BIBLIOTECA VIRTUAL MIGUEL DE CERVANTES

Universidad de Valladolid

Un nuevo marco matricial para la implementación de inferencia borrosa aplicados al procesado de información no numérica

Santiago Aja Fernández

Tesis de Doctorado

Escuela Técnica Superior de Ingenieros de Telecomunicación

Director: Dr. D. Carlos Alberola López

2003

UNIVERSIDAD DE VALLADOLID

Departamento de Teoría de la Señal y
Comunicaciones e Ingeniería Telemática

Escuela Técnica Superior de Ingenieros de Telecomunicación

TESIS DOCTORAL

BIBLIOTECA VIRTUAL
MIGUEL DE CERVANTES

Un nuevo marco matricial para la implementación eficiente de esquemas de inferencia borrosa aplicados al procesado de información no numérica

Autor: Santiago Aja Fernández
Director: Carlos Alberola López

Julio 2003

"Muchas veces me entran ganas
de gritar contra las universidades,
y de ponerme a despotricar con todas mis fuerzas
como loco y fuera de juicio.
Dirigiría a voces mi discurso
contra los que se preocupan de saber mucho
antes que de hacer que su ciencia
aproveche a los que tienen necesidad de ella."

Francisco Javier, 1544

"La facultad de juzgar bien y de distinguir
lo verdadero de lo falso, que es propiamente
lo que llamamos buen sentido o razón,
es por naturaleza igual en todos los hombres;
y, por lo tanto, la diversidad de nuestras opiniones
no procede de que unos sean más racionales que otros,
sino tan sólo de que dirigimos nuestros pensamientos
por caminos distintos y no consideramos las mismas cosas."

R. Descartes. Discurso del Método, 1637.

CONTENIDOS

ÍNDICE DE FIGURAS. **xi**

1 INTRODUCCIÓN **1**

1.1 Introducción 1

1.1.1 Planteamiento del problema 1

1.1.2 Algunos campos de aplicación 3

1.2 Objetivos 5

1.2.1 Estructura de la memoria 6

2 INTRODUCCIÓN A LA LÓGICA BORROSA **9**

2.1 Conjuntos Borrosos 9

2.1.1 Revisión de la Teoría Clásica de Conjuntos 9

2.1.2 Extensión a conjuntos borrosos 13

2.1.3 Definiciones básicas sobre conjuntos borrosos 16

2.1.4 Operaciones entre conjuntos borrosos 18

2.1.5 Propiedades de los conjuntos borrosos 22

2.1.6 Principio de extensión 23

2.1.7 α-cortes 24

2.1.8 Números borrosos 24

2.1.9 Modificadores lingüísticos 25

2.2 Lógica Borrosa 26

2.2.1 Variables lingüísticas 26

2.2.2 Relaciones borrosas: operadores lógicos 27

2.2.3 Composición de relaciones borrosas: procedimientos de inferencia borrosa 29

2.2.4 Entradas al sistema y activación de reglas 33

2.3 Sistemas de Lógica Borrosa 34

2.3.1 Tipos de sistemas basados en reglas borrosas 34

2.3.2 Elementos 36

2.3.3 Base de conocimiento 37

2.3.4 Motor de inferencias. Inferencia borrosa 38

2.3.5 Borrosificación y desborrosificación 39

2.3.6 Sistemas Borrosos Aditivos 40

BIBLIOTECA VIRTUAL MIGUEL DE CERVANTES

3 CÁLCULO LINGÜÍSTICO 43
3.1 Introducción 43
3.2 Definición de CWW 45
3.2.1 Teoría Computacional de las Percepciones 46
3.3 Variable lingüística 48
3.4 Lógicas Borrosas 49
3.5 Grafos Borrosos 50
3.5.1 Operaciones sobre grafos borrosos 51
3.6 Granulación de la Información 54
3.6.1 Gránulos 54
3.6.2 Caracterización de la granularidad 56
3.6.3 Captura de la granularidad. Modelos de semiótica relacional 56
3.7 Restricciones Generalizadas 57
3.7.1 Restricciones borrosas y formas canónicas 57
3.7.2 Restricciones generalizadas 60
3.7.3 Propagación de restricciones borrosas y reglas de inferencia 63
3.8 Teoría de la Posibilidad 67
3.8.1 El principio del mínimo específico 67
3.8.2 El mecanismo de combinación/proyección 68
3.9 Conclusiones 70

4 NUEVO MARCO DE TRABAJO MATRICIAL 71
4.1 Representación Granular de la Información 71
4.1.1 Discusión previa 71
4.1.2 Definición de conjuntos borrosos mediante el grado de solapamiento 72
4.1.3 Matrices de activación para conjuntos *definidos* 76
4.1.4 Modificadores lingüísticos 76
4.1.5 Generalización de los gránulos 77
4.2 Inferencia Borrosa Matricial 78
4.2.1 Estado del arte 78
4.2.2 Definición vectorial de entradas borrosas. Modelo SAM 80
4.2.3 SAM con una entrada y una salida 81
4.2.4 SAM de 2 entradas y una salida 83
4.2.5 SAM de múltiples entradas y una salida 88
4.2.6 Extensión a entradas numéricas 91
4.2.7 Reglas con múltiples consecuentes 94
4.2.8 Cálculo de centroides 95
4.2.9 Modificadores lingüísticos 95
4.2.10 Evaluación de la complejidad 97
4.3 Extensión a Sistemas No-lineales 99

4.3.1 Matrices de transición para un sistema genérico 100
4.4 Sistemas Borrosos Realimentados (FFS) 101
4.4.1 Matrices de transición para un sistema borroso realimentado 102
4.4.2 Análisis de un sistema realimentado en régimen estacionario 104
4.4.3 Generalización de los resultados 107
4.4.4 Inferencia recursiva 108
4.4.5 Sistema realimentado no lineal 108
4.5 Reglas de Inferencia y Composición de Matrices 110

5 APLICACIÓN A ESQUEMAS DE SEGUIMIENTO 117
5.1 Algoritmo MHT Borroso 117
5.1.1 Introducción 117
5.1.2 El algoritmo de Seguimiento mediante Múltiples Hipótesis (MHT) 118
5.1.3 MHT con entradas de texto 120
5.1.4 Implementación borrosa 121
5.1.5 Implementación del sistema borroso usando matrices de transición 130
5.1.6 Ejemplo de funcionamiento 133
5.2 Detector Borroso de Maniobra para un Filtro de Kalman en Entorno de Seguimiento Estéreo 134
5.2.1 Planteamiento del problema 134
5.2.2 Detector borroso de maniobra 135
5.2.3 Ejemplos prácticos y resultados 137

6 APLICACIÓN AL CÁLCULO AUTOMÁTICO DEL GRADO DE MADURACIÓN ÓSEA EN LA INFANCIA 143
6.1 Introducción 143
6.2 Cálculo de la Maduración Ósea en la Infancia 143
6.2.1 Radiografía de la mano 144
6.3 Automatización del Método TW2 147
6.3.1 Trabajos previos 147
6.3.2 Una nueva aproximación 149
6.4 Construcción de un Clasificador para TW2 150
6.4.1 Definición de características. Ejemplo del radio 151
6.4.2 Modelado borroso de las características 154
6.4.3 Elección del clasificador. ID3 borroso 155
6.4.4 Definición del conjunto de entrenamiento 157
6.4.5 Construcción de los árboles de decisión 157
6.5 Implementación Matricial 158
6.5.1 Inclusión del conocimiento a posteriori 160
6.6 Experimentos y Resultados 160

BIBLIOTECA VIRTUAL MIGUEL DE CERVANTES

6.7 Conclusiones 161

7 CONCLUSIONES, APORTACIONES Y LÍNEAS FUTURAS 165
7.1 Conclusiones 165
7.2 Aportaciones 167
7.3 Líneas futuras 168

GLOSARIO DE ABREVIATURAS. 169

BIBLIOGRAFÍA. 171

A DESARROLLOS AUXILIARES DEL CAPíTULO 4 183
A.1 Ecuaciones 183
A.1.1 Sobre la ecuación (4.15) 183
A.1.2 Sobre la ecuación (4.30) 183
A.2 Matriz de Transición de un Modificador en un SAM de Dos Entradas 185

B EXTENSIÓN DE LA METODOLOGÍA FITM A OPERADORES NO LINEALES 187
B.1 Definiciones de Operaciones Vectoriales y Matriciales 187
B.2 Sistema Borroso con una Entrada y una Salida 189
B.3 Sistema Borroso de 2 Entradas y una Salida 192
B.4 Sobre la Asociatividad del Operador Matricial $\times_{\cup\cap}$ 194
B.5 Sobre la Ecuación (b.32) 195

C APLICACIÓN DE LA METODOLOGÍA FITM AL CÁLCULO DE ÁRBOLES DE DECISIÓN BORROSOS 197
C.1 Introducción a los Árboles de Decisión 197
C.1.1 El algoritmo ID3 198
C.1.2 Descripción del algoritmo ID3 199
C.2 Algoritmo ID3 Borroso 201
C.3 Aplicación del Método FITM al Cálculo de Árboles de Decisión 203
C.3.1 Procesado en árbol 205
C.3.2 Procesado matricial 206
C.4 Completado de Reglas 209

D MHT BORROSO 213
D.1 Revisión de los Sistemas Borrosos Realimentados 213
D.2 Discriminación de Hipótesis del Algoritmo MHT borroso 215
D.3 Matrices del sistema MHT borroso 222
D.3.1 Matrices del bloque de compatibilidad lógica 222

BIBLIOTECA VIRTUAL MIGUEL DE CERVANTES

D.3.2 Matrices del bloque de compatibilidad física y lógica 224
D.3.3 Matrices del bloque de actualización de probabilidades 226

E FILTRO BORROSO DE KALMAN EN ENTORNOS DE SEGUIMIENTO ESTÉREO 229
E.1 Introducción 229
E.2 Esquema de seguimiento adoptado 230
E.3 Trabajos Previos 232
E.3.1 Esquemas de seguimiento basados en lógica borrosa 233
E.3.2 Detección de Maniobras y Filtrado Adaptativo 233
E.4 Filtro de Kalman con Control Borroso 234
E.4.1 Normalización del Error 235
E.4.2 Motor de Inferencias Borroso. Adaptación de la matriz de covarianza de ruido de planta 237
E.5 Ejemplos prácticos y resultados 239
E.5.1 Experimento 1: Trayectorias Rectilíneas Uniformes 240
E.5.2 Experimento 2: Trayectorias Rectilíneas Uniformes de Baja Velocidad 242
E.6 Conclusiones 243

F APÉNDICE AL MÉTODO TW2 245
F.1 Características de los 13 huesos RUS 245
F.1.1 Radio 245
F.1.2 Cúbito 246
F.1.3 Metacarpiano I 246
F.1.4 Metacarpianos III y V 247
F.1.5 Falange Proximal I 247
F.1.6 Falanges Proximales III y V 247
F.1.7 Falanges Medias III y V 248
F.1.8 Falange Distal I 248
F.1.9 Falanges Distales III y V 248
F.2 Características de los huesos para el método RUS 249
F.3 Datos de entrenamiento para los distintos huesos 251
F.3.1 Radio 251
F.3.2 Cúbito 252
F.3.3 Metacarpiano I 253
F.3.4 Metacarpianos III y V 254
F.3.5 Falange proximal I 254
F.3.6 Falange proximal III y V 255
F.3.7 Falange media III y V 256
F.3.8 Falange distal I 257

F.3.9 Falange distal III y V 258

RESUMEN

BIBLIOTECA VIRTUAL MIGUEL DE CERVANTES

ÍNDICE DE FIGURAS.

