H_c, \dots, H_p$.
- El conjunto de señales de control es calculada por medio de la optimización de una función de costo para mantener la señal de salida del proceso tan cerca como sea posible de la trayectoria de referencia $r(\tau + j)$, $j = H_c, \dots, H_p$. Este criterio usualmente requiere de una función cuadrática de los errores entre la señal de salida futura y la trayectoria de referencia. El esfuerzo de control es incluido en la función objetivo en la mayoría de los casos. Una

solución explícita puede ser obtenida si el criterio es cuadrático, el modelo es lineal y no hay restricciones. De lo contrario se requeriría de un método de optimización iterativo.

La diferencia entre los diversos métodos radica en la forma en que el problema de control es traducido dentro de una formulación matemática de tal forma de que se pueda resolver en un cantidad de tiempo determinada. No obstante, en todos los métodos, hay seis factores importantes para el diseño de controladores:

- El modelo: el término control predictivo hace claro uso de lo que se supone debería hacer un modelo que en este caso es usado para predecir el comportamiento futuro del sistema. El modelo del proceso juega un rol decisivo en el diseño del controlador. Las diferentes estrategias de MPC pueden usar varios modelos para representar la relación entre la salida y las entradas medibles, algunas de las cuales son variables manipuladas y otras son consideradas como perturbaciones medibles que pueden ser compensadas por una acción de retroalimentación. Para tratar los efectos desconocidos, se puede usar un observador que actualice los estados del modelo antes de proceder con el cálculo del control predictivo. Por otra parte, el modelo del proceso seleccionado debe ser capaz de capturar las dinámicas del proceso para predecir las salidas futuras y debe ser en lo posible simple de implementar. Por lo tanto, este modelo se selecciona por lo general de tipo lineal, sin embargo, muchos de estos procesos son altamente no lineales para excluir la aplicación exitosa del control predictivo basado en modelo lineal (LMPC). Estos procesos no lineales operan cerca de un punto de operación (Columnas de destilación) y existen procesos moderadamente no lineales con varios regímenes de operación. Esto ha permitido el desarrollo del control predictivo basado en modelos no lineales en la cual un modelo no lineal más exacto es usado para predicción (Henson 1998).
- Rechazo a perturbación y estimación de estado: Ante la presencia de perturbaciones no medibles y errores de modelación, el controlador MCP pudiera exhibir un error de estado estático nulo. Esta situación se maneja a través del diseño de un estimador de perturbaciones lo cual brinda al controlador una acción de control integral. El método más simple para incorporar acción integral es cambiar las referencias con las perturbaciones estimadas como lo muestra la figura 4.6, en donde la referencias corregidas $w'(\tau) = w(\tau) - \hat{d}(\tau)$ son modificadas basadas en la diferencia entre la salida del sistema y los valores estimados $\hat{d}(\tau) = y(\tau) - \hat{y}(\tau)$.

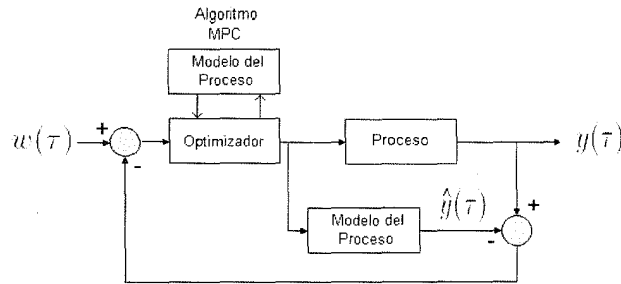


Figura 4.6: Controlador MPC con observador de estado

- Indice de desempeño: se selecciona para satisfacer una amplia variedad de objetivos, incluyendo la minimización de la función de costo del proceso. Sin embargo, la optimización económica puede ser desarrollada por un sistema de alto nivel el cual determina las referencias apropiadas para el controlador. En este caso, el índice de desempeño o la función de costo es formulada reflejando el error de seguimiento de la señal de referencia y la acción de control. La expresión general de dicha función objetivo es:

$$J(H_{p1}, H_{p2}, H_c) = \sum_{j=H_{p1}}^{H_{p2}} \delta(j) [w(\tau+j) - \hat{y}(\tau+j)]^2 + \sum_{j=1}^{H_c} \lambda(j) [\Delta u(\tau+j-1)]^2 \quad (4.1)$$

En donde $\Delta u(\tau)$ denota el cambio en la señal de control, los parámetros H_{p1} y H_{p2} son los horizontes de costos mínimo y máximo y H_c es el horizonte de control, el cual no necesariamente tiene que coincidir con el horizonte máximo. Los coeficientes $\delta(j)$ y $\lambda(j)$ son secuencias que consideran comportamientos futuros, usualmente se usan valores constantes o secuencias exponenciales.

- Restricciones: en la práctica todos los procesos está sujetos a restricciones. Los actuadores tienen un limitado campo de acción así como una determinada velocidad de operación. Razones de seguridad o ambientales pueden causar límites en las variables del proceso tales como niveles en los tanques, flujos en las tuberías, etc. Usualmente, restricciones de la variable de entrada tales como: $u_{min} \leq u(\tau+j) \leq u_{max}, j = 1, \dots, H_c$ o $\Delta u_{min} \leq \Delta u(\tau+j) \leq \Delta u_{max}, j = 1, \dots, H_c - 1$ son restricciones duras en el sentido de que estas deben cumplirse completamente. Por otra parte, las restricciones en la salida pueden ser vistas como restricciones suaves ya que su parcial incumplimiento pudiera ser necesario a fin de cumplir de manera aceptable con el problema de optimización.

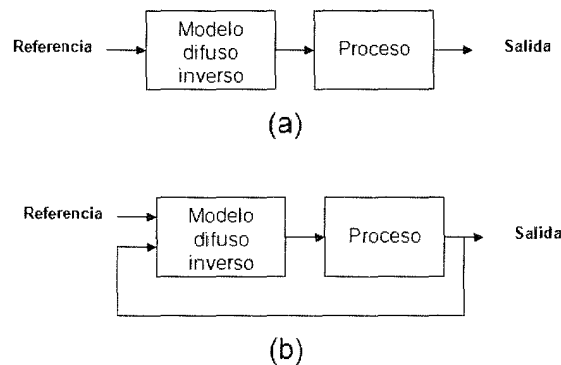


Figura 4.7: Esquema de control lazo abierto y lazo cerrado

- Optimización: es necesario aplicar un algoritmo de optimización para calcular la secuencia de las futuras señales de control que minimizan el índice de desempeño sujeto a las restricciones dadas. Para los sistemas de control basado en procesos lineales sin restricciones y una función de costo cuadrática, la secuencia de control puede ser calculada de manera analítica. Con la inclusión de modelos no lineales dentro de los esquemas MPC, se requiere resolver un problema de optimización no lineal en cada tiempo de muestreo que generalmente es no convexo dado la naturaleza no lineal del proceso (Henson 1998).
- Principio del horizonte deslizante: el control predictivo usa el principio del horizonte deslizante lo que permite calcular la secuencia de control óptima y una vez ejecutada actualizarla nuevamente y cumplir con el ciclo.

4.3. Inversión de modelos difusos

La forma más sencilla de controlar un proceso con un modelo inverso es a través de la configuración de lazo abierto tal y como se muestra en la figura 4.7a. Este esquema de control en particular no tiene retroalimentación de la salida del proceso, así que el control estable es garantizado para sistemas de fase no mínima (Abonji 2007). Dado que un determinado modelo de una planta no es completamente perfecto, se genera un error de estado estático que puede ser compensado por el sistema retroalimentado que se observa en la figura 4.7b.

Existe dos formas de obtener la inversa de un modelo difuso de un proceso: (i) identificación del modelo difuso en base a datos de entrada y salida (ii) inversión como tal del modelo difuso original (inversión analítica).

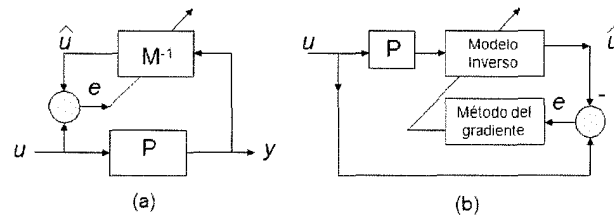


Figura 4.8: Aprendizaje inverso directo

El primer método es tal vez el más intuitivo dentro del área de la modelación inversa y consiste en ajustar los datos de una función inversa f^{-1} (Batur and Celad 1993). Existen dos enfoques bien diferenciados al respecto: aprendizaje inverso directo y aprendizaje inverso especializado (Posthlehwaite 1994) (Posthlehwaite 1997a) (Posthlehwaite 1997b). En el aprendizaje inverso directo, el proceso es excitado con una señal de entrenamiento y el sistema difuso reconstruye la señal de entrada del proceso a partir de la señal de salida como se observa en la figura 4.8a.

Diferentes tipos de algoritmos de identificación pueden ser usados para obtener el modelo inverso bajo esta modalidad. Un esquema bien conocido es el que se muestra en la figura 4.8b, el cual usa el método del gradiente para ajustar, por ejemplo, los parámetros escalares de las funciones de membresía del modelo de tal forma que el error e sea menor o igual a un ϵ previamente definido. En este sentido existen dos limitaciones bien importantes. La dinámica del sistema puede ser de mucha a una, esto quiere decir que en términos numéricos, para varios valores de u se puede obtener el mismo valor para y . Si por ejemplo, se usa algoritmos de mínimos cuadrados, se logra mapear a y con el promedio de todos los u lo que deterioraría la exactitud del modelo inverso. En segundo lugar, es difícil obtener una señal de entrenamiento apropiada para el aprendizaje directo inverso, porque el modelo inverso se supone que trabaja sobre un rango amplio de amplitudes de y y un ancho de banda grande. Sin embargo, la excitación del sistema es introducida como la activación de u y la variación de y es un rango aceptable no se puede garantizar.

Ambos inconvenientes del aprendizaje inverso directo pueden ser superadas por el aprendizaje inverso especializado, el cual se muestra en la figura 4.9 (Jordan and Rumelhart 1998). El modelo inverso se encuentra en cascada con el proceso. Los parámetros del modelo inverso M^{-1} son adaptados para minimizar la desviación entre la señal de referencia r y la salida y . Así que la adaptación es básicamente un esquema orientado a metas, es decir la meta es la misma del controlador, y

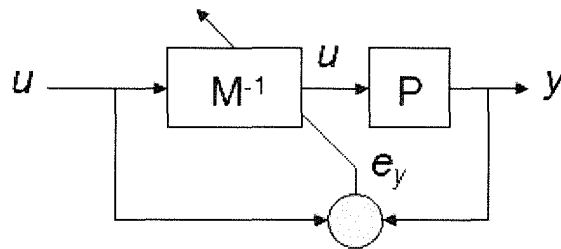


Figura 4.9: Aprendizaje inverso especializado

el proceso es automáticamente excitado con la señal correcta si la trayectoria de referencia típica debe ser seguida. Más aun, el nivel y la tasa de restricción pueden también ser considerados en la fase de aprendizaje.

Aunque el aprendizaje inverso especializado supera los problemas de excitación y la posible no invertibilidad, es bastante difícil de usar el modelo inverso en un esquema de control, debido a la imperfección del modelo y la influencia de perturbaciones. Un esquema de un observador de perturbaciones desarrollado por (Fisher, Fink, and Nelles 1999) puede ser implementado, pero requiere del ajuste de ciertos parámetros, y usa la linealización del modelo inverso en un cierto punto. Por lo tanto, una inversión exacta del modelo inverso difuso no se puede obtener.

La inversión exacta como tal del modelo difuso original usando algún método analítico es la opción más idónea. Si esta inversión es posible, las operaciones computacionales son hechas usando matrices estandares e interpolaciones lineales permitiendo la factibilidad de implementar aplicaciones de control en tiempo real usando por ejemplo el esquema de control no lineal por modelo interno.

4.3.1. Inversión de modelos difusos unitarios

Se partirá del hecho de que se calculará la inversa de un modelo difuso unitario con dos entradas y una salida expresado de la siguiente forma general:

$$R_k : \text{Si } x(\tau) \text{ es } A \text{ y } u(\tau) \text{ es } U \text{ Entonces } y(\tau + 1) \text{ es } c \quad (4.2)$$

Sea N el total de conjuntos difusos de A_i correspondientes al estado $x(\tau)$ y M el total de conjunto difusos U_j definidos para la entrada de $u(\tau)$. Se asume que la base de reglas consiste de todas las posibles combinaciones de A_i y U_j lo que genera una serie de consecuentes unitarios que se muestran en la figura 4.10a.

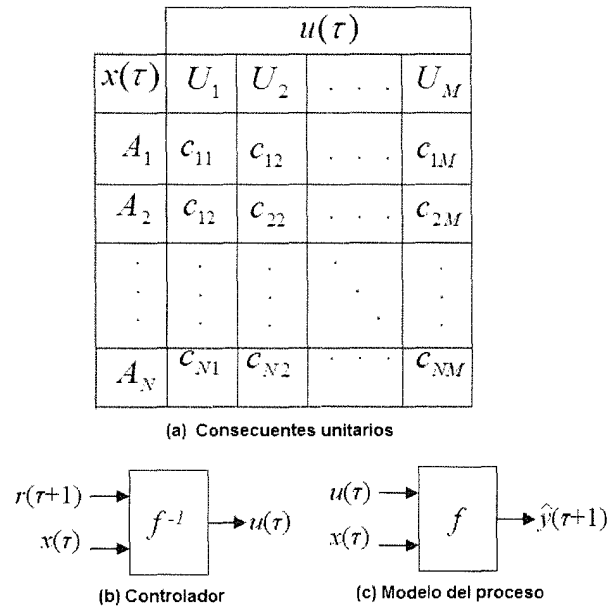


Figura 4.10: Modelo difuso y su inversa

Se asume que el conectivo y es representado por la t-norma del producto, de tal forma que el grado de cumplimiento de cada regla es expresado como:

$$\beta_{ij} = \mu_{A_i}(x(\tau)) \cdot \mu_{U_j}(u(\tau)) \quad (4.3)$$

En donde $\mu_{A_i}(x(\tau))$ es el grado de membresía particular del estado $x(\tau)$ en el conjunto difuso A_i y $\mu_{U_j}(u(\tau))$ es el grado de membresía de la entrada $u(\tau)$ en el conjunto difuso U_j . La salida estimada $\hat{y}(\tau + 1)$ del modelo es calculado por el método de la desfusificación de la media difusa, en donde el promedio de los consecuentes c_{ij} es ponderado por los grados de membresía β_{ij} tal que:

$$\hat{y}(\tau + 1) = \frac{\sum_{i=1}^N \sum_{j=1}^M \beta_{ij}(\tau) c_{ij}}{\sum_{i=1}^N \sum_{j=1}^M \beta_{ij}(\tau)} \quad (4.4)$$

Si esta clase de modelos difusos unitarios tiene funciones de membresía triangulares en los antecedentes, y forman una partición tal que $\sum_{i=1}^N \mu_{A_i}(x(\tau)) = 1, \forall x$, y $\sum_{j=1}^M \mu_{U_j}(u(\tau)) = 1, \forall u$ el modelo difuso unitario tiene la capacidad de proveer interpolación lineal entre los consecuentes de las reglas. La base de reglas corresponde al modelo de regresión no lineal (figura 4.10b) dado por:

$$\hat{y}(\tau + 1) = f(x(\tau), u(\tau)) \quad (4.5)$$

Dado el estado actual del sistema $x(\tau)$ y dada la referencia en el siguiente instante de tiempo $r(\tau + 1)$, el objetivo del algoritmo de control es encontrar $u(\tau)$ tal que la salida del modelo $\hat{y}(\tau + 1)$ esté tan cercana como sea posible de la referencia deseada $r(\tau + 1)$. Esto se logra al invertir el modelo de la planta tal y como lo muestra la figura 4.10c, reemplazando la referencia $\hat{y}(\tau + 1)$ por $r(\tau + 1)$ de tal manera que $u(\tau)$ quedaría expresado como:

$$u(\tau) = f^{-1}(x(\tau), r(\tau + 1)) \quad (4.6)$$

Esta técnica fue propuesta por Babuska (Babuska 1996). El mapeo multivariable del modelo difuso de la ecuación (4.5) se puede reducir a la expresión univariable $\hat{y}(\tau + 1) = f_x(u(\tau))$ haciendo uso de la estructura del modelo. El subíndice x denota que f_x se obtiene para el estado particular $x(\tau)$. Si el modelo es invertible, el mapeo inverso $u(\tau) = f_x^{-1}(r(\tau + 1))$ puede ser obtenido. El concepto de invertibilidad está relacionada con la monotonocidad del mapeo de entrada y salida del modelo. Un modelo difuso f dado por la base de reglas de la ecuación (4.5) es invertible si $\forall x$ y $\forall y$, existe una única u tal que $y = f(x, u)$. En términos de los parámetros del modelo, la monotonocidad se traduce por medio de la siguiente condición:

$$Nucleo(U_1) < \dots < Nucleo(U_M) \rightarrow c_{i1} < c_{i2} < \dots < c_{iM} \quad (4.7)$$

O en su defecto:

$$Nucleo(U_1) > \dots > Nucleo(U_M) \rightarrow c_{i1} > c_{i2} > \dots > c_{iM} \quad (4.8)$$

Por otra parte, se debe cumplir que la cardinalidad del núcleo de U_j denotada como $Card(Nucleo(U_j))$ sea igual a 1 para $j = 1, 2, \dots, M$. En la figura 4.11, se muestra un ejemplo en donde ambas condiciones no se cumplen. Por una parte, el conjunto difuso U_3 no cumple con la condición $Card(Nucleo(U_3)) = 1$; y por otra parte $c_{i1} < c_{i2} < c_{i3}$ pero $c_{i3} < c_{i4}$ no se cumple con lo cual no se cumple la condición de monotonocidad. Formalmente, se establecerá el siguiente teorema.

Teorema 4.1 Inversión de modelos difusos unitarios. Sea cualquier proceso representado por la ecuación (4.5) con método de desdifusificación por la media de centros expresado por medio de la ecuación (4.4). Además las funciones de membresía del antecedente forman particiones triangulares tales que $\sum_{i=1}^N \mu_{A_i}(x(\tau)) = 1, \forall x$, y $\sum_{j=1}^M \mu_{U_j}(u(\tau)) = 1, \forall u$. Para un cierto tiempo τ , el sistema está en el estado $x(\tau)$ y la inversa del modelo difuso viene dado por:

$$\text{Si } r(\tau + 1) \text{ es } C_j \text{ Entonces } u(\tau) \text{ es } U_j, j = 1, 2, \dots, M \quad (4.9)$$

En donde C_j son los conjuntos difusos que forman particiones tal y como lo muestra la figura 4.12. Los núcleos C_j se calculan a través de la siguiente ecuación:

4.3. INVERSIÓN DE MODELOS DIFUSOS

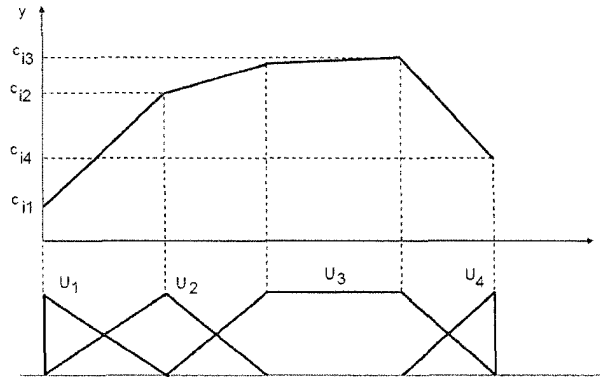


Figura 4.11: Ejemplo de condiciones no cumplidas de invertibilidad

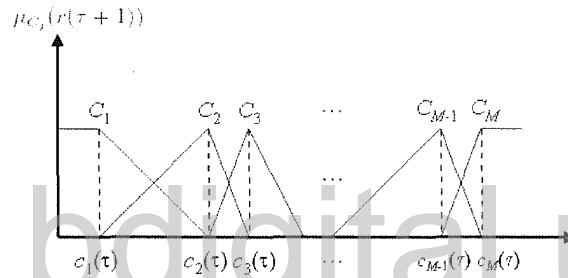


Figura 4.12: Particiones de los conjuntos difusos $C_j(\tau)$

$$C_j(\tau) = \sum_{i=1}^N \mu_{A_i}(x(\tau))c_{ij}, j = 1, \dots, M. \quad (4.10)$$

La inferencia y la desdifusificación de las reglas de la ecuación 4.9 se calcula por el método de la media difusa:

$$u(\tau) = \sum_{j=1}^M \mu_{C_j}(r(\tau+1)) \cdot \text{Nucleo}(U_j) \quad (4.11)$$

El lazo abierto que conecta la acción de control resultante de la inversión y el modelo difuso unitario permite obtener un mapeo idéntico (control perfecto):

$$y(\tau+1) = f_x(u(\tau)) = f_x(f_x^{-1}(r(\tau+1))) = r(\tau+1) \quad (4.12)$$

4.3.2. Inversión de modelos Takagi - Sugeno

La inversión de modelos difusos TS afines con respecto a la señal de control fue propuesto por (Sousa and Kaymak 2002). En esta sección se presentará el cálculo de la inversa de modelos difusos de la forma $y(\tau + 1) = f(x(\tau), u(\tau))$ en donde f es una función no lineal desconocida parametrizada por el modelo difuso TS, la cual puede aproximar una gran clase de sistemas no lineales. El modelo difuso TS más común usa una función afín dada en forma general por:

$$R_k : \text{ Si } x(\tau) \text{ es } A_k \text{ y } u(\tau) \text{ es } B_k \text{ Entonces } \hat{y}(\tau + 1) = a_k x(\tau) + b_k u(\tau) + c_k \quad (4.13)$$

La base de reglas dada por (4.13) puede ser expresada en una forma compacta al usar el vector de estado $x(\tau)$ y la acción de control $u(\tau)$ como:

$$R_k : \text{ Si } [x(\tau), y(\tau)] \text{ es } D_k \text{ Entonces } \hat{y}_k(\tau + 1) = a_k x(\tau) + b_k u(\tau) + c_k \quad (4.14)$$

En donde $k = 1, 2, \dots, K$ y D_k es el conjunto difuso multidimensional del antecedente, definida por su función de membresía dada por:

$$\mu_{D_k}(x(\tau), u(\tau)) : R^{n+1} \rightarrow [0, 1] \quad (4.15)$$

lo cual resulta de la agregación conjuntiva de A_k y B_k . La salida del modelo puede ser expresada como:

$$\hat{y}(\tau + 1) = \sum_{k=1}^K \hat{\beta}_k(\tau) [a_k x(\tau) + b_k u(\tau) + c_k] \quad (4.16)$$

En donde $\hat{\beta}_k(\tau)$ es el grado normalizado de cumplimiento de la k-ésima regla del antecedente y está dado por:

$$\hat{\beta}_k(\tau) = \frac{\mu_{D_k}(x(\tau), u(\tau))}{\sum_{j=1}^K \mu_{D_j}(x(\tau), u(\tau))} \quad (4.17)$$

La ecuación (4.16) puede ser reescrita de la siguiente forma:

$$\hat{y}(\tau + 1) = \sum_{k=1}^K \hat{\beta}_k(\tau) [a_k x(\tau) + c_k] + \sum_{k=1}^K \hat{\beta}_k(\tau) b_k u(\tau) \quad (4.18)$$

Si hacemos:

$$f_1(x(\tau)) = \sum_{k=1}^K \hat{\beta}_k(\tau) [a_k x(\tau) + c_k] \quad (4.19)$$

Y además:

$$f_2(x(\tau)) = \sum_{k=1}^K \hat{\beta}_k(\tau) b_k \quad (4.20)$$

Entonces tomando en cuenta que el objetivo de control es hacer que $\hat{y}(\tau + 1) = r(\tau + 1)$, la correspondiente señal de control es calculada haciendo una simple manipulación algebraica:

$$u(\tau) = \frac{r(\tau + 1) - f_1(x(\tau))}{f_2(x(\tau))} \quad (4.21)$$

Finalmente, en términos de la ecuación (4.16), $u(\tau)$ se expresa de la siguiente forma:

$$u(\tau) = \frac{r(\tau + 1) - \sum_{k=1}^K \hat{\beta}_k(\tau) [a_k x(\tau) + c_k]}{\sum_{k=1}^K \hat{\beta}_k(\tau) b_k} \quad (4.22)$$

Así que, de manera análoga al modelo difuso unitario, el modelo difuso TS afín con respecto a la señal de control puede ser invertido siempre y cuando la función $f_2(x(\tau))$ sea diferente de cero.

4.4. Modelación matemática de procesos de petróleo

La implementación, en términos computacionales, del esquema de control por modelo interno requiere de un modelo matemático del proceso que se desea controlar. En principio, sería suficiente con disponer de un modelo matemático lineal. Sin embargo, debe quedar claro, que la gran mayoría de los procesos industriales son de naturaleza no lineal.

Es por ello que partiendo de las ecuaciones que rigen el comportamiento de los procesos tanto de separación gas - líquido (cuya unidad de producción se conoce como separador de producción horizontal) como de calentamiento de crudo a través de un horno de fuego directo, en las próximas dos secciones se va a calcular un modelo matemático linealizado alrededor de un punto de operación de tal manera de poder implementar las simulaciones en Matlab-Simulink.

4.4.1. Modelación matemática del separador horizontal de producción

El modelo matemático de un separador de producción horizontal para propósitos de control debe contemplar tres elementos importantes: el actuador, la válvula y el separador. Específicamente, se trata de un separador bifásico horizontal y se aplicará el principio de la conservación de la masa del componente líquido. Al aplicar este principio, tomando en cuenta que $m = \rho V$, se obtiene que :

$$\rho V_{liq_{entrante}}(t) - \rho V_{liq_{saliente}}(t) = \rho \frac{dV_{sep}(t)}{dt} \quad (4.23)$$

En donde ρ es la densidad del crudo, $V_{liq_{entrante}}(t)$ es el volumen de líquido que entra en el separador, $V_{liq_{saliente}}(t)$ es el volumen de líquido que sale del separador y $dV_{sep}(t)/dt$ es la variación del volumen de líquido dentro del separador. El volumen de líquido que entra en un separador se va a asumir por ahora constante, sin embargo, es importante destacar que normalmente no es así ya que esta variable esta asociada a la producción de un grupo de pozos de petróleo y el flujo de salida de un pozo es no lineal dependiendo de su método de levantamiento y de su estado de operación.

El volumen de líquido que sale de un separador depende de la posición de la válvula, la cual a su vez, depende de la señal del actuador. Es importante aclarar que el volumen de crudo que sale del separador es directamente proporcional a la presión del separador, es decir, que para una determinada apertura de la válvula de control a mayor presión mayor será el flujo volumétrico de crudo. Para este caso particular, el actuador es de tipo neumático y la válvula es de tipo bola.

La señal eléctrica de la salida del controlador (4 a 20 miliamperios) es convertida a una señal de presión por el actuador neumático a través del convertidor I/P (convertidor de corriente a presión). La señal de presión neumática enviada de la salida del controlador a la línea neumática viene dada por:

$$p_c = pv_{min} + (pv_{max} - pv_{min}) \frac{(E_c - 4)}{16} \quad (4.24)$$

donde p_c es la salida del controlador convertida en presión neumática, pv_{min} es el límite de presión inferior y pv_{max} es el límite superior de la presión de trabajo y E_c es la salida del controlador en términos de la corriente (4 a 20 mA). El actuador neumático puede simularse, según (Shankar and Shoubo 2003) a través de la siguiente ecuación:

$$p_v = p_c + (p_{v0} - p_c) e^{-\frac{t}{\tau_0}} \quad (4.25)$$

4.4. MODELACIÓN MATEMÁTICA DE PROCESOS DE PETRÓLEO

donde p_v es la presión neumática recibida por la válvula, p_{v0} es la presión inicial de la válvula y τ_0 es la constante de tiempo de la línea neumática. Si se sustituye la ecuación (4.24) en la ecuación (4.23) y se toma en cuenta que las variables que dependen del tiempo son $E_c(t)$ y $p_v(t)$, entonces al derivar con respecto al tiempo, reagrupar términos y aplicar la transformada de Laplace, de acuerdo con (Shankar and Shoubo 2003), se obtiene que:

$$\frac{p_v(s)}{E_c(s)} \simeq \frac{p_c}{(\tau_0 s + 1)} \quad (4.26)$$

Con respecto a la función de transferencia de la válvula, ésta se considera de primer orden, en donde la entrada es la presión proveniente del actuador neumático y la salida es una tasa de flujo volumétrico. Se debe tomar en cuenta el rango de trabajo de la presión de operación de la válvula, la curva características de la válvula (lineal en este caso) y C_v el coeficiente de flujo de la válvula. Se asume la caída de presión de la válvula constante e igual a 10 Psig.

La constante de tiempo de la válvula puede ser determinada experimentalmente o dada por el fabricante. La tabla de los coeficientes de flujo de la válvula viene dada directamente por el fabricante. La función de transferencia de la válvula quedaría definida de la siguiente forma:

$$\frac{Q_o(s)}{p_v(s)} = \frac{-100/pv_{lim}}{(C_0 s + 1)} * C_v * \sqrt{\frac{\Delta P}{g_e}} \quad (4.27)$$

En donde el factor 100 corresponde al recorrido total de la válvula, el signo menos proviene de la acción reversa de la válvula, pv_{lim} es el rango de operación de la válvula de control (en este caso de 3 a 15 Psi), Q_o es el flujo de salida de crudo de la válvula que a su vez es el flujo de salida de crudo del separador, C_0 es la constante de tiempo de la válvula, ΔP es la caída de presión a lo largo de la válvula, asumida como una constante; y g_e es la gravedad específica del crudo, en este caso para el tipo de crudo que hemos considerado es igual a 0,92.

El cálculo del modelo matemático del separador horizontal contempla la ecuación de su volumen que viene expresado en función de la longitud L , el radio r medido desde x_1 y x_2 , el radio a medido desde x_2 y x_3 y la altura h que representa el nivel de crudo del separador tal y como lo muestra la figura 4.13. En base a las dimensiones del separador de producción horizontal se determinan los valores de L , r y a por lo que el volumen $V = f(h)$ es una función que depende de h que es el nivel instantáneo de crudo.

La clave para el cálculo del volumen de un separador horizontal, se basa en la figura 4.13, en donde, se aprecia que hay un cilindro horizontal parcialmente

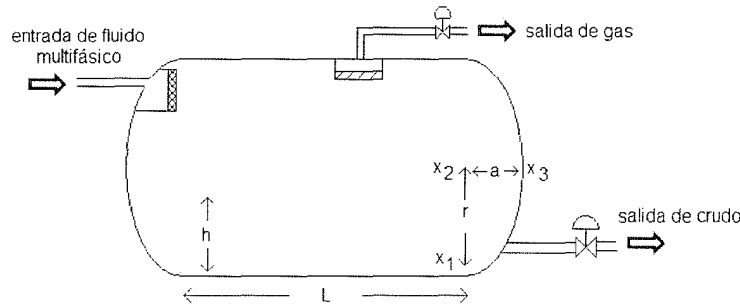


Figura 4.13: Dimensiones del separador horizontal de producción

lleno y a los lados hay una semi-elipsoide lo cual forma un elipsoide completo. El volumen del cilindro horizontal V_{ch} viene dado por la siguiente ecuación:

$$V_{ch} = L(r^2 \cos^{-1}(\frac{r-h}{r}) - (r-h)\sqrt{2rh-h^2}) \quad (4.28)$$

Mientras que el volumen de elipsoide completo viene dado por:

$$V_e = \frac{a\pi}{r^3}(h)(3hr-h^2) \quad (4.29)$$

Por lo tanto, el volumen del separador horizontal $V_{sep_H}(h)$ viene dado por la suma de $V_{ch} + V_e$:

$$V_{sep_H}(h) = L(r^2 \cos^{-1}(\frac{r-h}{r}) - (r-h)\sqrt{2rh-h^2}) + \frac{a\pi}{r^3}(h)(3hr-h^2) \quad (4.30)$$

En cuanto a las dimensiones reales del separador horizontal de producción se tiene que $L = 3,5m$, $r = 1,1m$ y $a = 0,90m$. A fin de obtener una relación lineal entre V_{sep_H} y h , es necesario expandir en series de Taylor la ecuación anterior hasta la primera derivada y evaluar en el punto de operación $\bar{h} = 3pies$ se obtiene que:

$$V_{sep_H} = V_{sep_H}(\bar{h}) + \frac{dV_{sep_H}(\bar{h})}{dh}(h-\bar{h}) = 0,00234 + 0,7656h - 0,00234 = 0,7656h \quad (4.31)$$

En términos de la transformada de Laplace tendríamos que:

$$\frac{H(s)}{V(s)} = \frac{1,306}{s} \quad (4.32)$$

Por ende el modelo de la planta puede ser representado tal y como lo muestra la siguiente figura 4.14.

