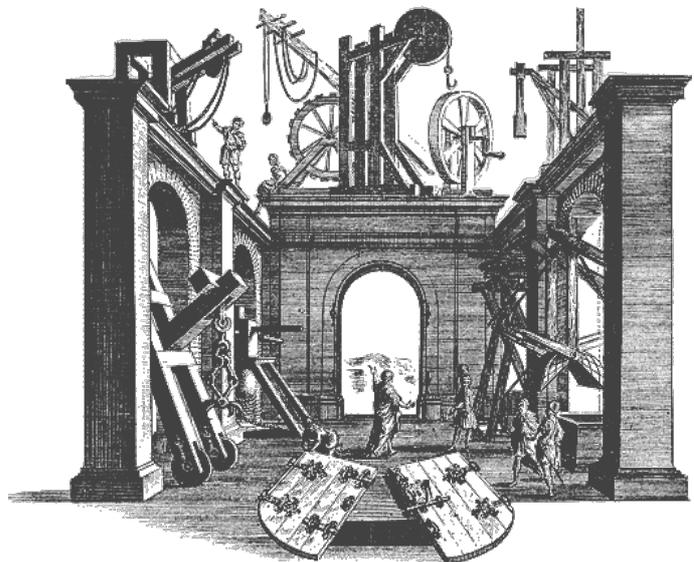


Arquitectura de Control Inteligente de Procesos

Ricardo Sanz



1990

aslab.org

Autonomous Systems Laboratory

THE AUTONOMOUS SYSTEMS LABORATORY
www.aslab.org

Arquitectura de Control Inteligente de Procesos

El sistema CONEX

Ricardo Sanz



ASL-T-1990-001
1990

Art is not to be found in a piece of canvas or in a pot of paint. Pots of the same paint may produce very different results from the brushes of different painters.

A computer is not intelligent nor is it set of basic instructions. Like a piece of canvas without a picture of paint, a computer is inert without a program of instructions. Using the same basic instructions, different programmers can produce quite different results in the same computer.

To ask if a programmed computer is intelligent is like asking if a painted canvas is beautiful. Intelligence, like beauty, seems to be "in the eye of the beholder", since what one calls beautiful or intelligent another scorns.

*John H. Andreae
Thinking with the Teachable Machine*

Resumen

En esta tesis se presenta una arquitectura de control, basada en técnicas de inteligencia artificial, de aplicación en control de procesos industriales complejos. Esta arquitectura de control permite integrar políticas de control que hacen uso de información sobre el proceso a todos los niveles. Desde las medidas de sensores de planta a los valores de análisis de laboratorio. Desde el conocimiento operacional representado por los procedimientos de control de los operadores hasta el conocimiento estructural-funcional de los ingenieros de proceso. El enfoque cooperativo y distribuido de esta arquitectura posibilita la construcción de sistemas robustos, eficientes y reutilizables, que han sido los principales objetivos del diseño de la misma.

En resumen, se propone una arquitectura de control inteligente para procesos complejos, desarrollando algunos aspectos técnicos que son fundamentales en la misma y no han alcanzado el suficiente nivel de madurez en la actualidad, en particular la descomposición de funciones de control y la construcción de modelos para razonamiento de control. De este modo se logra formular una metodología teórico práctica de desarrollo de sistemas avanzados de control, orientada a las industrias de proceso, que permita la implantación de toda una gama de sistemas de control inteligente en dichas industrias.

Arquitectura de Control Inteligente de Procesos

ASLab ASL-T-1990-001 v 1.0 Final of 1990

Abstract

Esta tesis doctoral se realizó en el Departamento de Automática de la Escuela Técnica Superior de Ingenieros Industriales de la Universidad Politécnica de Madrid, desde 1987 a 1990. Sus directores fueron D. Agustín Jiménez Avello, catedrático de Ingeniería de Sistemas y Automática y Doctor Ingeniero Industrial y D. Ramón Galán López, Profesor Titular de Ingeniería de Sistemas y Automática y Doctor en Informática.

Keywords

Control inteligente, control distribuido, arquitecturas de control, sistemas heterogéneos, control de hornos.

Acknowledgements

Deseo expresar mi agradecimiento a todos los que, de un modo u otro, han contribuido a que este proyecto de tesis, en ocasiones de perspectiva incierta, se culmine con mayor o menor éxito. Algunos de ellos son: La empresa ASLAND Tecnología S.A. por su apoyo; Juan Ramón Velasco, por tantas discusiones tan rentables; Eugenio Andrés Puente que tiene un espíritu increíble; Adolfo Yela, que ha aguantado estoicamente mis desvaríos, aportando una importante visión crítica de los mismos; Pedro y Paulo, por la envidia que me dá su capacidad de trabajo; Marta, Gamaliel y el resto de la gente de IA, por estar ahí, peleando; Fernando, Cecilio, Alvaro y Jose Antonio, por embarcarse; Luis, María, Angel, Enrique, Magalí, Aida y todos los demás les agradezco su manera de compartir tantas horas; Familia, amigos y demás conocidos que han soportado mis obsesiones, por eso mismo.

Tabla de Contenido

1. Introducción	1
1.1. Estado del Arte	1
1.2. Objetivos de la Tesis	2
1.3. Estructura de la Memoria	3
2. Control de Procesos Complejos	4
2.1. Introducción	4
2.2. Características de los Procesos Complejos	4
2.2.1. Complejidad volumétrica o primaria	5
2.2.2. Complejidad de interacción o secundaria	5
2.2.3. Incertidumbre de comportamiento o complejidad terciaria	6
2.3. Control de Procesos Complejos	6
2.3.1. Enfoques de control	6
2.3.2. Control de un proceso complejo	7
2.4. Control Supervisor	8
2.5. Inteligencia Artificial e Inteligencia Natural	10
2.6. Inteligencia Artificial en Control	10
2.6.1. Objetivos	12
2.6.2. Aplicabilidad	12
2.6.3. Reutilizabilidad	13
2.7. Campos de Aplicación	13
3. Modelos Simbólicos de Sistemas Físicos	15
3.1. Introduction	15
3.2. Modelos	15
3.2.1. ¿Qué es un Modelo?	15

3.2.2.	Características de los modelos	16
3.2.3.	Modelos informáticos	17
3.3.	Modelos para Control	18
3.3.1.	Utilización de modelos numéricos	19
3.3.2.	Utilización de modelos simbólicos implícitos	20
3.3.3.	Utilización de modelos simbólicos explícitos	21
3.4.	Modelos Cualitativos	22
4.	Control Inteligente de Procesos	25
4.1.	Introducción	25
4.2.	Control Inteligente: Objetivos y Definiciones	25
4.2.1.	Objetivos generales del control inteligente	26
4.2.2.	Definición	26
4.3.	Perspectiva Histórica	27
4.3.1.	Control clásico y supervisor	27
4.3.2.	Inteligencia artificial y sistemas expertos	28
4.3.3.	Sistemas de primera generación	28
4.3.4.	Sistemas de segunda generación	29
4.3.5.	Sistemas de tercera generación	30
4.4.	Elementos de Control Inteligente	30
4.4.1.	Objetivos de un sistema de control inteligente	30
4.4.2.	Actividades de control	30
4.4.3.	Aclaración de denominaciones	31
4.4.4.	Tendencias en control inteligente	31
4.4.5.	Características fundamentales del control inteligente	32
4.4.6.	Control	34
4.4.7.	Proceso	37
4.4.8.	Inteligencia artificial	39
4.5.	Paradigmas de Control Inteligente	42
4.5.1.	Control basado en reglas	42
4.5.2.	Control borroso	50
4.5.3.	Control basado en modelos	52
4.5.4.	Ventajas del control basado en modelos	54
4.6.	Desarrollos actuales	55

4.6.1.	Desarrollos específicos	55
4.6.2.	Desarrollos independientes	55
4.6.3.	Desarrollos con herramientas de control inteligente	56
4.6.4.	Evoluciones	56
4.7.	Control Inteligente Avanzado	58
4.7.1.	Aprendizaje en control inteligente	58
4.7.2.	Arquitecturas distribuidas	59
5.	Metodología de Diseño	60
5.1.	Introducción	60
5.2.	Objetivos de la descomposición	62
5.3.	Sistemas Jerárquicos	62
5.3.1.	Tipos de jerarquías	62
5.3.2.	Jerarquías en sistemas de control inteligente	65
5.3.3.	Descomposición por dominios	65
5.3.4.	Descomposición funcional	67
5.3.5.	Descomposición representacional	67
5.3.6.	Descomposición temporal y por paradigmas	69
5.3.7.	Estructuración jerárquica	70
5.4.	Operación en tiempo real	72
5.4.1.	Requisitos fuertes	72
5.4.2.	Requisitos débiles	73
5.4.3.	Sistemas basados en reglas borrosas	73
5.4.4.	Sistemas basados en reglas	73
5.4.5.	Sistemas basados en modelos	75
5.5.	Descomposición de Sistemas basados en el Conocimiento	75
5.5.1.	Descomposición de la base de conocimientos	75
5.5.2.	Planificación de bases de conocimiento	77
5.5.3.	Criterios de selección de bases	77
5.5.4.	Razonamiento progresivo	78
5.5.5.	Razonamiento conclusivo	79
5.6.	Valoración de la Metodología	79
6.	Modelado Multirresolucional	80
6.1.	Introduction	80

6.2. Modelos de Sistemas Continuos y Multirresolucionalidad	81
6.2.1. Conceptos de modelado de sistemas continuos	82
6.3. Modelado Multirresolucional	82
6.3.1. Necesidad de la multiresolucionalidad	83
6.3.2. Objetivos del modelo	84
6.3.3. Estructura del modelo	84
6.4. Modelo N	85
6.4.1. Técnicas básicas	85
6.4.2. Funciones del modelo	85
6.4.3. Implantación	85
6.5. Modelo Q	86
6.5.1. Funciones del modelo	86
6.5.2. Técnicas básicas	86
6.5.3. Simulación cualitativa	87
6.5.4. Extensiones a la representación	88
6.5.5. Funciones NQ y QN	90
6.5.6. Extensiones a la simulación	90
6.5.7. Simulación cualitativa borrosa	91
6.5.8. Implantación	94
6.6. Modelo K	95
6.6.1. Estructura del modelo K	95
6.6.2. Funciones QK y KQ	95
6.6.3. Funciones del modelo	95
6.6.4. Implantación	95
6.7. Utilización de Modelos en Control	96
6.7.1. Información sobre el estado	96
6.7.2. Predicción del comportamiento	96
6.7.3. Información estructural / funcional	97
7. Arquitectura de Control Inteligente	99
7.1. Introducción	99
7.2. Objetivos de la Arquitectura	99
7.2.1. Objetivos del sistema final	99
7.2.2. Objetivos de desarrollo	100

7.3. Fundamentos de la Arquitectura	100
7.3.1. Control de procesos	100
7.3.2. Etapas de control inteligente	101
7.3.3. Niveles de control	101
7.3.4. Descomposición heterogénea	102
7.3.5. Objetos activos y pasivos	103
7.4. Arquitectura Software	105
7.4.1. Estructura global	105
7.4.2. Estructura estándar de un OAN	109
7.4.3. Interfase estándar de un OAN	109
7.4.4. CONEX-CD: Control Directo	109
7.4.5. CONEX-IP: Interfase de Proceso	113
7.4.6. CONEX-MP: Monitor de Proceso	115
7.4.7. CONEX-MS: Modelo y Simulador	117
7.4.8. CONEX-CI: Control Inteligente	121
7.4.9. CONEX-IU: Interfase de Usuario	123
7.4.10. CONEX-EA: Evaluador de actuaciones	125
7.4.11. CONEX-MC: Monitor de CONEX	127
7.4.12. CONEX-IE: Interfase Externa	129
7.5. Arquitectura Hardware	129
7.6. Valoración de la Arquitectura	130
7.6.1. Desarrollo independiente de los subsistemas	130
7.6.2. Escalabilidad	130
7.6.3. Fiabilidad, disponibilidad y mantenibilidad	131
8. Control de Hornos de Clinker	132
8.1. Introducción	132
8.2. Características del proceso	132
8.2.1. Plantas de producción de cemento	132
8.2.2. Hornos de Cemento	133
8.2.3. Problemas de control	133
8.2.4. Desarrollos actuales	135
8.3. Sistema de Control de Hornos	135
8.3.1. Objetivos	135

8.3.2. Arquitectura Software	135
8.3.3. Arquitectura Hardware	136
8.4. Resultados esperados	137
9. Conclusiones	139
9.1. Aportaciones de esta tesis	139
9.2. Desarrollos futuros	140
A. Criterios de aplicación del control inteligente en procesos industriales	142
B. Glosario	143
B.1. Términos	143
B.2. Acrónimos	148
C. Bibliografía	149
C.1. Referencias Bibliográficas	149
C.2. Bibliografía adicional	154
D. Parámetros de los conjuntos borrosos	166

Lista de figuras

2.1. Esquema elemental de un sistema de control supervisor en el que un operador humano supervisa la operación de un sistema de control automático.	5
2.2. El proceso de control inteligente consta de dos etapas fundamentales: 1) la valoración de la situación y 2) la realización de la acción de control.	8
3.1. Objeto en caída libre.	20
3.2. Base de conocimiento de control del sistema de dos grifos.	21
3.3. Modelo cualitativo del sistema de dos grifos caliente/frio.	22
3.4. Ejemplo de simulación del modelo cualitativo del objeto en caída libre a partir de una cierta altura y velocidad inicial hacia arriba. Las cajas amarillas representan los diferentes estados cualitativos por los que pasa el sistema.	24
4.1. Esquema de control clásico.	28
4.2. Tipos de conocimiento y procedencia del mismo.	32
4.3. Sistema sin vínculos con el mundo. Las decisiones de acción las toma el operador humano.	34
4.4. Sistema con vínculos con el mundo. Las decisiones de acción las toma el sistema de control y el operador humano en casos especiales.	35
4.5. Comparación entre sistemas expertos clásicos y sistemas expertos de control.	37
4.6. Orígenes diversos de las diferentes partes de la base de conocimiento. El ingeniero de conocimiento cataliza el proceso de formalización y representación.	41
4.7. Estructura clásica de un sistema basado en el conocimiento expresado mediante reglas.	43
4.8. Sistema basado en reglas con estructura monocapa.	46
4.9. Sistema basado en reglas con estructura bicapa.	47
4.10. Estructura de un sistema de control basado en reglas borrosas.	52
4.11. Sistema basado en modelos.	53
5.1. Nivel físico, nivel de circuito y nivel lógico.	63
5.2. Jerarquía de control de Saridis [Saridis 88].	63

5.3. Descomposición en escalones de un sistema de control de robot usando la jerarquía de control de Saridis.	64
5.4. Triple criterio de descomposición.	65
5.5. El sistema de control de una fábrica de cemento se descompone en tres grandes subsistemas en relación con las unidades mayores del proceso: control de la torre de intercambio, control del horno y control del enfriador.	68
5.6. Correlación paradigma-abstracción-tiempo.	69
5.7. Ejemplo de sistema de control jerárquico elemental.	70
5.8. Ejemplo de sistema de control basado en el conocimiento con un metanivel de selección.	78
6.1. Estructura y transformaciones en el modelo multiresolucional NQK.	84
6.2. Ejemplo de árbol de comportamientos en simulación cualitativa.	89
6.3. Espacios de cantidades borrosos.	91
6.4. Detección de estados iguales. Dado que $E2.2 \equiv E2.4$ los subárboles que cuelgan de dichos estados son idénticos —de hecho son los mismos estados.	93
6.5. Detección de comportamientos periódicos por identificación de transiciones a estados previamente considerados.	94
6.6. Utilización del conocimiento sobre el proceso en control.	97
7.1. Estructura de control avanzado.	100
7.2. Jerarquía de niveles de control en la arquitectura CONEX.	101
7.3. Diagrama de contexto de CONEX.	104
7.4. Estructura global de CONEX con las principales interacciones entre los nueve objetos activos de alto nivel: CD: Control directo, IP: Interfase de proceso, MP: Monitor de proceso, MS: Modelo y simulador, CI: Control inteligente, IU: Interfase de usuario, EA: Evaluador de actuaciones, MC: Monitor de CONEX e IE: Interfase externa.	107
7.5. Estructura de software de CONEX. Distribución de objetos de alto nivel.	110
7.6. CONEX-CD: Control Directo	111
7.7. Procesado de patrones en CONEX-CD.	113
7.8. CONEX-IP: Interfase de Proceso	114
7.9. Transformaciones de señales en CONEX-IP.	116
7.10. CONEX-MP: Monitor de Proceso	116
7.11. CONEX-MS: Modelo y simulador	118
7.12. CONEX-CI: Control Inteligente	122
7.13. CONEX-IU: Interfase de Usuario	124
7.14. CONEX-EA: Evaluador de actuaciones	126
7.15. CONEX-MC: Monitor de CONEX	128

8.1. Estructura básica de un horno de clinker para fabricación de cemento.	134
8.2. Parte de la jerarquía de controles del horno de cemento.	137

Lista de tablas

2.1. Ventajas y desventajas del control supervisor.	11
2.2. Algunas aplicaciones de la inteligencia artificial en control.	12
3.1. Ejemplos de uso de modelos considerando dos aspectos fundamentales: explicitud y abstracción.	19
4.1. Algunos aspectos característicos de los sistemas de control inteligente.	33
5.1. Facetas primarias a emplear en la descomposición modular de sistemas inteligentes de control.	66
5.2. Denominaciones características empleadas para los sistemas complejos descompuestos en función de las facetas que han guiado dicha descomposición.	71
6.1. Restricciones cualitativas.	89
7.1. Misiones principales de los objetos de alto nivel de CONEX.	108
D.1. Parámetros de los conjuntos borrosos	166

Estas aplicaciones se llevan a cabo utilizando herramientas de construcción de sistemas expertos clásicas, herramientas especialmente diseñadas para control inteligente o bien constituyen desarrollos específicos. Como resultado del uso de herramientas convencionales, dichas herramientas resultan potenciadas con una serie de elementos que llegan a constituir nuevas herramientas (es el caso del sistema PLEXYS [Intellicorp 87a], evolución de la herramienta KEE).

Las herramientas específicas para control inteligente —centradas en el control basado en reglas— están logrando una progresiva introducción en el mercado, al ir mejorando sus prestaciones de representación de conocimiento, razonamiento y operación en tiempo real; es el caso de G2 [Gensym 89] —sucesor de PICON—, CHRONOS [Euristic 89], COMDALE y NEMO [S₂O 90].

Los desarrollos en control inteligente se dan fundamentalmente en el ámbito universitario o en el marco de grandes proyectos de investigación, como pueden ser los patrocinados por el DoD americano, la NASA o la comunidad europea a través de sus programas de I+D (ESPRIT, EUREKA, etc). Ejemplos de este tipo son los sistemas desarrollados —o en proceso de desarrollo— en los proyectos ESPRIT: TEX-I [Voss 88], QUIC [Cavanaugh 89], KRITIC, AIMBURN, IPCES, etc. De especial importancia son estos últimos por sus objetivos claramente industriales.

Las características más importantes de estos sistemas se resumen en que consiguen implantar estrategias de control avanzadas, basadas en técnicas de inteligencia artificial, pero con limitaciones en cuanto al tipo de políticas de control utilizables y el tipo de procesos a los que son aplicables, sobre todo debido a la lentitud de estas aplicaciones, que limita su operación en tiempo real. Los sistemas inteligentes —por ejemplo los basados en reglas— incurrir en cargas de cómputo en tiempo de ejecución que son impredecibles, lo que hace inviable su uso en determinado tipo de sistemas (sistemas con requerimientos fuertes de operación en tiempo real).

Todo ello nos permite alcanzar una conclusión final. El control inteligente de procesos es una área de I+D muy prometedora, en la que se están logrando importantes resultados, aunque con un ex-

cesivo grado de dispersión. No existe un cuerpo de conocimiento central que permita situar y valorar todos los desarrollos, debido fundamentalmente a que el problema del control inteligente es un problema de formulación de un modelo complejo de un sistema físico aún más complejo; esto hace que los factores psicológicos sean en gran medida críticos, ya que las políticas de identificación del estado del sistema y de razonamiento sobre la evolución del mismo se derivan de los modos de pensar del personal —informático y de proceso— que construye el sistema de control inteligente. Intentos de formulaciones teóricas independientes han tenido resultados dispares, aunque dentro de ellas cabe destacar al grupo de investigadores en torno a Saridis [Saridis 88b]. No obstante este tipo de teorías pecan de un excesivo alejamiento de lo que es ingeniería, siendo por tanto de aplicabilidad directa limitada.

Además existen problemas con soluciones precarias —como son los motores de inferencia en tiempo real o el uso de modelos profundos— que limitan la aplicabilidad de estos sistemas. Ello es debido por lo general tanto a consideraciones de operación en tiempo real como a seguridad y fiabilidad.

1.2. Objetivos de la Tesis

Los objetivos de esta tesis se centran en varios puntos:

- Determinar la factibilidad de los sistemas inteligentes de control de procesos. Se tratará de determinar la viabilidad técnica de los sistemas inteligentes de control de procesos, estableciendo los elementos fundamentales de los mismos, así como los posibles beneficios derivables de su utilización.
- Desarrollo de una técnica de modelado de sistemas físicos para razonamiento sobre los mismos. Esta técnica será el soporte de los procesos de razonamiento a todos los niveles, tanto en planificación como en control directo.
- Proponer una arquitectura para control inteligente de procesos complejos. Esta arquitectu-

ra deberá garantizar la robustez y adaptabilidad del sistema de control inteligente. Este es el objetivo fundamental.

- Desarrollo de una aplicación de control inteligente basada en dicha arquitectura. Esta aplicación tratará de controlar un proceso de complejidad media-alta, que presente problemas para su control clásico.

En resumen, el objetivo de la tesis es:

Proponer una arquitectura de control inteligente para procesos complejos, analizando algunos aspectos técnicos que son fundamentales en la misma y no han alcanzado el suficiente nivel de madurez en la actualidad, en particular la construcción de modelos para razonamiento de control.

De este modo se logrará formular una metodología teórico-práctica orientada a las industrias de proceso, que permita la implantación de toda una gama de sistemas de control inteligente en estas industrias.

1.3. Estructura de la Memoria

Esta memoria de tesis está dividida en nueve capítulos y tres apéndices. El contenido de cada uno de ellos se comenta a continuación.

Capítulo 1: INTRODUCCION

El presente capítulo, en el que se comenta la estructura de la tesis, junto con un breve análisis del estado del arte.

Capítulo 2: CONTROL DE PROCESOS COMPLEJOS

Se comenta la problemática del control de procesos complejos, analizando la aplicabilidad de técnicas de inteligencia artificial y los objetivos que se pretenden lograr con ello.

Capítulo 3: MODELOS SIMBOLICOS DE SISTEMAS FISICOS

Se estudian las técnicas de modelado simbólico para razonamiento sobre sistemas físicos.

Capítulo 4: CONTROL INTELIGENTE

Se analizan las características técnicas de los sistemas de control inteligente, estableciendo las bases para un sistema inteligente de control de procesos.

Capítulo 5: METODOLOGIA DE DISEÑO

En este capítulo se presenta una metodología de diseño, desarrollada en esta tesis, de sistemas de control inteligente, en la que se basa la arquitectura propuesta en esta tesis.

Capítulo 6: MODELADO MULTIRESOLUCIONAL

En este capítulo se presenta la técnica de modelado de sistemas desarrollada en esta tesis.

Capítulo 7: ARQUITECTURA DE CONTROL INTELIGENTE

En él se presenta CONEX, la arquitectura de control desarrollada en esta tesis, analizando cada uno de los módulos que la componen.

Capítulo 8: CONTROL DE HORNOS DE CLINKER

Se analiza una aplicación basada en la arquitectura CONEX que tiene como objetivo el control de hornos de clinker, uno de los subprocesos de fabricación de cemento.

Capítulo 9: CONCLUSIONES

En él se comenta el grado de satisfacción de los objetivos propuestos para esta tesis, analizando los posibles desarrollos posteriores.

Apéndice A: CRITERIOS

Estos criterios permiten decidir si un campo de aplicación es adecuado para el uso de un sistema inteligente de control.

Apéndice B: GLOSARIO

Términos de especial importancia dentro del área de control inteligente.

Apéndice C: BIBLIOGRAFIA

Referencias utilizadas en esta tesis y referencias adicionales.

Capítulo 2

Control de Procesos Complejos

COMPLEJO:... asunto en que hay que considerar muchos aspectos, por lo que no es fácil de comprender o resolver

MARIA MOLINER
Diccionario de Uso del Español

- Un bucle interno, en base a un regulador clásico (PID o PI).
- Un bucle externo, *llevado a cabo por un humano*, encargado de monitorizar la evolución del proceso y de realizar las correcciones pertinentes, por lo general a través de las consignas de los bucles internos.

2.1. Introducción

Existen gran cantidad de procesos industriales que sufren una falta grave de automatización, con las consiguientes repercusiones en la productividad y disponibilidad del proceso. Esta carencia de funcionamiento automático es debida a diversas circunstancias, pero que cabe agrupar bajo un sólo concepto: *COMPLEJIDAD*. Los procesos industriales son complejos en tres sentidos:

- Número de elementos que componen el sistema.
- Interacciones entre dichos elementos.
- Desconocimiento del modo exacto de interacción.

Estos aspectos de complejidad —*complejidad primaria, secundaria y terciaria*— dificultan la implantación de sistemas de control, sobre todo los aspectos secundario y terciario. Esta dificultad es máxima cuando nos limitamos al uso de técnicas clásicas de control, que son prácticamente las únicas que se emplean habitualmente en la industria. Por ello en muchos de estos sistemas se utiliza una estructura de control con dos bucles:

Como veremos en este capítulo y en los sucesivos, la inteligencia artificial puede servir de sustancial apoyo para la construcción de sistemas de control automático integrales, ya que la flexibilidad de las representaciones del conocimiento que utiliza va a permitir hacer frente a la complejidad del proceso. En este capítulo analizaremos los objetivos de la aplicación de técnicas de inteligencia artificial al control de procesos complejos. Para ello estudiaremos en primer lugar el concepto de proceso complejo, determinando los problemas principales para su control, estudiando a continuación la posible utilización de la inteligencia artificial y los resultados hasta ahora conseguidos.

2.2. Características de los Procesos Complejos

El aspecto característico que dificulta el control de determinados procesos industriales es la complejidad de los mismos. Como mencionábamos anteriormente la complejidad de un proceso viene determinada por tres factores: los elementos que lo componen, las interacciones entre ellos y la incertidumbre en el comportamiento interno de los mismos o el derivado de dichas interacciones.

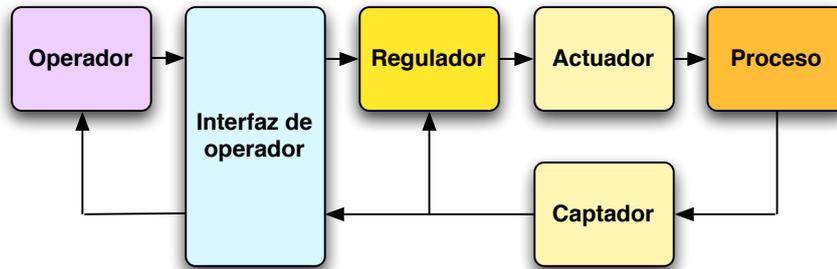


Figura 2.1: Esquema elemental de un sistema de control supervisor en el que un operador humano supervisa la operación de un sistema de control automático.

Cada uno de estos aspectos de complejidad dificultan de manera específica la introducción de sistemas de control en dichos procesos.

2.2.1. Complejidad volumétrica o primaria

Un proceso es volumétricamente complejo cuando el número de elementos que lo constituyen es muy elevado. Es lo mismo que decir que el número de variables a medir y controlar es muy grande.

Desde el punto de vista que nos ocupa, el control automático, este aspecto no es insalvable; para controlar un proceso volumétricamente complejo basta con utilizar un sistema de control proporcionalmente grande, en el caso de control digital un computador o sistema de computadores lo suficientemente grande como para hacer frente a todos los bucles de control que se precisen.

De hecho en la actualidad los sistemas de control de procesos complejos están en esta fase. Cientos de bucles simples de control implantados en grandes máquinas o en sistemas distribuidos. Estos bucles simples son por lo general PIDs o PIs, ya que el implantar controles más complejos es excesivamente costoso —no sólo económicamente— para los resultados que se obtienen.

No obstante en algunos casos se emplean sistemas más avanzados (adaptativos, predictivos, etc) que están disponibles comercialmente [ASEA 83]

[Adaptech 89].

2.2.2. Complejidad de interacción o secundaria

El segundo nivel de complejidad surge del acoplamiento entre los diferentes subsistemas. La ingeniería de proceso trata de conseguir que la complejidad de interacción sea mínima, dado que de este modo el control —manual o automático— del sistema es más simple y eficaz, lo que conduce a mejoras de productividad y disminución de necesidades de mantenimiento. No obstante este desacoplamiento de subsistemas no es siempre alcanzable, por lo que desde la perspectiva clásica el control del proceso sólo es viable de dos formas:

- Utilizando control multivariable, lo que lleva asociado problemas de gran dificultad. El uso de sistemas de control multivariable está restringido a procesos de complejidad limitada, sobre todo debido al coste de desarrollo, instalación y puesta en marcha de los mismos.
- Utilización de control manual mediante un operador humano, que es un experto en control del proceso.

La identificación del operador como *experto* es la que ha traído de la mano a la inteligencia artificial al campo de control de procesos. En una primera aproximación se trata de modelar el comportamiento del operador mediante reglas, logrando

resultados aceptables en algunos casos. No obstante el control de un proceso de complejidad elevada se basa en gran medida en la capacidad de *sentido común* del operador, difícilmente implantable en un sistema experto.

2.2.3. Incertidumbre de comportamiento o complejidad terciaria

Si se carece de un modelo del proceso a controlar, el control posible del mismo queda muy restringido.

La identificación del proceso a controlar es un factor fundamental si se desea usar un control avanzado, pero sólo se muestra viable cuando la dinámica del proceso es muy definida, con alteraciones debidas tan sólo a modificación o deriva de algún parámetro del proceso. En caso de sistemas con mucho ruido y con estructura muy cambiante la utilidad de un modelo reducido y la identificación disminuye.

El aspecto terciario de complejidad es justificable en último término como complejidad secundaria exacerbada. Las interacciones son tantas y a tan bajo nivel, que la predicción del comportamiento exacto del sistema es imposible.

Vuelve a surgir el operador y su sentido común para hacer frente a este problema, ya que con su capacidad de razonamiento y sobre todo con su *experiencia* en el funcionamiento del sistema, es capaz de determinar la acción de control adecuada —o al menos una que se le aproxima.

2.3. Control de Procesos Complejos

El control de procesos complejos se ve afectado por una serie de factores que dificultan su realización. Estos factores cabe agruparlos en generales —no exclusivos de este tipo de procesos— y de complejidad. Los factores generales son los que están asociados a las tareas de control, derivados esencialmente de la interacción con un sistema físico. Entre estos factores se pueden destacar: la necesaria operación en tiempo real, la interac-

ción con sistemas dinámicos, la falta de fiabilidad de equipos, etc.

Por otra parte los factores de complejidad tienen su origen en los aspectos de complejidad mencionados en el apartado. Las características que deben poseer los sistemas de control para hacer frente a estos problemas son: ser capaces de implementar un número elevado de bucles de control elevado, poder hacer control multivariable y manejar los diferentes aspectos de incertidumbre.

Esto nos conduce a plantear como necesidades de un sistema de control las siguientes: equipos potentes, algoritmos avanzados y modelos multi-resolucionales. Toda esta problemática de control para procesos complejos ha conducido a un punto en que el planteamiento del control se hace directamente en base a la utilización de controles automáticos simples, junto con personal humano encargado de la supervisión de los mismos.

2.3.1. Enfoques de control

El problema de la complejidad volumétrica es el de más fácil resolución, pues para ello basta la fuerza bruta: el tamaño del ordenador —o del sistema de ordenadores— se hace proporcionalmente grande al número de variables a controlar.

Este tipo de sistemas es el que se puede encontrar más fácilmente en la industria, existiendo gran cantidad de productos comerciales que son aplicables en este nivel [Foxboro 86]. Este tipo de productos tienen como principales clientes a industrias químicas; apareciendo la mayor parte de los paquetes comerciales en el campo de la industria petroquímica. Estos productos ofrecen la posibilidad de implantar multitud de bucles de control, manejando decenas de miles de variables simultáneamente. Sin embargo es preciso el uso de un operador de proceso; una persona encargada de supervisar el funcionamiento del sistema. Este enfoque —de fuerza bruta dijimos antes— da resultados buenos respecto a la operación en modo manual.

No obstante la economía es la que en último término controla el avance técnico, y es la necesidad de mejorar rendimientos la que conduce a los sistemas de control avanzado, en los que se em-

plean técnicas matemáticas para gestionar y optimizar la producción de la planta industrial [Nistal 89b]. Este tipo de sistemas de control es más difícil de encontrar en la práctica como paquete comercial. Por lo general los desarrollos en este campo son individuales, construidos por empresas de ingeniería para plantas específicas. El campo de aplicación de este tipo de sistemas es, no obstante, el mismo que el de los sistemas anteriores: las plantas *limpias*.

Cuando el proceso a controlar tiene dificultades de modelado, estas se derivan generalmente de la complejidad terciaria. Este es, con mucho, el problema más importante para el control de procesos *sucios*. En este caso los sistemas del control automático existentes de forma generalizada reducen su actuación al control de bucles simples del proceso, sin realizar un control global del mismo. Es preciso utilizar un operador *experto* para controlar el proceso.

Las necesidades de mejora de rendimientos y la supresión de dependencias humanas hacen que la búsqueda de soluciones a estos problemas de control sea cada vez más intensa. El enfoque que parece más prometedor es el uso de técnicas de ingeniería del conocimiento, que se justifican en base a que *si un humano puede hacerlo, la máquina debe poder también*.

2.3.2. Control de un proceso complejo

En el control de un proceso complejo es precisa la existencia de sistemas adicionales a lo constituye un control clásico y que posibiliten la realización de las tareas de control en un amplio rango de circunstancias. Este último punto es de fundamental importancia; la multiplicidad de circunstancias que se producen en la operación de procesos complejos es la que determina la inoperancia de los sistemas de control clásicos, debido a que estos son adecuados para situaciones muy determinadas. En una situación particular el comportamiento del proceso se rige por una serie de relaciones que determinan la idoneidad de un determinado regulador clásico. Cuando las circunstancias cambian es preciso determinar esta salida de *ámbito de aplicación* para evitar los problemas a que conduce el uso de un regulador inadecuado a una

situación. Este tipo de sistemas de determinación en línea de salida de márgenes de operación es de gran importancia para la robustez del sistema de control.

El número de maneras en que un proceso puede salirse fuera del margen de operación de un regulador es muy variado, siendo la causa de la progresiva complejidad de los sistemas de control clásicos, en particular los implantados con computador que son los que permiten realizar estas gestiones de forma eficaz. Los sistemas de control basados en *hardware* son bastante inflexibles en este sentido, disponiendo prácticamente de un único método de control de fuera de márgenes: el uso de umbrales de alarma en las variables medidas. Este tipo de mecanismos permiten acotar una zona del espacio de valores de las variables que se considera *controlable*. En control con computador la determinación de estas zonas es más flexible, siendo habitual el uso de diversos tipos de umbrales, que determinan varias zonas de operación diferenciadas, en las que se usan diferentes algoritmos de control.

Si analizamos este tipo de situaciones, así como la manera en que los humanos realizan el control de este tipo de sistemas, vemos que el proceso de control consta de dos etapas fundamentales:

Valoración de la situación, en la que se establece cual es el *estado funcional del proceso*, información que se empleará para determinar la estrategia de control a emplear. Esta determinación del estado funcional se lleva a cabo por medio de información sobre el estado del proceso, por lo general variables medidas, aunque también es habitual el uso de historias de valores u otro tipo de información como es el caso de análisis espectrales.

Determinación de la actuación, en la que se establecen las *acciones de control* a realizar sobre el proceso. Estas acciones se calculan a partir de la información sobre el estado del proceso — variables medidas— y empleado el algoritmo adecuado al estado funcional establecido por la etapa anterior.

Este tipo de mecanismo dual se implanta habitualmente en los sistemas de control con compu-

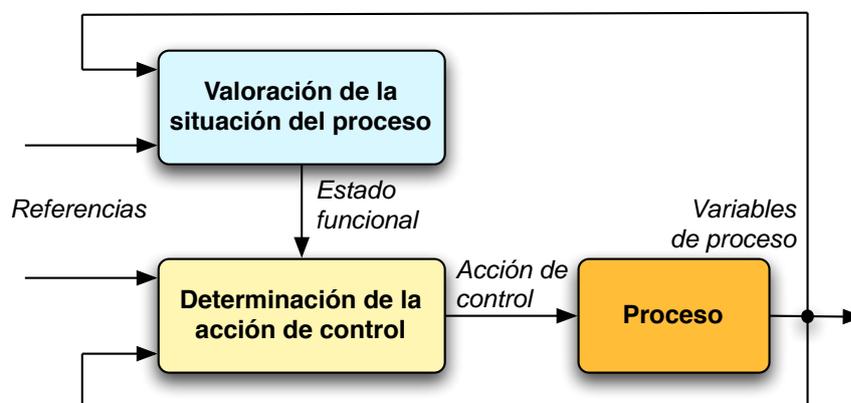


Figura 2.2: El proceso de control inteligente consta de dos etapas fundamentales: 1) la valoración de la situación y 2) la realización de la acción de control.

tador, aunque en muchos casos su diseño e implantación no se realizan teniendo en cuenta esta función doble, por lo que las dos etapas suelen estar integradas en una etapa única que puede llegar a ser bastante compleja. En un caso particular —el sistema COMLAND desarrollado por el Departamento de Automática— se tiene este tipo de estructura. El sistema implanta un doble regulador en cascada, al que se añade un bucle de optimización de producción cuando las condiciones del proceso son adecuadas. El sistema reconoce cuatro regiones de operación: optimizable, normal, atención y alarma. De haberse implantado desde un principio con estructura de dos etapas el desarrollo hubiera sido mucho más simple y el sistema resultante más eficaz y robusto.

El problema de la implantación de esta estructura dual se origina en la primera etapa. Esta es una etapa que puede requerir de procesos de información a muy alto nivel, siendo una etapa claramente basada en conocimiento declarativo —situaciones a reconocer— mientras que la segunda es una etapa fundamentalmente procedural. Por ello en sistemas de control poco avanzados la primera etapa se minimiza, desapareciendo incluso en el caso de los reguladores más simples: no se realiza ningún proceso de valoración del estado del proceso y por ello se emplea siempre el mismo algoritmo de determinación de actuaciones: el regulador convencional. El control adaptativo es

—en cierto sentido— una realización de esta estructura dual, en la que el número de reguladores a seleccionar es prácticamente infinito —un rango continuo— y en el que la identificación del estado funcional no se hace en cada ejecución del algoritmo de control, sino con otro tipo de periodicidad.

2.4. Control Supervisor

No obstante la complejidad inherente a muchos procesos, la estructura de control supervisor basada en un operador humano *tiene éxito*. Este éxito cabe atribuirlo a que el modelo mental que el operador tiene del proceso y los mecanismos de identificación que emplea para el ajuste de dicho modelo, son, cuando menos, mejores que los utilizados en sistemas de control automático; lo que le permite realizar la etapa de identificación del estado funcional de una manera más efectiva.

A pesar de lo anterior el control basado en humanos no es la panacea, ya que el operador es *humano*, y como tal esta sometido a cansancio, errores de percepción, errores de decisión, etc. Los tres conceptos fundamentales de fiabilidad, disponibilidad y mantenibilidad presentan características particulares cuando se habla del operador humano:

Fiabilidad : El hombre es menos fiable que la máquina, sobre todo porque sus modos de fallo son menos predecibles y menos identificables.

Disponibilidad : La disponibilidad del humano es menor; ello aparece claramente reflejado en uno de los problemas fundamentales para el mantenimiento de la estabilidad en operación de procesos: los turnos. Debido al cansancio normal de las personas es necesaria la existencia de turnos de operadores, cada uno con sus *modelos mentales*, sus *técnicas de identificación y ajuste* y sus *procesos de decisión*.

Mantenibilidad : Los operadores se jubilan, se ponen enfermos y precisan de bajas temporales o definitivas, se hacen viejos y encuentran dificultades para adaptarse a las nuevas situaciones del proceso. El transvase de *conocimiento* de un operador a su sustituto es difícil cuando no imposible, sobre todo cuando este transvase de información puede concluir en dificultades laborales para el operador-fuente.

Este tipo de problemas humanos están a la orden del día, pero no por ello son soslayables, ya que la carencia de los sistemas automáticos que sustituyan a los humanos determina su necesidad.

Otro aspecto que no hemos mencionado y que es fundamental es el de la seguridad. Al fin y al cabo los sistemas automáticos están contruidos para operar en un rango de situaciones determinado. La *robustez* del sistema de control es fundamental. Sin duda, si nuestra seguridad dependiera de la operación de un sistema automático, preferiríamos que éste estuviera vigilado por un humano entendido en la materia. Este tipo de situación es característico de sistemas con seguridad crítica, como puede ser una planta nuclear o un nave espacial.

De ello surge la necesidad de los sistemas de soporte de decisiones, que permiten al humano del más alto nivel tener un conocimiento lo más completo posible de la situación. Sheridan [Sheridan 83a] presenta una clasificación de niveles de automatización (cooperación hombre-máquina) que resulta interesante. Los niveles que She-

ridan propone son los que se derivan del siguiente protocolo:

1. El computador no ofrece asistencia, o
2. el computador ofrece un conjunto completo de alternativas, y
 - a) restringe la selección a un conjunto, o
 - b) sugiere una, y
 - 1) ejecuta la sugerencia si el humano la aprueba, o
 - 2) permite al humano vetarla antes de ejecutarla automáticamente, o
 - 3) informa al humano después de la ejecución, o
 - 4) informa al humano si este lo solicita, o
 - 5) informa al humano si así lo decide.

3. El computador hace todo sin comunicarse con el humano.

Desde la no automatización hasta la automatización total hay una serie de niveles de automatización basados en la cooperación hombre-máquina. Tenemos una gradación MANUAL-MIXTO-AUTOMATICO.

Como vemos el proceso de sustitución del hombre por la máquina en sistemas potencialmente peligrosos, jamás eliminará completamente al hombre, siempre habrá que poner un humano por encima de cualquier mecanismo. Esto no es necesariamente malo, sino todo lo contrario. Como trataremos de exponer en esta tesis, la colaboración *hombre-máquina* puede conducir a resultados adecuados, pero para ello es preciso construir mecanismos que posibiliten esta sinergia.

En esta tesis no se pretende pues proponer una solución de automatización total, sino presentar

una serie de técnicas que permitan potenciar el sistema de control (quizá avanzando en la jerarquía propuesta por Sheridan), pero con un objetivo final hombre-máquina.

Por lo general un sistema de control de este tipo tiene una estructura jerárquica, existiendo un nivel primario de regulación que suele estar implantado mediante autómatas programables y un nivel superior de monitorización y supervisión por el humano.

2.5. Inteligencia Artificial e Inteligencia Natural

La misión del operador es —por lo general— la de establecer estrategias de control mediante especificación de las consignas de los bucles de control. Esta misión se lleva a cabo mediante un proceso de toma de decisiones. La utilización del humano en este tipo de sistemas presenta ventajas y desventajas inherentes al comportamiento característico del mismo. Las desventajas de este tipo de control en sistemas complejos se resumen en los puntos presentados en la.

En los sistemas de control de procesos complejos se realizan dos tipos de actividades: *Conductistas* y *Cognoscitivas*. Las actividades conductistas son fácilmente implantables en un ordenador; no así las cognoscitivas, esencialmente las derivadas de la intuición y previsión. Por este motivo es preciso mantener al humano en el bucle de control, aunque se pueden desarrollar entornos que le ayuden en su labor.

Estos dos tipos de actividades están estrechamente relacionadas con las dos etapas fundamentales en cualquier sistema de control: Valoración de la situación, que es esencialmente cognoscitiva y determinación de la actuación que suele ser procedural.

Dado que parece imposible suprimir al hombre del bucle de control debido a la necesidad de los elementos que aporta (esencialmente sentido común), tradicionalmente se ha tratado de solventar esta problemática (automatización de las funciones del operador) mediante el desarrollo de sis-

temas de asistencia al humano. Ejemplos de este tipo de sistemas son los sistemas inteligentes de ayuda a la toma de decisiones. En ellos se realiza una gestión eficiente de la información disponible por el operador. De este modo el operador puede realizar sus actuaciones de una forma mucho más fiable, ya que se disminuye el elemento intuitivo de la misma. En este tipo de sistemas se produce una delegación dinámica de tareas del operador al sistema ayudante, permitiendo de este modo al operador estar en cada momento con el nivel de trabajo más adecuado. Esto es cierto salvo cuando se necesita gran cantidad de potencia en alto nivel, esto es, cuando existen varias tareas a realizar que precisan de la capacidad del humano; en esta situación el sistema ayudante no puede prestar ninguna ayuda, ya que su capacidad está limitada a las tareas más simples y repetitivas.

Ante esta situación surge la pregunta:

¿Es posible reemplazar al operador humano por un sistema artificial que presente una funcionalidad equivalente?

2.6. Inteligencia Artificial en Control

La inteligencia artificial es un grupo de técnicas muy fértil, capaz de dar soporte para solucionar problemas en los ámbitos más diversos de la informática. Por ello el control no ha escapado de la influencia de estas técnicas. El enfoque del uso de inteligencia artificial en control no ha sido nunca homogéneo, realizándose desde múltiples direcciones y con propósitos muy diferentes. Estos propósitos cabe diferenciarlos en dos grandes grupos: a) Ayudas al diseño y construcción de sistemas de control convencionales y b) control propiamente dicho.

El primer punto no entra dentro de los objetivos considerados en esta tesis, por lo que no se hará ninguna otra referencia a él, restringiéndonos al segundo punto.

De un tiempo a esta parte han venido apareciendo una serie de comunicaciones sobre la utilización de técnicas de inteligencia artificial en con-

Control Supervisor	
Ventajas	
Capacidad de determinación de acciones correctas aún con información incompleta	<i>El hombre tiene sentido común</i>
Capacidad de aprendizaje	
Desventajas	
Falta de uniformidad en la actuación	<i>El hombre no es una máquina</i>
Sobrecarga Informativa del operador	
Errores humanos: cansancio, inexperiencia, desidia, etc.	

Tabla 2.1: Ventajas y desventajas del control supervisor.

trol. Este interés parece proceder de tres fuentes diferentes:

- Ingeniería de control, que, ante la necesidad de hacer frente a problemas de control de difícil solución, tratan de encontrar en la inteligencia artificial técnicas que les permitan afrontar dicho problema.
- Inteligencia artificial, en la que el campo de aplicación se va ensanchando progresivamente.
- Grupos de gente de pensamiento sincrético que ven en la unión inteligencia artificial-control el marco adecuado para experimentar sobre temas esencialmente filosóficos.

Nuestro caso está dentro del primer grupo. El propósito que nos anima es controlar procesos complejos que se han resistido a los enfoques clásicos. La inteligencia artificial, con su enfoque heterodoxo, puede permitirnos formalizar y utilizar información sobre el proceso a controlar difícilmente integrable en un sistema clásico.

El surgir de nuestras ideas y objetivos de control es debido a una natural evolución dentro del

campo del control de procesos de producción de cemento. Esta ha sido un área tradicional de I+D dentro del departamento en que se ha desarrollado esta tesis, en colaboración con una importante empresa del sector. Las posibilidades de control clásico de este tipo de sistemas han venido mostrándose muy limitadas, al ser procesos con ruidos de todo tipo y comportamientos erráticos, fundamentalmente debidos a la variabilidad de los materiales de partida. La inteligencia artificial ofrece perspectivas nuevas para enfocar el problema del control de este tipo de procesos, y es por ello que los desarrollos presentados en esta tesis tienen una orientación decidida hacia la aplicación industrial más o menos inmediata.

Como veremos en capítulos posteriores, la automatización total de la planta —no intervención humana— será un objetivo de importancia progresivamente reducida. Otros objetivos —seguridad, estabilidad, *etc.*— nos conducirán a la idea del sistema hombre-máquina como la solución de control más adecuada; aunque la inteligencia artificial nos permitirá construir mecanismos de control muy avanzados basados en la manera en que los humanos controlan este tipo de procesos.

Inteligencia artificial en control				
	CONTROL	OPERADOR	DISEÑO	AUXILIAR
ON-LINE	Control basado en reglas	Ayudas	Análisis de Comportamientos	Diagnóstico
	Control borroso	Alarmas		Mantenimiento Preventivo
	Valoración de estado	Valoración de estado Lenguaje Natural Filtrado de acciones		
OFF-LINE	Planificación	Tutores	CAD	Diagnóstico post-mortem
	Simulación	Entrenadores	Exploración de espacios de diseño	

Tabla 2.2: Algunas aplicaciones de la inteligencia artificial en control.

2.6.1. Objetivos

Los objetivos que se tratan de conseguir con la introducción de inteligencia artificial en control se resumen en uno: *tratar de resolver determinados problemas de control frente a los que otras técnicas de control mas convencionales se han mostrado ineficaces*. El sistema que se describe en el Capítulo 7 de esta tesis se desarrolló después de explorar multiples alternativas de control de proceso incluyendo sistemas de control adaptativo-predictivo y sistemas de control borroso globales.

Las aplicaciones de la inteligencia artificial en control comenzaron ya en los primeros tiempos de la inteligencia artificial. Los primeros planificadores se desarrollaron de hecho como soporte del más alto nivel para control de robots [Fu 87]. No obstante, en el campo del control de procesos las aplicaciones pioneras se centraron en el área de sistemas de diagnóstico. La utilización en línea de estos sistemas ha sido sumamente dificultosa, debido fundamentalmente a sus malas características de funcionamiento en tiempo real; no obstante

su introducción es progresiva, empezando por sistemas de soporte de decisiones para operadores y concluyendo con auténticos sistemas de control.

2.6.2. Aplicabilidad

Si preguntamos ¿Es aplicable la inteligencia artificial en control de procesos complejos? la respuesta es inmediata: SI. Pero esta respuesta no debe conducirnos a apreciaciones erróneas. La inteligencia artificial es aplicable a cualquier cosa, y este ha sido siempre su punto fuerte; cuando no se sabe cómo resolver un problema por un método clásico se recurre a las técnicas de la inteligencia artificial. Por otra parte el punto en contra de las aplicaciones basadas en inteligencia artificial es su coste de desarrollo, comprobado fundamentalmente en el área de los sistemas expertos.

Sólo cuando la solución automatizada de un problema es difícil por los métodos clásicos y ofrece importantes promesas económicas es planteable el uso de técnicas de inteligencia artificial pa-

ra la solución del mismo.

Jakob [Jakob 90] afirma que las aplicaciones de la inteligencia artificial en este tipo de procesos que vale la pena desarrollar se caracterizan por tres factores:

- La complejidad generalizada debido a la diversidad del sistema; esto es, complejidades primaria y secundaria.
- La necesidad de tener en cuenta el tiempo.
- La necesidad de manejar incertidumbre aún en circunstancias normales; complejidad terciaria.

Sin embargo todos estos factores son tan sólo aspectos de la característica fundamental de estos sistemas: la complejidad estructural y de interacción del sistema a controlar que es el origen de un comportamiento dinámico complejo. Esta complejidad del comportamiento dinámico es la que conduce a la necesidad de incluir los factores temporal y de incertidumbre mencionados.

En resumen podemos concluir que la inteligencia artificial es aplicable en control fundamentalmente por un aspecto: su capacidad de utilizar representaciones del conocimiento heterogéneas, lo que va a permitir hacer frente a los problemas derivados de la complejidad de los procesos a controlar.

2.6.3. Reutilizabilidad

El desarrollo de sistemas de control inteligente es muy costoso, y como la mayor parte del software de alto nivel requiere en la mayoría de los casos partir de cero en su desarrollo.

La reutilización del software de control inteligente es un objetivo muy deseable por diversas razones:

Coste : El coste de desarrollo de este tipo de sistema es muy elevado, por lo que cualquier reutilización de software ya desarrollado supondrá un ahorro, siempre que la integración de tal software no sea más costosa que el propio desarrollo.

Seguridad : El software que se utiliza de manera repetida —y que se mantiene— es mucho más seguro que las aplicaciones de usar y tirar, ya que durante la utilización del mismo se descubren errores cuya corrección permite mejorar el sistema.

Tiempo : El uso de software ya desarrollado permite tener disponible el sistema de control en un plazo mucho menor, lo que redonda a su vez en el coste.

Las técnicas de inteligencia artificial son heterodoxas, y por ello permiten la adaptación muy ajustada de los mecanismos de representación a las entidades del dominio de aplicación. Esto dificulta sobremanera la reutilizabilidad de los desarrollos previos. No obstante se puede alcanzar una cierta reutilizabilidad si:

- Los desarrollos se modularizan, de forma que los elementos dependientes de la planta queden separados del resto del sistema. Esto es en el fondo lo que propugna la tecnología de los sistemas expertos, pero el problema del control de un proceso determinado requiere a menudo de enfoques particulares que se apartan del paradigma motor de inferencia - base de conocimiento (MI-BC).
- Se desarrollan los sistemas orientándolos a tipos de procesos, no tratando de realizar excesivas generalizaciones —que conducen a falta de prestaciones— ni particularizaciones —que limitan la reutilizabilidad.

2.7. Campos de Aplicación

Los campos de aplicación varían en un amplio espectro [Bartos 89], desde la operación fuera de línea hasta la aplicación en línea. Desde los sistemas de ayuda a los operadores hasta la implantación de reguladores basados en reglas. Una lista indicativa de las aplicaciones posibles es la siguiente:

- Sistemas de ayuda [Cross 86].
- Tutores inteligentes [Yazdani 86a][Sleeman 82][Brown 82].

- Sistemas de diagnóstico post-mortem [Cuenca 89].
- Sistemas de diagnóstico en línea [Karel 89][Scarl 87].
- Ayudas al diseño (CAD) [Hinkelmann 89][Lan 90].
- Sistemas de alarma inteligentes [Henkind 87].
- Reguladores basados en reglas [Bernard 88][Shirley 87].
- Reguladores borrosos [Efstahtiou 87][Cerezo 88][Tong 77].
- Sistemas de diálogo e interfases en lenguaje natural [Ehrich 83].
- Sistemas de mantenimiento [Cuenca 89].
- Sistemas de detección de averías [Kitowsky 87][Scarl 87].
- Predicción de comportamiento [D'Ambrosio 89a][Widman 89b].
- Valoración de estado del sistema [Bailey 88].
- Filtrado de acciones de operadores [Johansen 83].
- Interpretación de información de bajo nivel [Forbus 87][Pau 89].

Estas aplicaciones se pueden agrupar según varios criterios, que permiten caracterizar su funcionamiento:

- Modo de operación:
 - En línea.
 - Fuera de línea.
- Orientación
 - Operador: Orientada a operador.
 - Control: Orientada a proceso.
 - Auxiliar: Soporte de sistema.

Capítulo 3

Modelos Simbólicos de Sistemas Físicos

Experience has shown that designing a good representation is often the key to turning hard problems into simple ones

*Patrick Henry Winston
Artificial Intelligence, 1984*

3.1. Introduction

En un sistema de control es de fundamental importancia el conocimiento sobre el sistema a controlar. Este conocimiento se emplea de forma *implícita* o *explícita* en dicho sistema de control. En el caso de sistemas de control avanzados la representación de esta información es el factor más crítico y que determina las prestaciones del sistema. En el caso de un sistema de control clásico esta representación es simple y directa: un modelo matemático realizado mediante ecuaciones diferenciales. Este modelo se utiliza de forma implícita en el cálculo en la mayoría de los reguladores, y de forma explícita en alguno de ellos (MRAC).

Cuando el proceso es muy complejo la posibilidad de usar este tipo de representaciones ecuaciones diferenciales es remota. Es preciso utilizar otros enfoques como puede ser el de la inteligencia artificial. En este caso las necesidades del sistema de representación son extremas, al surgir la necesidad de manejar información en niveles de abstracción muy diferentes.

3.2. Modelos

Los modelos son parte fundamental en todos los sistemas que interaccionan, ya que son parte de los mecanismos que conducen dicha interacción. Las personas interaccionan con el mundo, y por ello precisan de modelos del mismo que les permitan interpretar lo que les llega por los sentidos percepción. Las máquinas, en particular los sistemas autónomos que realizan funciones cognoscitivas, precisan también de estos modelos para realizar de forma adecuada su función.

3.2.1. ¿Qué es un Modelo?

Según la Encyclopædia Britannica un *modelo* es una representación simplificada del mundo real, y, como tal, incluye solo las variables relevantes para el problema que se maneja. Si se modela un cuerpo en caída libre no parece necesario incluir el color del mismo en el modelo. Aún mas, un modelo puede no incluir todas las variables relevantes si con solo algunas de ellas responde suficientemente bien a los fenómenos a explicar. El modelo queda mermado en su precisión, pero sigue siendo útil.

Otras definiciones de *modelo* son las siguientes:

“Cosa en que alguien se fija para hacer otra igual” [Moliner 89]. Este es el sentido mundano del término.

“Un sistema artificial con el que se pretende reproducir, o imitar, el comportamiento de un

sistema concreto” [Aracil 86]. Este es el sentido sistemista.

Para Rothemberg [Rothemberg 89] modelar es hacer un uso de una cosa en vez de otra con un propósito cognoscitivo y de manera económicamente eficiente.

El término *modelo* aparece dentro de un proceso de representación, pero no obstante se utiliza con dos significados fundamentalmente opuestos como aparece reflejado en las dos anteriores definiciones [Aracil 86]:

- La cosa representada.
- La cosa representante.

La segunda acepción es la utilizada habitualmente en el ámbito técnico, quedando la primera para otros usos.

Los modelos son construidos por las personas, y por ello, un modelo no es más que una construcción subjetiva en la cual el humano reúne su experiencia sobre un entidad real. El término clave es *subjetivo*. La subjetividad del modelo es debida a que se construye mediante un proceso cognoscitivo llevado a cabo por un humano. Los intentos de construcción de sistemas de percepción del entorno para sistemas autónomos se encuentran con la problemática de la formulación del modelo que servirá de base a dicha percepción.

De especial importancia son los modelos matemáticos. Esta importancia radica en su *comunicabilidad*. Un modelo subjetivo está asociado estrechamente al sujeto que lo desarrolla, siendo de extrema dificultad su comunicación. Si el ente a modelar admite la formulación matemática del modelo, se puede construir un modelo *compartible* por varios individuos. Los modelos simbólicos¹ que se desarrollan en inteligencia artificial tienen esta característica. Están construidos en un nivel de abstracción mayor que el numérico pero son comparables.

La posibilidad de formular matemáticamente los modelos es muy relativa, restringiéndose a sis-

¹Cuando se utiliza el término **simbólico** se debe entender como no numérico.

temas de complejidad limitada. Las técnicas matemáticas avanzan, sin embargo, en el desarrollo de técnicas de modelado progresivamente eficaces en la construcción de modelos matemáticos de sistemas complejos [MacFarlane 70].

3.2.2. Características de los modelos

El modelado no es una actividad susceptible de estandarización; como ya se ha mencionado los modelos son el resultado de una actividad cognoscitiva de un humano, y por ello son susceptibles de sufrir grandes variaciones de una persona a otra. Aún en el caso de modelar un sistema simple los modelos pueden diferir.

Es preciso analizar cuales son las características fundamentales de los modelos y que nos van a permitir analizarlos. Estas características son las relacionadas fundamentalmente con los aspectos técnicos referentes a esta tesis, sin que se trate de realizar un estudio exhaustivo de las mismas.

El primer aspecto característico de un modelo es su *profundidad*. La profundidad de un modelo está relacionada con el proceso de abstracción mediante el cual se obtiene el modelo a partir del sistema real. Un modelo será tanto más profundo cuantos más aspectos relacionados con el comportamiento del sistema real estén expresados en el modelo. En este sentido Hinkelmann [Hinkelmann 89] dice:

“Consider two models of expertise M and M' . We will say that M' is deeper than M if there exists some implicit knowledge in M which is explicitly represented or computed in M' ”²

De acuerdo con Hinkelmann podemos formular un principio de profundización de modelos, afirmando que profundizar es explicitar:

PROFUNDIZACION \equiv EXPLICITACION

Los modelos pueden tratar de representar el sistema tal como es, o pueden intentar exclusivamente responder a determinados aspectos. En este sentido diremos que un modelo es *directo* u *orientado*. Un modelo orientado es un modelo que no tra-

²Consideremos dos modelos de pericia M y M' . Diremos que M' es más profundo que M si existe algún conocimiento implícito en M que es representado explícitamente en M' .

ta de reproducir el comportamiento fundamental del sistema modelado, sino que trata *exclusivamente* de generar una representación de determinados aspectos del sistema. En cierta medida todos los modelos son modelos orientados, ya que todos se construyen con un objetivo particular en mente, y es con este objetivo en el horizonte como se construye el modelo del sistema. No obstante entre dos modelos es por lo general posible determinar cual es más orientado a una aplicación determinada.

Este tipo de representaciones que modelan exclusivamente el comportamiento sin hacer referencia a la manera en que este se produce se denominan modelos de comportamiento o someros. Por contra los modelos que hacen uso de representaciones basadas en dependencias causales se los denomina modelos causales o profundos. Este es el sentido de la distinción que hace Klir entre modelos de comportamiento y modelos UC y ST [Klir 69].

Un ejemplo muy claro de orientación de modelos se da en los sistemas de diagnóstico. En diagnóstico se usan dos tipos de modelos [Hinkelmann 89]:

Fisiológicos: tratan de representar el sistema de forma que se pueda responder del comportamiento normal del mismo.

Patofisiológicos: tratan de responder al comportamiento anormal del sistema. En este caso el modelo se realiza por lo general en base a una lista de faltas establecidas a priori.

Esta distinción es clara en modelos someros, pero en modelos profundos no lo es tanto, ya que la patología es en esencia una modificación del comportamiento fisiológico normal. Con modelos profundos la patología se representa por medio de una alteración del modelo, una *alteración fisiológica*. El modelo debe ser lo suficientemente profundo como para representar todas estas posibles alteraciones, etiologías del comportamiento patológico.

Como ya hemos comentado con anterioridad un modelo será tanto más explícito cuanto más profundo sea. Explicitar supone declarar los mecanismos subyacentes que responden del comportamiento, sin dejar la justificación del mismo a representaciones ad hoc.

El modelo se puede construir de forma que exprese el comportamiento del sistema solo desde el punto de vista de las entradas y salidas del mismo o del comportamiento interno [Fernández 77]. Estos dos tipos de modelos se denominan respectivamente modelos *externos* e *internos*. En el primer caso se considera el sistema como una caja negra, y lo que se pretende con el modelado es construir otro sistema que se comporte de la misma forma por lo que a magnitudes medibles se refiere, esto es, debe responder con las mismas salidas a las mismas entradas. Por contra en los modelos internos lo que se pretende es conseguir un conjunto de variables internas que respondan a variables reales del sistema modelado. De este modo se logra también una reproducción del comportamiento externo en base a un modelo interno del sistema.

Los medios para construir los modelos son diversos, pasando desde las ecuaciones diferenciales [MacFarlane 70] a los modelos por lotes basados en autómatas finitos [Zeigler 90] y a los modelos cualitativos [Bobrow 84]. El uso que se pretende hacer de los modelos determina en gran manera su construcción.

Los modelos se pueden clasificar en dos grandes tipos de acuerdo con el tipo de sistema que modelan: los modelos de *sistemas continuos* y los modelos *por lotes*. En esta tesis nos centraremos específicamente en modelos de sistemas continuos, que son los mas aplicables en la industria de proceso. En estos modelos se trata de representar el comportamiento del sistema en base a una serie de variables que evolucionan en el tiempo de forma continua.

Las técnicas de modelado aplicables son muchas, situándose en niveles de abstracción diferentes. Así los modelos basados en ecuaciones diferenciales son los modelos menos abstractos, situándose en el extremo opuesto los modelos para simulación basada en el conocimiento.

3.2.3. Modelos informáticos

La informática se basa en la construcción de modelos del mundo, ya que la información no es mas que eso: modelo de la realidad. Los programas informáticos tratan de hacer frente a proble-

mas de la realidad mediante el manejo de información estructurada referente a ese mundo. Esta información estructurada no es mas que un *modelo* de la porción del mundo con la que se está trabajando. Esto aparece claramente ya en [Lehman 80], en el que se clasifican los tipos de programas en función de el tipo de modelo y la posibilidad de su complección. La inteligencia artificial es en esencia un enfoque heterodoxo para la construcción de modelos informáticos del mundo, y es desde esta perspectiva heterodoxa como las aplicaciones basadas en inteligencia artificial —léase fundamentalmente sistemas expertos— han conseguido su éxito.

En cualquier tipo de aplicación informática existe un modelo de la porción del mundo con la que trabaja. Este modelo puede formularse explícitamente o estar incluido de manera implícita en los procedimientos o estructuras de datos que componen la aplicación informática. Para realizar cualquier tipo de actividad sobre un objeto, persiguiendo unos determinados fines, es preciso conocer como va a reaccionar dicho objeto a nuestras acciones. Este conocimiento del comportamiento del objeto es un modelo de dicho objeto. Este modelo aparecerá representado de forma implícita en los mecanismos que implantemos para la intervención sobre el mismo.

El control, por otra parte, se basa también en modelos del mundo, aunque en este caso los modelos suelen tener dos características muy especiales:

- Su formulación matemática; suelen ser expresados mediante ecuaciones diferenciales.
- Su restringido alcance; modelan pequeñas porciones del mundo, esto es sistemas poco complejos.

Vemos que el ámbito de aplicación del control clásico son los sistemas simples y que admiten representaciones matemáticas, en los cuales aunque existe una cierta flexibilidad por lo que se refiere a la formulación matemática no sucede lo mismo con la simplicidad. El control de sistemas complejos es una asignatura pendiente, sobre todo cuando el sistema tiene complejidades secundaria y terciaria elevadas (ver Capítulo 2). Para comprobar

este punto no tenemos mas que observar la técnica habitual de hacer frente a sistemas complejos: reducción y desacoplamiento.

En esta tesis doctoral se trata de aplicar el punto de vista heterodoxo de construcción de modelos —tomado de la inteligencia artificial— a la construcción de modelos de sistemas con complejidad terciaria, para su utilización en control de los mismos.

3.3. Modelos para Control

En esta tesis se desarrolla una arquitectura de control que está basada en la utilización en línea de un modelo del proceso a controlar. Este tipo de técnica se enmarca dentro de lo que en ingeniería del conocimiento se ha dado en llamar *razonamiento basado en modelos*. En ocasiones se suele utilizar el término *control basado en el conocimiento* para referirse a una serie de métodos de control basados en técnicas tomadas de la ingeniería del conocimiento. El uso de este tipo de expresión es quizá inadecuado, ya que para controlar cualquier proceso hace falta un cierto *conocimiento* del mismo en el que se basa el sistema de control.