Capítulo 1

Capítulo 2

2.1 Funciones de pertenencias típicas: a) Gaussianas, b) Triangulares c) Trapezoidales. 15
2.2 Función Pseudo-trapezoidal (PTS) 15
2.3 Ejemplo de conjuntos borrosos 16
2.4 Operaciones básicas entre conjuntos borrosos. Definición estándar 18
2.5 Números Borrosos "aproximadamente 4" y "aproximadamente 6" 25
2.6 Ejemplo de modificadores aplicados sobre un conjunto borroso 26
2.7 Modificador *muy* aplicado sobre el conjunto GRANDE 26
2.8 Definición de la variable lingüística *Temperatura* 27
2.9 Producto cartesiano de dos universos de discurso: X e Y. 28
2.10 Representación Gráfica del *método de interpolación* 32
2.11 Sistema de Lógica Borrosa 37
2.12 Diagrama de bloques de un *modelo aditivo estándar* 40

Capítulo 3

3.1 Estructura conceptual de la Teoría Computacional de Percepciones 47
3.2 Representación de una función y su grafo borroso. 50
3.3 Algunas operaciones básicas sobre funciones y relaciones. (a) Evaluación y asignación. (b) Proyección. (c) Composición. (d) Intersección. (e) Combinación. (f) Maximización 51
3.4 Intersección de los grafos borrosos f^* y g^*. 52
3.5 Intersección de f^* y $\overline{A}$ 53
3.6 Relación con atributos borrosos 54
3.7 Representación de un gránulo 54
3.8 Restricción de grafo borroso. 62

Capítulo 4

4.1 Definición de la función de pertenencia del término *Joven* 71
4.2 Definición clásica de 3 conjuntos borrosos 72

4.3 Definición de 3 conjuntos mediante intersecciones. 73
4.4 Múltiples intersecciones en la definición de conjuntos borrosos. 74
4.5 Definición mediante gránulos de la variable lingüística *Altura* 75
4.6 Definición de la variable lingüística *Altura* mediante conjuntos trapezoidales 76
4.7 El modificador lingüístico *Muy* aplicado sobre un conjunto borroso (arriba) y sobre un gránulo borroso (abajo) 77
4.8 Sistema borroso de dos bloques 81
4.9 Sistema borroso genérico con dos entradas y una salida 84
4.10 Sistema borroso con múltiples entradas y una salida 90
4.11 Gránulos de la variable *Vehículo* 93
4.12 Conjuntos de la variable *Error* 93
4.13 FFS de dos entradas y una salida. 101
4.14 Conjuntos borrosos de la variable lingüística $\mathcal{L}$ 102
4.15 Gránulos borrosos de la variable lingüística $\mathcal{L}$ 102
4.16 Pixels de entrada 108
4.17 Sistema borroso realimentado 108
4.18 Proyecciones sobre la matriz **G** 113

Capítulo 5

5.1 Propuesta de arquitectura del sistema de seguimiento con entradas de texto 120
5.2 Interpretación de la variable lingüística *Tipo de Vehículo* usando conjuntos borrosos 123
5.3 Interpretación de la variable lingüística *Tipo de Vehículo* usando gránulos borrosos 123
5.4 Interpretación alternativa de la variable lingüística *Tipo de Vehículo* usando gránulos borrosos 124
5.5 Variable Lingüística *Error de Predicción* 125
5.6 Variable Lingüística *probabilidad* 126
5.7 Esquema del proceso de obtención de *etiquetas de probabilidad* 130
5.8 Diagrama de estados para el reinicio 136
5.9 Sistema borroso realimentado 136
5.10 Funciones de pertenencia de la variable *Error* 136
5.11 Funciones de pertenencia de la variable *Historia* 137
5.12 Trayectorias del experimento 138
5.13 Comparación del FKCP convencional con el filtro con control borroso. a) Trayectoria T1 b) Trayectoria T2 c) Trayectoria T3 d) Trayectoria T4 139
5.14 Comparación del FKCP convencional con el filtro con control borroso. a) Error de Predicción b) Error de Filtrado 139

5.15 Comparación de los decisores de maniobra borroso y clásico. Trayectoria T4 (log) 140
5.16 Comparación de los decisores de maniobra borroso y clásico. Trayectoria T2 140
5.17 Comparación de los decisores de maniobra borroso y clásico. a) Trayectoria T6 b) Trayectoria T6 (log) 141

Capítulo 6

6.1 Nomenclatura de las áreas epifisarias 145
6.2 Nomenclatura de las direcciones para los huesos de la mano 145
6.3 Esquema del esqueleto de la mano y muñeca. 146
6.4 Esquema del proceso de automatización del método TW2 149
6.5 Clasificador global formado por un microclasificador para cada hueso 150
6.6 Estadios TW2 para el radio 151
6.7 Variable lingüística *Epífisis* 155
6.8 Variable lingüística *Forma* 155
6.9 Variable lingüística *Superficies* 155
6.10 Variable lingüística *Diámetros* 156
6.11 Inclusión de la información a posteriori dentro del método 161
6.12 Árbol del radio 163

Capítulo 7

Apéndice A

Apéndice B

Apéndice C

C.1 Ejemplos de árboles de decisión generados por a) ID3 b) CART. 197
C.2 Etiquetado de uniones 203
C.3 Árbol borroso de ejemplo 204
C.4 Múltiples intersecciones en la definición de conjuntos borrosos. 206

Apéndice D

D.1 Sistema en bucle cerrado (Tong) 213
D.2 Sistema Borroso de Control (Chen-Tsao) 214
D.3 Sistema borroso dinámico (Kang) 215
D.4 Componentes de la Variable Lingüística *Probabilidad* 216

D.5 Ilustración de una situación ambigua. a) Existe una pista asociada a un vehículo. b) Llega una nueva observación del mismo vehículo c) Hipótesis 1 (correcta): La pista se alarga con la nueva observación. d) Hipótesis 2 (errónea): Se crea una nueva pista. Se asume que hay presentes dos vehículos en el escenario. 217

Apéndice E

E.1 Sistema de visión estereoscópica. a) Proyección sobre el plano xz. b) Proyección sobre el plano yz. 230
E.2 Funciones de pertenencia para la entrada borrosa *Error*. 238
E.3 Funciones de pertenencia para la entrada borrosa *Incremento del Error*. 238
E.4 Trayectorias del primer experimento 240
E.5 Comparación del FKCP clásico con el borroso. a) Trayectoria T1 b) Trayectoria T2. c) Trayectoria T3. d) Trayectoria T4. e) Trayectoria T5. f) Trayectoria T6. 241
E.6 Trayectorias usadas en el segundo experimento. 242
E.7 Comparación del FKCP convencional con el filtro con control borroso. a) Trayectoria T7 b) Trayectoria T8 243
E.8 Comparación entre el FKCP clásico y borroso para una trayectoria de alta velocidad (T9) 243

Apéndice F

1

INTRODUCCIÓN

BIBLIOTECA VIRTUAL MIGUEL DE CERVANTES

1.1 INTRODUCCIÓN

1.1.1 Planteamiento del problema

Numerosos problemas de la vida cotidiana se pueden plantear en términos de cantidades numéricas precisas, o con imprecisión cuantificable de forma natural; para tales problemas, asimismo, se espera una respuesta igualmente en términos numéricos. Sería el caso, por ejemplo, del seguimiento del peso o la talla de un recién nacido por parte de su pediatra. No obstante, existen otros problemas en los que, si bien se puede proporcionar una descripción numérica, incluso con determinado grado de incertidumbre, resulta más natural el empleo de términos lingüísticos. En el ejemplo del recién nacido, basta saber si éste ha sido grande, mediano o pequeño para conseguir encontrar una determinada prenda que se ajuste a sus necesidades.

Las descripciones lingüísticas están constituidas por palabras, y éstas, tal y como las percibimos las personas, tienen connotaciones borrosas; en muchas ocasiones no está claro dónde empieza y dónde termina su significado. Incluso en palabras que a primera vista parecen *unívocamente definidas* puede aparecer cierta dosis de ambigüedad. Por ejemplo, la palabra *circular* está asociada a la forma de un objeto, pero ¿existe un límite preciso entre un objeto *circular* y uno *elíptico*? Matemáticamente no habría posibilidad de discusión, pero en el lenguaje común, podemos denominar a un objeto *circular* incluso cuando no todos los puntos de su perímetro equidisten exactamente de su centro.

Las palabras que usamos están relacionadas con las percepciones que tenemos, y nuestras percepciones son intrínsecamente imprecisas. Según Zadeh [171], las percepciones son *borroso-granulares*, esto es:

1. Las percepciones tienen un cierto grado de imprecisión o de borrosidad, dado que los valores percibidos no están perfectamente definidos. Por ejemplo, para que etiquetemos una forma de *circular*, su frontera no tiene por qué cumplir estrictamente la ecuación de la circunferencia. Además, conceptos como *cerca*, *lejos*, *alto*, etc, son muy dependientes de la subjetividad del sujeto.

2. Por otro lado, podemos decir que las percepciones son granulares, ya que de alguna manera agrupamos aquellas percepciones parecidas bajo un mismo nombre o etiqueta; el término *alto*, referido a personas, constituiría un conjunto en el que se agrupan alturas tales como "2,00 m.", "1,98 m.", "2,06 m.", etc. Esta agrupación se hace siguiendo diversos criterios tales como indistinguibilidad, similitud, proximidad o funcionalidad.

Cuando un problema se resuelve usando información de carácter lingüístico, necesitamos técnicas computacionales que hagan uso de operadores lingüísticos. Estas técnicas se reúnen bajo el nombre de *Cálculo con palabras*, o *Cálculo lingüístico* (en adelante se usará el acrónimo CWW del término inglés *Computing with Words* [166]), introducido por Zadeh [160] en 1973.

BIBLIOTECA VIRTUAL

Este razonamiento es vital para muchos problemas a resolver por las Tecnologías de la Información como, por ejemplo, en toma de decisiones en entornos médicos (un ejemplo de ello se propone en el capítulo 6), búsqueda de información en bases de datos, seguimiento de móviles basado en lenguaje natural [7], control del flujo del tráfico aéreo e incluso detección de minas terrestres [61]. Asimismo, la metodología que se desarrolla en base a aplicaciones de estas características puede ser útil para otros problemas, a priori perfectamente adaptados para su resolución mediante métodos no basados en CWW. Un ejemplo de ello se presenta también en el capítulo 5.

Existen dos imperativos para usar CWW: primero, se empleará CWW cuando la información sea demasiado imprecisa como para trabajar con números y, segundo, cuando exista una cierta tolerancia a la imprecisión que pueda ser explotada para nuestros fines. Por ejemplo, en el caso de querer saber *cuánto podríamos tardar en atravesar la ciudad en coche en la hora punta*, la información de la que disponemos es imprecisa: no se especifica qué recorrido se ha de seguir, ni cuál va a ser la velocidad del coche ni la densidad de tráfico. La respuesta que se espera tampoco tiene que ser exacta. Es irrelevante que sean 18:23 minutos que sean 20:05. Lo que interesa saber es que se puede tardar entre 15 y 30 minutos, dependiendo del día, y más específicamente en torno a 20 minutos.

La aproximación lingüística, aun siendo menos precisa que la numérica, presenta algunas ventajas [46]: La descripción lingüística es fácilmente entendida por los seres humanos, incluso cuando los conceptos son abstractos o el contexto es cambiante. Más aún, disminuye el efecto del ruido, ya que cuanto más refinada es la escala con la que trabajemos, más sensible es ésta al ruido.