4.4. MODELACIÓN MATEMÁTICA DE PROCESOS DE PETRÓLEO

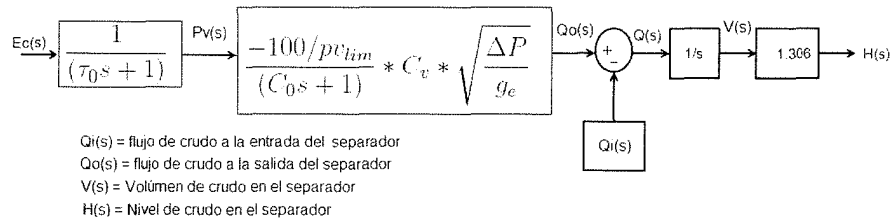


Figura 4.14: Modelo matemático aproximado del separador horizontal de producción

4.4.2. Modelación matemática del horno de fuego directo

Para definir un modelo matemático de un horno de fuego directo para propósitos de control se deben contemplar al igual que en el caso del separador tres elementos importantes: el actuador, la válvula y el horno de fuego directo. Las consideraciones realizadas para el actuador y la válvula planteadas en el separador de producción aplican para el modelo del horno sólo que los parámetros como C_0 , τ_0 , etc tienen otros valores.

La idea clave para obtener un modelo matemático para el horno de fuego directo es conocer el funcionamiento del horno, el tipo de transferencia de calor que ocurre y los equipos de instrumentación involucrados. Básicamente, el calor que se libera por la combustión del material combustible que en este caso es una parte del gas que proviene de los separadores de producción y que es generado por un equipo llamado quemador, se transfiere al fluido de crudo que circula dentro de largos tubos instalados cerca de las paredes del horno.

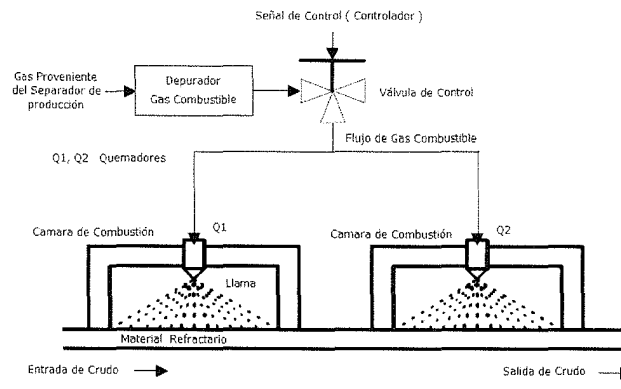


Figura 4.15: Esquemático del funcionamiento de un horno de fuego directo

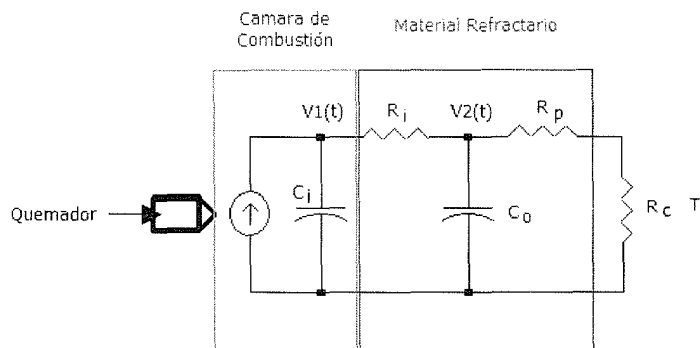


Figura 4.16: Configuración del circuito eléctrico

El flujo de gas combustible que sale del depurador es controlado por la válvula de control de temperatura del horno la cual distribuye de manera equitativa el flujo de gas combustible a cada quemador. Por lo general, los hornos de fuego directo instalados en las empresas petroleras tienen instalados seis, siete y hasta ocho quemadores dependiendo de la capacidad de calentamiento de cada horno. En la figura 4.15, se muestra un esquemático que muestra el funcionamiento del horno de fuego directo.

El quemador es un dispositivo mecánico diseñado para producir una llama estable con una forma y tamaño predeterminado. Gracias al quemador se lleva a cabo el proceso de transferencia de calor por radiación. En términos formales, la radiación térmica es la energía emitida por la materia que se encuentra a una temperatura dada y se produce directamente de la fuente hacia afuera en todas direcciones.

La cámara de combustión y la tubería trabajan en conjunto para transferir el calor al crudo que entra al horno de fuego directo. Del sistema mecánico conformado por la cámara de combustión y la tubería se determinará un sistema eléctrico con su respectivo circuito y su función de transferencia. Dicho circuito se muestra en la siguiente figura 4.16, en donde se puede realizar las siguientes analogías:

- La corriente I representa el flujo de calor entregado por el quemador.
- La capacitancia C_i representa la capacidad calorífica de la cámara de combustión.
- La capacitancia C_o representa la capacidad calorífica de la tubería por donde

4.4. MODELACIÓN MATEMÁTICA DE PROCESOS DE PETRÓLEO

circula el fluido.

- La resistencia R_i representa la resistencia térmica de la tubería por donde circula el fluido.
- La resistencia R_p representa la resistencia de pérdidas.
- La resistencia R_c representa la resistencia de consumo.

Tabla 4.1: Comparación de los parámetros de un circuito eléctrico y un horno

Parámetro Eléctrico	Parámetro Térmico	Unidades
Corriente (I)	Calor (W)	$[Kcal/seg]$
Tensión (V)	Temperatura (T)	$[F]$
Resistencia (R)	Resistencia Térmica (R)	$[F/(Kcal/seg)]$

En general es posible establecer una correspondencia entre la Ley de Ohm y la propagación térmica mediante la tabla 4.1.

Cálculo de la capacidad calorífica de la cámara de combustión (C_i): es la capacidad de calor de la cámara de combustión, la cual depende de las propiedades del material refractario del horno. Se calcula a través de la siguiente ecuación: $C_i = m.c$ en donde m es la masa del material refractario y c es el calor específico de dicho material y de acuerdo a datos suministrado por el fabricante es igual $0.239 Kcal/F.Kg$. A su vez $m = \delta.v$, en donde δ es la densidad del material refractario y es igual a $2000 Kg/m^3$ y v es el volumen de las paredes de la cámara de combustión. Por lo tanto $C_i = 1816 \frac{Kcal}{F}$.

Cálculo de la capacidad calorífica de la pared del horno (C_o): como su nombre lo indica es la capacidad calorífica que tiene la pared del horno para entregar calor a la tubería por donde circula el fluido. Se calcula a través de la siguiente ecuación: $C_o = v.\delta.c$. De acuerdo a los datos suministrado por el fabricante $C_o = 13,9m^3 \times 2000Kg/m^3 \times 0,239Kcal/(F.Kg) = 6668,1 \frac{Kcal}{F}$.

Resistencia Térmica de la pared del horno (R_i): es la resistencia que presenta el horno al paso del calor. Se define a través de la siguiente ecuación:

$$R_i = \frac{\Delta x}{k.A} \quad (4.33)$$

En donde Δx es la longitud de la pared del horno, A es el área de la pared del horno y k es la conductividad térmica del material refractario y es igual a $91Kcal/(m.h.F)$. De acuerdo a los datos del horno, se tiene entonces que

$$R_i = 2,452F/(Kcal/seg).$$

Resistencia de consumo (R_c): es la resistencia que se presenta a la pérdida de calor que sufre la capacidad calorífica de la pared del horno por el paso del fluido. por definición *Resistencia = temperatura/Flujodecalor*. La temperatura a la cual va a trabajar el horno es de 190 F y el flujo de calor se calcula a través de la ecuación de transmisión de calor entre dos cuerpos:

$$Q = (m_c)(C_{e_c})(T_f - T_i) \quad (4.34)$$

En donde Q es la suma de la potencia calorífica emitida por los calentadores m_b es la masa del fluido de crudo, C_{e_c} es el calor específico del crudo, T_f es la temperatura del crudo a la salida del horno y T_i es la temperatura del crudo a la entrada del horno. Si la transmisión de calor se divide por el tiempo se obtiene el flujo de calor, por lo que:

$$H = \frac{Q}{t} = \frac{(m_c)(C_{e_c})}{t}(T_f - T_i) = 60,12 \frac{F}{Kcal/seg} \quad (4.35)$$

Resistencia de pérdidas (R_p): es la resistencia que se debe a las pérdidas por conducción, convección y radiación. Se asume una eficiencia del horno de un 75 %. Por ende el flujo de calor que se pierde es de aproximadamente un 25 %, es decir, 23934,8Kcal/h. De tal forma que:

$$R_p = \frac{T}{H} = 180,36FKcal/seg \quad (4.36)$$

Ahora bien, del circuito mostrado en la figura, se aplica la Ley de Kirchoff, se obtendrían las siguientes ecuaciones:

$$-C_i \frac{dV_1(t)}{dt} - \frac{1}{R_i}(V_1(t) - V_2(t)) + i(t) = 0 \quad (4.37)$$

$$-C_o \frac{dV_2(t)}{dt} - \frac{1}{R_p + R_c}V_2(t) + \frac{1}{R_i}(V_1(t) - V_2(t)) = 0 \quad (4.38)$$

Si despejamos $V_1(t)$ de la ecuación 4.38 y los sustituimos en la ecuación 4.37 se obtendría que:

$$R_i C_i C_o \frac{d^2 V_2(t)}{dt} + (C_i + C_o + \frac{R_i C_i}{R_p + R_c}) \frac{dV_2(t)}{dt} + \frac{1}{R_p + R_c} V_2(t) = i(t) \quad (4.39)$$

Para obtener la función de transferencia $C(s)$, entre $V_2(s)$ y $I(s)$ del horno, se aplica la transformada de laplace, obteniéndose la siguiente ecuación:

$$C(s) = \frac{V_2(s)}{I(s)} = \frac{1}{(R_i C_i C_o) s^2 + (C_i + C_o + \frac{R_i C_i}{R_p + R_c}) s + \frac{1}{R_p + R_c}} \quad (4.40)$$

Reemplazando los valores de C_i , C_o , R_i y R_o se tiene que:

$$C(s) = \frac{1}{12,1234s^2 + 2,2345s + 1,3458} \quad (4.41)$$

4.5. Implementación del control basado en modelos difusos usando CMI para el separador de producción

En esta sección, en base al modelo matemático linealizado del separador de producción horizontal obtenido en la sección 4.4.1, se mostrarán los resultados del desempeño del sistema de control basado en el modelo difuso del proceso de separación a través del uso la arquitectura de control por modelo interno tal y como se desarrollo en (Ramirez and Colina 2008b).

El modelo difuso original del separador, basado en el desarrollo realizado en la sección 3.6.1 del capítulo 3, presenta la siguiente base de reglas:

- Si $y(\tau)$ es bajo y $u(\tau)$ es abierto Entonces $\hat{y}(\tau + 1)$ es 1,72
- Si $y(\tau)$ es bajo y $u(\tau)$ es poco abierto Entonces $\hat{y}(\tau + 1)$ es 1,80
- Si $y(\tau)$ es bajo y $u(\tau)$ es poco cerrado Entonces $\hat{y}(\tau + 1)$ es 1,91
- Si $y(\tau)$ es bajo y $u(\tau)$ es muy cerrado Entonces $\hat{y}(\tau + 1)$ es 2,10
- Si $y(\tau)$ es normal y $u(\tau)$ es abierto Entonces $\hat{y}(\tau + 1)$ es 2,18
- Si $y(\tau)$ es normal y $u(\tau)$ es poco abierto Entonces $\hat{y}(\tau + 1)$ es 2,42
- Si $y(\tau)$ es normal y $u(\tau)$ es poco cerrado Entonces $\hat{y}(\tau + 1)$ es 2,49
- Si $y(\tau)$ es normal y $u(\tau)$ es muy cerrado Entonces $\hat{y}(\tau + 1)$ es 2,60
- Si $y(\tau)$ es alto y $u(\tau)$ es abierto Entonces $\hat{y}(\tau + 1)$ es 3,18
- Si $y(\tau)$ es alto y $u(\tau)$ es poco abierto Entonces $\hat{y}(\tau + 1)$ es 3,23
- Si $y(\tau)$ es alto y $u(\tau)$ es poco cerrado Entonces $\hat{y}(\tau + 1)$ es 3,33
- Si $y(\tau)$ es alto y $u(\tau)$ es muy cerrado Entonces $\hat{y}(\tau + 1)$ es 3,43
- Si $y(\tau)$ es muy alto y $u(\tau)$ es abierto Entonces $\hat{y}(\tau + 1)$ es 3,65
- Si $y(\tau)$ es muy alto y $u(\tau)$ es poco abierto Entonces $\hat{y}(\tau + 1)$ es 3,92

- Si $y(\tau)$ es muy alto y $u(\tau)$ es poco cerrado Entonces $\hat{y}(\tau + 1)$ es 4,02
- Si $y(\tau)$ es muy alto y $u(\tau)$ es muy cerrado Entonces $\hat{y}(\tau + 1)$ es 4,18

De acuerdo al teorema 4.1, el modelo difuso unitario inverso o controlador difuso unitario es por naturaleza de tipo adaptativo dado que para un cierto tiempo τ , el modelo difuso del proceso se encuentra en el estado $x(\tau) = y(\tau)$ y en la medida en que $y(\tau)$ varía, entonces los núcleos $c_j(\tau)$ de los términos lingüístico C_j de la variable $r(\tau + 1)$ varían también, tal y como lo establece la ecuación (4.12). El controlador difuso, en forma genérica, vendría dado por la siguiente base de regla: Si $r(\tau + 1)$ es C_j Entonces $u(\tau)$ es $U_j, j = 1, 2, \dots, 4$.

Por ejemplo, en $\tau = 0$, el controlador difuso unitario viene dado por:

- Si $r(\tau + 1)$ es C_1 Entonces $u(\tau)$ es U_1
- Si $r(\tau + 1)$ es C_2 Entonces $u(\tau)$ es U_2

En donde el núcleo $c_1(\tau)$ de C_1 es igual a 1.84 y el núcleo $c_2(\tau)$ de C_2 es igual 2.03 mientras que $c_3(\tau)$ y $c_4(\tau)$ son iguales a 0. El setpoint en $t = 0$ fue cambiado de 2,0 pies a 2,75 pies. Entonces a medida que se incrementa $y(\tau)$, comienzan a variar los núcleos $c_j(\tau)$, hasta que el sistema en lazo cerrado se estabilice en 2.745 pies. En ese instante el controlador difuso unitario viene dado por:

- Si $r(\tau + 1)$ es C_2 Entonces $u(\tau)$ es U_2
- Si $r(\tau + 1)$ es C_3 Entonces $u(\tau)$ es U_3

En este caso, $c_1(\tau) = 0$ y $c_3(\tau) \neq 0$ y así sucesivamente. Por ejemplo, en $t = 60$ cuando la variable $y(\tau)$ tiende a $r(\tau + 1) = 3,75$, $c_1(\tau) = c_2(\tau) = 0$ y $c_3(\tau)$ y $c_3(\tau)$ son distintos de cero.

La figura 4.17, muestra el resultado del desempeño del sistema de control aplicado al proceso de separación gas-líquido. La señal de salida de control del controlador se muestra en la figura 4.18. La tabla 4.2 muestra la variación de los valores de los núcleos $c_j(\tau)$ de C_j de la entrada $\mu_{r(\tau+1)}$ a medida que $y(\tau)$ tiende a $\mu_{r(\tau+1)} = 2,75$ pies.

En cuanto al desempeño del controlador, de acuerdo a los resultados obtenidos en la gráfica de la figura 4.17, se puede afirmar que el controlador es estable de manera empírica y presenta alta robustez. El diseño apropiado del filtro F garantiza tal robustez ya que reduce la ganancia del sistema realimentado y permite suavizar los cambios rápidos y bruscos de la señal controlada producto de la presencia de perturbaciones y/o errores de modelación.

4.5. IMPLEMENTACIÓN DEL CONTROL BASADO EN MODELOS DIFUSOS USANDO CMI PARA EL SEPARADOR DE PRODUCCIÓN

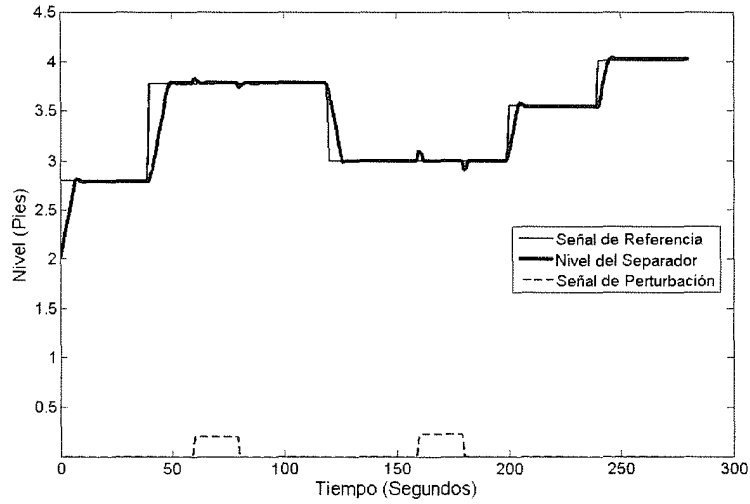


Figura 4.17: Desempeño del sistema de control ICM para el separador

www.bdigital.ula.ve

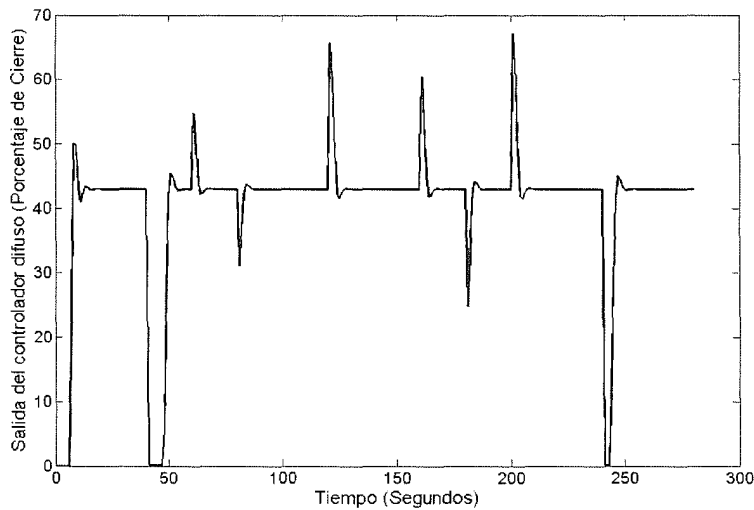


Figura 4.18: Comportamiento de la válvula de control del separador

Tabla 4.2: Variaciones de los núcleos $c_j(\tau)$ de C_j (separador)

Valores de los núcleos $c_j(\tau)$	t	$y(\tau)$	$r(\tau + 1)$
$c_1(\tau) = 1.84; c_2(\tau) = 2.03; c_3(\tau) = 0; c_4(\tau) = 0$	0	2	2.75
$c_1(\tau) = 1.91; c_2(\tau) = 2.14; c_3(\tau) = 0; c_4(\tau) = 0$	1	2.08	2.75
$c_1(\tau) = 1.97; c_2(\tau) = 2.24; c_3(\tau) = 0; c_4(\tau) = 0$	2	2.16	2.75
$c_1(\tau) = 1.99; c_2(\tau) = 2.29; c_3(\tau) = 0; c_4(\tau) = 0$	3	2.21	2.75
$c_1(\tau) = 0; c_2(\tau) = 2.11; c_3(\tau) = 2.37; c_4(\tau) = 0$	4	2.27	2.75
$c_1(\tau) = 0; c_2(\tau) = 2.18; c_3(\tau) = 2.43; c_4(\tau) = 0$	5	2.34	2.75
$c_1(\tau) = 0; c_2(\tau) = 2.23; c_3(\tau) = 2.65; c_4(\tau) = 0$	6	2.47	2.75
$c_1(\tau) = 0; c_2(\tau) = 2.25; c_3(\tau) = 2.78; c_4(\tau) = 0$	7	2.58	2.75
$c_1(\tau) = 0; c_2(\tau) = 2.35; c_3(\tau) = 2.81; c_4(\tau) = 0$	8	2.69	2.75
$c_1(\tau) = 0; c_2(\tau) = 2.36; c_3(\tau) = 2.89; c_4(\tau) = 0$	9	2.79	2.75
$c_1(\tau) = 0; c_2(\tau) = 2.44; c_3(\tau) = 2.98; c_4(\tau) = 0$	10	2.88	2.75

En cuanto al tiempo de asentamiento y el sobreimpulso, en este caso particular, se muestra que la respuesta del controlador es relativamente rápida con tiempo de asentamiento de 12 segundos y con un sobreimpulso de apenas un 5%.

4.6. Implementación del control basado en modelos difusos usando CMI para el horno de producción

En esta sección, en base al modelo matemático aproximado del separador de producción horizontal obtenido en la sección 4.4.2, se mostrarán los resultados del desempeño del sistema de control basado en el modelo difuso del proceso de calentamiento de crudo a través del uso la arquitectura de control por modelo interno. En la figura 4.19, se muestra el desempeño del contrador difuso unitario obtenido usando la inversa del mismo.

El modelo difuso original del horno de fuego directo, basado en el desarrollo realizado en la sección 3.6.2 del capítulo 3, presenta la siguiente base de reglas:

- Si $y(\tau)$ es muy baja y $u(\tau)$ es muy cerrada Entonces $\hat{y}(\tau + 1)$ es 165.32
- Si $y(\tau)$ es muy baja y $u(\tau)$ es cerrada Entonces $\hat{y}(\tau + 1)$ es 168.17
- Si $y(\tau)$ es muy baja y $u(\tau)$ es media abierta Entonces $\hat{y}(\tau + 1)$ es 172.41

4.6. IMPLEMENTACIÓN DEL CONTROL BASADO EN MODELOS DIFUSOS USANDO CMI PARA EL HORNO DE PRODUCCIÓN

- Si $y(\tau)$ es muy baja y $u(\tau)$ es abierta Entonces $\hat{y}(\tau + 1)$ es 173.50
- Si $y(\tau)$ es muy baja y $u(\tau)$ es muy abierta Entonces $\hat{y}(\tau + 1)$ es 177
- Si $y(\tau)$ es baja y $u(\tau)$ es muy cerrada Entonces $\hat{y}(\tau + 1)$ es 167.32
- Si $y(\tau)$ es baja y $u(\tau)$ es cerrada Entonces $\hat{y}(\tau + 1)$ es 170.10
- Si $y(\tau)$ es baja y $u(\tau)$ es media abierta Entonces $\hat{y}(\tau + 1)$ es 173.3
- Si $y(\tau)$ es baja y $u(\tau)$ es abierta Entonces $\hat{y}(\tau + 1)$ es 174.52
- Si $y(\tau)$ es baja y $u(\tau)$ es muy abierta Entonces $\hat{y}(\tau + 1)$ es 181.34
- Si $y(\tau)$ es normal y $u(\tau)$ es muy cerrada cerrado Entonces $\hat{y}(\tau + 1)$ es 169.82
- Si $y(\tau)$ es normal y $u(\tau)$ es cerrada Entonces $\hat{y}(\tau + 1)$ es 173.41
- Si $y(\tau)$ es normal y $u(\tau)$ es media abierta Entonces $\hat{y}(\tau + 1)$ es 176.72
- Si $y(\tau)$ es normal y $u(\tau)$ es abierta Entonces $\hat{y}(\tau + 1)$ es 180.4
- Si $y(\tau)$ es normal y $u(\tau)$ es muy abierta Entonces $\hat{y}(\tau + 1)$ es 184.8
- Si $y(\tau)$ es alta y $u(\tau)$ es muy cerrada Entonces $\hat{y}(\tau + 1)$ es 173.1
- Si $y(\tau)$ es alta y $u(\tau)$ es cerrada Entonces $\hat{y}(\tau + 1)$ es 176.9
- Si $y(\tau)$ es alta y $u(\tau)$ es media abierta Entonces $\hat{y}(\tau + 1)$ es 179.8
- Si $y(\tau)$ es alta y $u(\tau)$ es abierta Entonces $\hat{y}(\tau + 1)$ es 183.5
- Si $y(\tau)$ es alta y $u(\tau)$ es muy abierta Entonces $\hat{y}(\tau + 1)$ es 187.9
- Si $y(\tau)$ es muy alta y $u(\tau)$ es muy cerrada Entonces $\hat{y}(\tau + 1)$ es 176.1
- Si $y(\tau)$ es muy alta y $u(\tau)$ es cerrada Entonces $\hat{y}(\tau + 1)$ es 179.3
- Si $y(\tau)$ es muy alta y $u(\tau)$ es media abierta Entonces $\hat{y}(\tau + 1)$ es 184.8
- Si $y(\tau)$ es muy alta y $u(\tau)$ es abierta Entonces $\hat{y}(\tau + 1)$ es 189.5
- Si $y(\tau)$ es muy alta y $u(\tau)$ es muy abierta Entonces $\hat{y}(\tau + 1)$ es 194.9

El controlador difuso, en forma genérica, vendría dado por la siguiente base de regla: Si $r(\tau + 1)$ es C_j Entonces $u(\tau)$ es $U_j, j = 1, 2, \dots, 5$. Por ejemplo, en $t = 0$, el controlador difuso unitario del horno viene dado por:

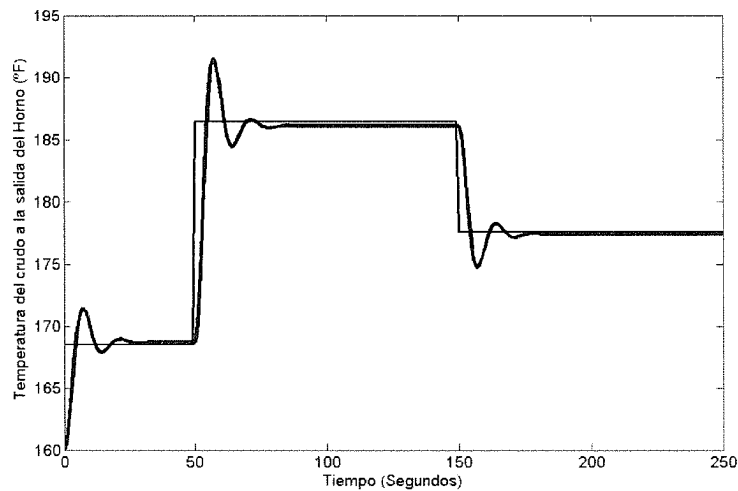


Figura 4.19: Desempeño del sistema de control ICM para el horno

- Si $r(\tau + 1)$ es C_1 Entonces $u(\tau)$ es U_1
- Si $r(\tau + 1)$ es C_2 Entonces $u(\tau)$ es U_2

En donde el núcleo $c_1(\tau)$ de C_1 es igual a 164.2 y el núcleo $c_2(\tau)$ de C_2 es igual 167.34 mientras que $c_3(\tau)$, $c_4(\tau)$ y $c_5(\tau)$ son iguales a 0. Como se puede observar en la figura 4.19, en $t = 0$, se realiza un cambio del setpoint de 160 a 168 grados fahrenheit. A medida que la temperatura instantánea $y(\tau)$ empieza a incrementarse para igualar a $r(\tau + 1) = 168$, los núcleos $c_j(\tau)$ empiezan a variar y en $t = 22$ cuando el sistema de control en lazo cerrado se estabiliza el controlador resultante queda de la siguiente forma:

- Si $r(\tau + 1)$ es C_2 Entonces $u(\tau)$ es U_2
- Si $r(\tau + 1)$ es C_3 Entonces $u(\tau)$ es U_3

En donde $c_1(\tau) = 0$, $c_2(\tau) = 166,3$, $c_3(\tau) = 172,3$ y $c_4(\tau)$ y $c_5(\tau)$ son iguales a cero.

Por otra parte, la señal de salida de control del controlador se muestra en la figura 4.20. Análogamente, la siguiente tabla muestra la variación de los valores de los núcleos $c_j(\tau)$ a medida que $y(\tau)$ tiende a $r(\tau + 1) = 168$ °F.

En cuanto al desempeño del controlador, podemos afirmar que ocurre la misma situación que en el caso del separador en cuanto a la estabilidad y robustez. Sin embargo, el sobrepulso es del orden del 25 % y el tiempo de asentamiento es de 27 segundos.

4.6. IMPLEMENTACIÓN DEL CONTROL BASADO EN MODELOS DIFUSOS USANDO CMI PARA EL HORNO DE PRODUCCIÓN

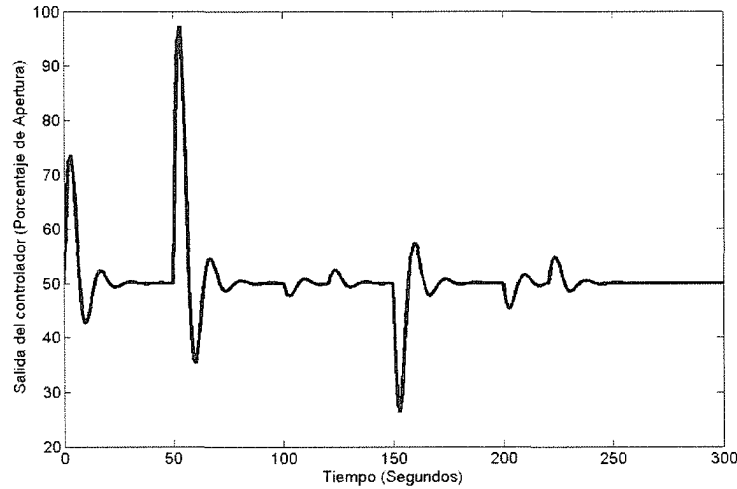


Figura 4.20: Comportamiento de la válvula de control de flujo de gas del horno

www.bdigital.ula.ve

Tabla 4.3: Variaciones de los núcleos $c_j(\tau)$ de C_j (horno)

Valores de los núcleos c_j	t	$y(\tau)$	$r(\tau + 1)$
$c_1(\tau)=158.3;c_2(\tau)=163.2;c_3(\tau)=0;c_4(\tau)=0$	0	168	180
$c_1(\tau) = 159.1;c_2(\tau) = 163.8;c_3(\tau) = 0;c_4(\tau) = 0$	1	169	180
$c_1(\tau) = 159.8;c_2(\tau) = 164.1;c_3(\tau) = 0;c_4(\tau) = 0$	2	170	180
$c_1(\tau) = 160;c_2(\tau) = 164.5;c_3(\tau) = 0;c_4(\tau) = 0$	3	171	180
$c_1(\tau) = 0;c_2(\tau) = 162.3;c_3(\tau) = 165.7;c_4(\tau) = 0$	4	172	180
$c_1(\tau) = 0;c_2(\tau) = 162.6;c_3(\tau) = 166.1;c_4(\tau) = 0$	5	174	180
$c_1(\tau) = 0;c_2(\tau) = 162.8;c_3(\tau) = 166.4;c_4(\tau) = 0$	6	176	180
$c_1(\tau) = 0;c_2(\tau) = 162.9;c_3(\tau) = 166.6;c_4(\tau) = 0$	7	179	180
$c_1(\tau) = 0;c_2(\tau) = 163.1;c_3(\tau) = 166.9;c_4(\tau) = 0$	8	185	180
$c_1(\tau) = 0;c_2(\tau) = 163.2;c_3(\tau) = 167.3;c_4(\tau) = 0$	9	182	180
$c_1(\tau) = 0;c_2(\tau) = 163.4;c_3(\tau) = 167.6;c_4(\tau) = 0$	10	178	180
$c_1(\tau) = 0;c_2(\tau) = 163.6;c_3(\tau) = 168;c_4(\tau) = 0$	11	182	180
$c_1(\tau) = 0;c_2(\tau) = 163.8;c_3(\tau) = 168.9;c_4(\tau) = 0$	12	180.3	180

4.7. Conclusiones

En este capítulo se hizo una revisión de las alternativas que permiten calcular la inversa de modelos difusos. Dos enfoques se plantean al respecto: el cálculo de la inversa en forma numérica y el cálculo de la inversa en forma analítica. En cuanto a la primera alternativa, el aprendizaje inverso directo y el aprendizaje inverso especializado son las metodologías más usadas. En ambas, metodologías pueden surgir inconvenientes con la determinación del modelo producto de una variación imprevista de los datos.