En la terminología sobre los tipos de control existen una serie de usos de la misma que conducen a confusiones y errores de interpretación.

El control se basa siempre en el conocimiento sobre el proceso a controlar, conocimiento que constituye un *modelo* del proceso. Por ello todos los controles serían agrupables bajo el término de *control basado en el conocimiento* o *control basado en modelos*. A pesar de lo mencionado, y para ajustarnos a los usos más generalizados, cuando utilizemos el término *control basado en modelos* nos referiremos al control basado en modelos *simbólicos explícitos*. En esta última frase es preciso aclarar el término *simbólico*. En este caso sucede lo mismo que con el *conocimiento*; habitualmente se usa el término *simbólico* en contraposición a *numérico*, y este será el uso que le demos, teniendo en cuenta no obstante que los números también son símbolos.

En la Tabla 3.3 se presentan algunos ejemplos de control, clasificados según el tipo de modelo

Ejemplo de Usos de Modelos en Control		
Tipo de Modelo	Explícito	Implícito
Simbólico	Control Basado en Modelos (CBM)	Control basado en Reglas (CBR)
Númérico	Control adaptativo con modelo de referencia (MRAC)	Control Proporcional, Integral, Diferencial (PID)

Tabla 3.1: Ejemplos de uso de modelos considerando dos aspectos fundamentales: explicitud y abstracción.

empleado: simbólico versus numérico y explícito versus implícito. En dicha tabla se presentan los siguientes ejemplos:

PID: Control PID. Es quizá el control más clásico. En él el modelo es implícito, ya que se utiliza para el diseño del regulador, y por tanto podemos afirmar que está *incluido* en él. Además este modelo es numérico, expresable en forma de ecuaciones diferenciales o de funciones de transferencia.

MRAC: Control Adaptativo con Modelo de Referencia. (Model Reference Adaptive Control³). Un tipo de control adaptativo en el que se formula de manera explícita un modelo que es empleado durante la actividad de control. Este modelo es al igual que en el caso del PID numérico, aunque no es un modelo del proceso a controlar sino de como nos gustaría que fuera el proceso. Otro tipo de control adaptativo, el STR, utiliza también un modelo del proceso, que es objeto de un ajuste periódico.

CBR: Control Basado en Reglas. En este caso el sistema de control se basa en un conjunto de reglas que constituyen la base de conocimiento de control. Estas reglas se extraen del conocimiento sobre control del proceso que tiene

una determinada persona, siendo por lo tanto una concreción informática de un modelo mental. En este caso el modelo es simbólico e implícito en las reglas, del mismo modo que era implícito en el PID.

CBM: Control Basado en Modelos. En este caso el modelo del proceso a controlar aparece de forma explícita en la base de conocimiento del sistema de control. Este modelo es simbólico en el sentido más general, pudiendo tener toda una gama de niveles de resolución: desde el modelo numérico ecuaciones diferenciales hasta un modelo en un nivel de abstracción máximo grafo de dependencias causales. Este modelo se empleará en mecanismos de razonamiento basado en modelos (RBM) para obtener las acciones de control.

3.3.1. Utilización de modelos numéricos

Los modelos utilizados en el desarrollo de los reguladores clásicos son modelos basados en ecuaciones diferenciales ordinarias. Por ejemplo, en el sistema de la figura un objeto en caída libre se tiene el siguiente juego de ecuaciones diferenciales:

$$\frac{dy}{dt} = v$$

³Se ha respetado el acrónimo anglosajón por estar muy extendido su uso.

$$\frac{dv}{dt} = a$$

$$a = cte = -g$$

Este conjunto de ecuaciones constituye un modelo del sistema lo suficientemente completo como para representar el comportamiento del mismo de interés para nosotros. La solución de la ecuación diferencial constituye una descripción de la evolución temporal del sistema al nivel de representación que hemos formulado el modelo caída libre, en el vacío, etc.

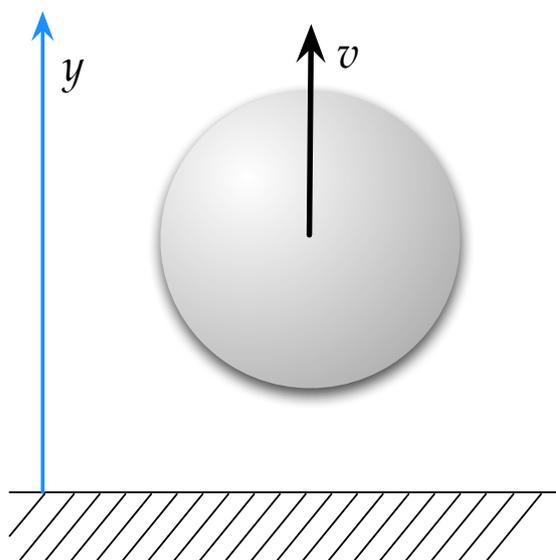


Figura 3.1: Objeto en caída libre.

El desarrollo de sistemas de control clásicos se basa fundamentalmente en el conocimiento de un modelo del proceso [MacFarlane 70]. Este modelo se expresa mediante ecuaciones diferenciales ordinarias o bien mediante ecuaciones en diferencias. Los métodos de construcción de reguladores utilizando estos modelos implícita o explícitamente no es objetivo de esta tesis, pudiendo encontrarse multitud de referencias sobre el tema [Åstrom 88][Åstrom 89b]. Profundizar aquí.

3.3.2. Utilización de modelos simbólicos implícitos

Los sistemas de control basados en reglas se construyen mediante un proceso de *adquisición de conocimiento*, mediante el cual se extrae de los expertos en control de un determinado sistema las reglas de control *que utilizan*. Estas reglas son en términos simples de la forma:

SI<Situación>

ENTONCES<Acción>

donde se trata de expresar protocolos de control similares a los que emplean los humanos.

De este modo se realiza un sistema de control en el que el modelo del proceso no aparece explícitamente, sino que este modelo es una representación informática del modelo subjetivo que el experto tiene en su mente. En las reglas tenemos una representación implícita del mismo; esto se puede comprobar si se observa que de las reglas se puede deducir, en parte, el comportamiento del sistema. El modelo implícito contenido en las reglas es una simplificación del modelo mental del experto. Esta simplificación se hace por abstracción del modelo, reteniendo sólo los aspectos de comportamiento del mismo relevantes al control. Esta abstracción se realiza en un determinado nivel de representación.

Consideremos por ejemplo un sistema simple: un par de grifos de agua fría y caliente de una ducha. El problema del control en este caso es conseguir dos objetivos simultáneamente: una temperatura adecuada y una cantidad suficiente de agua. En este caso los expertos somos prácticamente todos, por lo que será más fácil comprender el mecanismo de obtención de las reglas de control.

Las reglas de control en este caso se formulan con dos variables en las premisas y dos acciones en las conclusiones. La base de conocimiento de control de este sistema será algo así como el conjunto de reglas representado en la Figura 3.2.

En este caso hemos obtenido nueve reglas (3^2) en función de dos variables en las premisas y tres posibles valores *simbólicos* para cada una de ellas (alto, normal, bajo). Esta base de conocimiento está construida en un determinado nivel de

```

SI la temperatura es baja y el caudal es bajo
ENTONCES aumentar agua caliente

SI la temperatura es baja y el caudal es normal
ENTONCES aumentar agua caliente y disminuir el agua fría

SI la temperatura es baja y el caudal es alto
ENTONCES disminuir el agua fría

SI la temperatura es normal y el caudal es bajo
ENTONCES aumentar agua caliente y aumentar agua fría

SI la temperatura es normal y el caudal es normal
ENTONCES no hacer nada

SI la temperatura es normal y el caudal es alto
ENTONCES disminuir agua caliente y disminuir agua fría

SI la temperatura es alta y el caudal es bajo
ENTONCES aumentar agua fría

SI la temperatura es alta y el caudal es normal
ENTONCES disminuir agua caliente y aumentar agua fría

SI la temperatura es alta y el caudal es alto
ENTONCES disminuir agua caliente
    
```

Figura 3.2: Base de conocimiento de control del sistema de dos grifos.

abstracción, correspondiendo a un *modelo* en el que la temperatura puede ser alta, baja o normal. Es posible construir bases de conocimiento de control en base a utilizar un modelo con otro nivel de resolución. Los modelos mentales que emplean las personas tiene la propiedad de multi-resolucionalidad, lo que permite extraer modelos con diferentes niveles de abstracción. El juego de reglas se formularía, por ejemplo, en base a los siguientes valores para la temperatura: *MUY ALTA*, *ALTA*, *POCO ALTA*, *NORMAL*, *POCO BAJA*, *BAJA* y *MUY BAJA*. De este modo, y manteniendo los tres valores simbólicos para el caudal el número de situaciones distintas reconocibles en este nivel de abstracción sería de: $7 \times 3 = 21$; con lo que dispondríamos de 21 reglas posibles. Probablemente en este caso nos viéramos conducidos a utilizar más valores simbólicos para las variables de actuación (las que aparecen en las conclusiones de las reglas).

3.3.3. Utilización de modelos simbólicos explícitos

Si nos paramos a pensar un momento veremos que cuando nosotros humanos resolvemos un problema como el planteado en la ducha, no utilizamos un juego de reglas como las presentadas en el punto anterior, sino que utilizamos un modelo del sistema para averiguar cómo debemos actuar para conseguir unos determinados fines.

El modelo mental se estructura en torno a las variables características del sistema e incluye dependencias entre ellas. De este modo en el modelo para la ducha tendremos entre otras una variable denominada *Temp* y una variable denominada *Q_{Caliente}*. La primera hace referencia a la temperatura del agua que sale por la ducha y la segunda al caudal de agua caliente, controlado por el correspondiente grifo. Existe en dicho modelo también una *relación* entre ambas variables, que expresa la relación de causalidad existente entre la variación de caudal de agua caliente y la variación de temperatura del agua de la ducha. Esta relación podría expresarse como⁴:

⁴M⁺ corresponde a la terminología de Kuipers [Kuipers 84].

$$Temp = M^+(Q_{Caliente})$$

expresión en la que se ha empleado la notación de Kuipers [Kuipers 84] para representar la dependencia monótona creciente. Con ella queremos expresar que, de mantenerse las restantes variables iguales, un incremento en $Q_{Caliente}$ conlleva un incremento en $Temp$. Esto es si aumentamos el agua caliente, dejando el grifo de la fría igual, la temperatura del agua de la ducha aumentará. Del mismo modo podríamos expresar las dependencias entre las demás variables.

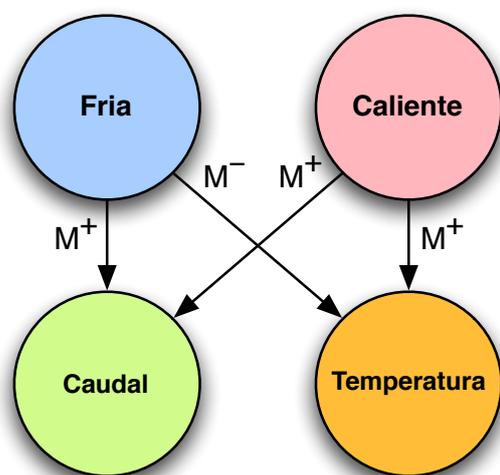


Figura 3.3: Modelo cualitativo del sistema de dos grifos caliente/frío.

Aparte de las dependencias funcionales (M^+ y M^-) existen otro tipo de restricciones más rígidas, por ejemplo que los caudales tienen sus límites. Los modelos tienen rangos de validez limitados, *regiones de operación* en las cuales el modelo es válido, pero que fuera de ellas el modelo adecuado que refleja mejor las características del sistema es diferente. Estas regiones están limitadas por restricciones a los valores de las variables del sistema. En nuestro caso, cuando el caudal de agua fría es nulo ($Q_{Fría} = 0$) la ecuación de dependencia funcional 3-1, deja de tener validez, ya que si no hay agua fría la $Temp$ es igual a la temperatura

del agua caliente, independientemente de lo que valga el caudal de la misma ($Q_{Caliente}$).

Debemos ver que la determinación de la acción de control no se lleva a cabo en este caso mediante reglas que especifiquen ésta en función de la situación, sino que se obtiene a través de un proceso de razonamiento causal. Este proceso conlleva una *exploración hacia atrás* del grafo de dependencias causales.

Así en vista de ese grafo vemos que si queremos bajar la temperatura, las posibilidades son dos:

1. Bajar el caudal $Q_{Caliente}$ (en virtud de la relación M^+ bajará $Temp$).
2. Subir el caudal $Q_{Fría}$ (en virtud de la relación M^- bajará $Temp$).

El determinar cual de las dos opciones hay que seguir se realizará en base a la otra restricción, la que fuerza el caudal deseado.

Vemos pues que no tenemos en nuestra base de conocimiento pares (situación, acción) como teníamos en el caso de usar modelos implícitos, sino que tenemos relaciones más abstractas pero en relación mucho más inmediata con el mundo a controlar.

3.4. Modelos Cualitativos

El tipo de representación utilizado en el último apartado se enmarca dentro de lo que se denomina modelado cualitativo. Este tipo de técnica tiene su origen en los intentos de formular mecanismos de razonamiento sobre sistemas físicos. La primera propuesta formal en este sentido se debe a Hayes [Hayes 85a], que en su "*Naïve physics manifesto*" propone una línea de investigación teórica en el campo del razonamiento de sentido común sobre sistemas. Cuando un humano razona sobre un sistema real gran parte de los conocimientos que emplea son conocimientos prefísicos, adquiridos durante la experiencia cotidiana. Este tipo de conocimiento es de muy difícil formalización, tanto por lo forma intrínseca del mismo, como por la dificultad inherente a su localización e identifica-

ción dentro de los procesos de razonamiento humano.

Fruto de esta propuesta son una serie de desarrollos en los que se proponen técnicas de modelado y simulación cualitativa de sistemas. Gran parte de estos trabajos aparecen referenciados en [Bobrow 84], destacando por su importancia los de de Kleer [de Kleer 84a], Forbus [Forbus 84] y Kuipers [Kuipers 86].

Dichos sistemas pueden realizar predicciones de comportamiento de un proceso determinado, en base a una representación de la estructura del proceso en términos de una serie de funciones cualitativas y restricciones y un estado inicial. El elemento característico son dichas funciones cualitativas. Estas son funciones del tiempo que toman valores sobre conjuntos finitos de valores cualitativos denominados espacios de cantidades. El espacio de cantidades más simple es el compuesto por los valores $\{negativo, cero, positivo\}$ que en ocasiones se representa por $\{-, 0, +\}$. Este espacio es el más utilizado y en ocasiones el único, como es el caso de de Kleer, ya que es el que aporta información cualitativa mas importante.

El paradigma fundamental de la simulación cualitativa es la generación y prueba. El mecanismo de simulación genera un grafo con los posibles estados sucesores de uno dado, grafo que es podado en base a las restricciones. De este modo se genera un grafo de comportamientos, en el que los nodos son estados y el comportamiento del sistema es cualquier trayectoria desde el nodo inicial a un nodo terminal. La simulación cualitativa tiene este rasgo característico de multiplicidad, sin que sea posible determinar efectivamente cual sería la trayectoria real. Enfoques basados en ponderaciones tratan de solventar este problema, como es el caso del de D'Ambrosio [D'Ambrosio 89], que usa técnicas basadas en conjuntos borrosos.

El objetivo final de todos estos sistemas es la derivación de comportamientos a partir de información estructural. Esto es un paso de la formulación de modelos de comportamiento someros a la formulación de modelos causales profundos. Estos modelos estructurales deben estar libres de asunciones a priori, de forma que se logre el objetivo de componibilidad, de acuerdo con el principio formulado por de Kleer de "estructura sin fun-

ción⁵" [de Kleer 84a].

Utilizando la formulación de Kuipers [Kuipers 86] el ejemplo anterior de un objeto en caída libre (ver Figura 3.3.1) con velocidad vertical positiva tendría la siguiente formulación. El comportamiento del sistema viene determinado por las variables del mismo, las restricciones (ecuaciones diferenciales y constancia de la gravedad) y el estado inicial:

Tres variables:

y : Posición vertical.

v : Velocidad vertical.

a : Aceleración vertical.

Cuyos espacios de cantidades son:

$$Q(y) = \{-, 0, +\}$$

$$Q(v) = \{-, 0, +\}$$

$$Q(a) = \{-, 0, +\}$$

Las restricciones son:

$$\frac{dy}{dt} = v$$

$$\frac{dv}{dt} = a$$

$$a = cte = g < 0$$

El estado inicial es:

$y = [+ , inc]$ Posición >0 y aumentando.

$v = [+ , dec]$ Velocidad >0 y disminuyendo.

$a = [g , std]$ Aceleración constante.

La simulación de este sistema nos permitiría determinar los posibles comportamientos, que en este caso sería esencialmente único dada la simplicidad del problema. El comportamiento de este sistema es el que se representa en la Figura 3.4.

⁵En el original "no function in structure".

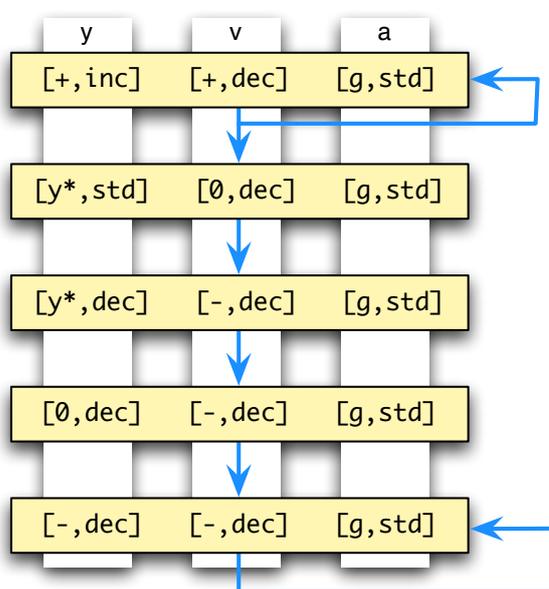


Figura 3.4: Ejemplo de simulación del modelo cualitativo del objeto en caída libre a partir de una cierta altura y velocidad inicial hacia arriba. Las cajas amarillas representan los diferentes estados cualitativos por los que pasa el sistema.

Capítulo 4

Control Inteligente de Procesos

... capability of a system to expose a variety of features which traditionally has been out of the scope of specialists in conventional control theory

Alexander Meystel
IEEE Workshop on Intelligent Control 1985

en el capítulo 5 una metodología de desarrollo de arquitecturas que permita hacer frente a este problema. En el capítulo 7 se presentará una arquitectura de control —objetivo principal de esta tesis— basada en esta metodología.

4.1. Introducción

Hemos visto en el capítulo 2 el tipo de problemas que deseamos resolver: control de procesos complejos. El enfoque tradicional de control es quizá demasiado restringido para hacer frente a determinados problemas; su potencia esta reñida con la flexibilidad.

La inteligencia artificial permite obtener —en circunstancias determinadas— resultados que se aproximan a los obtenidos por humanos. Dado que los humanos son capaces de controlar este tipo de procesos ¿Por que no tratar de emular esta capacidad?.

El enfoque heterodoxo de resolución de problemas proporcionado por la inteligencia artificial nos permitirá atacar el problema de control desde una perspectiva múltiple, utilizando información de las mas diversas fuentes y de los niveles de abstracción más dispares. Todo ello nos conducirá al concepto de control inteligente.

En este capítulo se realizará una presentación a fondo de las características de los sistemas de control inteligente, sus fundamentos y los objetivos perseguido en su desarrollo.

Se analizará la problemática de la operación en tiempo real de este tipo de sistemas, proponiendo

4.2. Control Inteligente: Objetivos y Definiciones

No es fácil dar una definición de *control inteligente*; para comprobarlo basta con las siguientes concepciones de diferentes autores [Saridis 85]:

G. Saridis: Propone un modelo probabilístico —entropía— del control inteligente, basado en el principio de Inteligencia Creciente con Precisión Decreciente.

Y.H. Pao: Define el control inteligente a través de la inteligencia artificial, como una combinación de las formulaciones analíticas del pasado con las capacidades numéricas de la era de los computadores.

Y.C. Ho: Propone la visión del control inteligente desde el punto de vista de la investigación operativa, como el estudio de los sistemas dinámicos con eventos discretos.

A. Bejczy: Recomienda una formulación analítica por extensión de la teoría de control no lineal.

J. Albus: Sugiere el enfoque de utilizar el modelo biológico del cerebelo como modelo de control jerárquico inteligente.

Prácticamente, a todo lo que no es PID se le ha llamado control inteligente. Para lograr una definición un poco más clara que las anteriores daremos un pequeño rodeo, que nos conducirá a plantear la necesidad de una cierta técnica, tecnología o metodología de desarrollo de sistemas de control.

4.2.1. Objetivos generales del control inteligente

El progreso en la automatización de sistemas se ha dado de manos de los técnicos y de los matemáticos a lo largo del tiempo. La fuerza impulsora de este tipo de avance ha sido la motivación económica. El coste creciente del personal lleva a la búsqueda de mejoras de rendimiento en la automatización de los procesos productivos, junto con mejoras de seguridad y calidad.

De la mano de la teoría de control automático se han venido logrando grandes éxitos en los últimos decenios. Pero, de un tiempo a esta parte, los problemas de control de determinados sistemas mantienen su imbatibilidad, a pesar del fárrago matemático desarrollado últimamente. Estos sistemas se distinguen por tener comportamientos sumamente anormales, caracterizados fundamentalmente por la fuerte no linealidad o la variabilidad del sistema. El control de este tipo de *sistemas complejos* se ha mostrado como una asignatura pendiente de la teoría del control automático.

En este punto surge la inteligencia artificial como una opción a tener en cuenta [Åstrom 89a] [Åstrom 89a]. Los sistemas basados en inteligencia artificial se caracterizan por su elevado coste de desarrollo y por la imposibilidad práctica de determinar a priori los resultados que se van a derivar de su operación. Pero aún así sistemas de este tipo se están desarrollando, no sólo en el ámbito académico, más dado a las exploraciones de futuro incierto, sino también en el industrial.

El control inteligente trata de hacer frente a problemas de control que se han mostrado insolubles con las técnicas analíticas clásicas y modernas. Estos sistemas de control inteligente se aplican cuando las prestaciones de los sistemas de control convencional no son lo suficientemente buenas. Esto se da especialmente en las siguientes cir-

cunstancias [Ilari 87]:

- Procesos fuertemente no-lineales.
- Procesos con incertidumbre en su comportamiento dinámico o con derivas de los parámetros de proceso.
- Procesos con variables ocultas, esto es variables no medibles.
- Procesos con sensibilidad elevada.
- Procesos con perturbaciones aleatorias intensas.
- Procesos con retardos elevados.

El utilizar técnicas de inteligencia artificial se debe no a su *inteligencia* sino a que la inteligencia artificial se ha mostrado tradicionalmente eficaz con problemas reticentes; no por que su enfoque sea *inteligente*, sino porque es —por lo general— un enfoque de fuerza bruta que tarde o temprano conduce a algo.

4.2.2. Definición

En este punto podemos dar una definición de control inteligente:

El control inteligente comprende una serie de técnicas —tomadas fundamentalmente de la inteligencia artificial— con las que se pretenden resolver problemas de control inabordables por los métodos clásicos.

En esta definición cabe destacar que *control inteligente* no es exclusivamente inteligencia artificial, sino que es un enfoque heterodoxo para la resolución de problemas de control. En los sistemas de control inteligente los mecanismos numéricos y los simbólicos van de la mano en busca de soluciones. El acoplamiento simbólico-numérico es una de las tarjetas de visita de los sistemas de control inteligente.

La otra característica de estos sistemas es que son sistemas basados en inteligencia artificial y *conectados al mundo*. Las aplicaciones de la inteligencia artificial, verbigracia los sistemas expertos, se han caracterizado fundamentalmente por su desvinculación del mundo. Se encuentran en

contacto con este casi exclusivamente a través de una interfase humana, que los aísla en su campana de immaculado razonamiento simbólico.

Por comparar podemos citar a Meystel [Meystel 86], que afirma que la primera definición de control inteligente —debida a Saridis— es:

“... capability of a system to expose a variety of features which traditionally has been out of scope of specialists in conventional control theory: decision making, games, image recognition, adaptation to uncertain media, self-organization, planning and scheduling of operations, etc.”¹

El control se lleva a cabo no exclusivamente por medio de un modelo matemático del proceso a controlar, sino que el centro del sistema de control inteligente es una *representación del mundo*, limitada al proceso y basada en un conocimiento extenso sobre el proceso y el tipo de operaciones a realizar. Veremos que el control inteligente admite toda una gradación desde el simple PID hasta el sistema más complejo, robusto, tolerante, etc. La adición de capas con funciones avanzadas permitirán implantar sistemas de control eficaces en una gran variedad de situaciones.

De hecho, algunos autores propugnan que lo adecuado es hablar de sistemas de control autónomos [Antsaklis 89], siendo el uso de la inteligencia artificial el medio para lograr esta autonomía. Esta concepción es sumamente acertada, y nosotros la suscribimos. No obstante el término *control inteligente*² está tan difundido que se utilizará prácticamente como sinónimo de control autónomo.

4.3. Perspectiva Histórica

El control inteligente tiene sus inicios en el final de los años 70, apareciendo como un área de in-

¹... capacidad de un sistema para presentar una serie de características que han estado tradicionalmente fuera del alcance de los especialistas en teoría de control convencional: toma de decisiones, juegos, reconocimiento de imágenes, adaptación a entornos inciertos, autoorganización, planificación y programación de operaciones, etc.

²Es opinión del autor que la tendencia al uso de *control inteligente* frente a *control autónomo* es debida fundamentalmente a un factor histórico y en algunos casos a un criterio comercial.

vestigación científico-técnica con visos de filosofía. Este área se sitúa en la intersección de toda una gama de tecnologías, que, según Saridis [Saridis 83], son la inteligencia artificial, el control y la investigación operativa. Lo cierto es que el control inteligente no es la intersección de estas tecnologías, sino que es mas bien la unión de las mismas y de muchas otras; probablemente los objetivos que se pretenden conseguir son tan ambiciosos que la fusión de todas las técnicas informáticas no sea suficiente para lograrlos.

El control inteligente es preciso observarlo como un paso en el desarrollo progresivo de sistemas de control. Desde los sistemas más clásicos —por ejemplo PID— hasta los más avanzados —como el control adaptativo experto [Adaptech 89] o el control robusto [Calle 90] — el objetivo de todos los sistemas es el mismo, ampliar el rango de condiciones de funcionamiento mejorando el comportamiento del sistema controlado. Es desde este punto de vista como hemos de contemplar el control inteligente, como un aporte de métodos nuevos para hacer frente a circunstancias que escapaban al control más convencional.

4.3.1. Control clásico y supervisor

En esta tesis nos referiremos con el término *control clásico* a todos los tipos de control que constituyen la materia de la teoría de control. El fundamento de este tipo de controles es matemático —basándose en modelos matemáticos de los sistemas a controlar.

No entraremos en profundizar en las características de este tipo de control, que pueden encontrarse en multitud de referencias de control automático. Los avances en este tipo de sistemas tratan de hacer frente a problemas inabordables con los sistemas anteriores, pero, en cualquier caso precisan de modelos mas o menos precisos del sistema a controlar. Ejemplos de este tipo de sistemas de control clásico son el control PID, el control adaptativo o el control robusto.

En esta tesis no se ha planteado como objetivo el analizar las características de este tipo de sistemas, pero baste decir que la problemática en su utilización y el motivo para tratar de buscar en

el control inteligente la solución, se centra en toda una serie de circunstancias anejas al control y determinadas por problemas de implantación de los mismos. Este tipo de circunstancias y su solución parcial a través del control inteligente son expuestas por primera vez por Åstrom [Åstrom 86], donde aparece por vez primera el término *control experto*. Cabe destacar entre ellas: selección de modos de operación, filtrado de señales, comprobación de alarmas, gestión de situaciones anormales, etc. Todo ello conduce a que las implantaciones utilizando técnicas convencionales de programación estén constituidas en su mayor parte por código auxiliar. Este hecho por ejemplo ha sido observado en particular en una aplicación desarrollada por el autor para control de un molino de cemento. El código correspondiente al regulador (dos PID en cascada) era del orden de 100 líneas, mientras que la aplicación completa rondaba las 10.000 (incluyendo interfases de usuario).

En un sistema de control clásico las tareas básicas a realizar son tres:

- Captura de datos (sensores).
- Cálculo de actuaciones (reguladores).
- Acciones de control (actuadores).

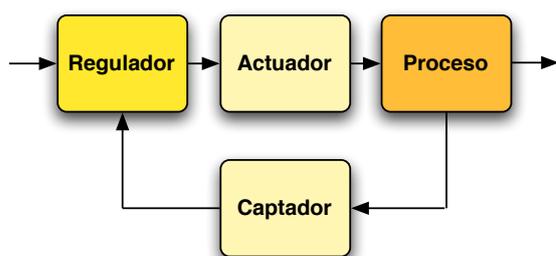


Figura 4.1: Esquema de control clásico.

Esto corresponde al esquema clásico de control que se presenta en la Figura 4.3.1. En el caso de control de procesos complejos se lleva a cabo una modificación de esta estructura para incluir al operador humano en la misma. Así se emplean estructuras como la que aparece en la Figura 2.1.

4.3.2. Inteligencia artificial y sistemas expertos

Desde otro punto de vista el control inteligente tiene su origen en las tecnologías de inteligencia artificial, y, en particular, de sistemas expertos. Este tipo de técnicas se han venido aplicando en el campo industrial con mayor o menor éxito, pero siempre limitadas estrechamente en sus rangos de aplicación. Estos se han restringido por lo general al diagnóstico y detección de faltas y a la generación de consejos para operadores. Ambos tipos de aplicaciones se basan en la utilización de procesos de toma de decisiones para determinar el estado funcional de la planta. Esta restricción es debida a la falta de prestaciones para operación en tiempo real y la falta de fiabilidad y verificabilidad de los sistemas expertos.

Desde el punto de vista de los sistemas expertos se puede hablar de tres generaciones de sistemas expertos progresivamente evolucionadas.

4.3.3. Sistemas de primera generación

Los sistemas expertos de primera generación son calificables de clásicos y se extienden desde las aplicaciones de diagnóstico a las aplicaciones de control, siendo por lo general sistemas basados en reglas. Se basan en métodos utilizados en los primeros sistemas expertos con éxito, caracterizados por asociaciones empíricas entre las premisas y las conclusiones. Sin embargo las prestaciones de estos sistemas de *primera generación* son limitadas [Alty 87] [Shirley 87], apareciendo una serie de problemas de crucial importancia. Algunos de estos problemas son comunes a todos los campos de aplicación de los sistemas expertos:

- Disminución brusca de prestaciones cuando se presenta un problema que cae fuera de su rango de conocimiento (efecto de *degradación de prestaciones en acantilado* o simplemente *efecto acantilado*).
- Dificultad en la obtención del conocimiento de los expertos, sobre todo cuando son varios. De hecho la *adquisición del conocimiento* se ha convertido en uno de los vocablos favoritos de los investigadores en el tema.

- Incapacidad del sistema experto de aprender de los propios errores. Salvo casos muy específicos los sistemas expertos no disponen de la realimentación —procedente de un experto humano— ni de los mecanismos para actualizar su conocimiento.

Otros problemas relacionados con los anteriores son especialmente importantes cuando se utilizan estos sistemas en control directo:

- Incapacidad para operar en tiempo real, exacerbada en este tipo de sistemas por la “recolección de basura”, dando problemas muy graves cuando esta es no incremental.
- Incapacidad de representar fenómenos distribuidos espacial y/o temporalmente.
- Dificultad para los ingenieros de conocimiento en asegurar la consistencia de la base de conocimiento.
- Incapacidad de detectar vacíos en la base de conocimiento. Este punto está relacionado con el efecto acantilado, en el sentido de la carencia de reglas para hacer frente a determinadas situaciones.

La inviabilidad de la construcción de sistemas expertos de primera generación para control de procesos de complejidad media-alta es fundamentalmente debida a factores de operatividad en tiempo real y de seguridad. La incapacidad de verificar la consistencia y la posibilidad de vacíos en la base de conocimiento —y en menor medida el efecto acantilado— hacen que la operación en modo automático de procesos con bases de conocimiento de control complejas sea, cuando menos, arriesgado. Este tipo de circunstancia es inadmisibles a priori en algunos sistemas, y en la mayoría las pérdidas derivables de la mala operación obligan a su no utilización.

4.3.4. Sistemas de segunda generación

En los sistemas de primera generación la construcción de la base de conocimiento se hacía ladrillo a ladrillo —regla a regla—, hasta que el edifi-

cio alcanzaba una altura o consistencia suficiente a juicio del ingeniero de conocimiento.

Este tipo de desarrollo produce problemas muy graves, que se mencionaron en el punto anterior: incoherencias y vacíos. Estos problemas se ven agravados por la inverificabilidad de la base de conocimiento. La calidad de un sistema experto de control viene pues determinada por dos factores:

- Los medios hardware/software, que van evolucionando progresivamente y que están dejando de ser cuellos de botella para algunos tipos de aplicaciones.
- El desarrollo; la calidad del equipo de desarrollo será la que determine la calidad final del sistema experto de control.

Esta construcción sobre la marcha del sistema experto es inaceptable en la mayor parte de los casos, esencialmente por su inverificabilidad, ya que es imposible la aceptación de un sistema que posiblemente pueda tener un error grave. Este tipo de problemas también aparecen en aplicaciones convencionales, aunque en menor medida y con soluciones parciales que, aunque costosas, son eficaces.

Los sistemas expertos de *segunda generación* tratan de paliar estos problemas de consistencia y completitud de la base de conocimientos utilizando un *plano de construcción*. Dado que el sistema experto va a trabajar sobre una porción del mundo —el dominio—, el conocimiento sobre dicha porción del mundo está en directa relación con el mismo. El enfoque utilizado es la *construcción de un modelo explícito* de dicha porción del mundo que constituirá parte de la base de conocimiento del sistema experto. Este enfoque se diferencia del de los sistemas expertos de primera generación en que el modelo es explícito, por lo que es verificable, ya que se debe corresponder con el dominio.

La construcción de la base de conocimiento se guía por la estructura del dominio para construir un modelo del mundo, que será tan completo como queramos y que será coherente. En este tipo de sistemas es fundamental el uso de mecanismos de representación de conocimiento adecuados. La utilización de *objetos* parece ser la opción más adecuada.

4.3.5. Sistemas de tercera generación

Los sistemas de tercera generación constituyen la punta de lanza del control inteligente. En este momento este frente tecnológico lo ocupan los sistemas con aprendizaje, que no suponen un cambio estructural respecto a los de primera o segunda generación, sino una mejora de éstos, por lo que quizá no es adecuado hablar de una nueva generación.

El término *aprendizaje* es susceptible de interpretaciones muy dispares. Nosotros consideraremos aprendizaje como *mejora del conocimiento*. En nuestro caso esta acepción tiene un significado preciso. Como ya se ha comentado en capítulos anteriores todo el conocimiento sobre el mundo se construye y utiliza en forma de modelos. Decíamos que un modelo era tanto mejor cuando su capacidad de predicción era tanto más exacta, esto es cuando se ajustaba más a la realidad que modelaba.

Es en este sentido en el que el conocimiento puede mejorarse; el aprendizaje supondrá una mejora de los modelos del dominio que tiene un determinado ente.

4.4. Elementos de Control Inteligente

Hemos visto desde un punto de vista histórico como se ha producido el acercamiento al control inteligente, y cuales han sido las trayectorias desde el control automático y la inteligencia artificial hasta alcanzar el punto actual.

En este apartado trataremos de analizar en mayor profundidad qué son los sistemas de control inteligente y cuales son sus características principales; facilitando de este modo el acceso a estas técnicas desde un punto de vista general, sin restringirse a ningún tipo de metodología concreta.

4.4.1. Objetivos de un sistema de control inteligente

En el caso de procesos complejos que requieren la intervención de humanos para su control, es misión del operador del proceso el conseguir que éste se comporte de forma que se logren unos determinados objetivos de alto nivel. Esta misión de alto nivel es la que hace necesaria la presencia humana en este tipo de sistemas, ya que es precisa su capacidad de trasladar los objetivos de alto nivel a acciones a realizar sobre el proceso. Los objetivos de alto nivel pueden ser diversos, pero entre ellos los principales son:

Seguridad: Para personas y equipos.

Calidad: Ni deficiente ni excesiva, que originaría un excesivo coste.

Producción: No sólo a corto sino a largo plazo, por lo que entran en consideración aspectos de mantenimiento (estabilidad de funcionamiento).

Ecología: Tratando de minimizar el impacto ambiental del proceso industrial.

Mantenibilidad: Tratando de evitar que la incorrecta operación cause una excesiva necesidad de mantenimiento o incluso impida el correcto mantenimiento.

La implantación de estos —u otros— objetivos de alto nivel en un sistema de control clásico es difícil; ello es debido fundamentalmente a que se formulan en un nivel de abstracción por encima del empleado para el sistema de control. Es preciso construir subsistemas adicionales al sistema de control para que tengan en cuenta estos objetivos globales. Este enfoque de sistemas adicionales es adecuado siempre que se realice de forma perfectamente integrada.

4.4.2. Actividades de control

El sistema de control inteligente reemplazará al humano en muchas de sus actividades, por lo que es preciso analizar en profundidad cuales son estas actividades. Voss [Voss 88] [Voss 88] presenta

un análisis bastante acertado en el que se afirma que las actividades de control llevadas a cabo en un sistema de control son las siguientes:

- Determinación de la *necesidad de la actuación*, en función del estado real y el estado deseado para el proceso. Para realizar esta determinación es preciso realizar una identificación del estado funcional del proceso.
- Identificación de los *posibles cursos de acción* que nos permitan alcanzar el estado deseado a partir del estado actual. El curso de acción se establecerá en forma de secuencia de operaciones a realizar.
- Determinación del *impacto sobre el sistema* de cada uno de los cursos, estableciendo la evolución previsible del sistema frente a cada una de las secuencias de acciones.
- *Interpretación de los impactos* en términos de objetivos globales. Lo que nos permitirá determinar la bondad de cada uno de los cursos de acción en lo que respecta a seguridad, calidad y producción.
- *Selección y ejecución de una alternativa de control* en función de la interpretación global realizada en el apartado anterior. La ejecución de una acción de control se lleva a cabo mediante un determinado protocolo.
- *Monitorización de los resultados*, para determinar si la respuesta del sistema a la acción realizada se corresponde con la prevista.

El sistema de control inteligente deberá reemplazar al operador en estas tareas —o auxiliarle, por lo que precisa de gran potencia en el manejo del conocimiento que se tiene del proceso.

4.4.3. Aclaración de denominaciones

En la bibliografía se usa de forma indiscriminada términos como *sistema experto de control*, *sistema de control inteligente* y *sistema de control basado en el conocimiento*. Es preciso realizar una matización de estos términos para aclarar el significado que se les da en esta tesis.

De acuerdo con la definición de *control inteligente* dada con anterioridad, es preciso que estos sistemas realicen actividades cognoscitivas en un nivel de información abstracta. Este es el tipo de proceso de información que se le requiere al operador humano del proceso. Por ello un sistema de control que le sustituya merecerá el calificativo de inteligente, independientemente de cual sea la técnica empleada en su construcción.

Dentro del área de control inteligente la utilización de los términos *sistema experto de control* y *sistema de control basado en el conocimiento* trata de hacer distinción en el modo en que se obtiene y expresa el conocimiento del proceso; así por *sistema experto de control* entenderemos un sistema experto en sentido clásico, esto es, una aplicación informática que implanta el paradigma motor de inferencia-base de conocimiento, utilizando para ello un formalismo de reglas obtenidas del conocimiento extraído del experto humano. Un *sistema basado en el conocimiento* es una aplicación que hace uso de las técnicas de representación de conocimiento, sin especificar cual sea la procedencia de dicho conocimiento. Estas técnicas han sido desarrolladas esencialmente en el área de los sistemas expertos, por lo que los sistemas basados en el conocimiento pueden considerarse una generalización de éstos. En particular un sistema de control basado en el conocimiento puede integrar conocimiento procedente de múltiples fuentes: los operadores, los ingenieros de proceso y los ingenieros de control.

4.4.4. Tendencias en control inteligente

Los primeros desarrollos en control inteligente han sido sistemas expertos de control basados en conocimiento de operadores, pero en los últimos tiempos se están desarrollando una serie de sistemas que se caracterizan por el uso de técnicas más orientadas al problema. Este tipo de sistemas se agrupan dentro de lo que denominamos *sistemas expertos de segunda generación*. La diferencia fundamental con los sistemas de primera generación (basados en el paradigma MI-BC) es la utilización de representaciones avanzadas del conocimiento en las que se consigue la implantación de un *modelo del sistema* con el que se está tratando.

De este modo aparecen dos tipos de conocimiento de los que hace uso el sistema experto de control de segunda generación:

- Conocimiento operacional, procedente de los operadores del sistema. Este tipo de conocimiento es esencialmente heurístico, creciendo en importancia a medida que el sistema aumenta en complejidad o en dificultad de modelado.
- Conocimiento estructural-funcional del sistema con el que está trabajando, del que se deducirán relaciones de causalidad entre los diferentes eventos.

En los últimos tiempos, el ensanche de los campos de aplicación gracias a la flexibilidad que este tipo de sistemas permite, ha conducido a una progresiva ampliación de objetivos. Ello ha conducido a la necesidad de integrar dentro de los sistemas de control otro tipo de conocimiento:

- Conocimiento procedural, procedente de los ingenieros de proceso y control, y referido a la manera en que se deben realizar determinadas operaciones complejas de carácter marcadamente secuencial y que requieren de métodos retorcidos para su implantación en sistemas de reglas convencionales³.

De ahora en adelante el término Sistema Experto de Control se referirá a Sistemas Expertos de Control de Segunda Generación, que es el tema que nos ocupa; cuando se desee utilizar otra acepción se indicará ésta explícitamente.

4.4.5. Características fundamentales del control inteligente

En el área de control inteligente se está comenzando a realizar un gran esfuerzo de investigación, por lo general debido al extenso campo de aplicación y a las perspectivas económicas que ofrece; y

³Esto se ha producido por ejemplo al tratar de implantar procedimientos secuenciales en un sistema basado en reglas. Se logró llevar a cabo el objetivo en base a la activación secuencial de reglas mediante asignación ordenada de prioridades a las mismas; lo cual no es, en absoluto, un método adecuado.

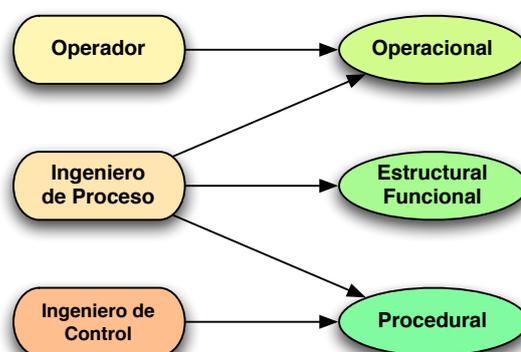


Figura 4.2: Tipos de conocimiento y procedencia del mismo.

por ello la terminología tiende a ser confusa, existiendo multiplicidad de términos para referirse a conceptos similares. En este apartado trataremos de aclarar algunos de los conceptos fundamentales y los términos utilizados para referirse a ellos. Existen varios aspectos cruciales que caracterizan y constituyen las piedras de toque de este tipo de sistemas; estos aspectos se pueden agrupar en dos categorías fundamentales:

- Aspectos conceptuales: Derivados de las necesidades específicas del control.
- Aspectos de implantación: Son los relativos a la forma en que se va a conseguir cumplir los requerimientos conceptuales anteriormente expresados.

Las características que deben presentar los sistemas de control inteligente vienen derivadas de las de los objetivos que persiguen —control—, del entorno en que deben alcanzar dichos objetivos —procesos— y de los medios que cuentan para ello —inteligencia artificial. Algunas de estas características son estudiadas por Laffey [Laffey 88].

En la Tabla 4.4.5 se presentan los aspectos más característicos de estos sistemas. En el lado izquierdo se incluyen los problemas a los que estos sistemas deben hacer frente, y en el lado derecho las técnicas utilizadas para ello.

SISTEMAS DE CONTROL INTELIGENTE	
PROBLEMAS	SOLUCIONES
CONTROL	
Operación continua	Contextos múltiples
Interfase con elementos externos	Sistemas abiertos
Requerimientos de tiempo real	Descomposición Paralelización
Interacción con sistemas dinámicos	Razonamiento temporal
Incertidumbre de actuación	Razonamiento hipotético Simulación
PROCESO	
Ocurrencia asíncrona de eventos	Interrumpibilidad
Datos inciertos o perdidos	Razonamiento con incertidumbre Validación del conocimiento Simulación on-line
Desconocimiento de dinámica	Modelos cualitativos
Ambientes agresivos	Equipos industriales
SISTEMAS BASADOS EN EL CONOCIMIENTO	
No monotonía de los datos sensoriales	Razonamiento no monótono Mantenimiento de verdad
Efecto acantilado y fiabilidad	Modelado profundo
Diferentes tipos de conocimiento	Modelos multiresolucionales
Procedimientos secuenciales	Integración de lenguajes procedurales

Tabla 4.1: Algunos aspectos característicos de los sistemas de control inteligente.

4.4.6. Control

Como todo sistema de control un sistema de control inteligente tiene que operar en tiempo real. Debe hacer frente a la resolución de un problema que evoluciona en el tiempo y al que hay que dar respuesta en un plazo temporal limitado. Este es el aspecto más característico de las aplicaciones de inteligencia artificial, en particular sistemas expertos, en control.

Un sistema de control es un ente encargado de que otro —sistema objeto— tenga un comportamiento determinado. La necesidad del sistema de control surge de las variaciones a lo largo del tiempo en el estado del sistema objeto —debidas a causas diversas. El funcionamiento en tiempo real es la característica más importante de un sistema de control. Esto requiere del sistema de control una actividad continua. Al sistema no se le presenta una tarea y cuando la termina queda a la espera de otra, sino que es preciso que el sistema tenga una gran disponibilidad. Esta disponibilidad se logra a través de la utilización de sistemas de razonamiento que permitan ejecuciones múltiples, lo que suele estar implantado por medio de un mecanismo basado en contextos.

Un contexto es la serie de circunstancias que determinan el comportamiento del sistema de razonamiento. En nuestro caso un contexto es una situación del proceso. La posibilidad de trabajar con contextos múltiples hace que sea posible mantener de forma simultánea varias representaciones del estado del sistema, permitiendo de este modo realizar razonamientos sobre diferentes problemas de control. Estos mecanismos de razonamiento múltiple generan dos problemas adicionales: el solape de dominios y la colisión de acciones. Existe un solape de dominios entre dos ejecuciones del sistema cuando existen objetos del proceso que intervienen en ambos razonamientos. Es preciso realizar una coordinación entre ambas. Esta coordinación puede ser un simple criterio de exclusión por prioridades o bien un mecanismo complejo de coordinación específico del problema. Se dice que existe colisión de acciones cuando como resultado del proceso de inferencia múltiple se especifican dos acciones para una misma variable de actuación. El problema es similar al de solape de dominios —

es de hecho un problema de solape de dominios— aunque sus consecuencias son más claras y pueden ser más graves.

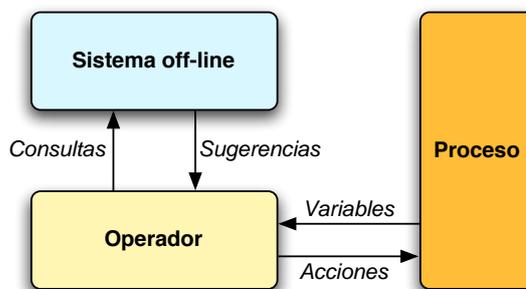


Figura 4.3: Sistema sin vínculos con el mundo. Las decisiones de acción las toma el operador humano.

Los sistemas de control inteligente se diferencian de la mayoría de los sistemas basados en inteligencia artificial en un aspecto fundamental: su vinculación al mundo. El sistema de control inteligente obtiene datos del mundo y actúa sobre él de forma automática, sin la intervención de humanos. En los sistemas inteligentes convencionales —en particular en los sistemas expertos— estos vínculos con el mundo no eran directos, sino que se producían a través de una interfase humana. El humano era el que proporcionaba al sistema experto toda la información sobre el problema; era el que interpretaba los resultados obtenidos por éste y el que realizaba las acciones asociadas a dicha interpretación. Los vínculos con el mundo hacen de los sistemas de control inteligente sistemas en *tiempo real*, con toda la problemática que ello conlleva. En particular, la actuación directa sobre el mundo, es sin duda el aspecto más característico de los sistemas de control inteligente.

Los aspectos de implantación derivados de la necesidad de operación en tiempo real son muchos y muy variados: comunicaciones con equipos de captación y actuación, interrupciones, potencia de cómputo simbólico y numérico. Todo ello conduce a la necesidad de grandes equipos y de software eficiente, por lo que en la mayoría de los casos es preciso recurrir a lenguajes de programación convencionales que son más eficientes que los emplea-

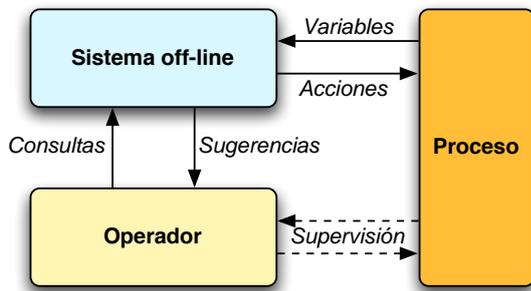


Figura 4.4: Sistema con vínculos con el mundo. Las decisiones de acción las toma el sistema de control y el operador humano en casos especiales.

dos en inteligencia artificial para determinado tipo de tareas. Por ello los entornos de construcción de sistemas de control inteligente deben ser *abiertos*, permitiendo desarrollos mixtos de software. La herramienta G2, por ejemplo, permite el uso de funciones en C o Fortran. NEXPERT OBJECT llega mas lejos, estando disponible como código objeto embebible en cualquier tipo de aplicación.

Los sistemas de control son sistemas en tiempo real por excelencia. Pero aún así hemos de distinguir entre dos tipos de requerimientos de operación en tiempo real: requerimientos débiles y requerimientos fuertes. Un sistema tiene requerimientos fuertes de operación en tiempo real cuando un determinado cómputo tiene un horizonte de valid, esto es, es preciso haberlo terminado antes de un tiempo establecido. Si el sistema en tiempo real no tiene requerimientos fuertes diremos que sus requerimientos de operación en tiempo real son débiles. Este es el caso más general, en el que el superar el horizonte temporal esporádicamente conduce a una simple degradación del comportamiento global del sistema y no a una falta en el mismo. Ejemplos típicos de sistemas con requerimientos fuertes y débiles son un sistema de aviónica y un control de proceso químico.

El garantizar el final de la inferencia antes del horizonte temporal es difícil, dado que este tipo de sistemas basados en inteligencia artificial son muy variantes temporalmente y muy dependientes del estado de la máquina sobre la que se ejecutan. Po-

demo alcanzar una cierta garantía estadística de que un sistema determinado terminará la inferencia antes del horizonte, pero no podremos tener la certeza de ese hecho. Aún en sistemas de estructura muy simple —por ejemplo monocapa— es compleja la determinación de los márgenes temporales. Veremos que el control basado en reglas borrosas permite realizar esta determinación de duración temporal de la inferencia, pero gracias a que este tipo de control es de estructura absolutamente fija y con procesos de inferencia absolutamente idénticos en cualquier circunstancia.

El hecho de que sea imposible garantizar el fin del proceso de inferencia antes del horizonte temporal no impide que este tipo de sistemas se apliquen en tiempo real. En sistemas con requerimientos débiles son perfectamente aplicables. Además la garantía estadística puede ser suficiente para determinados sistemas con requerimientos fuertes. El sistema obtendrá una conclusión que estará estadísticamente dentro del horizonte temporal. Esta conclusión puede no ser la mejor obtenible con los hechos disponibles si se dispusiera de tiempo indefinido. El problema de este tipo de razonamiento es el seleccionar la técnica más adecuada para que, sin tener que recorrer todo el espacio de soluciones, encontremos una aceptable. Este es el caso del *razonamiento progresivo*. En él se realiza una implantación en forma de *capas de conocimiento*. Estas capas de conocimiento llegan a conclusiones sucesivas, progresivamente refinadas. Dependiendo del tiempo disponible se podrá utilizar una o varias capas de conocimiento para encontrar la solución mejor.

En el capítulo 5 se presenta una metodología de construcción de sistemas de control inteligente basada en la descomposición del mismo que permitirá la construcción de sistemas que cumplan requerimientos de operación en tiempo real. Esta descomposición permitirá realizar procesamiento en paralelo, lo que contribuye a aumentar la potencia de cómputo. Además esta descomposición facilita la construcción de sistemas distribuidos, lo que facilita la paralelización, la verificación y la tolerancia a fallos.

El sistema de control puede estar distribuido en dos sentidos: utilización de fuentes de conocimiento distribuidas y estar constituido por módu-

los independientes funcionalmente. De acuerdo con esto la distribución puede ser:

- De las fuentes de conocimiento: El conocimiento que utiliza el sistema de control procede de tres fuentes principales:
 - Los sensores de proceso.
 - La información suministrada por el operador.
 - Sus bases de conocimiento, en forma de reglas y hechos o en forma de modelo del proceso; estas bases de conocimiento pueden estar a su vez distribuidas [Bailey 88].

Esta procedencia múltiple hace que sea preciso realizar una integración del conocimiento, que a su vez permite validar el mismo.

- De los subsistemas: El sistema de control puede estar descompuesto funcionalmente en subsistemas, cada uno de ellos con tareas definidas. Estas tareas pueden ser diferentes o bien iguales para aumentar la seguridad por redundancia. Ejemplos en este campo son sistemas expertos cooperantes con arquitectura *blackboard* [Nii 86] o los sistemas jerárquicos [Berger 89].

De especial importancia en la mejora de prestaciones por descomposición es el uso de mecanismos de focalización. Son aquellos que permiten restringir el área de interés a un determinado subsistema lanzando sólo las reglas pertinentes al mismo —focalización espacial— y a un determinado período temporal asociado al tempo característico de la situación bajo análisis —focalización temporal. Estos mecanismos posibilitan el que la ejecución del sistema experto sea lo suficientemente eficiente como para que pueda usarse en tiempo real. Las arquitecturas con metaniveles implantan una aproximación a los mecanismos de focalización. El razonamiento temporal en sistemas de control involucra además el concepto de escala temporal. Si nuestro sistema debe hacer frente al control de elementos con períodos característicos se pueden llevar a cabo cambios de escala temporal, en el sentido de centrarse en un determi-

nado *tempo* y considerar los subsistemas más lentos como constantes y los más rápidos como instantáneos. Esto es un mecanismo de focalización temporal que se propone en [Kuipers 88] en el ámbito de la simulación cualitativa.

En un sistema distribuido —física o funcionalmente— existen diferentes elementos, que en el caso de un sistema de control actúan sobre un mismo objeto. Esto hace que sea precisa la coordinación eficiente entre ellos. Esta coordinación exige mecanismos de comunicación eficientes y alta capacidad, ya que los volúmenes de información a manejar van a ser muy elevados. Las comunicaciones interprocesos dependen del tipo de arquitectura hardware, pudiendo ser internas a una máquina —basadas en los recursos del sistema operativo— o bien entre varias máquinas. El estándar que se está imponiendo es el uso de redes locales para estas comunicaciones interprocesos; de esta forma se facilita no sólo la comunicación entre los procesos del sistema de control, sino con otros sistemas externos (bases de datos, correo, etc).

La necesidad de utilizar módulos escritos en otros lenguajes —por lo general lenguajes procedurales— puede venir ocasionada por varios motivos:

- Interfase con equipos de captación que puede requerir el uso de código desarrollado en otros lenguajes.
- Potencia de cómputo numérico, como la que es necesaria para realizar una simulación clásica.
- Necesidad de ejecutar procedimientos secuenciales, difícilmente especificables y sobre todo verificables mediante reglas.

Los procesos a controlar son sistemas dinámicos, en los que el control debe basarse tanto en el estado actual del proceso como en su evolución temporal. En el caso del control clásico esto se consigue utilizando un formalismo de ecuaciones diferenciales —o sus transformadas— pero en el control inteligente el problema es un poco más especial. El razonamiento temporal [Shaw

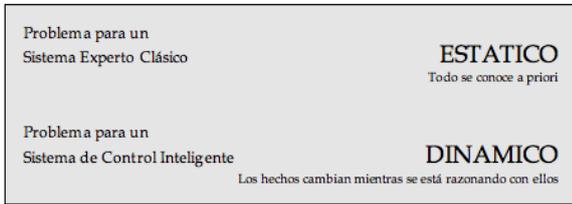


Figura 4.5: Comparación entre sistemas expertos clásicos y sistemas expertos de control.

89] [Shaw 89] supone utilizar en el razonamiento valores cambiantes con el tiempo. Es por ello preciso manejar *historias* de valores. El método habitual de trabajo es el de utilizar atributos específicos de las variables —objetos o marcos— para mantener información temporal; guardando en ellos el tiempo en que se asertó el hecho o bien la *fecha de caducidad* del mismo [Gensym 88]. Es por este aspecto dinámico que los problemas planteados a los sistemas de control inteligente son diferentes de los problemas planteados a sistemas expertos convencionales. Este mismo aspecto de temporalidad hace que una de las alternativas a considerar en el diseño de sistemas de control inteligente sea el utilizar arquitecturas dirigidas por eventos. El comportamiento de este tipo de sistemas es similar al encadenamiento hacia delante, salvo en que la selección de la siguiente etapa de inferencia se hace en base a los hechos que han cambiado o al resultado de la última etapa de inferencia.

En algunos casos el sistema de control inteligente puede obtener una conclusión múltiple. Esto sucede cuando en el conjunto conflicto aparecen distintas reglas que concluyen acciones de control. Es importante en estos sistemas realizar el proceso de inferencia hasta el final, obteniendo todas las conclusiones posibles hasta que no se pueda progresar más en la inferencia. De este modo se garantiza la generación de todas las acciones de control. En algunos casos el motor de inferencia utilizado limita en cierta medida esta posibilidad. Esto sucede cuando por ejemplo se selecciona la primera conclusión que representa una acción de control [Martínez 90]. Esto sin embargo es un procedimiento poco robusto y que puede dar lugar a graves problemas.

La generación de todas las alternativas de control nos garantiza que no se ha eliminado la importante en el proceso de inferencia, pero es preciso realizar una etapa posterior de selección de la acción de control adecuada. En algunos casos esta etapa se lleva a cabo de forma implícita en la ejecución de la inferencia de control, de modo que la acción propuesta sea única. La elección de la alternativa adecuada es una tarea compleja, en la que hay que realizar un razonamiento hipotético para tratar de evaluar las consecuencias de las diferentes acciones de control. En este razonamiento hipotético es de fundamental importancia el disponer de un mecanismo de contextos. Un contexto es la serie de circunstancias que determinan el comportamiento del sistema. En nuestro caso un contexto es una situación del proceso. La posibilidad de trabajar con contextos múltiples hace que sea posible mantener de forma simultánea varias representaciones del estado del sistema, permitiendo analizar las posibles evoluciones ante distintas hipótesis de control. Para realizar estas evaluaciones es importante también disponer de un simulador del proceso a controlar, de forma que se pueda determinar la evolución del sistema frente a las acciones de control.

4.4.7. Proceso

En los procesos reales se den de forma habitual situaciones que se salen de lo que podría denominarse funcionamiento normal. Ello hace que el sistema de control inteligente deba ser capaz de manejar estas situaciones *cuando ocurran*. La interrumpibilidad del razonamiento puede considerarse una cuestión de implantación, estando directamente relacionada con el razonamiento en tiempo crítico. Pero el hecho de tener que interrumpir una línea de razonamiento debido al cambio del valor de verdad de un hecho es algo nuevo. Por razonamiento interrumpible se entienden dos conceptos diferentes:

- **Cancelable:** Cuando por la alteración de las circunstancias el razonamiento que se está llevando a cabo carezca de sentido. Este es el caso de la alteración de los valores de verdad de los hechos de partida del razonamiento; si estos valores cambian durante el razonamiento

la conclusión derivada de ellos será por lo general no aplicable. Cuando se tengan hechos de este tipo lo más simple es la cancelación del proceso, pero se puede intentar un enfoque que trate de utilizar al máximo el trabajo ya realizado. En este sentido se pueden intentar dos tipos de acciones:

- Propagar las alteraciones a través del grafo de encadenamientos, de forma que se modifiquen las conclusiones a las que afecte los hechos alterados. Esto es viable si se utilizan enfoques del estilo de RETE [Forgy 82], en que la unificación es más eficiente.
 - Utilizar los hechos derivados con anterioridad si las condiciones son cualitativamente iguales. Esto es mucho más complejo ya que exige realizar razonamientos cualitativos sobre las analogías entre ambas situaciones.
- Suspendible: En función de las prioridades de las tareas de razonamiento del sistema de control puede ser necesaria la ejecución de una tarea mientras se está llevado a cabo otra. En este caso es preciso suspender la primera, guardar el entorno de la misma —toda la información precisa para su continuación desde el punto de suspensión, ejecutar la de mayor prioridad, recuperar el entorno y continuar la tarea anterior. El mecanismo de interrupción puede ser costoso en tiempo y/o en memoria, por lo que el enfoque más adecuado es el utilizar un entorno que permita la ejecución múltiple con asignación de prioridades a las distintas ejecuciones.

La necesidad del razonamiento cancelable es propia del control inteligente. En un sistema experto clásico, si han llegado nuevos hechos —o se han modificado los de partida— durante el proceso de razonamiento, basta con realizar una nueva ejecución del sistema con estos nuevos hechos para obtener la solución más adecuada. En un sistema de control esto no es aceptable, fundamentalmente porque las conclusiones derivadas de los hechos viejos pueden conducir a acciones de control erróneas, al no corresponderse con el estado

real del sistema a controlar. Se podría hacer lo mismo que en el sistema experto clásico —volver a lanzar la consulta— pero esto no es compatible con la criticidad temporal —necesidad de operación en tiempo real.

El otro aspecto que considerábamos importante es el de la incertidumbre o imprecisión. La información que se obtiene del proceso no es exacta, sino que se ve afectada por una serie de circunstancias que rodean el hecho de su determinación: malfunciones en sensores, ruidos en líneas de comunicación, etc. Además, aparte de la imprecisión en las medidas, existe imprecisión en las acciones a realizar, ya que por lo general la relación situación-acción no es 1-1, sino que para cada situación existe todo un cúmulo de posibles acciones de efectos más o menos conocidos. Así pues hemos de disponer de medios para operar —razonar— con conocimiento impreciso. El razonamiento sobre sistemas reales se ve plenamente afectado por la incertidumbre, que se presenta de diferentes formas y con diferentes orígenes. Para referirse a ello existen multitud de términos en la literatura: Inexactitud, incertidumbre, imprecisión, indeterminación, vaguedad, probabilidad, posibilidad, ambigüedad, generalidad, no especificidad, disonancia, confusión etc.

El sistema de control inteligente debe ser capaz de razonar con hechos afectados de incertidumbre, y alcanzar conclusiones que estarán —posiblemente— afectadas también de incertidumbre. Este mecanismo puede ser una herramienta adecuada para la selección de la alternativa de control adecuada en el caso de que existan varias.

En control la información procede de diferentes fuentes, como son el operador, los sensores, las bases de conocimiento, etc. La necesidad de operación segura obliga a que las acciones de control se correspondan con el estado real del sistema a controlar, lo que conduce a la necesidad de validar el conocimiento que se va a emplear en deducir la acción de control. La validación de esta información se llevará a cabo por medio de contraste entre las diferentes informaciones, en un enfoque similar al de la *fusión sensorial* [Pau 89]. Para realizar esta validación es preciso disponer de un modelo del sistema a controlar que nos permita establecer las correlaciones correctas entre las diferentes in-

formaciones. En algún caso estas fuentes de conocimiento pueden fallar en su misión de proporcionar determinada información, causando un posible bloqueo, o en base a un criterio de validación del conocimiento podemos alcanzar la conclusión de que determinado dato no tiene la calidad suficiente para ser utilizado. El disponer de mecanismos de suministro de valores adecuado es importante para aumentar la robustez del sistema de control. Estos mecanismos se basan esencialmente en historias, realizando una predicción del valor inaceptable. En este tipo de tareas se pueden utilizar simuladores en línea del proceso. Una tarea similar es la de suministrar valores *ocultos*, esto es valores de variables de proceso no medibles.

Tan importante es el manejo de la imprecisión de las medidas como lo es el de la imprecisión de las dependencias e interacciones. La complejidad terciaria mencionada en el capítulo 2 es el factor más difícil de manejar en control de procesos. Es importante que nuestros criterios de control no dependan de forma crítica de la validez del modelo de las interacciones del proceso. Es preciso aumentar la robustez para que un error de modelado de estas interacciones no concluya en desastre. La simulación cualitativa [Kuipers 86] [Forbus 84] [de Kleer 84a] es un enfoque prometedor y que está empezando a ser considerado en el campo del control inteligente [Berger 89].

Como hemos visto anteriormente los equipos a utilizar en control inteligente de procesos deben ser equipos muy potentes, ya que los requerimientos de operación en tiempo real exigen de ellos grandes prestaciones. Todo ello hace que las necesidades de equipos sean especialmente fuertes; además con el agravante de que al tener que instalarse en plantas industriales deben ser lo suficientemente robustos para hacer frente a los problemas de polvo, ruidos, temperatura, etc.

De todas formas el cuello de botella del control inteligente es el software y no el hardware, ya que la estructuración jerárquica de estos sistemas permite su implantación distribuida, con lo que es posible obtener grandes potencias con equipos convencionales, que, además, pueden situarse a conveniencia.

4.4.8. Inteligencia artificial

Desde el punto de vista de la inteligencia artificial la variabilidad temporal del proceso hace que los hechos que se manejan tengan valores de verdad cambiantes en el tiempo: son *verdades* con caducidad; la técnica empleada en hacer frente a este problema es el mantenimiento de verdad. El mantenimiento de verdad [Filman 88] está estrechamente relacionado con el razonamiento no monótono y el temporal. Consiste esencialmente en un mecanismo encargado de mantener en la base de conocimiento sólo los hechos verdaderos eliminando de la misma los caducados y los derivados de éstos.