El uso de conjuntos borrosos es claramente el centro de CWW ya que proveen un medio para modelar la vaguedad subyacente a la mayoría de los términos de lenguaje natural. Desde el punto de vista de modelado permiten dividir variables continuas en gránulos discretos de modo que haya una interpolación inherente a las fronteras de partición [76].

Sin embargo, desde un punto de vista operativo estos gránulos se han definido tradicionalmente como conjuntos borrosos asociados a una variable base. Esto quiere decir que para definir ciertos conceptos es necesario acudir a una variable (en casos incluso ficticia) sobre la que los conjuntos toman valores. Aunque esta definición de los conjuntos borrosos ha demostrado su utilidad en numerosos campos de aplicación, como el de control, es a todas luces inadecuado para el modelado de conceptos lingüísticos no directamente asociados con variables numéricas. Surge por lo tanto la necesidad de una nueva definición de los gránulos borrosos que tenga en cuenta exclusivamente las relaciones entre dichos gránulos y que no lleve necesariamente asociada una variable base. Tal definición constituye uno de los ejes centrales de la presente tesis.

Así mismo, es bien sabido que los procesos de inferencia borrosa sufren de la llamada *maldición de la dimensionalidad* [70], esto es, necesitan de una gran cantidad de reglas para aproximar la mayoría de las funciones, y este número de reglas crece exponencialmente con el número de variables de entrada y salida. Un mayor número de variables hará el modelo más realista, aunque el precio a pagar es un aumento del coste computacional. Algunos autores han tratado de reducir este coste computacional inherente a los razonamientos borrosos con diversas técnicas, como la elección de un subconjunto óptimo de reglas [53, 126, 156], el cambio de los operadores de inferencia [80] o un cambio en la configuración de las reglas [21].

En esta tesis se abordará también este problema, si bien desde otra perspectiva. La reducción del coste computacional se realizará cambiando la algorítmica propia del método, manteniendo la configuración y el tamaño de la base de reglas, pero precalculando una serie de parámetros en forma matricial que ahorrarán operaciones en el motor de inferencias. El modelado granular de la base de reglas, como se verá, encajará de forma directa en este nuevo paradigma de procesado.

El resultado neto será un marco matricial que permita una inferencia borrosa más eficiente. Los clásicos motores de inferencia borrosa, cuando las entradas son variables no numéricas (definidas mediante conjuntos borrosos o mediante gránulos), pueden modelarse como un proceso en el que existen unas matrices de transición que permiten el paso de los espacios de entrada al espacio de salida de un modo rápido y eficiente. Consecuencia de este nuevo marco será el aumento de la velocidad, disminución de la complejidad del motor de inferencias y la reducción notable de las estructuras de datos intermedias a emplear durante el proceso.

1.1.2 Algunos campos de aplicación

Las técnicas de procesado de información no numérica que se han de definir tienen su aplicación práctica en diversos problemas reales. Algunos de ellos serán examinados en la tesis. Concretamente:

- **Esquemas de seguimiento:** En primer lugar se usará en una versión borrosa del clásico algoritmo de seguimiento MHT (*Multiple Hypothesis Tracking*, Seguimiento mediante Múltiples Hipótesis [108]), implementado en la aplicación TextTrack[3, 7]. Esta aplicación acepta mensajes sencillos en lenguaje natural sobre tipos y localizaciones de vehículos, correla estos mensajes y asocia grupos de mensajes en la pista más probable, basada en la sucesión de posiciones. Dado que el propósito es modelar la ambigüedad propia del lenguaje natural, las variables lingüísticas parecen una elección natural. El modelado del motor de inferencias usando matrices de transición va a permitir estudiar el comportamiento en bloque del sistema así como su estabilidad y su comportamiento en régimen estacionario.

  Otro problema de seguimiento en el que encontramos utilidad a este nuevo esquema es un detector de maniobra para un filtro de Kalman en un entorno de visión estereoscópica [5, 112]. A pesar de que no existe información lingüística, las propiedades que se obtienen de los sistemas borrosos realimentados utilizando matrices de transición se pueden aplicar para la construcción de una máquina de estados que controle el posible reinicio del filtro en caso de un desacople continuado. La vaguedad se encuentra aquí en el posicionamiento de la frontera entre acople y desacople.

- **Análisis automático del estado de maduración ósea en la infancia:** La valoración del grado de maduración ósea es un proceso usado con mucha frecuencia dentro del campo de la radiología pediátrica. Basado en un examen radiológico del desarrollo de los huesos de la mano y la muñeca izquierdas (no dominantes, en general), se puede así obtener la edad ósea del paciente y compararla con la edad cronológica. Una discrepancia entre estos dos valores indica anormalidades en el desarrollo óseo.

  Una técnica muy usada para el cálculo de la edad ósea es el método Tanner-Whitehouse (TW2)[123], método basado en un análisis detallado de cada hueso, el cual es clasificado en un estadio de desarrollo. Cada estadio lleva asociado una puntuación, y el conjunto de las mismas, debidamente ponderado, proporciona la edad ósea total del paciente. En su conjunto el método proporciona una gran precisión, aunque tiene una componente algo subjetiva debida a la naturaleza visual del proceso; adolece, no obstante, de ser complicado y laborioso, por lo que sólo se usa en el 20% de los casos. El tiempo empleado en la inspección manual de las radiografías es excesivo y, además, requiere de un facultativo especialista, habida cuenta de la especifidad del proceso.

  Surge la necesidad de un método de diagnóstico apoyado en ordenador. Este método bien podría ser una implementación práctica del método TW2 o un nuevo planteamiento basado en reconocimiento de patrones a partir de las características de cada radiografía. En esta dirección se han realizado varios esfuerzos en los últimos años [106], lejos todos ellos de una solución óptima al problema.

  La descripción del método de forma lingüística le hace susceptible de ser implementado usando técnicas de CWW. Por otro lado, usar directamente las conclusiones del método implica el uso implícito de las seis mil muestras utilizadas por los autores para llegar a esas conclusiones. En entornos clínicos reales es prácticamente imposible conseguir tal cantidad de muestras etiquetadas y que dichas muestras

cubran uniformemente todos los posibles casos a estudiar. La construcción de un clasificador directamente sobre las conclusiones del TW2 ofrece la posibilidad real de llegar, eventualmente, a una solución aceptable con los datos radiográficos de que se puede aspirar a disponer.

1.2 OBJETIVOS

Como se ha indicado anteriormente, el objetivo del trabajo doctoral será la definición de un nuevo marco de trabajo eficiente para el procesado de información no numérica. Este objetivo se amplía con la aplicación práctica de dicho marco a problemas de interés. Dentro de cada uno de estas dos direcciones, se plantean los siguientes subobjetivos:

1. Definición de un nuevo marco matricial para el procesado eficiente de información basada en texto:
   (a) Definición de posibles modos de modelado de información no numérica: conjuntos borrosos con variable base o información granular definida por relaciones.
   (b) Definición de una técnica de inferencia borrosa que aproveche las características propias de la información no numérica. Inferencia matricial.
   (c) Estudio especial de los sistemas borrosos realimentados mediante el uso de matrices de transición.
   (d) Definición de reglas de inferencia en modo bloque.
   (e) Extensión de la inferencia basada en matrices de transición al cálculo de árboles de decisión borrosos [56, 134].
   (f) Información mixta: inclusión de información numérica en este marco.
2. Estudio de la aplicabilidad del nuevo marco de trabajo a problemas reales:
   (a) Problemas de seguimiento:
      - Seguimiento de información basada en texto. Aplicación del nuevo marco de trabajo para definir el algoritmo MHT borroso [3, 7].
      - Aplicación a la construcción de una máquina de estados para el control del reinicio de un filtro de Kalman en un entorno de visión estereoscópica en caso de desacople continuado [5].
   (b) Determinación del grado de maduración ósea en la infancia:
      - Definición de características numéricas y no numéricas del método TW2.
      - Modelado de las características para un procesado automático.
      - Construcción de un clasificador basado en árboles de decisión borrosos.
      - Validación de los resultados y refinado del clasificador.

1.2.1 Estructura de la memoria

La memoria de la tesis se estructura en 7 capítulos y 6 apéndices, el contenido de los cuales se detalla a continuación:

- El capítulo 1 corresponde a esta introducción.
- El capítulo 2 constituye una visión panorámica de la lógica borrosa así como de la teoría de los conjuntos borrosos. Se introducen la notación y los conceptos que serán utilizados en los desarrollos posteriores.
- El capítulo 3 presenta el marco general de las técnicas que se reúnen bajo el nombre de *Computación con palabras* o *Cálculo lingüístico*. En el capítulo se realiza una pequeña aproximación histórica, y se estudian por separado las distintas ramas que forman la *Teoría computacional de las percepciones*, haciendo especial hincapié en el concepto de gránulo borroso.

  Este capítulo, de marcado carácter abstracto y, por ello, un tanto alejado de aspectos aplicados, se ha incluido para conseguir que la memoria de la tesis sea autocontenida. Sin embargo, dado que la tesis se ha centrado en aspectos de eficiencia computacional, así como de búsqueda de aplicaciones de ingeniería de este cuerpo doctrinal, entendemos que no es imprescindible la lectura en profundidad de este capítulo como paso previo para abordar los capítulos posteriores, que es donde se encuentran las aportaciones (resumidas en la sección 7.2) de la tesis. De hecho, en el capitulo 4 se han incluido los punteros a aspectos concretos del capítulo que nos ocupa, que son aquellos que se han considerado centrales en la tesis.
- En el capítulo 4 se presentan las principales aportaciones de la tesis. En primer lugar se describe una nueva manera de representar los conceptos lingüísticos de forma granular, la cual hace uso de la información estrictamente necesaria de un conjunto borroso para llevar a cabo inferencias. En segundo lugar se describe un método de inferencia borrosa basado en matrices de transición. Se realiza un estudio especial de los sistemas borrosos realimentados.
- En el capítulo 5 se describen dos aplicaciones de seguimiento que hacen uso de los resultados expuestos en el capítulo 4; en primer lugar se describe una versión borrosa del algoritmo MHT [108], y en segundo lugar un detector borroso de maniobra para un filtro de Kalman en un entorno de visión estereoscópica.
- En el capítulo 6 se describe una nueva aplicación de los conceptos desarrollados en el capítulo 4: la automatización de un método de análisis del estado de maduración ósea en la infancia.
- En el capítulo 7, se presentan las conclusiones y aportaciones del presente trabajo. Se propondrán también algunas líneas de trabajo futuras.
- En el apéndice A se recogen algunos desarrollos auxiliares del capítulo 4.
- En el apéndice B se encuentran los desarrollos de la extensión de las matrices de transición a sistemas basados en t-normas y t-conormas genéricas.

- En el apéndice C se recoge la aplicación de las matrices de transición descritas en el capítulo 4 a la construcción de árboles de decisión borrosos que se emplean en el capítulo 6.
- El apéndice D recoge algunos complementos al capítulo 5, como un estado del arte de los sistemas borrosos realimentados, un estudio del poder discriminador de hipótesis del MHT borroso y los valores de las matrices de transición usadas en dicho capítulo.
- El apéndice E recoge los detalles de implementación del filtro borroso de Kalman utilizado en el capítulo 5.
- El apéndice F contiene datos complementarios a los mostrados en el capítulo 6. En dicho capítulo todos los ejemplos están referidos a un único hueso (el radio). En este apéndice aparecen los resultados para el resto de los huesos.