Se optó por usar el método analítico de Babuska (Babuska 1996) que permite el cálculo analítico del controlador difuso unitario. Fue necesario para ello considerar en el modelo difuso algunas restricciones en cuanto al tipo de función de membresía de las variables del antecedente y la monotonocidad de los consecuentes unitarios.

Adicionalmente se hizo una revisión exhaustiva de los diferentes enfoques para implementar sistemas de control basado en modelos difusos, entre los que se pueden destacar, el control por modelo inverso, el control adaptativo por modelo de referencia, el control predictivo basado en modelos y el control por modelo interno. Se optó por usar el control por modelo interno CMI dada sus características inherentes de estabilidad, control estable y error de estado estático pequeño.

Se obtuvieron los modelos matemáticos linealizados de los procesos de separación gas-líquido y calentamiento de crudo que son procesos complejos. Se destaca, como una contribución, el desarrollo del modelo matemático del horno de fuego directo en la cual se contemplan entre otros parámetros, el flujo másico de crudo que entra al horno, el calor específico del crudo y la temperatura de entrada del crudo.

Ambos modelos matemáticos permitieron implementar la arquitectura CMI en los procesos mencionados y se demostró el buen desempeño de los controladores tanto a nivel de estabilidad como de controlabilidad de manera empírica. Se destaca la alta robustez de este esquema de control gracias a la presencia del filtro, que en esencia, reduce la ganancia del sistema realimentado.

Finalmente, vale la pena destacar que los controladores difuso unitarios obtenidos a través método analítico son por naturaleza de tipo adaptativo, lo cual se demostró a través de la variación de los núcleos de los conjuntos difusos unitarios de las variables del antecedente cuando se cambia la consigna de operación.

Capítulo 5

PROPUESTA METODOLÓGICA DE UN SISTEMA DE SUPERVISIÓN

5.1. Introducción

El buen funcionamiento de una planta industrial viene condicionado por la eficacia y la eficiencia de cada uno de los tres niveles en los que se suele dividir la automatización de la misma: gestión, supervisión y control. Cada nivel tiene objetivos muy particulares y actúa sobre el nivel inmediatamente inferior para poder cumplirlos. En el nivel de gestión se establecen los objetivos, se realiza la planificación y tiene unos tiempos entre actuaciones sobre el nivel de supervisión relativamente largos.

En términos formales, la supervisión de acuerdo con (Lunze 2000), se basa en la evaluación global del actual punto de operación del proceso y permite caracterizar la evolución temporal de los parámetros de un sistema, dentro de la perspectiva de proveer diagnóstico y/o de anticipar posibles cambios no deseados en el estado de operación en un determinado proceso a través del ajuste de parámetros escalares del controlador o de cambios en la consigna de operación.

Basado en este hecho, el nivel de supervisión tiene la función de verificar el correcto funcionamiento de la planta, pudiendo incluirse en ello la búsqueda de la calidad de la producción, lo cual es de importancia trascendental para el cumplimiento de los objetivos económicos debido a la gran competencia existente en el mercado actual. Ello lo consigue por medio de las acciones sobre el nivel más bajo, el nivel de control, reconfigurando sus estrategias o inclusive realizando acciones como paradas completas, ante la posibilidad de fallas de grandes mag-

nitudes. El nivel de control interacciona en forma más cercana con los procesos de fabricación, actuando directamente sobre la maquinaria y requiriendo velocidades de actuación muy elevadas ante la presencia de perturbaciones. Por ende, la importancia de los sistemas de supervisión radica, primero en la necesidad de verificar que los procesos de control funcionan de manera adecuada, y segundo, que lo hagan con la precisión que los objetivos de calidad del producto final requieren.

En este capítulo se presenta una propuesta metodológica que permite diseñar sistemas de supervisión para sistemas complejos con el objetivo de detectar cambios en el estado de operación de un proceso específico de tal manera de modificar la consigna de operación y realizar ajustes en línea de los parámetros escalares del controlador de tal manera que se garantice una operación, confiable, coherente y sub-óptima del proceso.

Se propone para ello una arquitectura de supervisión local basada en los principios de la modelación multirresolucional de Sanz (Sanz 1990), combinada con el enfoque de sistemas de eventos discretos. Se incluye dentro de la arquitectura el módulo de coordinación y se presentarán algunas pruebas de concepto sobre la negociación entre sistemas de supervisión de unidades de producción.

En cuanto al contenido de este capítulo, en primer lugar, se describirán muy brevemente algunas funciones tradicionales de los sistemas de supervisión y el enfoque de sistemas holónicos para la fabricación. Luego, se presentará la propuesta metodológica para el diseño de un sistema de supervisión con la respectiva descripción de cada uno de los componentes tales como el detector de eventos, el modelo cualitativo del proceso, el sistema de toma de decisión y el módulo de coordinación. Finalmente, se mostrará el resultado de las simulaciones computacionales y se establecerán las conclusiones.

5.2. Funciones básicas de los sistemas supervisores

En general, la supervisión inteligente de procesos complejos de producción continua involucra la ejecución simultánea de tareas de medición, control y toma de decisiones que permiten mantener desempeños cualitativos y cuantitativos idóneos para garantizar condiciones de seguridad operacional y ambiental (Colina, Altamiranda, and Chacon 2005).

Los sistemas de supervisión son diseñados para cumplir funciones muy especí-

ficas como por ejemplo la detección y el diagnóstico de fallas y la detección de cambios en el punto de operación de un proceso con la finalidad de conmutar una familia de controladores que permitan mantener el proceso en condiciones sub-óptimas.

En general, existen múltiples funciones de los sistemas de supervisión establecida por varios investigadores, a continuación se describen algunas de ellas:

- La coordinación y selección de los puntos de operación de los lazos de control con el fin de mantener las variables en puntos óptimos (Wang, Li, and Zhang 2003) (Eigi, Villani, and Maruyama 2001) (Blanke, Staroswiecki, and Wu 2001).
- La evaluación del desempeño de los controladores locales. Se busca satisfacer criterios de desempeños previamente establecidos, se diagnostican las causas de desviación del desempeño y se ejecutan acciones correctivas (Jantzen 1998) (Yazdi 1997).
- Mejorar la confiabilidad y la seguridad del proceso a ser supervisado (Villa, Duque, Gauthier, and Rakoto 2003) (Vagelatos, Rigatos, and Tzafestas 2001) (Isermann 1998).
- La monitorización de datos del proceso, el análisis de estado actuales, la detección y diagnóstico de anomalías del proceso y ejecuciones de acción de control apropiadas (Uraikul, Chan, and Tontiwachwuthikul 2007)(Waismann 2000).
- El diagnóstico de fallas cuya tendencia ha dado origen al control tolerante a fallos (Montmain and Gentil 1999)(Esteva 2003). Desde este enfoque el objetivo de la supervisión consiste en sistematizar la tolerancia a fallas. En este sentido, la tolerancia a fallas se entiende como la capacidad de un sistema de control para mantener los objetivos de control a pesar de la aparición de una falla, admitiéndose una cierta degradación de sus prestaciones (Izadi and Blanke 1999)(Puig 2004).

En las siguientes secciones se explicarán brevemente tres de las funciones más comunes de los sistemas supervisores: el control supervisorio difuso, la detección y el diagnóstico de fallas y la tolerancia a fallas.

5.2.1. Control supervisorio difuso

El control supervisorio difuso se ha usado tradicionalmente como una estructura de control de dos niveles en donde por lo general el nivel de regulación

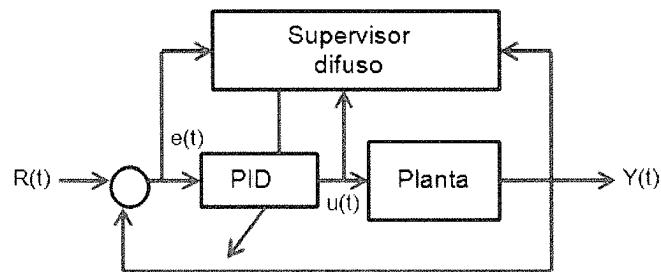


Figura 5.1: Esquema de control supervisorio difuso

contempla un controlador clásico PID y el nivel supervisor está formado por un sistema difuso.

Este esquema de supervisión, que se muestra en la figura 5.1, ha sido aplicado en la industria de procesos, dado que la mayoría de los sistemas de control que se encuentran en operación en las plantas son de tipo convencional que contienen algún controlador PID. Bajo ciertas condiciones de operación del proceso, los ajustes de los parámetros escalares del controlador no son los más adecuados. En tal sentido, el supervisor difuso contiene una serie de reglas heurísticas elaboradas en función de ciertas características dinámicas del proceso, tales como sobreimpulsos, tiempos de asentamiento, error de estado estático, etc.

Algunas reglas clásicas del sistema supervisor difuso son las siguientes (Pasino and Yurkovich 1998):

- Si el error de estado estático es grande entonces incrementar ganancia proporcional.
- Si la respuesta es oscilatoria entonces incrementar la ganancia derivativa.
- Si el error de estado estático es muy grande entonces ajustar la ganancia integral.
- Si el error de sobreimpulso es muy grande entonces decrementar ganancia proporcional.

En definitiva, esta función de supervisión tiene como objetivo sintonizar automáticamente controladores de tipo PID cuando su desempeño de funcionamiento es afectado por cambios en las condiciones de operación del proceso. Este supervisor se considera como un autosintonizador del controlador PID.

5.2.2. Detección y diagnóstico de fallas

Los procesos industriales automatizados pueden presentar fallas. Una falla es considerada, en términos generales, como una desviación no permitida o indeseable de las características propias del proceso como por ejemplo el sensor, actuador, controlador o el sistema. Además los lazos de control, pueden ocultar las fallas hasta alcanzar un grado tal que produzcan una avería irreparable y que obligue a detener el sistema o proceso (Blanke and Kinnaert 2003).

Se entiende por detección de fallas la tarea de determinar, a partir de observaciones realizadas, cuando existe un funcionamiento incorrecto del sistema sujeto a observación, y por diagnóstico de fallas cuáles son las causas de ese comportamiento incorrecto. Es bien sabido que las fallas producidas en los componentes y procesos pueden provocar paradas indeseables y deterioro de los sistemas, con el consiguiente aumento de costos y la disminución de la producción (Abad 2002).

El diagnóstico de fallas puede ser realizado en tres tipos de sistemas distintos cada uno de ellos con sus propias características:

- **Sistemas estáticos:** entendiéndose como tales aquellos cuyas salidas dependen únicamente de las entradas y de algunos parámetros constantes del sistema. No dependen del tiempo en su evolución, como pudiera ser el diagnóstico de componentes electrónicos (Dekleer, Mackworth, and Reiter 1992).
- **Sistemas dinámicos:** entendiéndose como tales aquellos sistemas donde existe una evolución temporal, y el fallo de algún componente del sistema provoca que su evolución temporal sea diferente. En estos sistemas, la salida depende tanto de la variable de entrada como de los parámetros del sistema y el instante en que se encuentre, lo que conforma el estado del sistema. El comportamiento del sistema puede entonces ser descrito en términos de evolución de un estado a otro (Struss 1997).
- **Sistemas híbridos:** son sistemas dinámicos complejos, cuyo comportamiento se modela como un sistema híbrido. Los modelos híbridos presentan dinámicas tanto discretas como continuas y son representadas típicamente como una secuencia de trozos de comportamiento continuo intercalados con transiciones discretas (McILRaith, Biswass, and Clandy 1999).

Cuando se habla de sistemas dinámicos, es necesario tener en cuenta que debido a la dinámica del sistema, los fallos que se pueden detectar son de tres tipos:

- **Fallos abruptos,** entendiéndose como tal el fallo que se produce de forma instantánea en un momento dado, y a partir de ese instante permanece presente en el sistema en la misma magnitud.

- Fallas incipientes, son fallas que, normalmente debido al desgaste de algún componente del sistema, comienza como una falla de magnitud pequeña y va evolucionando en el tiempo a una magnitud mayor, hasta alcanzar un máximo.
- Fallas intermitentes, son las fallas que se producen en el sistema de manera intermitente, remitiendo tras un tiempo y volviendo a producirse pasado un período de tiempo. Son las más difíciles de detectar y constituye uno de los principales problemas abiertos en el campo del diagnóstico de sistemas dinámicos.

Esta distinción no ocurre en los sistemas estáticos debido a que en ellos no existe dependencia del tiempo, con lo cual la falla está presente o no presente en el momento de la detección (Abad 2002).

5.2.3. Tolerancia a fallas

El avance tecnológico ha permitido desarrollar sistemas industriales muy complejos con alto grado de automatización, mejorando la calidad de sus productos e incrementado la eficiencia de sus procesos pero consecuentemente ha hecho más vulnerable a los sistemas a las fallas.

Actualmente, existe una creciente necesidad e interés en desarrollar sistemas de control que puedan operar de forma aceptable, incluso después de la aparición de una falla y que sean capaces de modificar, adecuar o parar el proceso antes de que se originen daños irreparables en el mismo. A este tipo de sistemas de control se les denomina tolerantes a fallas y contienen mecanismos que hacen que los sistemas de control sean más confiables (Rodríguez 2009).

Existen dos formas de abordar el diseño de un sistema tolerante a fallas: el enfoque pasivo y el activo (Rodríguez 2009). El enfoque pasivo se basa en el diseño de un sistema de control que sea capaz de alcanzar sus objetivos tanto en condiciones normales como en presencia de falla. Por otro lado, el enfoque activo se basa en el diseño de una ley de control en función del estado del sistema en condiciones normales o bajo fallas a partir de los parámetros y restricciones del sistema proporcionados por un ente diagnosticador. Es decir, el control tolerante pasivo se enfoca más por el control robusto en donde los objetivos de control se cumplen siempre y la magnitud de ciertos parámetros inciertos o disturbios estén dentro de las condiciones de operación tolerantes y el control tolerante activo consiste en determinar el componente averiado, el tipo de avería, tamaño e instante de aparición, y a partir de dicha información, activar algún mecanismo tolerante, como la acomodación de los parámetros del sistema o la reconfiguración de las

5.3. ENFOQUE DE SISTEMAS HOLÓNICOS PARA FABRICACIÓN

entradas/salidas de la planta o incluso, dependiendo de la gravedad, el bloqueo del sistema.

La acomodación a la falla consiste en resolver el problema manteniendo la estructura del controlador y modificando solamente los parámetros. Por otro lado, la reconfiguración consiste en cambiar las entradas y salidas del controlador así como reajustar la ley de control. Estos enfoques exigen disponer de un sistema de localización y detección de fallas que en tiempo real, pueda dar información a un sistema supervisor para que éste active algún mecanismo de acción correctora.

Con respecto al sistema tolerante a fallas en los sensores y actuadores, existen dos mecanismos para tolerar una falla, la redundancia material y la redundancia analítica (Gerter 1998). El mecanismo de redundancia analítica; consiste, para el caso de sensores, en la incorporación de un bloque que reconstruya la medición mediante la estimación de las mismas. Para el caso de los actuadores, consiste en el reajuste de señales alternativas para llevar a cabo la acción de control requerida, así se evita la incorporación de nuevo hardware en el sistema lo que se ve reflejado en los costos de instrumentación (Efrain 2001).

El mecanismo de redundancia material consiste, para el caso de los sensores, en contar con un número generalmente impar de éstos cuya salida se multiplexa dentro de un bloque de decisión, en donde se determina la respuesta correcta a partir de las salidas más comunes de estos mismos, a este proceso se le conoce simplemente como redundancia estática; para el caso de los actuadores, es más directo, pues en el caso de ocurrir una falla, éste se reemplaza totalmente, a este proceso se le conoce redundancia dinámica.

5.3. Enfoque de sistemas holónicos para fabricación

Los sistemas holónicos son un enfoque que ha sido propuesto para resolver los nuevos requisitos de la fabricación y fue definido inicialmente utilizando como elemento base el concepto de holón de (Koestler 1989). Después se desarrolló en el contexto de la fabricación por Suda (Suda 1990) y Christensen (Christensen 1994) con el fin de proveer una base para desarrollar y operar un sistema de fabricación. El más reciente propósito de este enfoque es el desarrollo de una arquitectura para sistemas de fabricación altamente descentralizados, contruidos de una mezcla de módulos de componentes estandarizados, autónomos, cooperativos e inteligentes y así implementar entornos rápidamente cambiantes (Christensen 2002).

CAPÍTULO 5. PROPUESTA METODOLÓGICA DE UN SISTEMA DE SUPERVISIÓN

La naturaleza de los sistemas holónicos de fabricación (HMS) impone requisitos nuevos y únicos en la arquitectura del sistema de producción, definida como la estructura y relaciones entre las unidades funcionales inteligentes en un sistema, posiblemente incluyendo las interfaces del sistema con su entorno (Christensen 2002). Los sistemas holónicos de fabricación están conformados por agentes u holones autónomos, inteligentes, flexibles, distribuidos y cooperativos. En un HMS cada actividad de un holón es determinada por medio de la cooperación con otros holones, en oposición a ser determinada por mecanismos centralizados.

Una de las características más prometedoras del enfoque holónico es que éste representa la transición entre los sistemas jerárquicos y heterárquicos (Bongaerts, Monostori, McFarlane, and Kadar 2002). De esta forma, el sistema se conduce hacia el paradigma de la “fabricación ágil”, donde los componentes de sistemas de producción son reprogramados y reconfigurados constantemente y se encuentran al mismo tiempo dentro de un sistema de fabricación con un volumen denso de información.

En los sistemas holónicos cada uno de los holones debe tener asociado un conjunto de características que permitan describirlo internamente y externamente, le permitan comunicarse con otros holones y ejecutar sus tareas apropiadamente. Existen básicamente tres tipos de holones:

- Holón producto: mantiene el conocimiento del proceso para asegurar la correcta fabricación del producto. Asimismo, almacena información sobre configuración, diseño, plan de proceso, lista de materiales, procedimientos de aseguramiento de calidad, etc., sobre el ciclo de vida del producto.
- Holón recurso: contiene los métodos para asignar los recursos de producción, así como el conocimiento y los procedimientos para organizar, utilizar y controlar los recursos durante la producción. Asimismo, ofrece su capacidad de producción y funcionalidad a otros holones del entorno. Un holón de recurso es una abstracción que puede representar una fábrica, una planta, máquinas, cintas, componentes, materia prima, herramientas, áreas de almacenamiento de material, personal, energía, etc.
- Holón orden: representa una tarea en el sistema de fabricación. Es responsable de realizar correctamente el trabajo asignado y a tiempo. El holón orden manipula el producto físico que será producido, el modelo de estado del producto, y todo el procesamiento de la información logística asociada al trabajo.

5.4. Propuesta metodológica para el diseño de un sistema de supervisión

Antes de describir la propuesta metodológica, es importante aclarar el tipo de proceso que se desea supervisar. En general, existen tres tipos de procesos de producción en manufactura industrial, de acuerdo a la clasificación de Fisher (Fischer 1990): proceso de manufactura, proceso por lotes y sistema de producción continuo. En particular, la presente propuesta va dirigida a sistemas de producción continua. La característica fundamental de estos sistemas es que su funcionamiento no puede detenerse, salvo aquellas situaciones donde se presenten fallas en los equipos o exista un mantenimiento programado de los recursos físicos.

Específicamente el funcionamiento de este tipo de proceso es de la siguiente forma: dado un flujo continuo de material se aplican procesos de transformación según un método determinado y luego el producto se envía inmediatamente al cliente usando una red de distribución o pudiera almacenarse en depósitos para distribuirse posteriormente (Parra 2009).

Una vez aclarado el tipo de proceso que se desea supervisar, la funcionalidad del sistema supervisor queda definida de la siguiente forma: *detectar cambios en el estado de operación de un proceso continuo producto de la variación de la cantidad o calidad de producción con la finalidad de realizar cambios en la consigna de operación y ajustar los parámetros escalares del controlador del proceso que permita garantizar una operación sub-óptima y coherente en el respectivo estado de operación.*

Para ello es necesario definir una arquitectura de supervisión local que se muestra en la figura 5.2, la cual comprende los siguientes elementos:

- Un controlador difuso unitario diseñado en base a la metodología CMI y que fue objeto de estudio en los capítulos 3 y 4. Dicho controlador se implementó en dos procesos continuos: separación gas-líquido y calentamiento de crudo.
- El módulo de medición que consolida la todas las variables de la planta o unidad de producción que se desea supervisar.
- El módulo detector de eventos: el evento se define como un suceso instantáneo que puede originar una transición entre estados. El detector de eventos trabaja en conjunto con el modelo cualitativo del proceso. Esto quiere decir que es necesario conocer el estado actual del proceso y determinar si un determinado evento genera o no una transición entre estados.

CAPÍTULO 5. PROPUESTA METODOLÓGICA DE UN SISTEMA DE SUPERVISIÓN

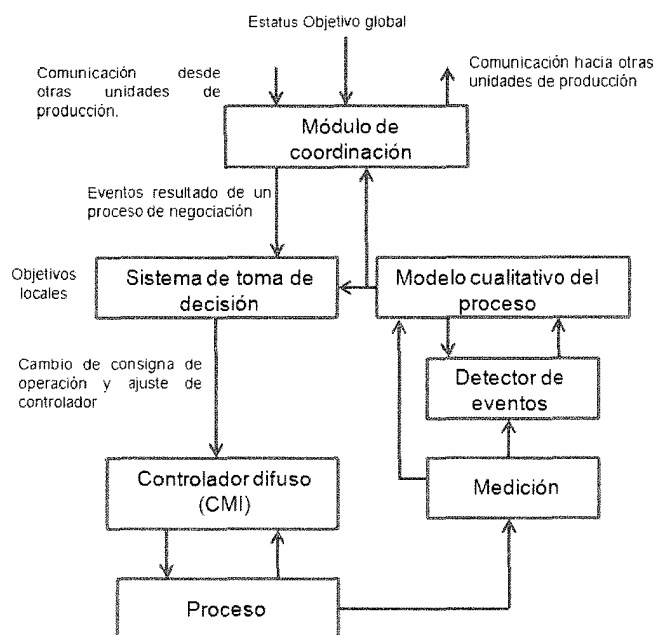


Figura 5.2: Arquitectura del sistema supervisor para procesos continuos

- El modelo cualitativo del proceso trata sobre un autómata de estados finitos que contiene los estados discretos de operación del proceso bajo condiciones normales de operación.
- El módulo de toma de decisión se encarga de cambiar la consigna de operación dependiendo del nuevo estado de operación y de ajustar el controlador del proceso para minimizar en caso de que sea necesario el error de estado estático.
- El módulo de coordinación se encarga de establecer la comunicación y coordinación con otros agentes de supervisión de otras unidades de producción y con el sistema supervisor global de todo el proceso quien se encarga de vigilar el cumplimiento del objetivo global del proceso.

En la arquitectura que se propone se han combinado los enfoques de modelación multirresolucional de Sanz y las ideas básicas de los sistemas de eventos discretos. Esta arquitectura está inspirada en los mismos enfoques usados en la arquitectura propuesta en (Parra 2009). La justificación de esta propuesta obedece al hecho de que las dinámicas en el nivel de supervisión son de naturaleza discreta no importa si el proceso es continuo. En este sentido, la dinámica de un proceso continuo puede representarse por medio de un sistema de eventos discretos (SED)

5.4. PROPUESTA METODOLÓGICA PARA EL DISEÑO DE UN SISTEMA DE SUPERVISIÓN

donde cada estado corresponde a un modo de operación mientras que los eventos se consideran como el cambio instantáneo de un estado a otro. Cada uno de los elementos que conforman la arquitectura de supervisión local será explicado a continuación.

5.4.1. Módulo de medición

El módulo de medición simplemente muestrea todas las señales asociada a la instrumentación de la unidad de producción que se desea supervisar. En muchos procesos industriales, además de la instalación de los sensores asociado al sistema de control en lazo cerrado, otros sensores se instalan para permitir el monitoreo de información de interés sobre el estado del proceso. Por ejemplo, en un separador de producción se controla el nivel de crudo instantáneo y además se monitorea la presión y temperatura de crudo en el separador.

5.4.2. Detector de eventos

Una condición fundamental para implementar la funcionalidad del sistema supervisor que se propone consiste en identificar correctamente el evento que implique un cambio de estado. En general, la detección de eventos se implementa principalmente sobre variables continuas, aunque, esto no excluye la detección de eventos en variables discretas, como lo es por ejemplo, el encendido de un motor.

Cuando se trata de variables continuas observables directamente es necesario aplicar alguna regla que permita deducir la ocurrencia de un evento análogo a la proyección de un valor continuo hacia un estado discreto, como en los sistemas híbridos.

Un caso sencillo ocurre cuando el valor de una variable continua está dentro de un intervalo que corresponde a un estado discreto y el cambio de un intervalo a otro se asocia a la ocurrencia de un evento. La figura 5.3 muestra un ejemplo sencillo para ubicar una variable continua (presión) en tres posibles valores discretos: baja, media y alta. En este caso se debe tener cuidado con la medición realizada por el dispositivo sensor ya que éste está sujeto a variaciones debido a la naturaleza fluctuante del proceso o por las características intrínsecas del sensor.

Estas fluctuaciones pueden originar una detección errónea de eventos o dejar pasar el evento sin detectarlo. Otra estrategia para detectar eventos consiste en examinar tendencias sobre un conjunto de datos. Un método de agrupación de-

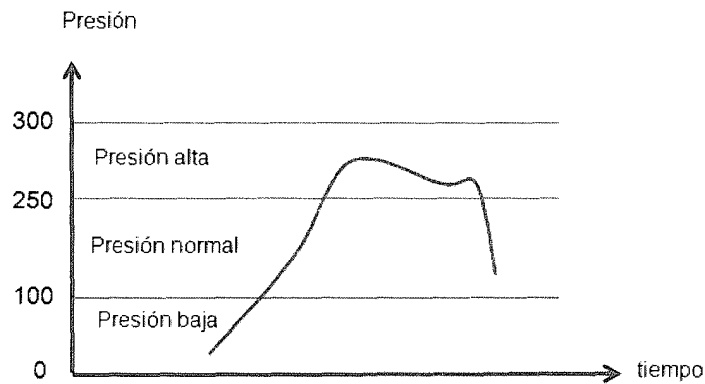


Figura 5.3: Ejemplo de mapeo directo de variable continua a discreta

nominado ventana deslizante fue diseñado por Sarrate y Aguilar, en su trabajo sobre vigilancia y control inteligente de procesos (Sarrate and Aguilar 2003).

El método de detección de eventos que se propone está concebido para detectar eventos que causan transición de estados y eventos que no causan dicha transición tal y como lo muestra la figura 5.4. Para ello se propone seguir los tres siguientes pasos:

- Se necesita determinar el estado actual del proceso en términos bien sea de la cantidad de producción y de la calidad de producción que maneja actualmente la unidad de producción. Ello implica la necesidad de medir directa o indirectamente las variables $q(\tau)$ que es la variable que indica el volumen o la tasa de producción y $c(\tau)$ que es la variable que indica como se está comportando el producto que procesa la unidad de producción en términos de su calidad.
- Luego se toma una serie de muestras de la variable controlada $y(\tau)$, tal variable se asume que es afectada por los cambios tanto de $q(\tau)$ como de $c(\tau)$ en un intervalo de tiempo T y se calcula la pendiente promedio m_m . Dicho parámetro indica cual es la magnitud de la variación de $y(\tau)$.
- Finalmente, se evalúa nuevamente el estado de operación del proceso y se verifica si se mantiene el mismo estado.

Los tres pasos anteriores se pueden sintetizar a través de las tres siguientes reglas básicas:

5.4. PROPUESTA METODOLÓGICA PARA EL DISEÑO DE UN SISTEMA DE SUPERVISIÓN DE SUPERVISIÓN

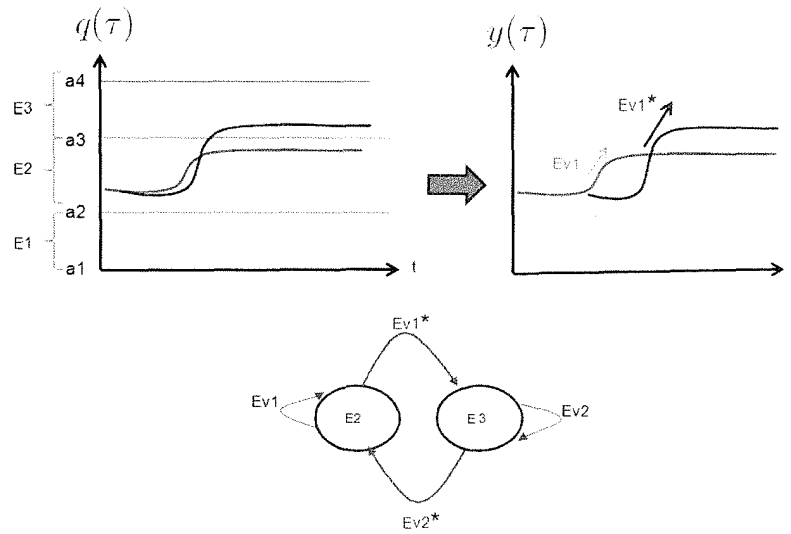


Figura 5.4: Detección de eventos

- Si $e_{anterior}$ es E_2 y $m_m \neq 0$ y e_{actual} es E_2 Entonces ocurre Ev_1
- Si $e_{anterior}$ es E_2 y $m_m \neq 0$ y e_{actual} es E_3 Entonces ocurre Ev_1^*
- Si $e_{anterior}$ es E_2 y m_m es 0 y e_{actual} es E_3 Entonces no ocurre ningún evento Ev_0

En donde tanto E_2 como E_3 son estados de operación discretos distintos, $e_{anterior}$ es el estado discreto antes del cálculo de la pendiente promedio y e_{actual} es el estado actual luego de que se efectúe el cálculo de m_m . Por otro lado, Ev_1 es un evento que no causa transición de estado, simplemente se trata de un cambio repentino de la variable $y(\tau)$ producto de un cambio de $q(\tau)$ tal y como se muestra en la figura 5.4. En cambio, Ev_1^* , es un evento que genera una transición del estado E_2 al estado E_3 tal y como se aprecia en la figura 5.4.

Evidentemente, dependiendo del signo de m_m , el evento Ev_1^* estaría relacionado con un aumento o disminución de producción dependiendo del tipo de proceso.

5.4.3. Modelo cualitativo del proceso

El modelo cualitativo del proceso que se propone es representado a través de una máquina de estados finitos. Una máquina G de estados finitos es una quintupla $G = (X, \Sigma, f_t, x_1, X_m)$ en donde:

- X es un conjunto finito de estados denotados por x
- Σ , alfabeto de entrada, es un conjunto de eventos de entrada, denotado cada uno con una letra de un alfabeto.
- $f_t : X \times \Sigma \mapsto 2^X$ es la función de transición o evolución del autómata.
- $x_1 \in X$ es el estado inicial del autómata.
- $X_m \in X$ es el conjunto de estados deseados de G .