En un sistema clásico la introducción de una aserción en la base de conocimiento era una acción definitiva, esto responde al sentido de que una cosa cierta (un hecho derivado a partir de otros considerados ciertos, usando reglas de inferencia válidas) será siempre cierta aunque se introduzcan nuevos hechos en la base de conocimiento, ya que siempre podríamos derivarla de los antiguos. Este punto de vista es correcto si el problema a resolver es *estático*, que no es el caso del problema de control. Es por ello preciso desarrollar para él mecanismos de razonamiento no monótono, que permitan retractarse de aserciones cuando los hechos cambien.

Los sistemas basados en el conocimiento presentan un problema que puede llegar a ser grave en el caso de aplicaciones en control. Este problema es el efecto acantilado, que consiste en una degradación brusca de prestaciones cuando el problema planteado cae fuera del dominio del sistema basado en el conocimiento. Un enfoque adecuado para disminuir este problema es el uso de modelos explícitos, que constituyen el elemento característico de los sistemas de segunda generación. El modelado del comportamiento de un sistema se puede hacer de dos modos diferentes, que denominaremos fenomenológico y causal. En el modo fenomenológico, expresaremos simplemente lo que sucede con el sistema desde el punto de vista de las magnitudes medibles. En el modo causal, se expresarán relaciones más profundas, derivadas de la estructura del sistema, y que expresarán relaciones causales entre magnitudes, siendo —por

lo general— precisa la utilización de magnitudes ocultas. Estos dos modos responden en grandes rasgos a lo que se ha venido denominando modelado somero y modelado profundo.

De especial importancia en el caso de control de procesos es el uso de modelos de sistemas de parámetros distribuidos, aunque este tema es lo suficientemente arduo como para que el número de trabajos en él sea mínimo [Nicolosi 88] [Nicolosi 88].

En muchos casos el control de un proceso requiere de la ejecución de protocolos para llevar a cabo determinadas tareas. La especificación de estos protocolos se hace en base a una experiencia grande o a criterios de seguridad y conocimiento profundo del proceso. El objetivo es facilitar el control del proceso en determinadas situaciones que son simultáneamente raras y de control difícil. En estos casos el uso de sistemas basados en el conocimiento —en particular basados en reglas— conduce a implantaciones más bien oscuras basadas en mecanismos poco claros o utilizados con fines distintos de aquellos para los que fueron diseñados. Estas implantaciones *anormales* son además difícilmente verificables, lo que las hace inaceptables en muchos casos. Es preciso habilitar mecanismos que permitan la ejecución de procedimientos secuenciales. Esto se hace mediante el uso de sistemas abiertos o mediante la integración, en el sistema de desarrollo de aplicaciones basadas en el conocimiento, de lenguajes procedurales. Esta última opción suele dar resultados mejores desde el punto de vista de la integración pero peores en eficiencia.

En un sistema experto el conocimiento procede de una persona —el experto. Pero en el caso de control de sistemas industriales, existen al menos tres fuentes de conocimiento importantes, asimilables al conocimiento de tres personajes. Estas tres fuentes son:

- El experto, en este caso el *operador del proceso*, que tiene el conocimiento para la operación del mismo. Este conocimiento es un modelo somero de la planta; un modelo de comportamiento, no causal.
- Documentación técnica sobre el proceso; correspondiente al diseño y construcción del

sistema. Esta documentación constituye un modelo de gran profundidad sobre el proceso, sin embargo carece de información causal, por lo que es precisa realizar una interpretación del mismo. A través de esta interpretación se extrae la información que constituirá un modelo causal del proceso. Esta actividad de interpretación se asociará a una persona que denominaremos *ingeniero de proceso*. Esta persona es la que dispone de la información sobre la estructura interna del proceso, que, en último término responde del comportamiento externo del mismo.

- Conocimiento procedural, que, aunque similar al conocimiento experto especificado en el punto primero merece ser considerado aparte. Se diferencia en esencia del primero en que este conocimiento no es declarativo, sino procedural, expresando secuencias de acciones, no hechos puntuales. Este conocimiento lo supondremos agrupado en un individuo al que denominaremos *ingeniero de control*. En realidad, gran parte de este conocimiento está en manos —o en la cabeza— del operador de la planta, no obstante, es muy diferente del conocimiento que hemos calificado como conocimiento de experto.

En un sistema de control avanzado es preciso hacer uso de estos tres tipos de conocimiento, ya que cada uno de ellos proporciona una información de importancia crucial en determinadas situaciones. Así el conocimiento del experto permite la conducción en régimen permanente del proceso. El conocimiento procedural permite la aplicación de protocolos de conducción en determinadas circunstancias como pueden ser un arranque o una parada. Por último el conocimiento estructural-funcional proporcionado por el ingeniero de proceso es importante en la gestión de situaciones que caen fuera del ámbito de conocimiento expresado en los otros dos. Este conocimiento permite afrontar el problema planteado por el efecto acantilado. No obstante, el uso de este conocimiento para control es complejo, siendo objeto de investigación en estos momentos, con perspectiva desconocida.

En realidad el modelado a utilizar en sistemas inteligentes no debe reducirse a un determinado

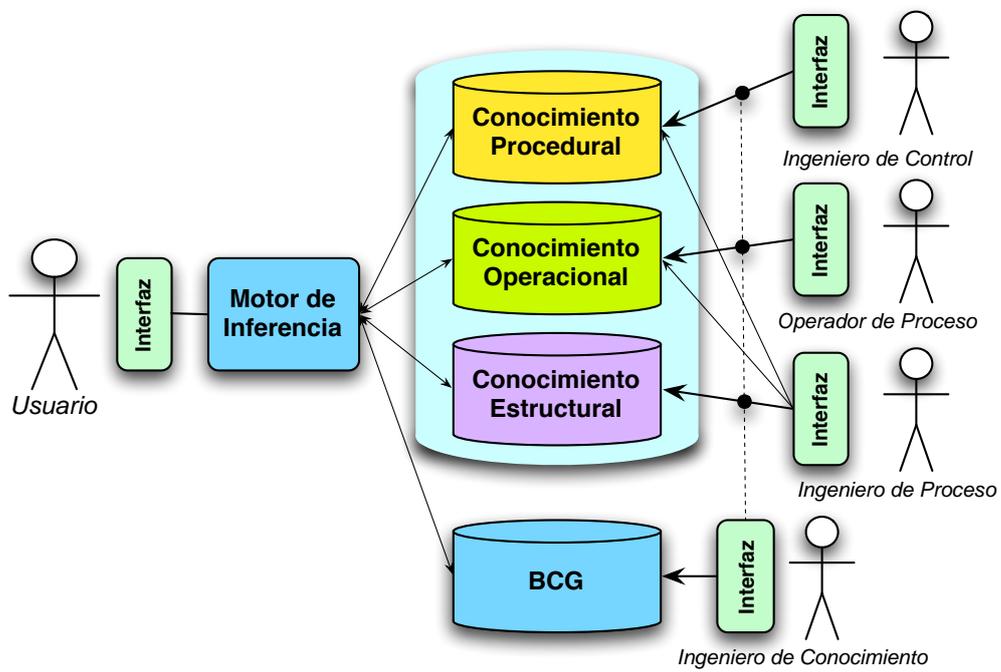


Figura 4.6: Orígenes diversos de las diferentes partes de la base de conocimiento. El ingeniero de conocimiento cataliza el proceso de formalización y representación.

modo, sino que debe realizarse en múltiples niveles de abstracción; debe ser un modelado multiresolucional. Antsaklis [Antsaklis 89] dice que para que un sistema logre la suficiente autonomía —la suficiente inteligencia— se deben utilizar en el sistema de control:

- Métodos algorítmico-numéricos, basados en el estado del arte en control convencional, identificación, estimación y teoría de la comunicación.
- Métodos simbólicos de toma de decisiones, tomados de la inteligencia artificial.

Esta integración de métodos es precisa para los sistemas inteligentes, dado que se necesita tanto las prestaciones de los sistemas numéricos, sobre todo velocidad, como la flexibilidad de los sistemas simbólicos. La autonomía procede en esencia de dicha flexibilidad, y por lo tanto de los mecanismos simbólicos, sin embargo, en aplicaciones reales, los criterios que dictaminan el éxito son fundamentalmente dos: si algo se ha hecho correctamente y si se ha hecho a tiempo.

4.5. Paradigmas de Control Inteligente

En el apartado 4.3 se mencionaron de pasada las características de los sistemas de control clásicos y en el 4.4 los elementos característicos de los sistemas de control inteligente. En este punto trataremos de ahondar en las características técnicas de los principales paradigmas de control inteligente. Estos paradigmas son modos establecidos de desarrollo de sistemas de control inteligente —tecnologías de control inteligente— que están ampliamente extendidos.

El concepto de paradigma de control nos trae a la mente los paradigmas científicos en el sentido de [Kuhn 71]. En este caso la transición entre paradigmas —*la revolución técnica* podríamos decir— se produce cuando surge una nueva teoría de representación y razonamiento sobre sistemas. Este tipo de transición está estrechamente relacionada con el desarrollo de nuevas metodologías de

representación del conocimiento. Los avances en control inteligente parecen conducir hacia la utilización de modelos explícitos —asimilables a los denominados modelos profundos— basados en la metodología de programación orientada a objeto. La creciente explicitación de los modelos ha pasado desde la lógica proposicional hasta las actuales representaciones orientadas a objeto, pasando por las ternas objeto-atributo-valor y los marcos.

En el comentario que sigue nos centraremos en tres paradigmas: el control basado en reglas, el control borroso y el control basado en modelos.

4.5.1. Control basado en reglas

Los sistemas de control basados en reglas (SCBRs) responden a los que se denominaron sistemas expertos de primera generación. Las características de este tipo de sistemas son de sobra conocidas y se pueden encontrar multitud de referencias sobre el tema [Galán 89][Frost 86].

Estructura de los SCBRs

Estos sistemas de control se basan en una base de conocimiento en la que se expresa mediante reglas el conocimiento de control de un determinado sistema. Este conocimiento de control puede extraerse de un operador de dicho sistema —sistema experto clásico— o bien formularse directamente por un ingeniero de control, en cuyo caso podríamos hablar de sistema experto o de sistema basado en el conocimiento según criterios. Un estudio más detallado de esta estructura puede encontrarse en [Basagoiti 89].

En la Figura 4.5.1 se puede observar la estructura habitual de un sistema basado en el conocimiento —en este caso un sistema basado en reglas. En esta figura se aprecian los siguientes subsistemas:

- Interfases.
- Motor de inferencia.
- Base de conocimiento.
- Base de hechos.

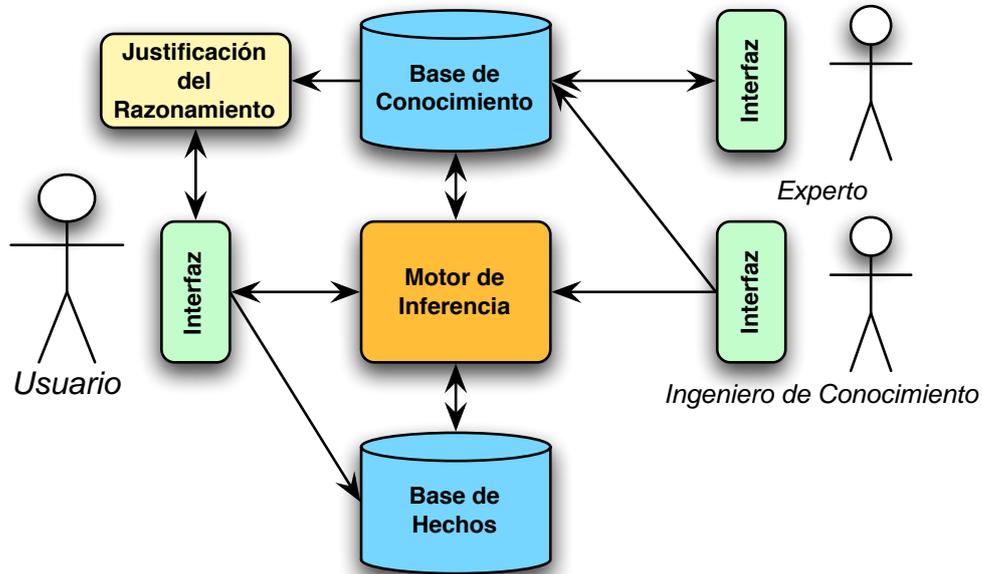


Figura 4.7: Estructura clásica de un sistema basado en el conocimiento expresado mediante reglas.

▪ Justificación del razonamiento.

Las personas que intervienen en la construcción y utilización de este tipo de sistemas son el ingeniero de conocimiento, el experto y el usuario.

Los hechos iniciales para el proceso de inferencia de este tipo de sistemas son, como cualquier sistema de control, información del proceso en forma de variables medidas y variables derivadas de éstas —variables de estado de la planta. Las conclusiones que se alcancen deben expresar las acciones de control a realizar sobre la planta.

El planteamiento de la estrategia de encadenamiento pueden hacerse tanto hacia delante como hacia atrás, pero con características muy diferentes. Por lo general en los sistemas de control el encadenamiento es hacia delante, con reglas de la forma:

```
SI La temperatura es mayor de 1300°
ENTONCES Bajar el fuel un 10 %
```

mientras que el encadenamiento hacia atrás se emplea en sistemas de diagnóstico, en los que las re-

glas expresan relaciones causales:

```
SI El fuel baja
ENTONCES La temperatura baja
```

Como vemos las reglas utilizadas en el ejemplo de operación hacia delante expresan acciones de control a realizar en una determinada situación de proceso. A primera vista parece posible expresar una estrategia humana de control de un proceso por medio de reglas de la forma⁴:

```
SI <Situación>
ENTONCES <Acción>
```

pero esto no es siempre viable. La expresión de las situaciones a reconocer (lados izquierdos de las reglas) puede llegar a ser muy compleja si el proceso a controlar es complejo. En las “reglas” que constituyen la experiencia de un humano las premisas no son simples, ni siquiera dobles o triples, sino que en realidad en la premisa de cada regla que conduce a una acción aparecen un patrón

⁴En sistemas de reglas de este tipo no se produce encadenamiento, constituyendo lo que se denomina sistemas monocapa.

dinámico⁵ complejo de las variables del proceso a controlar y que conoce el operador. El formular completamente una estrategia de control en estos términos puede llegar a ser muy tedioso y propenso a errores.

Por otra parte, surge un problema adicional con los lados derechos de las reglas: las acciones de control. En general las acciones a realizar no son acciones simples ni discretas, y sus valores posibles no son un número finito, sino que son asimilables a números reales. No es posible utilizar bases de conocimiento que contengan reglas que concluyan acciones con todos los valores posibles, ni siquiera parece viable utilizar un gran conjunto de reglas con diferentes acciones, sobre todo teniendo en cuenta la complejidad que deben tener las premisas. El control borroso resolverá en parte este problema, utilizando valores reales para los grados de satisfacción de las premisas, valores que son trasladados a las acciones a realizar.

Una forma más general de estructura de reglas en un SCBR es la siguiente:

```
SI          <Situación>
ENTONCES <Conclusión>
          <Acción>
```

Donde <Situación> es una expresión clásica de premisas de una regla, en la que aparecerán por lo general comparaciones de las variables de estado del proceso con valores característicos. <Conclusión> es una expresión indicando los hechos a incluir en la base de hechos del SCBR. Y <Acción> es una expresión indicando las acciones de control a llevar a cabo.

Uno de los problemas fundamentales del control basado en reglas es el aspecto representado por <Conclusión>. Este término representa una serie de hechos que deben introducirse en la base de hechos del SCBR. Pero estos hechos tienen una característica fundamental: su caducidad. Es precisa la existencia de sistemas de mantenimiento de verdad para hacer posible la operación de los SCBRs.

Observese que la cláusula <Acción> de una regla plantea un problema especial en este tipo de

sistemas. Si una regla como la anterior se aplicara repetidamente, el volver a incluir <Conclusión> en la base de hechos no plantea ningún problema especial, pero en cambio volver a ejecutar <Acción> puede ocasionar problemas graves. El control de la ejecución de reglas se hace por el lanzamiento controlado de las inferencias, de modo que los términos <Acción> están especificados teniendo en cuenta este hecho, o bien mediante el uso de variables de status. Este último enfoque ha sido el habitual en sistemas de control inteligente desarrollados con herramientas de sistemas expertos convencionales, por la incapacidad de realizar un control más fuerte del proceso de inferencia.

Algunos autores proponen otros formato de reglas, tal es el caso de Tzafestas [Tzafestas 89]. El formato propuesto es el siguiente:

```
SI          <Condición>
Y NO       <Condición Excluyente>
ENTONCES  Aserta <Hechos a añadir>
          Retracta <Hechos a eliminar>
          Haz <Acción>
          Siguiente <Regla>
```

Donde se hace manejo explícito de los hechos a añadir y a eliminar de la base de hechos, en un enfoque similar al de STRIPS [Nilsson 87] [Nilsson 87], y en el que se especifica la siguiente regla o reglas a aplicar, en un enfoque casi procedural.

Dos reglas se dirán con *solape de premisas* si es posible satisfacer ambas simultáneamente. En un sistema de control basado en reglas la existencia de solape de premisas puede generar inconsistencia en las acciones de control. Esto se observa muy claramente en un sistema monocapa. Supongamos que en nuestra base de conocimiento se tienen las dos reglas siguientes:

```
SI La temperatura es mayor de 1500°
ENTONCES Bajar el fuel un 10%
SI El oxígeno es mayor de 4%
ENTONCES Subir el fuel un 5%
```

Si en un determinado instante se verifica simultáneamente que *la temperatura es mayor de 1500°* y que *el oxígeno es mayor de 4%* la acción a realizar no está claramente definida, ya que una regla concluye que se debe bajar el fuel un 10%

⁵En el que intervienen derivadas temporales.

y la otra que se debe subir un 5%. La acción que se ejecutará —en un SCBR convencional— dependerá del mecanismo de selección de reglas del conjunto conflicto. Este problema de consistencia por solape de premisas puede resolverse de tres formas:

- Por eliminación física del solape. Realizando un desarrollo de la base de conocimiento que impida este tipo de situaciones. Esto se hará combinando en las premisas de las reglas todas las variables relevantes. De este modo se realiza una partición del espacio de entradas, de forma que a cada punto del espacio de entradas le corresponda una sola acción.
- Por eliminación funcional del solape. Esto es lo que se daría en un sistemas experto convencional, en el que se seleccionaría —según un determinado criterio— la regla a disparar del conjunto conflicto. Salvo que el algoritmo de selección este muy perfeccionado y adecuado a su utilización en sistemas de control, este enfoque es inaceptable.
- Por fusión de las actuaciones especificadas en las reglas con solape, combinando las múltiples acciones en una sola. Este tipo de combinación suele hacerse mediante técnicas de manejo de incertidumbre; especialmente adecuadas son las técnicas de lógica borrosa.

El sistema experto de control puede entenderse —al igual que un sistema convencional de control— como una aplicación del espacio de variables medidas en el espacio de actuaciones. Esta aplicación debe ser necesariamente una función.

Modos de funcionamiento

Los modos de funcionamiento de los SCBRs son esencialmente dos:

- Periódico, del mismo modo que funcionan la mayor parte de los sistemas de control implantados con computadores.
- Aperiódico, por respuesta a eventos. En este caso el mecanismo de inferencia se dispa-

ra en base al reconocimiento de un patrón —posiblemente complejo y con dinámica— en las variables del proceso.

Los modos periódicos se suelen emplear en SCBRs simples, en los que la base de conocimiento tiene un tamaño limitado y los tiempos de inferencia son adecuados al proceso y con poca varianza. En caso de sistemas complejos, la duración temporal de una inferencia tiene una variabilidad muy alta, siendo por lo general inadecuado su uso de forma periódica. Este tipo de sistemas se activan bajo demanda de otro sistema reconocedor de situaciones que requieran de la intervención del SCBR.

Desarrollo de SCBRs

El control basado en reglas es el resultado de la aplicación directa de la tecnología de sistemas expertos en control [Bernard 88] [Bernard 88]. En este tipo de sistemas la base de conocimientos de control se obtiene de un experto humano: el operador del proceso a controlar. Este enfoque, en procesos complejos⁶, conduce a sistemas expertos con bases de conocimiento de complejidad excesiva, siendo necesario el uso de técnicas que permitan reducir esta complejidad y aumentar la eficiencia. Una de estas técnicas es el uso de metaniveles. Por lo general el metanivel —suele haber sólo uno— es el encargado de determinar el estado del sistema y de seleccionar el grupo de reglas pertinente a esa situación. De este modo la base de conocimiento queda dividida en subbases que permiten una mejora en eficiencia. Por lo general estos sistemas alcanzan volúmenes —miles de reglas— que los hacen inadecuados para su utilización en tiempo real. Este tipo de arquitectura hace uso de conceptos intermedios —sobre el estado del sistema— que constituyen una versión aproximada del enfoque basado en modelos.

Una vez vistas estas características generales del los SCBRs podemos comentar las principales arquitecturas de los mismos, ya que casi todas las

⁶Especialmente en procesos con complejidades secundaria y terciaria. El caso de la complejidad primaria es más abordable mediante el uso de técnicas orientadas a objeto. Este enfoque es el utilizado por ejemplo por el entorno G2 [Gensym 88][Gensym 88].

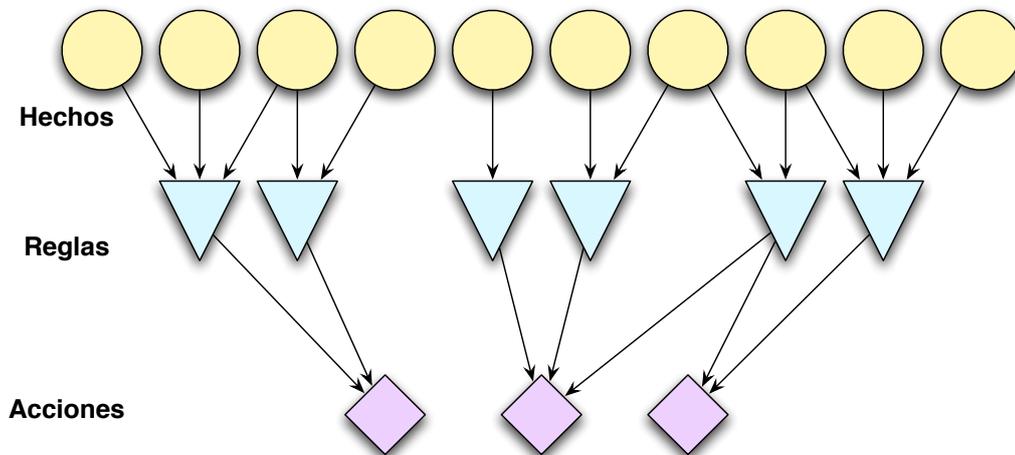


Figura 4.8: Sistema basado en reglas con estructura monocapa.

aplicaciones responden a una arquitectura monocapa o bicapa, aunque en algún caso existan arquitecturas multicapa.

SCBR monocapa

Este tipo de sistema es muy habitual por su simplicidad, ya que no requiere profundizar en las etapas de análisis del dominio y conceptualización.

En este tipo de arquitectura el sistema de control basado en reglas presenta una estructura como la de la Figura ???. La base de conocimiento de este sistema es homogénea, compuesta íntegramente por reglas de la forma:

```
SI <Situación>
ENTONCES <Conclusión>
      <Acción>
```

La existencia de conclusiones a incluir en la base de hechos tiene sentido en el uso de estas conclusiones en etapas posteriores de control.

SCBR bicapa

El término <Acción> es suprimible como término diferenciado de <Conclusión>, sin mas

que considerar la existencia de un dato en la base de hechos interpretado por un proceso independiente —un demonio— encargado de realizar la acción. Este enfoque es especialmente interesante desde el punto de vista de la independencia de los procesos de inferencia y control. Sea por ejemplo la regla:

```
SI La temperatura del horno es mayor
de 1500°
ENTONCES El horno esta caliente
      Disminuir fuel un 10%
```

El hecho *El horno está caliente* se introducirá en la base de hechos y la acción *disminuir el fuel un 10%* se llevará a cabo como consecuencia de la ejecución de la regla. No obstante el saber que el horno está caliente —hecho incluido en la base de hechos— puede ser utilizado de forma independiente por otro proceso para realizar la disminución del fuel de forma asíncrona respecto al proceso de inferencia. Este enfoque es efectivo para operación en tiempo real, sin embargo se aparta del enfoque SCBR puro.

La acción del proceso de control —asíncronamente con el de inferencia— puede llevarse a cabo por otro sistema basado en reglas. Este tipo de estructura es lo que denominaremos SCBR bicapa

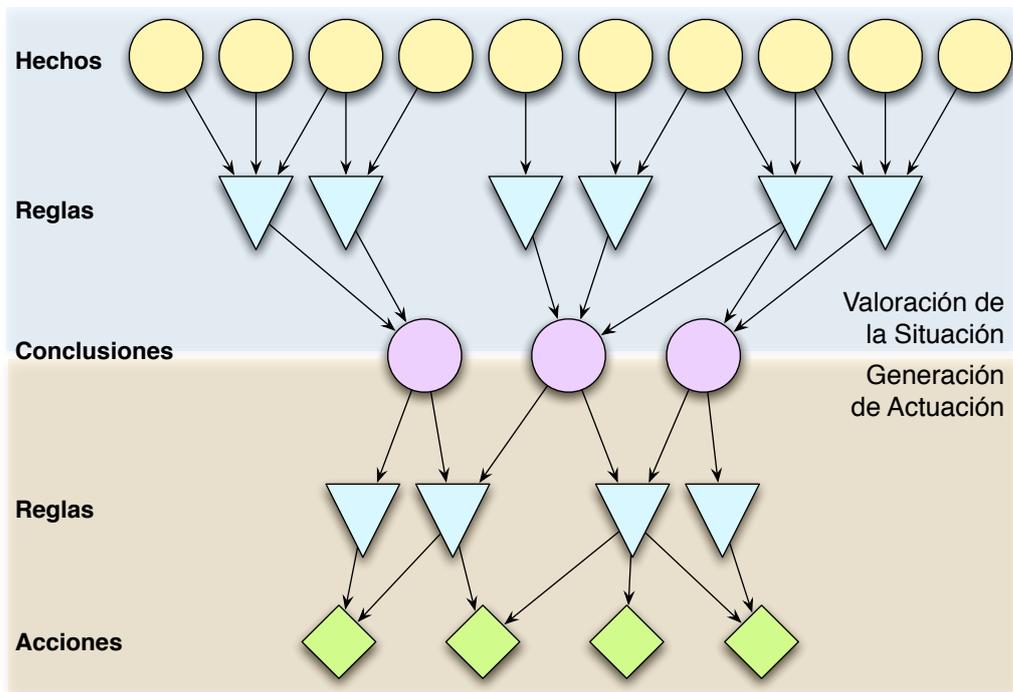


Figura 4.9: Sistema basado en reglas con estructura bicapa.

con etapa de identificación y etapa de control⁷. La base de conocimiento se descompone en dos que recogen dos tipos de información: reglas referentes a la interpretación del estado del proceso y reglas para la determinación de las actuaciones. En este caso la regla anterior quedaría descompuesta en dos reglas, cada una de las cuales entraría a formar parte de la base de conocimiento de la etapa correspondiente:

1. **Valoración de la situación:** Se determinará el estado funcional del proceso a controlar a partir de la información sobre el proceso —mediciones de variables— y posibles hechos incluidos en la base de hechos.

```
SI La temperatura del horno es
    mayor de 1500°
ENTONCES El horno esta caliente
```

2. **Generación de actuación:** Se determinará la acción de control en función del estado funcional del proceso a controlar

```
SI El horno esta caliente
ENTONCES Disminuir fuel un 10%
```

Como vemos en este tipo de estructura aparece una etapa de información intermedia que constituye una visión del estado del proceso a otro nivel que el puramente numérico; es un modelo conceptual parcial del proceso en un nivel de abstracción superior al de las puras medidas.

En la literatura se pueden encontrar multitud de referencias sobre lo que se ha dado en llamar sistemas expertos con metaniveles. Este tipo de sistemas expertos son por lo general sistemas bicapa —un solo metanivel— en los que el metanivel es un nivel de abstracción superior en el que se utiliza conocimiento sobre el nivel inferior para realizar una focalización de la base de conocimiento del nivel inferior sobre el subdominio del problema. De este modo se reduce el tamaño de la base

⁷Este tipo de estructura es habitual en sistemas expertos, como dice D' Ambrosio [D' Ambrosio 89], se divide el problema del control en dos etapas: valoración de la situación y planificación de la respuesta. Es la utilizada por ejemplo en CASNET y en EMYCIN [Buchanan 84].

de conocimiento activa de forma que se reduce sobremedida el costo computacional-temporal de la inferencia.

Este tipo de sistemas conduce a implantaciones multicapa —más de un encadenamiento— pero que conservan esta estructura de clara descomposición en dos etapas: valoración de la situación y generación de la actuación.

SCBR multicapa

En general los sistemas basados en reglas tienden a crecer de forma incontrolada, ya que la posibilidad de desarrollo incremental de los mismos, considerado por algunos como una ventaja, hace que se añadan sucesivas mejoras que tienden a saltarse a la torera los diseños estructurales primarios. Así se alcanzan sistemas en los que el modelo conceptual intermedio es un farrago de información inestructurada, y en los que el proceso de inferencia funciona en base a mecanismos basados en la suerte.

No obstante es posible el desarrollo de sistemas conceptualmente claros, en los que el proceso de inferencia avanza a través de sucesivas etapas de abstracción. En este tipo de sistemas la generación de las acciones de control se pueden producir en todos los niveles de abstracción dado que los criterios de control son por lo general múltiples, aunque por lo general se suelen alcanzar estructuras similares al bicapa.

En el capítulo 5 se expondrá una metodología de desarrollo de sistemas de control inteligente que permite la construcción de SCBRs multicapa en los que existe este reparto de autoridad por niveles de abstracción.

Modelado implícito en SCBRs

En este tipo de sistemas se utiliza un modelo implícito. El grado de explicitación del modelo es variable, dependiendo del tipo de SCBR. El los sistemas con varias capas este modelo es más explícito que en el caso de los sistemas monocapa. Los hechos intermedios que sirven de enlace entre las capas del SCBR constituyen modelos del proceso en diferentes niveles de abstracción. A estos

modelos utilizados en los SCBRs se los suele denominar modelos someros, ya que representan un conjunto reducido de aspectos del sistema a controlar, esencialmente comportamientos, sin expresar en ningún momento relaciones causales entre hechos que suceden en el proceso a controlar.

Ventajas del control basado en reglas

Las ventajas del control basado en reglas se aprecian desde la perspectiva del control convencional. En determinadas situaciones las prestaciones de los sistemas de control convencional no son lo suficientemente buenas. Esto se da especialmente en las circunstancias mencionadas por Ilari [Ilari 87] y comentadas en el apartado 4.2.1.

La ventaja de este tipo de controladores basados en reglas es que permiten expresar la estrategia de control de forma verbal, pudiendo utilizar el conocimiento de operadores humanos. Por ello el control experto no precisa de un modelo matemático del proceso a controlar. Otra ventaja es que el control experto tiene como base un conocimiento expreso —explícito—, que puede ser comprobado mas o menos exhaustivamente —aunque las técnicas de validación de sistemas expertos sean aún muy limitadas. El control experto es además un control más estable que el humano, en el sentido de que la estrategia de control no tiene variabilidad, lo que redundaría en una estabilidad de operación, que es uno de los grandes problemas de la industria de proceso, en la que los turnos de operadores hacen que el proceso esté sometido a oscilaciones originadas por el cambio de las estrategias de control de los operadores. Este aumento de estabilidad redundaría en una mejora de calidad del producto, un aumento de producción al permitir la conducción en condiciones mas cercanas a los límites y una disminución de las necesidades de mantenimiento al reducir las fatigas introducidas por la operación fluctuante.

Desventajas del control basado en reglas

Las desventajas del control basado en reglas se centran en dos puntos, uno relativo a la dificultad de manejo del conocimiento y otro relativo a los problemas de implantación. En el problema de

manejo del conocimiento aparecen dos puntos especialmente importantes:

- La adquisición del conocimiento, que debe hacerse en este caso de los operadores de proceso y eventualmente de los ingenieros de proceso o control. La intervención del ingeniero del conocimiento es necesaria, al menos en una primera etapa, para clarificar este conocimiento, dado que el personal encargado de estas misiones suele ser personal de baja capacitación técnica. En etapas sucesivas la idea que se persigue es la de aumentar la formación de los operadores y de los ingenieros de planta en este tipo de técnicas, de forma que sean ellos mismos capaces de realizar esta clarificación e incluso la implantación.
- La formalización, implantación y utilización del conocimiento en el sistema de control basado en reglas puede llegar a ser un problema muy complejo. Estas actividades son misión del ingeniero del conocimiento, pero la tendencia, como se mencionó en el punto anterior, es formar al personal de planta en estas técnicas; de esta forma se permite a dicho personal el mantenimiento y actualización del SCBR; se requiere, no obstante, el desarrollo de un entorno de proceso mediante el cual se puedan realizar estas actividades de mantenimiento y actualización.

Por lo que a problemas de implantación se refiere, los sistemas basados en reglas se caracterizan por su extremada lentitud y por la duración impredecible de sus cadenas de inferencia —salvo en sistemas simples como los monocapa. El tiempo que se requiere para determinar una acción de control es dependiente de las circunstancias particulares, tanto del problema, como de la máquina que soporta el proceso de inferencia, sin que exista una manera clara de realizar una estimación del mismo⁸. La dependencia del problema tiene su origen en la anisotropía del sistema basado en reglas con respecto a los vectores de estado introducidos en

⁸Por lo general la inteligencia artificial trabaja con gestión dinámica de memoria, por lo que la situación particular de la misma en un momento dado determina sobremedida las prestaciones del sistema.

el. En algunos casos, como son los sistemas de control basados en reglas borrosas clásicos —sistemas monocapa— el tiempo de inferencia sí es determinable, siendo además el mismo para cualquier estado inicial de partida —vector de estado. La lentitud, junto con las duraciones impredecibles dificultan sobremanera la utilización de SCBRs integrados en aplicaciones de tiempo real.

Valoración

Como puntos importantes a considerar en el uso de estos sistemas de control basados en reglas cabe destacar los siguientes; algunos de ellos se comentan en [Shirley 87]:

- Los sistemas expertos no son una panacea. Aunque resuelven problemas complejos, no son aplicables en cualquier caso.
- Los sistemas expertos no son expertos, sino que recogen una parte muy pequeña de la experiencia del humano, careciendo casi por completo de sentido común.
- Aunque sus costes de desarrollo son elevados están disminuyendo progresivamente, debido al aumento de experiencia de desarrollo, a la mejora de metodologías y a la aparición de herramientas específicas.
- La distinción entre sistema de desarrollo y sistema final es crucial. Los sistemas tendrán requerimientos muy diferentes, tanto en hardware como en software.
- Las herramientas de desarrollo deben ser abiertas.
- Es fundamental el desarrollo adecuado de la interfase de operador.
- En el caso de sistemas con vínculos con el mundo, las interfases con él son de una importancia crucial. Este tipo de elementos son característicos de los sistemas de control y de algunos de diagnóstico.
- El control experto es por lo general fuertemente no lineal. Esto en ocasiones puede ser interesante, pero en otras genera bastantes

problemas: oscilaciones, sacudidas, etc. Esta no linealidad puede compensarse en parte utilizando técnicas de ponderación de reglas, como son todas las utilizadas en manejo de incertidumbre [González 89]. El control borroso, que se explicará en el punto siguiente por su especial importancia, usa una de estas técnicas.

4.5.2. Control borroso

El control basado en reglas presenta en determinadas circunstancias un mal comportamiento, sobre todo debido a las fuertes no linealidades del mismo. El control borroso puede considerarse un control basado en reglas que utiliza las técnicas borrosas para manejar imprecisión [Tong 77]. El considerarlo como un paradigma independiente se debe a que su estructura (monocapa) y su implantación (compilación de reglas) son muy específicas de este tipo de control. Los reguladores borrosos tratan de implantar estrategias de control expresadas en términos lingüísticos por los operadores de proceso; para ello se basan en técnicas de lógica borrosa [Zadeh 73] [Mamdani 81].

Estructura de un regulador borroso

Esencialmente se basan en un conjunto de reglas de control borroso de la forma:

```
SI Variable A es Grande y
   Variable B es Pequeña
ENTONCES Actuación es Grande
```

Los términos que aparecen en las expresiones —grande, pequeña— se denominan términos lingüísticos [Zadeh 88]. Todo el conjunto de reglas equivale a una función del espacio de entradas en el espacio de salidas. Un aspecto fundamental de este tipo de reglas es el número de niveles de cuantificación de entradas: Corresponde al número de términos lingüísticos utilizados para representar los valores de las entradas. Esta conversión a valores borrosos es el primer paso en el proceso de control. Un número de valores empleado habitualmente es 7. La decisión de este número es impor-

tante desde dos puntos de vista: la representatividad y el volumen de los datos.

Las etapas del control son:

- Conversión de los valores medidos de las variables del proceso a su representación borrosa.
- Aplicación de las reglas para obtener los valores de las actuaciones.
- Obtención de valores *nítidos* para las actuaciones a partir de sus valores borrosos.

Técnicas de aplicación del control borroso

Para aplicar este tipo de control se usan dos técnicas fundamentales:

- Control directo por reglas: En este caso en cada ciclo de control se realiza el cálculo de la actuación a partir de la unificación —proceso de búsqueda de reglas que se adecúan a la situación actual— de la base de reglas de control con los hechos procedentes del proceso —valores medidos de las variables de proceso.
- Compilación de reglas: Este es un enfoque en el que la base de reglas se “compila”, en el sentido de pasarla a una representación más eficiente. De este modo se obtiene una tabla en la que para cada combinación de valores borrosos de las variables de control se tiene el valor borroso de la actuación. Este enfoque sólo es viable mientras el conjunto de reglas no cambie; si se produce este cambio es preciso volver a calcular la tabla.

Supervisión de reguladores borrosos

Los reguladores borrosos basados en reglas admiten la utilización de módulos de supervisión, con resultados más que aceptables [Cerezo 88] [Cerezo 88]. En [Oliveira 90] [Oliveira 90] se presenta una arquitectura para la supervisión de este tipo de reguladores, con la que se consiguen muy buenos resultados. En las experiencias con dicha arquitectura de supervisión se observa que los resultados conseguidos por los reguladores borrosos

supervisados son similares a los logrados por reguladores PID convencionales, con ventajas claras en caso de plantas con funciones de transferencia desconocidas o variantes.

En las pruebas realizadas se utiliza para la planta un modelo SISO con funciones de transferencia de segundo orden. El regulador borroso se comporta —por lo que a sobreoscilación y tiempo de establecimiento se refiere— sensiblemente igual que un PID ajustado por Ziegler-Nicholls. En caso de plantas variantes, el mecanismo de supervisión actúa en modo aperiódico, logrando la adaptación a la planta —recuperación de prestaciones— en un tiempo aceptable. Dicho regulador borroso obtiene resultados adecuados incluso con plantas con funciones de transferencia de fase no mínima.

Valoración

Este tipo de reguladores presentan algunas ventajas de cara a su implantación para control de proceso. Las más importantes son las siguientes:

- Permiten utilizar el conocimiento operacional sobre el control del sistema, obtenible de expertos humanos.
- Son reguladores más robustos.
- Permiten realizar control multivariable de forma simple.
- Pueden alcanzar comportamientos mejores que reguladores PID clásicos.
- Admiten diversos modos de supervisión.

Las metodologías a utilizar son claras, pero falta por determinar aspectos en los que se han realizado pocos desarrollos:

- Estabilidad. Se han hecho algún tipo de estudios de estabilidad [Cerezo 88][Aracil 89], aunque bajo condiciones muy estrictas para el sistema analizado. No existe un metodología genérica para realizar estudios de estabilidad de reguladores borrosos.

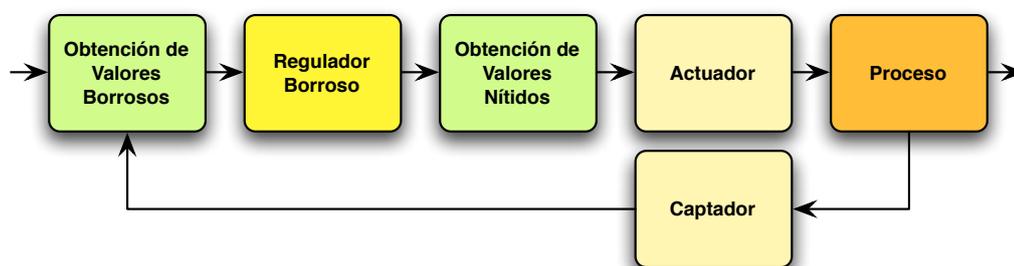


Figura 4.10: Estructura de un sistema de control basado en reglas borrosas.

- Viabilidad para procesos complejos. En caso de procesos complejos —muchas variables— el número de reglas necesario para el control puede llegar a ser excesivo. Es preciso restringir los reguladores borrosos a sistemas simples.
- Obtención de las reglas: Este es el conocido cuello de botella de los sistemas expertos, la adquisición del conocimiento.

La aplicación de este tipo de reguladores para el control de sistemas complejos es adecuado, siempre que su uso se sitúe en un nivel bajo, encargado del control de bucles simples. Este tipo de regulador es adecuado para sistemas jerárquicos de control de procesos complejos.

4.5.3. Control basado en modelos

El razonamiento basado en modelos es la denominación genérica de un conjunto de técnicas convergentes que constituyen el elemento central de los sistemas expertos de segunda generación. En este tipo de sistemas la base de conocimiento contiene un modelo explícito del sistema a controlar. En vez del modelo de comportamiento implícito expresado por las reglas del caso anterior, el modelo es en este caso fundamentalmente causal [Atwood 86] [Atwood 86]. A este tipo de modelos se los suele denominar modelos *profundos*, mientras que a los modelos de comportamiento se los denomina modelos *someros*.

En la Figura 4.5.3 se presenta la estructura de un sistema basado en modelos. En él la base de conocimientos aparece dividida en tres partes:

- El modelo. Compuesto a su vez de dos partes: la base de hechos (H), que contiene información factual sobre el estado del proceso y la base de estructura (E), que contiene la información que es propiamente el modelo del proceso.
- La base de conocimiento del dominio (BCD). Que contiene conocimiento similar al que se utiliza en los sistemas convencionales.
- La base de conocimiento general (BCG), que contiene conocimiento relativo a la manera en que se utiliza el conocimiento contenido en el modelo.

En el modelo se incluye toda la información relevante del sistema sobre el que se va a razonar. Esta información se estructura en forma de modelo explícito del sistema. La base de conocimiento general expresa conocimiento utilizado en la interpretación de la información del modelo. Esta base de conocimiento es propia de esquema representativo utilizado para la construcción del modelo, por lo que en muchos casos no tendrá una formulación declarativa sino procedural, implícita en los mecanismos de manejo del modelo. Este tipo de integración de la base de conocimiento general en el propio sistema de gestión del modelo es propio de sistemas que sólo admiten un mecanismo de construcción del modelo o de aquellos que

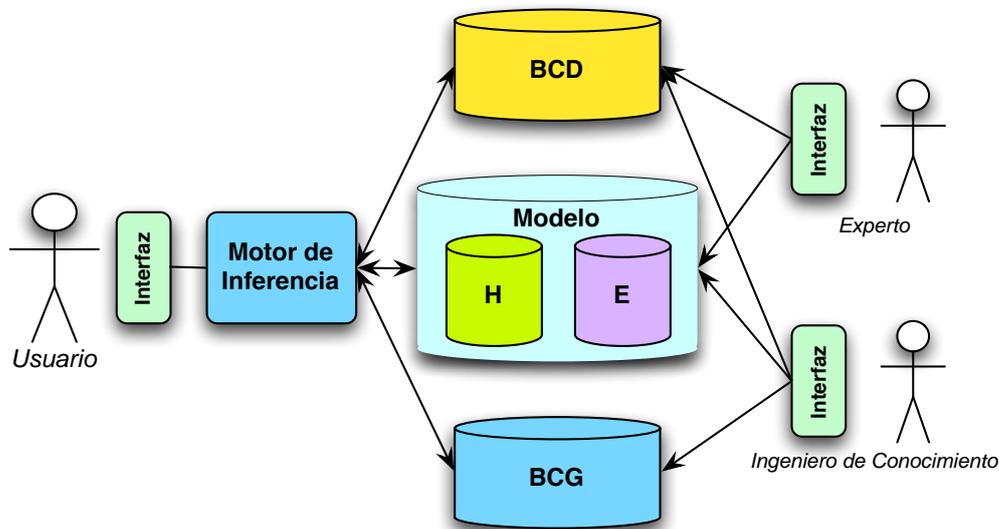


Figura 4.11: Sistema basado en modelos.

persiguen una gran eficiencia. El conocimiento del dominio es conocimiento particular sobre la aplicación y que no ha podido incluirse en el modelo por sus especiales características.

La utilización de programación orientada a objetos [Meyer 88] es lo que ha permitido el uso eficiente de representaciones de sistemas basadas en modelos. En particular en el campo de control de proceso, la orientación a objeto ha supuesto la diferencia entre la inoperancia y el funcionamiento en tiempo real.

Dimensiones del conocimiento

El modelo que se persigue en este tipo de sistemas es una representación del conocimiento que se dispone sobre el proceso. Es más extenso que el empleado en sentido matemático, teniendo cuatro facetas (dimensiones del conocimiento) [Simmonds 88]:

Amplitud: El modelo incluye información de toda una gama de objetos del mundo real en relación con nuestro sistema. Nos permite almacenar y utilizar nuestro conocimiento so-

bre estos objetos.

Profundidad: El modelo permite trabajar a diferentes niveles de profundidad, que enlazan de forma ascendente por mecanismos de abstracción. Esto hace posible la utilización de conocimientos por sistemas de razonamiento a diferentes niveles. Por ejemplo un sistema de diagnóstico puede profundizar en las estructura del modelo para determinar la causa de una determinada falta, realizando una navegación en profundidad —adentrándose en niveles sucesivos. Un sistema de planificación —por contra— trabaja en un nivel elevado, estableciendo la forma en que se debe operar el sistema para conseguir maximizar una función de utilidad.

Certeza: Nuestro conocimiento del sistema tiene diferentes grados de certeza, cuyo conocimiento es preciso para tomar las acciones correctas.

Riqueza: Diremos que nuestro conocimiento es rico cuando incluya diversos aspectos (que, como, cuando, por qué, etc).

Tipos de modelos

El enfoque de control por razonamiento basado en modelos parece viable desde dos puntos de vista:

- Utilizando *Modelos Externos*: Estos modelos están constituidos por reglas que expresan el comportamiento del sistema. Estas reglas permiten determinar en un momento dado el *estado* del sistema, y, en algunos casos, mediante razonamiento temporal la *evolución* del mismo. Este tipo de modelos son costosos en ejecución, ya que la base de reglas tiene un volumen muy elevado que no permite alegrías. Los mecanismos de focalización — ya mencionados— permiten disminuir el conjunto conflicto, y por ello aumentar las prestaciones del sistema. Este tipo de sistema de modelado precisa de paralelización para alcanzar la eficiencia necesaria para su uso en tiempo real. Los modelos someros utilizados en algunos SCBRs son de este tipo.
- Utilizando *Modelos Internos*: En estos se utiliza la técnica de especificar los elementos del sistema y sus interrelaciones. De este modo se reduce el volumen de la información a manejar, ya que el comportamiento quedará descrito en base a restricciones genéricas sobre los tipos de dispositivos a emplear. Estas técnicas de modelado profundo para razonamiento sobre sistemas se encuentran en la actualidad en etapas tempranas de desarrollo.

Los modelos del mundo que constituyen la base de razonamiento de este tipo de sistemas se pueden especificar a priori, o bien ser construidos sobre la marcha. Este enfoque es el que se emplea en sistemas avanzados de control de vehículos autónomos, en que el modelo del mundo que rodea al vehículo se va construyendo y/o modificando en base a la información sensorial recibida por el ente autónomo. Si el sistema autónomo debe interactuar con su entorno debe tener también un modelo de sí mismo.

En el caso del control de procesos, no se da esta circunstancia de forma completa, ya que el sistema de control no interactúa con el proceso salvo a través de los actuadores y captadores. Es por

ello necesario que el modelo incluya submodelos correspondientes a los sistemas de captación y actuación. Estos modelos permitirán a la interfase de proceso el filtrado inteligente de la información procedente del sistema, así como la actuación eficaz y segura.

Una de las ventajas fundamentales de disponer de conocimiento estructurado en distintos niveles de abstracción es la posibilidad de realizar razonamiento cualitativo.

4.5.4. Ventajas del control basado en modelos

El control por razonamiento basado en modelos tiene como principal desventaja su mayor complejidad en el desarrollo de mecanismos de razonamiento de control. Pero las ventajas que presenta hacen que su utilización sea progresivamente extendida:

- Flexibilidad, ya que es posible formular y utilizar el modelo de diferentes maneras.
- Extensibilidad, puesto que tanto el modelo como las aplicaciones que soporte se pueden extender de forma simple para reflejar los cambios en el sistema o en el problema.
- Compartibilidad, pudiendo ser utilizado *simultáneamente* por diferentes aplicaciones.
- Coste y tiempo reducido de desarrollo, ya que es un sólo modelo para varias aplicaciones, y además la construcción de modelos explícitos es más simple y directa que la de modelos implícitos como los contenidos en un sistema basado en reglas.
- Integrabilidad, ya que su no vinculación a una perspectiva de resolución de problemas determinada permite su integración en cualquier otro tipo de sistema, aportando una información coherente a todo un grupo de aplicaciones.
- Constructibilidad puesto que es posible la construcción y uso de librerías de modelos, que permiten construir modelos de sistemas más grandes.

- Fiabilidad, ya que en un modelo explícito es más fácil realizar una verificación que permita garantizar su coherencia y completitud.
- Jerarquizabilidad, admitiendo representaciones multiresolucionales.
- Mayor facilidad para expresar el conocimiento sobre el objeto modelado. Este tipo de ventaja es especialmente importante en los modelos orientados a dispositivos [Geneserteh 84].

Este último punto es especialmente importante desde el punto de vista del desarrollo, ya que el cuello de botella de la adquisición del conocimiento se reduce, puesto que el experto expresa sus conocimientos sobre el dominio en sus propios términos. En los últimos sistemas de desarrollo de aplicaciones de este tipo se dispone de una interfase muy potente que permite construir los modelos de forma gráfica e interactiva, utilizando los elementos habitualmente conocidos por el personal de planta. De este modo la misión del ingeniero del conocimiento se reduce, ya que el experto se convierte en ingeniero del conocimiento. Se produce una migración aún más acusada que lo que venía siendo habitual en sistemas convencionales.

4.6. Desarrollos actuales

Hasta ahora los desarrollos reales en el área de control inteligente han sido más bien escasos, debido a las dificultades de implantación de estas características, sin alcanzar en ningún caso la operatividad deseada en un sistema que incluyera todas ellas. Sin embargo existen algunos sistemas que presentan algunas de las características mencionadas con anterioridad. Estos sistemas se pueden clasificar en varios tipos:

- Desarrollos específicos:
 - Desarrollos independientes.
 - Desarrollos con herramientas de control inteligente.
- Evoluciones:

- Evoluciones desde el campo de control.
- Desde el campo de la inteligencia artificial.

El grado de operatividad alcanzado por los sistemas de control inteligente es muy variado, ya que intervienen en su valoración multitud de factores.

4.6.1. Desarrollos específicos

Los desarrollos específicos son sistemas de control y en especial entornos de desarrollo de sistemas de control inteligente orientados hacia este tipo de problemas, y no a aplicaciones generales. Por ello presentan una serie de características que los hacen especialmente interesantes.

4.6.2. Desarrollos independientes

Los desarrollos de sistemas de control inteligente han seguido en muchos casos una trayectoria similar compuesta de tres etapas:

- Desarrollo en un entorno para sistemas expertos de aplicabilidad general.
- Reescritura en un lenguaje convencional para alcanzar operatividad.
- Construcción de un nuevo entorno basado en el desarrollo anterior.

El alcanzar la segunda y tercera etapas dependen esencialmente del presupuesto del proyecto (segunda etapa) y del presupuesto y objetivos (tercera etapa).

Son de destacar en este grupo los proyectos dentro del marco de los programas ESPRIT y EUREKA de la Comunidad Europea. Varios de estos proyectos de investigación son asimilables al área de control inteligente. De ellos podemos mencionar:

- El sistema TEX-I [Voss 88]: Technical Expert Systems for Interpretation, Diagnosis and Control of Technical Systems.

- El sistema QUIC [Cavanna 89]: Design and Experimentation of a Knowledge Based System Development Toolkit for Real Time Process Control Applications.
- Otros como: IPCES, AIMBURN, TOPMUSS, KRITIC, FIABEX [Cuena 89].

Es de destacar que este tipo de proyectos no son de desarrollo de un sistema de control específico, sino que sus objetivos son de investigación y desarrollo fundamentalmente de herramientas y sistemas no específicos de una aplicación, aunque en casi todos ellos se construyan aplicaciones demostrativas. Estos proyectos tienen propósitos muy ambiciosos. Es de esperar que del esfuerzo conjunto, una vez que los resultados de dichos proyectos alcancen al público, surja un grupo de técnicas de clara aplicación en control de procesos.

Por ahora estas técnicas no existen —al menos claramente— por lo que cada diseñador debe hacer frente a una serie de problemas similares, pero nunca iguales; esto hace que las soluciones sean ad hoc en la mayor parte de los casos, limitando de este modo la posible reutilización del software desarrollado.

4.6.3. Desarrollos con herramientas de control inteligente

La necesidad de herramientas ha conducido a una serie de sistemas que tratan de implantar las características deseables en un sistema de control inteligente. Estas implantaciones tienen mayor o menor éxito, pero en cualquier caso han dado lugar a sistemas más adecuados para el desarrollo de aplicaciones con requerimientos de operación en tiempo real. Las perspectivas de este tipo de herramientas son cada vez más halagüeñas tanto por la mejora continua en hardware como por el desarrollo de nuevas técnicas y metodologías software que posibilitan la implantación de sistemas de control inteligente.

Los entornos específicos para sistemas de control inteligente utilizan mecanismos de reglas para la expresión de los conocimientos de control. Estos sistemas son pocos en la actualidad, frente al inabundante número de herramientas de propósito ge-

neral, pero están alcanzando una rápida difusión, sobre todo en los últimos dos años.

G2 es un entorno para desarrollo de sistemas de control inteligente para grandes procesos industriales [Gensym 89]. Es un sistema potente y versátil, pudiendo utilizarse en gran número de máquinas y permitiendo arquitecturas distribuidas sobre una red. El sistema dispone además de mecanismos de simulación para predicción de comportamientos del sistema bajo control. Es la única herramienta diseñada específicamente para control y que ha alcanzado un grado de difusión elevado. Es el sucesor de otra herramienta denominada PICON ligada a las máquinas Lisp de LMI con procesador adicional 68010 para cómputo numérico e interacción con el exterior. G2 no está ligado a ninguna arquitectura específica ni requiere de ningún tipo de lenguaje soporte⁹ (Common Lisp en el caso de PICON).

NEXPERT OBJECT Se presenta como un juego de herramientas para desarrollo de bases de conocimiento, junto con una librería de funciones de inteligencia artificial integrable en aplicaciones de usuario. Su estructura dirigida por eventos lo hace adecuado para su utilización en aplicaciones de control. Su orientación no es exclusiva hacia el control.

IDSEC es una herramienta de sistemas expertos en tiempo real para aplicaciones industriales ligada a las máquinas Lisp de Symbolics con procesador Ivory. Combina la potencia de cómputo simbólico del procesador lisp con un procesador convencional. Es muy similar a la herramienta G2.

Otros productos de menor tamaño son CHRONOS [Euristic 89], COMDALE y NEMO [S₂O 90] ofreciendo posibilidades de razonamiento temporal sin necesidad de utilizar grandes equipos.

4.6.4. Evoluciones

Las evoluciones desde otras áreas han conducido a sistemas que presentan utilidad en control inteligente, pero que no alcanzan la versatilidad o la potencia necesaria para su utilización en gran

⁹Aunque G2 corre sobre una versión especial de Common Lisp preparada para su uso en tiempo real, este Common Lisp forma parte del propio G2, no siendo accesible para el usuario.

cantidad de problemas de control.

Evoluciones desde el campo de control

Esta evolución conduce a herramientas obtenidas por generalización y evolución de sistemas de control clásicos. Los fabricantes de sistemas de control que implantan metodologías clásicas tratan de mantener sus cuotas de mercado por la introducción continua de avances en sus sistemas.

En este punto cabe hacer especial mención de dos sistemas que hacen uso de las técnicas borrosas para implantar reguladores basados en reglas. Estos sistemas son LinkMan de Sira Ltd. (U.K.) y FCL de la empresa danesa F.L. Smidth & Co. A/S [Holmblad 87].

Otro tipo de sistemas son los que incluyen técnicas de sistemas expertos para realizar reguladores calificables de adaptativos, en los que la adaptación se lleva a cabo por medio de un sistema experto. Ejemplo de este tipo de sistemas son EXACT [Foxboro 86] y EXPERT-AD [Adaptech 89].

La última moda es el dotar a estos sistemas de control clásico de elementos *inteligentes* que permitan la implantación de estrategias de control más versátiles. Este es el caso de Integrale 2000, un producto que incluye a G2 como elemento de más alto nivel.

Evoluciones desde el campo de la inteligencia artificial

KEE se ha venido introduciendo en los últimos tiempos [Intellicorp 86] como un entorno para desarrollo de aplicaciones de control basadas en el paradigma de razonamiento basado en modelos. En este entorno los modelos se construyen en base a los *marcos* tradicionales, que son una manera de enfocar la programación orientada a objeto, aunque dejan mucho que desear desde el punto de vista de la ocultación de información y el polimorfismo.

PLEXYS Es un entorno de desarrollo de sistemas expertos orientado a la industria de proceso: generación de energía, química, refinerías, etc. Ha sido desarrollado en dos etapas; en primer lugar

por EPRI (Electrical Power Research Institute) y posteriormente por Intellicorp, como extensión a su producto comercial KEE. No presenta una arquitectura especial, sino que tan sólo incluye herramientas para modelado profundo de la planta. La orientación principal de este producto es la supervisión de sistemas, ofreciendo pocas posibilidades de control directo.

La adecuación de las herramientas clásicas de inteligencia artificial al control en tiempo real es limitado, por lo que las aplicaciones desarrolladas con ellas se centran en su mayor parte en el campo del diagnóstico o la planificación fuera de línea.

Proyecto C2G

En el Departamento de Automática, Ingeniería Electrónica e Informática Industrial de la UPM se está desarrollando en la actualidad una aplicación de control inteligente avanzado denominada C2G. Esta aplicación, basada en la arquitectura CONEX propuesta en esta tesis, está siendo desarrollada en colaboración con una empresa del sector cementero.

CONEX se ha diseñado como una arquitectura de control inteligente y distribuido, orientada al control de procesos con dificultades de modelado. Esta arquitectura permite la operación en tiempo real al integrar una serie de capas de control que implantan mecanismos de control que se sitúan en diferentes niveles de velocidad de operación e inteligencia, de acuerdo con el principio de precisión creciente con inteligencia decreciente [Saridis 89].

Los paradigmas de control empleados en esta arquitectura son:

- Reguladores clásicos.
- Reguladores borrosos basados en reglas.
- Control basado en patrones.
- Control experto basado en reglas.
- Control experto basado en modelos.
- Operadores humanos.

El sistema corre sobre una red de computadores, consistiendo en un conjunto cooperativo de subsistemas.

C2G se está desarrollando para un proceso particular, pero con un diseño orientado a la transportabilidad y adaptabilidad a nuevas plantas y nuevos procesos. Para ello ha sido necesaria la descomposición de las bases de conocimiento en bases dependientes de planta y bases independientes de planta, de forma que la adaptación a otro proceso consista simplemente en la reconstrucción de las bases de conocimiento dependientes de planta.

4.7. Control Inteligente Avanzado

El control inteligente es de por sí un control avanzado, ya que se sitúa por delante de los paradigmas establecidos de control. No obstante el uso de algunas técnicas especiales conduce a sistemas de control inteligente que presentan características especiales.

Entre estas técnicas se pueden destacar:

- Aprendizaje
- Arquitecturas distribuidas
- Reconfiguración

4.7.1. Aprendizaje en control inteligente

En control los modelos tienen formatos muy dispares, que nosotros hemos clasificado en varias líneas diferentes, distinguiendo entre modelos implícitos y explícitos, entre modelos numéricos y simbólicos y entre modelos internos y externos. Cada uno de estos modelos es susceptible de mejora, y, en cada uno de ellos es planteable la utilización de técnicas de aprendizaje para ello. En algunos casos estas técnicas han existido durante largo tiempo, pero en otros —quizá por sus especiales características— están en etapas muy tempranas de desarrollo. El aprendizaje en sistemas de control no es un elemento introducido por la inteli-

gencia artificial sino que se había introducido con anterioridad [Fu 70].

El aprendizaje con modelos numéricos implícitos es el caso de los sistemas de supervisión de reguladores convencionales. En ellos se consigue una mejora de las prestaciones del sistema de control en base a modificaciones del algoritmo de regulación. Estas modificaciones son por lo general paramétricas, limitándose a la modificación de determinados parámetros de los reguladores. Este tipo de mecanismo de mejora de control se basa en criterios sobre el comportamiento del sistema controlado, como pueden ser la sobreoscilación o el tiempo de establecimiento. Se han aplicado con éxito en reguladores clásicos [Nikolic 86] [Ollero 89b] e incluso en reguladores borrosos [Oliveira 90].

En el caso de los controles adaptativos clásicos el modelo es numérico y explícito. En este tipo de sistemas se realiza una identificación en línea del proceso a controlar; de este modo se actualiza un modelo explícito de la planta que se emplea para el cálculo de los reguladores. Este tipo de aprendizaje por mejora del modelo se usa en los reguladores autoajustables (STR) y en los reguladores adaptativos con modelo de referencia (MRAC) [Åstrom 88].

En los sistemas expertos construidos con estructura de primera generación es posible el uso de aprendizaje con los modelos simbólicos implícitos de este tipo de técnicas. Ejemplo de este tipo de sistemas son los propuestos por Fogarty [Fogarty 89] y Velasco¹⁰. En ellos el sistema de control basado en reglas utiliza enfoques basados en algoritmos genéticos para la generación de reglas y técnicas de premio-castigo para corregir factores de validez de dichas reglas. Este tipo de sistemas logran ajustarse al sistema con velocidad adecuada, aunque su comportamiento se ve muy afectado por circunstancias diversas, llegando a ser en cierta medida impredecible¹¹. En el punto sobre aprendizaje con modelos numéricos implícitos se comentó la existencia de mecanismos de ajuste de reguladores basados en reglas borrosas, que son asimilables al uso de modelos simbólicos más que numéricos.

¹⁰J.R. Velasco, comunicación personal.

¹¹J.R. Velasco, *ibid.*

Para los sistemas expertos de segunda generación el modelo es simbólico y explícito. En ellos se dispone de un modelo explícito del sistema a controlar, en el que de manera general existirán parámetros susceptibles de mejora. El ajuste paramétrico de los modelos es importante tanto por los errores de modelado iniciales, como por las derivas y alteraciones que sufre el proceso. Este enfoque de ajuste de parámetros es similar al utilizado en el caso de modelos numéricos explícitos. No debemos olvidar que los modelos utilizados en este tipo de sistemas expertos suelen ser multi-resolucionales, con parte de los mismos realizada usando técnicas clásicas de modelado de sistemas, léase ecuaciones diferenciales. Los modelos basados en ecuaciones diferenciales son sin duda los más adecuados —cuando es posible formularlos [Fishwick 90].

4.7.2. Arquitecturas distribuidas

Las arquitecturas distribuidas para control inteligente suelen seguir dos enfoques diferentes: los sistemas jerárquicos y los sistemas cooperantes [Mladenov 89]. Este tipo de arquitecturas persiguen dos tipos de objetivos:

- Aumento de la potencia del sistema, gracias a la intervención de varias máquinas y a la posibilidad de dimensionar las máquinas para las tareas que van a realizar.
- Aumento de la seguridad de operación, utilizando enfoques de arquitecturas tolerantes a fallos tanto hardware como software.

Este tipo de avances no son específicos de los sistemas de control inteligente, sino que se dan en todo tipo de sistemas informáticos. En el campo de sistemas de control inteligente son especialmente importantes por la necesidad de grandes potencias de cómputo que tienen estos sistemas y por la dificultad de verificación de los sistemas basados en inteligencia artificial.

En las arquitecturas cooperantes los sistemas se descomponen en una serie de subsistemas que logran el objetivo final en base a una cooperación.

Cada uno de los subsistemas sólo es capaz de resolver una parte del problema, pero todos ellos de forma conjunta pueden resolver el problema total.

Este tipo de arquitecturas son especialmente interesantes desde el punto de vista del desarrollo, ya que la limitación de tamaño de los subsistemas —que son en sí sistemas completos— facilita sobremanera su construcción y verificación.

Entre las tecnologías de sistemas distribuidos y de cooperantes se produce un efecto sinérgico que está dando lugar a importantes desarrollos en el área de control inteligente.

Capítulo 5

Metodología de Diseño

When the decision time comes, the making and implementation of a decision cannot be postponed

*Mesarovic, Macko and Takahara
Theory of Hierarchical, Multilevel, Systems*

5.1. Introducción

Hasta ahora los sistemas de control inteligente han sido fundamentalmente sistemas basados en reglas. El uso de estos sistemas para control de procesos complejos conduce inexorablemente a la generación de bases de conocimiento masivas y poco eficientes. Esta masividad tiene su origen en la complejidad del proceso a todos los niveles y en que uno de los objetivos principales del control inteligente es la robustez en un amplio rango de condiciones de funcionamiento, por lo que es preciso especificar las acciones para cada una de las circunstancias que se pueden dar en el sistema mediante el conjunto de reglas pertinente.

Un enfoque habitual en los últimos tiempos es el uso de sistemas basados en reglas estructurados en metaniveles. Un metanivel es un conjunto de reglas cuyo dominio son también reglas. Por lo general este tipo de arquitecturas de sistemas expertos constan de una base de metaconocimiento y una serie de bases de conocimiento sobre el dominio. La base de metaconocimiento sirve para, a partir de los hechos iniciales, seleccionar la base o bases de conocimiento sobre el dominio a utilizar para resolver el problema en cuestión. De este modo se logra una focalización más adecuada sobre el problema, aumentando las prestaciones del sistema ya que la mayor parte de las reglas no per-

tinentes no se utilizan, con lo que se evita el costoso casamiento con las mismas. Este tipo de sistema —sistema experto con metanivel— es un claro ejemplo de un sistema descompuesto en dos niveles análogos a los empleados en sistemas bicapa.