Se incluye, así mismo, un glosario con las abreviaturas empleadas con más frecuencia en esta memoria.

2

INTRODUCCIÓN A LA LÓGICA BORROSA

BIBLIOTECA VIRTUAL

En 1965, L. A. Zadeh introduce la teoría de conjuntos borrosos [158], como un mecanismo para representar la vaguedad e imprecisión de los conceptos empleados en el lenguaje natural. Estos conjuntos *borrosos*[1] fueron definidos como una extensión de los conjuntos clásicos capaz de modelar la imprecisión propia de los conceptos humanos. A mediados de los 70 llega la ampliación del concepto de conjunto al de lógica, apareciendo las lógicas borrosas y las aplicaciones a sistemas de control. Hoy en día son muchas las aplicaciones tanto industriales como domésticas que hacen uso de este paradigma.

2.1 CONJUNTOS BORROSOS

Como se ha apuntado, el origen de la lógica borrosa es la noción de conjunto borroso. Antes de abordar el estudio de la Teoría de Conjuntos Borrosos, se revisarán algunos de los conceptos básicos de la Teoría Clásica, con el objeto de alcanzar una mayor comprensión de ambas.

2.1.1 Revisión de la Teoría Clásica de Conjuntos

Definiciones, terminología y notación

El punto de partida de la teoría de conjuntos son las nociones de *elemento* y de *conjunto*. Un conjunto se define genéricamente como una colección de elementos. Típicamente los elementos que forman parte de un conjunto tienen algún tipo de propiedad en común que les haga susceptibles de pertenecer al conjunto, pero tal requisito es meramente anecdótico. El conjunto se suele representar con una letra mayúscula, tipo A, B, C, etc..., y los elementos del mismo se representan con una letra minúscula (a,b,c, etc).

Sobre los conjuntos se define una relación de pertenencia, la cual se denota con el símbolo $\in$. Así pues, si el elemento a pertenece al conjunto A, este hecho se formaliza

[1] No existe un consenso claro respecto a la mejor traducción del término *fuzzy*. Algunos autores optan por *difuso*, mientras que otros lo hacen por *borroso*.

mediante la expresión

$$a \in A$$

En el caso en que b no pertenezca a A se escribe

$$b \notin A$$

Respecto a la forma de descripción del conjunto, ésta se puede realizar de manera enumerativa, $A = \{a_1, a_2, \ldots, a_n\}$, o bien, mediante la ley de formación a la que se ha hecho referencia $A =$'*los diez primeros números naturales*'. Tal definición, como puede imaginarse, es equivalente a escribir de forma enumerativa $A = \{1, 2, 3, \ldots, 10\}$.

Se define el *cardinal* de un conjunto como el número de elementos que forman parte de dicho conjunto. Si dicho cardinal es un número finito, el conjunto se denominará finito. Caso contrario será infinito. Dentro de estos últimos, deben distinguirse los de cardinal numerable, que serán aquellos cuyos elementos se pueden poner en relación 1:1 con los números enteros (por ejemplo, el conjunto de los números pares); por otra parte nos encontraremos conjuntos de cardinal no numerable, como, por ejemplo, el conjunto de los números reales comprendidos entre dos números a y b.

La relación de inclusión se deriva de la relación de pertenencia; un conjunto B se dice que está incluido dentro de un conjunto A cuando todos los elementos de B están en A. Si tal es el caso, podemos expresar de forma abreviada que $B \subset A$, o bien que $A \supset B$. Si se verifica que $B \subset A$ y que $A \subset B$ de forma simultánea, entonces es que los dos conjuntos son iguales.

Dos conjuntos de dice que son *disjuntos* si no tienen ningún elemento en común. A estos conjuntos se les denomina también mutuamente excluyentes.

Dado un problema, el conjunto universal, denotado por S[2], será el conjunto formado por todos los elementos del problema. De forma complementaria, el conjunto vacío, denotado por $\emptyset$, será un conjunto sin ningún elemento. Como es natural, los conjuntos S y $\emptyset$ son mutuamente excluyentes.

Sea S un universo del cual cualquier conjunto A es subconjunto, esto es:

$$A \subseteq S, \ \forall A$$

En teoría clásica de conjuntos cualquier elemento x perteneciente a S pertenece o no pertenece al subconjunto A de manera clara e inequívoca, sin que exista ninguna otra posibilidad al margen de estas dos.

La pertenencia o no de un elemento arbitrario x a un subconjunto A viene dada en la mayoría de los casos por la verificación o no de un predicado que caracteriza a A y da lugar a una bipartición del universo de discurso S.

[2]En lógica borrosa, el término universo suele ir acompañado del término "de discurso". En lo que sigue emplearemos tal denominación.

Por ejemplo, sea S el universo de discurso formado por todos los ríos del mundo. Definiremos el conjunto A como aquél que está formado por todos aquellos elementos de S que verifiquen el predicado "x fluye por Europa". Por citar algunos ejemplos ilustrativos:

$$\begin{array}{ll} \text{“Rhin”} & \in A \\ \text{“Nilo”} & \notin A \\ \text{“Ebro”} & \in A \end{array}$$

Debe hacerse notar que ha sido posible proporcionar una definición clásica del conjunto A porque su correspondiente predicado permite la bipartición del universo S.

Función característica

BIBLIOTECA VIRTUAL

El concepto de pertenencia o no de un elemento a un conjunto A puede expresarse numéricamente mediante una *función característica*[3]. Esta función asigna a cada elemento x del universo de discurso un dígito binario (1 ó 0) según x pertenezca o no al conjunto A [59].

$$\varphi_A : S \longrightarrow \{0,1\} \;\; \varphi_A(x) = \begin{cases} 1 & \text{si } x \in A \\ 0 & \text{si } x \notin A \end{cases} \tag{2.1}$$

Cualquier conjunto $A \subset S$ se puede definir por los pares que forman cada elemento x del universo y su función característica, expresándose de la siguiente forma:

$$A = \{(x, \varphi_A(x)) | x \in S\} \tag{2.2}$$

Por ejemplo, el conjunto $A = \{3,4,5,6,7,8,9,10\}$ se puede representar por su función característica

$$\varphi_A(x) = \begin{cases} 1 & \text{para } x \in \{3,4,5,6,7,8,9,10\} \\ 0 & \text{de otro modo} \end{cases}$$

Operaciones básicas entre conjuntos

Dados dos conjuntos cualesquiera A y B incluidos en S es posible definir un conjunto de operaciones básicas entre ellos [66, 130]:

Complemento: El complemento de A se denota por $\overline{A}$, y está formado por todos los elementos de S que no pertenecen a A (operador unario).

$$x \in \overline{A} \text{ si } x \notin A \tag{2.3}$$

Su función de característica será:

$$\varphi_{\overline{A}}(x) = 1 - \varphi_A(x) \tag{2.4}$$

[3] Algunos autores la denominan también *función de pertenencia*. En este texto se ha optado por usar *función característica* para los conjuntos clásicos y *función de pertenencia* para los borrosos.

Intersección: se denota por $A \cap B$ y se define como el conjunto formado por aquellos elementos de S que pertenecen a A y a B simultáneamente:

$$x \in A \cap B \text{ si } x \in A \text{ y } x \in B \tag{2.5}$$

La función característica correspondiente es:

$$\varphi_{A \cap B}(x) = \min(\varphi_A(x), \varphi_B(x)) \tag{2.6}$$

Unión: Es el conjunto formado por aquellos elementos que pertenecen a A, o pertenecen a B, o bien a ambos simultáneamente. Se denota por $A \cup B$ y su función característica es:

$$\varphi_{A \cup B}(x) = \max(\varphi_A(x), \varphi_B(x)) \tag{2.7}$$

Propiedades de los conjuntos clásicos

Las operaciones entre conjuntos clásicos presentan ciertas leyes y propiedades:

1. Propiedad conmutativa
$$\begin{aligned} A \cup B &= B \cup A \\ A \cap B &= B \cap A \end{aligned} \tag{2.8}$$
2. Propiedad asociativa
$$\begin{aligned} (A \cup B) \cup C &= A \cup (B \cup C) \\ (A \cap B) \cap C &= A \cap (B \cap C) \end{aligned} \tag{2.9}$$
3. Leyes de idempotencia
$$\begin{aligned} A \cup A &= A \\ A \cap A &= A \end{aligned} \tag{2.10}$$
4. Leyes de absorción
$$\begin{aligned} (A \cup B) \cap A &= A \\ (A \cap B) \cup A &= A \end{aligned} \tag{2.11}$$
5. Propiedad distributiva
$$\begin{aligned} A \cup (B \cap C) &= (A \cup B) \cap (A \cup C) \\ A \cap (B \cup C) &= (A \cap B) \cup (A \cap C) \end{aligned} \tag{2.12}$$
6. Propiedades de absorción por S y $\emptyset$
$$\begin{aligned} A \cup S &= S \\ A \cap \emptyset &= \emptyset \end{aligned} \tag{2.13}$$

7. Propiedades de identidad

$$\begin{aligned} A \cup \emptyset &= A \\ A \cap S &= A \end{aligned} \tag{2.14}$$

8. Involución del complemento

$$\overline{\overline{A}} = A \tag{2.15}$$

9. Leyes de Morgan

$$\begin{aligned} \overline{A \cup B} &= \overline{A} \cap \overline{B} \\ \overline{A \cap B} &= \overline{A} \cup \overline{B} \end{aligned} \tag{2.16}$$

10. Leyes complementarias

$$\begin{aligned} A \cup \overline{A} &= S \\ A \cap \overline{A} &= \emptyset \end{aligned} \tag{2.17}$$

BIBLIOTECA VIRTUAL

Lógica Clásica: Operadores y procedimientos de inferencia

El razonamiento en Lógica Clásica se realiza según dos esquemas de inferencia conocidos como *modus ponens* (MP) y *modus tollens* (MT). El primero de ellos es el fundamento del procedimiento de encadenamiento hacia adelante, mientras que el segundo se puede identificar con el encadenamiento hacia atrás [130, 131].

	Regla:	Si X es A,	entonces	Y es B
MP:	Hecho:	X es A		
	Conclusión:			Y es B

	Regla:	si X es A,	entonces	Y es B
MT:	Hecho:			Y es B
	Conclusión:	X es A		

En "razonamiento clásico" se compara el hecho, fruto de la observación, con el condicional de la regla. En caso de verificarse este último, se infiere de manera inmediata la correspondiente conclusión. La verificación ha de ser exacta, pues de lo contrario la regla en cuestión no será utilizable y no podrá efectuarse razonamiento alguno.

2.1.2 Extensión a conjuntos borrosos

En la sección 2.1.1 se ha visto cómo la mayoría de las veces los conjuntos clásicos se definen mediante un predicado que da lugar a una clara bipartición del universo de discurso S. Sin embargo, el razonamiento humano utiliza frecuentemente predicados de los cuales no resulta esa bipartición; son los denominados *predicados vagos*.

Siguiendo con el universo de discurso anterior, el formado por todos los ríos del mundo, podemos definir en él el conjunto B como aquél formado por los ríos "largos".