Es importante aclarar que el alfabeto de entrada Σ incluye los eventos que generan transición de estados y los eventos que no generan transición estados tal y como se explicó en la sección anterior. Para definir los estados discretos del proceso se seleccionan dos variables asociadas al mismo: la variable que determine la cantidad de producción que maneja la unidad de producción $q(\tau)$ y una variable que tenga influencia en la calidad del producto $c(\tau)$ que procesa la unidad de producción que está siendo supervisada.

Luego se definen n_1 rangos de valores para la variable relacionada con la cantidad de producción $q(\tau)$, cada rango con su respectiva etiqueta cualitativa y n_2 rango de valores para la variable relacionada con la calidad del producto por lo que el número de estados discretos es de $n_1 \times n_2$. De tal manera, que cada estado discreto de operación está asociado a una descripción cualitativa que indica de alguna manera tanto el volumen de producción como su calidad. En la figura 5.4a, hay un ejemplo muy específico en donde se definen tres rangos para la variable $q(\tau)$ y dos rangos para la $c(\tau)$ lo que daría un total de 6 estados discretos x_1, x_2, \dots, x_6 .

Estos estados discretos conforman una máquina de estados finitos y es importante definir el estado inicial x_1 . La máquina evolucionará por el resto de los estados dependiendo de la activación de los eventos que generan transición de estados cuya causa proviene de una variación significativa tanto de la cantidad como de la calidad de producción del producto que entra a la unidad de producción. Es por ello que tanto el detector de eventos como el modelo cualitativo basado en máquina de estados finitos dependen mutuamente entre sí.

5.4. PROPUESTA METODOLÓGICA PARA EL DISEÑO DE UN SISTEMA DE SUPERVISIÓN

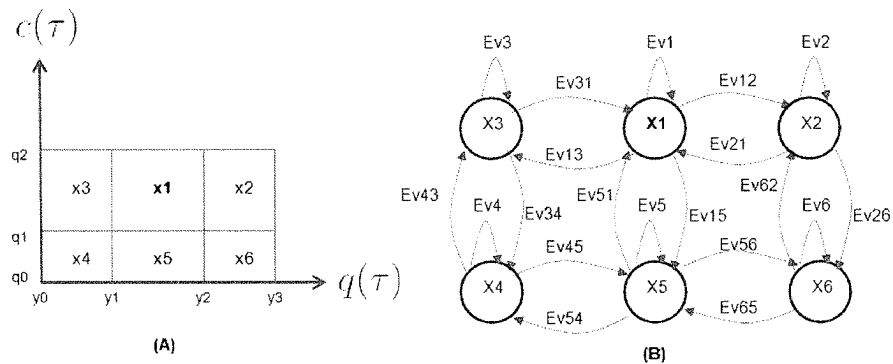


Figura 5.5: Modelo cualitativo general de un proceso

Partiendo del estado inicial x_1 como se observa en la figura 5.5b, si ocurre el evento Ev_{12} asociado a un aumento o disminución de la cantidad de producción entonces el nuevo estado del proceso sería x_2 . Una vez que el proceso está en el estado x_2 si ocurre el evento Ev_{21} , el proceso vuelve nuevamente al estado x_1 . Luego si ocurre el evento Ev_{13} entonces el proceso cambia al estado x_3 . Este evento Ev_{13} es similar al evento Ev_{21} , por lo que la máquina presenta varios eventos que son similares.

Cabe destacar que la máquina de estado finito que se muestra en la figura 5.5b es de tipo determinista y es válida si y sólo si las señales que representan las variables tanto de cantidad como de calidad de la producción están en condiciones normales, es decir, cuando no exista presencia de fallas tales como señales fuera del rango de operación o algún cambio abrupto en $q(\tau)$ o $c(\tau)$, por ejemplo, un cambio en $q(\tau)$ que implique un cambio del estado de operación x_2 al x_3 .

5.4.4. Sistema de toma de decisión

El sistema de toma de decisión realiza dos tareas muy importantes en función a la variación del estado de operación del proceso. La primera tarea está relacionada con el cambio de la consigna de operación o setpoint. Esto quiere decir que por cada estado de operación x_i existe un setpoint Sp_i en la cual se supone que la unidad de producción realiza el proceso de transformación del producto de la forma más idónea posible.

La segunda tarea trata sobre la minimización del error de estado estático a través de la adaptación de los parámetros escalares del controlador. En la figura

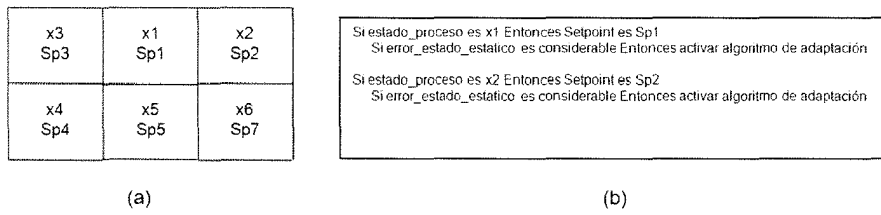


Figura 5.6: Sistema de toma de decisión

5.6 se muestran las dos tareas que realiza el sistema de toma de decisión.

Es importante definir un límite máximo y mínimo del setpoint en cada estado de operación del proceso en virtud de que eventualmente cuando ocurra un proceso de negociación del sistema supervisor local con otros sistemas supervisores, ya estén establecidas las restricciones sobre el incremento o decremento de la consigna de operación de la respectiva unidad de producción. En la próxima sección se explicará muy brevemente el funcionamiento de la unidad de coordinación en la cual se lleva a cabo un proceso de negociación.

La segunda tarea tiene que ver con la adaptación en línea del controlador difuso del proceso para minimizar el error de estado estático y mantener un desempeño idóneo del contrador independientemente del estado de operación en que se encuentra trabajando.

En relación al tema de adaptación en línea de controladores, existen dos políticas de control comunmente usadas (Abonyi, Andersen, Nagy, and Szeifert 1999):

- Control adaptativo directo: en el que los parámetros del controlador son ajustados utilizando directamente la información aportada por las variables de entrada y salida de la planta.
- Control adaptativo indirecto: en el que se realiza una estimación en tiempo real del modelo de la planta y, basándose en este modelo, los parámetros del controlador son optimizados.

Se requiere entonces adaptar los parámetros del controlador según el comportamiento actual de la planta. En cualquier caso, debe existir un nivel de supervisión encargado de ejecutar esta labor. En el caso particular de controladores difuso, existen dos alternativas para su adaptación: reglas difusas (controlador auto-organizado) y factores de escala de las funciones de pertenencias (controlador con auto-aprendizaje) (Abonji 2007).

5.4. PROPUESTA METODOLÓGICA PARA EL DISEÑO DE UN SISTEMA DE SUPERVISIÓN

Un controlador difuso auto-organizado o SOC (Self-Organizing Controller) es un sistema capaz de modificar automáticamente, y sin intervención humana, su base de reglas. A partir de las discrepancias con respecto a unos determinados criterios prefijados que calcula el nivel de supervisión, el bloque de ajuste del control auto-organizado establece las oportunas modificaciones en la base de reglas. Los SOC están mostrando especial interés en multitud de aplicaciones, por ejemplo, en (Mandic and Scharf 1995) son usados para control robots.

En forma general, el bloque de ajuste de un SOC, implementa una tabla de decisión que en función de los criterios de evaluación, devuelve una indicación sobre la acción requerida. Frecuentemente, este mecanismo de ajuste no modifica las reglas originales, sino que incluye nuevas reglas y por tal motivo se puede presentar una serie de problemas relacionados con la convergencia del algoritmo (Zeungnan and Jongcheol 1991).

Los controladores difusos con auto-aprendizaje son aquellos que son capaces de autoajustar los factores de escala del controlador y en general proporciona una mejor sintonización de los dominios de entrada y / o salida de la función difusa ya que modifican los soportes o núcleos de los términos lingüísticos en igual proporción para cada variable. La alteración del factor de escala de la salida proporciona una modificación en el rango del actuador a la planta. El ajuste de los factores de escala ha sido ampliamente estudiado en la bibliografía tanto para ajustar las variables de entrada como las de salida.

El mecanismo de adaptación en su mayor parte se corresponde a procedimientos y técnicas de aprendizaje del tipo descenso por gradiente, similar por ejemplo a lo que se plantea en redes neuronales y modelación neurodifusa.

La propuesta de adaptación de controladores difusos que se propone es de tipo autoaprendizaje. Específicamente, se va a variar los núcleos de las funciones de membresías de la variable manipulada $u(\tau)$. Si recordamos del capítulo 4, tenemos que la base de reglas del controlador diseñado es de la forma:

$$\text{Si } r(\tau + 1) \text{ es } C_j \text{ Entonces } u(\tau) \text{ es } U_j, j = 1, 2, \dots, M \quad (5.1)$$

En donde C_j son los términos lingüísticos de $r(\tau + 1)$ y que depende de $y(\tau)$ y de los consecuentes unitarios c_{ij} a través de la siguiente expresión:

$$C_j(\tau) = \sum_{i=1}^N \mu_{A_i}(y(\tau))c_{ij}, j = 1, \dots, M. \quad (5.2)$$

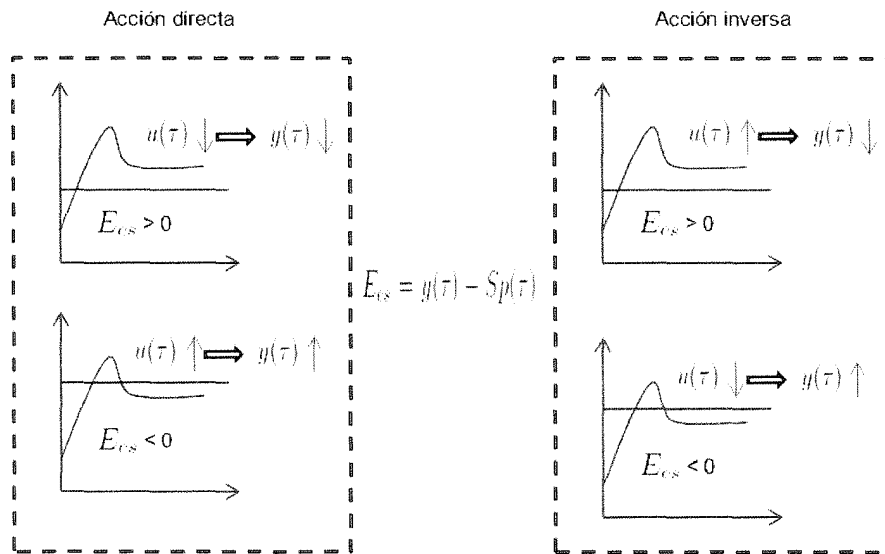


Figura 5.7: Estrategias para mejorar el error de estado estático

Y por último la salida del controlador es:

$$u(\tau) = \sum_{j=1}^M \mu_{C_j}(r(\tau + 1)) \cdot \text{Nucleo}(U_j) \quad (5.3)$$

Se observa claramente en esta ecuación que si $\text{Nucleo}(U_j)$ disminuye entonces $u(\tau)$ disminuye y viceversa. De manera análoga, sucede lo mismo con $\mu_{C_j}(r(\tau + 1))$ que depende de $y(\tau)$.

Se procede entonces a definir el error de estado estático. Dicho error, como se observa en la figura, es $E_{es} = y(\tau) - Sp(\tau)$. Luego, se define si el tipo de acción del lazo de control es directo o inverso. Si se incrementa $u(\tau) \uparrow$ y aumenta la variable controlada $y(\tau) \uparrow$ entonces la acción es directa. De lo contrario, si disminuye $u(\tau) \downarrow$ y aumenta $y(\tau) \uparrow$ entonces la acción es inversa.

Luego se define un ϵ tal que si $|E_{es}| \geq \epsilon$, entonces se activa el algoritmo de adaptación. El algoritmo de adaptación modifica iterativamente los núcleos U_j de las funciones de membresía de $u(\tau)$ de tal manera que $|E_{es}| < \epsilon$. Es importante aclarar que en cualquier instante de tiempo τ , la máxima cantidad de términos lingüísticos que afectan al valor instantáneo de $u(\tau)$ son dos, excepto en el caso en que el valor de $u(\tau)$ coincida exactamente con el núcleo de una de las funciones de membresía.

5.4. PROPUESTA METODOLÓGICA PARA EL DISEÑO DE UN SISTEMA DE SUPERVISIÓN

El algoritmo de adaptación, por ende, disminuye o aumenta en un valor Δu_j (según sea el caso) los núcleos de las funciones de membresía de los términos lingüísticos del valor instantáneo de $u(\tau)$ hasta que $|E_{es}| < \epsilon$. En el caso de que la acción de lazo de control sea directa tendríamos dos casos:

- Si el error de estado estático es positivo entonces es necesario disminuir iterativamente los núcleos de $u(\tau)$ de tal manera que $y(\tau)$ pueda disminuir su valor y se minimice el error de estado estático E_{es} , tal y como se aprecia en la figura 5.7.
- Si el error de estado estático es negativo entonces es necesario aumentar iterativamente los núcleos de $u(\tau)$ de tal manera que $y(\tau)$ pueda aumentar su valor y se minimice el error de estado estático E_{es} , tal y como se aprecia en la figura 5.7.

En el caso de que la acción de lazo de control sea inversa, tomando en cuenta la figura 5.7, tendríamos dos casos:

- Si el error de estado estático es positivo entonces es necesario aumentar iterativamente los núcleos de $u(\tau)$ de tal manera que $y(\tau)$ pueda disminuir su valor y se minimice el error de estado estático E_{es} .
- Si el error de estado estático es negativo entonces es necesario disminuir iterativamente los núcleos de $u(\tau)$ de tal manera que $y(\tau)$ pueda aumentar su valor y se minimice el error de estado estático E_{es} .

De tal manera que la salida $u(\tau)$ de la ecuación (5.3) se expresaría de la siguiente forma:

$$u(\tau) = \mu_{C_1}(r(\tau+1)).Nucleo(U_1 \pm \Delta u_1 * k) + \dots + \mu_{C_M}(r(\tau+1)).Nucleo(U_M \pm \Delta u_M * k) \quad (5.4)$$

En donde $\Delta u_1, \dots, \Delta u_M$ es la tasa de ajuste de cada núcleo de $u(\tau)$ y $k = 1, 2, \dots, n$ es el número de iteración que se requiere para minimizar el error.

5.4.5. Módulo de coordinación

El módulo de coordinación se encarga de establecer la comunicación y coordinación con otros sistemas de supervisión local asociados a otras unidades de producción y con el sistema de supervisión principal que en última instancia se encargará de garantizar el cumplimiento del objetivo global de todo el proceso. Se asume que tal módulo de coordinación es un agente que tiene la capacidad de

negociar, por ejemplo, la actual consigna de operación del proceso con el agente supervisor y eventualmente con otras unidades de producción.

El sistema supervisor local a su vez se concibe como un agente deliberativo ya que presenta las características asociadas a la autonomía, reactividad y proactividad. De tal manera que cada unidad de producción está conformado por dos agentes: el agente supervisor local y el agente de coordinación.

El objetivo global del proceso se asume que viene expresado en términos del cumplimiento de una cuota de producción con una cierta desviación. Cuando dicha cuota de producción está por debajo de este rango, el agente supervisor global debería solicitar negociar el cambio de la consigna de operación de los agentes de las unidades de producción que de acuerdo a su estado de operación puedan cooperar. En el apéndice C, se detalla el proceso de negociación, previo a una revisión de los aspectos teóricos de este tema y hay unos resultados preliminares producto de una prueba concepto implementada en un PLC (Controlador Lógico Programable) cuyos detalles se muestran en el apéndice D.

5.5. Validación de la propuesta

El proceso de separación gas líquido se usará para validar la metodología de supervisión propuesta en la sección anterior. La simulación del sistema supervisor local fue implementada en MATLAB / Simulink a través del empleo de los bloques de funciones S (S-function). La generación de eventos y los cambio de estado de operación en el proceso se programan previamente antes de realizar la corrida de la simulación.

El primer paso a seguir es definir las variables que determinan el volumen de producción que entra a la unidad de producción y una variable que de alguna manera sea un indicativo de la calidad de la producción. En el caso particular de un separador de producción, el flujo de crudo de entrada $q_i(t)$ proveniente de los pozos es una variable que determina de manera trivial el estado de la producción. Es importante aclarar que en el modelo matemático que se obtuvo para el separador en el capítulo 4, la variable $q_i(t)$ se asumió constante. En esta simulación se asume que tal variable puede cambiar para poder introducir cambios de estados de operación.

En cuanto a la variable que determina la calidad de la producción, idealmente sería conveniente tener disponible la viscosidad del crudo. Vale la pena destacar,

5.5. VALIDACIÓN DE LA PROPUESTA

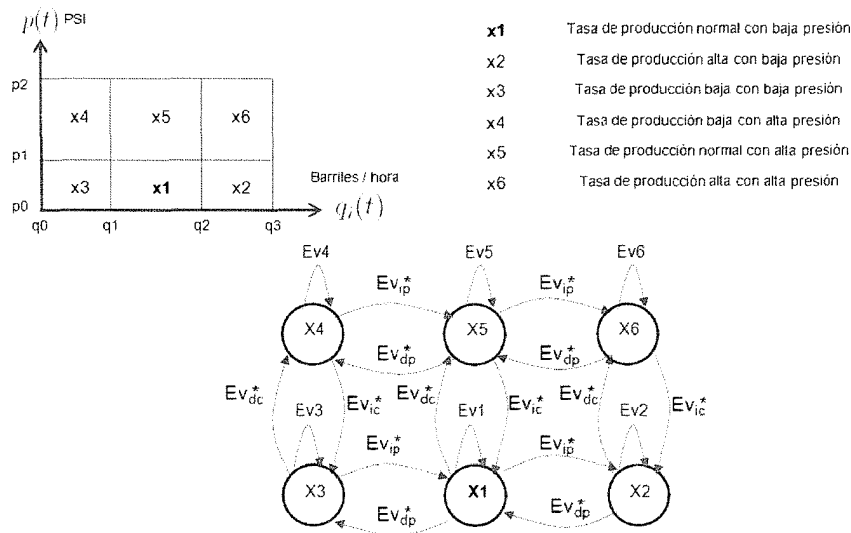


Figura 5.8: Modelo cualitativo del proceso

que la calidad del crudo que se maneja actualmente está relacionada con el transporte del crudo. Por ende, un crudo de menor viscosidad puede ser transportado mucho más fácilmente que un crudo más viscoso.

Se sabe que la viscosidad del crudo es inversamente proporcional a la temperatura del fluido del crudo y a su vez afecta la caída de presión en la tubería. Los pozos de petróleo de tipo pesado y extrapesado que tienen una gravedad API menor a 12 grados, presentan una viscosidad a una temperatura de operación de 110 grados Fahrenheit, que oscila entre 800 a 2000 centipoise dependiendo del tipo de yacimiento. En estos pozos se inyecta un flujo de diluyente (crudo de 28 grados API) para facilitar el transporte del crudo y cuando por alguna razón no se puede inyectar la cantidad adecuada de diluyente o la calidad del diluyente no es la más apropiada, entonces esto se refleja en un aumento en la presión del cabezal del pozo que a su vez se refleja en la presión del separador.

Ahora bien, dado que la viscosidad no está disponible en el modelo matemático, es necesario buscar otra variable cuya variación afecte a la viscosidad. La presión del crudo a la entrada del separador $p(t)$, que es la misma presión del separador y a su vez la misma presión a la entrada de la válvula de control, es una variable que de alguna forma refleja los cambios de viscosidad del crudo. Por ende, se selecciona esta variable para determinar los estados de operación relacionados a la calidad del crudo.

CAPÍTULO 5. PROPUESTA METODOLÓGICA DE UN SISTEMA DE SUPERVISIÓN

El segundo paso consiste en asignar una descripción cualitativa a cada estado discreto de operación y los respectivos rango de operación de estos estados. Para el caso de la variable $q_i(t)$ se designarán los siguientes estados asociados a la cantidad de producción: baja, normal y alta. El estado de operación “baja producción” va desde $q_0 = 10B/H$ hasta $q_1 = 27B/H$, el estado de operación “producción normal” va desde $q_1 > 27B/H$ hasta $q_2 = 40B/H$ y el estado de operación “producción alta” va desde $q_2 > 40B/H$ hasta $q_3 = 50B/H$.

En cuanto a la variable de producción presión, para efectos de simplicidad se designan dos estados de operación que reflejan el estado de la calidad de la producción: normal y alta. Los valores de p_0 , p_1 y p_2 son 70, 110 y 120 PSI. En total se han definido seis estados de operación que indican el rango de operación en cuanto a cantidad y calidad tal y como se muestra en la figura 5.8. El estado inicial x_1 se ha definido como tasa de producción normal con presión de operación normal. El resto de los estados de operación se muestran en la figura 5.8.

El tercer paso consiste en diseñar la máquina de estados finitos. Es necesario para ello identificar los eventos que generan transición de estados y aquellos que no. Los estados ya fueron definidos en el paso anterior. En cuanto a los eventos, se puede observar, en base a la figura 5.8, que hay cuatro eventos que generan transición: Ev_{ip}^* (incremento de la tasa de producción), Ev_{dp}^* (decremento de la tasa de producción), Ev_{ic}^* (incremento de la calidad de producción) y Ev_{dc}^* (decremento de la calidad de producción). Los eventos Ev_1 , Ev_2 , Ev_3 , Ev_4 , Ev_5 son en realidad similares y están relacionados con aumento o disminución de la cantidad o calidad de producción pero no generan cambio de estado.

El cuarto paso es establecer las reglas para determinar la ocurrencia de los eventos. En base a los pasos 2 y 3, y tomando en cuenta que m_m es la pendiente promedio de $y(\tau)$ en un intervalo de tiempo de $t = 3seg$, se define la siguiente base de reglas:

- Si $e_{anterior}$ es x_1 y $m_m \neq 0$ y e_{actual} es x_1 Entonces ocurre Ev_1
- Si $e_{anterior}$ es x_1 y $m_m > 0$ y e_{actual} es x_2 Entonces ocurre Ev_{ip}^* .
- Si $e_{anterior}$ es x_1 y $m_m < 0$ y e_{actual} es x_3 Entonces ocurre Ev_{dp}^* .
- Si $e_{anterior}$ es x_2 y $m_m \neq 0$ y e_{actual} es x_2 Entonces ocurre Ev_2 .
- Si $e_{anterior}$ es x_2 y $m_m < 0$ y e_{actual} es x_1 Entonces ocurre Ev_{dp}^* .
- Si $e_{anterior}$ es x_2 y $m_m < 0$ y e_{actual} es x_6 Entonces ocurre Ev_{dc}^* .

- Si $e_{anterior}$ es x_3 y $m_m > 0$ y e_{actual} es x_1 Entonces ocurre Ev_{ip}^* .
- Si $e_{anterior}$ es x_3 y $m_m < 0$ y e_{actual} es x_4 Entonces ocurre Ev_{dc}^* .
- Si $e_{anterior}$ es x_1 y $m_m < 0$ y e_{actual} es x_5 Entonces ocurre Ev_{dc}^* .
- Si $e_{anterior}$ es x_5 y $m_m \neq 0$ y e_{actual} es x_5 Entonces ocurre Ev_5 .
- Si $e_{anterior}$ es x_5 y $m_m > 0$ y e_{actual} es x_6 Entonces ocurre Ev_{ip}^* .
- Si $e_{anterior}$ es x_5 y $m_m < 0$ y e_{actual} es x_4 Entonces ocurre Ev_{dp}^* .
- Si $e_{anterior}$ es x_5 y $m_m > 0$ y e_{actual} es x_1 Entonces ocurre Ev_{dc}^* .
- Si $e_{anterior}$ es x_4 y $m_m > 0$ y e_{actual} es x_5 Entonces ocurre Ev_{ip}^* .
- Si $e_{anterior}$ es x_4 y $m_m > 0$ y e_{actual} es x_3 Entonces ocurre Ev_{ic}^* .
- Si $e_{anterior}$ es x_6 y $m_m < 0$ y e_{actual} es x_5 Entonces ocurre Ev_{dp}^* .
- Si $e_{anterior}$ es x_6 y $m_m > 0$ y e_{actual} es x_3 Entonces ocurre Ev_{ic}^* .

El quinto paso consiste en definir las consignas de operación (setpoint) que el sistema de toma de decisión debe seleccionar para cada uno de los estados de operación x_1, x_2, \dots, x_6 . En aras de garantizar un adecuado procesamiento en una determinada unidad de producción, es importante conocer las especificaciones de diseño de la unidad de producción de manera de poder seleccionar las consignas de operación más adecuadas dependiendo del volumen de producción que se está manejando en un determinado momento.

Para propósito de simulación, se han fijado las siguientes consignas de operación: $Sp1 = 2,8pies$, $Sp2 = 3pies$, $Sp3 = 2,5pies$, $Sp4 = 2,3pies$, $Sp5 = 2,7pies$ y $Sp6 = 3,3pies$.

Finalmente, el último paso es definir la tasa de incremento o decremento $\pm \Delta u_j$ de los núcleos de $u(\tau)$. Se ha seleccionado $\Delta u_1 = \pm 0,450$, $\Delta u_2 = \pm 1,675$ y $\Delta u_3 = \pm 2,355$. Para efecto de las pruebas de simulación, sólo se variará un núcleo de una de las funciones de membresía y se fijó a $E_{es} < 0,05$.

En la gráfica de la figura 5.9, se muestra el resultado del comportamiento de la variable controlada $y(\tau)$, los diferentes cambios de setpoint que el sistema de toma de decisión realiza y la implementación del algoritmo de adaptación para minimizar los errores de estado estático.

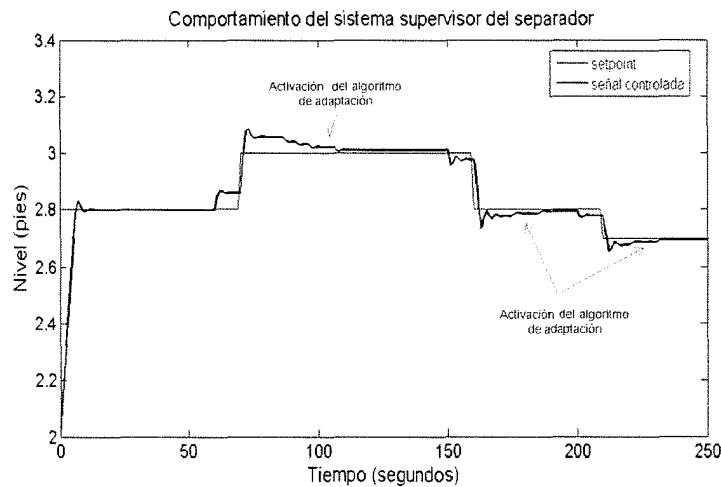


Figura 5.9: Comportamiento del sistema supervisor local

En $t = 60$ segundos, el flujo crudo a la entrada del separador cambia de $32B/H$ a $44B/H$ lo cual genera un incremento de nivel de 2,8 a 2,87 pies. Esto genera el evento “incremento de la tasa de producción” Ev_{tp}^* tal y como se muestra en la figura 5.10. Se trata de un evento que genera un cambio de estado de “tasa de producción normal con baja presión” a “tasa de producción alta con baja presión” tal y como se aprecia en la figura 5.11. En la figura 5.13 se puede apreciar el comportamiento del flujo de crudo a la entrada del separador. Por ende, el sistema de toma de decisión cambia el actual setpoint $Sp1 = 2,8pies$ a $Sp2 = 3pies$.

Dicho cambio de setpoint permite que el nivel de regulación actúe, sin embargo, se presenta un error de estado estático y se activa el mecanismo de adaptación de los núcleos de las funciones de membresía de $u(\tau)$ para minimizar el error de estado estático actual. El núcleo que se modificó fue u_2 del conjunto difuso $U2$ cuyo valor original era 41,66 y luego de la aplicación del algoritmo el valor de u_2 fue de 34,9 tal y como se muestra en la figura 5.16. La figura 5.12 muestra el comportamiento del porcentaje de cierre de la válvula de control. En $t = 150$ segundos se simula el evento decremento de la producción lo cual genera un cambio de estado (al estado inicial) y finalmente en $t = 200$ segundos se simula la ocurrencia del evento incremento de presión como se observa en la figura 5.14. Finalmente, en la figura 5.15 se muestra el comportamiento del error de estado estático.