Otro tipo de arquitectura, aunque menos habitual, es la representada por los sistemas cooperantes. En ellos una serie de sistemas expertos cooperan para resolver un determinado problema. Cada uno de los sistemas expertos tiene una base de conocimiento que le permite resolver un determinado tipo de situaciones. El enfoque de división de la base de conocimiento no es separar problemas diferentes en diferentes bases, sino separar los distintos aspectos de un determinado problema. Ejemplos típicos de sistemas con arquitectura de este tipo son los sistemas HEARSAY II y III [Gevarter 84][Bundy 86]. Este tipo de descomposición funcional conduce a sistemas especialistas en determinadas actividades.

En los sistemas autónomos es preciso el uso de información procedente de fuentes muy diversas y con niveles de abstracción muy diferentes, por ello es preciso utilizar mecanismos de razonamiento para todos estos niveles de abstracción. Esto nos conduce a la descomposición del sistema basado en el conocimiento en varios, tratando cada una información a un determinado nivel, que varía desde el puramente numérico hasta los niveles simbólicos más abstractos.

En resumen podemos decir que cuando el dominio es muy amplio la descomposición del sistema inteligente en subsistemas es un enfoque no solo adecuado, sino que tanto desde el punto de vista de la operación normal, como del desarrollo

parece inevitable su utilización. Este enfoque puede ser jerárquico o heterárquico, con descomposición guiada por dominios, por funcionalidad o por nivel de abstracción.

Ya [Saridis 83] menciona la necesidad de descomposición jerárquica de los sistemas de control inteligente. Su estructura en tres niveles — ejecución, coordinación, organización— permite realizar una descomposición funcional del sistema. Este es un objetivo muy deseable desde el punto de vista de la claridad de operación, facilidad de desarrollo, seguridad y eficiencia del sistema. No obstante, las descomposiciones jerárquicas pueden llevarse más lejos que la simple descomposición funcional para claridad intelectual. La descomposición —fundamentalmente jerárquica— de los sistemas de control nos permitirá alcanzar un objetivo esencial en control, y de difícil consecución en el caso de sistemas inteligentes: la operación en tiempo real.

Esta lentitud que caracteriza a los sistemas basados en reglas es debida a la explosión combinatoria asociada a algunas características del funcionamiento de dichos sistemas, en particular a los procesos de casamiento. Los enfoques utilizados para resolver este problema pasan por lo general a través del desarrollo de arquitecturas hardware específicas —máquinas LISP, ordenadores de 5ª generación, procesamiento en paralelo— que resuelven parcialmente el problema. En la mayoría de los casos este problema de velocidad no es crítico, ya que por lo general es un ser humano el que interactúa con dichos sistemas, siendo habitualmente más lento que él. Sin embargo este hecho no es generalizable a los sistemas de control basados en reglas (SCBRs), dado que estos sistemas tienen vínculos con el mundo. La necesidad de velocidad de operación se acentúa en la medida en que estos sistemas son requeridos para operar en tiempo real en sistemas progresivamente más rápidos, como puede ser el caso de utilización de sistemas inteligentes embebidos en aviónica.

En este capítulo analizaremos la problemática de la operación en tiempo real para sistemas de control basados en el conocimiento —en particular reglas—, proponiendo un método novedoso de diseño e implantación de los mismos —una metodología de diseño de arquitecturas— que permite

una mejora de prestaciones para sistemas de control inteligente. Estas arquitecturas permitirán el uso de estos sistemas en tiempo real. Esta metodología se basa en la *descomposición* del sistema por niveles de abstracción, dominios y funciones. Este enfoque de descomposición múltiple es novedoso, aunque se aproxima a métodos particulares de descomposición propuestos por otros autores.

¿Por qué hablar de una metodología de desarrollo de arquitecturas y no de una arquitectura específica?. Los problemas en los que el control inteligente es útil son problemas muy complejos, que requieren la integración de una serie de técnicas y el funcionamiento conjunto de gran cantidad de mecanismos —procedentes de todas las áreas de proceso de información— para obtener resultados aceptables. Por ello las arquitecturas de control inteligente deben ser *arquitecturas orientadas al dominio*. En realidad esto no es nada nuevo, ya que como se comentó en el capítulo anterior, casi todos los desarrollos independientes de sistemas de control, empiezan ciñéndose a arquitecturas genéricas proporcionadas por herramientas de desarrollo, pero cuando los requisitos de operación empiezan a ser fuertes —en particular, cuando hay requisitos fuertes de operación en tiempo real— es precisa la reconstrucción independiente del sistema. Estas reconstrucciones independientes implantan arquitecturas específicas para el problema de control a resolver.

Aparte de un aumento de prestaciones, la descomposición del sistema reporta otros tres grandes beneficios: facilita el desarrollo por equipos de trabajo diferentes, aumenta la verificabilidad y por lo tanto la fiabilidad y además favorece la reutilizabilidad. Como dice Parnas [Parnas 90] refiriéndose al software con seguridad crítica, éste debe mantenerse tan pequeño y simple como sea posible, por medio de trasladar las funciones que no son críticas a otros computadores, o en nuestro caso a otros subsistemas.

Las descomposiciones de sistemas de toma de decisiones pueden ser fundamentalmente de dos tipos: jerárquicas y heterárquicas. En una descomposición jerárquica existen relaciones de precedencia entre los diferentes sistemas, de forma que se tiene una relación de orden parcial. En una descomposición heterárquica, por contra, no se tiene

esta relación de orden parcial. Las arquitecturas heterárquicas son difíciles de encontrar, debido a los graves problemas para su desarrollo y verificación, por lo que la metodología más extendida — en el campo industrial sobre todo— es la jerárquica.

En este capítulo se persigue un objetivo triple:

1. Por una parte estudiar las arquitecturas jerárquicas como un medio de construcción de grandes sistemas de control en los que se deben integrar diferentes metodologías.
2. Por otra analizar las características de los principales paradigmas de control basado en el conocimiento para su utilización en control en tiempo real.
3. Por último proponer una metodología de análisis y diseño de arquitecturas de control inteligente basada en descomposición.

5.2. Objetivos de la descomposición

Los objetivos de la metodología de descomposición de sistemas de control inteligente son los siguientes:

- Aumentar las prestaciones, sobre todo el aspecto de velocidad, que permitan el uso de este tipo de sistemas en control de procesos.
- Posibilitar el control de los tiempos de ejecución, de forma que se puedan establecer arquitecturas de control que permitan la operación en tiempo real, tanto con restricciones débiles como fuertes.
- Facilitar el proceso de desarrollo, ya que el descomponer los sistemas permitirá el desarrollo de los mismos por equipos diferentes.
- Aumentar la verificabilidad de los sistemas. Las bases utilizadas por los sistemas basados en conocimiento son mas reducidas de tamaño, y además descompuestas funcionalmente, lo que facilita el análisis a fondo de las mismas.

- Posibilitar el desarrollo de arquitecturas avanzadas, con más dosis de *inteligencia*.
- Aumentar la reutilizabilidad del software desarrollado.

5.3. Sistemas Jerárquicos

En este apartado seguiremos a Mesarovic [Mesarovic 70] en la exposición de una serie de características generales de los sistemas jerárquicos. Veremos que estas características son directamente relacionables con el *modo de operar humano* y que tienen una clara aplicación en la construcción de sistemas de control inteligente.

Entenderemos por sistema *jerárquico de control* un sistema en el que existe una disposición vertical de subsistemas, con prioridad de actuación o intervención de los niveles superiores y dependencia por parte de los niveles superiores del comportamiento de los inferiores.

5.3.1. Tipos de jerarquías

Los sistemas jerárquicos están compuestos por una serie de niveles que reciben diferentes denominaciones dependiendo del tipo de criterio que ha guiado la descomposición. Este criterio de descomposición determina los tipos de jerarquías que se pueden llevar a cabo. Para Mesarovic los tipos básicos de jerarquías son tres: basadas en estratos, capas y escalones. Estos tipos de jerarquías tienen las siguientes características:

Estratos : Constituidos por *diferentes niveles de abstracción*. Un ejemplo de este tipo de descomposición jerárquica lo constituye un sistema de ayuda al diseño de circuitos electrónicos para computadores.

En este tipo de sistema la representación del conocimiento sobre el circuito se tiene en diferentes niveles de abstracción. En un primer nivel se tiene el circuito físico; un substrato de silicio con una serie de capas depuestas para constituir un circuito. En un segundo nivel tenemos el circuito eléctrico, en el que aparecen

como elemento fundamental de construcción los transistores; en un siguiente nivel tendremos el circuito lógico, en el que el sistema aparece como una serie de puertas lógicas interconectadas; así podríamos continuar por el nivel de circuitos integrados (biestables, memorias, multiplexores, etc), subsistemas (memoria, control de periféricos, procesadores, etc), hasta llegar al nivel de sistema (computador) o aún por encima de éste (multiprocesadores, redes, etc). Como vemos todos los estratos tratan con el mismo producto: el sistema computador.

En la figura 5.3.1 se pueden observar los tres niveles de abstracción comentados (los circuitos representados no se corresponden para no complicar excesivamente las figuras). En cada uno de ellos se tiene un modelo del mismo objeto, un determinado circuito, pero el nivel de abstracción es diferente, debiendo utilizar para razonar sobre su funcionamiento una serie de aspectos muy diferentes.

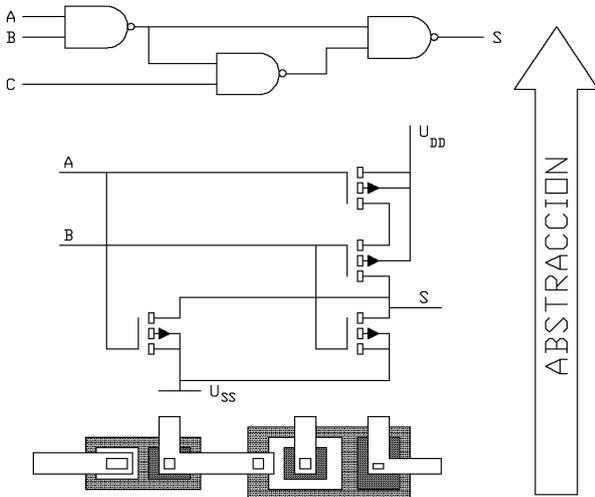


Figura 5.1: Nivel físico, nivel de circuito y nivel lógico.

a la creación de sistemas basados en una jerarquía de niveles de decisión. Niveles que se estructuran en base al principio de velocidad decreciente con comprensión creciente, que es análogo al principio expuesto por Saridis [Saridis 88a] de precisión creciente con inteligencia decreciente. La descomposición propuesta por Saridis es un ejemplo paradigmático de descomposición en capas (ver 5.3.1).

Escalones : Correspondientes a la manera en que interactúan un conjunto de sistemas de toma de decisiones con dependencias de prioridad entre ellos. En este tipo de sistemas son de especial importancia los aspectos de cooperación y coordinación. En la se presenta un ejemplo de estructura de este tipo, correspondiente a lo que podría ser un sistema de control de un robot.

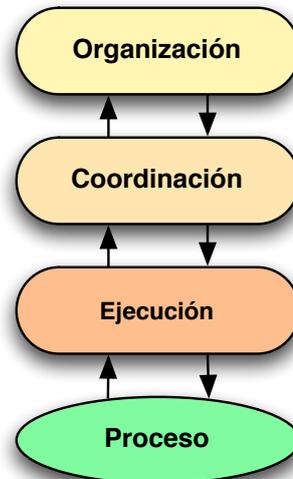


Figura 5.2: Jerarquía de control de Saridis [Saridis 88].

Capas : Constituidas por niveles de complejidad de decisión. La toma de decisiones se ve afectada por dos factores fundamentales: (i) cuando llega el momento de tomar la decisión esta no puede posponerse y (ii) para tomar la decisión mejor es preciso comprender mejor la situación. Esta problemática dual conduce

Estas jerarquías propuestas por Mesarovic son utilizables directamente al campo de control inteligente. Nosotros trataremos de centrar un poco esta aplicación analizando los criterios que guiarán la generación de estructuras jerárquicas de razonamiento.

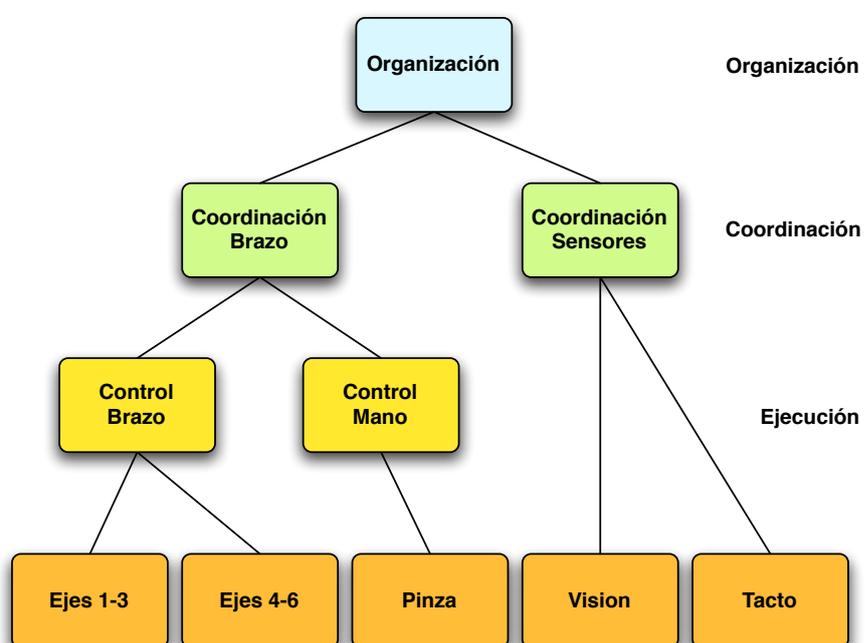


Figura 5.3: Descomposición en escalones de un sistema de control de robot usando la jerarquía de control de Saridis.

5.3.2. Jerarquías en sistemas de control inteligente

Los sistemas de control inteligente tienden a ser sistemas jerárquicos, por las ventajas inherentes a este tipo de estructuras:

- Facilidad de integración con sistemas ya existentes. Es habitual que la instalación de un sistema de control inteligente deba hacerse por encima de un sistema de control preexistente y que no se desea eliminar. La estructura jerárquica permite que la integración del sistema nuevo con el antiguo se haga sin excesivo perjuicio para ninguno de ellos.
- La información del proceso se tiene a diferentes niveles de abstracción, lo que conduce a la implantación de mecanismos de manejo de estas informaciones con una estructura jerárquica natural.
- Del mismo modo las políticas de control se establecen en términos diferentes, que responden a una estructura de generalización progresiva, por lo que es preciso implantar los mecanismos de generación de acciones de la misma forma.
- Las arquitecturas jerárquicas permiten limitar el tamaño de los módulos, lo que es adecuado tanto desde el punto de vista de operación como de desarrollo. En operación, la limitación de tamaño permite utilizar mas adecuadamente los recursos informáticos. En desarrollo, la descomposición modular facilita el reparto de trabajo y disminuye la posibilidad de errores, facilitando las tareas de depuración y mantenimiento [Sanz 89a].
- La modularización redundante en la flexibilidad del sistema —facilidad para el mantenimiento— y en la fiabilidad del mismo.

Todos estos factores hacen que en el caso de sistemas de control inteligente la descomposición sea necesaria. Pero es fundamental el seguir un criterio claro que permita desarrollar una arquitectura adecuada. El criterio que se propone en esta tesis se basa en a) la descomposición del sistema según

cinco facetas que se estructuran en tres criterios con direcciones cuasi ortogonales y b) la integración de los módulos en una estructura jerárquica basada en escalones. Estas facetas de descomposición son las que aparecen en la. El criterio de desarrollo por descomposición es un criterio polifacético de *divide y vencerás*.

Las facetas de descomposición se agrupan en tres criterios que reciben las denominaciones de DOMINIO, FUNCION y REPRESENTACION-PARADIGMA-TIEMPO. El agrupar los tres últimos aspectos en un sólo criterio se debe a que están estrechamente relacionados por cuestiones de implantación, como se verá a continuación.

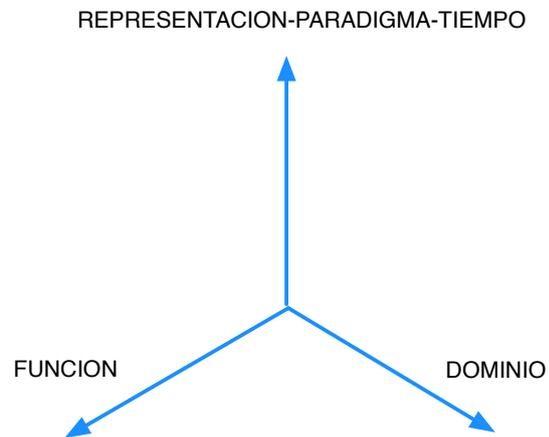


Figura 5.4: Triple criterio de descomposición.

5.3.3. Descomposición por dominios

El criterio de descomposición por dominios nos permite distinguir grupos de funciones y datos relativos a un dominio determinado y prácticamente independientes de otros grupos. Este tipo de descomposición por dominio ha sido el empleado tradicionalmente en sistemas de control distribuido. El sistema se reparte por *áreas de proceso*, esto es, por subprocesos del proceso total. En la Figura 5.3.3 se presenta una descomposición de este tipo aplicada a un horno de clinker. En ella cualquiera de los niveles inferiores presenta una descomposición por dominios. Por ejemplo el segun-

Facetas de descomposición	
Faceta	Significado
<i>DOMINIO</i>	<i>Conjunto de objetos del mundo real a los que aplica el subsistema.</i>
<i>FUNCION</i>	<i>Tipo de tarea a realizar por el subsistema (puede considerarse un dominio abstracto).</i>
<i>REPRESENTACION</i>	<i>Tipo y estructura de la información empleada por el subsistema.</i>
<i>TIEMPO</i>	<i>Velocidad de operación del subsistema, de importancia crucial para el control en tiempo real de un determinado proceso.</i>
<i>PARADIGMA</i>	<i>Tecnología empleada en la construcción del subsistema.</i>

Tabla 5.1: Facetas primarias a emplear en la descomposición modular de sistemas inteligentes de control.

do nivel está dividido en tres subsistemas: control de torre, control de tubo y control de enfriador.

Los sistemas con metaniveles presentan una descomposición de este tipo en la capa de resolución de problemas. La capa de selección —metaconocimiento— determina el dominio en que se sitúa el problema y activa la base de conocimiento correspondiente a dicho dominio.

La descomposición por dominios es el criterio más claro y eficaz, ya que estas descomposiciones, cuando se realizan de forma adecuada, generan elementos con acoplamientos débiles, lo que facilita la coordinación para operación conjunta.

5.3.4. Descomposición funcional

En un sistema de control inteligente las tareas a realizar son muy diversas —proceso de señales, diagnóstico en línea, razonamiento, interfase con humanos, etc.—, por lo que parece adecuado seguir un criterio funcional para descomponer nuestro sistema según especialidades. Dentro de la descomposición aparecerán dos tipos de funciones:

Funciones orientadas a proceso : Son aquellas que tienen que ver con las tareas que tienen relación directa con el proceso a controlar. Estas tareas son propiamente las tareas de un sistema de control, y se suelen estructurar linealmente, esto es, responden a las etapas progresivas de resolución de un problema. Por ejemplo en el caso de un sistema de control basado en reglas borrosas se distinguen tres etapas —funciones— características: difuminación de valores medidos, obtención del valor borroso de la actuación y conversión de este valor a un valor nítido.

Funciones orientadas a sistema : Son las funciones de soporte del sistema de control. Estas funciones tienen como objetivo conseguir el funcionamiento adecuado de los sistemas que implantan las funciones orientadas a proceso. El ejemplo más claro de este tipo de funciones son las de coordinación de diferentes módulos, por ejemplo en el caso de un sistema para el control de un horno de cemento, la coordinación entre los módulos de control del inter-

cambiador, de control del horno propiamente dicho y el de control del enfriador (Ver Figura ??).

Este criterio nos permitirá separar elementos que deben realizar funciones diferentes. Por ejemplo un grupo de tales funciones son las funciones atribuibles a una interfase inteligente de proceso: filtrado, detección de anomalías, estimación de valores ocultos, etc. Estas funciones constituyen un dominio funcional abstracto.

En muchos casos la descomposición de funciones es una descomposición basada en niveles de decisión, en la que se integran diferentes elementos de procesado de información que realizan toma de decisiones. Esta es una situación clásica en sistemas jerárquicos de control, como es el caso de dos bucles de control anidados.

En el apartado anterior se mencionaron los sistemas con metanivel como ejemplo de descomposición por dominios. En este tipo de sistemas se tiene también por lo general una descomposición funcional, seleccionándose de este modo la base más adecuada al dominio-problema. Las descomposiciones funcionales son en cierto sentido descomposiciones por dominios abstractos.

5.3.5. Descomposición representacional

El tercer criterio de descomposición se basa en el nivel de abstracción de la información a utilizar. La información que es de utilidad en control de un sistema complejo no es exclusivamente información numérica —valores medidos de las señales— sino que abarca todo un espectro de niveles de abstracción. La capacidad de manejar *coherentemente* representaciones en distintos niveles de abstracción y razonar con ellas, es —a juicio del autor— una de las características fundamentales de la inteligencia.

Los primeros sistemas inteligentes manejaban representaciones del mundo a un nivel simbólico, que se alejaba —por abstracto— de lo convencional en computación. El comportamiento de estos sistemas era a primera vista inteligente, decían y hacían cosas que sólo los humanos inteligentes

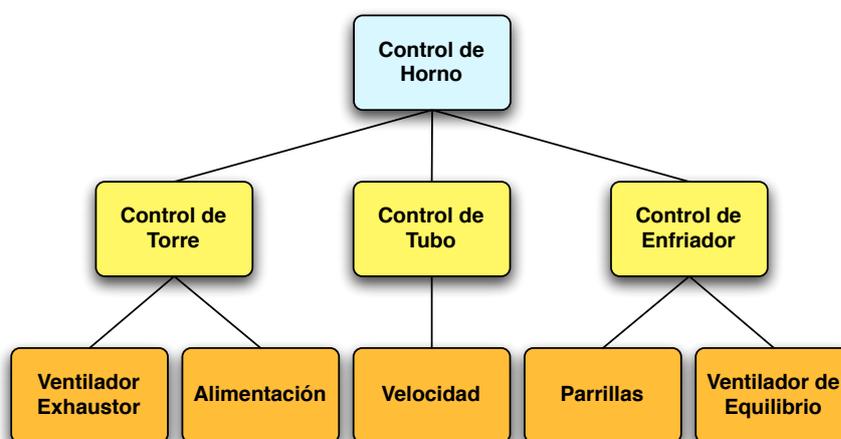


Figura 5.5: El sistema de control de una fábrica de cemento se descompone en tres grandes subsistemas en relación con las unidades mayores del proceso: control de la torre de intercambio, control del horno y control del enfriador.

eran capaces de decir o hacer. Con el tiempo, y con una valoración adecuada de estos sistemas, sus limitaciones han aflorado. La mayor parte de ellos aparecen ahora como sistemas de cómputo sin ninguna característica especial.

Cuando observamos con detenimiento un sistema experto vemos que sus principales limitaciones se dan en su campo de aplicación. Esta limitación es el origen del conocido efecto acantilado: la degradación de prestaciones del sistema experto cuando un problema cae fuera de su campo de aplicación. Este efecto acantilado se produce en dos dimensiones:

Dominio : Es la más conocida, refiriéndose a que un sistema experto sólo puede resolver problemas de su dominio de aplicación. Si se le plantea un problema que cae ligeramente fuera de este dominio las prestaciones del sistema experto disminuyen enormemente. Un sistema experto en diagnóstico de infecciones de la sangre es incapaz de diagnosticar infecciones de linfa.

Abstracción : Para que un sistema experto pueda resolver un problema hay que plantearlo en sus propios términos, en un determinado nivel de abstracción. Al fin y al cabo un sis-

tema experto no es más que un programa informático, y por ello peca de falta de flexibilidad en la introducción de las informaciones de partida. Los tan socorridos menús son uno de los mejores mecanismos para inflexibilizar la entrada de datos, lo que evita cometer demasiados errores en el planteamiento del problema al sistema experto.

A los sistemas inteligentes les falta *inteligencia para razonar en distintos niveles de abstracción*. Es importante pues dotar a los sistemas de mecanismos que permitan este razonamiento multiresolución. La creación de sistemas jerárquicos de razonamiento, que trabajen en distintos niveles de abstracción y que pueden intercambiar información entre ellos parece un camino adecuado para dotar a los sistemas inteligentes de mayor inteligencia.

Este problema de degradación de prestaciones es muy grave en el caso de sistemas de control en tiempo real. La resolución del mismo admite dos enfoques:

Limitativo : restringiendo la zona de aplicación al dominio, mediante mecanismos de detección de problemas planteados fuera del dominio.

En el caso de sistemas de control supone detectar las situaciones en que este control es inaplicable y conmutar a otro modo de funcionamiento, posiblemente pasando la responsabilidad del control a otro sistema, o en último término a un operador humano.

Extensivo : Completando la base de conocimiento del sistema para que no se den estas situaciones de *fuera de límites*. Para ello es preciso utilizar mecanismos de construcción de la base de conocimientos que permitan la eliminación de vacíos de la misma, verbigracia el uso de modelos profundos.

5.3.6. Descomposición temporal y por paradigmas

Los sistemas de procesamiento de información se implantan siguiendo paradigmas establecidos. La selección del paradigma a emplear se hace en base a una serie de criterios entre los que destacan la disponibilidad, la facilidad de implantación y las prestaciones. Cuando hay que realizar un sistema que implante una determinada función el número de paradigmas utilizable es limitado, debiendo realizar una elección entre ellos. Esto es una actividad de diseño del sistema.

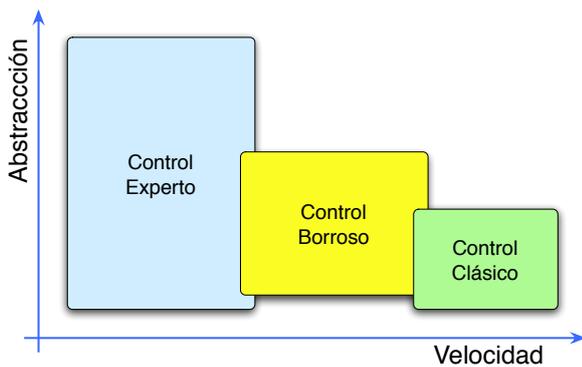


Figura 5.6: Correlación paradigma-abstracción-tiempo.

Cada paradigma de implantación tiene asociados por lo general un nivel de abstracción y un factor temporal. En el caso de los sistemas de inteligencia artificial, la asociación paradigma-nivel de abstracción no es rígida, pero sí suele serlo la

asociación paradigma-factor temporal. No obstante los avances en hardware y software en este campo hacen que estas diferencias de *tempo* entre paradigmas sean progresivamente reducidas. Esto nos lleva a enunciar un principio que caracteriza las implantaciones de sistemas jerárquicos inteligentes:

A medida que aumenta la inteligencia del sistema —asimilable al nivel de abstracción en que opera— se reduce su velocidad.

Este resultado implantacional —*inteligencia creciente con velocidad decreciente*— se deriva del ya conocido principio teórico formulado por Saridis de *precisión creciente con inteligencia decreciente*, que afirma la correlación entre inteligencia y abstracción. Desde el punto de vista del control en tiempo real es muy importante saber si un sistema será capaz de responder dentro de los requisitos temporales que le son impuestos por el sistema físico que controla. En el caso de sistemas de control, o más generalmente, de sistemas de toma de decisiones, el factor temporal es el más crítico. No se puede emplear un sistema lento para controlar un proceso rápido, dado que el control debe realizarse *en tiempo real*. Esto limita la aplicabilidad de determinados paradigmas para determinadas funciones, pero sólo desde el punto de vista temporal, no de las prestaciones teóricas de dichos sistemas.

La descomposición de un sistema puede ser homogénea o heterogénea, dependiendo de si los diferentes sistemas que realizan funciones similares —por ejemplo control— estén o no construidos utilizando los mismos paradigmas de control. Lo habitual es que un sistema de control jerárquico sea heterogéneo, pero es posible pensar en sistemas que no lo son. Por ejemplo consideremos el siguiente proceso:

Sea un horno continuo al que se suministra un material compuesto por la mezcla de tres componentes. En el horno, a una determinada temperatura y con un determinado tiempo de permanencia se produce una reacción entre los componentes para dar como resultado un producto. Este producto es analizado para determinar su composición química y utilizar esta información para el ajuste de las proporciones. Las variables sobre las que se puede actuar son: las cantidades de cada material, la velocidad de avance en el horno y la temperatura del mismo.

En este proceso vemos claramente diferenciadas dos actividades de control: 1) el control de la preparación y 2) el control de la reacción.

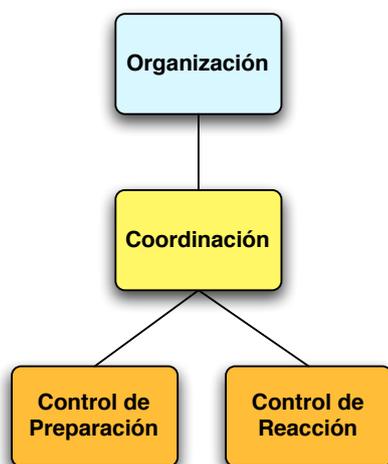


Figura 5.7: Ejemplo de sistema de control jerárquico elemental.

Para cada una de ellas es posible diseñar un regulador utilizando cualquier paradigma de control: PID, borroso, reglas, etc. Pero para que el funcionamiento sea correcto estos dos sistemas de control no pueden ser independientes, ya que la actuación de cada uno de ellos repercute en la del otro. Es preciso establecer un nivel de coordinación entre ambos sistemas de control (ver). Del mismo modo que para los controles de los subprocesos es posible emplear cualquiera de los paradigmas mencionados. La selección entre uno y otro se hará en función de las características de las implantaciones de estos sistemas y de los requisitos del proceso: velocidad, estabilidad, uniformidad, etc.

5.3.7. Estructuración jerárquica

Los criterios anteriores nos permiten guiar la descomposición de nuestro sistema de control inteligente. De este modo llegamos a varios aspectos característicos y fundamentales en un sistema de control inteligente que se presentan en la Tabla 5.3.7.

Los módulos resultantes de la descomposición se integrarán según una arquitectura jerárquica¹, con reparto horizontal por dominios y reparto vertical por funciones, este reparto vertical dará lugar a una serie de escalones en el sentido de Mesarovic. Esta organización no supone una elección específica de paradigma o nivel de abstracción para cada uno de los módulos situados en un nivel determinado, sino cada uno de ellos puede utilizar el paradigma más conveniente de acuerdo con *sus propios requisitos de operación*. A su vez cada uno de estos módulos puede sufrir ulteriores descomposiciones.

Los diferentes niveles de abstracción no tienen por que estar ligados a niveles específicos de una jerarquía de sistemas, pudiendo utilizarse niveles de abstracción bajos en niveles de decisión elevados y viceversa. Sin embargo esto no es lo habitual y el nivel de decisión y el nivel de abstracción presentan una fuerte correlación positiva (mayor nivel-mayor abstracción). Lo mismo sucede con los criterios temporal y paradigmático. Cada paradigma lleva asociado por lo general un nivel de abstracción determinado, aunque la flexibilidad de las técnicas de inteligencia artificial radica en una reducción de la fuerza de esta asociación. Igualmente cada paradigma tiene un tempo característico, que puede variar de una implantación a otra, pero que viene determinado por el tipo de algoritmo y estructura de datos que emplee. Esto hace que la correlación abstracción-tiempo-paradigma sea muy elevada, que es lo que conduce a la correlación fuerte con el nivel decisional. La toma de decisiones en los sistemas de interacción directa con el proceso debe ser necesariamente rápida, por lo que determinados paradigmas no pueden ser empleados a causa de su lentitud. Esto obliga a que los únicos niveles que pueden aprovechar los paradigmas de implantación ligados a abstracciones fuertes sean los niveles con menores requisitos temporales, más lentos, situados por lo general en niveles decisionales elevados.

Las arquitecturas clásicas de control distribuido son arquitecturas jerárquicas, distribuidas por

¹La descomposición funcional no conduce necesariamente a sistemas jerárquicos, ya que este es un aspecto estructural que se introduce con posterioridad. Sin embargo en esta tesis nos referiremos por lo general a estructuras jerárquicas, y no a estructuras cooperativas que podría ser la otra alternativa.

Características de un sistema complejo	
Faceta	Denominación
DOMINIO	Distribuido por dominios. Localizado.
FUNCION	Distribuido por funciones. Especializado.
REPRESENTACION	Multiresolucional.
TIEMPO	Multitemporal.
PARADIGMA	Multiparadigma.

Tabla 5.2: Denominaciones características empleadas para los sistemas complejos descompuestos en función de las facetas que han guiado dicha descomposición.

dominios y funciones. Por ello se tratará de aclarar el uso del término distribuido en esta tesis. Cuando nos refiramos a un sistema como distribuido queremos decir que el sistema está constituido por una serie de elementos *autónomos* pero que interaccionan entre sí. Para abreviar el uso de los términos *distribuido por funciones* y *distribuido por dominios* se emplearán como sinónimos de estos *separado* y *disperso*. Así diremos que un sistema distribuido convencional es jerárquico, separado y disperso y que un sistema de control inteligente es, además, multiresolucional.

La estructuración jerárquica de los sistemas responde a unos criterios de operatividad guiados por las circunstancias mencionadas para los criterios de descomposición. Esto hace que los sistemas jerárquicos de control presenten las siguientes características generales (salvo excepciones):

- Un módulo situado en un nivel superior está involucrado en las actividades relacionadas con un dominio más amplio que uno situado a nivel inferior.
- El tiempo característico de una unidad en un nivel superior es mayor que en un nivel inferior.

- Derivado de los dos anteriores, los módulos de los niveles superiores están involucrados con los aspectos más lentos del comportamiento global del sistema.
- Las representaciones en los niveles superiores suelen ser menos estructuradas, con más incertidumbre y en un mayor nivel de abstracción.

La descomposición por dominios es la de más fácil comprensión y realización. La integración de diferentes módulos de esta descomposición se realizará mediante un nivel funcional superior de coordinación entre ellos. La descomposición funcional es en esencia similar a un análisis funcional clásico. La descomposición representacional, queda un poco difuminada entre los diferentes módulos, ya que estas descomposiciones son ortogonales y el situar un módulo en un determinado nivel jerárquico no supone necesariamente que deba utilizar un determinado tipo de representación.

Las arquitecturas cooperativas que existen en la realidad son en cierta medida arquitecturas jerárquicas, con descomposición funcional y con poca profundidad en su estructura de niveles. No obstante puede ser interesante considerarlas como

arquitecturas heterárquicas, en las que la coordinación no se da gracias a un módulo con esta función expresa, sino que se alcanza por una funcionalidad distribuida entre todos los módulos.

Denominaremos *control jerárquico* a todo sistema de control descompuesto, con funcionalidad distribuida a lo largo de una serie de niveles con relaciones jerárquicas entre ellos. Cada uno de estos niveles es, por sus características, un sistema de control. Diremos que un sistema de control es *jerárquico inteligente* cuando alguno de los niveles de control esté implantado utilizando técnicas de inteligencia artificial. Los sistemas de control jerárquico inteligente que deban operar en tiempo real deben ser —casi por necesidad— heterogéneos. Ello es debido a que los sistemas basados en reglas no tienen la suficiente rapidez de operación y, lo que es más importante, sufren de impredecibilidad de tiempo de ejecución.

En sistemas heterogéneos es de fundamental importancia los aspectos de interfase entre los diferentes niveles representacionales. Los módulos *inteligentes* de la arquitectura se basarán por lo general en representaciones más abstractas que los basados en teoría clásica de control, pero en último término todos operan sobre el mismo proceso, por lo que deben manejar datos coherentes.

5.4. Operación en tiempo real

La operación en tiempo real es susceptible de definiciones múltiples, pero la característica fundamental en todas ellas es la caracterización en base al comportamiento temporal de un sistema informático en relación con un sistema físico.

Un sistema informático, que interactúa con un sistema físico, tiene como misión la realización de cierto proceso de información en relación con el sistema físico en cuestión. Cuando el correcto funcionamiento del sistema informático-físico depende del adecuado acoplamiento —flujos de datos correctos tanto en sus *valores* como en su *temporalidad*— diremos que el sistema tiene requisitos de operación en tiempo real. Si el sistema informático cumple estos requisitos diremos que dicho sistema es un sistema en tiempo real. Vemos pues que la

operación en tiempo real de un sistema informático se define *siempre* en base a un sistema físico con el que interactúa.

Los sistemas de control son sistemas en tiempo real *per se*. No es concebible un sistema de control que no funcione en tiempo real. Nadie desea un sistema de control que le diga:

Hace una hora había que haber subido el combustible un diez por ciento.

No obstante, en los sistemas de control existe toda una gama de requisitos de operación. Un sistema de control de una máquina herramienta tiene que trabajar evidentemente a mayor velocidad que un sistema de control de un proceso de purificación de agua. Las posibilidades de control no vienen dadas por el método de control, sino fundamentalmente por el sistema de soporte de dicho método. Un PLC permite implantar algoritmos de control que pueden operar a gran velocidad. Un computador de proceso tiene una velocidad menor, y si pensamos en un sistema basado en el conocimiento, la velocidad es aún menor.

Analizaremos a continuación las prestaciones de algunos paradigmas de control inteligente desde el punto de vista de su operación en tiempo real. Este análisis permitirá establecer su aplicabilidad en sistemas de control inteligente.

5.4.1. Requisitos fuertes

Los requisitos fuertes de operación en tiempo real se refieren a que el correcto funcionamiento del sistema en tiempo real, requiere de la realización de los procesos de información *dentro de las limitaciones temporales impuestas*. En un sistema de este tipo no es permisible una demora, aunque sea esporádica. El comportamiento del sistema debe ser *siempre* dentro de límites.

Ejemplo de este tipo de sistemas son algunos de los sistemas de aviónica avanzada. No es permisible que las actuaciones de este tipo de sistemas sufran demoras impredecibles, dado que la seguridad del aparato y la del piloto dependen de la correcta operación —tanto en valores como en tiempo— del sistema informático. Lo mismo sucede con sistemas de cuidado intensivo de pacientes

como con sistemas de gestión de emergencias en plantas nucleares.

5.4.2. Requisitos débiles

Los requisitos débiles de operación en tiempo real se refieren a que el correcto funcionamiento del sistema en tiempo real no requiere que la operación del sistema sea siempre entre límites. Es admisible por lo tanto que se produzcan retardos esporádicos. En estos sistemas —por ejemplo sistemas de control de plantas industriales— el no satisfacer los requisitos temporales se traduce en una disminución de la calidad de operación, con la consiguiente pérdida de rendimiento.

En algunos de estos sistemas, no obstante, pueden aparecer circunstancias que presenten requisitos de operación fuertes, como puede ser el caso de manejo de emergencias.

5.4.3. Sistemas basados en reglas borrosas

Los sistemas basados en reglas borrosas son, con mucho, los que ofrecen mejores características de operación en tiempo real. Con estos sistemas es posible determinar a priori el tiempo necesario para la determinación de la acción de control, ya que su estructura monocapa hace que no haya encadenamiento posible, por lo que el tiempo de cálculo es fijo. Este tipo de reguladores está muy estudiado, pudiendo encontrarse referencias adecuadas sobre ellos [Pedrycz 89].

Los desarrollos de controladores reales basados en reglas borrosas están a la orden del día, alcanzando una gran difusión. De hecho existen en el mercado paquetes de control basados en este tipo de reguladores o incluso procesadores específicos para hacer inferencias borrosas [Togai 89].

5.4.4. Sistemas basados en reglas

El estado actual del arte en entornos de ingeniería del conocimiento para desarrollo de aplicaciones con requisitos de tiempo real y en particu-

lar sistemas expertos muestra la precariedad de los sistemas de este tipo.

Como vimos anteriormente, estos sistemas ofrecen alternativas interesantes al control clásico, pero la necesaria operación en tiempo real hace preciso matizar las posibilidades de uso de los mismos. Es preciso pues analizar sus características desde algunos puntos de vista:

- ¿Existen sistemas de desarrollo de aplicaciones de control basadas en reglas?
- ¿Es posible su utilización en tiempo real?
- ¿Hasta que punto estos sistemas satisfacen requisitos débiles?
- ¿Hasta que punto estos sistemas satisfacen requisitos fuertes?
- ¿Existe algún desarrollo real?

Técnicas de desarrollo actuales

Las técnicas de desarrollo de sistemas de control basados en reglas son bastante heterodoxas; aunque suelen basarse en sistemas de desarrollo, cuando precisan de prestaciones elevadas deben ser construidos utilizando otro tipo de técnicas, por lo general lenguajes convencionales.

Un sistema de desarrollo es una herramienta software que permite el desarrollo, verificación e implantación de sistemas de control basados en reglas. Actualmente pocas herramientas permiten realizar los tres pasos, careciendo prácticamente todas de útiles para realizar la verificación.

En la actualidad el desarrollo de sistemas de control basados en reglas se puede agrupar en tres tipos:

1. Desarrollos basados en herramientas convencionales de inteligencia artificial.
2. Desarrollos basados en herramientas específicas para SCBRs.
3. Desarrollos independientes.

Características de operación en tiempo real

Los desarrollos basados en herramientas clásicas carecen por completo de características de operación en tiempo real, por lo que son absolutamente inadecuados como SCBRs. Este tipo de sistemas suelen utilizarse como ayuda a operadores o como elementos de diagnóstico.

Los sistemas construidos con herramientas específicas presentan mejores características; como por ejemplo, una mayor velocidad de ejecución, sacrificando aspectos como la flexibilidad del lenguaje de representación del conocimiento; una mejor capacidad de razonamiento temporal; y la capacidad de interrumpibilidad.

Los desarrollos en éste área han conducido a productos avanzados, que están alcanzando gran difusión debido al auge de los sistemas expertos y de la informatización en general, y a las carencias manifestadas por los sistemas convencionales.

Cuando los requisitos de operación en tiempo real son importantes, bien requisitos fuertes, bien requisitos débiles pero con tiempos pequeños, ni siquiera las herramientas específicas dan las suficientes prestaciones. En estos casos es preciso recurrir a desarrollos específicos. Este tipo de desarrollos se dan fundamentalmente en el campo de la defensa y, de un tiempo a esta parte, en las grandes industrias.

Satisfacción de requisitos temporales

Los requisitos de operación en tiempo real son fundamentalmente temporales, dados por un retardo máximo admisible en el tiempo de cómputo. Como hemos visto anteriormente, decir que un sistema satisface requisitos fuertes de operación en tiempo real quiere decir que dicho sistema genera la salida —señal de control— antes de este tiempo especificado a priori. Los requisitos débiles, por contra, permiten que este tiempo se supere esporádicamente.

Así pues el garantizar que un sistema cumple requisitos de tiempo real supone decir, en el caso de que los requisitos sean fuertes, que el SCBR realiza la determinación de la acción de control en un tiempo menor que el especificado; y en el caso

de que los requisitos sean débiles, que el SCBR realiza *por lo general* la determinación de la acción de control en un tiempo menor que el especificado.

El garantizar estos tiempos es muy difícil en SCBRs. Salvo que los tiempos límite especificados sean muy altos no existe en la actualidad ningún entorno que permita el desarrollo de sistemas capaces de satisfacer requisitos fuertes. El problema fundamental no es la satisfacción de los requisitos, sino el garantizar dicha satisfacción, ya que los mecanismos de verificación de sistemas basados en reglas son muy limitados y no permiten analizar la duración temporal de las inferencias.

La satisfacción de requisitos débiles es más viable, ya que la verificación por simulación permite garantizar la satisfacción de dichos requisitos.

Desarrollos reales

Los desarrollos reales de sistemas de control inteligente han sido hasta hace poco mas bien escasos, reduciéndose prácticamente a aplicaciones desarrolladas en el ámbito universitario y con propósitos de investigación y/o demostración.

En el ámbito industrial las aplicaciones de sistemas de control basados en reglas han sido prácticamente inexistentes por encima de un cierto nivel de complejidad. Prácticamente los únicos desarrollos comprobados —no experimentales— eran:

- Los llevados a cabo en el área de supervisión de procesos. Implantados mediante herramientas específicas de desarrollo.
- Los SCBRs que emplean lógica borrosa y bases de conocimiento monocapa, utilizados con éxito en gran cantidad de situaciones; de ellos existen incluso productos comerciales tanto en software como en hardware.
- Aplicaciones de SCBRs en control directo restringidas a sistemas muy simples.

Però desde la introducción de determinadas herramientas específicas —como es el caso de

G2— la aplicación de estos sistemas en el mundo industrial es cada vez más patente. Esta falta de introducción hasta hace poco cabe achacarla a la dificultad de garantizar un funcionamiento temporalmente correcto de los SCBRs.

5.4.5. Sistemas basados en modelos

La operatividad de estos sistemas en tiempo real es limitada, en el mismo sentido que lo es la de los sistemas basados en reglas. Sin embargo el uso de mecanismos de orientación a objeto para la construcción de los modelos hace que el número de reglas necesario sea muy reducido, de forma que las ejecuciones del motor de inferencia sean más rápidas.

5.5. Descomposición de Sistemas basados en el Conocimiento

El objetivo de este apartado es presentar una metodología original de desarrollo de arquitecturas para SCBRs que permita una mejora de prestaciones de este tipo de sistemas para operación en tiempo real. Esta mejora de prestaciones se logrará principalmente mediante una reducción de la base de conocimientos activa en cada pasada del motor de inferencia. De este modo se reduce el costo computacional asociado al casamiento. El tiempo empleado en el casamiento es un porcentaje muy elevado del tiempo total consumido en cada pasada del motor de inferencia; en sistemas con elevado número de reglas este porcentaje puede aumentar aún más, sobre todo cuando se generan conjuntos conflicto reducidos, como es habitual en SCBRs. Este tiempo es proporcional al número de reglas de la base de conocimientos. La mejora de velocidad en la etapa de casamiento que se propone en este documento se logrará mediante de una disminución del número de reglas activas.

La mejora de velocidad en SCBRs propuesta en esta tesis pasa a través de dos puntos cruciales:

- La descomposición de la base de conocimientos, de forma que en cada pasada del motor de

inferencia la base de conocimientos *activa* sea tan sólo una parte de la base de conocimientos total.

- Un mecanismo de planificación de las bases de conocimiento, de forma que se alcancen las conclusiones finales en tiempo mínimo.

Ambos aspectos son fundamentales para la reducción de la base de conocimiento activa y la mejora temporal, de forma que el motor de inferencia pueda establecer a priori una estimación de la duración de la inferencia.

5.5.1. Descomposición de la base de conocimientos

El descomponer la base de conocimientos supone dividir el conjunto de reglas que la componen en grupos de reglas relacionadas; esta relación puede ser:

Vinculación al dominio : un grupo de reglas se dirán vinculadas por dominio, cuando son aplicables a un conjunto de hechos determinado. Es clásico que en la premisas de las reglas aparezcan cláusulas relativas a la tarea a realizar. En sistemas más avanzados es posible la invocación de grupos de reglas —base de conocimientos descompuesta— asociados a determinados entes del dominio. Este es el caso de la acción *focus* de G2 [Gensym 89]. Esto es una descomposición por dominio.

Grado de razonamiento : Cuando se utiliza el término metaconocimiento nos estamos refiriendo a conocimiento sobre el conocimiento. Este conocimiento se emplea habitualmente para controlar el proceso de inferencia. Diremos que el conocimiento sobre el dominio es conocimiento de grado 0. Al metaconocimiento diremos que es grado 1 y así sucesivamente. Es importante separar el conocimiento por grados, ya que de esta forma se clarifican sobremanera las bases de conocimiento. Esta descomposición es una descomposición funcional orientada a sistema. Véase por ejemplo la siguiente regla de XCON [Barker 89]:

IF *The current step of the process of changing the characteristics involves resetting after a change was made*

and *The characteristic to be changed is still marked as such*

THEN *reset the characteristic so that it is no longer identified for change.*

En ella vemos que se está manejando conocimiento de grado 1. Es por ello importante separarla de las restantes reglas referentes al problema de configuración presentado a XCON.

Tipo de problema : Esto es una descomposición funcional orientada a dominio. Una base de conocimientos para la realización de una función determinada solo es preciso activarla cuando se requiera la utilización de dicha función. Un ejemplo es una base de conocimiento para identificación del estado funcional del proceso —valoración de la situación.

Mediante descomposición la base de conocimientos queda dividida en un conjunto de bases de conocimiento, entre las que posiblemente existirán solapes, tanto en dominio como en nivel de abstracción. La existencia de estos solapes indica que el problema o sistema a manejar es único. Si existe una base de conocimiento que no se solapa con ninguna otra quiere decir que el dominio y el nivel de abstracción son diferentes de los de las demás bases de conocimientos, esto es, maneja un problema completamente diferente.

Denominaremos *base de conocimientos activa* al conjunto de bases de conocimiento que se van a aplicar en una pasada particular del motor de inferencia.

La descomposición del sistema de control ofrece múltiples ventajas :

- La disminución de la base de conocimientos activa
- La desaparición de las cláusulas de estado en las premisas de las reglas.
- La posibilidad de realizar desarrollos independientes.

La descomposición de la base de conocimiento puede realizarse de formas múltiples, conduciendo a todo un rango de posibilidades. Desde las *descomposiciones gruesas* —pocas bases con muchas reglas— a las *descomposiciones finas* —muchas bases con pocas reglas— existen múltiples descomposiciones viables. Las descomposiciones finas garantizan la mejor adecuación de la base al problema, minimizando el número total de reglas. Las descomposiciones gruesas, por contra, minimizan el número de bases y con ello los retardos derivados de la planificación.

Descomposición homogénea

Diremos que la descomposición de la base de conocimientos es homogénea cuando todas las bases de conocimiento son del mismo tipo, esto es implantan conocimientos utilizando un determinado paradigma de ingeniería del conocimiento. De este modo tendremos una serie de bases de conocimiento aplicables a nuestro dominio que nos permiten realizar diferentes tipos de análisis y grados de profundización en la resolución del problema.

Diremos que dos bases de conocimiento se solapan en dominio cuando existe un subdominio del problema para el que hay conocimiento expresado en cada una de ellas.

La descomposición se dirá progresiva cuando existen bases de conocimiento que manejan conocimiento con distintos niveles de abstracción. Un sistema con descomposición progresiva tiene una serie de pares de bases de conocimiento $P_i = (B_j, B_k)$ ($i=1, n$); la aplicación de la base de conocimiento B_j dará un resultado R_j , la aplicación de la base de conocimiento B_k dará un resultado R_k , que será más profundo —menos abstracto— que R_j .

Las bases de conocimiento se reparten según tres ejes: función, dominio y abstracción.

La explicación anterior es quizá un poco caótica. Trataremos de aclararla con un ejemplo. Un sistema con metaniveles es un sistema con descomposición progresiva y con solape de bases de conocimiento de distintos niveles de abstracción. En un sistema con metaniveles se tiene una base de cono-

cimiento de alto nivel, que es la encargada de seleccionar bases de conocimiento del nivel inferior. Estas bases de conocimiento de nivel inferior tienen su aplicabilidad en dominios restringidos del problema. La descomposición se realiza para minimizar el número de reglas a utilizar en cada inferencia. La selección de las bases de conocimiento de bajo nivel se realiza en base a un análisis de los hechos en un alto nivel de abstracción. La base de conocimiento de alto nivel se solapa pues en dominio con las bases de bajo nivel, ya que el dominio al que se aplica la base de conocimiento de bajo nivel es un subdominio del dominio total.

Descomposición heterogénea

Diremos que la descomposición de la base de conocimientos es heterogénea cuando las bases de conocimiento son de tipos diferentes. Los paradigmas principales que se pueden emplear en la construcción de estas bases son a) reglas, b) reglas borrosas y c) modelos.

5.5.2. Planificación de bases de conocimiento

La planificación de las bases se debe realizar en base a conocimiento de metacontrol. Este metaconocimiento es de dos tipos:

De relación entre bases : Metaconocimiento de planificación.

De relación entre bases y dominio : Metaconocimiento de adecuación.

La selección de las bases debe realizarse de acuerdo con los siguientes objetivos:

- Minimización del tiempo total de inferencia, que se traduce en:
 - Minimización del número de bases.
 - Selección de las bases de tiempo mínimo.
- Optimización de los resultados a través de:
 - Profundización máxima.
 - Adecuación al problema, que es, con mucho, más difícil.

La expresión del metaconocimiento de planificación se puede realizar mediante reglas, enfoque seguido en sistemas con metaniveles actuales, o bien mediante procedimientos algorítmicos convencionales. Los métodos de planificación pueden ser por lo tanto de dos tipos:

Algorítmicos : basados en criterios deterministas y con técnicas clásicas de planificación.

Mediante reglas : será una serie de bases de conocimiento de grados superiores al 0, las que realizarán la planificación de las restantes bases. Este es el enfoque empleado en arquitecturas con metanivel. El nivel superior —de grado 1— se encarga de realizar la planificación, que por ser de una sola etapa es simplemente una selección de la base de conocimiento adecuada al problema.

El planificador genera un plan de razonamiento que indica, por etapas, las bases de conocimiento a aplicar en cada una de ellas. La especificación del plan de utilización de las bases de conocimiento puede realizarse de dos formas: con y sin fusión de bases. Cuando se hace sin fusión de bases, en cada etapa se activan una serie de bases de conocimiento determinadas. Si se realiza con fusión de bases, en cada etapa se indican las bases de conocimiento a activar y las bases de conocimiento a desactivar. Cuando las que se desactivan son todas se tiene el enfoque sin fusión de bases.

5.5.3. Criterios de selección de bases

Los criterios de selección de bases y de generación del plan de activación de las mismas son variados, refiriéndose tanto a la minimización temporal como a la adecuación al problema. Estos criterios son de grados superiores al 0 estando asociados a aspectos como son:

Adecuación : Las bases se seleccionarán en función de su adecuación al problema a resolver. Deben ser bases cuyo dominio comprenda el dominio del problema planteado. Es un criterio unitario.

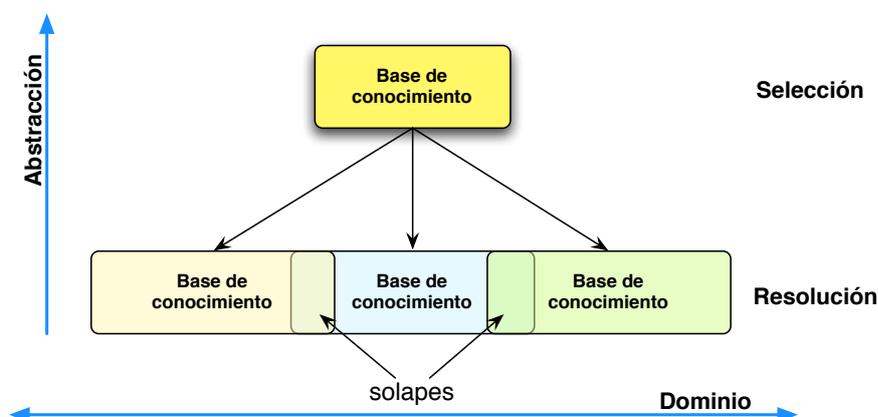


Figura 5.8: Ejemplo de sistema de control basado en el conocimiento con un metanivel de selección.

Minimización temporal : Los factores temporales son también criterios unitarios e indican el retardo estimado debido al uso de la base de conocimiento. Este factor se modificará en función de las ejecuciones del sistema y los tiempos conseguidos.

Adaptación : La adaptación de dos bases de conocimiento A y B en niveles de abstracción diferentes se medirá mediante factores de adaptación $A_f(A,B)$. Estos factores de adaptación de las bases se corregirán en función del tiempo de inferencia cuando se han aplicado ambas bases.

Dependencia : Diremos que una base es dependiente de otra menos abstracta cuando se requiera haber aplicado esta para poder utilizarla. La dependencia es un criterio fuerte de la planificación de bases.

lizando una serie de bases de conocimiento que conduzcan a la conclusión. Este planificador puede ser de dos tipos:

Generador de plan completo . Una vez planteado el problema al planificador, éste generará un plan completo de aplicación de bases de conocimiento. Este plan consta de una serie de etapas, en las que en cada momento está disponible la solución obtenida por la última etapa aplicada. En cada etapa se aplicarán unas determinadas bases de conocimiento.

Generador por etapas . El planificador, dada la situación actual selecciona el conjunto de bases de conocimiento a aplicar para obtener la siguiente conclusión.

5.5.4. Razonamiento progresivo

Diremos que un razonamiento es progresivo cuando las conclusiones se obtienen por aplicación sucesiva de bases de conocimiento hasta que se solicita la conclusión. No se conoce, pues, a priori el horizonte temporal de la inferencia.

El planificador organiza la aplicación de las bases de forma que en cada momento se esté uti-

En estos sistemas la dicotomía velocidad-calidad se resuelve en base a las características del problema. Si el sistema puede requerir soluciones con gran variabilidad temporal es más interesante la descomposición y la planificación fina. Si la variabilidad es menor es posible utilizar descomposiciones y planificaciones gruesas.

5.5.5. Razonamiento conclusivo

Diremos que un razonamiento es conclusivo cuando se conoce a priori el horizonte temporal de la inferencia.

En este caso el planificador debe generar un plan que maximice la calidad de la solución dentro de este tiempo admisible de inferencia.

5.6. Valoración de la Metodología

Se han analizado las características de los sistemas de control basados en reglas desde el punto de vista de su aplicabilidad a problemas de control con requisitos de operación en tiempo real débiles y fuertes.

Se ha presentado una metodología de desarrollo de arquitecturas para este tipo de sistemas que permite mejorar las prestaciones de los mismos. Esta arquitectura se basa en:

- Descomposición de la base de conocimientos, construyendo una base de conocimientos múltiple.
- La separación del conocimiento del dominio y el conocimiento de control.
- La utilización de mecanismos de planificación de bases de conocimiento basado en el conocimiento de control.

Esta arquitectura permite:

- La implantación de sistemas con razonamiento conclusivo y progresivo.
- Un aumento en la verificabilidad de los sistemas.
- Un aumento de la fiabilidad de los sistemas.

Capítulo 6

Modelado Multirresolucional

... the best way to overcome these limitations, particularly the issues of temporal and spatial reasoning, truth maintenance, and the “cliff” performance degradation effect, is to build symbolic models of the expert domain inside the expert system

*Lawrence E Widman and Kenneth A. Loparo
Artificial Intelligence, Simulation and Modeling*

... according to Zadeh, one of the most important facets of human thinking is the ability to summarize information ‘into labels of fuzzy sets which bear an approximate relation to the primary data’

*Didier Dubois and Henri Prade
Fuzzy Sets and Systems, Theory and Applications*

6.1. Introduction

En el capítulo 3 se han comentado algunas características de los modelos empleados en los sistemas de razonamiento simbólico. La arquitectura que se propone en esta tesis está orientada al control de procesos complejos descansando en gran medida sobre un modelo profundo del proceso a controlar.

Hasta ahora el control de este tipo de procesos se afronta exclusivamente desde tres perspectivas: control manual, control con reguladores clásicos supervisados y control experto. El control manual garantiza casi con seguridad un funcionamiento aceptable si la complejidad primaria no es excesiva; pero, por sus especiales características, su eliminación es un objetivo fundamental. Los otros

dos tipos de controles presentan por el contrario rangos de aplicabilidad muy limitados. El control clásico se restringe casi por completo a procesos simples, mientras que el control experto presenta problemas de desarrollo —adquisición del conocimiento— de funcionamiento —operación en tiempo real— y de mantenimiento —falta de fiabilidad. La alternativa por la que se opta prioritariamente en la actualidad es utilizar arquitecturas basadas en control clásico con un bucle de supervisión manual.

¿Por qué este tipo de arquitectura logra resultados que no logran las otras? El responder este interrogante es un factor importante en la resolución del problema. Esta arquitectura logra resultados aceptables porque *el control clásico le da operatividad en tiempo real, mientras que el bucle humano le proporciona mecanismos de resolución de problemas para hacer frente a las situaciones anormales del control*. En base a ello se intentan utilizar arquitecturas en que el bucle supervisor se implanta por medio de un sistema experto o en algunos casos un sistema basado en conocimiento de otro tipo; pero estas soluciones no logran los resultados del control supervisor humano, porque los mecanismos de razonamiento sobre el proceso no son lo suficientemente profundos y completos; profundos en el sentido de poder determinar causas y efectos a lo largo de toda una jerarquía de sistemas que componen el proceso y completos en el sentido de lo suficientemente extensos como para abarcar y poder razonar sobre todas las circunstancias que le acontecen al proceso y son relevantes para su control.

Los problemas fundamentales del uso de técnicas de inteligencia artificial en control son dos: la operación en tiempo real, que se solventa con el

uso de sistemas jerárquicos y el razonamiento profundo sobre el proceso. Este capítulo trata de hacer frente a este segundo problema —del mismo modo que el anterior hacía frente al primero— por medio de un esquema de modelado multiresolucional de procesos complejos. Este esquema permitirá modelar en profundidad pero sin dejar de lado los aspectos más abstractos de la información sobre un proceso. El esquema que se propone es análogo al empleado para los módulos del sistema de control jerárquico: un esquema descompuesto multiresolucional.

En cierto sentido éste es el capítulo más importante de la tesis, ya que en él se presenta la técnica de representación de la información sobre el proceso a controlar que se va a emplear en el sistema de control inteligente. La reutilizabilidad del software de control inteligente depende en gran medida de la capacidad de construir modelos profundos del proceso a controlar, ya que de esta forma se logra separar de forma clara el código general de la información específica de la aplicación. El modelo del proceso que se presenta es un modelo multiresolucional capaz de suministrar información a todos los subsistemas del controlador inteligente. En un primer nivel —modelo cuantitativo o numérico— incluirá información en forma de ecuaciones diferenciales ordinarias sobre aquellos elementos modelables de esta forma. Un segundo nivel estará constituido por un modelo cualitativo, en el que se utilizarán ecuaciones diferenciales cualitativas. Un tercer nivel contendrá la información de mayor nivel de abstracción, constituyendo una base de conocimiento clásica.

Mientras que los modelos basados en ecuaciones diferenciales y las técnicas de representación de conocimiento han sido objeto de gran cantidad de estudio, no sucede lo mismo con las técnicas de modelado cualitativo. Es por ello que se hace especial hincapié en dicha técnica en este capítulo. En particular se analizan en profundidad los mecanismos de abstracción que permiten pasar de información numérica a información cualitativa y se presenta un método de simulación la simulación cualitativa borrosa orientado al manejo de la explosión combinatoria asociada a la simulación cualitativa.

6.2. Modelos de Sistemas Continuos y Multirresolucionalidad

El conocimiento disponible sobre un determinado objeto, en particular un proceso, no se sitúa siempre en el mismo nivel de abstracción. Podemos conocer la temperatura en un determinado punto con un nivel de resolución de milésimas de grado y en otro saber simplemente que es *alta*. La utilización simultánea de estos dos tipos de información es compleja e imposible en muchos sistemas de proceso de información.

Los procesos complejos presentan características de multiresolucionalidad en la información disponible y utilizable en el control de los mismos, por lo general debido a la complejidad terciaria que se manifiesta en una incertidumbre en las medidas y en una incertidumbre en la interpretación de las mismas. Por ello el integrar en nuestros esquemas de representación ambos tipos de información es fundamental. Ya Mesarovic [Mesarovic 70] comenta esta necesidad de diferentes niveles de abstracción:

“Truly complex systems almost, by definition, evade complete and detailed descriptions. The dilemma is basically one between the simplicity in description, one of the prerequisites for understanding, and the need to take into account a complex systems’s numerous behavioral aspects. A resolution of this dilemma is sought in a hierarchical description. One describes the system by a family of models each concerned with the behavior of the system as viewed from a different level of abstraction. For each level, there is a set of relevant features and variables, laws and principles in terms of which the system behavior is described.”¹

Para realizar un análisis mas profundo de este

¹“Los sistemas verdaderamente complejos prácticamente eluden, por definición, cualquier tipo de descripción completa y detallada. El dilema se sitúa entre la simplicidad de la descripción, uno de los prerequisites para la comprensión, y la necesidad de tener en cuenta los numerosos aspectos del comportamiento de un sistema complejo. Una solución a este dilema se vislumbra en una descripción jerárquica. El sistema es descrito por medio de una familia de modelos referidos, cada uno de ellos, al comportamiento del sistema en un determinado nivel de abstracción. En cada nivel existen una serie de características y variables, leyes y principios, en base a los cuales se describe el comportamiento del sistema”.

aspecto de los sistemas complejos es preciso caracterizar cuales son los elementos de modelado que van a tener este tipo de características.

6.2.1. Conceptos de modelado de sistemas continuos

Los elementos fundamentales para el modelado de sistemas continuos son dos: las variables y las relaciones entre variables. Una variable es una función del tiempo. En el caso de sistemas físicos las variables son funciones reales que pueden tomar un conjunto continuo de valores salvo efectos cuánticos. Sin embargo los dispositivos de medida son de precisión limitada, por lo que las variables podrían considerarse —desde el punto de vista de proceso de información— como funciones sobre un conjunto discreto.

La evolución de un sistema físico queda plenamente caracterizada por la evolución de un conjunto de variables de dicho sistema denominadas variables de estado. Estas evoluciones están regidas por las relaciones entre dichas variables, que especifican la manera en que las variables resultan modificadas a lo largo del tiempo. Dicho en términos matemáticos las relaciones determinan los valores de las variables y de sus derivadas temporales. El esquema más conocido de representación de relaciones de un sistema son las ecuaciones diferenciales. Un conjunto de ecuaciones diferenciales suponen una serie de restricciones para los valores de las variables y de sus derivadas, que se deben satisfacer simultáneamente a lo largo de la vida del sistema físico.

Una forma de modelar un sistema físico es establecer un conjunto de variables del sistema y un conjunto de ecuaciones diferenciales que rijan el comportamiento temporal de éste. Este tipo de modelo es lo que en el capítulo 3 denominábamos un modelo matemático. Sin embargo no todos los sistemas físicos admiten el establecimiento en estos términos de un modelo aceptable. Por lo general esto es debido a la existencia de incertidumbre, bien en las variables, bien en las relaciones.

La incertidumbre en las variables puede deberse a que seamos incapaces de determinar cuales son las variables del sistema o bien en que seamos

incapaces de determinar los valores que toman. En el proceso de modelado de todo sistema físico aparecen este tipo de problemas de incertidumbre, y la manera de hacerles frente es el establecimiento de umbrales de significancia. En todo sistema físico las variables que intervienen son infinitas², por lo que para modelar es preciso seleccionar el conjunto de variables relevante para el tipo de resultados que esperamos obtener de nuestro modelo. Del mismo modo es preciso establecer el conjunto de valores que pueden tomar las variables, a estos conjuntos los denominaremos *espacios de cantidades*, por motivos que se verán a posteriori. Las matemáticas son bastante buenas para manejar espacios de cantidades densos por ejemplo los números reales: \mathbb{R} sin embargo no son tan buenas en manejar infinitas variables o relaciones. Por ello la técnica habitual de modelado de sistemas continuos es seleccionar un número finito —y preferiblemente pequeño— de variables, establecer como espacio de cantidades \mathbb{R} y expresar las relaciones por medio de ecuaciones diferenciales.