Por supuesto, es imposible dar a B una definición clásica, ya que su correspondiente predicado no divide el universo S en dos partes claramente diferenciadas. No resulta nada fácil afirmar con rotundidad que un río es "largo" o no lo es. El problema podría resolverse en parte considerando que un río es "largo" cuando su longitud supera cierto umbral fijado de antemano. Decimos que el problema tan sólo se resuelve en parte, y de manera no muy convincente, por dos motivos: de una parte el umbral mencionado se establece de una manera arbitraria, y por otro lado podría darse el caso de que dos ríos de longitudes muy diferentes fuesen considerados ambos como "largos". Evidentemente, el concepto "largo" así definido nos daría una información muy pobre sobre la longitud del río en cuestión.

La manera más apropiada de dar solución a este problema es considerar que la pertenencia o no pertenencia de un elemento x al conjunto B no es absoluta sino gradual. En definitiva, definiremos B como un conjunto borroso [90, 109] (*fuzzy set* en la bibliografía anglosajona). Su función característica (ahora "de pertenencia") ya no adoptará valores en el conjunto discreto $\{0, 1\}$, sino en el intervalo cerrado $[0, 1]$.

Mediante notación matemática se define un conjunto borroso A como:

$$A = \{(x, \mu_A(x)) | x \in S\} \tag{2.18}$$

Función de pertenencia

La función característica es reemplazada por una *función de pertenencia* que se define

$$\mu_A : S \longrightarrow [0, 1] \tag{2.19}$$

de tal modo que $\mu_A(x) \in [0, 1]$ es el grado con el que un elemento $x \in S$ (siendo S el universo de discurso) pertenece al conjunto borroso A. Cuando $\mu_A(x) = 0$ el elemento no pertenece al conjunto, y cuando $\mu_A(x) = 1$ pertenece totalmente.

La forma de la función de pertenencia tiene una cierta componente *subjetiva*, frente a la forma rígida (objetiva) de las funciones características de la lógica clásica. En función de la aplicación de los conjuntos o de los conceptos representados por ellos, estas funciones pueden adquirir muy diversas formas, y muchas veces pueden ser elegidas con un amplio grado de libertad por parte del "diseñador", lo que en la práctica puede traducirse como la posibilidad de incluir cierto conocimiento experto.

A pesar de que las funciones podrían tener cualquier forma, en la literatura se tiende a trabajar con funciones de pertenencia estándares [114] (Fig. 2.1):

1. Funciones Gaussianas o con forma de S (Fig. 2.1(a)). Usan la fórmula

$$\mu(x) = \frac{1}{s\sqrt{2\pi}} \exp\left(\frac{(x-m)^2}{2d^2}\right)$$

Permiten modelar fácilmente *modificadores* (sec. 2.1.9) pero son complicadas para el cálculo.

2. Funciones triangulares o trapezoidales [45]. Se definen en función de los vértices de las funciones; $\Delta(a, b, c)$ para las triangulares (Fig. 2.1(b)) y $T(a, b, c, d)$ para las trapezoidales (Fig. 2.1(c)). Son sencillas de manejar en algoritmos numéricos.

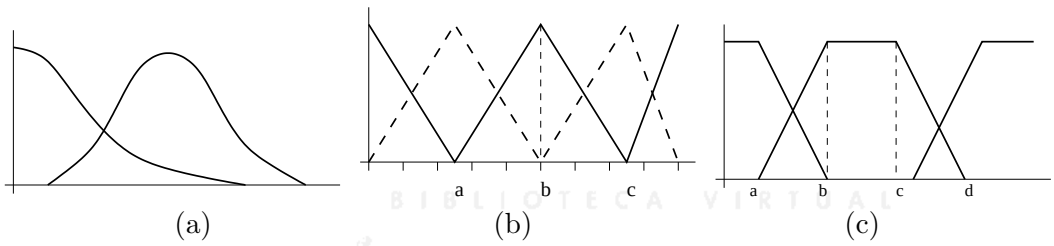


Figura 2.1 Funciones de pertenencias típicas: a) Gaussianas, b) Triangulares c) Trapezoidales.

Zeng y Singh [174] definen un modelo de función de pertenencia que agrupa a las principales clases, la función Pseudo-trapezoidal.

Definición 1 (Función Pseudo-trapezoidal (PTS))
La función PTS (Del inglés *pseudo trapezoid-shaped*) es una función continua dada por

$$A(x; a, b, c, d, h) = \begin{cases} I(x) & x \in [a, b) \\ h & x \in [b, c] \\ D(x) & x \in (c, d] \\ 0 & x \in U - [a, d] = \{x | x \in U, x \notin [a, d]\} \end{cases}$$

donde $a \leq b \leq c \leq d$, $a < d$, $I(x) \geq 0$ es una función monótona estrictamente creciente en $[a, b)$ y $D(x) \geq 0$ es una función monótona estrictamente decreciente en $(c, d]$ (Fig. 2.2). Cuando la función de pertenencia de un conjunto borroso A es una función PTS, se llama función de pertenencia PTS y se denota por $A(x) = A(x; a, b, c, d, h)$. Cuando el conjunto borroso es normal (es decir, $h = 1$), su función de pertenencia se denota simplemente por $A(x) = A(x; a, b, c, d)$.

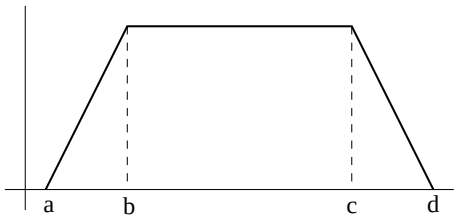


Figura 2.2 Función Pseudo-trapezoidal (PTS)

De acuerdo con la Definición 1, las funciones trapezoidales son un caso especial de las funciones PTS cuando $b < c$ y

$$I(x) = \frac{x - a}{b - a}, \quad D(x) = \frac{x - d}{c - d}$$

y las funciones triangulares es el caso especial cuando $b = c$ y

$$I(x) = \frac{x-a}{b-a} = \frac{x-a}{c-a}, \;\; D(x) = \frac{x-d}{c-d} = \frac{x-d}{b-d}$$

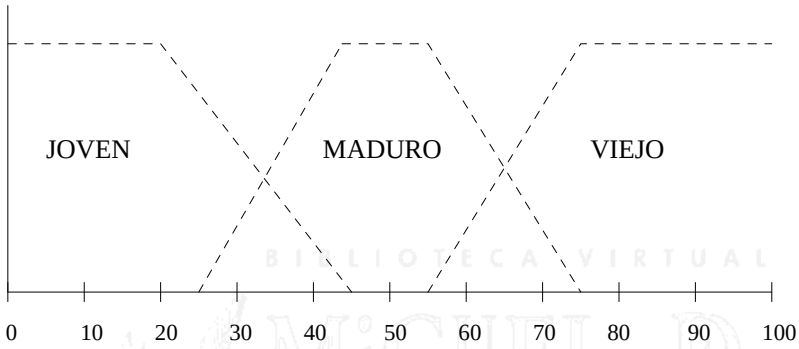


Figura 2.3 Ejemplo de conjuntos borrosos

Como ejemplo, en la figura 2.3 se muestran algunos conjuntos borrosos definidos en el universo de discurso *Edad*. El conjunto borroso "Joven" representa el grado de pertenencia respecto al parámetro juventud que tendrían los individuos de cada edad. Dicho de otra manera, el conjunto expresa la posibilidad de que un individuo sea caracterizado como "Joven". Un conjunto borroso podría ser considerado una distribución de posibilidad, que es diferente a una distribución de probabilidad [32].

Puede verse que los conjuntos borrosos de la figura 2.3 se superponen, entonces un individuo x, podría tener un grado de pertenencia en dos conjuntos: "Joven" y "Maduro", indicando que posee cualidades asociadas con ambos conjuntos.

Por razones prácticas, muchas veces se asume que el universo de discurso S es finito, esto es $S = \{x_1, \cdots, x_n\}$, y el par $\{(\mu_A(x), x)\}$ se denota por $\mu_A(x)/x$, y cada par $\mu_A(x)/x$ se denomina *individualidad borrosa* (*fuzzy singleton*) [59, 66]. El conjunto borroso A se puede reescribir como

$$\begin{aligned} A &= \{(\mu_A(x), x)\} = \{\mu_A(x)/x\} = \\ &= \mu_A(x_1)/x_1 + \cdots + \mu_A(x_n)/x_n = \textstyle\sum_{i=1}^{n} \mu_A(x_i)/x_i \end{aligned} \tag{2.20}$$

donde $+$ y $\sum$ deben entenderse en el sentido de la teoría de conjuntos. Por convenio, los pares $\mu_A(x)/x$ con $\mu_A(x) = 0$ se omiten.

2.1.3 Definiciones básicas sobre conjuntos borrosos

A continuación se presentan una serie de definiciones básicas de utilidad en el manejo de los conjuntos borrosos [59][4]:

[4] A partir de este punto se considerará $U \subset R$ el universo de discurso para los conjuntos borrosos.

Definición 2 (Conjunto vacío)
Se dice que un conjunto borroso A está vacío, y se escribe $A = \emptyset$, si y sólo si

$$\mu_A(x) = 0, \qquad \forall x \in U$$

Definición 3 (Igualdad)
Se dice que dos conjuntos borrosos A y B definidos sobre el mismo universo de discurso U son iguales, y se escribe $A = B$ si y sólo si

$$\mu_A(x) = \mu_B(x), \qquad \forall x \in U$$

Definición 4 (Contención)
Se dice que un conjunto borroso A definido en U está contenido en B en U (o alternativamente, es un subconjunto de B), y se escribe $A \subseteq B$, si y sólo si

$$\mu_A(x) \leq \mu_B(x), \qquad \forall x \in U$$

Definición 5 (Normalidad)
Se dice que un conjunto borroso A definido en U es normal si y sólo si

$$\max_{x \in U} \mu_A(x) = 1$$

Definición 6 (Soporte)
El soporte de un conjunto borroso A en U (escrito supp$A$ o bien S_A) es el conjunto no borroso

$$S_A = \{x \in U : \mu_A(x) > 0\}$$

y $\emptyset \subseteq S_A \subseteq U$.

Zeng y Singh [174] introducen unas nuevas definiciones que conducen al concepto de *orden* entre conjuntos borrosos.

Sea el espacio de entrada $U \subset R$ el universo de discurso; los conjuntos borrosos A_i definidos sobre S $(i = 1, 2, \cdots, N)$ son conjuntos borrosos y $A_i(x)$ $(i = 1, 2, \cdots, N)$ son las funciones de pertenencia borrosas correspondientes.

Definición 7 (Partición completa)
Se dice que los conjuntos borrosos $A_1, A_2, \cdots, A_N$ son una partición completa de U si $\forall x \in U$ existe al menos un A_i $(i = 1 \leq i \leq N)$ tal que $A_i(x) > 0$. Por simplicidad, se dice que los conjuntos $A_1, A_2, \cdots, A_n$ son completos si forman una partición completa.

Definición 8 (Consistencia)
Se dice que los conjuntos borrosos $A_1, A_2, \cdots, A_N$ son consistentes si se verifica que si $A_i(x_0) = 1$ para algún $x_0 \in U$, entonces $\forall j \neq i$, $A_j(x_0) = 0$.

Definición 9 (Subconjunto Normal de un Conjunto Borroso Normal)
Se define el subconjunto normal de un conjunto borroso normal A como

$$M(A) = \{x | x \in U \text{ y } A(x) = 1\}$$

que es un subconjunto de S_A (el soporte del conjunto borroso A). Si el conjunto borroso normal A tiene función de pertenencia PTS, entonces $M(A) = [b, c]$.