5.5. VALIDACIÓN DE LA PROPUESTA

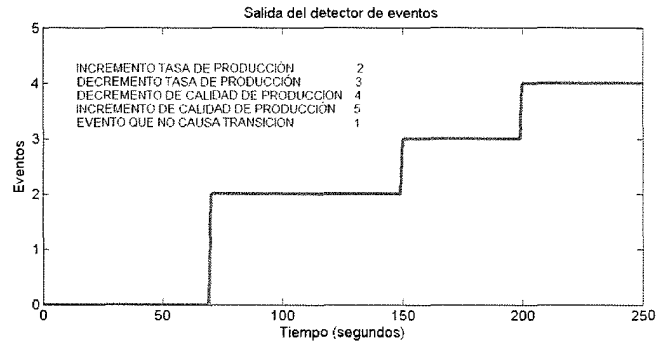


Figura 5.10: Salida del detector de eventos

www.bdigital.ula.ve

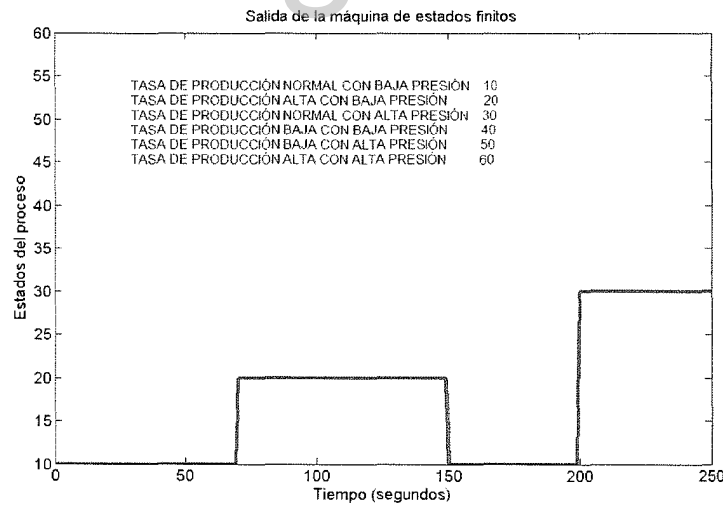


Figura 5.11: Salida de la máquina de estados finitos

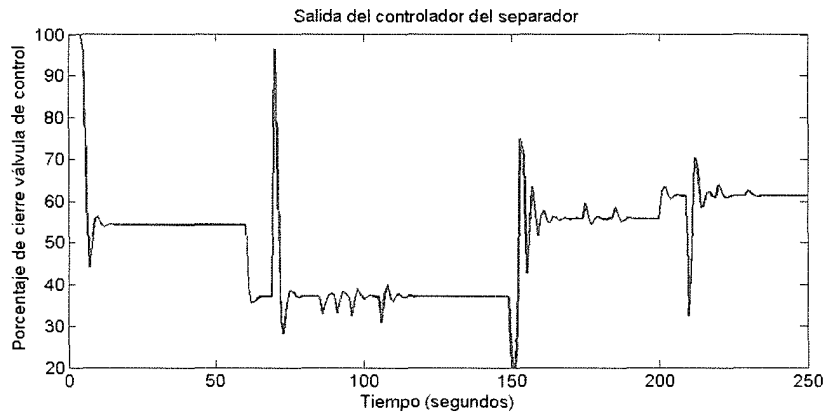


Figura 5.12: Salida del controlador

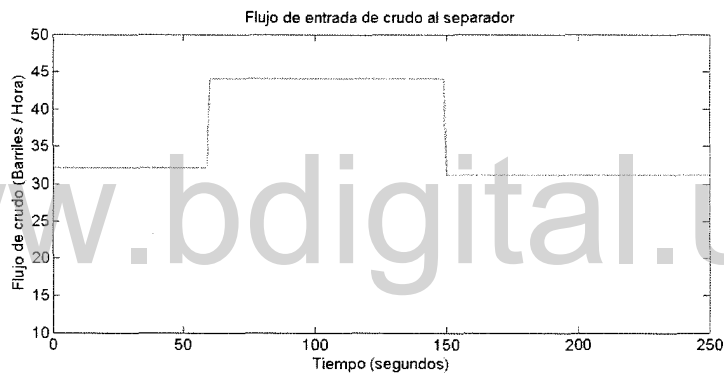


Figura 5.13: Flujo de entrada del separador

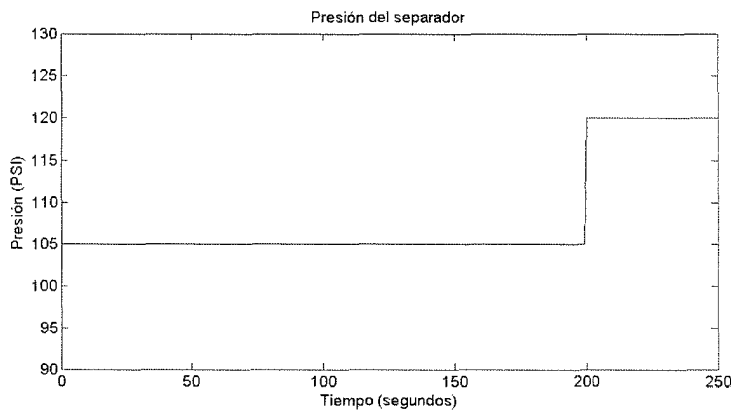


Figura 5.14: Presión del separador

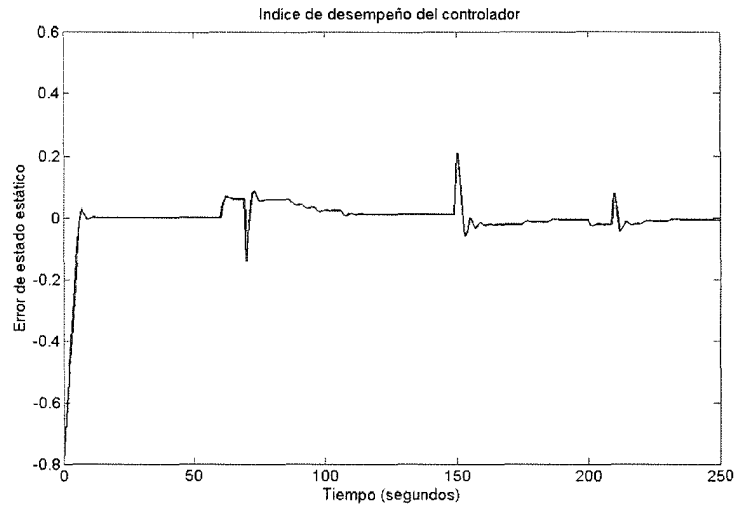


Figura 5.15: Índice de desempeño del controlador

www.bdigital.ula.ve

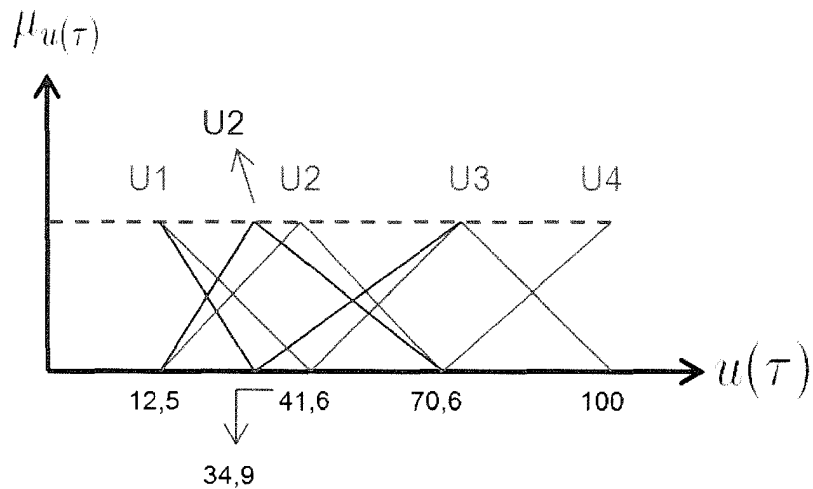


Figura 5.16: Modificación de los núcleos de las funciones de membresía de $u(\tau)$

5.6. Conclusiones

En este capítulo se ha expuesto la principal aportación de esta tesis, que consiste en el desarrollo de una metodología que permite diseñar un sistema de supervisión para sistemas complejos. La principal función del sistema supervisor propuesto es la detección de eventos que generen cambios de estados de operación que permita la toma de decisiones en relación al cambio de la consigna de operación y ajuste en línea del controlador para mejorar el error de estado estático, lo cual garantiza una operación confiable sobre el proceso local.

Se contempló una arquitectura para el sistema de supervisión conformada por los módulos de detección de eventos, el modelo cualitativo del proceso, el sistema de toma de decisión local y el módulo de coordinación. La premisa que se planteó para la definición de los estados de operación del proceso que se desea supervisar y de los eventos estuvo orientada a los cambios en la tasa de producción y calidad de producción.

El modelo cualitativo del proceso se diseñó en base a un autómata de estados finitos de tipo determinista, el cual contiene todos los estados descritos en términos cualitativos del proceso. Dicho autómata de estado finito interactúa directamente con el detector de eventos y es válido si y sólo si no existe la presencia de fallas en las variables que representan la cantidad y calidad de producción. El detector de eventos se encarga de capturar los eventos que generan o no una transición de estados.

El sistema de toma de decisión se encarga por una parte de cambiar la consigna de operación en función del nuevo estado de operación del proceso y por otro lado busca mejorar el desempeño del controlador en caso de ser necesario a través de activación de un mecanismo de adaptación en línea para controladores difusos que mejora el error de estado estático. En general, se puede afirmar que el sistema de supervisión local propuesto realiza toma de decisiones de tipo reactivo. El entorno de simulación usado fue Matlab-Simulink y los resultados de las simulaciones muestra en forma empírica que el controlador en general es estable y controlable.

Vale la pena resaltar que la estrategia de adaptación en línea del controlador difuso consistió en la modificación del núcleo de los consecuentes de las funciones de membresía de la señal de control, lo cual permitió minimizar el error de estado estático. Los resultados de las simulaciones demuestran el gran desempeño del mecanismo de adaptación. Este punto representa un aporte importante de esta tesis para el tema relacionado con la adaptación en línea de controladores difusos

unitarios.

Finalmente, vale la pena resaltar la prueba de concepto implementada sobre un mecanismo de negociación que permita a los agentes supervisores locales cooperar con el sistema supervisor principal para lograr cumplir con el objetivo global. Se utilizó el mecanismo de negociación basado en la teoría de juegos.

www.bdigital.ula.ve

Capítulo 6

Conclusiones, Trabajos futuros y Publicaciones

6.1. Sumario y Conclusiones

En este capítulo se recogen las principales aportaciones de esta tesis, las líneas futuras de investigación, así como las publicaciones realizadas, producidas como resultado del trabajo de investigación realizado. A continuación se describen las conclusiones:

- Se evaluaron las técnicas de agrupamiento difuso, la modelación neurodifusa ANFIS y los algoritmos genéticos como alternativas para la identificación de modelos difusos a partir de datos de entrada - salida. Las técnicas de agrupamiento difuso permitieron identificar modelos difusos Takagi -Sugeno de primer orden de muy buena exactitud a pesar del reducido número de reglas pero presentaron deficiencias en cuanto a la interpretabilidad. Con la implementación del algoritmo ANFIS se logró identificar modelos difusos Takagi -Sugeno de primer orden invertibles mucho más interpretables que las técnicas de agrupamiento manteniendo una buena exactitud. Finalmente el uso de algoritmos genéticos permitió optimizar modelos difusos unitarios invertibles de muy alta interpretabilidad con un número de reglas más reducido que los modelos difusos obtenidos usando ANFIS y logrando mantener una exactitud similar. Esto representa uno de los aportes más significativos de esta tesis.
- Se implementó el método de Babuska (Babuska 1996) que permite calcular de forma exacta la inversa de modelos difusos unitarios. Posteriormente, tras

una revisión de varios esquemas de control basado en modelos, se implementó el controlador difuso bajo el esquema de control por modelo interno. Para ello, fue necesario determinar modelos matemáticos de los procesos de separación gas - líquido y calentamiento de crudo. Se destaca, como una contribución, el modelo matemático obtenido para el horno de fuego directo realizando analogías con el modelo de circuitos eléctricos e incorporando parámetros reales del horno de fuego directo dentro del modelo matemático. Se demostró el buen desempeño del controlador difuso y se verificó de manera empírica su estabilidad y controlabilidad.

- Se ha logrado un aporte importante en cuanto al desarrollo de una metodología que permite diseñar un sistema de supervisión para sistemas complejos. La principal función del sistema supervisor propuesto es la detección de eventos que generen cambios de estados de operación que permita la toma de decisiones en relación al cambio de la consigna de operación y ajuste en línea del controlador para mejorar el error de estado estático, lo cual garantiza una operación confiable sobre el proceso local.
- Se implementó un mecanismo de adaptación en línea para controladores difusos unitarios que consiste en la modificación iterativa del núcleo de los consecuentes de las funciones de membresía de la señal de control y permitió minimizar el error de estado estático. Los resultados de las simulaciones demuestran el gran desempeño del mecanismo de adaptación. Este punto representa una aportación importante de esta tesis sobre todo en el área de adaptación de controladores difusos unitarios

6.2. Trabajos futuros

Algunas de las posibles líneas a seguir, en base a lo desarrollado en esta tesis, se concretan en una serie de cuestiones abiertas entre las que cabría destacar:

- Establecer una propuesta metodológica que permita obtener modelos difusos Takagi - Sugeno de orden 1 invertibles y de muy alta interpretabilidad usando algoritmos genéticos.
- Establecer una estrategia de control predictivo difuso basado en modelos e incorporarlo como una de las tareas de un sistema de supervisión.
- Investigar de manera formal todo lo relacionado a la estabilidad y controlabilidad en controladores difusos.

- Establecer una propuesta formal para la implementación de un sistema multiagente para la supervisión de procesos complejos que presente la suficiente flexibilidad como para que los agentes involucrados puedan negociar entre sí en aras de lograr el objetivo global del proceso.

6.3. Publicaciones relacionadas a esta tesis

En esta sección se describen brevemente los artículos publicados en revistas arbitradas internacionales, así como también las publicaciones presentadas en congresos tanto internacionales como nacionales producto del desarrollo de esta tesis doctoral.

Artículos Publicados en Revistas Arbitradas

- Miguel Ramírez *Sistema Multiagente Aplicado a la Supervisión Distribuida de Procesos de Petróleo*. Aceptado el 20 de Octubre del 2009 para ser publicado en la Revista Colombiana de Tecnologías de Avanzadas ISSN 1692-7257., Vol 2, Número 15. Universidad de Pamplona, IIDTA. Pamplona, Colombia.
- Miguel Ramírez, Jacinto Dávila and Eliezer Colina Morles *Intelligent Supervision Of Petroleum Processes Based On Multi-agent Systems* WSEAS Transactions on SYSTEMS and CONTROL, ISSN 1991-8763, Issue 1, Volume 1, August 2009
- Miguel Angel Ramírez Canelón And Eliezer Colina Morles *Fuzzy Clustering Based Models Applied To Petroleum Processes* WSEAS Transactions on SYSTEMS and CONTROL, ISSN 1991-8763, Issue 3, Volume 3, March 2008.

Artículos Presentados en Congresos Internacionales

- Miguel Ramírez and Eliezer Colina Morles *Fuzzy Clustering Techniques Applied to Petroleum Processes* Proceedings WSEAS Conference on Computational Intelligence, Man-Machine Systems and Cybernetics, Tenerife, Canary Islands, Spain on 14-16 December, 2007

The application of fuzzy clustering techniques has recently become in a very useful alternative in the area of modeling and identification of complex industrial processes. In particular, fuzzy clustering techniques such as Fuzzy C-Means and the Gustafson-Kessel (GK) algorithms will be analyzed and

applied in details in this paper. These algorithms will be implemented in the construction of Takagi-Sugeno fuzzy models for the gas-liquid separation process, the water-oil separation process and the oil-heating process, which are important processes in the oil industry. Validations of the obtained fuzzy models will be performed and some conclusions will be established.

- **Miguel Ramírez And Eliezer Colina Morles *Fuzzy Model Based Control: Application to an Oil Production Separator HIS (Hybrid Intelligent System)*, Universitat Politècnica de Catalunya, Barcelona, Spain on 12-14 September, 2008**

Classical fuzzy controllers have been of the rulebased type, where the rules in the controller attempt to model the response of an operator to a particular process situation. Based on the ability of fuzzy systems to approximate any nonlinear mapping, the nonlinear plant is represented by a zeroth-order Takagi Sugeno fuzzy model and it is analytically inverted for designing a fuzzy controller. In order to minimize the steady-state-error due to model-plant mismatch, an Internal Model Control (IMC) will be considered. The proposed fuzzy controller is applied to control the oil level in a production separator and is shown to be capable of providing good overall system performance.

- **Miguel Angel Ramírez, Jacinto Dávila And Eliezer Colina Morles *Intelligent Supervision Based On Multi-Agent Systems: Application to Petroleum Industrial Processes* Proceedings of the 13th WSEAS International Conference on System, Rhodes Island, Greece, July 22-24, 2009**

This work presents a methodology for the design of an intelligent supervisory system that combines the principles of fuzzy logic, the Internal Model Control (IMC) architecture and the paradigm of Multi-Agent Systems (MAS). The methodology has been conceived to be applied in an intelligent supervisory system, specifically for two kinds of complex petroleum industrial processes: the gas-oil separation process and the oilheating process. The supervision proposal takes into account the fact of using standard local supervisors schemes connected between themselves and to a global supervisor so that local objectives in each process can be met, thereby letting the global or social objective be obtained through the application of basic mechanism of communication, cooperation and coordination; where these objectives have been previously defined and structured in a hierarchical manner. The paper includes some computational simulations performed under MATLAB / SIMULINK and the results obtained show a good overall system performance.

- Miguel Ramírez *Sistemas Multiagentes Aplicados A La Supervisión Distribuida de Procesos de Petróleo VII Congreso Internacional de Electrónica y Tecnología de Avanzadas, Facultad de Ingeniería y Arquitectura de la Universidad de Pamplona, Pamplona, Colombia, del 19-21 de Noviembre, 2009*

Esta publicación presenta un novedoso esquema de supervisión inteligente basado en Sistemas Multiagentes aplicado a procesos industriales de petróleos usando los mismos principios de la propuesta presentada en (Ramírez et al., 2009) en donde básicamente además de la tecnología de agentes, se usa lógica difusa y la configuración CMI (Control por modelo interno). La nueva propuesta consiste en usar agentes especializados para cada tipo de proceso basados en sistemas de predicción de cambios de eventos y con ello se busca establecer un sistema supervisor con características más acentuadas de distribución. Algunos escenarios de simulación computacional fueron desarrollados y se comparan los resultados de ambas propuesta demostrándose el buen desempeño del nuevo esquema.

- Miguel Ramírez And Eliezer Colina *A Genetic-Algorithm Based Approach For Generating Fuzzy Singleton Models Proceedings of the 9th WSEAS International Conference on Computational Intelligence, Man-Machine Systems and Cybernetics (CIMMACS 10), University of Los Andes, Merida, Venezuela, December 14-16, 2010*

Methods for generating fuzzy singleton models from input-output data have been proposed by many authors. This paper introduces a genetic algorithm (GA) based method to generate a fuzzy singleton model taking into account all the necessary constraints to guarantee an analytically inverted representation of the process dynamics which may be used as a fuzzy controller in Internal Model Control (IMC) strategy. A major advantage of this sort of models is its high interpretability compared to first-order Takagi-Sugeno fuzzy models generated from fuzzy clustering techniques. The proposed method is applied to a liquid level control problem in an oil production separator based upon real input-output data, where obtaining an adequate fuzzy model is of crucial importance.

Artículos de Ámbito Nacional

- Miguel Ramirez *Identificación de Modelos Takagi - Sugeno Usando Técnicas de Agrupamiento. I Jornadas de Control Inteligente Laboratorio de Sistemas Inteligentes de la Universidad de los Andes (LabSIula) , Venezuela, 4 de Mayo, 2010.*

- Miguel Ramirez y Eliezer Colina Morles *Control Basado en Modelos Difusos Aplicado A Procesos Industriales Complejos*. XXIII Congreso Latinoamericano de Control Automático y XI Congreso Venezolano de Automatización y Control Universidad de los Andes, Merida, Venezuela, 26-28 de Noviembre, 2008.
- Miguel Ramirez, Felix Garcia y Danilo Navarro *Aplicación De La Lógica Difusa Al Control De Un Pozo De Petróleo Actuado Por Una Bomba De Cavidad Progresiva*. XXIII Congreso Latinoamericano de Control Automático y XI Congreso Venezolano de Automatización y Control Universidad de los Andes, Merida, Venezuela, 26-28 de Noviembre, 2008.

www.bdigital.ula.ve

Apéndice A

Métodos de mínimos cuadrados

Los métodos de mínimos cuadrado han sido concebidos como una herramienta tradicional que brindan una solución al problema de identificación de parámetros. Denótese a g como el sistema físico que se desea identificar y a G como el conjunto de datos experimentales de entrada y salida (tal y como se muestra en la figura A.1). En la identificación de sistemas lineales, se puede usar el siguiente modelo de referencia:

$$y(k) = \sum_{i=1}^q \theta_{a_i} y(k-i) + \sum_{i=0}^p \theta_{b_i} u(k-i) \quad (A.1)$$

En donde $u(k)$ y $y(k)$ son las entradas y salidas en el tiempo k . La ecuación A.1 se puede representar en forma compacta y matricial de la siguiente manera:

$$f(x|\theta) = \theta^T x(k) \quad (A.2)$$

En donde $x(k) = [y(k-1), \dots, y(k-q), u(k), \dots, u(k-p)]^T$ y $\theta = [\theta_{a_1}, \dots, \theta_{a_q}, \theta_{b_0}, \dots, \theta_{b_p}]$. Además se tiene que $N = q + p + 1$ de tal forma que $x(k)$ y θ son vectores $N \times 1$ y

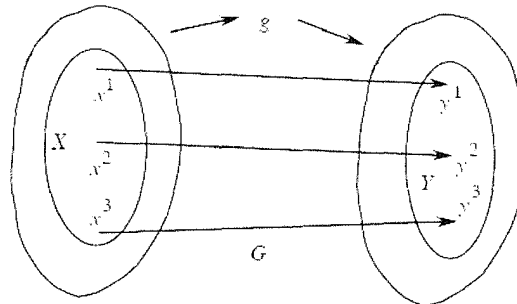


Figura A.1: Mapeo de Función con tres pares de datos

$x(k)$ se denomina el vector de regresión. Por ende para la derivación del método del mínimos cuadrado por batch se define:

$$Y(M) = [y_1, y_2, \dots, y_m]^T$$

como el vector de datos de salida $M \times 1$. Luego se define:

$$\Phi(M) = \begin{bmatrix} x_1^T \\ x_2^T \\ \vdots \\ x_M^T \end{bmatrix} \quad (\text{A.3})$$

como una matriz $M \times N$ que consiste de los x_i vectores. Sea:

$$\epsilon_i = y_i - (x_i)^T \theta$$

el error de aproximación en el par de datos $(x_i, y_i) \in G$ usando θ . Por otro lado, sea:

$$E(M) = [\epsilon_1, \epsilon_2, \dots, \epsilon_M]^T$$

de tal manera que:

$$E = Y - \Phi\theta$$

Y seleccionese a:

$$V(\theta) = \frac{1}{2} E^T E$$

como medida del grado de exactitud de la aproximación de todo el conjunto de datos dado por θ . Se requiere calcular θ tal que $V(\theta)$ sea minimizado. Se asume que $V(\theta)$ es convexo en θ de tal manera que un mínimo local implica un mínimo global. Si se toma algunas ideas básicas del calculo entonces al tomar la derivada parcial de V con respecto a θ , se obtiene una ecuación para $\hat{\theta}$ con la mejor estimación posible en el sentido del mínimo cuadrado. Otra opción consiste en plantear la expresión anterior como:

$$2V = E^T E = Y^T Y - Y^T \Phi \theta - \theta^T \Phi^T Y + \theta^T \Phi^T \Phi \theta$$

Asumiendo que $\Phi^T \Phi$ es invertible, entonces al sumar y restar en el lado izquierdo de la ecuación anterior el termino $(Y^T \Phi (\Phi^T \Phi)^{-1} \Phi^T Y)$ se tiene que:

$$2V = Y^T Y - Y^T \Phi \theta - \theta^T \Phi^T Y + \theta^T \Phi^T \Phi \theta + (Y^T \Phi (\Phi^T \Phi)^{-1} \Phi^T Y) - (Y^T \Phi (\Phi^T \Phi)^{-1} \Phi^T Y) \quad (\text{A.4})$$

Factorizando y organizando los términos de la ecuación anterior se tiene que:

$$2V = Y^T(I - \Phi(\Phi^T\Phi)^{-1}\Phi^T)Y + (\theta - (\Phi^T\Phi)^{-1}\Phi^T\Phi)^T\Phi^T\Phi(\theta - (\Phi^T\Phi)^{-1}\Phi^T\Phi Y) \quad (\text{A.5})$$

Como se puede observar, el primer término de la ecuación anterior no depende de θ y así no es posible minimizar V en función de θ . Para obtener el valor más pequeño posible de V , se selecciona θ de tal forma que el segundo término del lado izquierdo de la ecuación A.5 sea cero. Se denotará $\hat{\theta}$ el valor de θ que minimiza a V y por lo tanto:

$$\hat{\theta} = (\Phi^T\Phi)^{-1}\Phi^TY \quad (\text{A.6})$$

es la ecuación del mínimo cuadrado de tipo batch y que permite obtener el cálculo de $\hat{\theta}$ a partir de los datos contenidos en la matriz de Φ y Y . Para la versión de mínimos cuadrados ponderados se usa la siguiente ecuación:

$$V(\theta) = \frac{1}{2}E^TW E \quad (\text{A.7})$$

En donde W es una matriz de $M \times M$ diagonal con sus elementos diagonales $w_i > 0$ para $i = 1, 2, \dots, M$ y el resto de los elementos iguales a cero. Los términos w_i son usados para asignar mayor importancia a ciertos elementos de G más que a otros. Se puede demostrar (siguiendo los pasos anteriores) que el estimado de θ que minimiza a $V(\theta)$ bajo esta modalidad viene dado por:

$$\hat{\theta}_W = (\Phi^TW\Phi)^{-1}\Phi^TWY \quad (\text{A.8})$$

En terminos generales, el método de mínimos cuadrados por Batch ha sido probado de manera exitosa en una gran variedad de aplicaciones y dada su naturaleza los datos se recolectan y luego se procesan. Para valores pequeños de M no habría ningún problema en usar este método, sin embargo, cuando M es grande entonces los calculos son bastante tediosos debido al término $(\Phi^T\Phi)^{-1}$. En tal sentido, una posible solución a este problema es obtener una versión de mínimos cuadrados de tipo recursivo que permita actualizar a $\hat{\theta}$ cada vez que se presente un grupo de datos nuevos, sin tener que usar todos los datos anteriores y sin tener que calcular la inversa de $\Phi^T\Phi$.

Dado que en lo sucesivo se considerará grandes valores de G , se asume el incremento de este por uno en cada instante de tiempo, sea el índice de tiempo $k = M$ e i tal que $0 \leq i \leq k$. Sea la matriz $N \times N$:

$$P(k) = (\Phi^T\Phi)^{-1} = \left(\sum_{i=1}^K x_i x_i^T\right)^{-1} \quad (\text{A.9})$$

Sea, por otra parte, $\hat{\theta}(k-1)$ el estimado de mínimos cuadrados basado en los $k-1$ pares de datos ($P(k)$ se denomina la matriz de covarianza). Se asume que $\Phi^T\Phi$ es

no singular para todo k . Se tiene entonces que $P^{-1}(k) = \Phi^T \Phi = \sum_{i=1}^k x_i x_i^T$ que de manera equivalente se puede representar como:

$$P^{-1}(k) = \sum_{i=1}^{k-1} [x_i x_i^T + x_k x_k^T] \quad (\text{A.10})$$

Y por lo tanto:

$$P^{-1}(k) = P^{-1}(k-1) + x_k x_k^T \quad (\text{A.11})$$

Usando la ecuación $y(k+1) = f(y(k), y(k-1), \dots, y(k-n_y+1), u(k), u(k-1), \dots, u(k-n_u+1))$ se tiene que:

$$\begin{aligned} \hat{\theta}(k) &= (\Phi^T \Phi)^{-1} \Phi^T Y \\ &= \left(\sum_{i=1}^k x_i x_i^T \right)^{-1} \left(\sum_{i=1}^k x_i y_i \right) \\ &= P(k) \left(\sum_{i=1}^k x_i y_i \right) \\ &= P(k) \left(\sum_{i=1}^{k-1} x_i y_i + x_k y_k \right) \end{aligned} \quad (\text{A.12})$$

De aquí entonces se tiene que:

$$\hat{\theta}(k-1) = P(k-1) \sum_{i=1}^{k-1} x_i y_i \quad (\text{A.13})$$

Esta ecuación se puede reescribir como:

$$P^{-1}(k-1) \hat{\theta}(k-1) = \sum_{i=1}^{k-1} x_i y_i \quad (\text{A.14})$$

Ahora, al reemplazar el valor de $P^{-1}(k-1)$ en esta ecuación con el resultado de la ecuación A.11 se obtiene que:

$$(P^{-1}(k) - x_k x_k^T) \hat{\theta}(k-1) = \sum_{i=1}^{k-1} x_i y_i \quad (\text{A.15})$$

Usando el resultado de la ecuación A.5 esto nos conduce a que:

$$\begin{aligned} \hat{\theta}(k) &= P(k) (P^{-1}(k) - x_k x_k^T) \hat{\theta}(k-1) + P(k) x_k y_k \\ &= \hat{\theta}(k-1) - P(k-1) x_k x_k^T \hat{\theta}(k-1) + P(k) x_k y_k \\ &= \hat{\theta}(k-1) + P(k) x_k (y_k - x_k^T \hat{\theta}(k-1)) \end{aligned} \quad (\text{A.16})$$

La ecuación A.6 provee un método para calcular un estimado de los parámetros $\hat{\theta}(k)$ en cada paso de tiempo k desde la muestra estimada anterior $\hat{\theta}(k-1)$ y el par de datos recientes (x_k, y_k) . Notese que $(y_k - x_k^T \hat{\theta})$ es el error en la predicción de y_k usando $\hat{\theta}(k-1)$. Para actualizar $\hat{\theta}$ en la ecuación A.13 se requiere de $P(k)$, así que se pudiera usar la ecuación A.11. Pero para ello, se requiere calcular la inversa de una matriz en cada instante de tiempo. Claramente, esto no es deseable para la implementación en tiempo real, así que se requiere evitar esta situación. Para ello es importante recordar una propiedad fundamental del calculo matricial que indica que si A, C , y $(C^{-1} + DA^{-1}B)$ son matrices cuadradas no singulares, entonces $A + BCD$ es invertible por otro lado se tiene que:

$$(A + BCD)^{-1} = A^{-1} - A^{-1}B(C^{-1} + DA^{-1}B)DA^{-1}$$

Se usará este hecho para eliminar la necesidad de calcular la inversa de $P(k)$ obtenida en la ecuación A.11 para que pueda ser usada en la ecuación A.13 y así poder actualizar $\hat{\theta}$. Es de hacer notar que:

$$\begin{aligned} P(k) &= (\Phi^T(k)\Phi(k))^{-1} \\ &= (\Phi^T(k-1)\Phi(k-1) + x_k x_k^T)^{-1} \\ &= (P^{-1}(k-1) + x_k x_k^T)^{-1} \end{aligned}$$

En base a la propiedad fundamental del calculo matricial mencionada y haciendo $A = P^{-1}(k-1)$, $B = x_k$, $C = I$ y $D = x_k^T$, se obtiene que:

$$P(k) = P(k-1) - P(k-1)x_k(I + x_k^T P(k-1)x_k)^{-1}x_k^T P(k-1) \quad (\text{A.17})$$

Esta ecuación en conjunto con la ecuación A.9 conforman el algoritmo de mínimos cuadrados recursivos (MCR). Se puede demostrar que el termino $(I + x_k^T P(k-1)x_k)$ es un escalar, así que la inversión de una matriz se convirtió aplicando propiedades del calculo matricial en la inversión de un escalar. Finalmente vale la pena mencionar que existe una versión modificada del MCR denominada algoritmo de mínimos cuadrados recursivos ponderados (MCRP). Esta versión es muy útil para el caso en donde los parámetros de un sistema físico varían de manera apreciable, para ello es necesario minimizar la siguiente expresión:

$$V(\theta, k) = \frac{1}{2} \sum_{i=1}^k \lambda^{k-i} (y_i - x_i \theta)^2 \quad (\text{A.18})$$

En donde $0 < \lambda \leq 1$ se denomina factor de olvido dado que le da mayor o menor peso a los datos más recientes. Usando un enfoque similar para la deducción del algoritmo MCR se puede demostrar que el algoritmo MCRP viene dado por las

dos siguientes ecuaciones:

$$P(k) = \frac{1}{\lambda}(I - P(k-1)x_k(\lambda I + x_k^T P(k-1)x_k)^{-1}x_k^T)P(k-1) \quad (\text{A.19})$$

$$\hat{\theta}(k) = \hat{\theta}(k-1) + P(k)x_k(y_k - (x_k)\hat{\theta}(k-1)) \quad (\text{A.20})$$

Ejemplo: Entrenamiento de modelos difusos Takagi-Sugeno En este breve ejemplo se mostrará que los modelos Takagi-Sugeno pueden ser parametrizados de tal manera que sean lineales en sus parámetros (De manera similar a la ecuación A.2) y se puedan aplicar las técnicas de mínimos cuadrados. Se sabe que la salida de un modelo difuso Takagi-Sugeno puede ser expresada de la siguiente manera:

$$y = \frac{\sum_{i=1}^C \beta_i(x) f_i(x)}{\sum_{i=1}^C \beta_i(x)}$$

En donde, en este caso particular:

$$f_i(x) = a_{i,0} + a_{i,1}x_1 + \dots + a_{i,n}x_n$$

Por lo que entonces la salida del sistema difuso TS se puede expresar como:

$$y = \frac{\sum_{i=0}^C a_{i,0} \beta_i(x)}{\sum_{i=1}^C \beta_i(x)} + \frac{\sum_{i=0}^C a_{i,1} x_1 \beta_i(x)}{\sum_{i=1}^C \beta_i(x)} + \dots + \frac{\sum_{i=0}^C a_{i,n} x_n \beta_i(x)}{\sum_{i=1}^C \beta_i(x)}$$

Tomando en cuenta la definición de $\gamma_i(x)$ en (2.18) entonces podemos redefinir a $\gamma(x)$ como:

$$\gamma(x) = [\gamma_1(x), \gamma_2(x), \dots, \gamma_C(x), x_1\gamma_1(x), x_1\gamma_2(x), \dots, x_1\gamma_C(x), \dots, x_n\gamma_1(x), x_n\gamma_2(x), \dots, x_n\gamma_C(x)]$$

Y θ como:

$$\theta = [a_{1,0}, a_{2,0}, \dots, a_{C,0}, a_{1,1}, a_{2,1}, \dots, a_{C,1}, \dots, a_{1,n}, a_{2,n}, \dots, a_{C,n}]^T$$

De tal manera que:

$$y = f(x|\theta) = \theta^T \gamma(x)$$

representa un modelo Takagi-Sugeno sobre el cual se pueden aplicar las técnicas de mínimos cuadrados descritas en este apéndice.