Sin embargo existen casos en que esto no es posible, porque no existe el conocimiento suficiente como para formalizar las relaciones. Además en muchos casos la continuidad de valores no es necesaria y puede ser un engorro. Los modelos mentales —utilizados por las personas— no se construyen en base a relaciones matemáticas, sino que se expresan mediante algo que podríamos denominar *influencias*. Este tipo de relación es similar a la que se emplea en la física cualitativa [Hayes 85a] y que ha derivado en una serie de desarrollos importantes en el campo de la simulación cualitativa de sistemas.

6.3. Modelado Multirresolucional

La potencialidad del ser humano su inteligencia se debe en gran medida en su capacidad de hacer frente a los problemas de formas diferentes. El *utilizar otro enfoque* es a menudo la clave para resolver un problema. Por contra las máquinas programadas para resolver un problema lo hacen siem-

²Suponiendo que el universo sea infinito en algún sentido, ya que los sistemas cerrados no existen en la realidad.

pre de la misma forma. Si el problema no responde con exactitud al tipo de problema para el que se diseñó la máquina, ésta produce malos resultados. Este es el caso habitual con los sistemas software. Cada programa está preparado para resolver determinados tipos de problemas y no otros.

La inteligencia que en ocasiones se aprecia en algunas aplicaciones de la *inteligencia artificial* es achacable a la flexibilidad en el planteamiento de los problemas que dichas aplicaciones resuelven. Evidentemente el planteamiento de un problema de diagnóstico a un sistema experto es —en cierta medida— más flexible que un problema planteado a un programa de nóminas. Esta diferencia se aprecia directamente en la diferente consideración de ambos procesos: el sistema experto *resuelve un problema*, mientras que el programa de nóminas realiza un proceso *mecánico*.

Esta impresión de *inteligencia*, tanto en el humano como en la máquina, procede de la flexibilidad con que admite el planteamiento del problema; pero esta capacidad de admitir diferentes planteamientos está fundamentada en la capacidad interna de manejar modelos flexibles del mundo, modelos capaces de plantearse un problema desde diferentes puntos de vista y no exclusivamente desde uno como era el caso del programa de nóminas.

Esta flexibilidad se ve radicalmente incrementada cuando el sistema inteligente es capaz de manejar modelos multiresolucionales. Diremos que un sistema inteligente dispone de un modelo multiresolucional cuando puede tener representaciones de un problema en diferentes niveles de abstracción. En el caso de razonamiento sobre sistemas físicos, la multiresolucionalidad es fundamental.

6.3.1. Necesidad de la multiresolucionalidad

Ante los anteriores comentarios sobre el modelado multiresolucional surge claramente una pregunta:

¿Por qué?

Debemos aclarar cuáles son los motivos que nos llevan a plantear la necesidad de este tipo de modelo. El modelo es necesario para realizar predicciones de comportamiento del sistema, predicciones que nos permitan determinar si la respuesta del mismo será adecuada frente a una acción de control. El tipo de procesos que tratamos de manejar presentan grandes dificultades de modelado, por lo que el modelado basado en ecuaciones diferenciales ordinarias modelo N puede presentar desviaciones importantes respecto al comportamiento real del sistema. Estas desviaciones se deben por lo general a la dificultad de estimar parámetros del proceso por los excesivos ruidos o variabilidad del mismo. El modelo cualitativo trata de solventar en parte éste problema, realizando simultáneamente una función triple:

- Puede predecir todos los comportamientos posibles, lo que nos permite determinar si alguno de ellos es problemático. En este sentido se puede aplicar antes o después del modelo numérico, para validar a posteriori la predicción de éste o para hacerla innecesaria si las predicciones cualitativas son todas aceptables a priori.
- Sirve como base de información para la comunicación con el usuario, ya que los valores manejados en el mismo tienen un fundamento lingüístico.
- Sirve como base de hechos para realizar razonamientos basados en reglas de forma mas clara que los valores numéricos.

La necesidad del modelo basado en técnicas de representación del conocimiento surge de los dos últimos puntos comentados para el modelo cualitativo. Aunque este modelo puede llegar a tener capacidad predictiva su utilización en este sentido parece innecesaria. Sin embargo su función de base de hechos es fundamental, ya que permite eliminar del proceso de inferencia etapas de abstracción para llegar a una serie de conceptos de alto nivel empleados en el control del proceso.

La utilización de modelos multiresolucionales en control debe contemplarse desde la perspectiva de las dos etapas fundamentales del control inteligente (ver Figura ??): valoración de la situación

y determinación de la actuación. El formular un método de determinación de la actuación supone el establecer las acciones a realizar para cada situación de proceso, por ello la etapa de valoración de la situación genera una descripción del estado del proceso. Esta descripción del estado del proceso se dá por necesidad en un determinado nivel de abstracción. El modelo que se presenta en esta tesis permite implantar la etapa de valoración de la situación por medio de modelos de alto nivel de abstracción y funciones de abstracción que permitirán pasar de los datos numéricos a la formulación abstracta sobre el estado del sistema.

6.3.2. Objetivos del modelo

Los objetivos del modelo responden en gran medida a los puntos mencionados en el apartado sobre su necesidad. Resumiendo estos puntos podemos hablar de las tres funciones principales del modelo:

- Función **informativa**, proveyendo a todos los sistemas que lo precisen de la información que deseen sobre el estado del proceso en el *nivel de abstracción adecuado*.
- Función **predictiva**, suministrando información sobre el comportamiento o comportamientos posibles del proceso frente a acciones de control.
- Función **lingüística**, sirviendo de base de vocabulario para la comunicación con el usuario.

El satisfacer estos tres objetivos es una tarea ardua, sobre todo en lo referente a la función predictiva.

6.3.3. Estructura del modelo

Como ya se ha comentado anteriormente, en la arquitectura de control que se desarrolla en esta tesis, se propone el uso de un modelo con tres niveles de resolución. Este modelo multiresolucional comprende tres capas de modelado de diferentes características. Estas capas de modelado son las siguientes:

- *Modelo numérico*, al que nos referiremos en lo que sigue como modelo N.
- *Modelo cualitativo*, en adelante modelo Q.
- *Modelo basado en técnicas de representación del conocimiento*, en adelante modelo K.

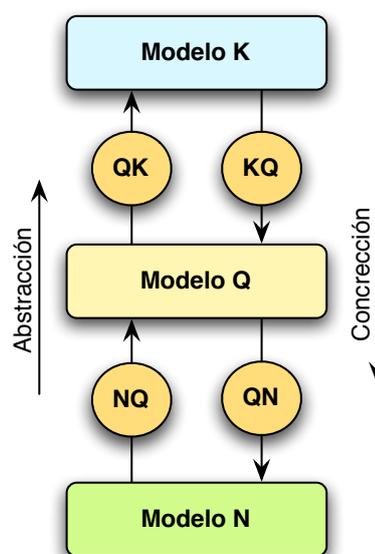


Figura 6.1: Estructura y transformaciones en el modelo multiresolucional NQK.

Estos tres niveles de modelado no son independientes, sino que están estrechamente acoplados. El enlace entre estos tres modelos se establece en base a funciones de abstracción y funciones de concreción (ver Figura 6.3.3). Estas funciones nos permiten establecer valores de variables en un determinado nivel a partir de valores de variables en otros niveles. Las funciones de abstracción permitirán pasar de un modelo al inmediatamente superior; las de concreción permitirán realizar la operación inversa.

Los modelos N y K responden a paradigmas informáticos plenamente establecidos simulación e ingeniería del conocimiento por lo que en la presentación que se haga de los mismos no se hará un estudio en profundidad. Sin embargo, el

uso de modelos cualitativos en aplicaciones industriales es bastante nuevo, existiendo un número limitado de desarrollos [Zhang 90a] [Charbonnaud 90] [Guerrin 90] orientados fundamentalmente al diagnóstico. El uso de modelos multiresolucionales está ganando una aceptación progresiva, existiendo también algún desarrollo de carácter industrial [Juuso 90].

Los modelos N y Q están estrechamente relacionados, ya que la composición del modelo Q es parte de la del modelo N. Existe un isomorfismo entre parte del modelo N y el modelo Q. El modelo K sin embargo es en gran parte independiente de la estructura de los otros dos, ya que en él se representa conocimiento *humano*, siendo pues un modelo *subjetivo*, mientras que los modelos N y Q pueden calificarse en cierta medida de *objetivos*.

6.4. Modelo N

El modelo N es un modelo cuantitativo del proceso a controlar. Este modelo se realiza utilizando técnicas clásicas de simulación de sistemas continuos.

6.4.1. Técnicas básicas

Las técnicas a emplear en la construcción de un modelo basado en ecuaciones diferenciales están perfectamente establecidas, siendo perfectamente factible el conseguir paquetes software comerciales para ello, como pueden ser CSMP, ACSL, SIMNON, etc.

El desarrollo de un esquema o lenguaje de representación de sistemas continuos no es un objetivo de esta tesis, ya que tanto la bibliografía como las implantaciones de sistemas de este tipo son más que abundantes [Fernández 77].

6.4.2. Funciones del modelo

La función de este modelo es dual:

- Por una parte informativa, ya que es la base de datos sobre el estado del proceso y la

evolución del mismo. Como veremos en el capítulo 7 esta información es utilizada para realizar el control del proceso en varios de los niveles de control.

- Por otra predictiva, ya que debe generar predicciones de comportamiento del proceso. Estas predicciones serán predicciones numéricas, en forma de evolución temporal de las variables del proceso. Cada predicción de comportamiento se da pues como función temporal del espacio de variables del proceso en R^n .

La función predictiva del modelo N es mucho más importante que la función informativa, dado que la información sobre el estado N es suministrable por otros elementos de la arquitectura, como se verá en el capítulo 7.

A primera vista el modelo N entra en franca contradicción con los planteamientos iniciales de esta tesis: *control de procesos complejos, con dificultades de modelado*. Obviamente cuando un proceso tiene complejidad terciaria el desarrollo de un modelo veraz del mismo es prácticamente imposible, pero si el proceso tiene sólo complejidades primaria y secundaria el desarrollo del modelo es posible. En este tipo de procesos el desarrollo de sistemas de control convencionales es problemático, por lo que la arquitectura propuesta en esta tesis puede dar resultados adecuados.

6.4.3. Implantación

La implantación de este modelo se realizará por medio de técnicas convencionales de simulación de sistemas continuos. Para su integración en la arquitectura de control inteligente será preciso sin embargo unas ciertas capacidades software del sistema empleado en el desarrollo de este modelo.

En el capítulo 9, dentro del apartado de desarrollos futuros se comentará la necesidad de desarrollar herramientas de simulación convencional integradas en entornos de control inteligente.

6.5. Modelo Q

El modelo cualitativo de l proceso permitirá representar información disponible sobre el proceso en unos términos cercanos al modo en que los humanos representan el conocimiento sobre los sistemas físicos. Este tipo de conocimiento representa los aspectos cualitativos del comportamiento del sistema bajo estudio, no siendo formalizable con las técnicas empleadas en simulación convencional.

Evidentemente un modelo convencional — ecuaciones diferenciales— es mucho mas preciso que cualquier otro tipo de modelo que podamos formular. Sin embargo, cuando el conocimiento sobre el proceso está afectado de incertidumbre o cuando se desean realizar otras tareas aparte de la simple predicción de comportamiento, es preciso hacer uso de otras técnicas más cercanas a las representaciones de conocimiento empleadas en inteligencia artificial.

De este modo el modelado cualitativo es una herramienta de formalización de conocimiento sobre sistemas físicos, en una forma cercana a la representación matemática del comportamiento del mismo, pero con la flexibilidad suficiente como para no atarse a una representación puramente numérica y permitir formulaciones también cercanas al modo de razonar humano. Este es el sentido de los trabajos desarrollados en el área de *razonamiento cualitativo*, inicialmente propuesta por Hayes en su “Naivè Physics Manifesto” y que en la actualidad está alcanzando una gran difusión, fundamentalmente en el campo de razonamiento para diagnosis.

6.5.1. Funciones del modelo

Las funciones que realiza el modelo cualitativo son todas las mencionadas en 6.3.2, esto es:

- **Informativa:** suministra información sobre el estado *cualitativo* así como sobre las dependencias funcionales cualitativas entre las distintas variables del proceso. La importancia de los valores cualitativos es que permiten su utilización inmediata en procesos de infe-

rencia, mientras que la información estructural/funcional será empleada en razonamiento basado en modelos.

- **Predictiva:** los modelos cualitativos tienen capacidad predictiva, esto es, es posible realizar simulaciones con ellos. Sin embargo estas simulaciones tienen una característica especial: bifurcaciones. La predicción sobre el comportamiento del proceso no es única, sino que está formada por un árbol de estados cualitativos. Los posibles comportamientos del proceso son los posibles caminos desde el estado inicial hasta los estados terminales.
- **Lingüística:** los términos empleados en el modelo cualitativo son términos lingüísticos que son empleados de forma directa en la comunicación con el operador; de este modo existe una estrecha relación entre la representación interna del estado del proceso y la representación externa del mismo.

El modelo cualitativo es una representación *orientada al proceso y cercana a los modos humanos de representar el conocimiento sobre el mismo*.

6.5.2. Técnicas básicas

Las técnicas básicas que se emplean en la construcción del modelo Q y su enlace con el modelo N son:

Modelado cualitativo: caracterizado fundamentalmente por la limitación de dependencias funcionales a funciones monótonas crecientes y decrecientes y por especificar espacios de cantidades finitos y de cardinalidad reducida [Barreto 90] [Bratko 88] [Clancey 89] [Grant 89] [Shrager 87].

Simulación cualitativa: basada en el algoritmo QSIM de Kuipers [Kuipers 86], aunque con matizaciones de otros métodos [Forbus 84] [de Kleer 84a] [Leyval 90].

Extensiones de QSIM: desarrolladas por diferentes autores y orientadas a aumentar la aplicabilidad de la simulación cualitativa. De des-

tacar son la contribución de D'Ambrosio³ [D'Ambrosio 89a] y la de Dalle Molle [Dalle Molle 89], aunque existen otras [Bridgeland 90] [Franke 90] [Widman 89b].

Conjuntos borrosos: empleados en la definición de las funciones de abstracción así como en la construcción de los estados cualitativos borrosos y en una extensión del algoritmo QSIM [Zadeh 77] [Kandel 86] [Dubois 80].

6.5.3. Simulación cualitativa

La simulación cualitativa basada en el algoritmo QSIM [Kuipers 86] es un método de simulación por propagación de restricciones en conjuntos de variables. Estas restricciones son derivadas de un conjunto de ecuaciones diferenciales ordinarias —como las empleadas en el modelo N— siendo idénticas en su mayor parte salvo en la relajación de dependencias funcionales.

En el capítulo 3 se presentó un pequeño ejemplo de una simulación cualitativa mediante QSIM de un objeto en caída libre. En este ejemplo se puede apreciar la forma que adquiere una predicción cualitativa de comportamiento.

La simulación cualitativa se caracteriza por algunos aspectos fundamentales:

- Los valores que pueden tomar las variables constituyen conjuntos finitos y pequeños denominados *espacios de cantidades* [Hayes 85a], que en algunos casos se limitan a conjuntos de tres valores : $-$, 0 y $+$ [de Kleer 84a]. Estos espacios de cantidades tiene estructura de retículo, aunque en el caso particular de QSIM la relación de orden en el conjunto es un orden total.
- Las restricciones empleadas son *aritméticas* (suma, negación, producto) *funcionales* (dependencia monótona creciente y decreciente) y *derivativas* (derivada) (ver Tabla 6.5.4).
- Las predicciones de comportamiento son múltiples. Una simulación no genera una trayectoria del sistema, sino que genera un árbol

de comportamientos en el cual cada camino desde el estado inicial hasta un estado final es un posible comportamiento del sistema (ver Figura 6.5.4). En algunos casos QSIM predice comportamientos imposibles para el sistema real, pero en ningún caso deja de predecir los comportamientos posibles.

- Los modelos cualitativos son válidos en zonas específicas de valores de las variables que se denominan *regiones de operación*. La necesidad de manejar juegos de restricciones adecuados a cada una de las regiones de operación viene originado por la debilidad de las restricciones funcionales M^+ y M^- .

Como dice Dalle Molle [Dalle Molle 89] los primeros desarrollos basados en QSIM fueron desarrollos de modelos y simulaciones de sistemas simples, sin utilizar en ningún caso sistemas reales de cierta complejidad. Dicho autor es el primero en construir este tipo de modelos y en desarrollar extensiones a QSIM para hacer frente a los problemas que presenta este método de simulación.

Los problemas principales de QSIM para simular procesos complejos se derivan de la explosión combinatoria de comportamientos que llega a producir una ramificación inabordable. Una gran parte de dichos comportamientos son predicciones espurias que son imposibles para el sistema real. Gran parte de las extensiones a QSIM tratan de eliminar estos comportamientos espurios.

Las variables empleadas en los modelos cualitativos toman valores en conjuntos denominados espacios de cantidades. Algunos autores utilizan espacios de cantidades muy reducidos, como es el caso de [de Kleer 84a] en que se emplea el espacio de cantidades:

$$Q = \{-, 0, +\}$$

Con estos valores lo único que se conoce de una variable es si ésta es negativa, nula o positiva. Kuipers utiliza espacios de cantidades más complejos y que pueden ser diferentes para cada una de las variables. Estos espacios de cantidades están compuestos por puntos de la recta real —hitos— y por intervalos entre ellos. Así un espacio de cantidades asociado a una temperatura puede ser:

$$Q_{Temp} = \{-, 0, (0, 3100), 3100, (3100, -)\}$$

³Aunque desarrollada sobre la teoría cualitativa de procesos formulada por Forbus [Forbus 84].

Los valores que componen el espacio de cantidades pueden recibir nombres específicos, con lo que podemos decir que las variables toman *valores lingüísticos*. Así un espacio de cantidades como el anterior puede traducirse en:

$$Q_{Temp} = \{NEGATIVA, CERO, NORMAL, CRITICA, EXCESIVA\}$$

con lo que la expresión $Temp = 3100$ es traducible en *La TEMPERATURA es CRITICA*.

6.5.4. Extensiones a la representación

Los espacios de cantidades como el anterior presentan algunas limitaciones para ser integrados en el modelo multiresolucional. El paso de valores de variables de un modelo a valores en los términos utilizados por un modelo superior —tarea llevada a cabo por las funciones de abstracción— es, en cierta medida, un proceso de interpretación de mediciones. Por ello es preciso que esta interpretación se haga de forma adecuada, teniendo en cuenta las incertidumbre inherentes al proceso de medición así como en lo referente al conocimiento teórico sobre el sistema medido.

Este problema es de especial importancia para las funciones NQ. En particular la posibilidad de utilizar hitos puntuales en dichas funciones es remota, ya que en muchos casos el conocimiento que se tiene del proceso no es suficiente como para establecer estos valores teóricos con tal precisión, ni para afirmar la exactitud de las medidas y la igualdad de las mismas con dichos valores teóricos.

Cuando asignamos a una variable de proceso un valor cualitativo estamos haciendo una abstracción de su valor. Esto es inmediato en casos en que el valor de la variable no es cercano a ninguno de los hitos. Sin embargo cuando sucede ésto, el afirmar que el valor de una variable es *CRITICA* conlleva sus riesgos, ya que en muchos casos los valores exactos de los hitos son desconocidos⁴ y los valores de las variables no se miden con absoluta precisión. El afirmar que una variable *de un proceso real* tiene un determinado valor es una afirmación que inevitablemente es incierta.

⁴Salvo casos especiales como por ejemplo *la temperatura de ebullición del agua pura a una atmósfera de presión* que tiene un valor exacto de 100.0 °C.

La correspondencia entre el modelo N y el modelo Q debe realizarse en base a representaciones más adecuadas que los espacios de cantidades propuestos por Kuipers. La representación desarrollada en esta tesis se basa en el uso de conjuntos borrosos para representar los diferentes términos lingüísticos o elementos del espacio de cantidades. En la Figura 6.5.5 se presenta un esquema que aclara este concepto.

En ella los términos a emplear en la representación cualitativa aparecen asociados a los hitos e intervalos en el sentido de Kuipers. Sin embargo cada uno de estos elementos tiene asociada una función de pertenencia que se empleará en el enlace con la representación puramente numérica. De esta forma las separaciones entre términos cualitativos no son nítidas sino que se definen en base a funciones en $[0, 1]$ que presentan un cierto solape. Las funciones de pertenencia son funciones $\mu : R \rightarrow [0, 1]$, mientras que para Kuipers estas funciones se definen sobre $\{0, 1\}$. Esto permite cierta flexibilidad con las aserciones de valores de las variables, ya que no establecen correspondencias rígidas con los valores numéricos de las mismas.

El valor cualitativo de una variable —valor Q— se hace determinando el término cualitativo cuya función de pertenencia dá un valor máximo. En este sentido esta representación cualitativa es idéntica a la basada en intervalos nítidos. Sin embargo en este modelo las variables admiten otro tipo de valor que denominaremos *valor cualitativo borroso* o valor FQ. Este valor es un conjunto borroso sobre el retículo de valores cualitativos nítidos⁵. Los factores de este conjunto borroso sobre el retículo vienen dados por las funciones asociadas a cada uno de los términos borrosos. Así por ejemplo en una determinada situación diremos que la variable *TEMPERATURA* tiene los siguientes valores:

N: 3000

Q: NORMAL

FQ: NORMAL/0.8, CRITICA/0.2

⁵En la literatura anglosajona este tipo de conjunto es denominado *L-Fuzzy set*, esto es, conjunto borroso sobre un retículo —lattice.

RESTRICCIONES CUALITATIVAS			
Restricción	Equivalencia	Debilidad	Tipo
$SUMA(f,g,h)$	$f + g = h$	Igual	Aritmética
$MULT(f,g,h)$	$f \cdot g = h$	Igual	Aritmética
$MENOS(f,g)$	$f = -g$	Igual	Aritmética
$M^+(f,g)$	$g = H(f), H \text{ m.c.}$	Mayor	Funcional
$M^-(f,g)$	$g = H(f), H \text{ m.d.}$	Mayor	Funcional
$DERIV(f,g)$	$df/dt = g$	Igual	Derivativa

m.c.: monótona creciente, m.d.: monótona decreciente

Tabla 6.1: Restricciones cualitativas.

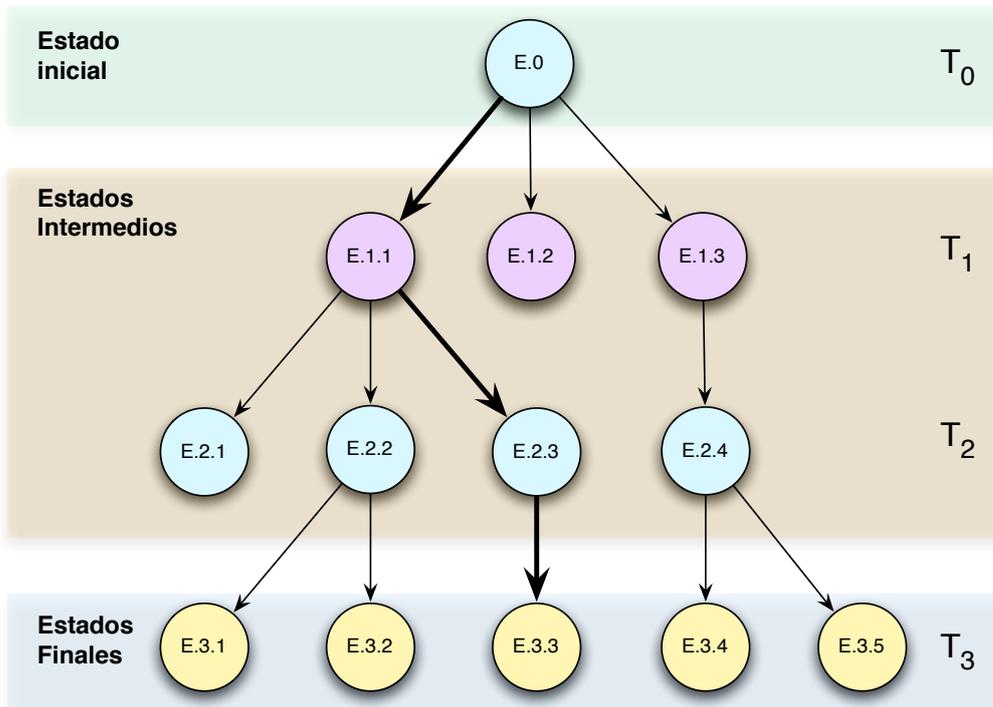


Figura 6.2: Ejemplo de árbol de comportamientos en simulación cualitativa.

Este tipo de valores FQ se emplearán en la extensión al algoritmo de simulación propuesta en esta tesis y denominada simulación cualitativa borrosa.

6.5.5. Funciones NQ y QN

Las funciones que nos permiten pasar de los valores numéricos a los valores cualitativos son funciones de abstracción, ya que pasan de un determinado espacio de cantidades a un espacio de cantidades en un nivel de abstracción superior. Estas funciones de abstracción —funciones NQ— son las funciones asociadas a los términos cualitativos de la representación cualitativa.

El valor cualitativo de una variable es aquel cuya función de pertenencia asociada da un valor máximo, esto es, es el conjunto borroso de los que componen el espacio de cantidades cualitativo que presenta un grado de pertenencia mayor:

$$V_Q(Y) = T^*, \quad T^* \in Q_Y : \\ \mu_{T^*}(y) = \max \mu_T(y) \quad \forall T \in Q_Y$$

El paso de valores en sentido contrario —funciones de concreción— no presenta demasiado interés en este tipo de sistema salvo para dos aplicaciones:

- Utilización en el módulo de control borroso. El método habitual en este caso es utilizar el método del centroide para calcular valores nítidos a partir de los valores borrosos. En este caso el paso de valores Q a N es inmediato: el centro de la función de pertenencia. En caso de valores FQ aparecen dos tipos de situaciones:
 - Situaciones con multiplicidad igual a uno, en las que todos los factores son nulos salvo uno. En este caso, dependiendo de la función asociada al término con factor no nulo será posible determinar un valor N único, doble o con un intervalo. En caso de que sea doble o intervalo se seleccionará el punto medio.
 - Situaciones con multiplicidad mayor de uno. En este caso es posible determinar un valor N único.

- Utilización para establecer valores N de variables —p.e. variables de actuación— a partir de valores Q especificados por el usuario o por sistemas de razonamiento.

6.5.6. Extensiones a la simulación

La simulación cualitativa se ha mostrado como una técnica potente y versátil aunque tiene graves problemas que limitan su aplicabilidad a la simulación de sistemas complejos:

- Por una parte la explosión combinatoria asociada a este tipo de simulación. Este es, con mucho, el problema mas grave, y en el que se ha puesto un mayor esfuerzo en su solución.
- Por otra parte los problemas de indecidibilidad derivados de las influencias múltiples.
- La construcción de los modelos, que llega a ser muy complicada. Además un modelo mal construido puede dar lugar a predicciones de comportamiento espurias.

La solución de este tipo de problemas pasa a través del desarrollo de técnicas avanzadas de simulación cualitativa. Algunas de estas técnicas ya fueron introducidas en los desarrollos originales de Kuipers, como es el reconocimiento de estados iguales y la detección de bucles de comportamiento. En la Figura 6.5.7 se presenta el resultado de detectar que los estados E.2.2 y E.2.4 de la Figura 6.2 son iguales.

Del mismo modo, en la Figura 6.5.7 se presenta el resultado de detectar un estado idéntico a otro estado anterior. En este caso se produce un comportamiento periódico, en virtud de la situación de ambos estados en un mismo comportamiento. En ambas figuras se ha realizado una reubicación de los estados para simplificar las mismas.

Este tipo de mejoras permiten eliminar gran cantidad de estados repetidos en los árboles de predicción de comportamiento, que pasan a ser realmente grafos de comportamiento. No obstante estas mejoras no son suficientes para hacer viable la simulación cualitativa de procesos complejos. Es

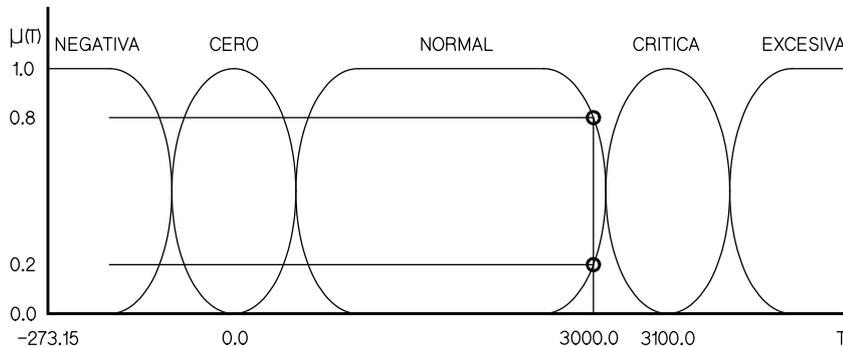


Figura 6.3: Espacios de cantidades borrosos.

preciso recurrir a técnicas adicionales como las desarrolladas por otros autores. Estas técnicas permiten reducir e incluso eliminar en parte los problemas planteados, posibilitando el uso de este tipo de simulación en sistemas más complejos. Algunas de las técnicas empleables son las siguientes:

- Utilización de derivadas de orden superior [Dalle Molle 89].
- Utilización de vectores cualitativos de estado [Morgan 90].
- Utilización de información cuantitativa [Kuipers 90].
- Utilización de técnicas de ponderación de influencias [D'Ambrosio 89a].
- Utilización de modelos componibles [Franke 90].

6.5.7. Simulación cualitativa borrosa

Sin embargo las anteriores extensiones no resuelven un problema adicional que surge cuando se tratan de relacionar representaciones cualitativas —o cualquier otro tipo de representación— con entidades reales. Este problema es el problema de la *interpretación*.

El problema de la interpretación surge de la incertidumbre asociada a las mediciones realizadas

sobre un proceso real así como en la especificación de los valores que requieren especial consideración —hitos— asociados a las variables del proceso. Este problema de interpretación hace que las predicciones cualitativas de comportamiento lleven asociada una cierta incertidumbre, tanto en el punto de partida como en las posibles transiciones.

Para resolver en parte este problema se ha desarrollado en esta tesis un método de ponderación de comportamientos aplicable al algoritmo de Kuipers. Este método recibe el nombre de simulación cualitativa borrosa y se basa en la utilización de los estados cualitativos borrosos mencionados en la sección 6.5.4. Este método es una extensión al algoritmo QSIM que es también de utilidad para hacer frente a la explosión combinatoria por medio de podas selectivas del árbol de comportamientos, y al problema de la indecidibilidad del comportamiento por medio de factores de posibilidad de transición.

Por ello la simulación cualitativa borrosa trata de realizar ponderaciones de comportamientos, de forma que se pueda:

- Estimar cual es el comportamiento más posible⁶.

⁶Se emplea el término *posible* en un sentido genérico, sin tratar de hacer las distinciones que hacen otros autores respecto a otros conceptos asociados al manejo de incertidumbre. En este sentido ver [González 89] o [Klir 89].

- Podar el árbol de comportamientos.

Esta extensión se basa en utilizar la información cuantitativa suministrada por el modelo N así como las funciones de abstracción que generan valores FQ, para reducir el número de predicciones y hacerlas más fiables.

El algoritmo de simulación cualitativa borrosa es el siguiente:

- Se parte de un estado inicial dado por el estado FQ del proceso obtenido mediante las funciones de abstracción del estado N desde el que se quiere hacer partir la simulación.

Por ejemplo consideremos que nuestro proceso tiene dos variables X e Y con los siguientes valores FQ:

$$X = \text{MEDIA}/0.9, \text{ALTA}/0.3$$

$$Y = \text{BAJA}/0.4, \text{MEDIA}/0.6$$

- De dicho estado FQ se obtienen una serie de estados Q que se ponderarán con unos factores de posibilidad. Para cada variable el número de grados de pertenencia no nulos es reducido uno o dos en la mayor parte de los casos. El número de estados Q derivables de dicho estado oscila pues entre 1 y 2^N , siendo N el número de variables del proceso.

En nuestro ejemplo obtenemos los siguientes estados Q:

$$Q_1 = \{X=\text{MEDIA}, Y=\text{BAJA}\}$$

$$Q_2 = \{X=\text{MEDIA}, Y=\text{MEDIA}\}$$

$$Q_3 = \{X=\text{ALTA}, Y=\text{BAJA}\}$$

$$Q_4 = \{X=\text{ALTA}, Y=\text{MEDIA}\}$$

- Para cada estado Q generado del anterior modo se determina el factor de posibilidad de dicho estado empleando una norma triangular [González 89] sobre todos los grados de pertenencia asociados a las diferentes variables.

Los factores de posibilidad asociados a cada uno de los estados Q obtenidos será (se emplea el producto como norma triangular):

$$Q_1 = NT(0.9, 0.4) = 0.36$$

$$Q_2 = NT(0.9, 0.6) = 0.54$$

$$Q_3 = NT(0.3, 0.4) = 0.12$$

$$Q_4 = NT(0.3, 0.6) = 0.18$$

- Se realiza una simulación cualitativa para cada uno de dichos estados Q. De esta forma se genera un grafo de comportamientos a partir de cada uno de los estados Q. Cada uno de estos grafos de comportamiento
- Se realiza una fusión de los grafos de forma que se realiza una ponderación de estados empleando para ello una conorma triangular [González 89] de forma que los estados se ponderan de acuerdo con los diferentes factores asociados a los estados cualitativos que generaron los diferentes comportamientos.

El resultado final de la simulación es un grafo de comportamiento en el que los nodos representan estados y aparecen rotulados con factores numéricos que representan en cierta medida la posibilidad de que el comportamiento real del sistema pase por dicho estado.

Desde el estado inicial al final aparecerá un camino privilegiado (posiblemente varios) que será el comportamiento mas plausible.

Este método de simulación permite:

- Aumentar la fiabilidad de las predicciones, ya que se tienen en cuenta los posibles errores asociados a la determinación del estado inicial.
- Paralelizar el proceso de simulación, ya que es preciso realizar una simulación por cada estado Q de partida.
- Podar por umbrales el grafo de comportamientos, eliminando nodos que tengan factores de posibilidad por debajo del umbral. De este modo se reduce el número de ramas a prolongar en caso de que se deseen realizar nuevas simulaciones.
- Si la fusión de estados se realiza en cada paso del proceso de simulación —cada vez que se generan estados sucesores— se puede realizar una poda en marcha, lo que limita el número de ramas a prolongar aunque hace perder la característica de *predicción total* de la simulación cualitativa.

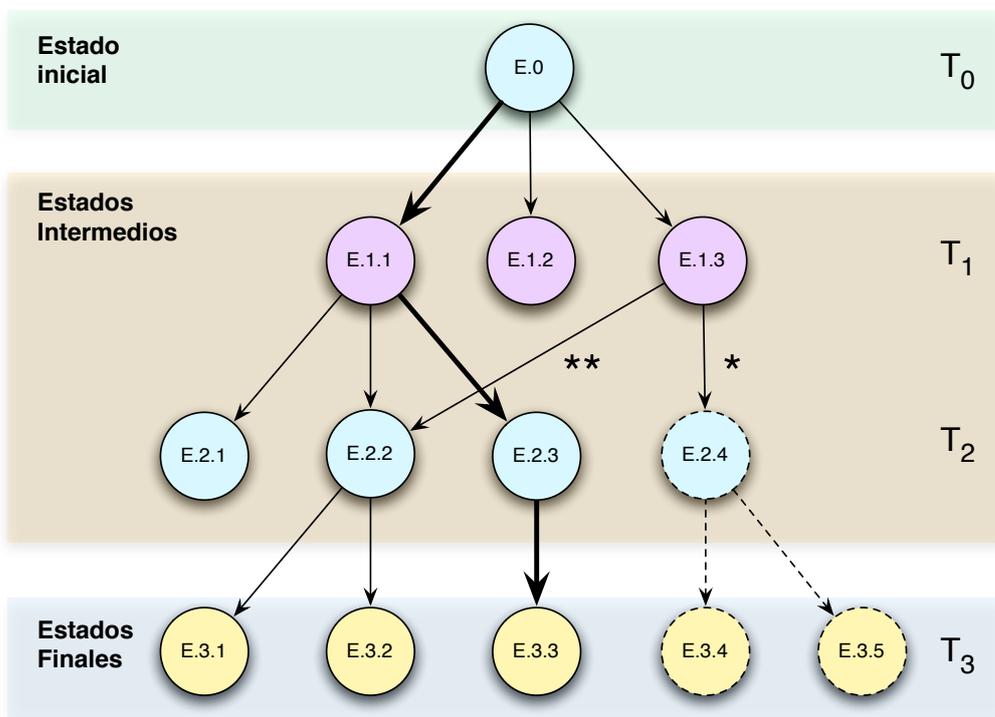


Figura 6.4: Detección de estados iguales. Dado que $E.2.2 \equiv E.2.4$ los subárboles que cuelgan de dichos estados son idénticos —de hecho son los mismos estados.

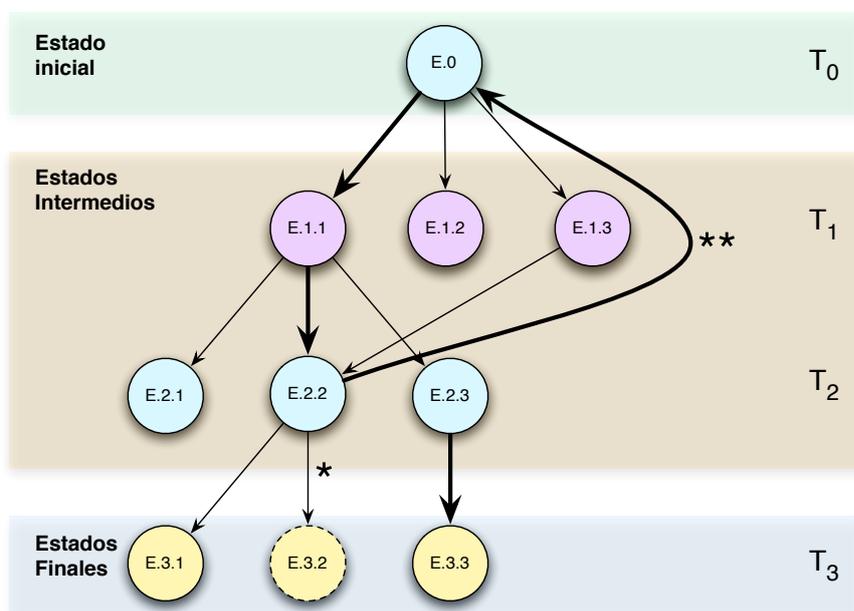


Figura 6.5: Detección de comportamientos periódicos por identificación de transiciones a estados previamente considerados.

El aspecto negativo principal de la simulación cualitativa borrosa es que para hacer frente al problema de la interpretación agrava el problema de la explosión combinatoria. Sin embargo este método permite realizar la poda del árbol de comportamientos, de forma que se acote en cierta medida la explosión, aunque perdiendo la seguridad de que se generará el comportamiento real del sistema.

6.5.8. Implantación

La implantación de este tipo de representación requiere de técnicas de simulación cualitativa para las que no existen sistemas de desarrollo comerciales. El grupo de razonamiento cualitativo de la Universidad de Texas en Austin distribuyen de forma gratuita un entorno de simulación cualitativa que implanta el algoritmo QSIM de Kuipers. Este sistema —denominado Q— está desarrollado en LISP, lo que limita sobremanera su integrabilidad con aplicaciones complejas, en particular aplicaciones de control.

También es posible emplear el sistema Q2 desarrollado por el mismo equipo que introduce la técnica de incorporar información cuantitativa —en forma de intervalos— para podar el árbol y hacer más fácilmente interpretables los comportamientos generados.

Además el problema más grave no es la simulación, sino la construcción del modelo cualitativo. En algunos casos este es directamente derivable del modelo en ecuaciones diferenciales ordinarias, pero en ciertos casos las necesidades derivadas de la aplicación de ciertas extensiones hace que este método sea inadecuado [Dalle Molle 89].

El trabajo que hay que realizar en este campo antes de que sea realmente viable el uso de simulaciones cualitativas en control es muy importante. La serie de comentarios sobre la misma que aparecen en este capítulo deben entenderse desde este punto de vista, como indicación del camino a seguir para que este tipo de simulaciones se útil en control. Este trabajo debe orientarse en dos sentidos principales:

- Reducir la explosión combinatoria.
- Reducir la indecibilidad de comportamientos.
- Posibilitar el modelado de sistemas distribuidos.

Aunque en algunos casos las restricciones cualitativas se pueden obtener de las ecuaciones diferenciales ordinarias, para el control de procesos sería preciso que se emplearan ecuaciones diferenciales en derivadas parciales. Este es un tema pendiente de investigación, por lo que se desconoce hasta que punto es esto posible. Quizá el sentido de influencias propugnado por Forbus [Forbus 84] sea aplicable: la variable *tal* crece cuando crece la variable *cual* si todo lo demás sigue igual. Este es un desarrollo futuro derivado de las necesidades planteadas por la arquitectura propuesta en esta tesis.

6.6. Modelo K

Existen hechos de alto grado de abstracción que no admiten representación en los dos niveles anteriores. Ello es debido a que dichos niveles de modelado tienen una orientación física: tratan de modelar sistemas físicos mediante variables, valores de las mismas y relaciones entre ellas. Supongamos un sistema como el que se comenta en el capítulo 8: un horno de clinker. En este sistema existen variables físicas claramente definidas como es el caso de la velocidad de rotación. Así mismo existen variables más abstractas que éstas, pero que admiten valores más o menos cuantizables. Es el caso de la temperatura de zona según vimos en el apartado 6.5. Pero también existen otro tipo de información, asociable a valores de variables solo mediante consideración de un sistema en un nivel de abstracción superior. Las variables más típicas de este tipo son las variables de status⁷ del proceso.

Es habitual en los sistemas basados en el conocimiento la existencia de este tipo de variables, ya que son utilizadas habitualmente por los humanos para formular sus líneas de razonamiento y por

ello son las que van a permitir conducir la inferencia. En nuestro caso todos los conceptos asociados al sistema y que no tienen representación física directa se encuadrarán en este modelo.

6.6.1. Estructura del modelo K

La estructura del modelo K dependerá de cada aplicación del mismo modo que la estructura del conocimiento de un sistema experto depende de la aplicación particular, ya que en la mayor parte de los casos esta estructura refleja el conocimiento de una determinada persona.

6.6.2. Funciones QK y KQ

Este tipo de funciones son muy diferentes a las funciones NQ y QN, dado que asocian dos espacios absolutamente diferentes. Mientras que los modelos N y Q representan sistemas físicos, el modelo K ha sido definido como aquel en el que se incluyen todas las variables que no representan magnitudes físicas.

6.6.3. Funciones del modelo

Las funciones del modelo K son la informativa y la lingüística. La información de este modelo es utilizada por los sistemas de razonamiento para la determinación de las acciones de control.

Por lo general este modelo estará eximido, en una primera aproximación, de la función predictiva, aunque en algunos casos la dificultad de modelado del proceso requiera del uso de técnicas de simulación basada en el conocimiento, técnicas que se deben aplicar en este nivel.

6.6.4. Implantación

El modelo de representación de conocimiento a utilizar puede basarse en una serie de paradigmas perfectamente definidos:

- Cálculo de predicados.
- Cálculo proposicional.

⁷No confundir con las variables de estado.

- Marcos.
- Objetos.
- Redes semánticas.
- etc.

De manera análoga al caso del modelo N, no se encuentra entre los objetivos de esta tesis el desarrollo de un sistema de representación del conocimiento a este nivel. Las alternativas en el mercado son múltiples, por lo que este tipo de desarrollo carece de sentido.

6.7. Utilización de Modelos en Control

Ya en el capítulo 3 se analizaron las características de los modelos empleados en control. En este capítulo se ha presentado un método de construcción de modelos orientado al control, para su utilización en línea. Estos modelos para control son modelos explícitos que se pueden emplear desde un punto de vista múltiple, realizando tres funciones principales:

- Mantener la información sobre el estado del proceso a controlar. Esta información estará disponible en diversos niveles de abstracción, lo que permite su utilización por diferentes aplicaciones.
- Realizar predicciones de comportamiento del sistema.
- Mantener información estructural/funcional, que es necesaria para el razonamiento basado en modelos.

Los sistemas avanzados de control se basan necesariamente en modelos profundos del proceso a controlar, modelos que se construyen utilizando múltiples paradigmas de modelado. En sistemas inteligentes de control la integración de varios de estos modelos puede conducir a sistemas que operen con garantías en una amplia gama de condiciones de funcionamiento.

6.7.1. Información sobre el estado

El método de construcción de modelos presentado en anteriores puntos permite que la información sobre el estado del proceso a controlar tenga las deseables características de multiresolucionalidad. Esto permite que la formulación de estrategias de control sea mucho más clara e inmediata.

En anteriores capítulos hemos comentado que las dos etapas fundamentales de control son la valoración de la situación y la determinación de la acción de control. Las funciones de abstracción utilizadas en la obtención de valores de niveles de abstracción superiores, implantan —en cierta medida— un primer acercamiento a la *valoración de la situación* que constituye la primera etapa del control inteligente.

Es por ejemplo posible que en la determinación de la acción de control sobre un proceso determinado⁸ se base en que dicho proceso tenga una *tendencia térmica positiva*. Para realizar esta determinación de acción de control es preciso pues llevar a cabo primero una determinación de dicha *tendencia térmica*. Esto puede interpretarse como un mero proceso de cálculo asociado al cálculo de la acción, pero su verdadero sentido es el de obtener el valor de una variable de estado del proceso, el *estado térmico*, que corresponde a un determinado nivel de abstracción, superior al de las medidas directas de planta.

Por ello el disponer de un modelo multiresolucional para control inteligente, simplifica sobremanera el desarrollo de los mecanismos en que se va a basar dicho control.

6.7.2. Predicción del comportamiento

En muchos casos los sistemas de control llegan a establecer acciones de control que carecen por completo de garantías de funcionamiento. Esto se produce en el caso de controles clásicos cuando el estado funcional del proceso no coincide con aquel para el que se diseñó el control; o en el caso de control basado en reglas porque las bases de conocimiento son prácticamente inverificables.

⁸Tomado de un desarrollo real.

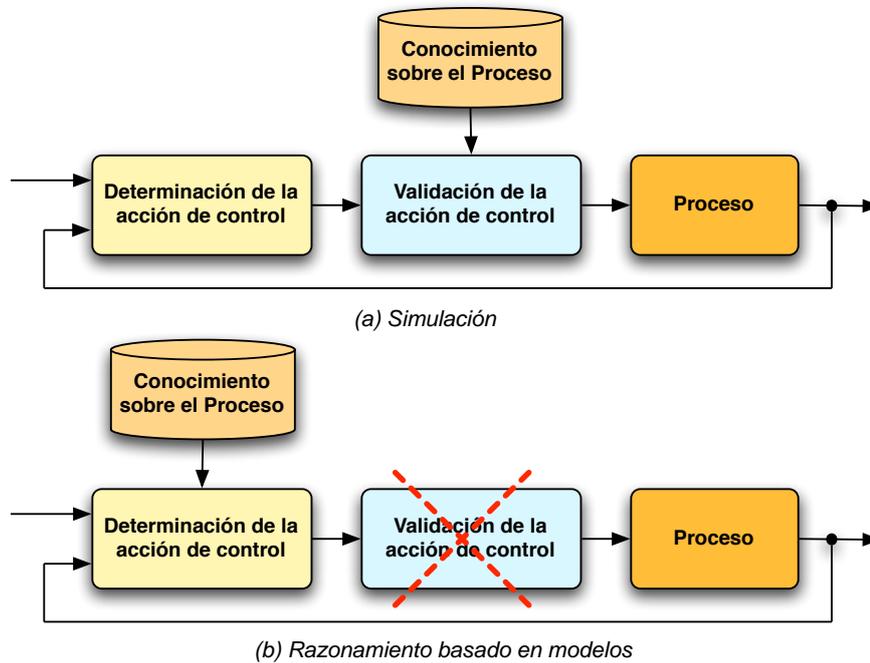


Figura 6.6: Utilización del conocimiento sobre el proceso en control.

Es por ello interesante disponer de medios para validar dichas acciones de control. Esta validación se puede hacer a priori, en base a conocimiento incluido en el algoritmo de determinación de la acción de control o puede realizarse a posteriori, viendo los resultados que produce sobre el proceso. Sin embargo, en muchos casos no es posible utilizar la primera por la escasez de conocimiento disponible para realizarla, ni la segunda por las consecuencias que pudiera tener sobre el proceso.

El disponer de un modelo adecuado del proceso nos permite utilizarlo para validar las acciones de control por simulación. Sin embargo, este mecanismo basado en el conocimiento —el modelo— presenta dos problemas fundamentales:

- La veracidad del modelo: un mal modelo puede conducir a conclusiones erróneas.
- El costo temporal: un sistema de control debe operar en tiempo real, por lo que determinado tipo de procesamiento queda excluido por superar los horizontes temporales admisibles.

La predicción de comportamiento para validación de acciones de control requiere que el modelo utilizado permita especificar las acciones de control a realizar sobre el proceso, por lo que el modelo debe incluir también los reguladores que van a realizar el control de dicho proceso.

En el caso de utilizar sistemas convencionales de control (p.e. PID) este modelado de reguladores es inmediato; pero en el caso de usar sistemas basados en el conocimiento el problema alcanza unos niveles de complejidad enorme. En las actuales circunstancias sólo es factible el realizar la simulación de sistemas del primer tipo — *i.e.* con reguladores clásicos.

6.7.3. Información estructural / funcional

El conocimiento sobre el proceso que se empleaba de forma indirecta simulación para determinar acciones de control, puede ser empleado para la determinación directa de las mismas.

En la Figura 6.7.2 se presentan ambas formas de utilización en línea del conocimiento sobre el proceso en control del mismo. En el caso a) el conocimiento se emplea para validar —por simulación— las acciones de control establecidas por otro elemento. En el caso b) el modelo se emplea directamente para establecer las acciones de control. En este caso puede no ser necesario el emplear una etapa de validación, lo que se trata de indicar en la figura por un aspa.

La determinación de las acciones de control en base al uso de modelos se enmarca dentro de lo que se denomina razonamiento basado en modelos (ver capítulo 4).

Capítulo 7

Arquitectura de Control Inteligente

- Excelencia Ferromagnética, puesto que el electrodragón que estás creando ha de derrotar al que está en nuestro satélite y ocupar su puesto, ¿como haremos para echarle a su vez? - Muy sencillo: fabricaremos otro dragón aún más poderoso, y luego otro para matar a éste, y así sucesivamente ...

*Stanislaw Lem
Fábulas de Robots, 1981*

7.1. Introducción

Es de importancia fundamental al desarrollar una filosofía de control inteligente que sea coherente con los requisitos del control en tiempo real. Por ello la definición de funciones y de arquitecturas que las soporten es quizá el paso más importante el desarrollo de un sistema inteligente de control. En este capítulo se presenta una arquitectura de un sistema de control para procesos complejos: CONEX. Esta arquitectura pretende resolver los problemas mas importantes del control de este tipo de sistemas, recurriendo para ello a diversas técnicas de control inteligente.

El campo de aplicación de esta arquitectura de control son los sistemas con las siguientes características:

- Necesidad de procesamiento de un alto volumen de información, derivado de la complejidad primaria del sistema.
- Necesidad de realizar control de subsistemas fuertemente acoplados origen de la complejidad secundaria.

- Carencia de modelos precisos de comportamiento, derivado de la complejidad secundaria y terciaria.

La operación en tiempo real del sistema se consigue por integración de múltiples paradigmas de control, estructurados de acuerdo con el principio formulado por Saridis de *precisión creciente con inteligencia decreciente*, y con el principio implantacional de *inteligencia creciente con velocidad decreciente*. Esta descomposición permite realizar un desacoplamiento entre el sistema real el proceso y los mecanismos de razonamiento al más alto nivel. Este enfoque es el habitual en la mayor parte de las arquitecturas de control inteligente. Cada implantación llevada a cabo con esta arquitectura será adecuada para controlar determinado tipo de sistemas, que tengan un tempo característico por debajo del horizonte de aplicabilidad del sistema en cuestión.

7.2. Objetivos de la Arquitectura

Los objetivos de esta arquitectura se agrupan en dos tipos: objetivos a satisfacer por el sistema de control final, y objetivos para etapas de desarrollo.

7.2.1. Objetivos del sistema final

El objetivo fundamental de esta arquitectura es el posibilitar el control de procesos que sean complejos en los tres sentidos comentados en el capítulo 2. Estos procesos se caracterizan por que habitualmente precisan de un operador humano para

su correcto funcionamiento. Para lograr este objetivo la arquitectura debe permitir implantar una serie de técnicas para la realización de sistemas inteligentes de control que suministren los servicios necesarios para suplantar al operador de una planta en sus tareas o bien servirle de ayuda en las mismas; el campo de aplicación de este tipo de control serán procesos complejos y/o de operación compleja, bien por el gran volumen de información bien por la baja calidad de la misma. Además, como condición añadida y fundamental en un sistema basado en inteligencia artificial, el sistema final debe tener una alta fiabilidad y disponibilidad.

7.2.2. Objetivos de desarrollo

La construcción de sistemas de control inteligente es una tarea compleja, requiriendo en la mayor parte de los casos el uso de equipos multidisciplinares y de arquitecturas específicas orientadas al problema que se pretende resolver. Esto hace que los costes de desarrollo, así como los tiempos, sean muy elevados. La arquitectura que se presenta pretende resolver en parte esta problemática de forma que se haga factible:

- El desarrollo independiente de los subsistemas.
- La mantenibilidad y extensibilidad de los sistemas desarrollados.
- La reutilizabilidad del software.

7.3. Fundamentos de la Arquitectura

7.3.1. Control de procesos

El analizar las funciones que se realizan para controlar un proceso complejo nos permitirá realizar una descomposición funcional del sistema de control. En control de procesos distinguimos entre tres tipos diferentes de actividades:

Actividades a corto plazo: Por ejemplo las actividades de control orientado al mantenimiento de la temperatura en un determinado punto.

Actividades a medio plazo: Actividad de arranque o de cambio de punto de funcionamiento.

Actividades a largo plazo: Optimización de producción y calidad.

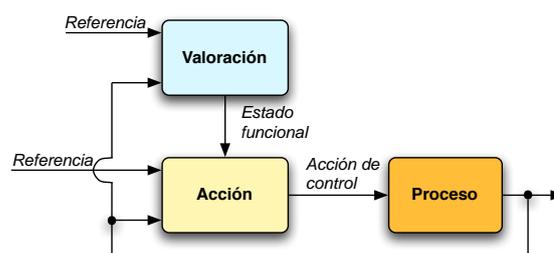


Figura 7.1: Estructura de control avanzado.

Todas estas actividades precisan de diferentes tipos de información para realizar el control adecuado del proceso. Asimismo la actividad de control está orientada en distintos aspectos que deben gestionarse de forma simultánea. Estos tres tipos de actividades están ligadas a los tres niveles estándar de sistemas jerárquicos de control: ejecución, coordinación y organización.

Todo ello hace que el volumen de información y la complejidad de la misma necesarios para los sistemas de control inteligente sean varios órdenes de magnitud mayores que los utilizados en sistemas expertos convencionales. Los sistemas de control clásicos y los sistemas basados en el conocimiento basan sus decisiones en información que se sitúa en diferentes niveles de abstracción. Para realizar con éxito la integración de sistemas expertos en sistemas de control ya existentes o en caso de emplear arquitecturas multinivel —como la que se propone en esta tesis— es preciso el desarrollo de un mecanismo de interconexión entre los datos manejados por el sistema de control o monitorización (señales) y los que maneja el sistema basado en el conocimiento (conceptos abstractos).

El nivel de información abstracto usado por el sistema basado en el conocimiento puede tener elementos en correspondencia directa con las

señales de proceso (nivel de información del sistema de control), aunque quizá la correspondencia sea a través de una conversión simbólica (abstracción). También pueden existir conceptos sin correspondencia directa, que son determinados a partir de los anteriores. Este conjunto de conceptos a diferentes niveles constituye un modelo del proceso. Es un modelo profundo, con estructura jerárquica.

7.3.2. Etapas de control inteligente

En el capítulo 5 decíamos que las etapas a seguir en un sistema de control inteligente son las siguientes:

1. Identificación del estado funcional del proceso
2. Identificación de los posibles cursos de acción
3. Determinación del impacto sobre el sistema
4. Interpretación del impacto
5. Selección y ejecución de una alternativa de control
6. Monitorización de los resultados

La primera corresponde a lo que en el capítulo 4 denominamos valoración de la situación y las restantes salvo la última a lo que llamabamos determinación de la actuación. En un sistema de control clásico la etapa 1 no existe, por lo que las etapas 2, 3 y 4 pierden en gran parte su sentido y desaparecen. La etapa 5 se alcanza con una sola alternativa de control, por lo que la elección es inmediata. La etapa 6 queda un poco al margen de lo que es el proceso de control, aunque es fundamental para la mejora del mismo, bien de forma manual o automática.

En nuestro sistema se tiene una jerarquía de sistemas de control, lo que da lugar a implementaciones particulares para cada nivel de los anteriores procesos.

7.3.3. Niveles de control

En la literatura se habla de sistemas de control y de sistemas de soporte de decisiones; en nuestro caso —procesos complejos— lo habitual es tener sistemas que incluyen aspectos de ambos, realizando el control de subsistemas simples y simultáneamente sirviendo de apoyo a los operadores. En CONEX este reparto de funciones se implanta mediante una jerarquía de niveles de control, que van desde el control automático clásico hasta el control manual. En esencia la supervisión del control consiste en determinar si una situación dada cae fuera del campo de actuación de un determinado nivel, solicitando la intervención de niveles superiores.

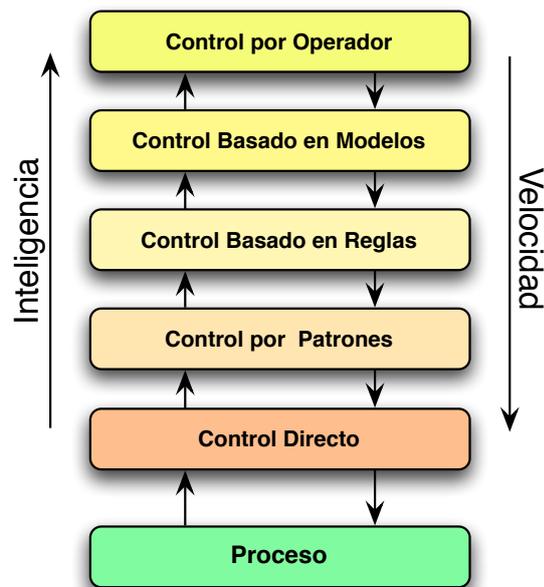


Figura 7.2: Jerarquía de niveles de control en la arquitectura CONEX.

Esta jerarquía de control es la siguiente:

Control directo. En el modo de funcionamiento normal será el único nivel activo de CONEX. En esta situación todas las acciones de control se determinan al nivel más bajo, esto es, por los controladores directos. El funcionamiento

del sistema en esta situación será estable dentro de rangos de valores aceptables. En este nivel se emplearán tanto reguladores clásicos como reguladores borrosos basados en reglas. En este nivel no existe etapa de valoración de situación, por lo que el sistema no puede calificarse de control inteligente.

Control por patrones situación-acción. Los subsistemas superiores al control directo se encuentran en estado de latencia salvo un módulo, denominado monitor, encargado de detectar y reconocer patrones anormales. La detección de un patrón anormal en el proceso supone la consulta a la base de datos de patrones así como a una base de conocimiento para detección de patrones anormales. El monitor determinará si existe una secuencia de acciones de control asociada a dicho patrón; se dirá en este caso que el patrón ha sido reconocido. Si el patrón no es reconocido se invocará al nivel superior: control inteligente. En este nivel la valoración de la situación se hace en base a patrones establecidos. Las etapas de determinación de alternativas de control, valoración y selección, se hacen a priori por un humano que establece los protocolos de control para cada situación reconocida.

Control basado en reglas. El subsistema de control basado en reglas propondrá una serie de acciones de control para dicha situación, utilizando el conocimiento de expertos en control de proceso: operadores e ingenieros de planta. En este nivel las etapas de control se diferencian claramente; por ello es el primer nivel calificable de *inteligente*. La determinación de las alternativas de control se hará mediante razonamiento basado en reglas, con una base de conocimiento para determinación de acciones de control específica del dominio.

Control basado en modelos. Permitirá, en base a un modelo de la planta, proponer acciones de control que serán validadas por un módulo específico. En este nivel las etapas de control también se diferencian como en el anterior. La determinación de las alternativas de control se hace en este caso mediante razonamiento basado en modelos, utilizando un modelo es-

pecífico y una base de conocimientos de control general.

Operador. Si después de todo estas etapas el sistema no alcanza una decisión de acción se requerirá la intervención del operador. En este caso el operador realiza las diferentes etapas asociadas al control inteligente, aunque cuenta con el apoyo del sistema para realizarlas. El interfase de usuario realiza interpretaciones previas de los datos, y realiza valoraciones de acciones de control.

7.3.4. Descomposición heterogénea

El enfoque utilizado en esta arquitectura es el de descomposición funcional y por dominios, empleando una serie de paradigmas específicos. Esta descomposición dará lugar a una jerarquía de niveles de control, cada una con sus capacidades específicas tanto en situaciones controlables como en velocidad de operación. En CONEX se emplean una serie de módulos basados en inteligencia artificial, contruidos con diferentes paradigmas y que utilizan bases de conocimiento diferenciadas. Estas bases de conocimiento son:

- Un conjunto de bases de conocimiento para control borroso. En ellas se tienen una serie de reglas de control que implantan reguladores borrosos para control de bucles simples.
- Una base de conocimiento de control. Esta base de conocimiento contiene reglas de control global del proceso. Estas reglas permiten establecer estrategias globales de control para todo el proceso.
- Una base de conocimiento de metacontrol y una base de conocimiento sobre el proceso modelo del proceso. La base de conocimiento de metacontrol contiene reglas generales de determinación de acciones de control global a partir de la base de conocimiento del proceso. La base de conocimiento del proceso constituye un modelo profundo del mismo, que permite el uso de técnicas de razonamiento basado en modelos.

Los conocimientos contenidos en cada una de las bases de conocimiento son muy diferentes y además son utilizados para diferentes funciones, por lo que apenas existirán problemas de coherencia y de planificación, ya que el diseño de la arquitectura está muy orientado al proceso, siendo todas las bases de conocimiento muy específicas y separadas, por lo que utilización está planificada a priori, lo que hace innecesario el empleo de un planificador de bases en tiempo de ejecución.

7.3.5. Objetos activos y pasivos

Un sistema CONEX se estructura en forma de una colección de *objetos activos y pasivos* [Ellis 89], desarrollados con la metodología de programación orientada a objeto que permitirá alcanzar algunos de los objetivos planteados, en particular la reutilizabilidad y extensibilidad.

En la mayoría de los sistemas orientados a objeto, los objetos deben recibir un mensaje antes de realizar cualquier acción. Un objeto activo es un objeto que puede iniciar acciones de forma asíncrona, sin ser requerido para ello por otro objeto mediante un mensaje. Los objetos activos nos permitirán construir un sistema basado en elementos independientes, con interfases bien definidas entre ellos, y que se comportará como un sistema cooperante. En caso de objetos pasivos la recepción de un mensaje produce la invocación de un método que realizará alguna función demandada por el mensaje. En el caso de objetos activos, éstos, además de dichos métodos, disponen de una serie de procesos que responden de la actividad de los objetos. La invocación de métodos puede ser llevada a cabo de dos formas:

Con bloqueo: Es la única forma posible en el caso de objetos pasivos. Es análoga a la invocación de un procedimiento en programación convencional. El flujo de control pasa del objeto/método invocante al objeto/método invocado, devolviéndose al primero al terminar la ejecución.

Sin bloqueo: Es característico de los objetos activos, produciéndose una bifurcación del flujo de control. El método invocante continúa su

ejecución después de realizar la invocación; el método invocado se ejecuta de forma concurrente con el método invocante.

En sistemas con invocaciones sin bloqueo el flujo de control no es una línea temporal, sin que es un árbol en el que las bifurcaciones son debidas a las invocaciones sin bloqueo. Este tipo de software conlleva problemas de gran dificultad y que precisan de soluciones robustas en el caso de utilización para control. Ejemplo de este tipo de problemas son la recuperación de errores de comunicación entre objetos, fallos en la respuesta del objeto invocado, la programación y control de invocaciones múltiples de métodos de un objeto particular, etc. El fallo de comunicación o de respuesta de un objeto puede producir la interrupción del funcionamiento de un objeto si la invocación se hizo con bloqueo; este tipo de problemas aparecen en tiempo de ejecución por lo que es preciso dotar al sistema de estrategias de desbloqueo timeouts y de recuperación objetos redundantes, métodos alternativos, etc. Las invocaciones múltiples de métodos de un objeto se pueden atender de diversas formas en función del objeto, que puede o no permitir ejecución simultánea de varios métodos. En caso de permitirlo aparecen problemas de coherencia de las ejecuciones, requiriendo por lo general de mecanismos basados en transacciones para el acceso a los datos. En caso de permitir una sola línea de ejecución, esta debe compartirse entre los procesos internos del objeto y los métodos invocados por otros objetos. Por lo general es preciso el uso de mecanismos de interrupción, colas y planificadores¹ para gestionar esta concurrencia.

Los objetos activos permiten además la aparición de un tipo nuevo de mensaje, inexistente en sistemas pasivos: el mensaje múltiple. Este tipo de mensaje no tiene un único destinatario, sino que va dirigido a un grupo de objetos. Un tipo particular de mensaje múltiple es el que va dirigido a todos los objetos y que denominaremos mensaje global.

¹Schedulers.