Definición 10 (Orden entre Conjuntos Borrosos Normales)
Para dos conjuntos borrosos normales A y $B \subset U$, se dice que $A > B$ si $M(A) > M(B)$, definiéndose esta desigualdad como

$$\max_x M(A) > \max_x M(B)$$

Proposición 1
[5] Si A_i son conjuntos borrosos normales en $U \subset R$ con funciones de pertenencia PTS $A_i(x) = A_i(x; a_i, b_i, c_i, d_i)$ $(i = 1 \leq i \leq N)$, existe una reordenación $\{i_1, i_2, \cdots, i_N\}$ de $\{1 \leq i \leq N\}$ tal que

$$A_{i_1} < A_{i_2} < \cdots < A_{i_N}$$

2.1.4 Operaciones entre conjuntos borrosos

Las tres operaciones básicas definidas sobre los conjuntos clásicos (complemento, intersección y unión) pueden ser generalizadas a los conjuntos borrosos de diversas formas. Dentro de la teoría de los conjuntos borrosos tiene especial relevancia la que hace uso de operaciones conocidas como *operaciones estándar* (Fig 2.4), definidas como:

$$\begin{array}{lrcl} \text{Intersección:} & \mu_{A\cap B}(x) & = & \min(\mu_A(x), \mu_B(x)) \\ \text{Unión:} & \mu_{A\cup B}(x) & = & \max(\mu_A(x), \mu_B(x)) \\ \text{Complemento:} & \mu_{\overline{A}}(x) & = & 1 - \mu_A(x) \end{array}$$

No obstante, al contrario que pasa con los conjuntos clásicos, ésta no es la única forma posible de definir estas operaciones; diferentes funciones pueden ser apropiadas para representarlas en diferentes contextos. Por lo tanto, no sólo las funciones de pertenencia de los conjuntos borrosos van a ser dependientes del contexto sino también las operaciones sobre dichos conjuntos [66].

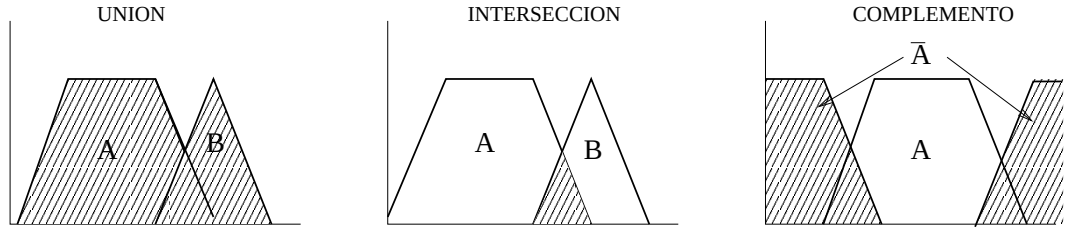


Figura 2.4 Operaciones básicas entre conjuntos borrosos. Definición estándar

Complemento borroso

Dado un conjunto borroso $A \subset U$, se define su complemento como el conjunto borroso $\overline{A}$ cuya función de pertenencia viene dada por la expresión

$$\mu_{\overline{A}}(x) = C(\mu_A(x)),\ \forall x \in U \tag{2.21}$$

donde $C(x)$ es una función que debe cumplir las siguientes propiedades:

[5] La demostración de esta proposición puede encontrarse en [174]

1. Condiciones de contorno: $C(0) = 1$, $C(1) = 0$.
2. Monotonía: para todo $a, b \in [0, 1]$, si $a \leq b$, entonces $C(a) \geq C(b)$

La función $C(x)$ es conocida por algunos autores como *c-norma* [93].

En la mayoría de los casos, es deseable considerar algunos requerimientos adicionales para estas funciones:

3. $C(x)$ es una función continua.
4. $C(x)$ es involutiva, lo que significa que $C(C(a)) = a$, $\forall a \in [0, 1]$

Existen muchas funciones que cumplen las propiedades antes descritas, y que por lo tanto pueden ser usadas para representar el complemento borroso. Algunas de ellas son:

$C(x) = 1 - x$	Negación
$C(x) = \frac{1-x}{1-\lambda x}$ $\lambda \in (0, \infty)$	Sugeno
$C(x) = (1 - x^w)^{1/w}$ $w \in (0, \infty)$	Yager

Intersección borrosa: t-norma

Dados dos conjuntos borrosos A y B, definidos sobre un mismo universo de discurso U, se define su intersección como un conjunto borroso $A \cap B$ cuya función de pertenencia viene dada por la expresión

$$\mu_{A \cap B}(x) = T[\mu_A(x), \mu_B(x)], \ \forall x \in U \tag{2.22}$$

donde la función $T(x, y)$ es una *norma triangular* o *t-norma* [66, 130]. Una t-norma es una aplicación $T : [0, 1] \times [0, 1] \to [0, 1]$ que verifica las siguientes propiedades:

1. Conmutativa: $T(x, y) = T(y, x), \ \forall x, y \in [0, 1]$.
2. Asociativa: $T(T(x, y), z) = T(x, T(y, z)), \ \forall x, y, z \in [0, 1]$.
3. Monotonía: si $(x \leq y)$ y $(w \leq z)$ entonces $T(x, w) \leq T(y, z)$, $\forall x, y, w, z \in [0, 1]$.
4. Elemento absorbente: $T(x, 0) = 0$, $\forall x \in [0, 1]$.
5. Elemento neutro: $T(x, 1) = x$, $\forall x \in [0, 1]$.

Existen muchas funciones que cumplen estas propiedades y que por lo tanto pueden ser utilizadas para representar la intersección entre conjuntos borrosos. Algunas de ellas son las siguientes:

$T(x,y) = \min(x,y)$	Mínimo
$T(x,y) = \max(0, x+y-1)$	Diferencia acotada
$T(x,y) = x \cdot y$	Producto algebraico
$T(x,y) = \begin{cases} \min(x,y) & \text{si } \max(x,y) = 1 \\ 0 & \text{en cualquier otro caso} \end{cases}$	Producto drástico

En ocasiones es necesario restringir las posibles t-normas considerando tres requerimientos adicionales [66]:

1. Continuidad: $T()$ es una función continua.
2. Subidempotencia: $T(x,x) < x$.
3. Monotonía estricta: $a_1 < a_2$ y $b_1 < b_2$ implica $T(a_1,b_1) < T(a_2,b_2)$.

El axioma de continuidad previene de una situación en la que un pequeño cambio en el grado de pertenencia de los conjuntos A ó B produzca un cambio grande (discontinuo) en el grado de pertenencia de $A \cap B$. La subidempotencia se tiene en cuenta cuando los grados de pertenencia de A y B para alguna x tienen el mismo valor. Este axioma expresa el requerimiento de que el grado de pertenencia de $A \cap B$ en este caso no exceda este valor. El tercer requerimiento es una condición más fuerte de monotonía.

Es habitual encontrar en la bibliografía la intersección de dos conjuntos borrosos expresada como

$$\mu_{A \cap B}(x) = \mu_A(x) \wedge \mu_B(x) \tag{2.23}$$

donde $\wedge$ representa al operador mínimo (intersección borrosa estándar). En el capítulo 4 y en el apéndice B se empleará el símbolo $\bigcap$ como indicativo de la t-norma.

Unión borrosa: t-conorma

Dados dos conjuntos borrosos A y B definidos sobre el mismo universo de discurso U, se define su unión como un conjunto borroso $A \cup B$ cuya función de pertenencia viene dada por la expresión:

$$\mu_{A \cup B}(x) = S(\mu_A, \mu_B), \ \forall x \in U \tag{2.24}$$

donde la función $S(x,y)$ es una *conorma triangular*, también llamada *t-conorma* o *s-norma*. Es una aplicación $S : [0,1] \times [0,1] \longrightarrow [0,1]$ que satisface los siguientes requisitos:

1. Conmutatividad: $S(x,y) = S(y,x)$, $\forall x, y \in [0,1]$
2. Asociatividad: $S(x, S(y,z)) = S(S(x,y), z)$, $\forall x, y, z \in [0,1]$
3. Monotonía: si $(x \leq y)$ y $(w \leq z)$ entonces $S(x,w) \leq S(y,z)$, $\forall x, y, w, z \in [0,1]$.
4. Elemento absorbente: $S(x,1) = 1$, $\forall x \in [0,1]$.

5. Elemento neutro: $S(x,0) = x$, $\forall x \in [0,1]$.

Al igual que en los casos anteriores, existe un gran número de funciones que cumplen estas propiedades y que pueden ser utilizadas para representar la unión. Algunos ejemplos son:

$$
\begin{array}{ll}
S(x,y) = \max(x,y) & \text{Máximo} \\
S(x,y) = \min(1, x+y) & \text{Suma acotada} \\
S(x,y) = x + y - x \cdot y & \text{Suma algebraica} \\
S(x,y) = \begin{cases} \max(x,y) & \text{si } \min(x,y) = 0 \\ 1 & \text{en cualquier otro caso} \end{cases} & \text{Suma drástica}
\end{array}
$$

En ocasiones es necesario restringir las posibles t-conormas considerando tres requerimientos adicionales [66], que tengan en cuenta casos especiales, tal y como se hizo para la t-norma:

1. Continuidad: $S()$ es una función continua.
2. Superidempotencia: $S(x,x) > x$
3. Monotonía estricta: $a_1 < a_2$ y $b_1 < b_2$ implica $S(a_1,b_1) < S(a_2,b_2)$

Es corriente encontrar en la bibliografía la unión de dos conjuntos borrosos expresada como

$$\mu_{A \cup B}(x) = \mu_A(x) \vee \mu_B(x) \tag{2.25}$$

donde $\vee$ representa al operador máximo (unión borrosa estándar). En el capítulo 4 y en el apéndice B se empleará el símbolo $\bigcup$ como indicativo de la t-conorma.

Relación entre operaciones borrosas

De las propiedades generales de las t-normas y t-conormas es fácil deducir que estas funciones están acotadas por las funciones mínimo y máximo:

$$T(x,y) \leq \min(x,y) \ \forall x, y \in [0,1] \tag{2.26}$$

$$S(x,y) \geq \max(x,y) \ \forall x, y \in [0,1] \tag{2.27}$$

Una determinada elección de los operadores unión, intersección y complemento pueden verificar las leyes de Morgan generalizadas:

$$C(T(x,y)) = S(C(x), C(y)) \tag{2.28}$$

$$C(S(x,y)) = T(C(x), C(y)) \tag{2.29}$$

En ese caso se dice que la t-norma y la t-conorma son duales respecto al complemento borroso. En general, dada una función de complemento, se puede asociar una t-norma a cada s-norma (y viceversa). Por ejemplo, utilizando la negación como complemento, las parejas mínimo-máximo y producto-suma verifican las leyes de Morgan generalizadas.

No todas las t-normas y t-conormas van a ser duales ni van a ser distibutivas. De acuerdo con [66]:

1. La terna $\{\min, \max, C\}$ es dual respecto al complemento borroso C.
2. Los operadores mínimo y máximo también verifican la propiedad distributiva:

$$\min(x, \max(y, z)) = \max(\min(x, y), \min(x, z)) \tag{2.30}$$

$$\max(x, \min(y, z)) = \min(\max(x, y), \max(x, z)) \tag{2.31}$$

3. Dada una t-norma T y un complemento borroso involutivo C, la operación binaria S en $[0, 1]$ definida por

$$S(a, b) = C(T(C(a), C(b))$$

$\forall a, b \in [0, 1]$, es una t-conorma tal que $\{T, S, C\}$ es una terna dual.