Apéndice B

Aplicación de técnicas neurodifusas para la identificación de modelos difusos

El modelo difuso del separador obtenido al aplicar técnicas neurodifusas se muestra a continuación:

- Si $u(\tau)$ esta Abierta y $y(\tau)$ es Bajo Entonces $\hat{y}(\tau + 1) = 0,0080u(\tau) + 0,00076y(\tau) + 0,00042$
- Si $u(\tau)$ esta Abierta y $y(\tau)$ es Normal Entonces $\hat{y}(\tau + 1) = 0,1336u(\tau) + 0,0135y(\tau) + 0,0066$
- Si $u(\tau)$ esta Abierta y $y(\tau)$ es Alto Entonces $\hat{y}(\tau + 1) = 0,1521u(\tau) + 0,017y(\tau) + 0,0067$
- Si $u(\tau)$ esta Abierta y $y(\tau)$ es Muy-Alto Entonces $\hat{y}(\tau + 1) = 0,2134u(\tau) + 0,05286y(\tau) + 0,01449$
- Si $u(\tau)$ esta Media-Abierta y $y(\tau)$ es Bajo Entonces $\hat{y}(\tau + 1) = 0,2575u(\tau) + 0,07883y(\tau) + 0,0204$
- Si $u(\tau)$ esta Media-Abierta y $y(\tau)$ es Normal Entonces $\hat{y}(\tau + 1) = 0,02144u(\tau) + 0,0008y(\tau) + 0,0006$
- Si $u(\tau)$ esta Media-Abierta y $y(\tau)$ es Alto Entonces $\hat{y}(\tau + 1) = 0,0348u(\tau) + 0,0024y(\tau) + 0,0008$
- Si $u(\tau)$ esta Media-Abierta y $y(\tau)$ es Muy-Alto Entonces $\hat{y}(\tau + 1) = 0,047u(\tau) + 0,0042y(\tau) + 0,0018$

APÉNDICE B. APLICACIÓN DE TÉCNICAS NEURODIFUSAS PARA LA IDENTIFICACIÓN DE MODELOS DIFUSOS

- Si $u(\tau)$ esta Cerrada y $y(\tau)$ es Bajo Entonces $\hat{y}(\tau + 1) = 0,014u(\tau) + 0,0001y(\tau)$
- Si $u(\tau)$ esta Cerrada y $y(\tau)$ es Normal Entonces $\hat{y}(\tau + 1) = 0,025u(\tau) + 0,0012y(\tau)$
- Si $u(\tau)$ esta Cerrada y $y(\tau)$ es Alto Entonces $\hat{y}(\tau + 1) = 0,0374u(\tau) + 0,0037y(\tau)$
- Si $u(\tau)$ esta Cerrada y $y(\tau)$ es Muy-Alto Entonces $\hat{y}(\tau + 1) = 0,042u(\tau) + 0,0007y(\tau) - 0,0001$
- Si $u(\tau)$ esta Muy-Cerrada y $y(\tau)$ es Bajo Entonces $\hat{y}(\tau + 1) = 0,051u(\tau) + 0,0020y(\tau) + 0,0005$
- Si $u(\tau)$ esta Muy-Cerrada y $y(\tau)$ es Normal Entonces $\hat{y}(\tau + 1) = 0,010u(\tau) + 0,00001y(\tau)$
- Si $u(\tau)$ esta Muy-Cerrada y $y(\tau)$ es Alto Entonces $\hat{y}(\tau + 1) = 0,0188u(\tau) + 0,0006y(\tau) + 0,0001$
- Si $u(\tau)$ esta Muy-Cerrada y $y(\tau)$ es Muy-Alto Entonces $\hat{y}(\tau + 1) = 0,025u(\tau) + 0,00023y(\tau)$

El modelo difuso del horno obtenido al aplicar técnicas neurodifusas se muestra a continuación:

- Si $u(\tau)$ esta Muy-Abierta y $y(\tau)$ es Muy-Baja Entonces $\hat{y}(\tau + 1) = 0,1078u(\tau) + 0,9899y(\tau) + 0,0058$
- Si $u(\tau)$ esta Muy-Abierta y $y(\tau)$ es Baja Entonces $\hat{y}(\tau + 1) = 0,1093u(\tau) + 0,9926y(\tau) + 0,0057$
- Si $u(\tau)$ esta Muy-Abierta y $y(\tau)$ es Normal Entonces $\hat{y}(\tau + 1) = 0,101u(\tau) + 0,989y(\tau) + 0,0054$
- Si $u(\tau)$ esta Muy-Abierta y $y(\tau)$ es Alta Entonces $\hat{y}(\tau + 1) = 0,1135u(\tau) + 1,003y(\tau) + 0,0053$
- Si $u(\tau)$ esta Muy-Abierta y $y(\tau)$ es Muy-Alta Entonces $\hat{y}(\tau + 1) = 0,04922u(\tau) + 0,3748y(\tau) + 0,0001$
- Si $u(\tau)$ esta Abierta y $y(\tau)$ es Muy-Baja Entonces $\hat{y}(\tau + 1) = 0,1402u(\tau) + 0,9782y(\tau) + 0,0057$

-
- Si $u(\tau)$ esta Abierta y $y(\tau)$ es Baja Entonces $\hat{y}(\tau + 1) = 0,1596u(\tau) + 0,9708y(\tau) + 0,0056$
 - Si $u(\tau)$ esta Abierta y $y(\tau)$ es Normal Entonces $\hat{y}(\tau + 1) = 0,1713u(\tau) + 0,9716y(\tau) + 0,0054$
 - Si $u(\tau)$ esta Abierta y $y(\tau)$ es Alta Entonces $\hat{y}(\tau + 1) = 0,1174u(\tau) + 0,9773y(\tau) + 0,0050$
 - Si $u(\tau)$ esta Abierta y $y(\tau)$ es Muy-Alta Entonces $\hat{y}(\tau + 1) = 0,14u(\tau) + 1,02y(\tau) + 0,0053$
 - Si $u(\tau)$ esta Media-Abierta y $y(\tau)$ es Muy-Baja Entonces $\hat{y}(\tau+1) = 0,3204u(\tau)+ 1,04y(\tau) + 0,006$
 - Si $u(\tau)$ esta Media-Abierta y $y(\tau)$ es Baja Entonces $\hat{y}(\tau + 1) = 0,2769u(\tau) + 0,9523y(\tau) + 0,005$
 - Si $u(\tau)$ esta Media-Abierta y $y(\tau)$ es Normal Entonces $\hat{y}(\tau+1) = 0,2285u(\tau)+ 0,9405y(\tau) + 0,005$
 - Si $u(\tau)$ esta Media-Abierta y $y(\tau)$ es Alta Entonces $\hat{y}(\tau + 1) = 0,2359u(\tau) + 0,9341y(\tau) + 0,0051$
 - Si $u(\tau)$ esta Media-Abierta y $y(\tau)$ es Muy-Alta Entonces $\hat{y}(\tau+1) = 0,2432u(\tau)+ 0,9386y(\tau) + 0,0050$
 - Si $u(\tau)$ esta Poco-Abierta y $y(\tau)$ es Muy-Baja Entonces $\hat{y}(\tau+1) = 0,2343u(\tau)+ 0,9354y(\tau) + 0,004$
 - Si $u(\tau)$ esta Poco-Abierta y $y(\tau)$ es Baja Entonces $\hat{y}(\tau + 1) = 0,0524u(\tau) + 0,1704y(\tau) + 0,001$
 - Si $u(\tau)$ esta Poco-Abierta y $y(\tau)$ es Normal Entonces $\hat{y}(\tau+1) = 0,3484u(\tau)+ 0,886y(\tau) + 0,0051$
 - Si $u(\tau)$ esta Poco-Abierta y $y(\tau)$ es Alta Entonces $\hat{y}(\tau + 1) = 0,3733u(\tau) + 0,8646y(\tau) + 0,0051$
 - Si $u(\tau)$ esta Poco-Abierta y $y(\tau)$ es Muy-Alta Entonces $\hat{y}(\tau+1) = 0,3055u(\tau)+ 0,8913y(\tau) + 0,0048$
 - Si $u(\tau)$ esta Cerrada y $y(\tau)$ es Muy-Baja Entonces $\hat{y}(\tau + 1) = 0,3554u(\tau) + 0,8766y(\tau) + 0,0045$

APÉNDICE B. APLICACIÓN DE TÉCNICAS NEURODIFUSAS PARA LA IDENTIFICACIÓN DE MODELOS DIFUSOS

- Si $u(\tau)$ esta Cerrada y $y(\tau)$ es Baja Entonces $\hat{y}(\tau + 1) = 0,3397u(\tau) + 0,9235y(\tau) + 0,0048$
- Si $u(\tau)$ esta Cerrada y $y(\tau)$ es Normal Entonces $\hat{y}(\tau + 1) = 0,34u(\tau) + 0,7632y(\tau) + 0,0044$
- Si $u(\tau)$ esta Cerrada y $y(\tau)$ es Alta Entonces $\hat{y}(\tau + 1) = 0,3736u(\tau) + 0,8332y(\tau) + 0,0049$
- Si $u(\tau)$ esta Cerrada y $y(\tau)$ es Muy-Alta Entonces $\hat{y}(\tau + 1) = 0,3752u(\tau) + 0,8298y(\tau) + 0,0046$

www.bdigital.ula.ve

Apéndice C

Negociación entre agentes

C.1. Introducción

El objetivo de este apéndice consiste en implementar una prueba de concepto que permita evaluar un proceso de negociación basado en la teoría de juegos en un sistema multiagente aplicado muy específicamente al procesamiento de petróleo.

Para ello se va a revisar brevemente todo lo relacionado a la teoría de negociación en sistemas multiagente haciendo énfasis en la teoría de juegos. Luego se presenta brevemente como se implementaría la metodología INGENIAS para finalmente mostrar un caso de estudio que involucra agentes supervisando pozos y separadores de producción.

Finalmente, en el siguiente apéndice se muestran algunos diagramas escaleras de la implementación del mecanismo de negociación realizada en plataforma de controladores lógicos programables.

C.2. Negociación entre agentes

De acuerdo con (Rizo, Llorens, and Pujol 2002) , la negociación es importante en la modelización de sistemas multiagente. Se puede considerar como un proceso de comunicación entre un grupo de agentes para cumplir un contrato mutuamente aceptado. La negociación en los sistemas multiagente se observa desde varios puntos de vista. Por una parte, tanto la asignación de subproblemas como la asignación de recursos puede contemplarse como una negociación; por otro lado, también deben existir negociaciones entre agentes individuales para desarrollar una cooperación en beneficio del sistema(Sierra, Farantin, and Jennings 1997). La

negociación puede ser competitiva o cooperativa, según el comportamiento de los agentes individuales.

Existen varias situaciones diferentes a la hora de entablar una negociación:

- Ninguno de los agentes obtiene beneficio a través de la negociación, puesto que cada agente tiene su propio objetivo y no existen dependencias directas entre ambos objetivos.
- Al menos uno de los agentes logra el objetivo de forma más rápida o con menos esfuerzo.
- Existen situaciones reales de conflicto entre los agentes, debido a recursos, logros de objetivos similares, etc.

En términos formales, la negociación se define como el encuentro para resolver una situación entre diferentes participantes (Farantin, Sierra, and Jennings 1998). En el encuentro, cada uno de los participantes oferta y demanda un conjunto de recursos así como las condiciones de negociación, que permiten llegar a un acuerdo. Si el resultado de la negociación es positivo, cada uno de los participantes se compromete a desarrollar los acuerdos realizados. Este compromiso no es permanente debido a que se producen situaciones posteriores al acuerdo donde el entorno sufra modificaciones.

Para lograr acuerdos entre los diferentes elementos de la comunidad, los mecanismos de interacción entre todos ellos deben estar claramente definidos y deben ser conocidos y aceptados por todos. Estos mecanismos deben definir el vocabulario, el lenguaje de comunicación y las reglas que definen la negociación. El conjunto de estos elementos se denomina protocolo de negociación (Rizo, Llorens, and Pujol 2002).

Uno de los enfoques usado para estudiar la negociación es la teoría de juegos. La teoría de juegos estudia las interacciones entre los distintos elementos que participan en una negociación. Un juego es una descripción de la estrategia de interacción que incluye las restricciones sobre las acciones que los elementos pueden realizar así como de sus intereses pero en ningún momento especifica las acciones que se lleva a cabo. Los elementos que componen un juego son: participantes o jugadores, acciones, información o estrategias, beneficios, resultados y equilibrios (Tomas 2007).

Los participantes son elementos individuales que pueden tomar decisiones a lo largo del juego con el objetivo de maximizar su beneficio. Las acciones definen el conjunto posible de actuaciones que un participante puede desarrollar,

dependiendo del conocimiento que el jugador tenga del entorno de negociación, esto es de la información. Las estrategias definen los planes de acción que cada participante ejecuta a lo largo del juego y dependiendo de ellas obtienen unos beneficios. Los beneficios cuantifican el resultado final de la negociación. El resultado define los acuerdos tanto positivos como negativos obtenidos en el proceso. Por último, el equilibrio define la equidad del protocolo independientemente del participante (Tomas 2007).

C.2.1. Fundamentos de un protocolo de negociación desde el punto de vista de la teoría de juegos

El modelo de negociación basado en la teoría de juegos necesita de una descripción explícita de la estructura secuencial del proceso y de los problemas de decisión de los jugadores. Esta descripción se denomina protocolo de negociación y juega un papel significativo para determinar el resultado de la negociación. Los componentes de un protocolo de negociación son los siguientes (Kraus 2001) (Rosenchein 1994) :

- El entorno de negociación, con los agentes que participan, los roles que cada uno de ellos puede desarrollar, los servicios, recursos y la agenda donde se determinan los servicios a negociar.
- Las reglas de encuentro, donde se define el modo en que los agentes realizan las ofertas y contraofertas, la función de utilidad que permite cuantificar una oferta y determinar cuando se acepta o no, y por último, cuando se implementan los servicios acordados.
- La situación de conflicto donde se especifica lo que ocurre en el entorno sino se llega a un acuerdo en el protocolo. Esto es, si se sobrepasa el tiempo máximo de la negociación o si alguno de los agentes implicados en ella la abandonan.
- Los estados de información de los agentes que definen el conocimiento que estos tienen del entorno de negociación y las estrategias de los agentes que permiten determinar cual es la siguiente acción que un jugador debe tomar en el siguiente instante de tiempo.
- Los escenarios de negociación en los que los agentes se pueden encontrar, el conjunto de posibles resultados dependiendo de la composición de la agenda y cuales son las estrategias óptimas que maximizan el beneficio de los jugadores durante la negociación.

C.2.2. El Entorno de negociación

Las principales características de un entorno de negociación son las siguientes (Tomas 2007):

- El tiempo y la noción que los agentes implicados tienen del mismo. Por un lado, los agentes pueden trabajar con un horizonte infinito, donde no existe ningún tiempo máximo donde el acuerdo debe alcanzarse o bien con un horizonte finito donde existen unos instantes de tiempo que definen los valores temporales tope hasta los que se puede negociar (Osborne and Rubinstein 1994).
- El conocimiento del entorno de negociación que los distintos agentes tienen. En entornos de información completa, los agentes conocen todas las posibles variables que intervienen en la negociación incluyendo la información privada del resto de los agentes. En contraposición a este conocimiento, se encuentran los entornos donde la información que disponen los agentes es incompleta, esto es, desconocen algunas de las variables que median en el protocolo de negociación (Osborne and Rubinstein 1994).
- La tercera característica que define un protocolo de negociación es el número de elementos que intervienen en la negociación. Los protocolos que negocian sobre más de un elemento llevan asociada una agenda de negociación. Ésta define los elementos que se negocian y el orden de negociación si es que existiera.

C.2.3. Las reglas de encuentro

Las reglas de encuentro definen las pautas de comportamiento de los agentes a lo largo de la negociación, es decir, definen las circunstancias bajo las cuales los participantes en la negociación realizan las interacciones. Para ello se debe manejar los siguientes conceptos (Tomas 2007):

- Función de utilidad: permite valorar los beneficios de las ofertas que se reciben y las contraofertas que se realizan.
- Generación de ofertas y contraofertas: a partir de la función de utilidad que cuantifica las ofertas y las contraofertas, un agente puede determinar cual es el siguiente paso que debe realizar cuando recibe una oferta. Dada una oferta en un instante de tiempo t , un agente puede realizar tres acciones: (i) aceptar la oferta, (ii) rechazar la oferta y realizar una contraoferta y (iii) abandonar la negociación con lo que se alcanza la situación de conflicto.

La generación de los valores de la oferta inicial o de las siguientes contraofertas se realiza mediante la aplicación de funciones llamadas tácticas (Tomas 2007) (Farantin, Sierra, and Jennings 1998). Estas funciones o tácticas se calculan en base a uno de los componentes del protocolo de negociación (tiempos, recursos, etc). Existen tres aproximaciones para la generación de tácticas.

Tácticas dependientes del tiempo: en este tipo de tácticas, el principal criterio a utilizar es el tiempo que resta para alcanzar el tiempo límite de la negociación. Así pues, éstas tácticas varían el valor de la oferta a realizar dependiendo del instante de tiempo t en el que se encuentre la negociación y del tiempo límite, T_{max} .

Estas tácticas definen una función $\phi_r^a(t)$ dependiendo del tiempo que modela la evolución de las ofertas. El resultado de esta función debe estar siempre dentro del intervalo $[0, 1]$, es decir, $0 \leq \phi_r^a(t) \leq 1$ y que $\phi_r^a(T_{max}) = 1$. Esto garantiza que el valor resultante de aplicar la función se mantenga siempre dentro del rango de intervalos válidos para la negociación.

La forma de implementar la función $\phi_r^a(t)$ proporciona un amplio rango de funciones dependientes del tiempo. (Tomas 2007) distingue dos tipos de familias de funciones: polinómicas y exponenciales (Farantin 2000). Ambas están parametrizadas por un valor $\beta \in \mathfrak{R}$ que determina el grado de convergencia de la función $\phi_r^a(t)$. Para un mismo valor elevado de β la función polinómica concede más rápidamente al inicio que la exponencial, teniendo a partir de entonces un comportamiento similar. Para valores pequeño de β , la función exponencial espera más tiempo que la polinómica hasta empezar a conceder. La función polinómica se define entonces como:

$$\phi_r^a(t) = \left(\frac{\min(t, T_{max})}{T_{max}} \right)^{\frac{1}{\beta}} \quad (C.1)$$

Y en modo exponencial:

$$\phi_r^a(t) = e^{\left(1 - \frac{\min(t, T_{max})}{T_{max}}\right)\beta} \quad (C.2)$$

En donde T_{max} es el tiempo máximo que dura la negociación y t es el tiempo transcurrido. La función $\phi_r^a(t)$ proporciona un infinito numero de tácticas posible, una para cada valor de β . Este infinito conjunto de valores ha sido clasificado y agrupado en tres conjuntos de tácticas:

- Boulware: donde para valores de β inferiores a 1 la oferta inicial se mantiene hasta que casi se agota el tiempo de negociación.

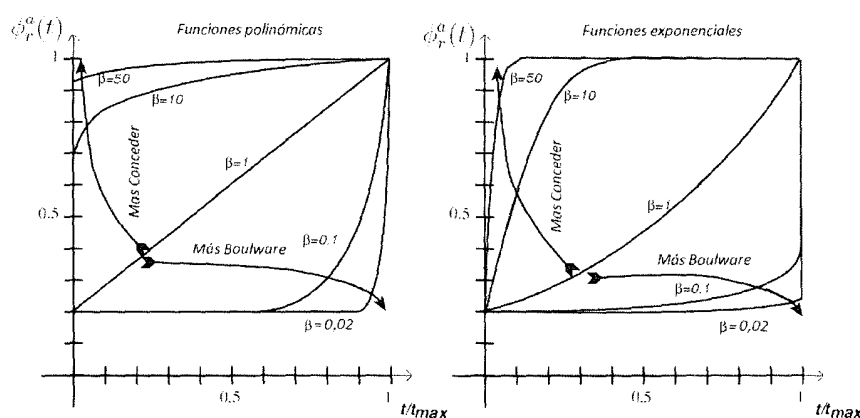


Figura C.1: Funciones para el calculo de $\phi_r^a(t)$

- Conceder: donde para valores de β superiores a 1, la oferta converge rápidamente hacia el máximo valor que el jugador puede negociar sobre este recurso.
- Lineal: donde para el valor de β igual a 1, la oferta se incrementa linealmente conforme avanza el tiempo.

En la figura C.1 se muestra tanto a la función polinómica (a la izquierda) y la exponencial (a la derecha) de $\phi_r^a(t)$. En las gráficas se observan los posibles resultados dependiendo del valor de β .

Tácticas dependientes de los recursos: La generación de tácticas dependientes de los recursos se realiza de manera similar a la generación de tácticas basadas en el tiempo. Es posible que existan casos en donde se identifican las tácticas basadas en tiempo como un tipo de tácticas basadas en recursos donde el recurso que usa la táctica es el tiempo de negociación (Sierra, Farantin, and Jennings 1997).

Por consiguiente, el modelado de tácticas dependientes de los recursos se realiza de manera similar a la expuesta en el apartado anterior, utilizando las mismas funciones, pero utilizando el valor del cual depende la negociación, cualquier tipo de recurso, dinámico. Este valor representa un heurístico sobre cuantos recursos hay en el entorno. Para situaciones en las que los recursos son escasos es necesario aumentar la presión de la negociación para llegar a acuerdos lo antes posible, mientras que si los recursos son abundantes la negociación puede ser relajada (Sierra, Farantin, and Jennings 1997).

C.2.4. La Situación de conflicto

La situación de conflicto se define como la peor situación resultante con el fin de que los agentes no estén tentados a abandonar la negociación. En (Kraus 2001) se argumenta la participación de agentes autointeresados en la negociación en la que, por un lado, normalmente los procesos de negociación no tienen mayor costo que el tiempo empleado y por otro lado, es importante negociar para prevenir posibles situaciones futuras en las que esté interesado en concretar la negociación.

C.2.5. Estrategia e información sobre la negociación

Una estrategia de negociación es una especificación de acciones normalmente ofertas y respuestas, que un agente planea realizar a lo largo de la negociación. Las estrategias combinan las tácticas en función del histórico de la negociación y de la información que el jugador tiene sobre el entorno de la negociación. Una estrategia de negociación de un agente es una función que a partir del histórico de la negociación genera el siguiente movimiento.

El propósito de un agente es encontrar la estrategia óptima. Una estrategia óptima es aquella que le permite determinar la mejor acción de todas las posibles para lograr un acuerdo sobre un elemento de negociación. Este acuerdo es el que maximiza la función de utilidad definida por el agente (Kraus, Winkenfeld, and Zlotkin 1995).

Por otra parte, los estados de información de los agentes determinan el conocimiento que cada jugador tiene sobre el entorno de negociación. La información que un agente posee puede clasificarse como pública, esto es conocida por todos los agentes involucrados en la negociación, o privada si sólo es conocida por éste. Los modelos basados en entornos de negociación donde los agentes no conocen todos los elementos del entorno de negociación se denominan entornos de negociación con información incompleta. Esta falta de información se centra en los parámetros específicos de los oponentes en la negociación, como por ejemplo los factores de descuento sobre el tiempo, los instantes máximo de la negociación o los valores máximos de aceptación de un determinado recurso (Kraus 2001).

C.2.6. Protocolos de negociación basados en la teoría de juegos

Se han desarrollado diferentes protocolos de negociación que utilizan de manera diversa los elementos que intervienen en el entorno de negociación. Cada uno de estos protocolos se encuentran en un dominio concreto, aunque la mayor parte se centra en el dominio de la logística, la localización de recursos y de la economía y los mercados. Algunos de los protocolos adaptados al área de agentes más relevantes se describen a continuación.

Rubinstein define el concepto de protocolo de ofertas alternativas (Rubinstein 1952). En este protocolo dos agentes negocian sobre el contenido de una tarta de tamaño 1. Un acuerdo sobre la negociación es un par $X = (X_1, X_2)$ donde X_i define la parte de la tarta que comparte el agente i . Las preferencias de los agentes sobre X son diametralmente opuestas, es decir, cada agente prefiere tener un trozo de tarta lo más grande posible. El protocolo de negociación es el siguiente: los agentes pueden realizar acciones en instantes de tiempo en el conjunto infinito de valores $T = \{1, 2, \dots\}$. En cada instante un agente i propone un acuerdo a otro agente j . El agente j tiene dos posibles alternativas: aceptar la oferta o rechazarla. Si j acepta la oferta realizada por i , la negociación culmina y el acuerdo es implementado. Pero si j no acepta la oferta, el instante de tiempo pasa a ser $t+1$ y el agente j realiza ahora una oferta al agente i que puede aceptar o rechazar.

El entorno de negociación de este modelo presenta dos inconvenientes: por un lado utiliza un horizonte de negociación infinito, lo que es imposible aplicar a situaciones reales, donde el tiempo es un factor importante en la negociación, y por otro lado no incluye penalizaciones según avanza el tiempo lo que hace que los agentes no tengan prisa para alcanzar un acuerdo.

(Harsanyi and Selten 1972) introduce el concepto de negociación con información incompleta y desarrolla un protocolo que proporciona una solución general para la negociación de dos agentes con información incompleta, aunque sigue manteniendo un horizonte infinito de negociación. Otro modelo de información incompleta (Fudenberg, Levine, and Tirole 1985) analiza la negociación entre un agente vendedor y un agente comprador con un horizonte infinito en el cual, los precios de reserva de los agentes es desconocido.

Los protocolos presentados en (Kraus 2001)(Fatima, Wooldridge, and Jennings 2004b) permiten ser implementados en entornos reales debido a sus características: información incompleta y límites temporales “deadlines”. Sin embargo, es importante aclarar que todos ellos negocian un sólo elemento. Actualmente se está

C.3. PROPUESTA DE UN PROTOCOLO DE NEGOCIACIÓN PARA UN ENTORNO MULTIAGENTE EN PROCESOS CONTINUOS

trabajando en el desarrollo de protocolos multielemento, en los que hay más de un elemento como objeto de negociación. Estos protocolos introducen el concepto de agenda (Tomas 2007). La agenda de negociación identifica no sólo los elementos que intervienen en la negociación sino que también define el orden en que se negocian los elementos. La agenda de negociación puede ser exógena, es decir, definida externamente por los agentes antes de que comience la negociación o por el contrario endógena, donde una vez alcanzado un acuerdo/desacuerdo sobre un elemento de la misma se decide cual es el siguiente elemento a negociar (Fatima, Wooldridge, and Jennings 2004a).

Uno de los factores a determinar en protocolos multielemento es cuando la negociación ha tenido éxito. (Busch and Horstmann 1999) presenta un modelo en el que sólo se considera que la negociación ha finalizado con éxito si se ha llegado a un acuerdo sobre todos los elementos que componen la agenda. Además, la implementación de los acuerdos sobre los elementos de la agenda se realiza una vez la negociación ha terminado.

C.3. Propuesta de un Protocolo de Negociación Para un Entorno Multiagente en Procesos Continuos

El diseño del protocolo de negociación que se propone a continuación tiene su motivación en los modelos de de coordinación provenientes de la teoría de juegos. En este protocolo las interacciones se modelan como una secuencia de ofertas y contraofertas alternativas tal y como se proponen en (Rubinstein 1952). Los intervalos de tiempo en los que se producen las ofertas y contraofertas son de tiempo discretos y equidistantes.

La secuencia de ofertas y contraofertas pueden terminar en un acuerdo mutuamente aceptable para ambos agentes o pueden terminar sin éxito alguno. El protocolo se inicia con un diálogo que permite establecer las condiciones de la negociación. Estas condiciones se satisfacen mutuamente en la fase de pre-negociación tal y como se muestra en la figura C.2 (Farantin 2000).

En términos formales, sea $i \in \{a, b\}$ dos agentes negociantes y $j \in \{1, 2, 3, \dots, n\}$ los elementos que se desean negociar. Se asume que el conjunto de elementos a negociar es finito. Sea $D_j^i = [\min_j^i, \max_j^i]$ los intervalos de los valores de cada elemento a negociar. Entonces cada agente tiene una función de utilidad U_j^i para

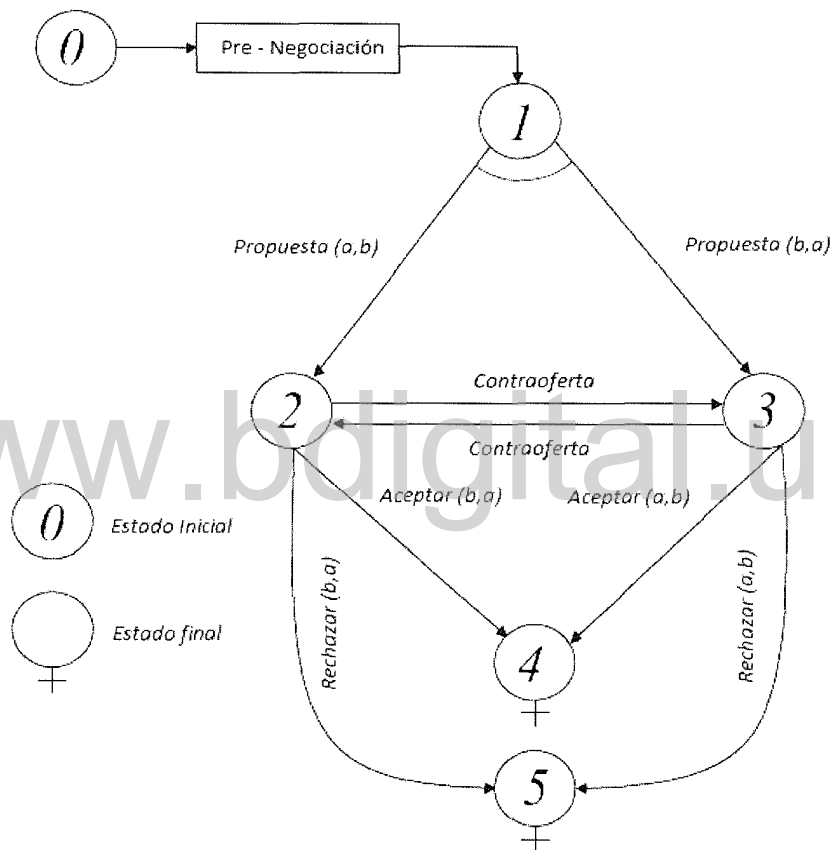


Figura C.2: Protocolo de negociación

C.3. PROPUESTA DE UN PROTOCOLO DE NEGOCIACIÓN PARA UN ENTORNO MULTIAGENTE EN PROCESOS CONTINUOS

cada elemento j tal que $U_j^i : D_j^i \rightarrow [0, 1]$.

La manera tradicional de definir a U_j^i es a través de funciones lineales. Supongase que el agente a (agente que inicia la negociación) solicita negociar el incremento del valor de la consigna de operación de un agente b , entonces la función de utilidad que va a permitir al agente a valorar alguna oferta propuesta por el agente b se definiría como:

$$U_j^a = \frac{j_b - \min_j^a}{\max_j^a - \min_j^a} \quad (C.3)$$

En donde \max_j^a es el valor inicial de la oferta sobre j (valor máximo) que solicita el agente a negociar, \min_j^a es el valor mínimo aceptable de la oferta sobre j que aceptaría el agente a y j_b es la respuesta del agente b sobre la oferta enviada por el agente a ; se asume que $j_b > \min_j^a$, de lo contrario $U_j^a = 0$. Ahora bien, la función de utilidad del agente b , que permitiría valorar la ofertas recibidas del agente a quedaría definida de la siguiente forma:

$$U_j^b = 1 - \frac{j_a - \min_j^b}{\max_j^b - \min_j^b} \quad (C.4)$$

En donde \max_j^b es el valor máximo que aceptaría el agente b (contraoferta) sobre j , \min_j^b es el valor mínimo que aceptaría el agente b sobre j y j_a es la oferta que envía el agente a al agente b , se asume que $\min_j^b \leq j_a \leq \max_j^b$, de lo contrario $U_j^b = 0$.