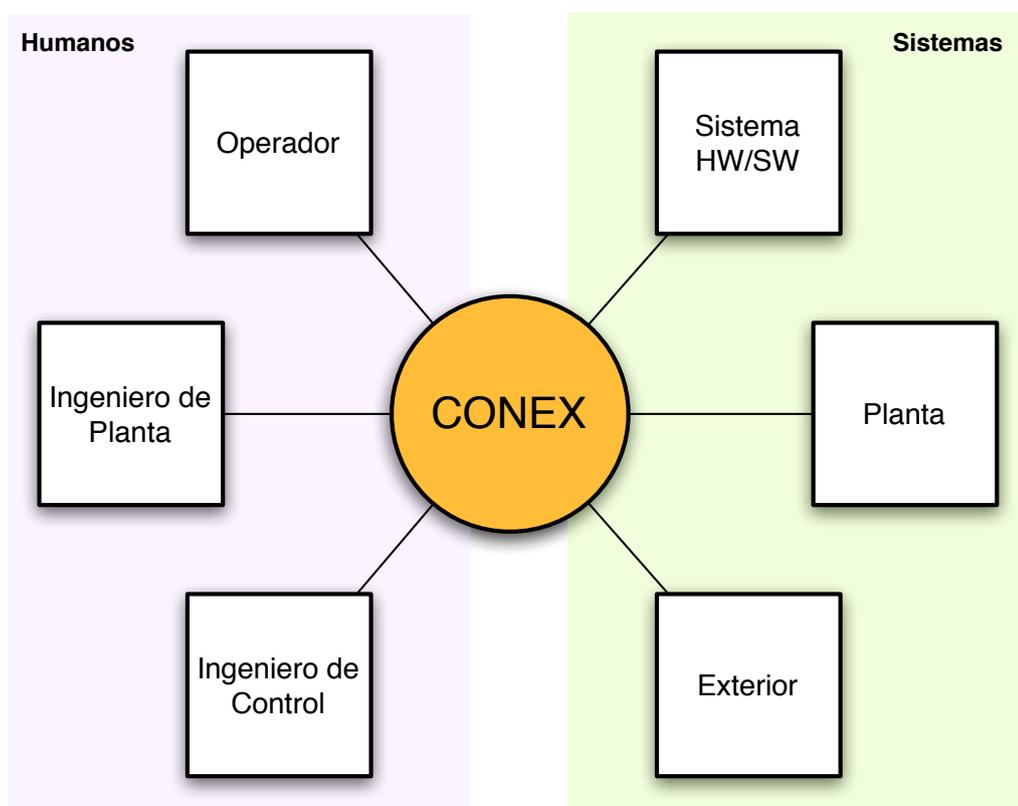


Figura 7.3: Diagrama de contexto de CONEX.

7.4. Arquitectura Software

CONEX es un sistema de control inteligente distribuido. Es un sistema con restricciones débiles de operación en tiempo real debido al tipo de proceso a controlar. No obstante su diseño permite su utilización, con potencialidad limitada, en aplicaciones con requerimientos fuertes. La arquitectura global del sistema se basa en una serie de subsistemas independientes, que interactúan a través de paso de mensajes. Esto hace que las restricciones de cada uno de ellos sean particulares y muy diferentes entre sí. CONEX interactúa con los siguientes sistemas externos:

PLANTA: CONEX es un sistema de control, por lo que debe interactuar con la planta. Esta interacción puede ser directa o bien a través de sistemas de bajo nivel, que permitan un cierto aislamiento y preproceso de señales.

EXTERIOR: CONEX es un sistema abierto, por lo que presenta una interfase de comunicaciones con el mundo exterior que permite la interacción y el intercambio de órdenes y datos con otros procesos externos a CONEX.

SISTEMA HW/SW: CONEX es un sistema de software, y como tal, debe ejecutarse sobre una computadora. La integración con el sistema operativo es fundamental en un sistema avanzado, sobre todo si este debe operar en tiempo real y de forma robusta.

OPERADOR: CONEX trata de mejorar las características del control por integración de modos automáticos y manuales.

INGENIERO DE PLANTA: CONEX se basa, en los niveles más altos de control, en la existencia de modelos multiresolucionales de la planta. El conocimiento del ingeniero de planta se integra junto con el conocimiento del operador para conseguir una base de conocimiento más profunda.

INGENIERO DE SISTEMAS: CONEX es un desarrollo nuevo, y por lo tanto su instalación, puesta en marcha y operación requerirán por lo general de la intervención de ingenieros de sistemas.

CONEX es un desarrollo novedoso, en el que se realiza la integración de sistemas basados en diversas técnicas. Las técnicas necesarias son variadas y en algunos casos es preciso realizar una adecuación de las mismas al control de procesos.

7.4.1. Estructura global

Un sistema CONEX se compone de una serie de objetos activos que denominaremos a partir de ahora OANs (**Objetos de Alto Nivel**), que intercambian información de forma activa mediante paso de mensajes para lograr una serie de fines propios de cada uno. Los OANs que componen un sistema CONEX son los siguientes:

CONEX-CD: Control directo. Encargado del control directo del proceso.

CONEX-IP: Interfase de proceso. Realiza las misiones de conexión con el proceso.

CONEX-MP: Monitor de proceso. Es en esencia un reconocedor de patrones dinámicos en las variables de proceso.

CONEX-MS: Modelo y simulador. Es el sistema de soporte de la base de conocimiento sobre el proceso modelo.

CONEX-CI: Control inteligente. Implanta los mecanismos de control basado en reglas y basado en modelos.

CONEX-IU: Interfase de usuario. Encargado de la interacción con los operadores e ingenieros de proceso.

CONEX-EA: Evaluador de actuaciones. Permite evaluar en términos de objetivos globales las acciones a realizar sobre el proceso.

CONEX-MC: Monitor de CONEX. Es el módulo de vigilancia y control de la propia aplicación CONEX.

CONEX-IE: Interfase externa. Permite interactuar con CONEX a elementos exteriores al mismo.

La comunicación entre los mismos se realiza a través de paso de mensajes, teniendo cada uno de los OANs una interfase bien definida, accesible a los demás OANs. El acceso a los recursos de un OAN por parte de los demás debe hacerse *siempre* a través de uno de los mensajes de la interfase. Todos los datos de CONEX son internos a alguno de los OANs.

Todos los OANs son independientes en virtud del mecanismo de paso de mensajes, por lo que admiten implantaciones independientes, con el único requerimiento del acceso al medio de comunicación común. La dependencia entre OANs está determinada por las necesidades de información o servicios de cada uno de ellos, especificados mediante los grupos OAN/mensajes asociados a cada OAN.

Los tipos de mensajes en CONEX son dos:

Mensajes simples. Son mensajes enviados por un OAN a otro OAN.

Mensajes de grupo. Son mensajes enviados por un OAN a un grupo de OANs. En este caso el destinatario del mensaje es un OAN virtual. Los módulos de gestión de mensajes de los OANs son los encargados de reconocer si un determinado mensaje va destinado a ellos o no. Existe un grupo especialmente importante, denominado G-CONEX, y que comprende a todos los OANs. Cuando un mensaje tiene como destino G-CONEX, diremos que es un mensaje global.

Existe otro mecanismo de envío de un mensaje a un grupo de OANs. Este mecanismo se conoce como lista de clientela. Una lista de clientela es una lista de OANs que requieren de otro un determinado servicio que quede ser servido de forma asíncrona, sin necesidad de un mensaje de solicitud por parte del OAN cliente. El OAN que implanta la lista de clientela determina cuándo se debe servir a los clientes de la lista. Las entradas de las listas de clientela se componen de tres partes *cliente*, *condición* y *objeto* (a quien, cuando y qué). Mientras que los grupos son estáticos las listas de clientela son dinámicas. Esto no es debido a problemas de implantación sino a filosofía de funcionamiento de CONEX. Los grupos

son entes estructurales, asociados al funcionamiento de CONEX. Por contra las listas de clientela están asociadas al comportamiento dinámico de los OANs de CONEX y de los elementos externos a CONEX. Los mecanismos de grupo se manejan por los clientes el servidor sólo envía un mensaje mientras que los mecanismos de clientela son manejados por los servidores.

En los apartados que siguen se analizan cada uno de los OANs de CONEX. Para cada uno de ellos se comentan los siguientes puntos:

Misión: Corresponde a la función de alto nivel que debe implantar el subsistema.

Descripción: Comentario sobre las características generales de dicho subsistema, referentes a los aspectos particulares de la implantación para satisfacer los requerimientos impuestos por la misión de dicho subsistema. Estos apartados se corresponden casi literalmente con los correspondientes en [Sanz 89].

Técnicas: Se comentan las principales técnicas en que se basa el OAN. La diversidad de estas técnicas es la que hace precisa la existencia de un equipo multidisciplinar para el desarrollo de aplicaciones de control inteligente. El conocimiento de estas técnicas es una de las condiciones principales para el buen fin de un proyecto en este campo.

En un documento de análisis más profundo que no se ha incluido en la presente tesis se recogen los siguientes elementos de los OANs:

Objetos internos: Conjunto de objetos que constituyen el OAN y que realizan las funciones que éste sirve al exterior. Estos objetos internos no responden a la estructura estándar de un OAN, sino que se realizarán de acuerdo con la implementación particular que se seleccione para cada uno de ellos.

Interfase: En este punto se analizan dos aspectos. Por una parte la **entrada**, referente al conjunto de mensajes que reconoce el objeto. Por otra parte la **salida**, que recoge los conjuntos de datos generados y los destinatarios de los

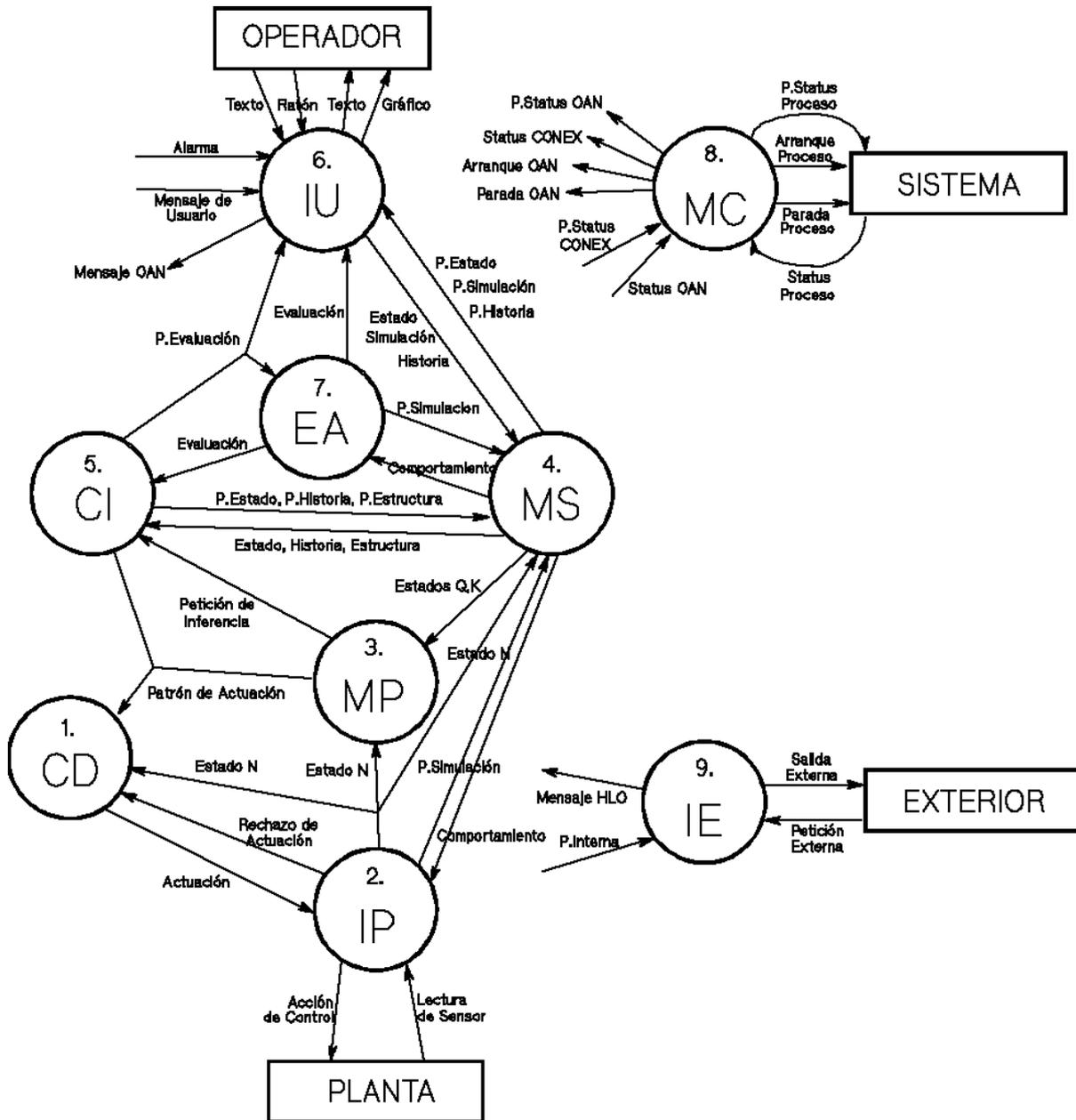


Figura 7.4: Estructura global de CONEX con las principales interacciones entre los nueve objetos activos de alto nivel: CD: Control directo, IP: Interfase de proceso, MP: Monitor de proceso, MS: Modelo y simulador, CI: Control inteligente, IU: Interfase de usuario, EA: Evaluador de actuaciones, MC: Monitor de CONEX e IE: Interfase externa.

Objetos de Alto Nivel		
Acrónimo	Nombre	Misión
CD	<i>Control Directo</i>	Generación de referencias para los actuadores o para los bucles de control de bajo nivel.
IP	<i>Interfaz de Proceso</i>	Comunicación / aislamiento con el proceso, tanto desde el punto de vista de sensores como de actuadores.
MP	<i>Monitor de Proceso</i>	Determinación del estado funcional del sistema en base a una biblioteca de patrones y una base de conocimiento de identificación.
MS	<i>Modelo y Simulador</i>	Mantenimiento de la información sobre el proceso, tanto la estructural/funcional —utilizada en las simulaciones— como la factual —valores de las variables de proceso.
CI	<i>Control Inteligente</i>	Control basado en reglas y control basado en modelos.
IU	<i>Interfase de Usuario</i>	Comunicación con el operador y con los ingenieros de proceso.
EA	<i>Evaluador de actuaciones</i>	Análisis de las actuaciones de control en base a una evaluación de los efectos de las mismas en términos de objetos de alto nivel.
MC	<i>Monitor de CONEX</i>	Monitorización y control de CONEX.
IE	<i>Interfase Externa</i>	Comunicación con elementos externos a CONEX.

Tabla 7.1: Misiones principales de los objetos de alto nivel de CONEX.

mensajes asociados a dichos datos. Estas salidas se corresponden con mensajes de las interfases de otros objetos. En teoría en un sistema orientado a objeto la especificación de las interfases debe incluir exclusivamente los métodos propios de cada objeto. Pero en el caso de sistemas cooperantes basados en objetos activos, es preciso determinar todas las necesidades de los objetos, ya que será de importancia crucial a la hora del funcionamiento real. Por ello hay que analizar para cada objeto para cada OAN todas las relaciones de clientela con otros OANs.

Ficheros: Se refiere a los ficheros más importantes asociados a un determinado objeto. Estos ficheros son especialmente importantes en esta arquitectura de control porque contienen toda la información relevante sobre el estado del OAN, de forma que pueda recuperar su estado a partir de los ficheros si, por cualquier motivo, viera interrumpida su operación.

7.4.2. Estructura estándar de un OAN

Todos los OANs comparten parte de su estructura interna, de modo que todos respondan a ciertos modos de funcionamiento común especificados en la interfase estándar del OAN.

Por ello todos los OANs disponen de dos objetos internos que responden a los nombres de **gestor de mensajes** y **gestor de status**.

7.4.3. Interfase estándar de un OAN

Todos los OANs disponen de un objeto interno denominado GM que es el encargado de realizar las tareas de interfase del objeto con el resto de los OANs de CONEX. Este módulo de interfase recibe mensajes de los objetos internos al OAN y los traduce en mensajes externos según el formato estándar para CONEX.

Todos los GM implantan una serie estándar de mensajes, tanto de entrada como de salida. Este grupo de mensajes constituyen lo que se denomina *interfase estándar* de un OAN.

Mensajes de entrada

Petición de status: Mensaje de solicitud de información sobre el estado interno del objeto. Este mensaje es público, pudiendo recibirse y aceptarse de cualquier otro OAN.

Configuración: Cambio de los parámetros internos del OAN, esencialmente relacionados con el modo de funcionamiento del OAN. Este mensaje es semipúblico, pudiendo recibirse de un grupo determinado de objetos, fundamentalmente CONEX-MC y CONEX-CI.

Arranque: Este mensaje es enviado exclusivamente por CONEX-MC, por lo que se trata de un mensaje privado. Indica al OAN que debe realizar el procedimiento de arranque.

Parada: Análogamente al mensaje de arranque es un mensaje privado de CONEX-MC. Indica al OAN que debe iniciar el procedimiento de parada.

Mensajes de salida

Status: Este mensaje se envía en respuesta a una solicitud del mismo (Petición de status).

Mensaje al usuario: es un mensaje destinado al operador, por lo que va dirigido a CONEX-IU.

Alarmas: Pueden ser de tipos diferentes, pero van dirigidas en cualquier caso a G-CONEX.

7.4.4. CONEX-CD: Control Directo

Misión

CONEX-CD es el módulo de control directo de CONEX. Tiene como objetivo principal la generación de referencias para los actuadores o para los bucles de control de bajo nivel (PLCs).

Descripción general

El corazón de CONEX-CD está constituido por una serie de reguladores basados en diversas

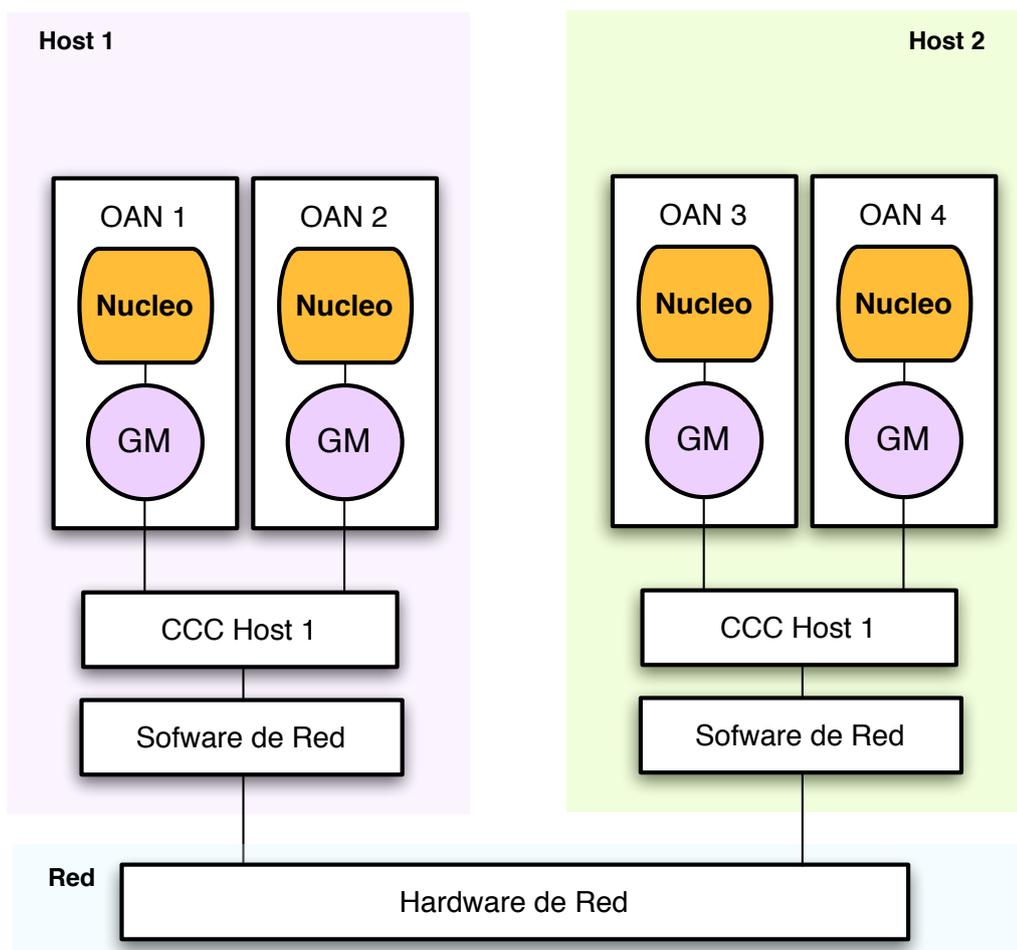


Figura 7.5: Estructura de software de CONEX. Distribución de objetos de alto nivel.

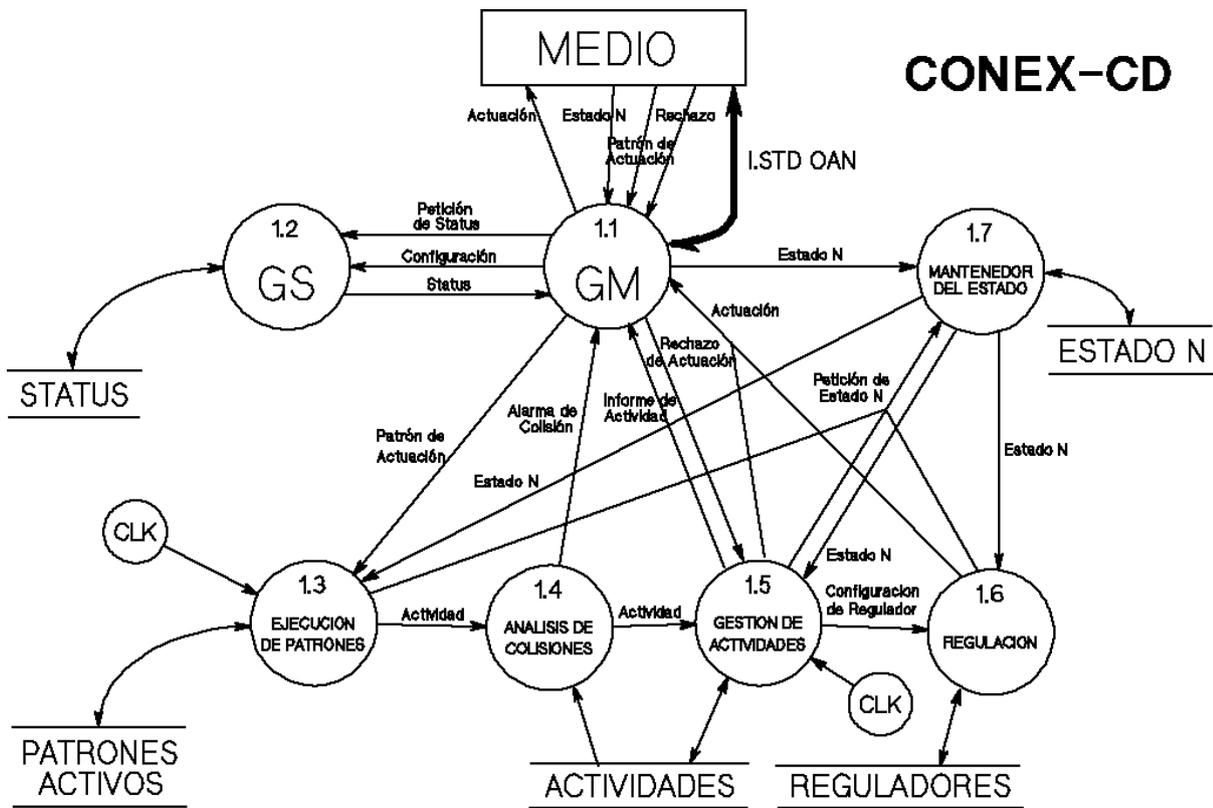


Figura 7.6: CONEX-CD: Control Directo

técnicas y encargados del control de bucles simples dentro del proceso global. Estos reguladores de bajo nivel están encargados de las tareas de mantenimiento de consignas de los diferentes bucles de control. Este OAN debe ser capaz de mantener el funcionamiento adecuado del sistema controlado en un entorno próximo al punto de funcionamiento. Es necesariamente un sistema en tiempo real. Para realizar el control directo el sistema implanta una serie de reguladores encargados del control de subprocesos. Estos reguladores son simples, en el sentido de realizar un control de pocas variables. Los aspectos de interacción entre subprocesos se gestionan en un nivel superior. Los algoritmos de control se pueden desarrollar e implantar de forma independiente del resto del sistema. La filosofía de control a utilizar puede ser cualquiera, variando en un amplio rango de posibilidades: PID, adaptativo, borroso, etc.

Las intervenciones de los OANs de nivel superior se hacen en forma de especificación de patrones de actuación de CONEX-CD. Un *patrón de actuación* es una lista temporal de actividades con bifurcaciones. En este patrón de actuación se especifican las evoluciones temporales de:

- Estructuras de control: tipos de reguladores a utilizar.
- Parametrización de los reguladores.
- Consignas de los bucles de control.

Las actividades especificables en un patrón de actuación son de dos tipos: instantáneas y temporales. Una actividad instantánea puede ser incondicional o condicionada a determinados valores de variables del proceso (estado N)². Una actividad temporal debe tener una condición de inicio y una condición de terminación. Un ejemplo típico de actividad instantánea es un cambio de tipo de regulador (paso de un PID a un regulador borroso por ejemplo). Un ejemplo típico de actividad temporal es el cambio de referencia de forma incremental, desde el valor inicial al final en un período de tiempo determinado.

²No se considera la utilización de condiciones sobre los otros dos estados para evitar dependencias directas de los niveles superiores. El utilizar este tipo de criterios conduce a la necesidad de realizar un diseño de patrones dual, con y sin sistema de alto nivel.

Los patrones de actuación se ejecutarán en base a un reloj interno del OAN. La ejecución de un patrón originará en algunos casos una colisión de actividades. Estas colisiones se detectan y gestionan por un módulo específico de CONEX-CD denominado analizador de colisiones. Una colisión por lo general conducirá a la cancelación de las anteriores actividades, en función de criterios de prioridad. Ello conduce a que el sistema que genere el patrón de actuación deba tener en cuenta este hecho. La ejecución de actividades tiene como resultado la modificación del fichero de reguladores, aunque es posible el uso de especificaciones directas de actuación.

En este sistema se propone en principio la utilización de dos tipos de reguladores sin que ello signifique la exclusión de otro tipo de posibilidades:

- Reguladores PID en aquellos bucles simples de control que no ofrezcan problemas de seguridad y tengan grados aceptables de fiabilidad de las medidas.
- Reguladores borrosos basados en reglas cuando el subproceso sea complejo y/o exista alguno de los problemas mencionados en el apartado anterior. Los reguladores borrosos presentan en estos casos un comportamiento más robusto y fiable, con unas prestaciones (tiempo de subida, sobreoscilación, etc) sólo ligeramente peores; aunque en casos de sistemas complejos este comportamiento puede incluso llegar a ser mejor que el de el PID.

Estos reguladores estarán diseñados para un acoplamiento adecuado con el resto del sistema de control, pero siempre teniendo en cuenta que, en funcionamiento autónomo, se deben comportar de forma adecuada en cualquier circunstancia, por lo que evitarán en cualquier caso realizar acciones que puedan resultar peligrosas o costosas. En funcionamiento estacionario el sistema de control se comporta como si sólo constara de estos reguladores, ya que no se dispararán actuaciones de los niveles superiores de control (CONEX-MP y CONEX-CI). En la especificación de actividades de modificación de reguladores se establecen las condiciones de aplicación de las acciones de con-

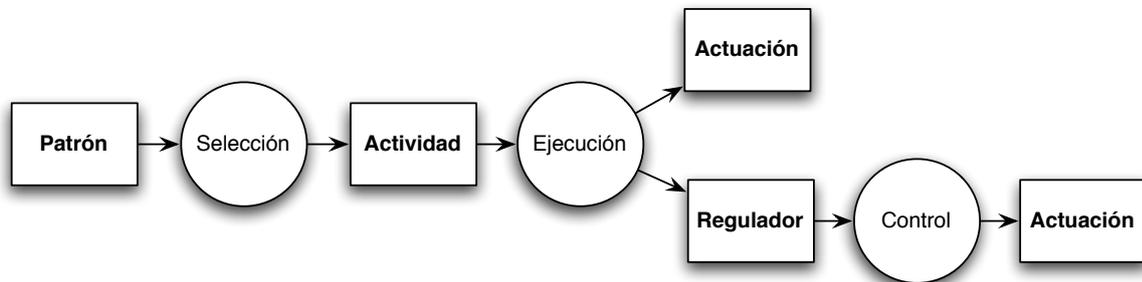


Figura 7.7: Procesado de patrones en CONEX-CD.

trol calculadas con estos reguladores. Estas condiciones expresan las circunstancias bajo las cuales el regulador especificado es adecuado.

El objetivo de este módulo es garantizar el funcionamiento estable del sistema en condiciones normales y con rendimientos adecuados. Este objetivo parece más fácil de alcanzar mediante el uso de reguladores borrosos, ya que son más robustos y permiten manejar de manera más eficaz condiciones de funcionamiento más extremas.

Técnicas

- Control clásico.
- Control borroso.
- Lógica temporal.
- Programación en tiempo real.
- Programación orientada a objeto.
- Redes.

7.4.5. CONEX-IP: Interfase de Proceso

Misión

CONEX-IP es el interfase de proceso de CONEX; sirve de comunicación/aislamiento con el proceso, tanto desde el punto de vista de sensores como del de actuadores.

Descripción general

La interfase de proceso es un subsistema que interacciona con los sensores y actuadores de la planta, por un lado, y por el otro con el resto del sistema inteligente de control. Su misión es la de obtener las mejores estimaciones posibles para las variables del proceso. Esta misión es fundamental debido a la mala calidad de las medidas habitual en plantas industriales, y que es debida a anomalías de diferente origen. Esto hace que un filtrado convencional no sea lo suficientemente adecuado. Este módulo incorpora funciones de alto nivel para detectar dichas anomalías en las señales de la planta. Estas anomalías son debidas a diversas causas, entre las que destacan: perturbaciones eléctricas, mal funcionamiento del transductor, medidas de mala calidad debido al tipo de magnitud a medir, perturbaciones del proceso en la zona de medición, etc.

Además realiza misiones de interfase de control, enviando los valores de las actuaciones a los sistemas de bajo nivel, previa verificación de las condiciones de dicha actuación.

CONEXIP dispone de un reloj interno que es el que genera las demandas de medición de variables. El envío de la información sobre el estado del proceso, estado N, se hará por medio de un mensaje de grupo dirigido a G-CLIENTES-N; los receptores de este mensaje serán CONEXMS, CONEXMP y CONEX-CD pero el utilizar un mensaje de grupo va a posibilitar la inclusión inmediata de nuevos clientes, liberando a CONEXIP de la ne-

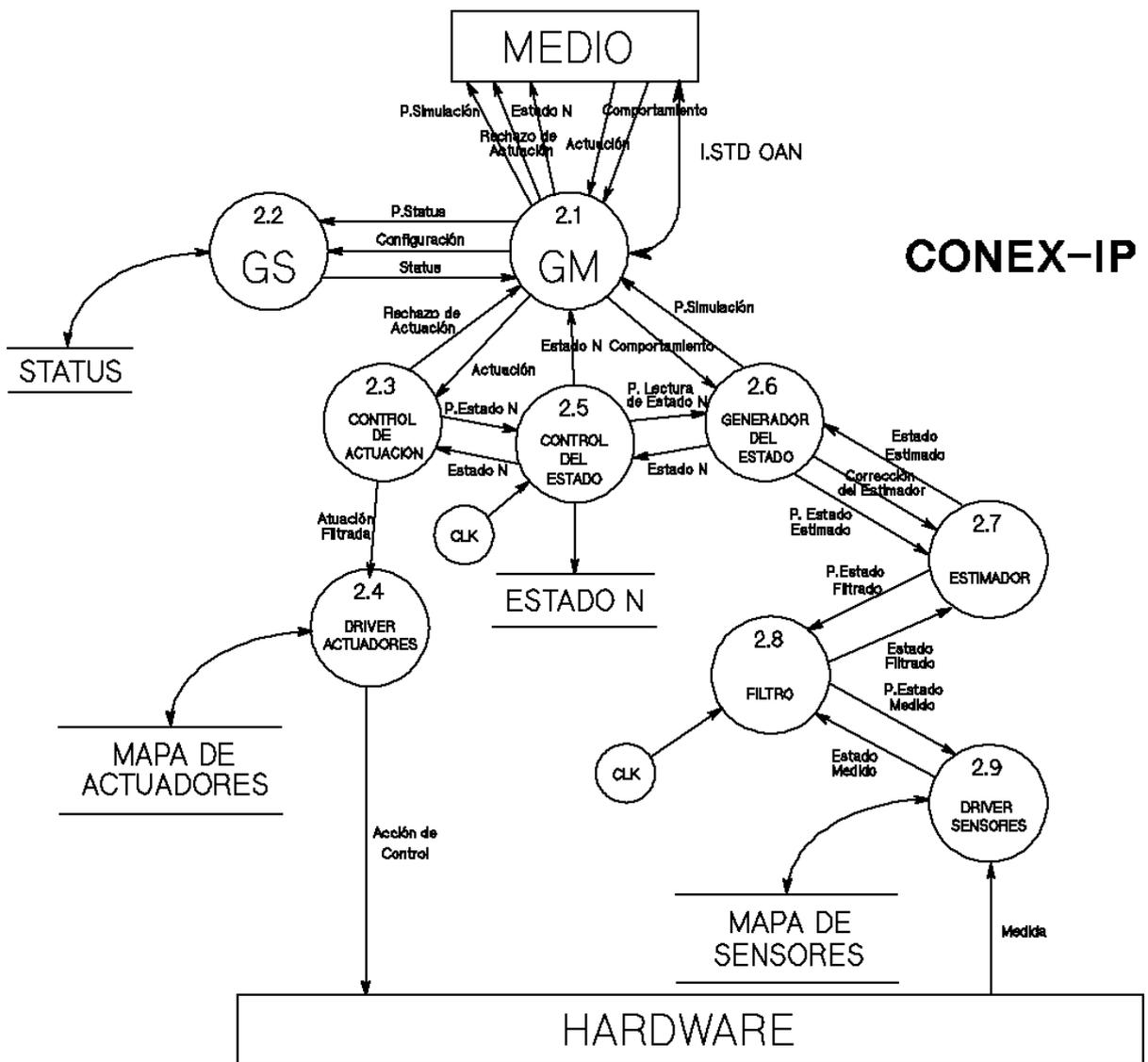


Figura 7.8: CONEX-IP: Interfase de Proceso

cesidad de gestionar una lista de clientes.

La interfase de proceso se estructura pues en dos partes: la interfase de entrada y la interfase de salida. La interfase de entrada utiliza como datos de partida los siguientes:

- Medidas procedentes del sensor, que pueden venir afectadas por diversos tipos de ruidos, procedentes de cualquiera de los elementos del sistema (transductor, línea, conversor A/D, etc).
- Estado esperado del sistema, calculado por el módulo de simulación de CONEXMS. Este estado esperado se recibe en forma de comportamiento predicho en respuesta a una petición de simulación.

En caso de ser necesario se considerará la utilización de los siguientes datos:

- Modelo del sensor, para estimar las probabilidades de los diferentes modos de fallo.
- Condiciones de funcionamiento del sensor (modelo del entorno del mismo, características de la magnitud a medir, etc)
- Información adicional suministrada por el operador.

Con todo ello la interfase de proceso determinará para cada sensor lo siguiente:

- Valor estimado de la magnitud.
- Factor de confianza en dicha medida.
- En caso de factor de confianza pequeño, posible causa de la malfunción.

La utilización del modelo del proceso — estructura del proceso— permitirá a este sistema determinar también:

- Incongruencias entre valores medidos por los sensores.

- Valores estimados para variables ocultas del proceso, esto es, variables que no tienen sensores para su medida directa. Aquí nos referimos a variables del proceso que tienen existencia real, por ejemplo la temperatura de zona de un horno.

Las sucesivas transformaciones que sufren los valores se presentan en la Figura ???. Los valores esperados se obtienen de una petición de simulación al simulador. Esta simulación devuelve también los valores de variables ocultas. Si el valor estimado y el esperado coinciden se aceptan los estimados y las ocultas esperadas.

La interfase de salida genera acciones de control elementales a partir de las actuaciones que le son remitidas por CONEXCD. Antes de generar las acciones se filtran en función de las condiciones de aplicación expresadas para la actuación. En caso de rechazo se informará a CONEXCD de este hecho.

Técnicas

- Filtrado.
- Reconciliación.
- Equipos de control de bajo nivel.
- Programación en tiempo real.
- Programación orientada a objeto.
- Redes.

7.4.6. CONEX-MP: Monitor de Proceso

Misión

CONEX-MP es el monitor de proceso de CONEX, realizando la función de valorar la situación de proceso. Para ello determinar el estado funcional del sistema en base a una biblioteca de patrones y una base de conocimiento de identificación.

En determinados casos, establece patrones de actuación o bien requiere la intervención del sistema superiores CONEX-CI.

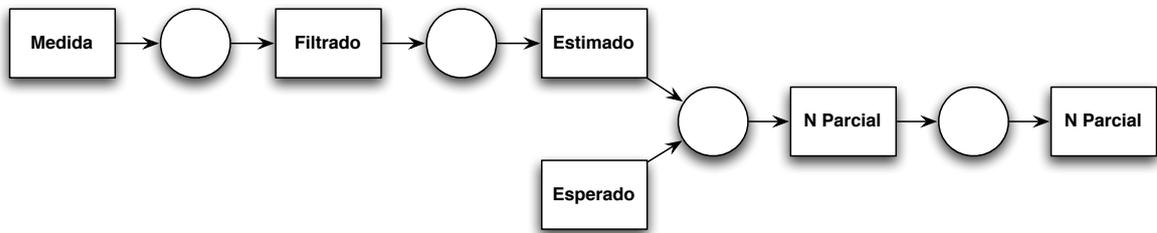


Figura 7.9: Transformaciones de señales en CONEX-IP.

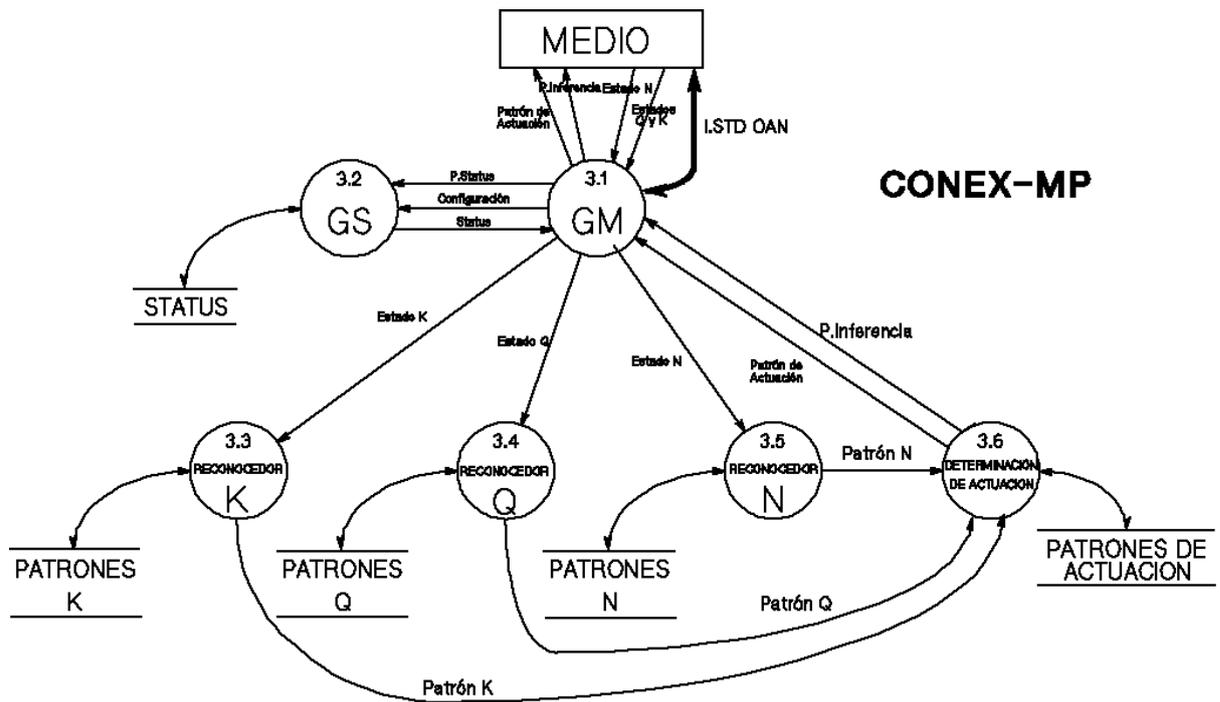


Figura 7.10: CONEX-MP: Monitor de Proceso

Descripción general

Determinará el estado del proceso en base a reconocimiento de patrones en las señales del proceso. Dispone de una biblioteca de patrones de actuación asociados a determinadas situaciones del proceso.

Sus misiones se sitúan en ambas facetas del control:

Valoración de la situación. Esta etapa permitirá establecer la necesidad o no de la actuación. La situación del proceso se valora por medio de una base de conocimiento de identificación de estado y una biblioteca de patrones de comportamiento. El monitor inicia pues acciones de control si el estado del proceso es NO ambiguo o se requiere una intervención urgente (tiempo crítico).

Determinación de la actuación. En esta etapa se dan del mismo modo dos posibilidades: a) que el sistema se esté comportando de acuerdo con un patrón para el que se establecido de forma unívoca un patrón de actuación y b) que no se disponga de un patrón de actuación, por lo que se requerirá la intervención de CONEX-CI.

Este reparto de funciones está orientado a conseguir un razonamiento progresivo distribuido, orientándolo hacia un aumento de eficiencia y seguridad de operación.

Aspecto fundamental del sistema es el almacenamiento de conocimientos (patrones situación-actuación). Para ello es preciso desarrollar un mecanismo eficiente de almacenamiento, basado en abstracción y razonamiento por analogía. El reconocimiento de patrones se hará en tres los niveles utilizados por CONEX: N, Q y K. Estos patrones se especificarán en la mayor parte de los casos como patrones Q, ya que el conocimiento proviene de expertos humanos. Los patrones de actuación son listas temporales, expresando protocolos de actuación a lo largo del tiempo; estas protocolos especifican:

- Consignas de los bucles de control

- Configuraciones de los bucles de control y parámetros de los mismos.
- Actuaciones directas.

El establecimiento de un patrón de actuación supone que el proceso se encuentra en una determinada situación. la ejecución de estos patrones se realiza en un nivel inferior y a lo largo del tiempo, por lo que es posible que la situación del proceso varíe. CONEX-IP verifica antes de realizar una actuación que las condiciones que determinaron la necesidad de la actuación se sigan manteniendo. Para ello los patrones de actuación incluyen información sobre el estado funcional del proceso que determino su actuación y que condicionará su aplicación.

Técnicas

- Programación en tiempo real.
- Bases de datos en tiempo real.
- Reconocimiento de patrones.
- Programación orientada a objeto.
- Redes.

7.4.7. CONEX-MS: Modelo y Simulador

Misión

CONEX-MS es el gestor del modelo y el simulador de CONEX. Tiene como misión el mantener toda la información sobre el proceso, tanto la estructural/funcional utilizada en las simulaciones como la factual valores de las variables de proceso.

Descripción general

CONEX-MS es el OAN encargado de manejar la información sobre la planta y suministrarla a los restantes OANs. Consta esencialmente de tres partes: el gestor del modelo, la base de datos y el simulador.

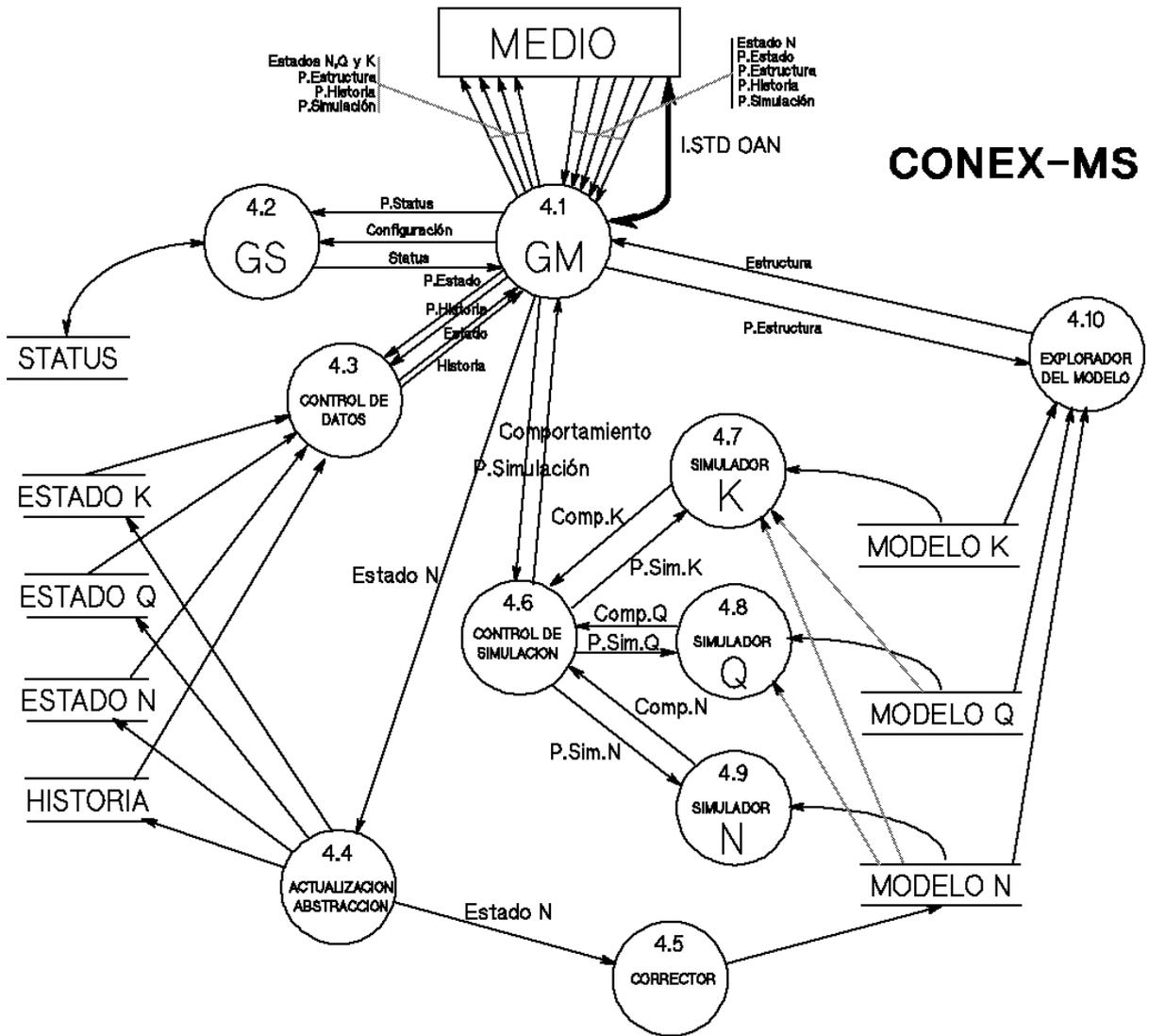


Figura 7.11: CONEX-MS: Modelo y simulador

CONEX-MS es el gestor de la base de datos del proceso, por ello es el encargado de gestionar la información específica del proceso a controlar. Esta información está compuesta por los tres niveles de modelado así como de los valores de las variables de dichos modelos. Este sistema es el encargado de recabar y suministrar toda la información sobre la planta. El acceso a los ficheros de modelo y datos debe hacerse mediante peticiones a CONEX-MS.

Este sistema se construirá sobre un gestor de bases de datos relacionales. Mantendrá información histórica sobre evoluciones de las variables. La estructuración de la información histórica se hará en base a episodios. Este sistema implanta las funciones de abstracción para el paso de valores del nivel inferior N a los niveles superiores Q y K.

Su misión fundamental es la de servir de soporte de información sobre el proceso a los diferentes elementos del sistema de control inteligente.

Así proporciona información a:

- CONEX-IP: Predicciones de evolución de las variables del proceso, de forma que la interfase pueda detectar incongruencias o estimar valores de variables ocultas.
- CONEX-IU: Sobre el estado del sistema.
- CONEX-MP: Estados Q y K del proceso.
- CONEX-CI: Sobre el estado del proceso actual e histórico.
- CONEX-EA: Sobre la evolución del sistema frente a acciones de control.

Misión del modelo

El modelo y los elementos asociados se implantan como un proceso independiente (OAN). Debe considerarse no sólo como una base de datos sino como un sistema activo, en el sentido de permitir evaluar la evolución del proceso desde diferentes puntos de vista (simulación en varios niveles de abstracción).

CONEX-MS consta de varios elementos fundamentales:

- La base de datos: Constituida por varios ficheros que almacenan tres tipos de información:
 - El modelo del proceso en tres niveles de abstracción.
 - Los datos sobre el estado actual del proceso, separados en grupos de datos diferentes según las capas de modelado.
 - La historia del proceso. Este es el grupo de datos mas importante, y que se usa con un gestor de bases de datos relacionales.
- El gestor de datos: Encargado de mantener y suministrar la información del estado actual e histórica. Este subsistema implanta los mecanismos de abstracción.
- El explorador del modelo: Gestiona la información contenida en cada uno de los grupos de datos que constituyen el modelo, proporcionando información sobre las influencias entre variables de proceso..
- El simulador: Realiza predicciones sobre la evolución del proceso, utilizando para ello los tres niveles de modelado.
- Corrector de modelos: Si la estructura del sistema lo permite y se vé su viabilidad, se implantará un módulo de corrección del modelo numérico ajuste de parámetros en base a las mediciones sobre el estado del proceso.

Toda la información sobre el proceso se estructura en base al modelo del mismo y a los tres niveles de abstracción de modelado:

Modelo cuantitativo: Corresponde con un modelo para simulación en sentido clásico.

Modelo cualitativo: Supone una primera etapa de abstracción a partir del modelo cuantitativo. En esta capa los parámetros del modelo se manejarán utilizando procedimientos de la teoría de conjuntos borrosos y dependencias entre variables basadas en ecuaciones diferenciales cualitativas.

Modelo de conocimiento: Corresponde con un conjunto de aserciones que constituyen parte de la base de conocimiento sobre el proceso, y que permiten expresar conocimientos sobre el comportamiento del proceso.

El gestor de datos es el encargado de gestionar los flujos de información internos —entre capas— y externos —con otros elementos del sistema inteligente de control. Entre las funciones más importantes cabe destacar la presencia de las funciones de abstracción, que permiten pasar de la información de un nivel a la del inmediatamente superior. De este modo se harán las siguientes abstracciones:

Estado N → Estado Q → Estado K

Otra misión importante llevada a cabo por el explorado del modelo es la exploración inversa del modelo para razonamiento causal orientado a la determinación de acciones de control.

El simulador implanta métodos de simulación continua, simulación cualitativa y simulación basada en el conocimiento para realizar predicciones sobre las evoluciones del sistema. La característica más innovadora es que el sistema dispone de un **modelo cualitativo** del proceso que sirve de base para una simulación del mismo tipo —determinación de tendencias. De este modo se puede establecer la posible evolución del sistema frente a las acciones de control propuestas por el operador o por el planificador en términos lingüísticos.

El nivel de exactitud del simulador es uno de los puntos cruciales, dado que hay que llegar a una solución de compromiso entre la exactitud de la predicción y la velocidad con que se obtiene la misma. La simulación cualitativa permite un enfoque adecuado de este problema, permitiendo de forma adicional un rápido desarrollo de los modelos particulares de cada aplicación.

El simulador debe predecir el comportamiento del proceso. Esta determinación del comportamiento se podrá hacer en cualquiera de los niveles:

- En el nivel más alto se llevará a cabo mediante encadenamiento de reglas de comportamien-

to.

- En el nivel intermedio se realizará una simulación cualitativa de tendencias.
- En el nivel más bajo la simulación a realizar será una simulación clásica.

Para realizar la simulación del proceso se debe incluir en el modelo la información referente a los bucles de control del mismo, tal como son implantados por CONEX-CD. Esta información es suministrada por CONEX-CD cion su mensaje de status, del que CONEX-MS será un cliente, por lo que todas las modificaciones del status de CONEX-CD le serán comunicadas.

El modelo —entendiendo como tal no sólo la representación del estado del sistema, sino también todas las tareas que lo acompañan— debe suministrar al sistema una serie de funciones:

- Sirve como base de datos del sistema, informando del estado actual del proceso.
- Permite desarrollar razonamientos hipotéticos —gracias al soporte que ofrece para contextos múltiples. Esto será de utilidad para el módulo de validación de acciones del operador y para la planificación de acciones por parte del sistema experto.
- Permite razonamiento temporal, posibilitando el almacenar historias de la evolución del proceso.
- Permite representaciones restringidas del estado del proceso, que se corresponden con abstracciones del estado real. Estas representaciones restringidas serán de importancia crucial para mantener limitado el volumen de datos, sobre todo en lo referente al razonamiento temporal (historias de estados) y al razonamiento por analogía que será el fundamento del aprendizaje (función de muy alto nivel).

Los niveles de modelado se estructuran de acuerdo con el principio de precisión creciente con inteligencia decreciente, lo que hace que aun cuando ambos aspectos precisión e inteligencia sean

deseables, no es posible su concreción en un modelo único. Sin embargo si el modelo numérico es lo suficientemente bueno la necesidad de los otros dos modelos se reduce sobremanera. Su supresión no será en general posible dado que la representación del conocimiento empleada en estos dos modelos de alto nivel es muy adecuada para fundamentar la comunicación con el operador.

Técnicas

- Bases de datos.
- Bases de conocimiento.
- Modelado multiresolucional.
- Simulación continua.
- Simulación cualitativa.
- Sistemas basados en el conocimiento.
- Programación orientada a objeto.
- Redes.

7.4.8. CONEX-CI: Control Inteligente

Misión

CONEX-CI es el módulo de control inteligente de CONEX, debiendo generar patrones de actuación para CONEX-CD.

Descripción general

Es un sistema de control basado en el conocimiento, con requerimientos débiles de operación en tiempo real. Para generar las estrategias de control utiliza dos tipos de mecanismos:

- El primero es *control basado en reglas*, para el que se emplea una base de conocimiento de control obtenida del operador de la planta y una base de conocimiento de metacontrol, que es independiente de la planta.

- El segundo es *control basado en modelos*, en el que la determinación de las actuaciones posibles se hace en base al modelo estructural/funcional del proceso y a reglas generales de determinación de acciones de control.

Este sistema está estructurado en niveles y basado en lógica borrosa, determinando patrones de actuación en los que se especifican los valores para las variables de control así como los reguladores más adecuados en función del estado del proceso. La detección de patrones de cambio en señales del proceso —por parte del monitor— producen el lanzamiento del controlador inteligente en lo que se denominará una *tarea*. El sistema de gestión de tareas del controlador inteligente debe tener las siguientes características:

- Debe permitir la **ejecución múltiple**, necesaria para la resolución de varios problemas, permitiendo el uso de diferentes bases de conocimiento bases de conocimiento descompuestas para facilitar la focalización de la tarea de resolución de problemas.
- Debe existir un **método de interrupción** de tareas (ejecuciones del sistema experto) en función de prioridades.
- Debe poder realizar **razonamiento temporal** lo que supone manejar secuencias de estados (Historias de variables):
 - Debe disponer de suficiente potencia de cálculo en los diferentes niveles de abstracción (cálculo cualitativo y cuantitativo).
 - Deben existir mecanismos de focalización con orientación episódica.
 - Debe incorporar métodos adecuados de olvido.
- Los **contextos múltiples** son necesarios en algunas tareas de razonamiento hipotético.
 - Hipótesis de faltas.
 - Simulación de acciones de control.
- El controlador inteligente dispone de varias fuentes de conocimiento para la determinación de acciones de control:

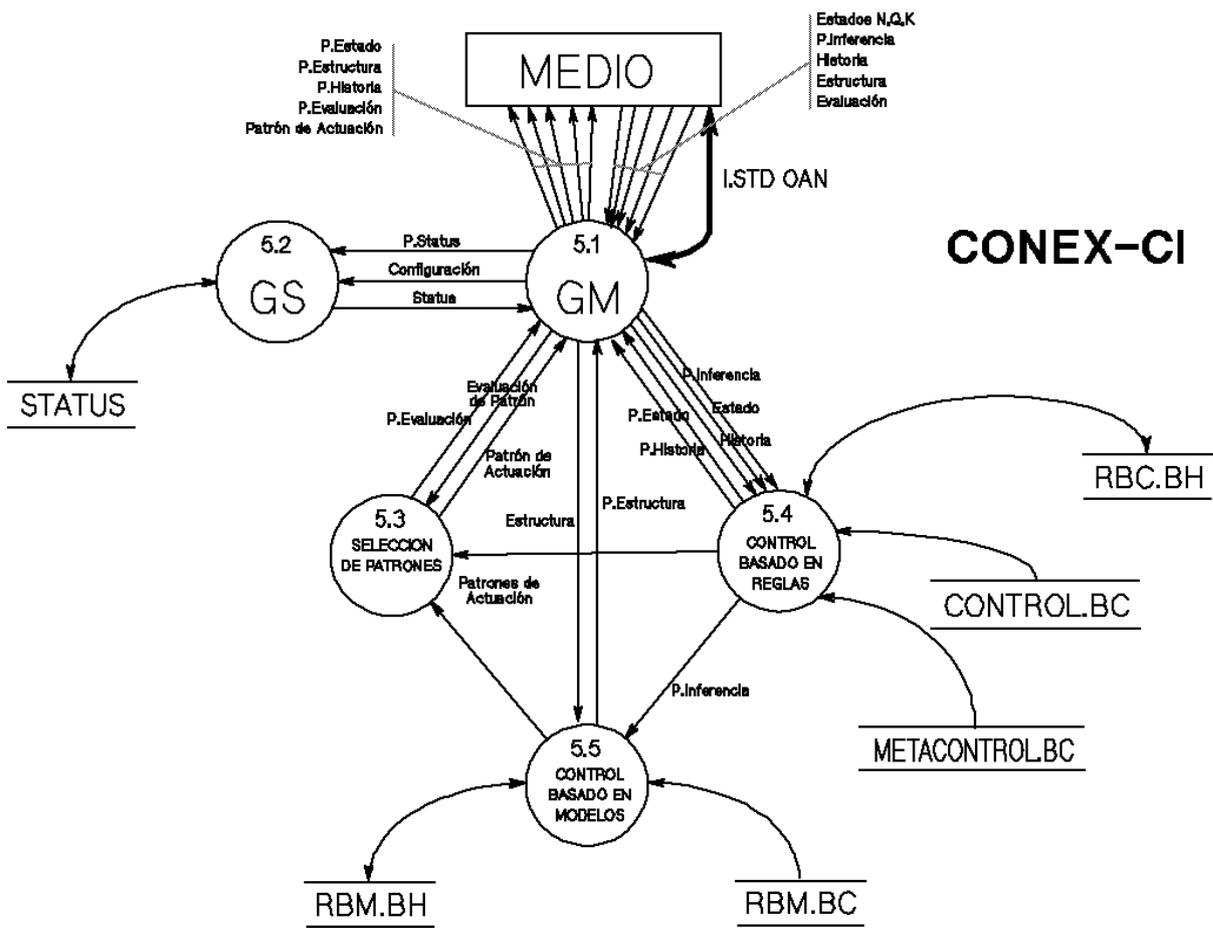


Figura 7.12: CONEX-CI: Control Inteligente

- La **base de conocimiento de metacontrol**. Que es independiente del proceso controlado.
- La **base de conocimiento de control** específica del proceso.
- El **modelo**, que es también específico.
- La **base de conocimiento para control basado en modelos**.
- Los hechos, procedentes de CONEX-MS.

La determinación de patrones de actuación puede concluir en un conjunto de varios patrones posibles, o en patrones singulares pero con baja confianza, por lo que un elemento importante del controlador inteligente es el evaluador de actuaciones, que permitirá determinar la bondad de un determinado patrón de actuación.

Técnicas

- Inteligencia artificial.
- Sistemas expertos.
- Razonamiento temporal.
- Programación orientada a objeto.
- Redes.

7.4.9. CONEX-IU: Interfase de Usuario

Misión

CONEX-IU es la interfase de usuario de CONEX. Sirve de medio de comunicación con el operador y con el ingeniero de planta.

Descripción general

Es el sistema encargado de la comunicación con los usuarios del sistema, tanto operadores como ingenieros de planta e ingenieros de sistemas. El sistema debe disponer por ello de interfaces múltiples/usuarios múltiples:

- Menús/Gráficos.

- Lenguaje natural.
- Intérprete de comandos.

A pesar de que un sistema experto de control mejora sobremanera la robustez del control, las razones de seguridad se seguirán imponiendo, y la presencia del operador seguirá siendo necesaria. Por ello es preciso que la interfase de operador sea mejorada de forma sustancial respecto a las ahora existentes. La ventaja de este tipo de sistemas se presenta no sólo en la dirección SISTEMA→OPERADOR, sino también en la contraria, ya que este OAN realiza una serie de funciones de utilidad para la robustez y operabilidad del sistema. Entre ellas destacan:

- Interpretación de datos de bajo nivel para la presentación de los mismos.
- Validación de las órdenes del operador, si es preciso solicitando evaluaciones a CONEX-EA.
- Gestión de diálogos en lenguaje natural restringido.

La posibilidad de manejar contextos múltiples debe permitir al sistema contestar a preguntas de la forma ¿Que pasaría si... ?. Además la estructura de conocimiento multinivel permite que el razonamiento en estos contextos se desarrolle a diferentes niveles de abstracción, con la consiguiente diferencia de costo computacional.

Entre los elementos fundamentales que constituyen esta interfase cabe destacar tres:

- El módulo de comunicación con el operador, basado en lenguaje natural y con un sistema de gestión de diálogos orientado a las tareas de control.
- El módulo de supervisión de actividades del operador. Este utiliza a CONEX-EA para determinar los efectos, en términos de objetivos globales, que van a tener sobre el sistema las acciones ordenadas por el operador.
- El módulo de interpretación cualitativa de datos de bajo nivel. Permitirá presentar la información sobre el proceso al usuario de forma

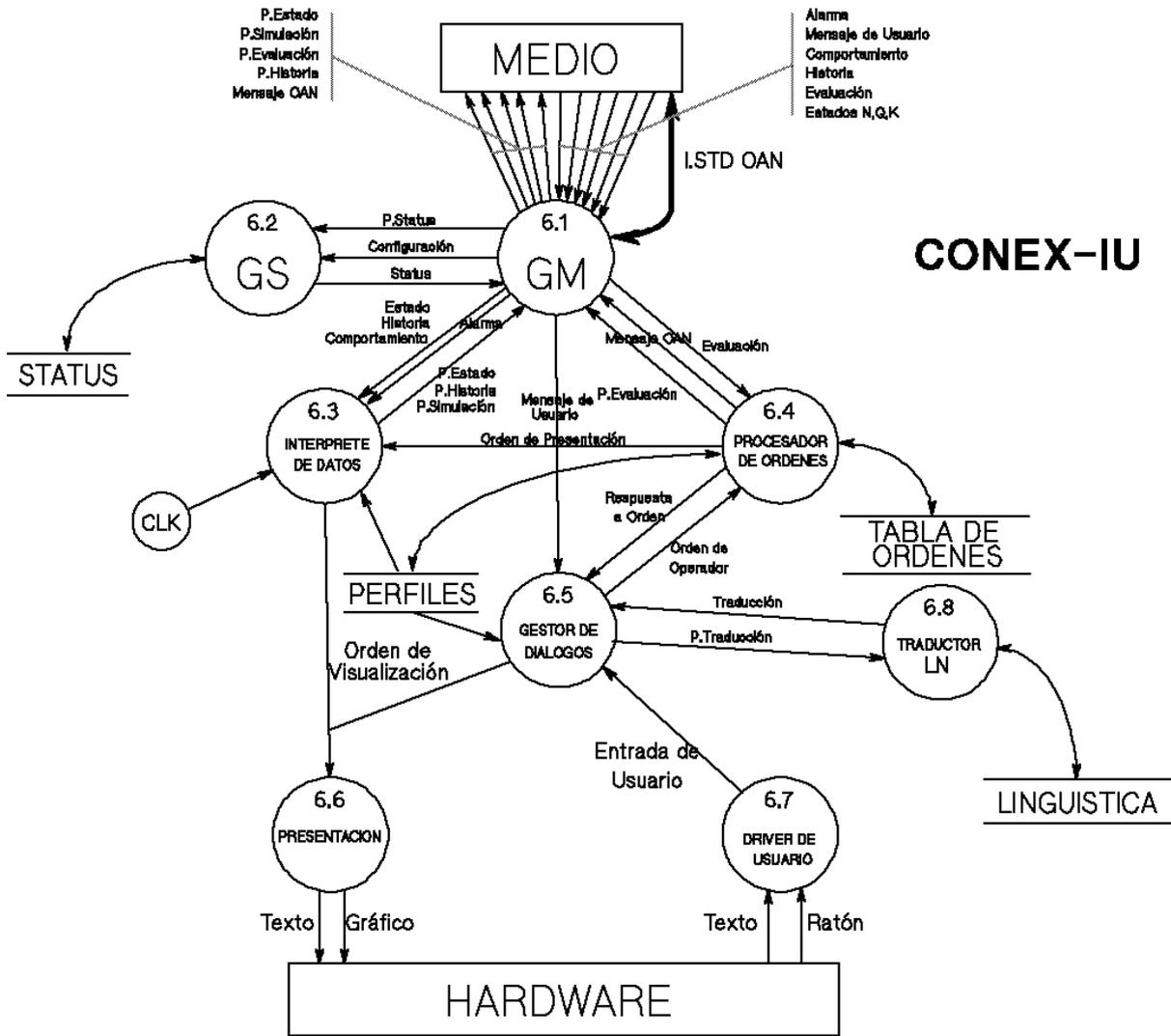


Figura 7.13: CONEX-IP: Interfase de Usuario

que se minimize la sobrecarga cognoscitiva de éste.

El sistema de comunicación con el operador se encargará de manejar el diálogo con el operador. Para ello utilizará tres modos de comunicación:

Gráficos: La interfase gráfica se basa en una representación esquemática del proceso. Mediante ella se puede acceder —con un mecanismo de posicionamiento (ratón, lápiz óptico, etc)— a diferentes elementos, y seleccionar las operaciones permisibles sobre ellos de un menú.

Lenguaje natural: Una interfase en lenguaje natural, restringido a actividades de control permitirá la gestión integral del sistema. La interfase gráfica debe permitir realizar las operaciones rutinarias, mientras que ésta debe dar soporte para todas aquellas operaciones que requieran de un conocimiento detallado del proceso a seguir.

Lenguaje de comandos: Se implantará también una interfase mediante un lenguaje de comandos para actividades de expertos en el sistema.

Todas las interfases realizarán una traducción a un lenguaje interno, del que la interfase basada en lenguaje de comandos será la representación externa. Este lenguaje interno es el que se empleará en las comunicaciones interprocesos.