4. Dada una t-conorma S y un complemento involutivo C, la operación binaria T en $[0, 1]$ definida por

$$T(a, b) = C(S(C(a), C(b))$$

$\forall a, b \in [0, 1]$, es una t-norma tal que $\{T, S, C\}$ es una terna dual.

5. Sea $\{T, S, C\}$ una terna dual que satisface las leyes complementarias, esto es

$$S(x, C(x)) = U$$

$$T(x, C(x)) = \emptyset$$

Entonces $\{T, S, C\}$ no satisface las leyes distributivas.

2.1.5 Propiedades de los conjuntos borrosos

Las leyes y propiedades que, según se ha visto, cumplen los conjuntos clásicos no siempre se cumplen en el caso de los conjuntos borrosos. A continuación se analizará qué leyes verifican los conjuntos borrosos y cuáles no:

1. Propiedad conmutativa: siempre se verifica, debido a que las t-normas y las t-conormas son conmutativas por definición.
2. Propiedad asociativa: también se verifica puesto que las t-normas y las t-conormas también son asociativas.

3. Leyes de idempotencia: se cumplen si se elige el mínimo y el máximo como operadores para la intersección y la unión respectivamente. Pero si se escoge por ejemplo el producto algebraico y la suma algebraica, no se cumplen.

4. Leyes de absorción: también se cumplen si se elige el par mínimo-máximo. Con otras normas no ocurre necesariamente lo mismo.

5. Propiedad distributiva: también se cumple para el mínimo y el máximo, pero no necesariamente para otras normas.

6. Propiedades de absorción e identidad: siempre se cumplen debido a la última propiedad de las t-normas y t-conormas.

7. Involución del complemento: es cierta si se define $\mu_A(x) = 1-\mu_A(x)$, ya que entonces:

$$\mu_{\overline{A}} = 1 - \mu_A(x) = 1 - (1 - \mu_{\overline{A}}(x)) = \mu_{\overline{A}}(x) \tag{2.32}$$

8. Leyes de De Morgan: se garantiza su cumplimiento si la t-norma y s-norma elegidas se derivan la una de la otra. Es decir: $T(x, y) = 1 - S(1 - x, 1 - y)$.

9. Leyes complementarias: en general no se cumplen. Es quizás la consecuencia más clara de introducir el concepto de borrosidad en los conjuntos.

Se puede comprobar fácilmente que en el caso de que los conjuntos sean clásicos (función de pertenencia restringida a 0 ó 1) las diferencias entre las diversas normas desaparecen, convirtiéndose en los operadores intersección y unión clásicos.

Algunos autores críticos de la Teoría de Conjuntos Borrosos achacan a ésta el hecho de que exista arbitrariedad a la hora de elegir los operadores de unión e intersección. No obstante esto, que parece un inconveniente, puede ser por otro lado una ventaja, ya que nos permite una gran flexibilidad a la hora de abordar distintos problemas que involucren conceptos vagos. Si se precisa mantener ciertas propiedades de los conjuntos clásicos, deben elegirse una t-norma y una t-conorma que lo permitan. Esta elección dará lugar a un tipo u otro de lógica borrosa.

2.1.6 Principio de extensión

Este principio proporciona un mecanismo para calcular los conjuntos borrosos obtenidos por medio de una transformación concreta (no borrosa) de cierto número (N) de conjuntos borrosos. Específicamente, si X_1,X_2,...,X_N son conjuntos borrosos con funciones de pertenencia $\mu_1(x_1)$, $\mu_2(x_2)$, ..., $\mu_N(x_N)$, el nuevo conjunto borroso $Y=f(X_1, X_2, \ldots, X_N)$ tendrá como función de pertenencia:

$$\mu(y) = \max_{\mathbf{x}=f^{(-1)}(y)} \left[\min_{i=1}^{N} \mu_i(x_i) \right] \tag{2.33}$$

2.1.7 α-cortes

Existe una manera directa de pasar de conjuntos borrosos a conjuntos clásicos mediante los llamados α-cortes [66]. Dado un número $\alpha \in [0,1]$ y un conjunto borroso A, definimos el α-corte de A como el conjunto A_α, cuya función característica se define:

$$\varphi_{A_\alpha}(x) = \begin{cases} 1 & \text{si } \mu_A(x) \geq \alpha \\ 0 & \text{en cualquier otro caso} \end{cases} \tag{2.34}$$

En definitiva, el α-corte se compone de aquellos elementos cuyo grado de pertenencia supera o iguala el umbral α.

Hablamos de α-cortes estrictos si:

$$\varphi_{A_\alpha}(x) = \begin{cases} 1 & \text{si } \mu_A(x) > \alpha \\ 0 & \text{en cualquier otro caso} \end{cases} \tag{2.35}$$

Cualquier conjunto borroso A se puede representar mediante la unión de sus α-cortes de la siguiente manera:

$$\mu_A(x) = \max_{\alpha \in [0,1]} [\alpha \cdot \varphi_{A_\alpha}(x)] \tag{2.36}$$

Los α-cortes son de especial utilidad en el estudio de propiedades tales como la reflexividad, simetría y transitividad en conjuntos borrosos.

2.1.8 Números borrosos

Un caso particular y de especial interés de los conjuntos borrosos son los llamados *números borrosos*. Surgen éstos como un intento de introducir vaguedad en los números reales.

Un número borroso es un conjunto borroso A definido en la recta real R y que cumple además las siguientes propiedades:

1. Es normal, o lo que es lo mismo, existe al menos un elemento x de R tal que $\mu_A(x) = 1$.
2. Es convexo, lo cual quiere decir que

$$\forall \delta \in [0,1] \;\; \forall x, y \in R$$
$$\mu_A(\delta x + (1-\delta)y) \geq \min(\mu_A(x), \mu_A(y))$$

   Geométricamente esta propiedad quiere decir que todos los α-cortes de A son intervalos cerrados en R. La función de pertenencia es creciente hasta llegar al punto en que $\mu_A(x) = 1$ y decreciente a partir de él.

3. La función de pertenencia es continua en intervalos.

4. El soporte de A es acotado.

La enorme importancia de los números borrosos estriba en que son muchos los ejemplos prácticos en los que el grado de pertenencia de un determinado elemento del universo $U \subset R$ se puede expresar como función de una característica mensurable del mismo.

Como ejemplo visual, se presenta unas posibles definiciones gráficas de los números borrosos que expresan los predicados "aproximadamente 4" y "aproximadamente 6'. De un modo arbitrario se han escogido funciones de tipo gaussiano y triangular, respectivamente, tal y como se muestran en la Fig. 2.5.

BIBLIOTECA VIRTUAL MIGUEL DE CERVANTES

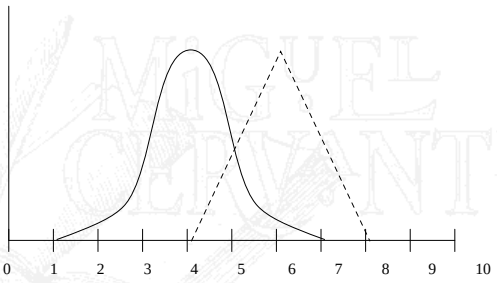


Figura 2.5 Números Borrosos "aproximadamente 4" y "aproximadamente 6"

2.1.9 Modificadores lingüísticos

Un modificador lingüístico, es una operación que modifica el significado de un término, o de manera más general de un conjunto borroso [32, 66, 90]. Por ejemplo, si un conjunto borroso denota *presión débil*, entonces *presión muy débil*, *presión más o menos débil*, *presión extremadamente débil* son ejemplos de modificadores aplicados a este conjunto. Los modificadores se pueden ver como operadores que actúan sobre la función de pertenencia de un conjunto borroso para modificarla. Algunos de estos operadores pueden ser:

1. Concentración:
$$\mu_{con(U)}(x) = [\mu_U(x)]^2 \tag{2.37}$$

2. Dilatación:
$$\mu_{dil(U)}(x) = [\mu_U(x)]^{1/2} \tag{2.38}$$

3. Modificadores artificiales:
$$\begin{array}{rcl} \mu_{plus(U)}(x) & = & [\mu_U(x)]^{1.25} \\ \mu_{minus(U)}(x) & = & [\mu_U(x)]^{0.75} \end{array} \tag{2.39}$$

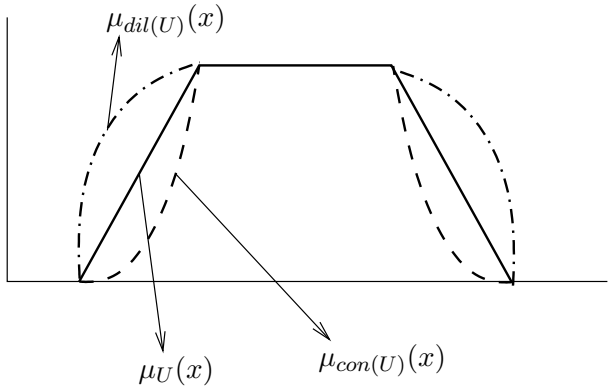


Figura 2.6 Ejemplo de modificadores aplicados sobre un conjunto borroso

En la Fig. 2.6 se muestra gráficamente el efecto de los dos primeros modificadores sobre un conjunto borroso con función de pertenencia trapezoidal. Además de cambiar la forma de las funciones de pertenencia, algunos modificadores también cambian la posición de los conjuntos sobre el eje de ordenadas; por ejemplo el modificador *muy* aplicado sobre el conjunto borroso GRANDE desplaza a esta hacia la derecha (Fig. 2.7).

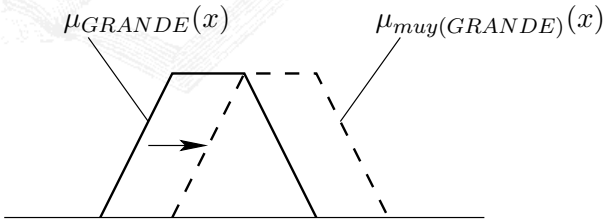


Figura 2.7 Modificador *muy* aplicado sobre el conjunto GRANDE

2.2 LÓGICA BORROSA

La Teoría de Conjuntos Borrosos puede utilizarse para representar expresiones lingüísticas de las que se hace uso para describir conjuntos o algoritmos. Los conjuntos borrosos son capaces de captar por sí mismos la vaguedad lingüística de palabras y frases comúnmente aceptadas, como "flor roja" o "ligero cambio". La habilidad humana de comunicarse mediante definiciones vagas o inciertas es un atributo importante de la inteligencia.

2.2.1 Variables lingüísticas

Una Variable Lingüística es una variable cuyos valores son palabras o sentencias que se enmarcan en un lenguaje predeterminado. Cada una de estas palabras o términos se conoce como *etiqueta lingüística* y se representa por medio de un conjunto borroso

definido sobre el universo de discurso de la variable. Por ejemplo, la temperatura del cuerpo humano puede catalogarse como '*baja*', '*normal*', '*alta*' o '*muy alta*'. Cada uno de estos términos es una etiqueta lingüística que puede definirse como un conjunto borroso. Los conjuntos que forman la variable se muestran en la Fig. 2.8.