El vector de valores propuesto por un agente a a un agente b en el tiempo t es representado por $x_{a \rightarrow b}^t$ y específicamente el valor para el elemento j se denota como $x_{a \rightarrow b}^t[j]$. A partir de esto se define el proceso de negociación entre agentes, $a, b \in \text{Agentes}$, en el tiempo t_n (tiempo de negociación), denotado como $x_{a \rightarrow b}^{t_n}$, como una secuencia finita n de la forma $(x_{a \rightarrow b}^{t_1}, x_{a \rightarrow b}^{t_2}, \dots)$ con $t_1, t_2, \dots, \leq t_n$, en donde:

- $t_{i+1} > t_i$, la secuencia está ordenada en el tiempo.
- Para cada elemento j , $x_{a \rightarrow b}^{t_i}[j] \in D_j^a$, donde $D_j^a = [\min_j^a, \max_j^a]$

Un elemento de negociación se encuentra activo en el tiempo t_n , si $Ultimo(X_{a \rightarrow b}^{t_n}) \notin \{\text{aceptat}, \text{rechazar}\}$, donde, $Ultimo()$ es una función que retorna el último elemento en una secuencia.

En general, los agentes generan ofertas y contraofertas a través de combinacionales lineales de funciones simples denominadas tácticas. Las tácticas generan

una oferta o una contraoferta para un simple elemento usando un criterio sencillo (tiempo, recursos o el comportamiento de otros agentes).

De acuerdo a lo anterior, si el agente a solicita negociar el cambio de valor sobre el parámetro j del agente b , entonces dicho valor es negociado en el tiempo t , $0 \leq t \leq t_{max}$, a través de la función $\phi_j^a(t)$, por medio de la siguiente expresión:

$$x_{a \rightarrow b}^t[j] = \min_j^a + \phi_j^a(t)(\max_j^a - \min_j^a) \quad (C.5)$$

La ecuación 6.5 es válida si U_j^a tiende a disminuir. Si U_j^a tiende a aumentar entonces $x_{a \rightarrow b}^t[j]$ quedaría definida como:

$$x_{a \rightarrow b}^t[j] = \min_j^a + (1 - \phi_j^a(t))(\max_j^a - \min_j^a) \quad (C.6)$$

En general una amplia gama de funciones dependientes del tiempo pueden ser definidas simplemente variando la forma en que $\phi_j^a(t)$ es calculada.

C.4. Entorno multiagente para la supervisión de procesos continuos

Es importante destacar, que en el desarrollo inicial de los sistemas multiagentes, los diseñadores se centraron en el estudio del agente, es decir, en la estructura interna del mismo y en su comportamiento. Las organizaciones, como mucho, emergían de las interacciones de los agentes (Boissier 2007), como por ejemplo con los protocolos tipo “ContraNet” o la formación de coaliciones de dependencia. Sin embargo, los métodos de análisis y diseño de sistemas multiagentes no consideraban a las organizaciones como entidades propias, ni tampoco los agentes los trataban como conceptos sobre los que razonar. En realidad, los agentes eran vistos como entidades autónomas y dinámicas que evolucionaban en función de sus propios objetivos, sin que existieran restricciones explícitas externas sobre sus comportamientos.

En los últimos años, el interés de los desarrolladores se ha dirigido a los aspectos organizacionales de la sociedad de agentes (Ferber and Gutknech 1998)(Argente 2008). De este modo, el proceso de desarrollo del sistema se guía por los conceptos de organización, grupo, normas, roles, funciones, etc. Los agentes son vistos ahora como entidades dinámicas que evolucionan dentro de las organizaciones, las cuales restringen sus comportamientos y son también el resultado de las propias actividades de los agentes.

Tomando las premisas mencionadas, se puede afirmar que en general, de un sistema multiagente debe proporcionar los modelos y métodos que describen los conceptos claves del sistema a implementar. Así pues, el diseño debe:

- Identificar el proceso de desarrollo y ejecución de los agentes tanto individualmente como en conjunto.
- Definir como se trata situaciones imprevistas.
- Identificar cuales son los elementos sobre las cuales un agente toma sus decisiones que permiten lograr sus objetivos individuales.
- Como tratar la coordinación entre los agentes que componen el sistema y los agentes externos a él.

Se ha decidido, en forma general, usar los lineamientos de la metodología INGENIAS(Gomez 2002)(Gomez 2010). Esta metodología concibe el SMA como la representación computacional de un conjunto de modelos. Cada uno de estos modelos muestra una visión parcial del sistema: los agentes que lo componen, las interacciones que existen entre ellos, como se organizan para proporcionar la funcionalidad del sistema, que información es relevante en el dominio y como es el entorno en el que se ubica el sistema a desarrollar.

La organización de los modelos se realiza mediante la utilización de metamodelos. Un metamodelo es una representación de los tipos de entidades que pueden existir en un modelo, sus relaciones y restricciones de aplicación. Se elaborarán 5 modelos (desde el punto de vista descriptivo sin entrar en detalle en la elaboración de diagramas UML) basados en los metamodelos propuestos por la metodología INGENIAS: el modelo de la organización, el modelo del entorno, el modelo de tareas y objetivos, el modelo de agentes y el modelo de las interacciones. Se aclara que el propósito es tener una referencia con la cual diseñar el sistema multiagente que se propone. Desde luego que INGENIAS, es una metodología formal para el desarrollo de aplicaciones software para sistemas multiagente que sigue el proceso de RUP (Rational Unified Process).

C.4.1. Modelo de la organización

El modelo de la organización define el marco en el que el sistema multiagente existe. En el se describen la estructura de la organización, la funcionalidad y las relaciones sociales entre los agentes. En este modelo también se define la arquitectura del sistema y los diferentes agentes que lo componen. En la figura C.3

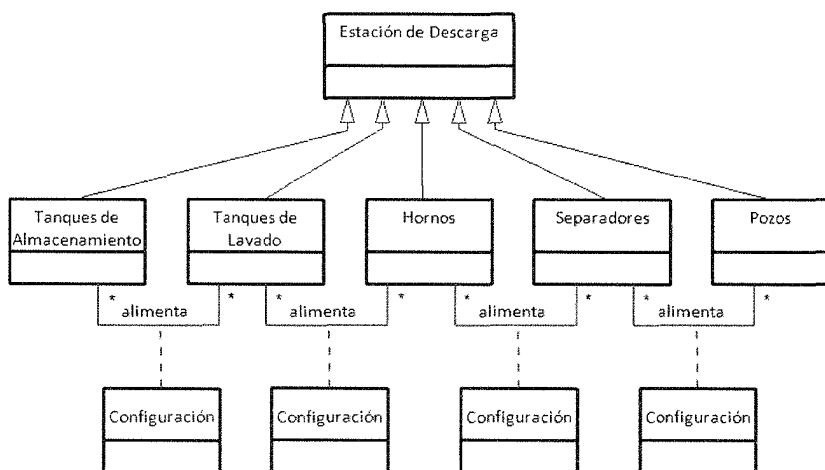


Figura C.3: Modelo de la organización de una estación de descarga de petróleo

se muestra un esquemático de la organización de una estación de flujo de procesamiento de petróleo.

Para el caso de una estación de descarga para procesamiento del petróleo, la organización del sistema estaría compuesta de los siguientes agentes: agente para cada pozo de petróleo, agente para cada separador, agente para cada horno, agente para cada tanque de Lavado, agente para cada tanque de almacenamiento, el agente supervisor global y el agente de configuración. La cantidad de pozos, separadores, hornos, tanques, etc., dependerá de la topología física de la red cuya información está registrada en el agente de configuración. El agente de configuración y el agente supervisor global son los únicos que tienen una sola instancia. El resto de los agentes presentan varias instancias dependiendo de la cantidad de pozos, separadores hornos, etc., que existan en la red de producción.

El agente configuración contiene la configuración de topología física de la red de producción y se encarga de notificar a cada agente pozo con cual separador se va a comunicar. Análogamente, cada agente separador debe saber con cual agente horno se va a comunicar y así sucesivamente hasta llegar a la última fase del procesamiento del producto.

En general, tomando como base la figura C.4, se puede afirmar que la arquitectura del sistema multiagente propuesto es de tipo jerárquica debido a la forma natural en que las redes de producción está organizadas. Es importante aclarar que cada agente tiene su propia autonomía.

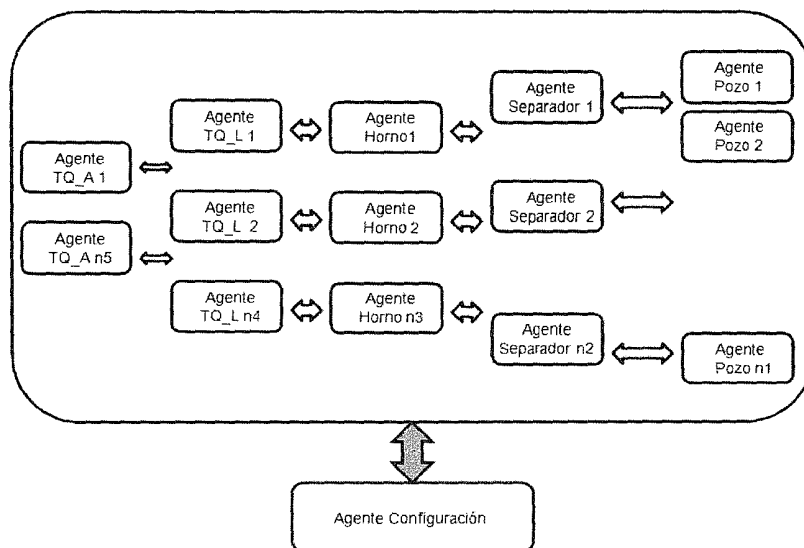


Figura C.4: Arquitectura del sistema multiagente propuesto

www.bdigital.ula.ve

C.4.2. Modelo del entorno

El modelo del entorno identifica el conjunto de elementos que permiten al agente percibir como se encuentra el estado de dominio del sistema. La determinación de estos elementos o entidades es imprescindible para el correcto funcionamiento del sistema ya que describe los elementos externos con los que el sistema interactúa. Las entidades existentes en el entorno del sistema multiagente propuesto contemplan los siguientes elementos:

- La instrumentación de campo: esto contempla los sensores y transmisores que están conectados a las tarjetas de entradas analógica o tarjetas de comunicación de los PLC. Adicionalmente se incluyen los actuadores de las válvulas de control.
- Los controladores lógicos programables (PLC): que es en donde justamente residen los agentes de la propuesta. Existen dos alternativas al respecto: un agente puede ser implementado en un PLC como es el caso de cada horno o varios agentes pueden ser implementados en un PLC como es el caso de los separadores de producción en donde normalmente está instalado un PLC para todas unidades existentes.

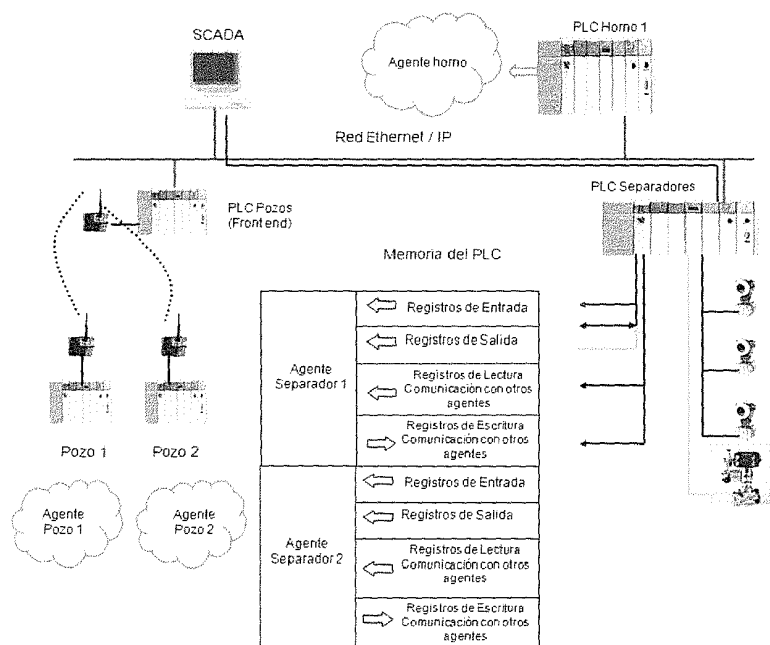


Figura C.5: Entorno del sistema multiagente

- El sistema de control y adquisición de datos (SCADA): a través de éste, el operador no solamente puede visualizar toda la información del proceso de producción sino que también establece los objetivos de producción al cambiar las consignas de operación en algunas unidades de producción.

En la figura C.5, se muestra un esquemático que muestra en el entorno del sistema multiagente. Tal figura, muestra la arquitectura típica de una estación de descarga automatizada de la empresa petrolera venezolana PDVSA. En la estación propiamente como tal, existe una red Ethernet/IP compuesta por PLC (separadores, horno, pozo) y la aplicación SCADA.

C.4.3. Modelo de tareas y objetivos

Este modelo permite modelar el control de cada agente, describiendo cuando se puede ejecutar las tareas, que cambios sobre el estado mental del agente se producen y cual es la motivación del agente para ejecutar dichas tareas. Este modelo contempla la descomposición de objetivos en sub-objetivos y tareas en sub-tareas hasta donde sea posible.

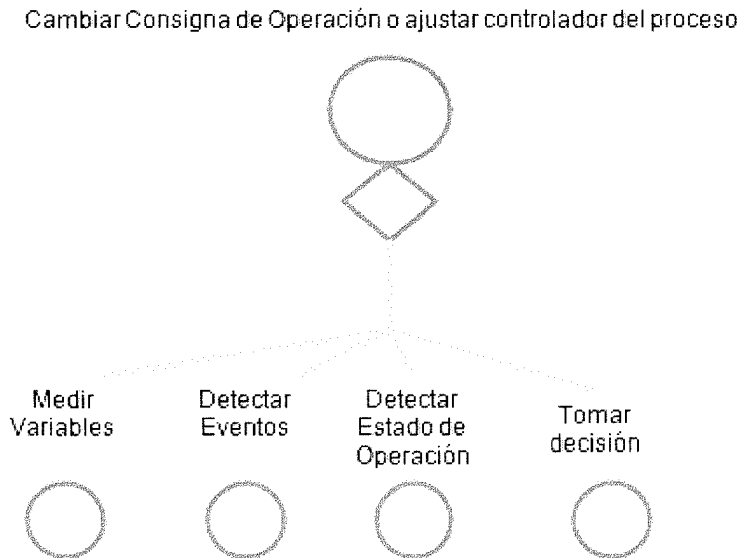


Figura C.6: Modelo de objetivos agente especializado

En la figura C.6, se muestra el objetivo y los sub-objetivos de un agente especializado (agente supervisor de cualquier planta).

C.4.4. Modelo del agente

En el modelo de Agente de INGENIAS se describen los agentes particulares, excluyendo sus interacciones con otros agentes, centrándose en su funcionamiento y en el diseño de su control. La funcionalidad estudia las responsabilidades del agente, identificando y definiendo las tareas y las acciones que es capaz de ejecutar.

El control del agente está enfocado al comportamiento durante su ejecución que permitirá el desarrollo de las tareas adecuadas para alcanzar el objetivo. El comportamiento está relacionado con el estado mental, con los objetivos, creencias, hechos y compromisos que el agente haya adquirido. A continuación, se explicara brevemente el funcionamiento de cada agente supervisor que se ha dado la denominación de agente especializado.

Agente Pozo: se encarga del control y la supervisión de los parámetros operacionales de un pozo. En particular, se controla la velocidad de operación de

cada pozo. El objetivo del agente es detectar los eventos que determinen ciertas condiciones óptimas de operación de tal manera de aumentar la velocidad de extracción de la bomba o disminuir la velocidad de ésta si se detectan condiciones adversas de operación. En concreto este agente suministra el envío de la materia prima a la estación de descarga. Dicha materia prima es recibida por los separadores que representan la segunda fase del procesamiento de la producción.

En cuanto a la arquitectura del agente, de acuerdo a su comportamiento, se pudiera catalogar como un agente tanto deliberativo como reactivo. En definitiva, se trata de un agente híbrido. Por otro lado, de acuerdo su organización interna se puede clasificar como una arquitectura modular ya que esta compuesto de un módulo de control y un módulo de supervisión.

Agente Separador: partiendo de las premisas de diseño establecidas en el capítulo 5, éste agente se encarga de detectar desviaciones en el punto de operación actual del separador producto de la variación del volumen y/o la cantidad del crudo recibido por los pozos. En base a estas variaciones, el agente toma la decisión de ajustar la consigna de operación o de ajustar los parámetros escalares del controlador que permita reducir el error de estado estático. El objetivo es mantener un nivel de crudo idóneo en el separador de tal manera que la eficiencia de separación se mantengan en el valor más alto posible.

Agente Horno: partiendo de las premisas de diseño establecidas en el capítulo 5, éste se encarga de detectar desviaciones en el punto de operación actual del horno debido a variaciones en la calidad y / o cantidad de crudo proveniente de cada separador. Analogamente al caso del agente separador, éste agente toma la decisión de ajustar la consigna de operación y / o ajustar el controlador. El objetivo del agente es mantener la temperatura del crudo en un rango de valores tal que en el tanque de lavado se pueda separar con facilidad el componente del agua del petróleo.

Agente Tanque de Lavado y Agente Horno: ambos agentes en cuanto a su estructura están basados en los mismos elementos que componen al Agente Separador y al Agente Horno. El objetivo del Agente Tanque de Lavado es controlar y supervisar que el nivel del tanque de lavado sea lo más óptimo posible para que el agua se separe completamente del petróleo y en cuanto al Agente Tanque de Almacenamiento.

Agente Configuración: finalmente el Agente Configuración se encarga de registrar la configuración de la red de producción por lo que la principal función de éste agente es lograr que cada Agente Especializado conozca con quien va a

interactuar directamente. Este agente en esencia permitirá controlar la entrada y salida de agentes dentro de la red de configuración.

C.4.5. Modelo de interacciones

Como parte de la ejecución de las tareas existe la necesidad de que diversos agentes interactuen entre sí para el intercambio de la información entre ellos y es por ello que las interacciones entre agentes se especifica a través del Modelo de Interacción. La interacción entre agentes se especifica mediante modelos de interacción. En un Modelo de Interacción se especifican los agentes iniciadores de una interacción y los agentes que colaboran en la misma.

Las interacciones entre agentes son parte de la ejecución de una tarea dentro de un flujo de trabajo. En nuestro caso particular existen tres tipos de interacciones fundamentales: (i) Interacciones en el Agente de Configuración y cualquier Agente Especializado, (ii) Interacción entre Agentes Especializados para informar el estado de operación de cada sub-unidad de producción y (iii) Interacción entre dos agentes especializados con la necesidad de establecer un proceso de negociación en cuanto a la asignación del valor de una consigna de operación; punto que fue tratado en la sección anterior.

Estas tres interacciones se pueden sintetizar en los tres siguientes modelos de interacciones: *Notificación del estado de operación del proceso*, *Obtención de red de dependencia* y *Negociación el establecimiento de una consigna de operación*.

En el modelo *Notificación del estado de operación del proceso*, un agente especializado productor asociado a una unidad de producción notifica el estado de operación actual a un agente especializado consumidor. Sucede que éste último agente especializado se convierte en un agente productor de otro agente especializado consumidor y así sucesivamente hasta llegar a la fase final de procesamiento del crudo. En principio cada agente productor envía los datos asociados a estado de operación actual y el evento. Luego si y solo si ocurre algún cambio de evento, se vuelve a enviar la información sobre el estado de operación junto al cambio de evento.

En el modelo *Obtención de red de dependencia*, el agente configuración notifica una vez a cada agente especializado la dirección del agente con quien debe establecer la interacción para notificar el estado del proceso. Si después de cierto tiempo se incorpora o se elimina algún agente especializado (Por ejemplo, cuando se realizan operaciones de mantenimiento mayor a un separador), el agente con-

figuración procede a actualizar la red de producción. Con esto se pretende ofrecer ciertas características de reconfigurabilidad dentro del SMA propuesto.

Finalmente, el modelo de *Negociación del establecimiento de una consigna de operación*, un agente consumidor solicita iniciar un proceso de negociación con el agente especializado productor. Los detalles sobre este proceso de negociación fueron estudiados en el punto anterior. En el ámbito industrial, existen varios protocolos de comunicación usados por una inmensa cantidad de empresas que fabrican equipos de PLC como son el protocolo ControlNet, Ethernet/IP, Datahighway (DH), etc.

C.5. Implementación del sistema multiagente

Por una parte, se ha implementado un prototipo del sistema multiagente en la plataforma de controladores lógicos programables de la familia PLC Control Logix fabricado por la Empresa Rockwell Automation. Un resumen sobre este desarrollo está en el apéndice D. La implementación de los agentes en un PLC tuvo como objetivo mostrar el funcionamiento del mecanismo de negociación de los agentes encargados de supervisar cada una de las unidades de producción en una empresa de procesamiento de petróleo.

Por otro lado, también se realizaron simulaciones computacionales en MATLAB / SIMULINK, usando un elemento clave llamado bloque de función S (S-function) tal y como se desarrollo en (Ramirez, Davila, and Colina 2009). Este bloque de función S es configurable tanto en lenguaje m, en lenguaje C y lenguaje C++ y permiten incorporar algoritmos personalizados dentro del ambiente SIMULINK facilitando la creación de esquemas de simulación de alta complejidad.

Adicionalmente se configuró una interfaz gráfica usando GUIDE que es un entorno de programación visual disponible en MATLAB. Para este caso particular, la intención de usar GUIDE fue con el propósito de interactuar con el archivo de SIMULINK, que contiene el esquema de supervisión basado en SMA, de una manera más amigable.

Para probar el esquema de supervisión propuesto, a través del esquema de simulación usando MATLAB / SIMULINK, se definió primeramente la estructura de la red de producción, la cual, está conformada por dos separadores de producción conectados a un horno. Luego se definió un escenario de ocurrencia de

C.5. IMPLEMENTACIÓN DEL SISTEMA MULTIAGENTE

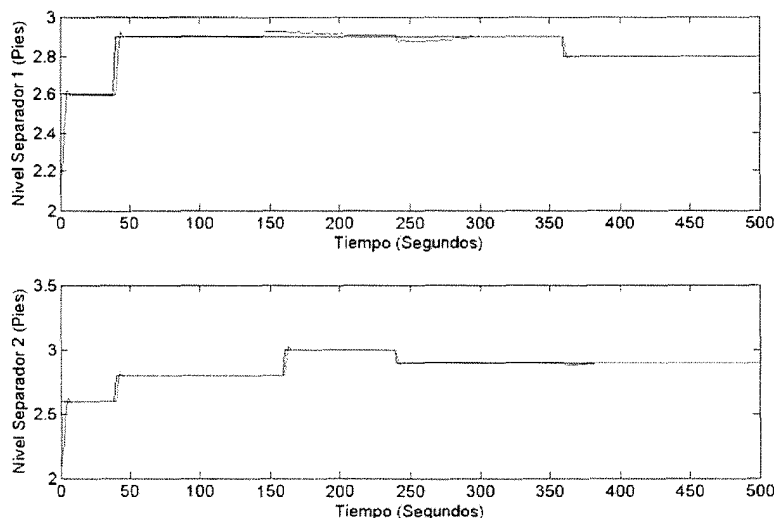


Figura C.7: Salida agente separador 1 y 2

eventos de la siguiente forma:

- En $t = 150$ segundos se introduce simultáneamente en los dos separadores un incremento significativo del creó, tal y como se muestra en la figura C.7.
- En el proceso de calentamiento no se introdujó ningún cambio, es decir, el estado de operación es normal.

El resultado de la simulación muestra que cuando los agentes están intercambiando información continuamente es posible tomar decisiones más oportuna que mantengan las condiciones del proceso con la menor cantidad de fluctuaciones. La

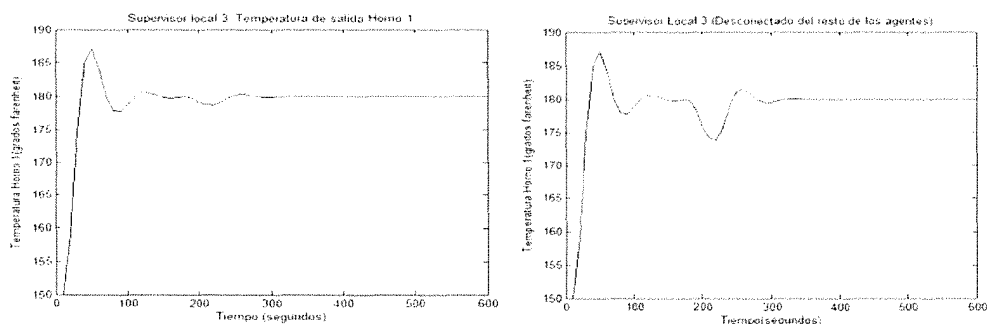


Figura C.8: Salida agente horno 1

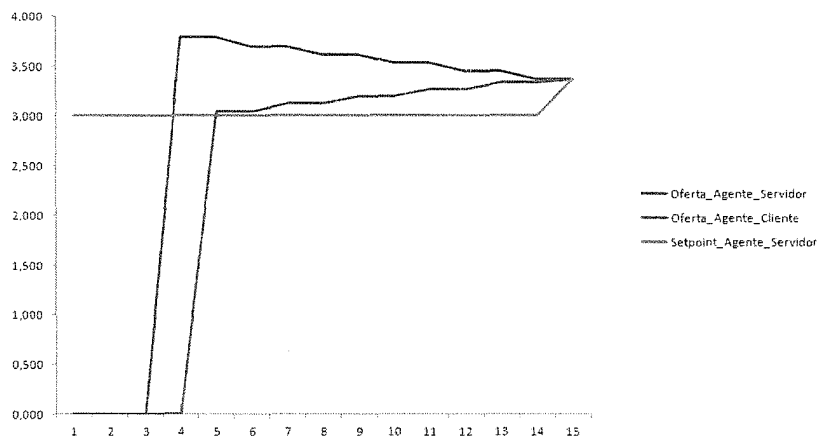


Figura C.9: Simulación de un proceso de negociación

salida del agente horno 1 que está a la derecha en la figura C.8 ocurre cuando los agentes separadores continuamente notifican el estado de operación del proceso y el evento actual. En cambio, la gráfica de la izquierda muestra el comportamiento del sistema supervisor del horno aislado.

En cuanto a las simulaciones realizadas usando controladores lógicos programables, se implementó un proceso de negociación basado en la teoría de juegos. Se definió el siguiente escenario: un agente separador tiene su setpoint en 3,0 pies y un agente horno solicita negociar el cambio de tal consigna a 3,8 pies. Se estableció como tiempo de negociación 20 segundos. La gráfica de la figura C.9 muestra el resultado del proceso de negociación, en donde, se aprecia el acuerdo final al que ambos deciden fijar el valor de la consigna de operación cuyo valor es 3,37 pies.

Apéndice D

Implementación del algoritmo de negociación en controladores lógicos programables

En esta sección del apéndice, se describirá brevemente la estructura, de algunos rungs (peldaño) de programación en PLC, que ejecuta el modelo negociación descrito en el capítulo 6. La plataforma que estuvo disponible para realizar estas pruebas fue la del PLC-5 marca ALLEN-BRADLEY a través de la implementación de una red Datahighway.

En la figura C.1, se muestra el primer rung del programa del PLC Agente A que es el que inicia el proceso de negociación en el cual el bit B3:0/0 activa la instrucción de tipo mensaje MSG (de tipo escritura) tal y como se muestra en dicha figura. Esta instrucción escribe el valor de un bit desde el PLC Agente A hasta el PLC Agente B usando el puerto de comunicación DH(Datahighway de los PLC), y en caso de que la comunicación sea exitosa se activaría el bit MG11:0.DN que a su vez desactivaría el bit B3:0/0, de lo contrario, se activa el bit MG11:0.ER (Error de comunicación) y se ejecuta tres reintentos antes de declarar la falla de comunicación a través del bit C5:0.DN.

En la figura C.2, se muestra el segundo y tercer rung. El segundo rung, el agente A cuenta un segundo y envía las condiciones de negociación que en este caso es simplemente el tiempo de duración del proceso de negociación (Fase de pre-negociación). El agente B, al aceptar esta condición envía un mensaje al agente A y se activa el bit MG11:1.DN, correspondiente al tercer rung. Este bit cuenta un segundo y envía la primera oferta del agente A.

Una vez enviada la primera oferta, se empieza a contabilizar el tiempo de nego-

APÉNDICE D. IMPLEMENTACIÓN DEL ALGORITMO DE NEGOCIACIÓN EN CONTROLADORES LÓGICOS PROGRAMABLES

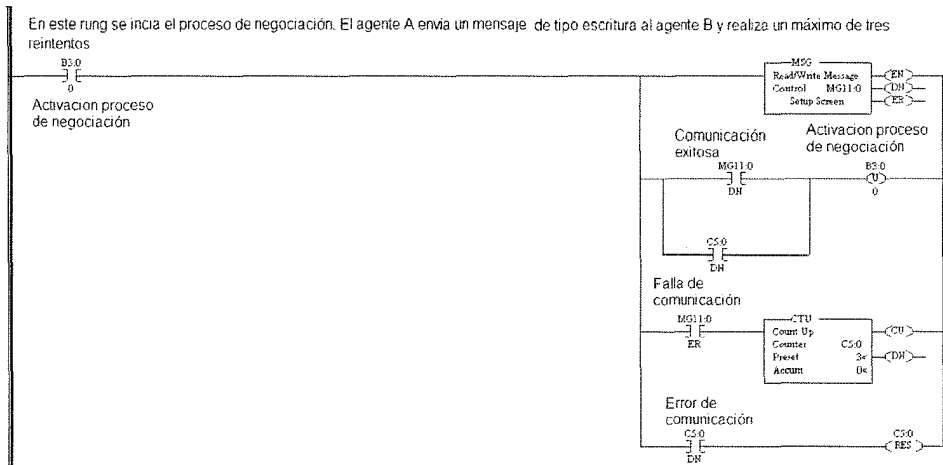


Figura D.1: Rung que activa el proceso de negociación (Agente A)

www.bdigital.ula.ve

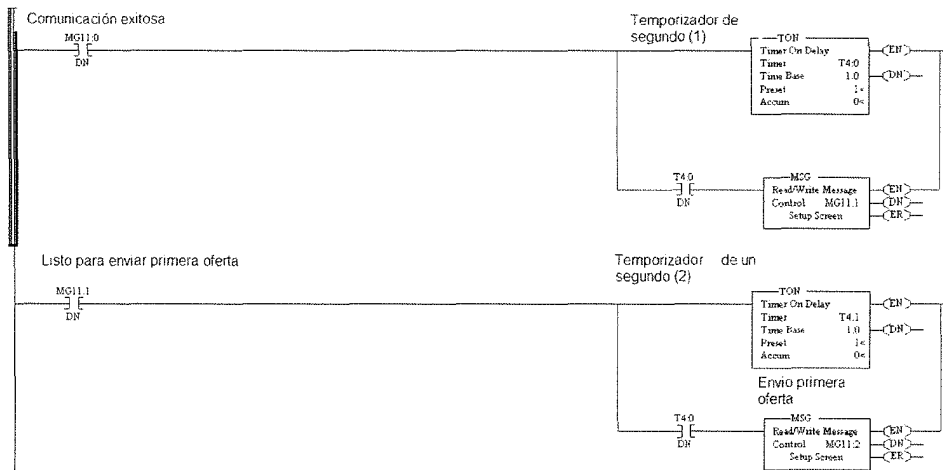


Figura D.2: Rungs que establece las condiciones de negociación y envía primera oferta

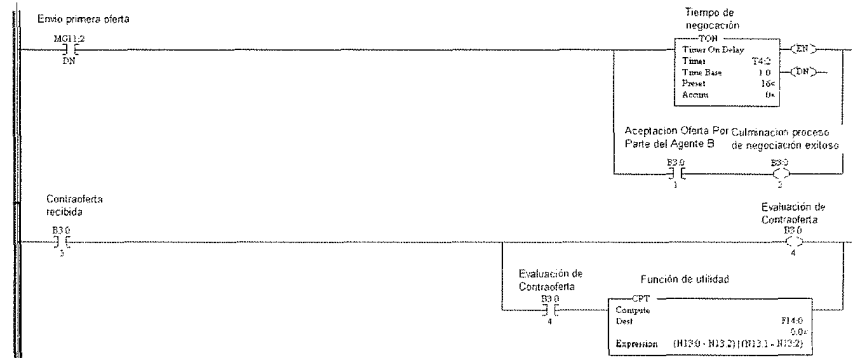


Figura D.3: Rungs que determina si la oferta es aceptada o si evalúa una contraoferta

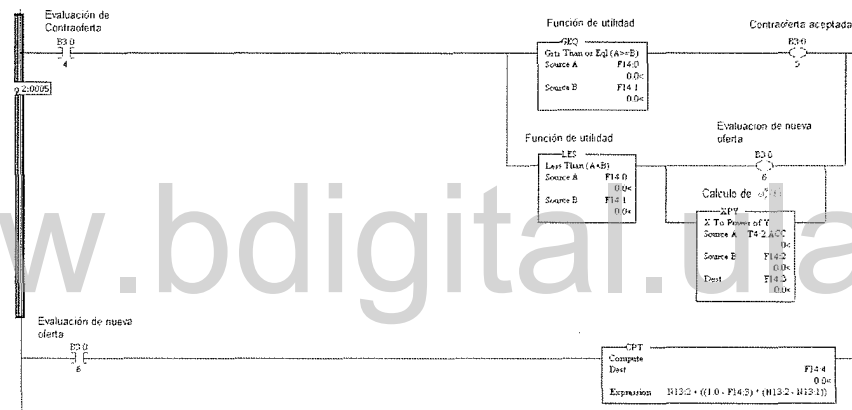


Figura D.4: Rungs que determina si acepta contraoferta o envía una nueva oferta

ciación establecido lo cual se lleva a cabo en el cuarto rung mostrado en la figura C.3. En este rung, se espera por la respuesta del agente B. Si el agente B, luego de evaluar la oferta recibida por el agente A, la acepta entonces envía un mensaje al agente A lo cual se refleja en la activación del bit B3:0/1 que a su vez activa al bit B3:0/2 que indica la culminación exitosa del proceso de negociación.