Los datos procedentes del proceso son valores N, Q o K que se deben interpretar por el operador en base al conocimiento que el operador tiene sobre el proceso. Esta interpretación es fundamental para la operación correcta del proceso. Este subsistema utilizará la información estructural del proceso —que le suministra el modelo— para realizar un filtrado e integración de esa información de proceso.

La validación de órdenes del operador es fundamental desde dos puntos de vista:

- El operador puede cometer errores debido a sus errores en la valoración del estado del proceso.

- La comunicación en lenguaje natural puede conducir a interpretaciones erróneas del significado de las frases.

Estos problemas se resuelven utilizando dos tipos de mecanismos de seguridad:

Determinación de órdenes conflictivas . Una orden es conflictiva cuando su ejecución va en contra de los objetivos globales. La determinación de la conflictividad de una orden se hace en base a las características de la orden misma hay órdenes intrínsecamente conflictivas y a los efectos de la misma sobre el proceso. Esta determinación se hará en base a una petición de evaluación de actuación a CONEX-EA.

Confirmación de órdenes conflictivas . Las órdenes conflictivas serán denegadas o se pedirá confirmación de las mismas en función del tipo de operador. La confirmación se llevará a cabo presentado al operador la traducción a lenguaje natural de lo que ha entendido el sistema y pidiendo confirmación de que esa es efectivamente la orden.

Técnicas

- Lenguaje natural.
- Gráficos.
- Tutores inteligentes.
- Programación orientada a objeto.
- Redes.

7.4.10. CONEX-EA: Evaluador de actuaciones

Misión

CONEX-EA es el módulo de evaluación de actuaciones de control de CONEX. Una evaluación de una actuación es un juicio sobre los efectos de una determinada actuación de control en términos de objetivos de alto nivel. Esta evaluación se utilizará para determinar si una actuación de control determinada es aceptable o no.

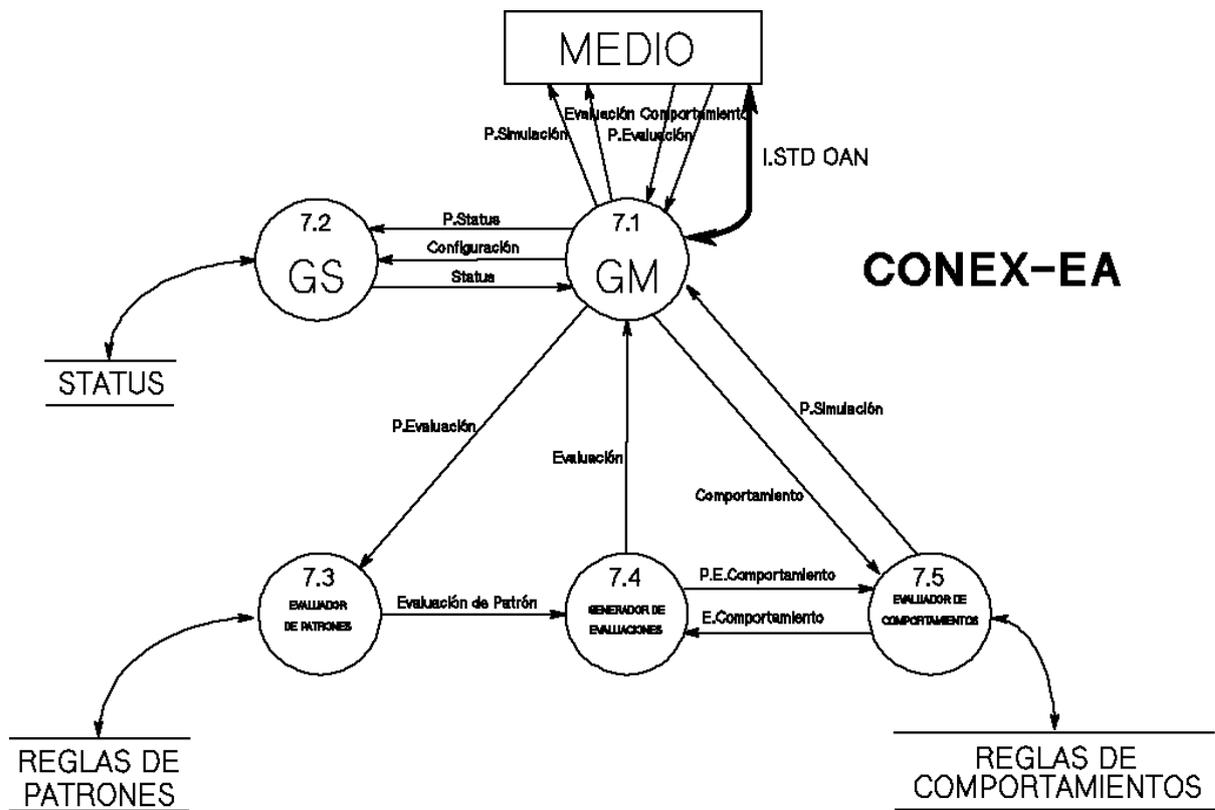


Figura 7.14: CONEX-EA: Evaluador de actuaciones

Descripción general

Este OAN recibe peticiones de evaluación de patrones de actuación por parte de diferentes OANs. Los principales clientes de este OAN son:

- CONEX-CI, ya que sus técnicas de determinación de actuaciones de control en particular el razonamiento basado en modelos pueden conducir a actuaciones inadecuadas.
- CONEX-IU, para determinar los efectos de las actuaciones de control sobre el proceso y validar así las órdenes del operador.

La evaluación de dichos patrones se realiza fundamentalmente en dos etapas denominadas evaluación a priori y evaluación por simulación. Estas dos etapas se llevan a cabo en base a:

- Un conjunto de reglas ad hoc de evaluación, que permiten identificar a priori actuaciones de control inadecuadas. Este tipo de actuaciones tendrán su origen fundamentalmente en el operador o en el control basado en modelos, ya que el control basado en reglas no tiene por que generar actuaciones inadecuadas a priori.
- Mediante peticiones de simulación y análisis del comportamiento simulado del proceso. El análisis y evaluación del comportamiento se hará en base a mecanismos procedurales de determinación de características y reglas de evaluación de comportamientos en base a dichas características.

La evaluación del patrón se realizará en función de la satisfacción de los objetivos globales de CONEX: Seguridad, estabilidad, calidad, producción. Esta satisfacción de objetivos depende del estado de la planta, ya que no es lo mismo una acción de control durante el funcionamiento normal que durante un proceso de parada. La información que permite identificar estos diferentes *estados de control* se encuentra contenida en la parte K del modelo de la planta.

CONEX-EA admitirá dos tipos de peticiones de evaluación por los restantes OANs:

- Evaluaciones singulares: en las que se evaluará un sólo patrón de actuación. El resultado de la evaluación es un informe de evaluación, ponderando la satisfacción de los objetivos globales.
- Evaluaciones múltiples: en las que se seleccionará la mejor alternativa. El resultado será un informe de evaluación para cada alternativa así como un consejo sobre cual es la más adecuada para los objetivos de control actuales.

Los criterios de evaluación son susceptibles de reformulación durante el funcionamiento de CONEX, de forma que se puedan replantear los objetivos de conducción mientras se realiza esta.

Técnicas

- Programación en tiempo real.
- Programación orientada a objeto.
- Sistemas expertos.
- Procesamiento de señales.
- Reconocimiento de patrones.
- Redes.

7.4.11. CONEX-MC: Monitor de CONEX

Misión

CONEX-MC es el módulo de monitorización y control de CONEX.

Descripción general

CONEX es un sistema software complejo, por lo que son necesarias herramientas para su gestión y operación. CONEX-MC es esta herramienta, que permitirá conocer el estado del sistema y realizar el control del mismo.

Entre las misiones principales que debe realizar CONEX-MC destacan las siguientes:

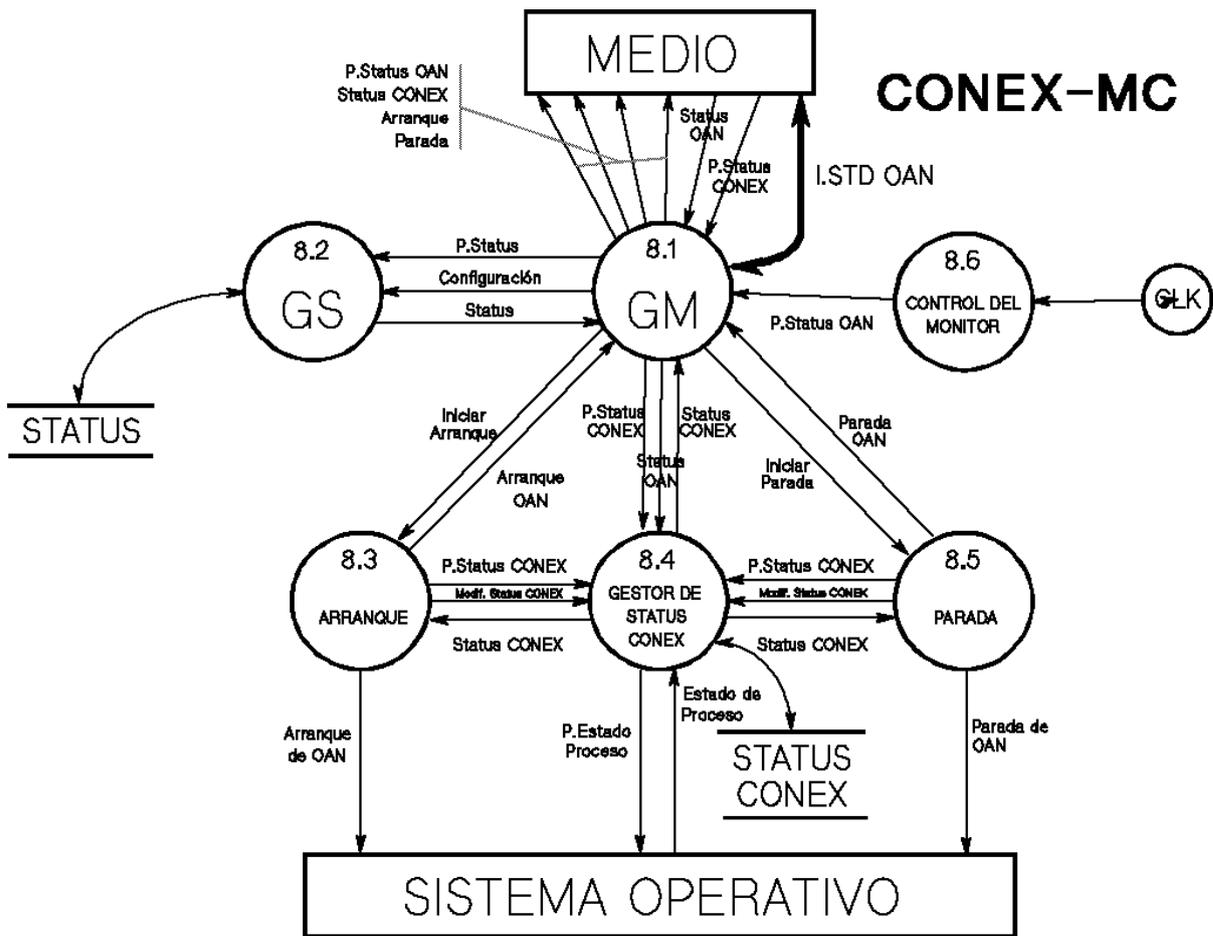


Figura 7.15: CONEX-MC: Monitor de CONEX

- CONEX-MC realiza peticiones de status a todos los OANs generando una descripción global del estado de CONEX.
- Monitoriza las comunicaciones, haciendo estadísticas que permitan diseñar mejoras al sistema.
- Realiza el arranque y la parada de CONEX. Lo reconstruye desde un caída del sistema.
- Posibilita la monitorización y el control remoto de CONEX.

Toda la información sobre el estado de CONEX se almacena en un fichero del status.

Los status de los OANs se reciben por CONEX-MC en base a listas de clientelas de status implantadas por cada uno de los OANs. CONEX-MC es cliente de todas ellas por lo que cualquier modificación de los status de los OANs resulta en una remisión de un mensaje de status del OAN en cuestión a CONEX-MC.

Técnicas

- Programación en tiempo real.
- Programación orientada a objeto.
- Redes.

7.4.12. CONEX-IE: Interfase Externa

Misión

CONEX-IE es la interfase externa de CONEX, sirviendo de medio de comunicación con elementos externos a CONEX.

Descripción general

CONEX-IE proporciona un medio de acceso desde el exterior a los OANs de CONEX, mediante un interfase externo estandarizado. Este acceso desde el exterior estará controlado mediante listas de autorización.

La conexión de salida precisará del desarrollo de módulos de interfase con cada sistema externo.

Técnicas

- Estándares de comunicaciones.
- Programación en tiempo real.
- Programación orientada a objeto.
- Redes.

7.5. Arquitectura Hardware

El sistema está compuesto por una serie de OANs que intercambian mensajes. Aunque esta arquitectura es independiente de la máquina, el elemento crítico de la misma es como se realizará este intercambio de mensajes. El medio mas claro de realizar el soporte de este mecanismo es mediante un sistema de red.

La red proporciona un medio de intercambio común entre todos los OANs, lo que tiene una serie de ventajas:

- Simplicidad de desarrollo ya que el acceso a los medios de comunicación es estándar, por lo que todos los objetos harán uso del mismo mecanismo, lo que facilita y reduce el tiempo de desarrollo.
- Simplificación de las comunicaciones, como es el caso de los mensajes globales, que con un sistema de red tiene la misma complejidad que los mensajes individuales.
- Adecuación HW/SW, dado que todas las comunicaciones se realizan a través de la red el único requerimiento de un OAN es tener acceso a dicha red, por lo que puede residir en cualquier máquina, de forma que la distribución de OANs se puede hacer de la manera mas eficiente para obtener un máximo rendimiento de los recursos hardware.
- Facilita el desarrollo de arquitecturas tolerantes a fallos, al permitir de forma simple la redundancia de equipos y procesos.

La optimización de funcionamiento del sistema conduce a la necesidad de implantar una *capa de comunicación CONEX (CCC)* situada entre los

OANs y el software de red de la máquina de forma que se acelere la interacción entre OANs de la misma máquina.

7.6. Valoración de la Arquitectura

En este capítulo se ha presentado una arquitectura de control que presenta niveles de operación diferenciados. Es una arquitectura jerárquica en la que se pueden apreciar tres grandes niveles de operación:

- Nivel de control directo.
- Nivel de control experto basado en representaciones avanzadas de conocimiento.
- Nivel de control supervisor por parte de un operador humano.

Como vemos el sistema es en realidad un *sistema de control supervisor con dos bucles de supervisión superpuestos*. Esta estructura está derivada de la del control supervisor clásico, en la que el bucle supervisor se descompone en dos: uno realizado por el humano —análogo al existente— y otro obtenido por extracción e implantación en un computador de determinadas funciones realizadas por el humano.

El sistema de control propuesto se basa en una arquitectura distribuida, tanto funcional como espacialmente. Desde el punto de vista funcional la arquitectura es identificable como un sistema *blackboard*, en el que diversos elementos con funciones diferenciadas interactúan a través de un depósito común de información. Los diferentes subsistemas se construyen utilizando programación orientada a objeto, y la comunicación entre ellos se lleva a cabo a través del *blackboard* o mediante paso directo de mensajes.

En la actualidad existen sistemas ya desarrollados correspondientes a cada uno de los módulos empleados en CONEX, sin embargo de la integración de todos ellos se espera conseguir un efecto de sinergia que aumente sobremanera la potencia del sistema completo.

El enfoque modular —funcional y físico— propuesto se ha diseñado de forma que posibilite una serie de puntos:

- Desarrollo independiente de los subsistemas.
- Escalabilidad.
- Fiabilidad, disponibilidad y mantenibilidad.

7.6.1. Desarrollo independiente de los subsistemas

Esta arquitectura posibilita el desarrollo independiente de los subsistemas. Para ello es preciso, no obstante un etapa inicial de análisis y especificación de las interfases entre los mismos, tanto a nivel hardware como software.

Una vez establecidos las interfases se puede independizar completamente el desarrollo, pudiendo incluso realizarse versiones múltiples de los subsistemas, que se utilizarán después como redundancias o entre las que se seleccionará la más idónea.

7.6.2. Escalabilidad

La potencia y aplicabilidad del sistema radica esencialmente en la implantación de funciones hasta entonces realizadas por el operador. Los requerimientos de potencia de cómputo para el desarrollo de ese tipo de actividades son difíciles de establecer a priori.

Además del éxito de una primera y restringida aplicación de un sistema de este tipo puede conducir a una ampliación de los objetivos de control, en el sentido de automatizar la operación del proceso en situaciones previamente gestionadas de forma manual. Estos dos factores —impredecibilidad de la potencia necesaria y ampliación de objetivos— hacen necesaria la escalabilidad del sistema de control.

El uso de una arquitectura *blackboard*, con paso de mensajes, sobre una red de área local posibilita la inclusión inmediata de nuevos subsistemas, que incorporen nuevas funciones o que proporcionen potencia de cómputo adicional.

7.6.3. Fiabilidad, disponibilidad y mantenibilidad

Los objetivos RAM, fiabilidad, disponibilidad y mantenibilidad son más asequibles dentro del marco de un sistema modularizado, sobre todo cuando se ha utilizado un enfoque de programación orientada a objetos [Sanz 89].

La fiabilidad del sistema es:

- Por una parte mejorable en desarrollo, ya que la verificación del software orientado a objetos es mucho más factible. Al estar los interfaces de cada clase perfectamente definidos la verificación se reduce a un método, que será por lo general simple.
- Por otra es intrínsecamente mayor, ya que el fallo en uno de los subsistemas no implica necesariamente el fallo en los demás, aunque existirán una serie de dependencias que limitarán la funcionalidad de dichos subsistemas.
- Por otra mejorable en operación, sin más que introducir nuevos subsistemas redundantes y seguir políticas de votación.

La disponibilidad del sistema es mayor, ya que la modularización y la orientación a objeto facilitan la detección y reparación de los problemas.

La mantenibilidad es con mucho el factor que se ve más potenciado con el uso de programación orientada a objeto. La independencia y reutilizabilidad del código es uno de los factores más característicos de este tipo de programación. La funcionalidad de un subsistema es incrementable sin que esto suponga alteración de los restantes subsistemas, siempre que las interfaces entre los mismos se respeten.

Capítulo 8

Control de Hornos de Clinker

Yo aún diría mas ...
Hergé
Tintín y los Pícaros, 1976

de minerales que en el producto final, el cemento, reaccionan con el agua formando cristales de gran dureza.

8.1. Introducción

En este capítulo se presenta un bosquejo de una aplicación basada en la arquitectura CONEX: el sistema C2G. El objetivo que se pretende conseguir con este desarrollo es dual:

- Objetivo industrial: desarrollar un sistema de control avanzado para plantas de producción de cemento.
- Objetivo técnico: determinar la aplicabilidad de dicha arquitectura de control a un proceso continuo que ha presentado tradicionalmente problemas de control.

8.2. Características del proceso

El proceso que se pretende controlar es un subproceso de fabricación de cemento: el horno. Este subproceso es, con diferencia, el que presenta más problemas para su control.

8.2.1. Plantas de producción de cemento

El cemento se produce en plantas de producción continua por medio de reacciones de descarbonatación y sinterización que generan una serie

Este proceso productivo tiene un gran consumo energético, dado que hay que realizar procesos físicos que requieren gran cantidad de energía y procesos químicos que requieren alta temperatura.

El proceso general de producción es el siguiente:

1. Preparación del material: Extracción del material —fundamentalmente caliza— de canteras, trituración y mezcla para obtener una composición suficientemente homogénea. Al material obtenido se le denomina prehomo.
2. Molienda de crudo y homogeneización: La prehomo se muele en un molino de bolas para obtener un material pulverulento denominado crudo. Este material se somete a un proceso de homogeneización en silos para mejorar la estabilidad de la composición. Al material de salida se le denomina crudo.
3. Cocción: El proceso de cocción permite realizar las reacciones químicas necesarias en el crudo para que se produzcan los minerales que debe tener el cemento. El material de salida del horno se le denomina clinker.
4. Molienda de cemento: El clinker, en forma de bolas en torno a 1 cm de diámetro es molido en un molino similar al utilizado para el crudo. El resultado de la molienda es el cemento, que se ensaca o se introduce en silos para su venta a granel.

De todos los procesos mencionados anteriormente, el horno es el que presenta más problemas para su control. De hecho los restantes procesos están bastante automatizados. En particular el autor ha desarrollado un sistema de control convencional de aplicación a molinos, tanto de crudo como de cemento. Este sistema ha sido objeto de ulterior desarrollo por parte de una empresa del sector y se comercializa en la actualidad como un paquete cerrado.

8.2.2. Hornos de Cemento

En una planta de producción de cemento el horno es, sin duda, el punto más crítico y en muchas de ellas el cuello de botella que limita la producción.

Los hornos de clinker son de varios tipos, utilizando varios tecnologías de cocción diferentes, pero la que presenta mayor importancia es la denominada de vía seca, ya que es la más adecuada desde el punto de vista energético-económico.

En este tipo de proceso la cocción del crudo para producir el clinker se realiza en tres etapas:

1. Un **intercambiador**, en el que se realiza un precalentamiento y una descarbonatación parcial del material utilizando gases de etapas posteriores.
2. El **horno** propiamente dicho, en el que se finaliza la descarbonatación y se realiza la cocción del material.
3. Un **enfriador**, en el que se realiza una reducción rápida de temperatura del material cocido.

Estas tres etapas están fuertemente acopladas, ya que en ellas se tienen dos flujos opuestos: uno de material sólido y otro de gases en contracorriente, lo que dificulta enormemente su control. Además en este tipo de procesos suele emplearse precalcinación, que es una inyección y quema de combustible en el intercambiador, de forma que parte del proceso químico se produzca en esta etapa.

El proceso que sufre el material es el siguiente. El material crudo pulverizado entra en el intercambiador, donde se calienta utilizando los gases de salida del horno hasta una temperatura elevada, produciéndose una descarbonatación parcial. El material entra en el horno donde sufre dos transformaciones. La primera supone la conclusión de la reacción de descarbonatación. La segunda reacción es una reacción de sinterización, en la que se generan los minerales que componen el clinker. La reacción de descarbonatación es endotérmica, por lo que precisa de aporte de energía; sin embargo la de sinterización es exotérmica, aunque precisa de una alta temperatura para que tenga lugar. La temperatura adecuada se logra por medio de un combustible —habitualmente carbón pulverizado— que se quema en una corriente de aire —aire primario— que se inyecta en el horno. El material sale del horno a una temperatura cercana a los 1000°, siendo preciso un enfriamiento rápido del mismo en la tercera etapa, el enfriador. En esta etapa se emplea aire frío —a temperatura ambiente— para enfriar los gránulos de clinker. El enfriamiento debe ser rápido para que el clinker tenga las propiedades químicas adecuadas para el cemento y físicas que faciliten la molienda. A este enfriamiento rápido se le denomina templado. El aire empleado en el enfriador pasa en parte al horno, recibiendo la denominación de aire secundario. También es posible utilizar parte de este aire caliente en el intercambiador, lo que se conoce como aire terciario.

8.2.3. Problemas de control

El control de este proceso es complejo por los siguientes aspectos:

- No es un sistema simple sino que son tres sistemas fuertemente acoplados: intercambiador, horno, enfriador.
- No existen buenos modelos del proceso, ya que es un proceso con mucha variabilidad —sólidos en suspensión a altas velocidades por conductos rugosos.
- La medición de variables es difícil debido a las condiciones de temperatura, polvo, erosión, etc.

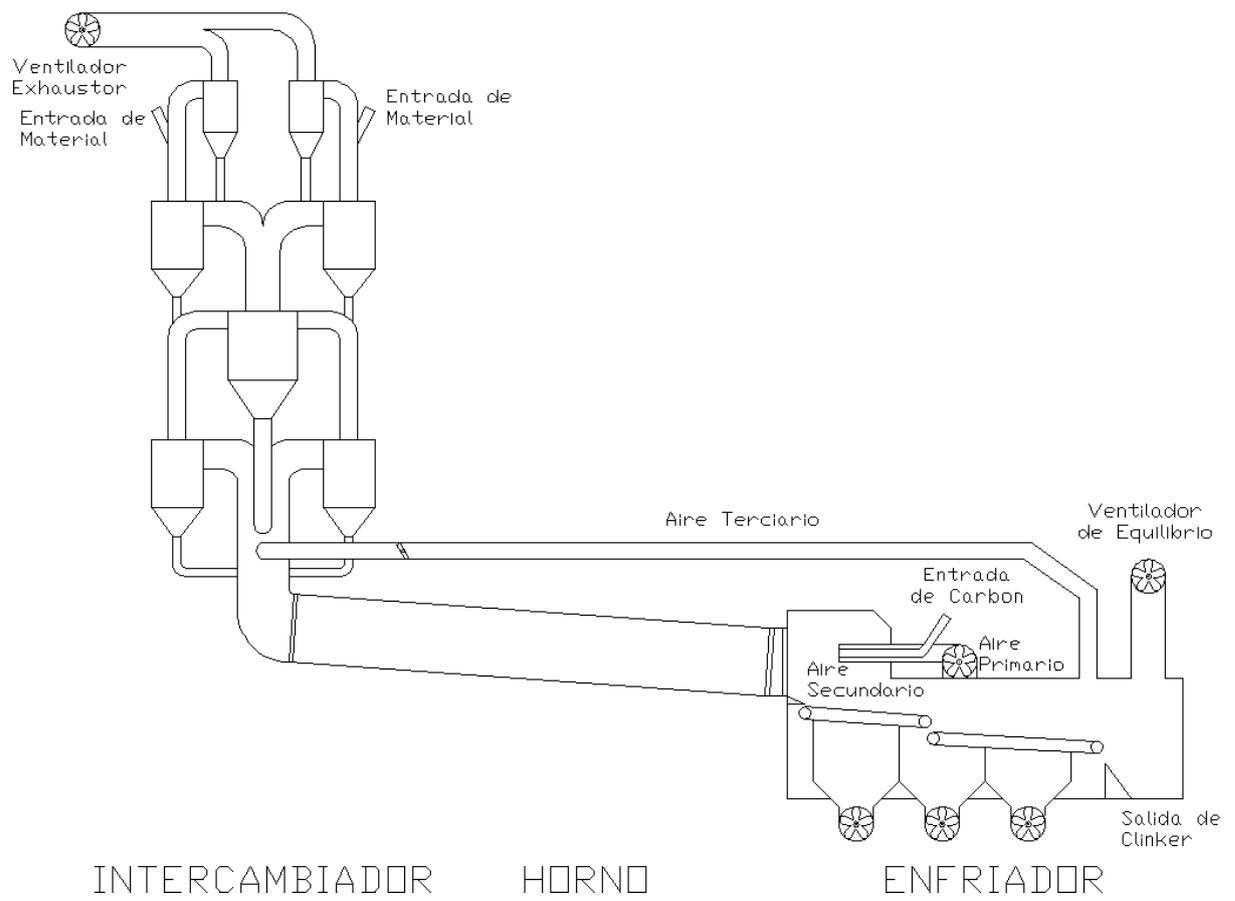


Figura 8.1: Estructura básica de un horno de clinker para fabricación de cemento.

- La reacción en el horno es asimétrica. En condiciones de funcionamiento estacionarias la reacción de sinterización proporciona parte del calor que precisa la de descarbonatación, por lo que una pequeña caída de temperatura en la zona de sinterización produce un efecto en cadena que puede *apagar* el horno.

8.2.4. Desarrollos actuales

En la actualidad existen múltiples sistemas de conducción automática de hornos. Los sistemas clásicos han dado resultados más bien pobres, siendo los sistemas basados en el conocimiento los que han obtenido mejores resultados. Entre ellos cabe destacar:

- Los sistemas FUZZY de F.L. Smidth y Link-Man de Sira Ltd. son sistemas de control basado en reglas borrosas.
- Los sistemas KALHAS y FAROS, desarrollados por el grupo cementero griego Heracles.
- El sistema experto de control de hornos de Industar.

Los resultados obtenidos con estos sistemas han sido moderadamente buenos, aunque precisan de una supervisión prácticamente continua por personal especializado para su funcionamiento. Consiguen operación bastante estable gestionando un número muy limitado de circunstancias anormales.

8.3. Sistema de Control de Hornos

La aplicación que presenta de manera somera en este capítulo se está desarrollando en la fecha en que esta tesis ve la luz. Esta aplicación es un desarrollo conjunto entre una empresa del sector y el departamento en que se he realizado este trabajo. Esta aplicación tiene como objetivo el control del subproceso más crítico de una planta cementera:

el horno. El sistema de control inteligente de hornos se integrará con los sistemas de automatización actualmente existentes en la planta, haciendo uso de los mecanismos que éstos provean.

8.3.1. Objetivos

El objetivo fundamental que se persigue con este desarrollo es el de obtener un sistema que permita la conducción automática de hornos con una disponibilidad cercana al 100 %.

Esto supone que el sistema debe manejar una serie de circunstancias anormales que suceden en los hornos de clinker pero que no impiden su correcto funcionamiento. Estas circunstancias se derivan de las condiciones especiales del proceso: variabilidad de materiales de partida, temperaturas críticas, fuertes erosiones, etc.

8.3.2. Arquitectura Software

La arquitectura software del sistema es la arquitectura CONEX, sobre la que se realizan particularizaciones para este proceso.

El interfase de proceso es un punto crucial en esta aplicación, dado que el proceso tiene todas las características no deseadas que lo hacen necesario:

- Ruidos: El accionamiento de la mayoría de los sistemas de proceso se realiza por medio de motores eléctricos de gran potencia, que inducen ruidos eléctricos.
- Variabilidad sensorial. El proceso se lleva a cabo en una suspensión de sólidos en gases, con flujos a altas velocidades que ocasionan turbulencias que dificultan las medidas estables.
- Anomalías de funcionamiento de sensores, debido a las condiciones de medida. Las altas temperaturas hacen que se formen pegaduras en los termopares, lo que impide una medida correcta, ya que aumenta la inercia térmica de la sonda.
- Deterioro de sensores. Los materiales en suspensión a altas velocidades son muy abrasivos.

vos, por lo que los elementos sensoriales sufren unos desgastes importantes.

- Variables ocultas. Las condiciones del proceso hacen muy difíciles las medidas, cuando no imposibles. La medida de temperatura en la zona de sinterización es quizá la más importante de todas, pero sin embargo es prácticamente imposible realizara con un mínimo de garantías. La temperatura en dicha zona ronda los 1400° centígrados, por lo que es prácticamente imposible situar ningún elemento de medida. Los pirómetros ópticos se ven muy afectados por el polvo, sobre todo en procesos en vía seca, siendo su medida de muy baja calidad. Es preciso realizar una obtención indirecta de dicha temperatura a partir de otras como son el peso por litro del clinker, las concentraciones de NO_x en las salidas de gases o el par del accionamiento del horno.

Los sistemas de control de bajo nivel se encargan del control de varios bucles simples en el sistema. Estos controles directos se implantan mediante PIDs en los más simples, control adaptativo en otros con más problemas como es el caso del control de velocidad de las parrillas del enfriador, y control basado en reglas borrosas que se emplea en los casos mas complejos, como es el del propio horno.

El monitor se basa en patrones establecidos por el experto de proceso y particularizados para la planta por los ingenieros y operadores de la misma. Estos patrones son esencialmente pocos, ya que las situaciones anormales en que se debe seguir en operación o que se pueden recuperar son bastante pocas.

Las reglas de alto nivel se extraen de un experto de proceso, así como de expertos de la planta particular para la que se desarrolla la aplicación. El sistema experto de control basado en reglas se implanta por medio de una herramienta comercial de desarrollo de sistemas expertos de control. Esta herramienta, G2, proporciona unos mecanismos de construcción adecuados a plantas de parámetros concentrados, por lo que es preciso realizar un desarrollo compartamental del modelo. El razonamiento basado en modelos se aplica fundamentalmente en modo no automático, sugiriendo

al operador alternativas de control en una situación dada. Este módulo de razonamiento se desarrolla en un lenguaje independiente, orientado a objetos, para facilitar su reutilización y adaptación a otros sistemas.

El modelo se estructura en los tres niveles de CONEX, realizándose el modelo N en términos de ecuaciones diferenciales según el formalismo empleado por un sistema de simulación convencional (ACSL).

El simulador para esta aplicación está dividido en los niveles Q y N. No se hace uso de simulación basada en el conocimiento. El simulador N es en este caso el que lleva la mayor parte de la carga de evaluación dado que este simulador se desarrolla a partir de versiones anteriores desarrolladas por el mismo equipo con lo que se espera lograr muy buenos resultados. El simulador Q tiene como misión principal su uso en la interacción con el operador.

Los servicios de cara al operador que suministra esta aplicación no se reducen a una mera presentación de datos, sino que el sistema efectivamente validará las órdenes del operador en base a una valoración de los efectos de las mismas. El módulo de lenguaje natural es muy reducido, limitándose a una gramática sencilla y a un léxico reducido al tipo de aplicación. No se pretende desarrollar el interfase hasta sus últimas consecuencias porque la necesidad de intercambio de información entre el operador y el sistema de control es limitada.

8.3.3. Arquitectura Hardware

El sistema corre sobre dos máquinas conectadas en red. Una de ellas es la encargada de soportar las tareas de alto nivel —interfase de operador, razonamiento basado en modelos, razonamiento basado en reglas— mientras que la otra soporta la comunicación con el proceso y las tareas de bajo nivel —interfase de proceso, monitor, control directo. De este modo el sistema puede operar como un sistema de control mas o menos clásico cuando aparezcan circunstancias que impidan la operación de la máquina de alto nivel. Estas circunstancias estarán asociadas al software en dos

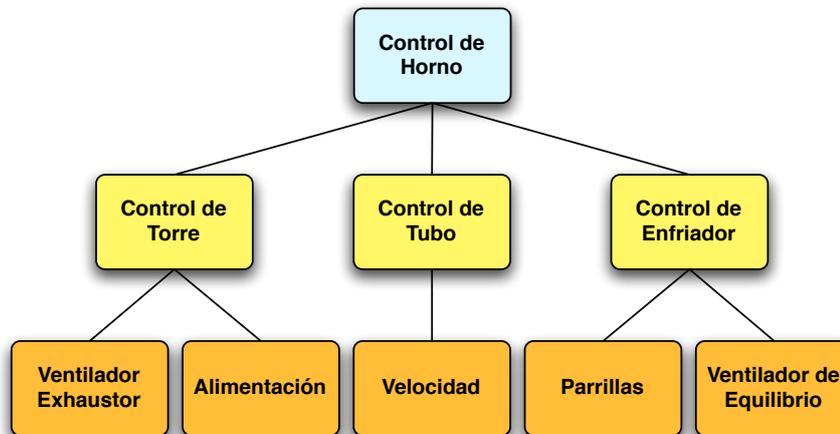


Figura 8.2: Parte de la jerarquía de controles del horno de cemento.

formas:

Problemas: ya que pueden aparecer errores software, mas difíciles de detectar en las aplicaciones de alto nivel por ser aplicaciones basadas en inteligencia artificial.

Mantenimiento: Ya que la aplicación debe ser actualizada, dado que las plantas no son estáticas, sino que sufren modificaciones casi continuas y además el conocimiento que se obtenga de la planta a través del uso del sistema permitirá mejorar el conocimiento implantado en el sistema de control inteligente.

Los equipos elegidos para estas dos funciones son dos estaciones de trabajo VAXstation 3100. La conexión entre ambos equipos se realiza por medio de una red Ethernet, dado que la temporalidad del proceso y la baja carga de la red no obligan a utilizar una red determinista como ARCNET.

El sistema se implanta sobre los sistemas de control previos de la planta, realizando la comunicación con ellos de acuerdo con las prestaciones de los mismos. Para sistemas de control supervisor modernos se empleará comunicación mediante red, mientras que para sistemas convencionales basados en PLCs se emplearán líneas serie dedicadas.

8.4. Resultados esperados

Los resultados que se esperan de la introducción de este sistema de control en la planta son los siguientes:

- Un aumento de la estabilidad de operación que redundará tanto en los resultados que se comentan a continuación como en una disminución de los problemas de mantenimiento.
- Un aumento de la calidad del producto, ya que se mantendrán mas controlados los aspectos que la determinan, en particular la temperatura de la zona de sinterización.
- Un aumento de la producción del horno, ya que el funcionamiento estable permitirá la conducción del horno en condiciones más cercanas al límite.
- Una disminución del consumo de combustible, ya que el control continuo garantizará la estabilidad cerca del límite térmico, por lo que se reducirá la necesidad de llevar el horno sobrecalentado para evitar caídas de temperatura y apagado de horno.
- No se pretende conseguir una reducción de personal —al menos de control— dado que las características del proceso no permiten

garantizar completamente un funcionamiento autónomo. Además el sistema precisará de personal especializado para su mantenimiento y actualización.

Capítulo 9

Conclusiones

Los sistemas inteligentes de control se reducen en la actualidad a dos tipos: sistemas que están controlando algún proceso industrial y sistemas que no lo están haciendo. En ambos casos hay una serie de características generales que merece la pena destacar:

Sistemas de control operativos: Son aplicaciones mas o menos directas de la tecnología de sistemas expertos, con modificaciones orientadas a su funcionamiento en tiempo real. Este enfoque está esencialmente basado en reglas, pudiendo incluir específicamente técnicas más avanzadas como pueden ser el uso de técnicas borrosas o razonamiento no monotónico.

Sistemas de control experimentales: En desarrollo en laboratorios de investigación, por lo general universitarios, en los que se manejan *mundos de juguete*, tratando de desarrollar técnicas utilizables en control real. Este tipo de sistemas implantan técnicas mucho mas avanzadas que los anteriores.

Cabe destacar el esfuerzo que se está realizando dentro de programas de investigación, en proyectos que tratan de desarrollar las metodologías avanzadas que se han venido comentando en esta tesis y de utilizarlas en aplicaciones reales.

9.1. Aportaciones de esta tesis

Las aportaciones de esta tesis se pueden plantear —en cierta medida— como una fundamentación teórico-práctica que permite el traslado de las

técnicas avanzadas a aplicaciones reales, ya que se presenta en ella una arquitectura de control orientada a la aplicación real de la misma —de hecho mientras se escribe esta tesis se está desarrollando una aplicación basada en dicha arquitectura.

Esta arquitectura supone un acercamiento a la solución de los siguientes problemas:

- *Realización efectiva de sistemas inteligentes de control en tiempo real*, en base a una descomposición y jerarquización funcional de sistemas de proceso de información de acuerdo con el principio de inteligencia creciente con velocidad decreciente. Ello hace posible utilizar tecnologías como la de sistemas expertos en aplicaciones de control de procesos que requieran de una elevada velocidad de operación.
- *La integración de paradigmas de control diferentes*, basados en diferentes metodologías de control, en base a la jerarquización funcional de los sistemas construidos con dichos paradigmas. La utilización de paradigmas diferentes es necesaria desde dos puntos de vista: a) la necesidad de operación en tiempo real, que obliga a utilizar métodos rápidos y b) la necesidad de manejar información abstracta, que obliga a emplear técnicas de inteligencia artificial.
- *La utilización de información de planta disponible en distintos niveles de abstracción*, ya la existencia de varios paradigmas de control permite hacer uso de todo tipo de información sobre el proceso. Esto es de importancia crucial, ya que los sistemas de captación trabajan en un nivel de abstracción prácticamente

mínimo mientras que las estrategias de control de situaciones anormales se plantean en términos de estados del proceso establecidos en términos muy abstractos.

- La *existencia de una fuente coherente de información del estado de la planta*, capaz de suministrar información en diferentes niveles de abstracción. Esta fuente está constituida por el modelo multiresolucional del proceso. Esta fuente suministra información sobre: a) el estado de la planta en diferentes niveles de abstracción b) la estructura de la planta para utilizarla en razonamiento basado en modelos c) la evolución histórica de la planta y d) predicciones sobre el comportamiento de la misma frente a acciones de control.
- La *integración de mecanismos procedurales junto con mecanismos de razonamiento simbólico*, que permite lograr operatividad en tiempo real e implantar conocimiento procedural sobre la operación del proceso.
- La *utilización de información de expertos en operación, en proceso y en control* de forma integral. Información que permite utilizar conocimiento operacional, estructural-funcional y procedural en el control del proceso. En sistemas expertos convencionales la información utilizada es exclusivamente la de los operadores —expertos en operación— cuando la información que suministran los ingenieros de proceso es fundamental para realizar una valoración correcta del estado de la planta.
- La *reutilizabilidad de sistemas de control adaptados a la planta*, ya que el conocimiento sobre el proceso está en forma de bases de conocimiento convencionales y en particular en forma de modelo multiresolucional.
- La *seguridad y fiabilidad del controlador*, que es de especial importancia en el caso de control basado en inteligencia artificial. El razonamiento basado en modelos, la evaluación de actuaciones y la integración del operador en el sistema permite aumentar la robustez del controlador basado en inteligencia artificial.

- La *apertura del sistema de control*, de forma que sea factible la comunicación desde y hacia él. Esta comunicación se basa en el mecanismo de comunicación entre objetos de alto nivel.
- La *facilidad de incorporación de nuevas funciones* a un sistema ya existente, ya que la estructura de los objetos de alto nivel, así como el mecanismo de comunicación entre los mismos permite el desarrollo de nuevos elementos software directamente integrables con una aplicación de control construida con la arquitectura CONEX.

9.2. Desarrollos futuros

Para la redacción de este capítulo final fué preciso analizar los posibles desarrollos futuros asociados a esta tesis, encontrándose el autor con una situación de horizonte infinito —dimensión de lo que queda por hacer.

Ante este estado de cosas —enormidad del trabajo a realizar— cabe preguntarse si el trabajo realizado ha sido suficiente. Es opinión personal que el formular el método de control a utilizar sirve de gran ayuda a la hora de dirigir los posteriores trabajos en el área, y de hecho, a raíz de los desarrollos llevados a cabo en esta tesis se está formando un grupo de trabajo en el área de un tamaño importante.

Intentaremos situar los trabajos llevados a cabo en esta tesis en el marco de los posibles desarrollos en este área, tratando de indicar que dichos trabajos han sido en parte *necesarios*, en parte *aconsejables* y en parte *exploratorios*.

Clasificaremos los desarrollos futuros derivables de esta tesis en tres grupos:

Necesarios: Son aquellos sin los cuales este trabajo no tiene ninguna utilidad. Entre estos desarrollos cabe destacar:

- Desarrollo de técnicas fundamentales de la arquitectura que posibiliten la implantación de la misma.
- Desarrollo de aplicaciones basadas en esta arquitectura, de forma que se genere

una realimentación que permita su mejora.

Aconsejables: Desarrollos que pueden resultar económicamente interesantes desde el punto de vista de el uso de este tipo e controles y desarrollos orientados en profundizar en la fundamentación teórica de este tipo de sistemas.

- Desarrollo de entornos de construcción de sistemas inteligentes de control. En la actualidad, a raíz de los desarrollos asociados a esta tesis se está llevando a cabo un estudio de la necesidad de desarrollar un entorno para desarrollo de aplicaciones de control basadas en esta arquitectura. En este entorno se incluirán desde elementos preconstruidos hasta herramientas de desarrollo.
- Construcción de motores de inferencia en tiempo real, tanto conclusivos como progresivos.
- Herramientas de simulación de sistemas continuos embebibles.
- Simulación cualitativa de sistemas distribuidos.

Exploratorios: Destinados fundamentalmente a incluir otras tecnologías en el campo de control inteligente.

- Utilización de redes neuronales para determinación del estado de la planta (etapa de *valoración de la situación*).
- Inclusión de mecanismos de aprendizaje para la mejora de modelos o el establecimiento de patrones.

En la actualidad se están llevado a cabo desarrollos en todos los grupos mencionados, bien en la forma de proyectos fin de carrera, bien como proyectos industriales y posiblemente como tesis doctorales.

Apéndice A

Criterios de aplicación del control inteligente en procesos industriales

Ante la complejidad del problema de decidir qué tecnología de control emplear, se nos plantea la pregunta ...

¿ES ESTE PROBLEMA DE CONTROL ADECUADO PARA USAR TECNICAS DE CONTROL INTELIGENTE?

El siguiente cuestionario proporciona algunas pistas para decidir sobre la adecuación o no del uso de un sistema inteligente de control en su caso particular.

¿Ha intentado controlar su proceso con medios clásicos habiendo fracasado, o ha obtenido resultados muy pobres?

Si [] No []

¿Tiene su proceso complejidad terciaria?

Si [] No []

¿El estabilizar la operación del proceso o aumentar ligeramente su rendimiento sería muy rentable?

Si [] No []

¿Hay alguien capaz de controlar el proceso?

Si [] No []

¿Está dispuesto a confiar en un sistema inteligente de control?

Si [] No []

Si se contestan afirmativamente a todas las preguntas el caso es adecuado para intentar el desarrollo de un sistema de control inteligente. Cada una de las preguntas trata de caracterizar uno de los aspectos fundamentales del control inteligente. Estos aspectos son los siguientes:

- Los sistemas de control inteligentes pueden hacer frente a problemas de control inabordables con enfoques clásicos.
- Permiten controlar a pesar de las dificultades de modelado del proceso.
- Su coste de desarrollo es muy elevado.
- Como todos los sistemas basados en el conocimiento precisan que alguien aporte este conocimiento.
- Su funcionamiento es de difícil comprensión, aún utilizando justificación.

Apéndice B

Glosario

En este glosario se recogen las acepciones con que se usan una serie de términos importantes en esta tesis. En algunos de ellos se incluye el término inglés del que procede en la forma I(XXX)¹.

B.1. Términos

Acantilado, efecto: Se denomina así a la disminución brusca de prestaciones de un sistema experto ante un problema que cae ligeramente fuera de su ámbito de aplicación. I(cliff effect)

Acción de control: Una acción de control es una valoración de una variable de actuación.

Activación periódica: Un mecanismo de activación de reglas mediante el cual una determinada regla se activa de forma periódica. Este tipo de mecanismo es muy importante en sistemas de control directo basado en reglas.

Activar: Activar una base de conocimiento es hacer que el conocimiento contenido en esa base de conocimiento se utilice en las siguientes inferencias. Es una de las posibles acciones que puede aparecer en el lado derecho de una regla.

Actividad de control: Dícese de una tarea de control directo: cambio de consignas, especificación de reguladores, ajuste de parámetros, etc. Las actividades pueden ser instantáneas o estar asociadas a intervalos temporales. Actividades especiales son la bifurcación de patrones y la actividad vacía.

Actuación: Establecimiento de acciones de control para el sistema de bajo nivel.

Análisis de sensibilidad: Proceso de determinar la variabilidad del comportamiento de un sistema debido a pequeñas perturbaciones en sus parámetros y/ entradas. I(sensitivity analysis)

Antecedente: Lado izquierdo de una regla.

Atributo: Cada uno de los campos de datos asociados a un objeto. Se utiliza habitualmente dentro de las representaciones objeto-atributo-valor. En programación orientada a objeto se habla de variable de ejemplar (instance variable). Cuando se emplean marcos se habla de ranuras (slots).

Automatización total: Un *nivel de automatización* en el que el humano no tiene ninguna función.

Base de conocimiento: Conjunto de información que recoge conocimiento sobre un determinado dominio. Habitualmente ha estado asociado a representaciones mediante reglas, aunque puede estar construída en base a cualquier otro paradigma de representación.

Biblioteca de conocimiento: Cuando se emplea un esquema de representación de conocimiento constructivista —por ejemplo modelos— es posible reutilizar las bases de conocimiento, por lo que estas se pueden almacenar en bibliotecas de conocimiento.

Blackboard: Tipo de estructura de un sistema de manejo de información, en el que varios procesadores de información toman y dejan información en un depósito común que se denomi-

¹El autor pide disculpas por las posibles discrepancias sobre la correcta correspondencia de términos, respecto a lo establecido por otros autores.

na blackboard. Se le ha denominado en ocasiones enfoque oportunista para la resolución de problemas.

Bucle externo: En esta tesis se refiere al bucle de *control* llevado a cabo por encima de otro bucle de control, con la misión de supervisar la actividad de éste. Por lo general se lleva a cabo mediante un humano.

Bucle interno: Bucle de *control* por debajo de un bucle supervisor y que es objeto de la actividad de éste.

Casamiento: Proceso de búsqueda de valores que encajan en una determinada expresión. I(matching)

Categoría focal: Grupo de reglas que se activan de forma separada. La descomposición en categorías focales permite dividir la base de conocimiento en bases que se seleccionan de acuerdo con el proceso de inferencia. El término *focal* tiene sentido sólo en relación con su uso para referirse a focalización espacial (ver clase focal).

Causalidad: Es un tipo de relación entre variables y cambios de las mismas que va a permitir realizar razonamientos sobre el funcionamiento de del sistema al más alto nivel.

Clase: Una clase es una estructura abstracta de datos, en la que incluyen tanto los datos propiamente dichos, como una serie de métodos asociados a los mismos; es uno de los conceptos básicos de la programación orientada a objeto.

Clase focal: Grupo de reglas que constituyen una base de conocimiento en un sistema descompuesto por dominios. Son reglas referidas a una determinada clase de objetos —objetos focales.

Complejidad de interacción: Uno de los tres aspectos de complejidad del proceso mencionados en esta tesis. Se refiere a los volúmenes elevados de interacciones entre los diferentes elementos del proceso.

Complejidad volumétrica: Uno de los tres aspectos de complejidad del proceso mencionados

en esta tesis. Se refiere al número elevado de elementos del proceso.

Comportamiento: El comportamiento de un sistema es la sucesión de estados del mismo. En ocasiones se le denominará *historia* o *evolución*.

Conclusión: Parte derecha de una regla.

Concurrente: Se dice que dos procesos son concurrentes cuando se ejecutan al mismo tiempo y realizan intercambio de información entre ellos.

Conocimiento declarativo: Conocimiento relativo a la estructura de algún objeto; es conocimiento de lo *que es*.

Conocimiento heurístico: Un tipo de conocimiento basado en la experiencia y fundamentado en el sentido común.

Conocimiento procedural: Conocimiento relativo a la manera de realizaa algo; es conocimiento de *cómo*.

Conocimiento profundo: Conocimiento que fundamenta en mecanismos causales el comportamiento de un sistema; en ocasiones se le denomina conocimiento causal.

Conocimiento somero: Conocimiento que expresa comportamientos sin dar razón de su causa.

Control: Sistema que permite modificar el comportamiento de otro sistema respecto a su libre funcionamiento. Por lo general se entenderá que el CONTROL trata de que las variables 'interesantes' del sistema sigan una determinada evolución.

Control clásico: Control llevado a cabo utilizando exclusivamente técnicas tomadas de la teoría de sistemas.

Control inteligente: Control que utiliza técnicas tomadas de la inteligencia artificial.

Control inteligente de procesos: Control inteligente aplicado a los sistemas industriales de producción continua.

- Control supervisor:** Control cuyo objetivo es que otro control —control supervisado— se comporte de manera adecuada.
- Demonio:** Proceso latente que realiza su función cuando se dan determinadas circunstancias sin ser requerido para ello por ningún otro proceso.
- Desactivar:** Ver *activar*.
- Diagnosis:** Proceso de identificación de una malfunción a partir de los síntomas y de la historia de un elemento.
- Dirigido por eventos:** Dícese del sistema de razonamiento que actúa como consecuencia de la aparición de determinados sucesos. I(event driven)
- Disponibilidad:** Cualidad o estado de estar disponible. Por lo general en el ámbito técnico se refiere a la cualidad de un sistema que permite su reparación y puesta en funcionamiento en caso de avería de forma rápida. I(availability)
- Ejemplar:** Cada uno de los objetos de una determinada clase. I(instance)
- Encadenamiento hacia atrás:** Una forma de inferencia en que se parte de los hechos a demostrar, buscando las reglas y premisas que los satisfagan.
- Encadenamiento hacia delante:** Una forma de inferencia en que se parte de los hechos para alcanzar las conclusiones.
- Encapsulación de datos:** Inaccessibilidad de los datos internos de un objeto salvo a través del interfase de éste.
- Enlace dinámico:** Resolución de una referencia a un método —procedimiento— en tiempo de compilación o de ejecución, en función de la clase del objeto que la recibe. Conceptos asociados son los de enlace temprano —resolución en compilación— y enlace tardío —resolución en ejecución. I(dynamic binding, early binding, late binding)
- Episodio:** Elemento temporal de duración no nula.
- Espacio de cantidades:** Conjunto de todos los posibles valores de una *variable*.
- Espacio de estados:** Conjunto de todos los posibles estados del sistema.
- Estado FQ:** Representa el conjunto de valores cualitativos borrosos de todas las variables del proceso susceptibles de tomar este tipo de valores.
- Estado K:** Representa el conjunto de valores numéricos de todas las variables del proceso susceptibles de tomar este tipo de valores.
- Estado N:** Representa el conjunto de valores numéricos de todas las variables del proceso susceptibles de tomar este tipo de valores.
- Estado Q:** Representa el conjunto de valores cualitativos de todas las variables del proceso susceptibles de tomar este tipo de valores.
- Estado:** Estado del proceso. Se compone de los estados K, Q y N. I(state)
- Estado:** Para un *sistema* dado utilizaremos el término ESTADO al referirnos al conjunto de características del mismo que van a determinar su comportamiento futuro. Se utilizará como determinación completa del sistema. Conjunto de valores para todas las variables del sistema.
- Estrategias de control:** Política de generación de acciones de control a lo largo del tiempo.
- Experto:** Persona con conocimiento amplio sobre un tema que le permite resolver problemas sobre dicho tema. El conocimiento en cuestión es tanto datos como procedimientos. Este conocimiento se suele conseguir a través de la experiencia.
- Exploración de reglas:** Mecanismo por el cual se determinan las reglas a lanzar en una inferencia. Es de especial importancia en caso de disponer de reglas con activación periódica.
- Fiabilidad:** Cualidad de un elemento de tener un número reducido de fallos. I(reliability)

- Focalizar:** Centrar el proceso de inferencia en un determinado subdominio del dominio completo. Es un mecanismo de descomposición por dominios que permite aumentar la eficiencia de los sistemas basados en el conocimiento. La focalización suele ser llevada a cabo por medio de mecanismos de activación y desactivación de bases de conocimiento.
- Fuente de datos:** Cualquier sistema que suministre datos al sistema de control. Fuentes de datos son: los sensores, bases de datos, simuladores, etc.
- Genérico:** Dícese de todo aquello referido a todos los objetos de una clase.
- Grafo de comportamiento:** Por lo general la predicción del comportamiento no dará un resultado único, sino que se generará un grafo que representará todos los posibles comportamientos del sistema. Los nodos del grafo son estados del sistema.
- Herencia:** Concepto crucial en la programación orientada a *objetos*. La herencia es la capacidad por la que una clase dispone —como propia— de la estructura de datos y los métodos de las clases de las que hereda. I(inheritance)
- Herramientas (software):** Programa que ayudan en la construcción de sistemas de proceso de información.
- Historia:** Conjunto de valores anteriores de una variable, con indicaciones del instante temporal al que corresponden.
- Hito:** Es un valor característico de una *variable* y que recibe una designación especial.
- Incertidumbre de comportamiento:** Uno de los tres aspectos de complejidad del proceso mencionados en esta tesis. Se refiere a la incertidumbre derivada del desconocimiento del modo exacto de interacción entre los elementos del proceso.
- Instante:** Corresponde a un instante temporal.
- Inteligencia artificial:** Conjunto de técnicas informáticas que emplean representaciones simbólicas complejas, y que tratan de emular el modo como opera la mente humana.
- Interfase:** Se referirá a un subsistema situado entre otros dos y que permite la comunicación entre ambos. El objetivo del mismo es adaptar los datos disponibles a un lado a los tipos de datos empleados en el otro. El interfase podrá ser pues UNIDIRECCIONAL ó BIDIRECCIONAL. También se denomina interfase o protocolo a la colección de mensajes a los que responde un *objeto*.
- Interfase externo:** Mecanismo de conexión con el exterior de un sistema.
- Intervalo:** Período de tiempo entre dos instantes.
- Intervalo de exploración:** Atributo característico del motor de inferencia que establece el período mínimo de activación de una regla con activación periódica.
- Intervalo de validez:** Período de tiempo durante el cual se considera válido el valor de una variable.
- Invocar:** Activar una regla.
- Jerarquía:** Forma de estructurar sistemas complejos en la que los subsistemas se relacionan con relaciones asimétricas.
- Jerarquía de clases:** Jerarquía compuesta por las clases utilizadas en una determinada representación de conocimiento. La relación asimétrica empleada es una relación ES-UN (IS-A).
- Mantenibilidad:** Cualidad de un elemento que permite su fácil adaptación a nuevas circunstancias. I(maintainability)
- Mensaje:** Invocación de uno de los métodos que constituyen el interfase de un *objeto*.
- Mensaje global:** Mensaje dirigido a todos los objetos activos de un sistema.
- Metacontrol:** Es un *control* que se sitúa por encima de otro control. El sistema controlado puede ser entendido a su vez como un sistema global, con lo cual es posible pensar en establecer un control que mejore el comportamiento de dicho sistema.

- Modos de fallo:** Cada una de las maneras en que se puede producir un fallo en un sistema. (Falta-Error-Fallo en software). Su estructuración permite generar un *árbol de modos de fallo* o *árbol de fallo*.
- Motor de inferencia:** Proceso encargado de la activación, selección y ejecución de los procesos elementales de un sistema basado en conocimiento. Por lo general estos procesos elementales serán disparos de reglas.
- Niveles de automatización:** Cada uno de los posibles modos en que se puede realizar el control de un proceso, desde el puramente manual al completamente automático, pasando a través de una serie de alternativas.
- Objeto:** Un ejemplar de la estructura de datos definida en una clase.
- Objeto focal:** Dícese de aquel objeto con una clase focal de reglas.
- OOP:** Object Oriented Programming. Programación orientada a objetos.
- Operador:** Es la persona encargada de supervisar la operación de un sistema. Si el sistema es controlado el operador establece los parámetros de funcionamiento del control (consignas, parámetros del regulador, etc).
- Paquete:** Unidad empleada en comunicación mediante redes.
- Paradigma MI-BC:** Paradigma del *sistema experto* clásico, en el que se separa el conocimiento procedural genérico del conocimiento sobre el dominio. De esta forma se generan el motor de inferencia (MI) y la base de conocimiento (BC):
- Patrón de actuación:** Se refiere a una especificación temporal de actividades de control.
- Pericia:** Cualidad que distingue al *experto*. No se ha utilizado el término *experiencia*, dado que este último se emplea para referirse al medio por el cual se llega a ser *experto*. I(*expertise*)
- Polimorfismo:** Capacidad de los *objetos* de responder de forma diferente a los mismos mensajes. Esta estrechamente relacionada con el enlace dinámico.
- Prioridad:** Criterio por el que se decide que actividad debe ejecutarse primero.
- Prioridad de actuaciones:** Criterio por el que se controla cual de dos actuaciones de control es más importante, lo que permite decidir cual se ejecutará. Es de especial importancia en caso de existencia de colisiones en las especificaciones de actuación originadas por dos procesos de inferencia diferentes.
- Prioridad de mensajes:** La respuesta de los objetos a los mensajes puede ser con múltiples líneas de ejecución o mediante una programación de las mismas. Los criterios de prioridad de mensajes permiten gestionarlos de forma que se responda antes a los más importantes.
- Prioridad de reglas:** Mecanismo de control del orden de activación de reglas. Por lo general conduce a malos usos ya que se emplea para hacer programación procedural.
- Prioridad de tareas:** En caso de sistemas basados en conocimiento que no permitan ejecuciones múltiples es preciso organizar las tareas de inferencia demandadas al mismo. Para ello se emplean criterios de prioridad de tareas basados en el tipo de demanda y en el demandante.
- Procedimiento:** Mecanismo de cómputo determinista.
- Proceso:** Un proceso es un sistema que evoluciona en el tiempo. También se entiende por tal un programa en ejecución en un computador.
- Programador:** Mecanismo de disposición a lo largo del tiempo de actividades a realizar. I(*scheduler*)
- Protocolo:** En ocasiones se refiere con este término al conjunto de mensajes a los que responde un objeto —la interfase de objeto.
- Región de operación:** Es el conjunto de intervalos de validez de un conjunto de *restricciones*. La simplicidad de los mecanismos de expresión de relaciones hace que la validez de la misma sea limitada a unos intervalos determinados de valores para las variables del sistema.

- Regla genérica:** Regla aplicable a cualquier objeto de una clase determinada.
- Regulador:** Mecanismo de cálculo de las señales de actuación sobre el sistema a controlar que permite al *control* conducir al sistema por una evolución determinada.
- Restricción:** Dícese de aquello que establece una restricción entre los posibles valores de un grupo de *variables*.
- Reutilizabilidad:** Término por lo general asociado al software, significando la propiedad de un determinado software para ser utilizado de nuevo en otro sistema sin requerir modificaciones. I(reusability)
- Seguridad:** Se dirá que un sistema es seguro cuando no se producen fallos o estos no producen daños.
- Sentido común:** Cualidad de los seres humanos por la que son capaces de realizar razonamientos sobre los hechos cotidianos sin apenas darse cuenta.
- Servidor de datos:** Ver *fuentes de datos*.
- Sistema:** Conjunto de elementos interrelacionados.
- Sistema hombre-máquina:** Sistema resultante de la interacción de un hombre y una máquina. En el estado actual la misión del hombre es proporcionar sentido común.
- Sistemas de soporte de decisiones:** Sistema informático de presentación de información a humanos que, mediante técnicas de filtrado e interpretación previa facilita la toma de decisiones por parte del humano.
- Situación:** Estado parcial del proceso.
- Status:** Información sobre el *estado* de un sistema. Estado interno de un HLO. Mensaje informando del estado interno (status real) de un HLO. Incluye un subconjunto de las variables de estado del HLO.
- Subclase:** Clase relacionada con otra por una relación ES-UN.
- Superclase:** Dícese de aquella clase de la que otra es subclase.
- Supervisión:** Actividad del operador que controla el funcionamiento del sistema CONTROL-PROCESO. I(oversight)
- Tiempo de expiración:** Instante en que concluye el intervalo de validez del valor de una variable.
- Valor:** Cada uno de los valores que puede tomar una *variable*. Estos valores serán habitualmente números reales, aunque en la representación simbólica se sustituirán por símbolos referentes a un valor real o a un conjunto de ellos.
- Variable:** Cada una de las variables del sistema a simular. Por lo general se refiere a funciones reales de variable real, siendo esta variable el tiempo.
- Variable K:** Es una variable que admite valores simbólicos, su espacio de cantidades es un conjunto finito.
- Variable mixta:** Variable que admite valoraciones de dos o más tipos.
- Variable N:** Es una variable que admite valores numéricos, su espacio de cantidades son los reales (punto flotante).
- Variable Q:** Es una variable que admite valores cualitativos, su espacio de cantidades es un conjunto finito con una relación de orden.

B.2. Acrónimos

- MRAC:** Model Reference Adaptive Control. Control adaptativo con modelo de referencia.
- RAM:** Reliability, Availability & Maintainability
- MI-BC:** Motor de Inferencia - Base de Conocimiento.
- IA:** Inteligencia Artificial.
- SE:** Sistema Experto.
- PID:** Proporcional, Integral y Diferencial.

Apéndice C

Bibliografía

C.1. Referencias Bibliográficas

- [Adaptech 89] EXPERT-AD, Expert Adaptive Control Software. ADAPTECH, 1989.
- [Alty 87] The Limitations of Rule-Based Expert Systems. J. L. Alty. *Knowledge Based Expert Systems in Industry*. Ellis Horwood, England 1987, pp. 12-16.
- [Andreae 77] *Thinking with the Teachable Machine*. John H. Andreae Academic Press, London, 1977.
- [Antsaklis 89] Towards Intelligent Autonomous Control Systems: Architecture and Fundamental Issues. P.J. Antsaklis and K.M. Passino. *Journal of Intelligent and Robotic Systems*, Vol. 1, No. 4, pp. 315-342, 1989.
- [Aracil 86] *Máquinas, Sistemas y Modelos*. Javier Aracil. Tecnos, 1986.
- [Aracil 89] Stability Indices for the Global Analysis of Expert Control Systems. Javier Aracil, Aníbal Ollero y Alfonso García Cerezo. *IEEE Trans. on Systems, man and Cybernetics*, Vol 19, No. 5, pp. 998-1007, 1989.
- [ASEA 83] Adaptive Controllers - A New Age in Process Control?, *Asea Innovation*, November 1983.
- [Åstrom 86] Expert Control. K.J. Åstrom, J.J. Anton y K.E. Årzen. *Automática*, Vol. 22, pp. 277-286. Pergamon Journals Ltd. 1986.
- [Åstrom 88] *Sistemas Controlados por Computador*. Karl Johan Åstrom y Björn Wittenmark. PARRANINFO, 1988.
- [Åstrom 89a] Towards Intelligent Control. Karl Johan Åstrom. *IEEE Control Systems Magazine*, April 1989, pp. 60-64.
- [Åstrom 89b] *Adaptive Control*. Karl Johan Åstrom y Björn Wittenmark. Addison-Wesley, 1989.
- [Bailey 88] Autonomous Decision Sources for Today's Automated Plants. S.J. Bayley. *Control Engineering*, July 1988.
- [Barreto 90] Qualitative and Quantitative Models of Systems. J. Barreto y M. de Neyer. *IMACS MIMS²*, 1990.
- [Bartos 89] AI Now More Realistically at Work in Industry. Frank J. Bartos. *Control Engineering*, July 1989.
- [Basagoiti 89] *Sistemas Expertos para Control Industrial*. Pedro Basagoiti. Tesis Doctoral, Escuela Técnica Superior de Ingenieros Industriales de la U.P.M. 1988.
- [Berger 89] A Hierarchical Framework for Learning Control. Kurt M. Berger y Kenneth A. Loparo. En *Artificial Intelligence, Simulation and Modelling*. Lawrence E. Widman, Kenneth A. Loparo y Norman R. Nielsen (eds). Wiley-Interscience, 1989.
- [Bernard 88] Use of a Rule-Based System for Process Control. John A. Bernard. *IEEE Control Systems Magazine*, pp 3-13. October 1988.
- [Bobrow 84] *Qualitative Reasoning about Physical Systems*. Daniel G. Bobrow (Ed). North-Holland. 1984. Reprinted from *Artificial Intelligence*, Vol. 24.