En caso de que el agente B no este de acuerdo con la oferta enviada por el agente A, se envía una contraoferta que consiste en la activación de un bit y el valor de la nueva propuesta. Esta contraoferta es analizada usando la función de utilidad descrita en la ecuación 6.3., y si esta dentro del rango de valoración, el agente A envía un mensaje al agente B de aceptación y culmina el proceso de negociación. De lo contrario, se procede a calcular una nueva oferta, a través de la ecuación 6.5 y se envía al agente B y así sucesivamente hasta lograr llegar a

APÉNDICE D. IMPLEMENTACIÓN DEL ALGORITMO DE
NEGOCIACIÓN EN CONTROLADORES LÓGICOS PROGRAMABLES

un acuerdo. Si transcurrido el tiempo de negociación, no se ha llegado a ningún acuerdo, se culmina el proceso de negociación sin éxito alguno.

www.bdigital.ula.ve

Referencias

- Abad, P. (2002). Aplicación de herramientas de aprendizaje supervisado a la diagnosis de sistemas dinámicos. *Tesis doctoral, Universidad de Sevilla. Departamento de lenguaje y sistemas informáticos.*
- Abonji, J. (2007). Fuzzy model identification for control. *Birkhauser Boston Springer - Verlag New York.*
- Abonyi, J., L. Andersen, L. Nagy, and F. Szeifert (1999). Inverse fuzzy process model based direct adaptive control. *Mathematics and Computers in Simulation, 51(1-2):119-132.*
- Abonyi, J., R. Babuska, and F. Szeifert (2001). Modified gath geva fuzzy clustering for identification of takagi sugeno models. *University of Vezprem, Hungary.*
- Ahvenlampi, T. and J. Diez (2003). New methods for validation of local models in fuzzy clustering identification. *Proceedings of IFAC Conference On Intelligent Control and Signal Processing, pp9-15.*
- Alexandres, M. (2007). Arquitectura tolerante a fallos mediante un sistema multiagente para el sistema de control de un robot movil. *Tesis Doctoral, Universidad Politecnica de Valencia, España.*
- Alonso, J. and L. Magadalena (2008). Equilibrio entre interpretabilidad y precisión en sistemas basados en reglas. *Centro Europeo de Soft Computing, Edificio Científico Tecnológico, Asturias, España.*
- Amigoni, F. and M. Solmalvico (2003). A multiagent system for enviromental perception. *Proceedings of the 3rd Conference on Artificial Intelligence Applications to the Environmental Science. American Meteorological Society.*
- Andersen, H., A. Lotfi, and A. Tsoi (1997). A new approach to adaptive fuzzy control: The controller output error. *IEEE Transactions on Systems, Man, and Cybernetics.*
- Argente, E. (2008). Gormas: Guías para el desarrollo de sistemas multiagentes basados en organizaciones. *Tesis Doctoral, Universidad Politécnica de Valencia, España.*

REFERENCIAS

- Arva, P. and S. Nemeth (1991). Learning algorithm in fuzzy control. *Annales University Budapest*, 12:25-32.
- Astrom, K. and B. Wittermark (1989). *Adaptive Control*. (2 ed.).
- Aznarte, J. (2008). Modelling times series through fuzzy rules based models: A statistical approach. *Tesis doctoral, Universidad de Granada, departamento de Ciencias de la Computación e Inteligencia Artificial, España.*
- Babuska, R. (1996). Fuzzy modeling and identification, phd dissertation. *Delft University of Technology, The Netherland.*
- Barragan, A. (2009). Síntesis de sistemas de control borroso estables por diseño. *Tesis doctoral, Universidad de Huelva, España.*
- Batur, B. and D. Celad (1993). Inverse fuzzy model controllers. *IEEE Transactions on System and Control.*
- Bedia, M., C. J., E. Corchado, and C. Fyfe (2002). Analytical model for constructing deliberative agents. *Engineering Intelligent Systems, Vol 3: 173-185.*
- Bezdek, J. (1973). Fuzzy mathematics in patten classification, phd dissertation. *Delft University of Technology, The Netherland.*
- Bezdek, J. (1987). Patter recognition with fuzzy objective function algorithms. *Plenum Press, New York.*
- Blanke, M. and M. Kinnaert (2003). Diagnosis and fault-tolerant control. *Springer Verlag, Germany.*
- Blanke, M., M. Staroswiecki, and N. Wu (2001). Concepts and methods in fault-tolerant control. *Proceedings of the 9 TH American Control Conference, Vol 4, 2606-2620.*
- Bodenhofer, U. and P. Bauer (2000). Towards an axiomatic treatment of interpretability. *In proceedings of IIZUKA2000, pages 334-339, Iizuka.*
- Bodenhofer, U. and P. Bauer (2003). A formal model of interpretability of linguistic variables. *Studies in fuziness and Soft Computing, Srpinger Verlag, Heidelberg.*
- Boissier, O. (2007). Organization oriented programming from closed to open organizations. *In proceedings of the Engineering Societies in the Agents World VI, Sixth International Workshop, ESAW06, Lecture Notes in computer science.*
- Bongaerts, L., L. Monostori, D. McFarlane, and B. Kadar (2002). Basic principles of hms architecture. *In proceedings HMI International Symposium.*

- Busch, L. and I. Horstmann (1999). Signaling via an agenda in multi issue bargaining with incomplete information. *Economy Theroy 13(3), April.*
- Camacho, E. and C. Bordons (1995). Model predictive control in the process industry. *Springer Verlag, Berlin, Germany.*
- Chan, L., N. Narasbadi, and V. Mirelli (1996). Multi-stage target recognition using modular vector quantizers and multilayer perceptron. *Proceedings IEEE Conference in Computer Vision and Pattern Recognition*, 114–119.
- Chang, K., V. Lee, and H. Leung (1997). Generating fuzzy rules for target tracking using steady - state genetic algorithm. *IEEE Transaction on Evolutionary Computation, Vol 1*, 189–200.
- Cheng, C. and C. Wong (1995). Self-generating rule mapping fuzzy controller design using genetic algorithm. *IEEE Transactions on Fuzzy Systems, Vol 3*, 129–139.
- Christensen, J. (1994). Holonic manufacturing system - initial architecture and standard directions. *First Euro Conference on HMS, Hannover, Germany.*
- Christensen, J. (2002). Basic principles of hms architecture. *In proceedings HMI International Symposium.*
- Colina, E., E. Altamiranda, and E. Chacon (2005). Intelligent decision making support aid for control systems. *Revista Colombiana de Tecnología de Avanzadas ISSN: 1692-7257 Volumen 1 Número 5 año - 2005 Universidad de Pamplona, Colombia, 2005.*
- Conry, S. (1998). Multistage negotiation in distributed artificial intelligence. *Foundations of Distributed Artificial Intelligence, Wiley-Interscience Publication.*
- Contreras, J., R. Misa, and L. Urueta (2007). Algoritmos para identificación de modelos difusos interpretables. *IEEE Latin American Transactions, Vol 5, Número 5*, 2–4.
- Davis, L. (1991). *The Handbook of Genetic Algorithm.* (2 ed.).
- Dekleer, J., A. Mackworth, and R. Reiter (1992). Characterizing diagnoses and systems artificial intelligence.
- Delgado, M., F. Gomez, and M. F. (1997). A fuzzy clustering-based rapid prototyping for fuzzy rule-based modelling. *IEEE Transaction on Fuzzy System*, 223–233.
- Denai, M. and F. Palis (2007). Modeling and control of nonlinear systems using soft computing techniques. *Applied soft computing Journal, Vol7, pp728-738.*

REFERENCIAS

- Diez, J. (2003). Técnicas de agrupamiento para identificación y control por modelos locales. *Tesis Doctoral, Universidad Politecnica de Valencia España*.
- Diez, J. and J. Navarro (1999). Fuzzy models of complex systems by means of clustering techniques. *Universidad Politecnica de Valencia España*.
- Driankov, D., H. Hellendoorn, and M. Reinfrank (1993). *An introduction to fuzzy control* (2 ed.).
- Economou, C., M. Morari, and B. Palsson (1986). Internal model control: Extension to nonlinear systems. *Chemical Engineering science*.
- Efrain, G. (2001). Detectando fallas mediante redundancia analítica. *Ingeniería, Vol 4, No 11*.
- Eigi, M., E. Villani, and N. Maruyama (2001). Modelling of hybrid supervisory systems using uml and petri nets. *8TH IEEE International Conference on Emerging Tecnologies and Factory Automation, Vol 2, 149-158*.
- Esteva, S. (2003). Modelling, control and supervision for a class of hybrid system. *Tesis de Maestria Universitat de Girona, España*.
- Farag, W., V. Quintana, and L.-T. G. (1998). A genetic-based neuro-fuzzy approach for modeling and control of dynamical systems. *IEEE Transactions On Neural Networks, vol 9, 756-767*.
- Farantin, P. (2000). Automated service negotiations between autonomous computational agents. *PhD Thesis, Department of Electronic Engineering, University of London*.
- Farantin, P., C. Sierra, and R. Jennings (1998). Negotiation decision functions for autonomous agents. *Robotics and autonomous systems 3-4(24):159-182*.
- Fatima, S., M. Wooldridge, and R. Jennings (2004a). An agenda based framework for multi issue negotiation. *Artificial Intelligence, 1(152):1-45*.
- Fatima, S., M. Wooldridge, and R. Jennings (2004b). Optimal negotiation strategies for agents with incomplete information. *In J.J. Meyer and M. Tambe, editors, Lecture Notes in Artificial Intelligence, volume 2333*.
- Ferber, J. and O. Gutknech (1998). Aalaadin: A meta model for the analysis and design of organization for multiagent systems. *In Proceedings of the 3rd International Conference of Multi-Agent Systems (ICMAS-98), 128135*.
- Fernandez, F. (2008). Aproximación funcional mediante redes de base radial, una alternativa para la predicción en el proceso de la reducción del mineral de la tecnología de caron de producción de níquel. *Tesis doctoral, departamento de arquitectura y tecnología de computadores, Universidad de Granada, España*.

- Ferreiro, R. (2005). Trends on modelling techniques applied on ship's propulsion system monitoring. *Journal of maritime research, Vol II, No1, pp 87-104*.
- Fischer, T. (1990). Batch control systems: design, application and instrumentation. *Instrument society of America*.
- Fisher, M., A. Fink, and O. Nelles (1999). Supervision of nonlinear adaptive controllers based on fuzzy models. *In IFAC World Congress, Volume Q, pages 335-340, Beijing, China*.
- Freidman, J. (1991). Multivariate adaptive regression splines. *Annal of statistical 19(1), 1-141*.
- Fudenberg, D., D. Levine, and J. Tirole (1985). Infinite horizon models of bargaining with one sided incomplete information. *Game Theoretic Models of Bargaining. University of Cambridge Press*.
- Garcia, C., D. Prett, and M. Morari (1989). Model predictive control: Theory and practice - a survey. *Automatica 25(3): 335-348, 1989*.
- Garcia, E. and M. Morari (1982). Internal model control: A unifying review and some new results. *Chemical Engineering science*.
- Gertler, J. (1998). Fault detection and diagnosis in engineering systems. *Editorial Marcel Dekker, Inc. Chapter 1,2*.
- Giampapa, J. and K. Sycara (2003). Conversational case-based planning for agent team coordination. *Robotics Institute, Carnegie Mellon University*.
- Giret, B., P. Infrán, P. Pastor, and U. Cernuzzi (2000). Método de desarrollo de sistemas multiagentes. *Departamento de Sistemas Informáticos y Computación, Universidad Politécnica de Valencia España*.
- Goldberg, D. (1989). *Genetic Algorithm in Search, Optimization and Machine Learning*. (1 ed.).
- Gomez, J. (2002). Modelado de sistemas multiagente. *Tesis Doctoral, Universidad Complutense de Madrid*.
- Gomez, P. (2010). Programación de la producción en un taller híbrido sujeto a incertidumbre: Arquitectura y algoritmos. aplicación a la industria cerámica. *Tesis doctoral, departamento de organizaciones de empresa, Universidad Politecnica de Valencia, España*.
- Gracia, I., J. Moreno, and M. Zapatero (2004). Proyecto inteligencia artificial: Sistema de reconocimiento por satélite. *Universidad Pontificia de Salamanca*.
- Grosz, B. and C. Sidner (1990). Plans for discourse. intentions for communication. *pp 417-444, MIT Press*.

REFERENCIAS

- Gruber, T. (2006). Towards principles for the desing of ontologies used for knowledge sharing. *In proceeding of the international Workshop on formal Ontology, Padova, Italy.*.
- Gustafson, D. and W. Kessel (1979). Fuzzy clustering with a fuzzy covariance matrix. *Proceedings of IEEE Conference on Decision and Control, San Diego USA.*
- Hang, C. and P. Parks (1973). Comparative studies of model reference adaptive control systems. *IEEE Transactions on Automatic Control.*
- Harsanyi, J. and R. Selten (1972). Organizational rules as an abstraction for the analysis and design of multiagent systems. *Management Science, 18(5), 80-106.*.
- Hathaway, R. and J. Bezdek (1991). Grouped coordinate minimization using newton's methods for inexact minimization in one vector coordinate. *Journal of Optimization Theory and Applications, 503-516.*
- Henson, M. (1998). *Nonlinear Model Predictive Control: Current status and Future directions* (3 ed.).
- Homaifar, A. and E. McCormick (2002). Simultaneous design of membership functions and rule sets for fuzzy controllers using genetic algorithm. *IEEE Proceedings on Control Theory Application, Vol 149, No 2, 129-139.*
- Isermann, R. (1998). On fuzzy logic applications for automatic control, supervision and fault diagnosis. *IEEE Transactions On Systems, Man and Cybernetics, Part A: Systems and Humans, 28(2), 221-235.*.
- Izadi, R. and M. Blanke (1999). A ship propulsion system as a benchmark for fault tolerant control. *Control Engineering Practice, 7(2), 227-239.*
- Jang, J. (1993). Anfis: Adaptive-network based inference systems. *IEEE Transactions On System and Cybernetics, 665-685.*
- Jang, J., C. Sun, and E. Mizutani (1997). *Neuro-fuzzy and Soft computing* (3 ed.).
- Jantzen, J. (1998). Fuzzy supervisory control. *Technical University of Denmark.*.
- Jennings, N. and M. Woolridge (1998). Agent tecnology: foundations, applications and market. *Springer - Verlang.*
- Jin, Y. (2000). Fuzzy modeling of high dimensional systems. *IEEE Transactions On Fuzzy Systems, 212-221.*
- Johnson, C. (1988). *Lectures On Adaptive Parameter Estimation.* (1 ed.).

- Johnson, C. and E. Tse (1978). Adaptive implementation of one-step-ahead optimal control via input matching. *IEEE Transactions on Automatic Control*, *AC-23(5)*:865-872..
- Jordan, M. and D. Rumelhart (1998). Fuzzy models: Supervised learning with a distal teacher. *Cognitive Science*, (16):307-354.
- Kasabov, N. and Q. Song (2002). Denfis: Dynamic evolving neural-fuzzy inference system and its application for times series prediction. *IEEE Transactions on Fuzzy System*, vol 10, 144-154.
- Keller, J. and H. Tahani (1992). Implementation of conjunctive and disjunctive fuzzy logic rules with neural networks. *International Journal of Approximate Reasoning*, vol 6, 221-240.
- Kim, E., S. Minkee, and M. Park (1997). A new approach to fuzzy modeling. *IEEE Transactions On Fuzzy Systems*, 328-337.
- Kinzel, J., F. Klawonn, and R. Kruse (1994). Modifications of genetics algorithms for designing and optimizing fuzzy controllers. *IEEE Conference on Evolutionary computation, Orlando, USA.*, 328-337.
- Klawonn, F. and A. Keller (1998). Fuzzy clustering and fuzzy rules. *Technical University of Braunschweig, Germany.*
- Klawonn, F. and K. R. (1995). Clustering methods in fuzzy control. *German.*
- Klawonn, F. and K. R. (1997). Constructing a fuzzy controller from data. *German.*
- Koestler, A. (1989). *The Ghost in the machine.* (1 ed.).
- Kraus, S. (2001). Strategic negotiation in multiagent environments. *MIT, Press.*
- Kraus, S., J. Winkelfeld, and G. Zlotkin (1995). Multiagent negotiation under time constraints. *Artificial Intelligence*, 2(75):297-345..
- Krishnapuram, R. and C. Freg (1992). Fitting an unknown number of lines and planes to image data through compatible cluster merging. *Patter recognition*, 385-400.
- Kruse, R., C. Doring, and M. Lesot (1998). Fundamentals of fuzzy clustering. *Department of knowledge processing and language engineering, University of Magdeburg, Germany.*
- Lakos, C. (1995). From coloured petri nets to objetc petri nets. *In proceedings of the Aplication and Theory of Petri Nets.*
- Lee, S. and T. Lee (1975). Fuzzy neural networks. *Mathematical Biosciences*, vol 23, 151-177.

REFERENCIAS

- Leonaritis, I. and S. Billings (1985). Input-output parametric models for non-linear systems. *International Journal of Control*, 303–324.
- Lin, C. and C. Chen (2008). Efficient self-evolving evolutionary learning for neuro-fuzzy inference systems. *IEEE Transactions On Fuzzy Systems*, vol 16, 1476–1490.
- Ljung, L. (1983). *Theory and Practice of Recursive Identification* (3 ed.).
- Ljung, L. (1999). *System Identification* (1 ed.).
- Lunze, J. (2000). Process supervision by means of qualitative models. *Technische Universität Hamburg-Harburg, Germany, December 2000*.
- Mandic, N. and E. Scharf (1995). Practical application of a heuristic fuzzy rule-based to the dynamic control of a robot arm. *IEEE Proceedings-G*.
- McILRaith, S., G. Biswass, and C. Clandy (1999). 10 th international workshop on principles of diagnosis.
- Michalewicz, Z. (1994). *Genetic Algorithm + Data Structure = Evolution Programs*. (1 ed.).
- Mitra, S. and S. Pal (1995). Fuzzy multi-layer perceptron, inferencing and rule generation. *IEEE Transactions on Neural Networks*, 683–697.
- Montmain, J. and S. Gentil (1999). Causal modeling for supervision. *Proceedings of the 1999 IEEE International Symposium on Intelligent Control / Intelligent Systems and Semiotics*, 231-236..
- Mouratidis, H., P. Giorgini, and G. Manson (2003). Modelling secure multiagent systems. *Proceedings of the Second International joint Conference on Autonomous Agents and Multiagent Systems AAMAS-03. Vol. ISBN:1-58113-683-8, ACM Press, Melbourne, Australia, pp. 859866*..
- Murilo, J. and R. Melo (2002). Projetando sistemas multiagentes em organizações empresariais gramado. *Simpósio Brasileiro de Engenharia de Software*.
- Nakamori, Y. and M. Ryoke (1994). Identification of fuzzy prediction models through hyperellipsoidal clustering. *IEEE Transactions on System, Man and Cybernetics*, 1153–1173.
- Nauck, D., F. Klawonn, and R. Kruse (1996). *Neural Network and Fuzzy System* (1 ed.).
- Nauck, D., F. Klawonn, and R. Kruse (1997). *Foundations of Neuro-Fuzzy Systems* (2 ed.).
- Nearchou, A. (1999). Adaptive navigation of autonomous vehicles using evolutionary algorithms. *Artificial Intelligence in Engineering, Vol 13*, 159–173.
- Osborne, J. and A. Rubinstein (1994). A course of game theory. *MIT, Press*..

- Pal, S. and S. Mitra (1992). Multilayer perceptron, fuzzy sets and classification. *IEEE Transactions on Neural Networks*, vol 23, 151–177.
- Parra, C. (2009). Modelado y simulación del control supervisorio para sistemas holónicos de producción continua. *Tesis doctoral, Facultad de Ingeniería, Universidad de los Andes, Merida, Venezuela.*
- Pasha, E., T. Razzaghnia, and T. Allaviranloo (2007). Fuzzy linear regression model with fuzzy entropy. *Applied Mathematics Sciences* 01.
- Pasino, K. and S. Yurkovich (1998). Fuzzy control. *Addinson-Wesley.*
- Pedrycz, W. (1990). Relevancy of fuzzy models. *Information Sciences* 52, 285–302.
- Pedrycz, W. (1993). *Fuzzy Control and Fuzzy System* (2 ed.).
- Perez, J., J. Serrano, and F. Trigue (2005). Decisión support for traffic management based on organizational and communicative multiagent abstractions. *Transportation Research Part C.*
- Posada, J. (2003). Arquitectura para el control de robots móviles mediante delegación de código y agentes. *Tesis doctoral, Departamento de Informática de Sistemas y Computadores, Universidad Politécnica de Valencia España.*
- Posthlehwaite, B. (1994). A model-based fuzzy controller. *Chemical Engineering Research and Design, Transactions on IChemE*, 72(A1):38-46.
- Posthlehwaite, B. (1997a). Building a model-based fuzzy controller. *Fuzzy sets and systems*, 79:3-13.
- Posthlehwaite, B. (1997b). A new identification algorithm for fuzzy relational models and its applications in model based control. *Chemical Engineering Research and Design, Transactions on IChemE*, 75(A1):453-458.
- Prett, D. and E. Morari (1988). *Fundamental Process Control*. (2 ed.).
- Procyk, T. and E. Mamdani (1979). A linguistic self-organising process controller. *Automatica*, 15:15-30.
- Psichogios, D. and L. Ungar (1991). Direct and indirect model based control using artificial neural networks. *Chemical Engineering science.*
- Puig, V. (2004). Control tolerante a fallos (parte ii): Mecanismos de tolerancia y sistema supervisor. *Revista Iberoamericana de automática e informática industrial (RIAI)*, 1(2), 5-21.
- Ramirez, M. and E. Colina (2008a). Fuzzy clustering based models applied to petroleum processes. *WSEAS Transactions On Systems and Control, ISSN 1991-8763, Issue 3, Volume 3, March 2008.*

REFERENCIAS

- Ramirez, M. and E. Colina (2008b). Fuzzy model based control: Application to an oil production separator. *Hybrid Intelligent Systems (HIS)*, Universitat Politècnica de Catalunya, Barcelona, Spain on 12-14 September, 2008.
- Ramirez, M. and E. Colina (2010). A genetic algorithm based approach for generating singleton fuzzy models. *Proceedings of the 9th International Conference On Computational Intelligence, Man-Machine Systems and Cybernetics, (CINMACS)*, University of Los Andes, Merida, Venezuela, December 14-16 2010.
- Ramirez, M., J. Davila, and E. Colina (2009). Intelligent supervision of petroleum processes based on multiagent systems. *WSEAS Transactions On Systems and Control, ISSN 1991-8763, Issue 1, Volume 1, August 2009*.
- Renders, J., M. Saerens, and R. Hugues (1997). Fuzzy adaptive control of a certain class of siso discrete-time processes. *Fuzzy Sets and Systems*, 85:49-61.
- Rizo, R., F. Llorens, and M. Pujol (2002). Agentes inteligentes: Sistemas multiagentes y aplicaciones. *Editorial Club Universitario, D. L., chapter Arquitecturas y comunicación entre Agentes, pp. 181-214*.
- Rodriguez, M. (2009). Desarrollo e implantación de un sistema de inversor - motor tolerante a fallas basado en la medición de la señal de compuerta del igbt. *Tesis doctoral, CENIDET, Departamento de Ingeniería Electrónica*.
- Rodriguez, M., J. Abreu, R. Gomez, and B. Martinez (2007). Simulación borrosa de un reactor con reacción exotérmica no lineal. *Universidad Central de las Villas, Santa Clara, Cuba*.
- Rosenchein, J. (1994). Rules of encounters. *MIT, Press*.
- Roubos, J. and M. Setnes (2010). Compact fuzzy models through complexity reduction and evolutionary optimization. *Proceeding of the IEEE International Conference on Fuzzy System, San Antonio, USA*, 762-767.
- Rubinstein, A. (1952). Perfect equilibrium in a bargaining model. *Econometrica*, 50:97-109.
- Rumbau, J. and M. Blaha (1991). Object oriented modelling and design. *Prentice Hall*.
- Salas, A. (1998). Validación y aproximación funcional en sistemas de control basados en lógica borrosa. *Universidad Politécnica de Valencia, Tesis Doctoral*.
- Samper, J. (2005). Ontología para servicios webs semánticos de información de tráfico: descripción y herramientas de explotación. *Tesis Doctoral, Universidad Politécnica de Valencia, España*.

- Sanz, R. (1990). Arquitectura de control inteligente de procesos. *Tesis doctoral, Departamento de Automática de la escuela superior Ingenieros Industriales de la Universidad Politécnica de Madrid.*
- Sarrate, R. and J. Aguilar (2003). Vigilancia de un proceso a partir de la detección de eventos significativos. *En XXIV Jornadas de Automática, Universidad de León, España.*
- Shankar, E. and W. Shoubo (2003). Slug detection as a tool for predictive control of glcc compact separators. *Mechanical and Petroleum Engineering Department, University of Tulsa, Transactions of the ASME, Vol 125.*
- Sierra, C., P. Farantin, and R. Jennings (1997). A service-oriented negotiation model between autonomous agents. *8th European Workshop on modelling autonomous agents in a multi-agent world. Pages 17-35, Sweden.*
- Smith, R. and D. Corner (1991). Automated calibration of a fuzzy controller using a cell statespace algorithm. *Conference on System and Control, Brighton, England, 11-13.*
- Smith, R. and T. Johansen (1997). *Multiple Model Approaches to modeling and control* (2 ed.).
- Song, Q. and N. Kasabov (2005). Nfi: A neuro-fuzzy inference method for transductive reasoning. *IEEE Transactions On Fuzzy Systems, vol 13, 799-808.*
- Song, Q. and N. Kasabov (2006). Twnfi - a tranductive neuro-fuzzy inference system with weighted data normalization for personalized modeling. *Neural Networks, vol 19, 1591-1596.*
- Sousa, J. and U. Kaymak (2002). *Fuzzy Decision Making in Modeling and Control* (2 ed.).
- Struss, P. (1997). Fundamentals of model - based diagnosis of dynamic systems. *IJCAI97.*
- Suda, H. (1990). Future factory system formulated in japan. *Japanese Journal of Advanced Automation Technology.*
- Sugeno, M. and T. Yasukawa (1993). A fuzzy logic based approach to qualitative modeling. *IEEE Transactions on Fuzzy Systems, 7-10.*
- Sundermeyer, K. (1996). Belief-desire-intention agent architectures. *Foundations of Distributed Artificial Intelligence, Wiley-Interscience Publication.*
- Takagi, H., N. Suzuki, T. Koda, and Y. Kojima (1992). Neural networks designed on approximate architecture and their applications. *IEEE Transactions on Neural Networks, vol 3, 752-760.*
- Tanaka, H. (1987). Fuzzy data analysis by possibilistic linear models. *Fuzzy Sets and Systems, 363-375.*

- Tanaka, H., J. Hayashi, and J. Watada (1989). Possibilistic linear regression analysis for fuzzy data. *European Journal of Operational Research*, 389–396.
- Tanaka, M. and M. Sugeno (1992). *Stability analysis and design of fuzzy control systems* (1 ed.). Fuzzy Set And Systems.
- Tomas, R. (2007). Tacticas mixtas para la negociación automática de múltiples servicios con información incompleta en entornos multiagentes. aplicación a problemas de gestión de tráfico. *Tesis doctoral, Universidad Politécnica de Valencia, España.*
- Uraikul, V., C. Chan, and P. Tontiwachwuthikul (2007). Artificial intelligence for monitoring and supervisory control of process systems. *Engineering Applications of Artificial Intelligence*, 20(2), 207-216.
- Vagelatos, G., G. Rigatos, and S. Tzafestas (2001). Incremental fuzzy supervisory controller design for optimizing the injection molding process. *Intelligent Robotics and Automation Laboratory, Department of Electrical and Computer Engineering, National Technical University of Athens*, 20(2), 207-216.
- Vasquez, A. (2003). Apuntes sobre agentes y sistemas multiagente. *Universidad de Campus Monterrey.*
- Villa, J., M. Duque, A. Gauthier, and N. Rakoto (2003). Supervision and optimal control of a class of industrial processes. *Proceedings of the IEEE Conference On Emerging Technologies and Factory Automation, Vol 2, 117-180, September 12-16, Lisbon, Portugal.*
- Waismann, J. (2000). Wastewater treatment process supervision by means of a fuzzy model. *Proceedings of the 2000 IEEE International Symposium on Intelligent Control*, 163-168.
- Wang, L. (1994). *Adaptive Fuzzy Systems and Control: Design and Stability Analysis* (2 ed.).
- Wang, L. (1997). *A course in Fuzzy Systems and Control* (1 ed.).
- Wang, L. and J. Mendel (1992). Fuzzy basis function, universal approximation and orthogonal least-squares learning. *IEEE Transactions on Neural Networks*, vol 3, 807–814.
- Wang, W., H. Li, and J. Zhang (2003). A hybrid approach for supervisory control of of furnace temperature. *Control Engineering Practice*, 11(11), 1325-1334.
- Weiming, S., H. Douglas, and J. Barthes (2001). *Multi-Agent Systems for concurrent intelligent design and manufacturing* (2 ed.).
- Whitaker, H., J. Yamron, and A. Kezer (1958). *Design of model reference adaptive control systems for aircraft.* (1 ed.).

-
- Wooldridge, M. (2002). *An Introduction to Multiagent Systems* (1 ed.).
- Yager, R. and D. Filev (1994). *Essentials of Fuzzy Modeling and Control* (1 ed.).
- Yang, M. and K. C. (1997). On cluster-wise fuzzy regression analysis. *IEEE Transactions on Systems, Man and Cybernetics*, 1–13.
- Yazdi, H. (1997). Control and supervision of event-driven systems. *PhD Thesis, Technical University of Denmark (DTU), Denmark.*
- Zadeh, L. (1973). Outline of a new approach to the analysis of complex systems and decision processes. *IEEE Transactions on Systems, Man and Cybernetics*, 28–44.
- Zeungnan, B. and V. Jongcheol (1991). Fuzzy self-organizing visual tracking controller. *IFSA World Congress*, pp 21-24.

www.bdigital.ula.ve