- [Bratko 88] Qualitative Modelling for Medical Diagnosis. I. Bratko. *1st IFAC Symposium on Modelling and Simulation in Biomedical Systems*, 1988, pp. 151-157.
- [Bridgeland 90] Three Qualitative Simulation Extensions for Supporting Economics Models. David Murray Bridgeland. *6th Conference on Artificial Intelligence Applications*, pp. 267-273, 1990.
- [Brown 82] Pedagogical, Natural Language and Knowledge Engineering Techniques in SOPHIE I, II and III. J.S. Brown, R.R. Burton and J. de Kleer. En *Intelligent Tutoring Systems*. D. Sleeman y J.S. Brown (Eds). Academic Press, 1982.
- [Bundy 86] *Catalogue of Artificial Intelligence Tools*. Alan Bundy. Springer Verlag, 1986.
- [Calle 90] Software para Control Robusto. Benedicto Calle. *Automática en Instrumentación*, Abril 1990.
- [Cavanna 89] QUIC Toolkit Demonstrator Applications. A. Cavanna, J.C. Chautard, C. Honorat, M. Levin y B. Klausen. *CIM Europe*. 1989.
- [Cerezo 88] *Aplicaciones del Razonamiento Aproximado en el Control y Supervisión de Procesos*. A. Cerezo. Ph.D. Thesis. U. Santiago de Compostela, España 1987.
- [Charbonnaud 90] A Quasi-Qualitative Calculus for Technical Diagnosis Modelling and Reasoning. Philippe Charbonnaud y Roger Bertin. *IMACS MIMS²*, 1990.
- [Clancey 89] Viewing Knowledge Bases as Qualitative Models. William J. Clancey. *IEEE Expert*, pp. 9-23, Summer 1989.
- [Cross 86] Knowledge-Based Pilot Aids: A case study in Mission Planning. Stephen E. Cross, Robert B. Bahnij, y Douglas O. Norman. *Artificial Intelligence and Man-Machine Systems*. Springer-Verlag, 1986, pp. 141-174.
- [Cuena 89] A Knowledge Based Environment for Artificial Intelligence Modelling of industrial Problems: Preliminary Concepts. J. Cuena y A. Salmerón. *IFAC Symposium on Information Control Problems in Manufacturing Technology*, Madrid 1989.
- [D'Ambrosio 89a] *Qualitative Process Theory Using Linguistic Variables*. Bruce D'Ambrosio. Springer-Verlag. 1989.
- [Dalle Molle 89] *Qualitative Simulation of Dynamic Chemical Processes*. David Dalle Molle. Ph. D. Thesis. Univ. of Texas at Austin. 1989.
- [de Kleer 84a] A Qualitative Physics Based on Confluences. Johan de Kleer y John Seely Brown. *Artificial Intelligence Vol.24*, pp. 7-83, 1984.
- [Dubois 80] *Fuzzy Sets and Systems: Theory and Applications*. Didier Dubois y Henri Pradé. Academic Press, 1980.
- [Efstathiou 87] Rule-Based Process Control Using Fuzzy Logic. J. Efstathiou. *Approximate Reasoning in Intelligent Systems, Decision and Control*. Elie Sanchez y Lofti A. Zadeh (Eds). Pergamon Press, pp. 145-158, 1987.
- [Ehrlich 83] DMS - A System for Defining and Managing Human-Computer Dialogues. R.W. Ehrlich. *Automática*, Vol. 19, No. 6, pp. 605-612, 1983.
- [Ellis 89] Active Objects: Realities and Possibilities. Clarence A. Ellis y Simon J. Gibss. En *Object Oriented Concepts, Databases and Applications*, Won Kim y Frederick H. Lochowsky (Eds). ACM Press, pp. 561-572, 1989.
- [Euristic 89] *CHRONOS User Manual*. Euristic Systems, 1989.
- [Fernández 77] *Modelos Matemáticos y de Simulación para Sistemas Continuos*. Gregorio Fernández Fernandez. Dpto. Publicaciones. ETSIT Madrid, 1977.
- [Filman 88] Reasoning with Worlds and Truth Maintenance in a Knowledge-Based Programming Environment. Robert E. Filman. *Communications of the ACM*. Vol.31 No.4, April 1988.
- [Fishwick 90] Qualitative Physics: Towards the Automation of System Problem Solving. Paul A. Fishwick and Bernard P. Zeigler. *IEEE Symposium on AI, Simulation and Planning in High Autonomy Systems*, pp. 118-134, 1990.

- [Fogarty 89] The Machine Learning of Rules for Combustion Control in Multiple Burner Installations. Terence C. Fogarty. *CAIA 89*, pp. 215-221. IEEE Computer Society Press, 1989.
- [Forbus 84] Qualitative Process Theory. Kenneth D. Forbus. *Artificial Intelligence*, Vol. 24, pp. 85-168, 1984.
- [Forbus 87] Interpreting Observations of Physical Systems. Kenneth D. Forbus. *IEEE Transaction on Systems, Man and Cybernetics*, Vol. 17 No. 3, May/June 1987, pp. 350-359.
- [Forgy 82] Rete: A Fast Algorithm for the Many Pattern/Many Object Pattern Match Problem. Charles L. Forgy. *Artificial Intelligence*, Vol. 19, pp. 17-37, 1982.
- [Foxboro 86] Process Automation: A 14-Part Series. Reprints from *Chemical Engineering*. Foxboro, 1986.
- [Franke 90] *CC: Component Connection Models for Qualitative Simulation - A User's Guide*. David Franke y Dan Dvorak. Univ. Texas AI Lab. AI90-126, 1990.
- [Frost 86] *Introduction to Knowledge Based Systems*. Robert A. Frost. Collins, 1986.
- [Fu 70] Learning Control Systems - Review and Outlook. King-Sun Fu. *IEEE Trans. on Automatic Control*, Vol. 15, No. 2, April 1970.
- [Fu 87] *Robotics: Control, Sensing, Vision and Intelligence*. K.S. Fu, R.C. González y C.S.G. Lee. McGraw-Hill 1987. (Trad. Esp. Robotica: Control, Detección, Visión e Inteligencia. McGraw-Hill, 1988).
- [Galán 89] *Introducción a los Sistemas Expertos*. Ramón Galán López. Publicaciones ETSII. Universidad Politécnica de Madrid. 1989.
- [Genesereth 84] The Use of Design Descriptions in Automated Diagnosis. Michael R. Genesereth. *Artificial Intelligence*, Vol. 24, pp. 411-436, 1984.
- [Gensym 88] *G2 User Manual*. Gensym Corporation. 1988.
- [Gensym 89] *An Introduction to G2*. Gensym Corporation. January 1989.
- [Gevarter 84] *Artificial Intelligence, Expert Systems, Computer Vision and Natural Language Processing*. William B. Gevarter. Noyes Publications, 1984.
- [González 89] *Arquitectura para Sistemas Expertos con Razonamiento Aproximado*. José Carlos González Cristóbal. Tesis Doctoral. Escuela Técnica Superior de Ingenieros de Telecomunicación. Universidad Politécnica de Madrid. 1989.
- [Grant 89] A Qualitative Modeling Toolbox for Control Architectures. E. Grant y D. Pearce. *IEEE International Symposium on Intelligent Control 1989*, pp. 526-530.
- [Guerrin 90] Biological Process Interpretation Using a Qualitative Reasoning Approach. François Guerrin. *IMACS MIMS²*, 1990.
- [Hayes 85a] The Second Naive Physics Manifesto. Patrick J. Hayes. *Formal Theories of the Commonsense World*, pp. 1-36. ALEX Publishing Corporation, Norwood, New Jersey, 1985.
- [Henkind 87] A Clinical Alarm System Using Techniques from Artificial Intelligence and Fuzzy Set Theory. S.J. Henkind, R.R. Yager, A.M. Benis y M.C. Harrison. *Approximate Reasoning in Intelligent Systems, Decision and Control*. Elie Sanchez and Lofti A. Zadeh (Eds). Pergamon Press, pp. 91-104, 1987.
- [Hinkelmann 89] Deep Knowledge for Configuration and Diagnosis in a Technical Environment. H. Hinkelmann y D. Karagiannis. *INCOM 89*, pp. 579-584. 1989.
- [Holmblad 87] Automatic Control of Cement Kilns by Fuzzy Logic Techniques. Lauritz Peter Holmblad. *Seminar on Expert Systems and their Applications*. UIMP, Santander, España. 1987.
- [Ilari 87] Control Experto. Juan Ilari Valenti. Tema de oposición, 1987.

- [**Intellicorp 86**] Model Based Reasoning in the KEE and SimKit Systems. *Intellinews*, Vol.2 No.2. August 1986.
- [**Intellicorp 87a**] Intellicorp Extends KEE for Electric Power Industry. *Intellicorp Application Note*, 1987.
- [**Jakob 90**] Situation Assesment for Process Control. François Jakob y Pierre Suslenschi. *IEEE Expert*, April 1990, pp. 49-59.
- [**Johansen 83**] Human System Interface Concerns in Support System Design. Gunnar Johansen, John E. Rijnsdorp y Andrew P. Sage. *Automática*, Vol. 19, No. 6, pp. 595-603, 1983.
- [**Juuso 90**] Expert Systems Combined with a Multilayer Simulation System. Esko K. Juuso y Kauko Leiviskä. *IMACS MIMS²*, 1990.
- [**Kandel 86**] *Fuzzy Mathematical Techniques with Applications*. Abraham Kandel. Addison-Wesley. 1986.
- [**Karel 89**] KLUE: A Diagnostic Expert System Tool for Manufacturing. Gerald Karel y Martin Kenner. *Intellinews*, Vol.5 No.1, Winter/Spring 1989.
- [**Kitowski 87**] Diagnostics of Fault States in Complex Physical Systems Using Fuzzy Relational Equations. J. Kitowski y M. Bargiel. *Aproximate Reasoning in Intelligent Systems, Decision and Control*. Elie Sanchez y Lofti A. Zadeh (Eds). Pergamon Press, pp. 175-194, 1987.
- [**Klir 69**] *An Approach to General Systems Theory*. George J. Klir. Van Nostrand Reinhold, New York, 1969.
- [**Kuhn 71**] *La Estructura de las Revoluciones Científicas*. T.S. Kuhn. Fondo de Cultura Económica, 1977.
- [**Kuipers 84**] Comonsense Reasoning about Causality: Deriving Behavior from Structure. Benjamin Kuipers. *Artificial Intelligence Vol.24*, pp. 169-204, 1984.
- [**Kuipers 86**] Qualitative Simulation. Benjamin J. Kuipers. *Artificial Intelligence*, Vol. 29, pp. 289-338.
- [**Kuipers 88**] Abstraction by Time-Scale in Qualitative Simulation for Biomedical Modelling. Benjamin J. Kuipers. *1st IFAC Symposiun on Modelling and Simulation in Biomedical Systems*, 1988, pp. 158-162.
- [**Kuipers 90**] *A Smooth Integration of Incomplete Quantitative Knowledge into Qualitative Simmulation*. Benjamin J. Kuipers y Daniel Berleant. Univ. Texas AI Lab. AI90-122, 1990.
- [**Laffey 88**] Real-Time Knowledge Based Systems. Thomas J. Laffey, Preston A. Cox, James L. Schmidt, Simon M. Kao y Jackson Y Read. *AI Magazine*, Spring 1988.
- [**Lan 90**] Printing Press Configuration. M.S. Lan y R.M. Panos. *IEEE Expert*, February 1990. pp. 65-73.
- [**Lehman 80**] Programs, Life Cycles and Laws of Software Evolution. Meir M. Lehman. *Proceedings of the IEEE*, Vol. 68, pp. 1060-1076, 1980.
- [**Leyval 90**] Event Oriented Versus Interval Oriented Qualitative Simulation. L. Leyval, S. Feray Beaumont y S. Gentil. *IMACS MIMS²*, 1990.
- [**MacFarlane 70**] *Dynamical System Models*. A.G.J. MacFarlane. Harrap, 1970.
- [**Mamdani 81**] *Fuzzy Reasoning and its Applications*. E.H. Mamdani y E.R. Gaines (Eds). Academic Press, 1981.
- [**Martínez 90**] *Realización de un Sistema Experto para el Control de la Fermentación de un Antibiótico*. Gamaliel Martínez Muñoz. Proyecto Fin de Carrera. ETSI Industriales, 1990.
- [**Mesarovic 70**] *Theory of Hierarchical, Multilevel, Systems*. M. Mesarovic, D. Macko, e Y. Takahara. Academic Press, 1970.
- [**Meyer 88**] *Object Oriented Software Construction*. Bertrand Meyer. Prentice Hall. 1988.
- [**Meystel 86**] Knowledge-Based Controller for Intelligent Mobile Robots. A. Meystel *Artificial Intelligence and Man-Machine Systems*. Springer-Verlag, 1986, pp. 113-140.

- [**Milne 87**] Strategies for Diagnosis. Robert Milne. *IEEE Transactions on Systems Man and Cybernetics*, Vol. 17 No. 3, May/June 1987, pp. 333-339.
- [**Mladenov 89**] *Distributed Intelligence Systems: Methods and Applications*. D. Mladenov (Ed.). Pergamon Press, 1989.
- [**Moliner 89**] *Diccionario de Uso del Español*. María Moliner. Gredos, 1989.
- [**Morgan 90**] State Descriptions Through Qualitative Vectors. Tony Morgan. *IMACS MIMS²*, 1990.
- [**Nicolosi 88**] Qualitative Simulation of Compartmental Systems. Emma Nicolosi y Mark S. Lenanig. *1st IFAC Symposium on Modelling and Simulation in Biomedical Systems*, 1988, pp. 193-198.
- [**Nii 86**] Blackboard Systems. H. Penny Nii. *The AI Magazine*. Summer 1986.
- [**Nikolic 86**] An Algorithm for Learning Without External Supervision and Its Application to Learning Control Systems. Z.J. Nikolic y K.S. Fu. *IEEE Trans. on PAMI*, Vol. 8, No. 3, pp. 304-312, 1986.
- [**Nilsson 87**] *Principios de Inteligencia Artificial*. Nils J. Nilsson. Díaz de Santos, Madrid 1987.
- [**Nistal 89b**] Control Optimo. José Manuel Nistal Bartolomé. *Ingeniería Química*, Octubre 1989.
- [**Oliveira 90**] An Architecture for the Supervision of Fuzzy Controllers. P.J. Oliveira, P.U. Lima, J.J. Sentieiro, R. Sanz, R. Galán y A. Jiménez. *IEEE International Workshop on Intelligent Robots and Systems*, IROS '90, 1990.
- [**Ollero 89b**] Los Sistemas Expertos en Control de Procesos. Aníbal Ollero Baturone. *Automática e Instrumentación*, Julio 1989.
- [**Pau 89**] Knowledge Representation Approaches in Sensor Fusion. L. Pau. *Automática* Vol. 25, No. 2, pp. 207-214. 1989.
- [**Pedrycz 89**] *Fuzzy Control and Fuzzy Systems*. Witold Pedrycz. Research Studies Press Ltd. John Wiley and Sons Inc. 1989.
- [**Rothenberg 89**] The Nature of Modelling. Jeff Rothenberg. En *Artificial Intelligence, Simulation and Modelling*. Lawrence E. Widman, Kenneth A. Loparo y Norman R. Nielsen (eds). Wiley-Interscience, pp. 75-92, 1989.
- [**S₂O 90**] NEMO. S₂O Informational Brochure. 1989.
- [**Sanz 89a**] *Métricas de Complejidad Software*. R. Sanz, L.F. Solórzano y J.R. Velasco. Documento Interno DIT, 1989.
- [**Sanz 89b**] *Control Inteligente de Procesos en la Industria del Cemento*. R. Sanz, A. Jiménez, R. Galán, C.Peral y V. Rubio. Informe ASTECNO/DISAM-12.89/01, 1989.
- [**Saridis 83**] Intelligent Robotic Control. George N. Saridis. *IEEE Trans. on Automatic Control*, Vol. 28, No. 5, pp. 547-557, 1983.
- [**Saridis 85**] Intelligent Control: A New Engineering and Scientific Reality. George N. Saridis. *IEEE Workshop on Intelligent Control 1985*, pp. 220-221, 1985.
- [**Saridis 88a**] Analytical Design of Intelligent Machines. George N. Saridis y Kimon P. Valavanis. *Automatica*, Vol. 24, No. 2, pp. 123-133, 1988.
- [**Saridis 88b**] Knowledge Implementation: Structures of Intelligent Control. George N. Saridis. *Journal of Robotic Systems*, Vol. 5, No. 4, pp. 255-268, 1988.
- [**Saridis 89**] Analytic Formulation of the Principle of Increasing Precision with Decreasing Intelligence for Intelligent Machines. George N. Saridis. *Automatica*, Vol. 25, No. 3, pp. 461-467, 1989.
- [**Scarl 87**] Diagnosis and Sensor Validation Through Knowledge of Structure and Function. E.A. Scarl, J.R. Jamieson y C.I. Delaune. *IEEE Systems, Man and Cybernetics*, Vol. 17, No. 3, pp. 360-368. 1987.
- [**Shaw 89**] Reasoning About Time in High-Level Language Software. Alan C. Shaw. *IEEE Transactions on Software Engineering*. Vol.15 No. 7, July 1989.

- [**Sheridan 83a**] Adapting Automation to Man, Culture and Society. Thomas B. Sheridan, T. Vámos y Shuhei Aida. *Automática*, Vol. 19, No. 6, pp. 605-612, 1983.
- [**Shirley 87**] Some Lessons Learned Using Expert Systems for Process Control. Richard T. Shirley. *IEEE Control Systems Magazine*. December 1987.
- [**Shrager 87**] Issues in the Pragmatics of Qualitative Modelling: Lessons Learned from a Xerographics Project. Jeff Shrager, Daniel S. Jordan, Thomas P. Moran, Gregor Kiczales y Daniel M. Russell. *Communications of the ACM*, Vol. 30 No. 12, December 1987, pp. 1036-1047.
- [**Simmonds 88**] Representation of Real Knowledge for Real-Time Use. W.H. Simmonds. *IFAC Workshop on AI in Real Time Control*. 1988.
- [**Sleeman 82**] *Intelligent Tutoring Systems*. D. Sleeman y J.S. Brown (Eds). Academic Press. 1982.
- [**Togai 89**] Fuzzy Control Package Informational Brochure. Togai Infralagic, 1989.
- [**Tong 77**] A Control Engineering Review of Fuzzy Systems. R.M. Tong. *Automatica*, Vol. 13, pp. 559-569. Pergamon Press 1977.
- [**Tzafestas 89**] A Framework for Knowledge Based Control. S. Tzafestas y S. Ligeza. *Journal of Intelligent and Robotic Systems*, Vol. 1, No. 4, pp. 407-426, 1989.
- [**Voss 88**] Architectural Issues for Expert Systems in Real Time Control. H. Voss. *IFAC Workshop on AI in Real Time Control*. 1988.
- [**Widman 89a**] *Artificial Intelligence, Simulation and Modelling: A Critical Survey*. Lawrence E. Widman y Kenneth A. Loparo. En *Artificial Intelligence, Simulation and Modelling*. Lawrence E. Widman, Kenneth A. Loparo y Norman R. Nielsen (eds). Wiley-Interscience, 1989.
- [**Widman 89b**] Semi-Quantitative "Close-Enough" Systems Dynamics Models: An Alternative to Qualitative Simulation. Lawrence E. Widman. En *Artificial Intelligence, Simulation and Modelling*. Lawrence E. Widman, Kenneth A. Loparo y Norman R. Nielsen (eds). Wiley-Interscience, 1989.
- [**Yazdani 86a**] Intelligent Tutoring Systems Survey. M. Yazdani. *AI Review*, No. 1, pp. 43-52, 1986.
- [**Zadeh 73**] Outline of a New Approach to the Analysis of Complex Systems and Decision processes. Lofti A. Zadeh. *IEEE Transactions on Systems, Man and Cybernetics*, Vol. 3, No. 1, pp. 28-44. January 1973.
- [**Zadeh 77**] PRUF - A Meaning Representation Language for Natural Languages. Lofti A. Zadeh. *Fuzzy Reasoning and its Applications*. Academic Press. pp. 1-66, 1981.
- [**Zadeh 88**] Fuzzy Logic. Lofti A. Zadeh. *IEEE Computer*. Abril 1988.
- [**Zeigler 90**] *Object Oriented Simulation with Hierarchical, Modular, Models*. Bernard P. Zeigler. Academic Press 1990.
- [**Zhang 90a**] Fault Diagnosis of Mixing Process Using Deep Qualitative Knowledge Representation of Physical Behaviour. J. Zhang, P.D. Roberts y J.E. Ellis. *Journal of Intelligent and Robotic Systems*, Vol. 3, No. 2, pp. 103-116, 1990.

C.2. Bibliografía adicional

- [**Abbott 87**] Knowledge Abstraction. Russel J. Abbott. *Communications of the ACM*, Vol. 30 No. 8, pp. 664-670. 1987.
- [**Aiello 86**] Representation and Use of Metaknowledge. L. Aiello, C. Cecchi y D. Sartini. *Proceedings of the IEEE*, Vol. 74, No. 10, pp. 1304-1321, 1986.
- [**Allen 83**] Maintaining Knowledge about Temporal Intervals. James F. Allen. *Communications of the ACM*, Vol. 26, No. 11, pp. 832-843, 1983.
- [**Allen 85**] A Model of Naive Temporal Reasoning. James F. Allen y Henry A. Kautz. *Formal Theories of the Commonsense World*, pp. 251-268. ABLEX Publishing Corporation, Norwood, New Jersey, 1985.

- [Alty 89] Knowledge-based Dialogue for Dynamic Systems. J. L. Alty y G. Johansen. *Automatica*, Vol. 25, No. 6, pp. 829-840, 1989.
- [Aracil 83] *Introducción a la Dinámica de Sistemas*. Javier Aracil. Alianza Universidad Textos, 1983.
- [Asse 87] Diagnosis Based on Subjective Information in a Solar Energy Plant. A. Asse, A. Mairzener, A. Moreau y D. Willaëys. *Approximate Reasoning in Intelligent Systems, Decision and Control*. Elie Sanchez y Lofti A. Zadeh (Eds). Pergamon Press, pp. 159-174, 1987.
- [Atwood 86] Modelos Causales: la Nueva Generación de Sistemas Expertos. M.E. Atwood, R. Brooks, E.R. Radlinski. *Comunicaciones Eléctricas*, Vol. 60, No. 2, pp. 180-184, 1986.
- [Azmani 90] Artificial Intelligence Approach for the Causal Analysis of Bond-Graph Models. A. Azmani, R. Bouayad y G. Dauphin-Tanguy. *IMACS MIMS²*, 1990.
- [Årzen 89] An Architecture for Expert System Based Feedback Control. Karl-Erik Årzen. *Automática*, Vol. 25, No. 6, pp. 813-827. Pergamon Journals Ltd. 1989.
- [Åstrom 85] Process Control-Past, Present and Future. Karl Johan Åstrom *IEEE Control Systems Magazine*, pp. 3-10. August 1985.
- [Babb 90a] "Object Oriented" Control Configuration. Michael Babb. *Control Engineering*, April 1990, pp. 69.
- [Babb 90b] New Expert System Guides Control Operators. Michael Babb. *Control Engineering*, June 1990, pp. 80-81.
- [Baldwin 87] An Uncertainty Calculus for Expert Systems. J.F. Baldwin. *Approximate Reasoning in Intelligent Systems, Decision and Control*. Elie Sanchez y Lofti A. Zadeh (Eds). Pergamon Press. pp. 33-54. 1987.
- [Bare 90] Design of a Self-Tuning Rule Based Controller for a Gasoline Catalytic Reformer. Walter H. Bare, Robert J. Mulholland y Sam S. Sofer. *IEEE Trans. on Automatic Control*, Vol. 35, No. 2, pp. 156-164, 1990.
- [Bartos 87] Expert System Tools Emerge from the Technology of AI. Frank J. Bartos. *Control Engineering*, December 1987.
- [Basagoiti 87] *Sistema Experto para Verificar la Calidad de Estrategias para un Robot Manipulador en una Célula de Mecanizado*. Pedro Basagoiti y Ramón Galán. Informe Interno DISAM 01/25-3-87.
- [Bassiuny 90] Modeling and Simulation of the Homogenization Process in the Dry Process Cement Production. A.M. Bassiuny, H.A. Nour Eldin y A.A. Hegazy. *IMACS MIMS²*, 1990.
- [Bellot 90] X Window System: Sistema Estándar de Ventanas. Marisa Bellot, Oriol Sánchez y Nacho Navarro. *Mundo Electrónico*, Julio 1990, pp. 77-87.
- [Bennett 86] Tools for Building Advanced User Interfaces. J.L. Bennett. *IBM Systems Journal*, Vol. 25, No. 3/4, pp. 354-368, 1986.
- [Bergadano 87] Learning from Examples in Presence of Uncertainty. F. Bergadano, A. Giordana y L. Saitta. *Approximate Reasoning in Intelligent Systems, Decision and Control*. Elie Sanchez y Lofti A. Zadeh (Eds). Pergamon Press, pp. 105-124, 1987.
- [Boden 77] *Artificial Intelligence and Natural Man*. Margaret A. Boden. The Harvester Press Ltd. 1977 (Trad. Esp. *Inteligencia Artificial y Hombre Natural*. Tecnos, 1984).
- [Bocca 87] A logic Programming System for KBMSs - Educe. J. Bocca. *Knowledge Based Expert Systems in Industry*. Ellis Horwood, England 1987, pp. 23-35.
- [Boettcher 83] Modelling and Analysis of Teams of Interacting Decisionmakers with Bounded Rationality. K.L. Boettcher y A.H. Lewis. *Automática*, Vol. 19, No. 6, pp. 703-710, 1983.
- [Booch 86] Object-Oriented Development. Grady Booch. *IEEE Trans. on Software Engineering*, Vol. 12, No. 2, 1986.

- [**Buckley 87**] Portfolio Analysis Using Possibility Distributions. *Aproximate Reasoning in Intelligent Systems, Decision and Control*. Elie Sanchez y Lofti A. Zadeh (Eds). Pergamon Press. pp. 69-76. 1987.
- [**Buchanan 84**] *Rule Based Expert Systems. The MYCIN Experiments of the Stanford Heuristic Programming Project*. Bruce G. Buchanan y Edward H. Shortliffe. Addison-Wesley, 1984.
- [**Bunt 85**] Formal Representation of (Quasi-) Continuous Concepts. Harry Bunt. *Formal Theories of the Commonsense World*, pp 37-70. ABLEX Publishing Corporation, Norwood, New Jersey, 1985.
- [**Burns 89**] *Real-Time Systems and their Programming Languages*. Alan Burns y Andy Wellings. Addison Wesley. 1989.
- [**Bürle 87**] The Role of Qualitative Reasoning in Modelling. Gabriele Bürle. *AI, ES and Languages in Modelling and Simulation*, Barcelona, 1987.
- [**Camacho 90**] Centros de Control: la Experiencia Empieza con la Simulación. Eduardo F. Camacho. *Automática en Instrumentación*, Mayo 1990.
- [**Carbonell 85**] Metaphor and Commonsense Reasoning. Jaime G. Carbonell y Steven Minton. *Formal Theories of the Commonsense World*, pp. 405-426. ABLEX Publishing Corporation, Norwood, New Jersey, 1985.
- [**Cardelli 85**] On Understanding Types, Data Abstraction and Polymorphism. Luca Cardelli y Peter Wegner. *ACM Computing Surveys*, Vol.17 No.4. pp. 471-523. 1985.
- [**Chu 90**] Modeling Diagnostic Problem-Solving at Multiple Levels of Abstraction. Bei-Tseng Bill Chu. *6th Conference on Artificial Intelligence Applications*, pp. 55-61, 1990.
- [**Citrin 87**] Extracting Paralelism from Sequential Prolog: Experiences with the Berkeley PLM. W. Citrin. *Knowledge Based Expert Systems in Industry*. Ellis Horwood, England 1987, pp. 36-56.
- [**Coad 89**] *Object Oriented Analysis*. Peter Coad y Edward Yourdon. YOURDON Press. 1989.
- [**Cois 89**] Qualitative Analysis in Simulating Analog Circuits. A. Cois, A. Fanni, S. Pes y A. Giua. *IEEE International Symposium on Intelligent Control 1989*, pp. 531-536.
- [**Crowley 89**] Knowledge, Symbolic Reasoning and Perception. James L. Crowley. 1989.
- [**Cuena 85**] *Lógica Informática*. José Cuena. Alianza Editorial AI-1, 1985.
- [**Cuena 86**] *Inteligencia Artificial: Sistemas Expertos*. José Cuena, Gregorio Fernández, Ramón López de Mántars y M^a Felisa Verdejo. Alianza Editorial AI-2, Madrid 1986.
- [**Cullingford 83**] A Heuristically 'Optimal' Knowledge Base Organization Technique. R.E. Cullingford y L.J. Joseph. *Automática*, Vol. 19, No. 6, pp. 647-654, 1983.
- [**D'Ambrosio 87**] *Real-Time Process Management for Materials Composition in Chemical Manufacturing*. Bruce D'Ambrosio, Michael R. Fehling, Peter Raulefs y B. Michael Wilber. *IEEE Expert*, Summer 1987.
- [**D'Ambrosio 89b**] *Extending the Mathematics in Qualitative Process Theory*. Bruce D'Ambrosio. En *Artificial Intelligence, Simulation and Modelling*. Lawrence E. Widman, Kenneth A. Loparo y Norman R. Nielsen (eds). Wiley-Interscience, 1989.
- [**Dague 87**] DEDALE: an Expert System in VM/Prolog. Ph. Dague, Ph. Devès, Z. Zein y J.P. Adam. *Knowledge Based Expert Systems in Industry*. Ellis Horwood, England 1987. pp. 57-68.
- [**Davis 83**] Representing Structure and Behavior of Digital Hardware. Randall Davis y Howard Srhobe. *IEEE Computer*, Vol. 16, No. 10, pp. 75-82, 1983.
- [**Davis 84**] Diagnostic Reasoning Based on Structure and Behavior. Randall Davis. *Artificial Intelligence Vol.24*, pp. 347-410, 1984.

- [Davis 88] Model-Based Reasoning: Troubleshooting. Randall Davis. *MIT AI Lab. Memo 1059*, 1988.
- [Davis 90] A Comparison of Techniques for the Specification of External System Behavior. Alan M. Davis. *Communications of the ACM*, Vol. 31, No. 9, pp. 1098-1115, 1988.
- [de Kleer 84b] How Circuits Work. Johan de Kleer. *Artificial Intelligence Vol.24*, pp. 205-280, 1984.
- [de Prada 90] *Control Anticipativo: mas fácil de lo que parece*. César de Prada. *Automática e Instrumentación*, Abril 1990.
- [deMarco 79] *Structured Analysis and System Specification*. Tom de Marco. YOURDON Press. 1979.
- [Deaño 81] *Introducción a la lógica formal*. Alfredo Deaño. Alianza Editorial AUT-11, Madrid 1981.
- [Doraiswami 89] Performance Monitoring in Expert Control Systems. R. Doraiswami y J. Jiang. *Automática*, Vol. 25, No. 6, pp. 799-811, 1989.
- [Efros 87] *Física y Geometría del Desorden*. A. Efros. MIR, 1987.
- [Elwart 90] New Software Structures Extend Control Capabilities. Steven P. Elwart y Peter G. Martin. *Control Engineering*, June 1990, Vol. II, pp. 16-17.
- [Farquhar 90] *QSIM User's Manual*. Adam Farquhar y Benjamin Kuipers. Univ. of Texas, AI Lab. AI90-123, 1990.
- [Fikes 85] The Role of Frame Based Representation in Reasoning. Richard Fikes y Tom Keller. *Communications of the ACM*, Vol. 28, No. 9, pp. 904-920. 1985.
- [Fischer 83] Symbiotic, Knowledge-based Computer Support Systems. Gerhard Fischer. *Automática*, Vol. 19, No. 6, pp. 627-637, 1983.
- [Fischer 86] Cognitive Science: Information Processing in Humans and Computers. Gerhard Fischer. *Artificial Intelligence and Man-Machine Systems*. Springer-Verlag, 1986, pp. 84-112.
- [Fishwick 89] Process Abstraction in Simulation Modeling. Paul A. Fishwick. En *Artificial Intelligence, Simulation and Modelling*. Lawrence E. Widman, Kenneth A. Loparo y Norman R. Nielsen (eds). Wiley-Interscience, 1989.
- [Forbus 85] The Role of Qualitative Dynamics in Naive Physics. Kenneth D. Forbus. *Formal Theories of the Commonsense World*, pp. 185-226. ABLEX Publishing Corporation, Norwood, New Jersey, 1985.
- [Fouché 90] Qualitative Simulation of Feedback Control Systems. Pierre Fouché y Benjamin Kuipers. *IMACS MIMS²*, 1990.
- [Fox 90] AI and Expert System Myths, Legends and Facts. Mark S. Fox. *IEEE Expert*, February 1990. pp. 8-20.
- [Frenkel 88] The Art and Science of Visualizing Data. Karen a. Frenkel. *Communications of the ACM*, Vol. 31, No. 2, pp. 111-123, 1988.
- [Friedrich 90] MOMO - Model Based Diagnosis for Everybody. Gerhard Friedrich. *6th Conference on Artificial Intelligence Applications*, pp. 207-213, 1990.
- [Fu 71] Learning Control Systems and Intelligent Control Systems: An Intersection of Artificial Intelligence and Automatic Control. King-Sun Fu. *IEEE Trans. on Automatic Control*, Vol. 16, No. 1, February 1971.
- [Funt 83] Analogical Modes of Reasoning and Process Modeling. Brian V. Funt. *IEEE Computer*, October 1983, pp. 99-104.
- [Gero 87] *Expert Systems in Computer Aided Design*. John S. Gero (Ed.). North Holland, 1987.
- [Gevarter 82] An Overview of Expert Systems. William B. Gevarter. *National Bureau of Standards Report No. NBSIR 82-2505*, May 1982.
- [Giles 75] Lukasiewicz Logic and Fuzzy Set Theory. R. Giles. *Fuzzy Reasoning and its Applications*. Academic Press. pp. 117-132, 1981.
- [Goguen 73] Concept Representation in Natural and Artificial Languages: Axioms, Extensions and Applications for Fuzzy Sets. J.A. Goguen Jr. *Fuzzy Reasoning and its Applications*. Academic Press. pp. 67-116, 1981.

- [**Good 84**] Building a User-Driver Interface. M. D. Good, J.A. Whiteside, D.R. Wixon y S.J. Jones. *Communications of the ACM*, Vol. 27, No. 10, pp. 1032-1043, 1984.
- [**Graham 82**] Linguistic Decision Structures for Hierarchical Systems. James H. Graham y George N. Saridis. *IEEE Transactions on Systems, Man and Cybernetics*, Vol. SMC-12, N° 3, Mayo-Junio 1982.
- [**Green 87a**] Qualitative Reasoning During Design About Shape and Fit: a Preliminary Report. Douglas S. Green y David C. Brown. En *Expert Systems in Computer Aided Design*. John S. Gero (Ed.), pp. 93-118. North Holland, 1987.
- [**Green 87b**] A Fuzzy Decision Support System for Strategic Planning. N. Green Hall. *Approximate Reasoning in Intelligent Systems, Decision and Control*. Elie Sanchez y Lofti A. Zadeh (Eds). Pergamon Press. pp. 77-90. 1987.
- [**Green 87c**] Verification and Validation of Expert Systems. C.J.R. Green y M.M. Keyes. *Proceedings WESTEX-87*, pp. 38-43, 1987.
- [**Haest 90**] ESPION: an Expert System for System Identification. M. Haest, G. Bastin, M. Gevers y V. Wertz. *Automatica*, Vol. 26, No. 1, pp. 85-95, 1990.
- [**Hammond 87**] Logic-Based Tools for Building Expert and Knowledge-Based systems: Successes and Failures of Transferring the Technology. P. Hammond. *Knowledge Based Expert Systems in Industry*. Ellis Horwood, England 1987. pp. 69-90.
- [**Hamscher 90a**] Modeling Digital Circuits for Troubleshooting: An Overview. Walter Hamscher. *6th Conference on Artificial Intelligence Applications*, pp. 2-8, 1990.
- [**Hamscher 90b**] XDE: Diagnosing Devices with Hierarchic Structure and Known Component Failure Modes. Walter Hamscher. *6th Conference on Artificial Intelligence Applications*, pp. 48-54, 1990.
- [**Hanbaba 89**] Comunicación Hombre-Máquina con el Sistema de Control-Mando de Centrales PROCONTROL P para Centrales Térmicas. Peter Hanbaba. *Revista ABB*, No. 5, pp. 11-18, 1989.
- [**Handelman 88**] Perspectives on the Use of Rule Based Control. D.A. Handelman and R.F. Stengel. En *Artificial Intelligence in Real-Time Control*. M.G. Rodd y G.J. Suski (eds). Pergamon Press, 1988.
- [**Hayes 85b**] Naive Physics I: Ontology for Liquids. Patrick J. Hayes. *Formal Theories of the Commonsense World*, pp. 71-108. ALEX Publishing Corporation, Norwood, New Jersey, 1985.
- [**Hayes-Roth 84**] The Knowledge-Based Expert System: A Tutorial. Frederick Hayes-Roth. *IEEE Computer*, September 1984, pp. 11-28.
- [**Henkind 88**] An Analisis of Four Uncertainty Calculi. S.J. Henkind y M.C. Harrison. *IEEE Transactions on Systems, Man and Cybernetics*, Vol. 18, No. 5, pp. 700-714, 1988.
- [**Henneman 84**] Measures of Human Problem Solving Performance in Fault Diagnosis Tasks. Richard L. Henneman y William B. Rouse. *IEEE Trans. on Systems, Man and Cybernetics*, Vol. 14, No. 1, pp. 99-112, 1984.
- [**Hunt 84**] A Fuzzy Rule-Based Model of Human Problem Solving. Ruston M. Hunt y William B. Rouse. *IEEE Trans. on Systems, Man and Cybernetics*, Vol. 14, No. 1, pp. 112-120, 1984.
- [**Hunt 90**] Performing Augmented Model-Based Diagnosis. John Hunt y Chris Price. *IMACS MIMS²*, 1990.
- [**Husson 86**] Sistema Experto para control de Procesos Continuos en Tiempo Real. Jean-Michel Husson y Joaquín Tejedor. *Automática e Instrumentación*, Junio 1986.
- [**Intellicorp 87b**] Two-Weeks: Rapid Prototyping at Babcock & Wilcox. *Intellicorp Application Note*, 1987.
- [**Intellicorp 87c**] Vitro Builds Circuit Fault Diagnosis Expert System. *Intellicorp Application Note*, 1987.

- [**Intellicorp 87d**] NASA Scientists Use KEE in Expert System Design Study. *Intellicorp Application Note*, 1987.
- [**Intellicorp 87e**] SIAC System Monitors Trading Network. *Intellicorp Application Note*, 1987.
- [**Intellicorp 87f**] Taking Control of Complexity. *Intellicorp Application Note*, 1987.
- [**Intellicorp 88**] How to Get Started in AI. *Intellicorp*, 1988.
- [**Israel 85**] A Short Companion to the Naive Physics Manifesto. David Israel. *Formal Theories of the Commonsense World*, pp. 427-448. ABLEX Publishing Corporation, Norwood, New Jersey, 1985.
- [**Jiang 89**] Design of a Real-Time, Knowledge-Based Controller with Applications in Hydraulic Turbine Generator Systems. J. Jiang y R. Doraiswami. *Journal of Intelligent and Robotic Systems* Vol.2 No.2&3, 1989. pp. 229-244.
- [**Kickert 76**] Application of a Fuzzy Controller in a Warm Water Plant. W.J.M. Kickert y H.R. Van Nauta Lemke. *Automática*, Vol. 12, pp. 301-308. 1976.
- [**Kim 89**] *Object Oriented Concepts, Databases and Applications*, Won Kim y Frederick H. Lochowsky (Eds). ACM Press, pp. 561-572, 1989.
- [**King 77**] The Application of Fuzzy Control Systems to Industrial Processes. P.J. King y E.H. Mamdani. *Automatica*, Vol. 13, pp. 235-242. Pergamon Press 1977.
- [**King 88**] Intelligent Control in the Cement Industry. Robert King. *IFAC Conference on Distributed Intelligence*, Varna, Bulgaria, June 1988.
- [**King 89**] Distributed Intelligent Industrial Control - Aprocess Industry Paradigm. Robert King. 1989.
- [**Klir 89**] *Fuzzy Sets, Uncertainty and Information*. George J. Klir y Tina A. Folger. Prentice-Hall. 1988.
- [**Kokar 89**] Accumulating Qualitative Knowledge. Mieczyslaw M. Kokar. *IEEE International Symposium on Intelligent Control* 1989, pp. 514-519.
- [**Konolige 85**] Belief and Incompleteness. Kurt Konolige. *Formal Theories of the Commonsense World*, pp. 359-404. ABLEX Publishing Corporation, Norwood, New Jersey, 1985.
- [**Kook 90**] Representation of Models for Solving Real-World Problems. Hyung Joon Kook y Gordon S. Novak Jr. *6th Conference on Artificial Intelligence Applications*, pp. 274-280, 1990.
- [**Kopetz 79**] *Software Reliability*. H. Kopetz. Mac-Millan, 1979.
- [**Kreutzer 86**] *System Simulation. Programming Styles and Languages*. Wolfgang Kreutzer Addison-Wesley, 1986.
- [**Krijgsman 88**] Knowledge Based Real Time Control. A.J. Krijgsman, H.B. Verbruggen y P.M. Bruijn. En *Artificial Intelligence in Real-Time Control*. M.G. Rodd y G.J. Suski (eds). Pergamon Press, 1988.
- [**Kriz 87**] *Knowledge Based Expert Systems in Industry*. Jiri Kriz (Ed). Ellis Horwood, England, 1987.
- [**Kuipers 87**] Qualitative Simulation as causal Explanation. Benjamin J. Kuipers. *IEEE Transactions on Systems, Man and Cybernetics*, Vol. 17 No. 3, May/June, 1987. pp. 432-444.
- [**Kuipers 89a**] Qualitative Reasoning: Modeling and Simulation with incomplete Knowledge. Benjamin J. Kuipers. *Automatica* Vol. 25 No. 4, pp. 171-185, 1989.
- [**Kuipers 89b**] Qualitative Reasoning with Causal Models in Diagnosis of Complex Systems. Benjamin Kuipers. En *Artificial Intelligence, Simulation and Modelling*. Lawrence E. Widman, Kenneth A. Loparo y Norman R. Nielsen (eds). Wiley-Interscience, 1989.
- [**Kurtz 89**] An Object-Oriented Methodology for Systems Analysis and Specification. B.D. Kurtz, D. Ho y Teresa A. Wall. *Hewlett-Packard Journal*, April 1989.
- [**Laduzinsky 90**] Looking for Mr. Good LAN. Alan J. Laduzinsky. *Control Engineering*, April 1990, pp. 74-78.

- [Laranjeira 90] Software Size Estimation of Object-Oriented Systems. Luiz A. Laranjeira. *IEEE Trans. on Software Engineering*, Vol. 16, No. 5, pp. 510-522, 1990.
- [Lebailly 87] Use of Fuzzy Logic in a Rule-Based System in Petroleum Geology. J. Lebailly, R. Martin-Clouaire y H. Prade. *Approximate Reasoning in Intelligent Systems, Decision and Control*. Elie Sanchez y Lofti A. Zadeh (Eds). Pergamon Press, pp. 125-144, 1987.
- [Li 89] Development of Fuzzy Algorithms for Servo Systems. Y.F. Li y C.C. Lau. *IEEE CS Magazine*, April 1989, pp. 65-72.
- [Lim 90] Implementing Fuzzy Rule-Based Systems on Silicon Chips. M.H. Lim y Y. Takefuji. *IEEE Expert*, February 1990. pp. 31-45.
- [Lifschitz 89] Miracles in Formal Theories of Action. Vladimir Lifschitz y Arkady Rabinov. *Artificial Intelligence*, Vol 38, pp. 225-237. 1989.
- [López 89] Sistemas Expertos, Limitaciones y Perspectivas. Ramón López de Mántaras. *Mundo Electrónico*, Noviembre 1989.
- [Lounamaa 89] An Incremental Object-Oriented Language for Continuous Simulation Models. Pertti Lounamaa. En *Artificial Intelligence, Simulation and Modelling*. Lawrence E. Widman, Kenneth A. Loparo y Norman R. Nielsen (eds). Wiley-Interscience, 1989.
- [Lu 89] Modern Computer Control Strategies of Iron and Steel Making. Yong-zai Lu. *6th MMM-89*.
- [Luo 89] Multisensor Integration and Fusion in Intelligent Systems. Ren C. Luo y Michael G. Kay. *IEEE Trans. on Systems Man and Cybernetics*, Vol. 19, No. 5, pp. 901-931, 1989.
- [Maderlechner 87] Model Guided Interpretation Based on Structurally Related Image Primitives. G. Maderlechner, E. Egeli y F. Klein. *Knowledge Based Expert Systems in Industry*. Ellis Horwood, England 1987, pp. 91-97.
- [Maki 89] Application of Fuzzy Theory for Automatic Control of Hot Stove Combustion Gas Flow Rate. Y. Maki et al. *6th MMM-89*.
- [Mahmoud 89] A Real-Time Expert Control System for Dynamical Processes. Magdi S. Mahmoud, Shawki Z. Eid y Ahmed A. Abou-El-soud. *IEEE Trans. on Systems, man and Cybernetics*, Vol 19, No. 5, pp. 1101-1105, 1989.
- [Mañas 87] Word Division in Spanish. José A. Mañas. *Communications of the ACM*, Vol. 30, No. 7, pp. 612-616, 1987.
- [Margulies 83] Man's Role in Man-Machine Systems. *Automática*, Vol. 19, No. 6, pp. 605-612, 1983.
- [McCusker 90] Neural Networks and Fuzzy Logic, Tools of Promise for Controls. Tom McCusker. *Control Engineering*, May 1990, pp. 84-85.
- [McDermott 85] Reasoning about Plans. Drew McDermott. *Formal Theories of the Commonsense World*, pp. 269-318. ABLEX Publishing Corporation, Norwood, New Jersey, 1985.
- [McMahon 81] Fiber Optic Transducers. Donald H. McMahon, Arthur R. Nelson y William B. Spillman Jr. *IEEE Spectrum*, December 1981, pp. 24-29.
- [Meystel 88] Intelligent Control in Robotics. A. Meystel. *Journal of Robotic Systems*, Vol. 5, No. 4, pp. 269-308, 1988.
- [Meystel 89] Intelligent Control: A Sketch of the Theory. A. Meystel. *Journal of Intelligent and Robotic Systems* Vol.2 No.2&3, 1989. pp. 97-108.
- [Montmain 90] On Line Qualitative Interpretation of a Dynamic Symulation for Diagnosis. J. Montmain, L. Leyval y S. Gentil. *IMACS MIMS²*, 1990.
- [Moore 85] A Formal Theory of Knowledge and Action. Robert C. Moore. *Formal Theories of the Commonsense World*, pp. 319-358. ABLEX Publishing Corporation, Norwood, New Jersey, 1985.
- [Moore 90] Using DECwindows as a Control System Interface. Katrina Moore y Ralph Rio. *Control Engineering*, April 1990, pp. 90-92.

- [**Morant 89**] Hierarchical Expert System as Supervisory Level in Adaptive Control. F. Morant, P. Albertos, M. Martínez y J.J. Serrano. *IEEE International Symposium on Intelligent Control*, 1989.
- [**Muller 87**] Modula- -Prolog: A Programming Environment for Building Knowledge-Based Systems. C. Muller. *Knowledge Based Expert Systems in Industry*. Ellis Horwood, England 1987, pp. 98-110.
- [**Mylopoulos 86**] Knowledge Representation in the Software Development Process: A Case Study. J. Mylopoulos, A. Borgida, S. Greenspan, C. Meghini y B. Nixon. *Artificial Intelligence and Man-Machine Systems*. Springer-Verlag, 1986, pp. 23-44.
- [**Nakayama 90**] Model-based Automatic Programming for Plant Control. Yasuko Nakayama, Hiroyuki Mizutani y Katsuya Sadashige. *6th Conference on Artificial Intelligence Applications*, pp. 282-287, 1990.
- [**Neat 89**] Expert Adaptive Control for Drug Delivery Systems. Gregory W. Neat, Howard Kaufman y Rob J. Roy. *IEEE Control Systems Magazine*. June 1989, pp. 20-24.
- [**Neiman 86**] Consideraciones Tecnológicas sobre el Uso Industrial de los Sistemas Expertos. D. Neiman. *Comunicaciones Eléctricas*, Vol. 60, No. 2, pp. 185-189, 1986.
- [**Nistal 88**] Sistemas Expertos, Explotacion en plantas Industriales. José Manuel Nistal Bartolomé. *Mundo Electrónico*, Diciembre 1988.
- [**Nistal 89a**] El Diseño Optimo del Control de Procesos por Computador: Aplicaciones Prácticas. José Manuel Nistal Bartolomé. *Boletín COII*, 1989.
- [**Ng 90**] Model-Based, Multiple Fault Diagnosis of Time-Varying, Continuous Physical Devices. Hwee Tou Ng. *6th Conference on Artificial Intelligence Applications*, pp. 9-15, 1990.
- [**Novak 87**] Automatic Generation of Verbal Comments on Results of Mathematical Modelling. *Aproximate Reasoning in Intelligent Systems, Decision and Control*. Elie Sanchez y Lotfi A. Zadeh (Eds). Pergamon Press. pp. 55-68. 1987.
- [**gård 88**] Intelligent Alarm Handling. O. gård y E. Woods. En *Artificial Intelligence in Real-Time Control*. M.G. Rodd y G.J. Suski (eds). Pergamon Press, 1988.
- [**Ollero 89a**] Inteligencia Artificial: Aplicaciones recientes en Control y Supervisión de Procesos. Aníbal Ollero Baturone y A. García Cerezo. *Automática e Instrumentación*, Febrero 1989.
- [**Ollero 89c**] Expert Tuning of Automatic Controllers. A. Ollero, J. González, A. García y C. Fernández. *IEEE International Symposium on Intelligent Control*, 1989.
- [**Olsson 90**] An Architecture for Diagnostic Reasoning Using Causal Models. Jan Olsson. *IMACS MIMS²*, 1990.
- [**Oshuga 83**] Knowledge Based Man-Machine Systems. Setsuo Ohsuga. *Automática*, Vol. 19, No. 6, pp. 685-692, 1983.
- [**Pau 88**] Sensor Data Fusion. L. Pau. *Journal of Intelligent and Robotic Systems* Vol. 1, pp. 103-116. 1988.
- [**Perkins 90**] Adding Temporal Reasoning to Expert-System-Building Environments. *IEEE Expert*, February 1990. pp. 23-30.
- [**Permantier 88**] Representation of Inexact Engineering Knowledge about Real Time Systems. G. Permantier. En *Artificial Intelligence in Real-Time Control*. M.G. Rodd y G.J. Suski (eds). Pergamon Press, 1988.
- [**Peters 88**] *Advanced Structured Analysis and Design*. Lawrence Peters. Prentice Hall International. 1988.
- [**Peterson 87**] *Object Oriented Computing*. Gerald E. Peterson (Ed). IEEE Computer Society Press. 1987.
- [**Pew 83**] Perspectives on Human Performance Modelling. Richard W. Pew y Sheldon Baron. *Automática*, Vol. 19, No. 6, pp. 663-676, 1983.

- [Post 90] An Overview of Automated Reasoning. Stephen Post y Andrew P. Sage. *IEEE Systems, Man and Cybernetics*, Vol. 20 No. 1, pp. 202-224. 1990.
- [Procyk 79] A Linguistic Self Organizer Process Controller. T.J. Procyk y E.H. Mamdani. *Automatica*, Vol. 15, pp. 15-30. Pergamon Press Ltd. 1979.
- [Puente 88] An Integrated Multisensor Robot System. E.A. Puente, C.Balaguer, A.Barrientos y A. Yela. *SYROCO 88*. Karlsruhe, October 1988.
- [Radig 86] Design and Applications of Expert Systems. Bernd Radig. *Artificial Intelligence and Man-Machine Systems*. Springer-Verlag, 1986, pp. 45-61.
- [Raeth 90] *Expert Systems. A Software Methodology for Modern Applications*. Peter G. Raeth. IEEE Computer Society. 1990.
- [Ramamurthi 90] Real Time Expert System for Predictive Diagnostics and Control of Drilling Operation. K. Ramamurthi, Donald P. Shaver y Alice M. Agogino. *6th Conference on Artificial Intelligence Applications*, pp. 63-69, 1990.
- [Raman 88a] A Production System for Intelligent Monitoring Systems. Ravi S. Raman, Roy S. Nutter JR. y Y.V. Reddy JR. *IEEE Trans. on Industry Applications* Vol. 24 No. 5, September/October 1988, pp. 862-865.
- [Raman 88b] Knowledge Organization and Inference Engine for the WVU Face Decision Support System. Ravi S. Raman, Narendar A. Reddy, Y.V. Reddy, Roy S. Nutter JR., Steven Carrow y R. Larry Grayson. *IEEE Trans. on Industry Applications* Vol. 24 No. 5, September/October 1988, pp. 866-869.
- [Rauch-Hindin 89] *Aplicaciones de la Inteligencia Artificial en la Actividad Empresarial, la Ciencia y la Industria*. Wendy B. Rauch-Hindin. Diaz de Santos, 1989.
- [Rim 90] Transforming SYntactica Graphs into Semantic Graphs. Hae-Chang Rim, Jungyun Seo y Robert F. Simmons. Univ. of Texas at Austin AI Lab. AI90-127, 1990.
- [Rodd 88] *Artificial Intelligence in Real-Time Control*. M.G. Rodd y G.J. Suski (eds). Pergamon Press, 1988.
- [Ross 77] Structured Analysis (SA): A Language for Communicating Ideas. Douglas T. Ross. *IEEE Trans. on Software Engineering*, Vol. 3, No. 1, pp. 16-34, 1977.
- [Ross 85] Applications and Extensions of SADT. Douglas T. Ross. *IEEE Computer*, April 1985, pp. 25-34.
- [Rouse 83] Models of Human Problem Solving: detection, Diagnosis and Compensation for System Failures. William B. Rouse. *Automática*, Vol. 19, No. 6, pp. 613-625, 1983.
- [Rouse 88] On the Design of Man-Machine Systems: Principles, Practices and Prospects. William B. Rouse y William J. Cody. *Automática*, Vol. 24, No. 2, pp. 227-238, 1988.
- [Rubin 88] OFMspert: Inference of Operator Intentions in Supervisory Control Using a Blackboard Architecture. K.S. Rubin, P.M. Jones y C.M. Mitchell. *IEEE Trans. on Systems, Man and Cybernetics*, Vol. 18, No. 4, pp. 618-637.
- [Ruiz-Mier 89] A Hybrid Paradigm for Modeling of Complex Systems. Sergio Ruiz-Mier y Joseph Talavage. En *Artificial Intelligence, Simulation and Modelling*. Lawrence E. Widman, Kenneth A. Loparo y Norman R. Nielsen (eds). Wiley-Interscience, 1989.
- [Ruspini 82] Possibility Theory Approaches for Advanced Information Systems. Enrique H. Ruspini. *IEEE Computer*, September 1982. pp. 83-91.
- [Sacks 88] Automatic Analysis of Ordinary Differential Equations Using Piecewise Linear Approximations. Elisha Sacks. *MIT AI Lab. TR 1031*. 1988.
- [Saito 89] Application of Artificial Intelligence in the Japanese Iron and Steel Industry. Toyokazu Saito. *6th MMM-89*.
- [Sanchez 87] *Aproximate Reasoning in Intelligent Systems, Decision and Control*. Elie Sanchez y Lofti A. Zadeh (Eds). Pergamon Press. p. 195. 1987.

- [Savory 87] CONAD: A Knowledge-Based Configuration Adviser, Built Using Nixdorf's Expert System Shell TWAICE. S.E. Savory. *Knowledge Based Expert Systems in Industry*. Ellis Horwood, England 1987, pp. 111-116.
- [Schmidt 85] Partial Provisional Planning: Some Aspects of Commonsense Planning. Charles F. Schmidt. *Formal Theories of the Commonsense World*, pp 227-250. ABLEX Publishing Corporation, Norwood, New Jersey, 1985.
- [Shafer 76] *A Mathematical Theory of Evidence*. Glenn Shafer. Princeton University Press, 1976.
- [Sheridan 83b] Measuring, Modeling and Augmenting Reliability of Man-Machine Systems. Thomas B. Sheridan. *Automática*, Vol. 19, No. 6, pp. 637-645, 1983.
- [Shlaer 88a] The Object-Oriented Method for Analysis. Sally Shlaer, Stephen J. Mellor, Deborah Olsen y Wayne Hywari. Proc. 10th Structured Development Forum, 1988.
- [Shlaer 88b] Understanding Object-Oriented Analysis. Sally Shlaer y Stephen J. Mellor. *HP Design Center*, No. 3, pp. 8-13, 1988.
- [Shlaer 89a] An Object-Oriented Approach to Domain Analysis. Sally Shlaer y Stephen J. Mellor. En *Software Engineering Notes*, ACM Press, 1989.
- [Shlaer 89b] Real-Time Recursive Design. Sally Shlaer y Stephen J. Mellor. *Project Technology Inc.*, 1989.
- [Shoham 88] *Reasoning About Change*. Yoav Shoham. MIT Press. 1988.
- [Siler 87] Functional Requirements for a Fuzzy Expert System Shell. W. Siler, J.J. Buckley y D. Tucker. *Approximate Reasoning in Intelligent Systems, Decision and Control*. Elie Sanchez y Lofti A. Zadeh (Eds). Pergamon Press. pp. 21-32. 1987.
- [Simmons 83] Representing and Reasoning about Change in Geological Interpretation. Reid G. Simmons. *MIT AI Lab. TR 749*. 1983.
- [Steelman 82] *Distributed Cement Plant Control - The Intelligent Approach*. D. Steelman, *IEEE Trans. on Industri Applications*, Vol. 18, No. 2, pp. 192-198, 1982.
- [Stein 83] Human Display Monitoring and Failure Detection: Control Theoretic Models and Experiments. W. Stein y P.H. Wewerinke. *Automática*, Vol. 19, No. 6, pp. 711-718, 1983.
- [Struss 87] Multiple Representation of Structure and Function. Peter Struss. En *Expert Systems in Computer Aided Design*. John S. Gero (Ed.), pp. 57-92. North Holland, 1987.
- [Suganuma 89] Learning Control and Knowledge Representation. Yoshinori Suganuma y Masami Ito. *Journal of Intelligent and Robotic Systems* Vol.2 No.2&3, 1989. pp. 337-358.
- [Sugaya 87] A Prolog Frame System for Knowledge-Based Design and Diagnosis. H. Sugaya. *Knowledge Based Expert Systems in Industry*. Ellis Horwood, England 1987, pp. 117-129.
- [Tamisier 90] A Software Tool for the Automatic Operation of Behavioral Simulations of Technological Systems. S. Tamisier y J.L. Petit. *IMACS MIMS²*, 1990.
- [Tate 87] Application of Knowledge-Based Planning Systems. A. Tate. *Knowledge Based Expert Systems in Industry*. Ellis Horwood, England 1987, pp. 130-144.
- [Taunton 88] Empleo de Sistemas Expertos en Línea en Control de Procesos. J.C. Taunton. *Automática e Instrumentación*, Marzo 1988.
- [Tenteris 90] Structural Modeling and Fuzzy Simulation of Metallurgical Processes. Jänis K. Tenteris, Esko K. Juuso y Kaulo Leiviskä. *IMACS MIMS²*, 1990.
- [Terano 83] Human Reliability and Safety evaluation of Man-Machine Systems. T. Terano, Y. Murayama y N. Akiyama. *Automática*, Vol. 19, No. 6, pp. 719-722, 1983.
- [Theuretzbacher 86] Tecnología de sistemas expertos para sistemas de seguridad crítica en tiempo real. N. Theuretzbacher. *Comunicaciones Eléctricas*, Vol. 60, No. 2, pp. 147-153, 1986.

- [**Tichy 87**] What can Software Engineers Learn from Artificial Intelligence?. Walter F. Tichy. *IEEE Computer*, November 1987, pp. 43-54.
- [**Trillas 80**] *Conjuntos Borrosos*. Enrique Trillas. Vicens-Vives, 1980.
- [**Tse 89**] An Intelligent Control System. Edison Tse and Michael Fehling. *IEEE International Symposium on Intelligent Control*, 1989.
- [**Valavanis 89**] Hardware and Software for Intelligent Robotic Systems. Kimon P. Valavanis y Peter H. Yuan. *Journal of Intelligent and Robotic Systems*, Vol. 1, pp. 343-373, 1989.
- [**Van der Rhee 90**] Knowledge Based Fuzzy Control of Systems. Floor Van der Rhee, Hans R. Van Nauta Lemke y Jaap G. Dijkman. *IEEE Trans. on Automatic Control*, Vol. 35, No. 2, pp. 148-155, 1990.
- [**Vangheluwe 90**] Application of a Multifaceted Modelling Methodology: An Example in Physiology. Hans Vangheluwe, Jorge Barreto y G.C. Vansteenkiste. *IMACS MIMS²*, 1990.
- [**Vescovi 90**] A new Model for Qualitative Reasoning Representing Time Explicitly. Marcos R. Vescovi. *IMACS MIMS²*, 1990.
- [**Vitins 87**] A Prototype Expert System for Configuring Technical Systems. M. Vitins. *Knowledge Based Expert Systems in Industry*. Ellis Horwood, England 1987, pp. 145-158.
- [**Völckers 86**] Dynamic Plannig and Time-Conflict Resolution in Air Traffic Control. Uwe Völckers *Artificial Intelligence and Man-Machine Systems*. Springer-Verlag, 1986, pp. 175-197.
- [**Wahlster 86**] The Role of Natural Language in Advanced Knowledge-Based Systems. Wolfgang Wahlster *Artificial Intelligence and Man-Machine Systems*. Springer-Verlag, 1986, pp. 62-83.
- [**Ward 86**] The Transformation Schema: An Extension of the Data Flow Diagram to Represent Control and Timing. Paul T. Ward. *IEEE Trans. on Software Engineering*, Vol. 12, No. 2, pp. 198-210, 1986.
- [**Watanabe 89**] Intelligent Control in the Hierarchy of Automatic Manufacturing Systems. Tohru Watanabe. *Journal of Intelligent and Robotic Systems* Vol.2 No.2&3, 1989. pp. 171-186.
- [**Weiss 84**] *A practical Guide to Designing Expert Systems*. Sholom Weiss y Casimir Kulikovski. Rowman & Allanheld, 1984.
- [**Weld 84**] Switching Between Discrete and Continuous Process Models to Predict Genetic Activity. Daniel S. Weld. *MIT AI Lab. TR 793*. 1984.
- [**Weld 86**] The Use of Aggregation in Causal Simulation. Daniel S. Weld. *Artificial Intelligence*, Vol. 30, pp. 1-34, 1986.
- [**Weld 88**] Theories of Comparative Analysis. Daniel S. Weld. *MIT AI Lab. TR 1035*. 1988.
- [**Wewerinke 83**] Model of Human Observer and Decision Maker - Theory and Applications. P.H. Wewerinke. *Automática*, Vol. 19, No. 6, pp. 693-696, 1983.
- [**Williams 84**] Qualitative Analysis of MOS Circuits. B.C Williams. *Artificial Intelligence Vol.24*, pp. 281-346, 1984.
- [**Winston 84**] *Artificial Intelligence. 2nd Edition*. Patrick Henry Winston. Addison Wesley 1984. 524 p.
- [**Winter 86**] Artificial Intelligence in Man-Machine Systems and Automation. Heinz Winter. *Artificial Intelligence and Man-Machine Systems*. Springer-Verlag, 1986, pp. 1-22.
- [**Wittig 87**] Expert Systems for Real-Time Applications. Thies Wittig. *Seminar on Expert Systems and Applications*. UIMP. June-July 1987.
- [**Wuwongse 83**] An Interactive System for Supporting Multiobjective Decision Making. V. Wuwongse, S. Kobayashi y A. Ichikawa. *Automática*, Vol. 19, No. 6, pp. 697-702, 1983.
- [**Yadav 90**] Comparison of Analysis Techniques for Information Requirement Determination. S.B. Yadav, R.R. Bravoco, A.T. Chatfield y T.M. Rajkumar. *Communications of the ACM*, Vol. 31, No. 9, pp. 1090-1097, 1988.

- [**Yamakawa 87**] Fuzzy Hardware Systems of Tomorrow. T. Yamakawa. *Approximate Reasoning in Intelligent Systems, Decision and Control*. Elie Sanchez y Lofti A. Zadeh (Eds). Pergamon Press, pp. 1-20, 1987.
- [**Yazdani 86b**] Intelligent Tutoring Systems: An Overview. Masoud Yazdani. *Expert Systems*, Vol. 3, No. 3, pp. 154-162, July 1986.
- [**Yoon 86**] Fiber-Optic Gyroscopes. B. Yoon Kim y Herbert John Shaw. *IEEE Spectrum*, March 1986, pp. 54-60.
- [**Zadeh 86**] Outline of a Computational Approach to Meaning and Knowledge Representation based on the Concept of a Generalized Assignment Statement. Lofti A. Zadeh *Artificial Intelligence and Man-Machine Systems*. Springer-Verlag, 1986, pp. 198-211.
- [**Zadeh 89**] Syllogistic Reasoning in Fuzzy Logic and its Application to Usuality and Reasoning with Dispositions. Lofti A. Zadeh. *IEEE Transactions on Systems, Man and Cybernetics*, Vol. 15, No. 6, pp. 754-763. November/December 1985.
- [**Zeigler 89**] DEVS Representation of Dynamical Systems: Event-Based Intelligent Control. Bernard P. Zeigler. *Proceedings of the IEEE*, Vol. 77, No. 1, pp 72-80, 1989.
- [**Zhang 89**] The System Entity Structure: Knowledge Representation for Simulation Modeling and Design. Guoqing Zhang y Bernard P. Zeigler. En *Artificial Intelligence, Simulation and Modelling*. Lawrence E. Widman, Kenneth A. Loparo y Norman R. Nielsen (eds). Wiley-Interscience, 1989.
- [**Zhang 90b**] A Fault Diagnosis System with Self Reasoning Facility Based on Qualitative Modelling. J. Zhang, P.D. Roberts y J.E. Ellis. *IMACS MIMS²*, 1990.

Apéndice D

Parámetros de los conjuntos borrosos

PARAMETROS DE LOS CONJUNTOS BORROSOS				
	\square	S	Z	S-Z
P_1	P	$-\infty$	P_1	P_1
P_2	∞	P	P_2	P_2
β_1	β	-	β_1	β_1
β_2	-	β	β_2	β_2

Tabla D.1: Parámetros de los conjuntos borrosos

Reference: ASL-T-1990-001 v 1.0 Final
Title: Arquitectura de Control
Inteligente de Procesos
Subtitle: El sistema CONEX
URL: <http://www.aslab.org/public/documents/ASL-T-1990-001.pdf>
Date: 1990

© 1990 ASLab

Autonomous Systems Laboratory

UNIVERSIDAD POLITÉCNICA DE MADRID
c/José Gutiérrez Abascal, 2
Madrid 28006 (Spain)



www.aslab.org