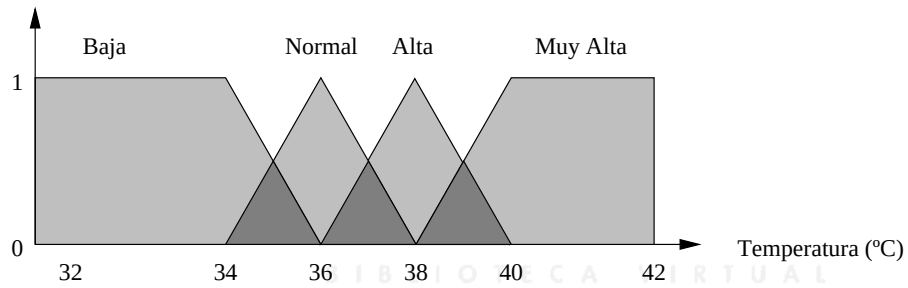


Figura 2.8 Definición de la variable lingüística *Temperatura*

El concepto de variable lingüística se ampliará en el capítulo 3.

2.2.2 Relaciones borrosas: operadores lógicos

Como se ha expuesto en las secciones anteriores, las operaciones unión, intersección y complemento operan todas ellas en un único universo de discurso. Ahora bien, el producto cartesiano permite productos de más de un universo de discurso.

Producto cartesiano

Sean U y V dos universos de discurso cualesquiera. Se define una relación borrosa R entre U y V como un conjunto borroso cuyo universo es el producto cartesiano $U \times V$. Es decir:

$$\begin{aligned} &R = \{((x,y), \mu_R(x,y)) \ / \ (x,y) \in U \times V\} \\ &\mu_R : U \times V \longrightarrow [0,1] \end{aligned} \tag{2.40}$$

Si $A_1 \subset U$ y $A_2 \subset V$, y si se define el producto cartesiano de A_1 y A_2 como:

$$\mu_{A_1 \times A_2}(x,y) = \min(\mu_{A_1}(x), \mu_{A_2}(y)) \tag{2.41}$$

entonces, esta función vendría representada como se muestra en la figura 2.9.

Por ejemplo, si $U = V = R$, se puede definir la siguiente relación borrosa que expresa en qué medida los números reales x e y son similares:

$$R = \{((x,y), \mu_R(x,y)) | (x,y) \in R^2\} \tag{2.42}$$

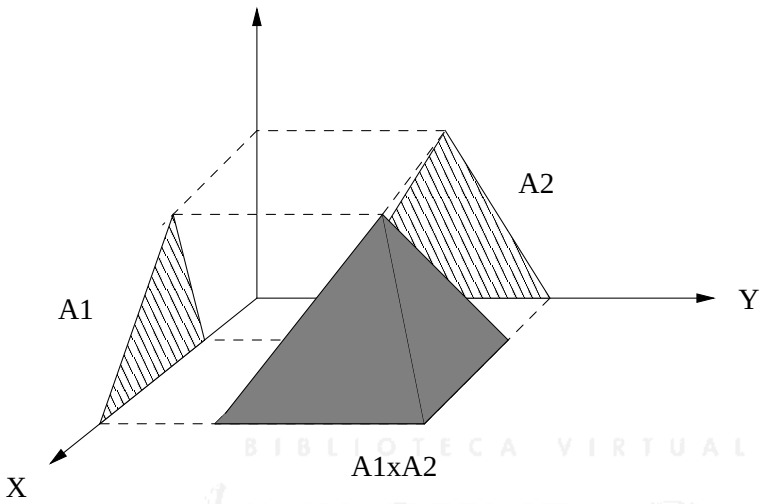


Figura 2.9 Producto cartesiano de dos universos de discurso: X e Y.

$$\mu_R(x,y) = \begin{cases} (1+(x-y)^4)^{-1} & \text{si } |x-y| \leq 5 \\ 0 & \text{en cualquier otro caso} \end{cases} \tag{2.43}$$

No obstante, la utilidad más destacada de las relaciones borrosas es la de poder actuar como operadores conectivos.

Operador conectivo "AND" ("Y")

Si tenemos dos conjuntos borrosos $A \subset U$ y $B \subset V$, y un par $(x,y) \in U \times V$, el conectivo "AND" que nos indica en qué medida $x \in A$ e $y \in B$ se puede implementar mediante la siguiente relación borrosa:

$$\mu_{AND}(x,y) = \min(\mu_A(x), \mu_B(y)) \tag{2.44}$$

Se puede observar que esta noción es muy parecida a la de intersección de conjuntos borrosos. No es exactamente igual, ya que la operación de intersección se define para conjuntos en el mismo universo de discurso. De todos modos, y debido precisamente a esa similitud, es frecuente definir el conectivo "AND" mediante cualquier t-norma y no sólo mediante el empleo del mínimo.

Operador conectivo "OR" ("O")

El conectivo "OR", que da idea de en qué medida $x \in A$ ó $y \in B$, se define habitualmente como la relación

$$\mu_{OR}(x,y) = \max(\mu_A(x), \mu_B(y)) \tag{2.45}$$

o bien mediante cualquier otra t-conorma.

Operador implicación "THEN" ("ENTONCES")

De igual forma, se buscan realizaciones para el operador de implicación "THEN" como relaciones borrosas entre antecedentes y consecuentes. En Lógica Borrosa hay muchas maneras con las que puede definirse una implicación; basándose en dichas definiciones, pueden generarse muchas funciones de implicación diferentes en base a t-normas y t-conormas. A continuación expresamos las principales implicaciones borrosas, de las que la implicación de Mamdani (mínimo) y de Larsen (producto) son las de más fácil implementación.

$\mu_M(x,y) = \min(\mu_A(x), \mu_B(y))$	Mamdani
$\mu_P(x,y) = \mu_A(x) \cdot \mu_B(y)$	Larsen
$\mu_R(x,y) = 1 - \mu_A(x) + \mu_A(x) \cdot \mu_B(y)$	Reichenbach
$\mu_L(x,y) = \min(1 - \mu_A(x) + \mu_B(y), 1)$	Lucasiewick
$\mu_W(x,y) = \max(1 - \mu_A(x), \min(\mu_A(x), \mu_B(y))$	Willmott
$\mu_{KD}(x,y) = \max(1 - \mu_A(x), \mu_B(y))$	Kleene-Dienes
$\mu_{RG}(x,y) = \begin{cases} 1 & \forall(x,y)\ \mu_A(x) \leq \mu_B(y) \\ 0 & \forall(x,y)\ \mu_A(x) > \mu_B(y) \end{cases}$	Reschner-Gaines
$\mu_G(x,y) = \begin{cases} 1 & \forall(x,y)\ \mu_A(x) \leq \mu_B(y) \\ \mu_B(y) & \forall(x,y)\ \mu_A(x) > \mu_B(y) \end{cases}$	Brouwer-Godel
$\mu_{RG}(x,y) = \begin{cases} \min(\mu_A(x)/\mu_B(y), 1) & \forall(x,y)\ \mu_B(y) \neq 0 \\ 1 & \forall(x,y)\ \mu_A(x) = 0 \end{cases}$	Goguen

Cuando μ_A y μ_B no toman nada más que los valores 0 y 1, las definiciones anteriores son compatibles con la implicación de la lógica clásica (correspondiendo 0 al valor de falsedad y 1 al de verdad), salvo en las implicaciones μ_M y μ_P.

2.2.3 Composición de relaciones borrosas: procedimientos de inferencia borrosa

Al contrario que en Lógica Clásica, en Lógica Borrosa el razonamiento no es preciso, sino que éste tiene lugar de una manera aproximada. Esto quiere decir que se puede inferir una conclusión aunque el hecho no verifique la regla plenamente ("Razonamiento Aproximado"). Dicha conclusión se parecerá tanto más a la conclusión formal de la regla original cuanto mayor sea el grado de cumplimiento de la regla por parte del hecho. El razonamiento aproximado se resume generalmente, por extensión del razonamiento clásico, en los esquemas de *modus ponens generalizado* y *modus tollens generalizado*, que veremos a continuación.

Proposiciones borrosas

En una proposición borrosa, a diferencia de lo que ocurre en las proposiciones clásicas, el grado de verdad puede estar en un valor intermedio entre uno y cero. Hay un tipo de proposición borrosa que será particularmente útil, llamada proposición borrosa *incondicional* y *no cualificada*. Se define como:

$$\text{p: Si } X \text{ es } A \text{ entonces } Y \text{ es } B$$

donde X e Y son variables con valores en los universos U y V respectivamente y A y B son conjuntos borrosos sobre U y V respectivamente. Esta expresión se puede escribir más formalmente por medio de la relación

$$< X, Y > \ \ es \ \ R \tag{2.46}$$

donde R es un conjunto borroso definido en $U \times V$, y la función de pertenencia de cada una se puede derivar de cada punto $(x, y) \in U \times V$ por medio de una *implicación borrosa*

$$R(x, y) = \mathcal{I}[A(x), B(y)] \tag{2.47}$$

Hay muchas definiciones de la implicación superior. Uno de ellos, es el conocido como la implicación de *Lukasiewicz*, (ver apartado 2.9) definida por:

$$\mathcal{I}[A(x), B(y)] = \min(1, 1 - A(x) + B(y)) \tag{2.48}$$

Modus Ponens Generalizado (MPG)

En este esquema, se expresa una regla, se da un hecho y a partir de ambos se obtiene una conclusión.

Regla:	Si X es A	entonces Y es B
Hecho:	X es A'	
Conclusión:	Y es B'	(MPG)

donde A, B, A' y B' son conjuntos borrosos. La conclusión B' es un conjunto borroso caracterizado por una generalización del Modus Ponens propuesto por Zadeh:

$$\mu_{B'}(y) = \max_{x \in U}[\min(\mu_{A'}(x), R(x, y))] \tag{2.49}$$

siendo $R(x, y)$ la implicación definida en la ecuación (2.47). De hecho, si el conjunto borroso A' se reduce al elemento x_0, entonces,

$$\mu_{A'} = \begin{cases} 1 & \text{para } x = x_0 \\ 0 & \text{para } x \neq x_0 \end{cases} \tag{2.50}$$

tomando la implicación $R(x, y)$ por la correspondiente de Mamdani, se obtiene:

$$\mu_{B'}(y) = \min(\mu_A(x_0), \mu_B(y)) \tag{2.51}$$

En suma, el MPG se reduce al esquema de *modus ponens* clásico si $A' = A$ y $B' = B$.

Modus Tollens Generalizado (MTG)

En el Modus Tollens Generalizado (MTG) se cumple que:

Regla:	Si X es A	entonces	Y es B
Hecho:			Y es B'
Conclusión:	X es A'		(MTG)

En este caso la regla composicional de inferencia tiene la forma

$$\mu_{A'}(x) = \max_{y \in V} \min(\mu_{B'}(y), R(x, y)) \tag{2.52}$$

Razonamiento Aproximado Multicondicional

Este razonamiento es una generalización del *modus ponens* a múltiples condiciones [66]:

Regla:	Si X es A_1,	entonces Y es B_1
Regla:	Si X es A_2,	entonces Y es B_2
...	...	...
Regla:	Si X es A_n,	entonces Y es B_n
Hecho:	X es A'	
Conclusión:	Y es B'	

Esta clase de problemas de inferencia se puede manejar también a partir de lo que llamamos el *método de interpolación*, que consiste en dos pasos: (ver figura 2.10)

1. Encontrar el grado de consistencia del antecedente $A_j, j \in \{1, \ldots, n\}$ con el hecho A'

   $$dc_j(A') = h(A' \cap A_j) \tag{2.53}$$

   siendo $h()$ una medida de la altura del conjunto, que puede calcularse usando una t-conorma

   $$dc_j(A') = \bigcup_{x \in U} (A'(x) \bigcap A_j(x)) \tag{2.54}$$

   Usando las opraciones estándar resulta

   $$dc_j(A') = \max_{x \in U} \min[A'(x), A_j(x)] \tag{2.55}$$

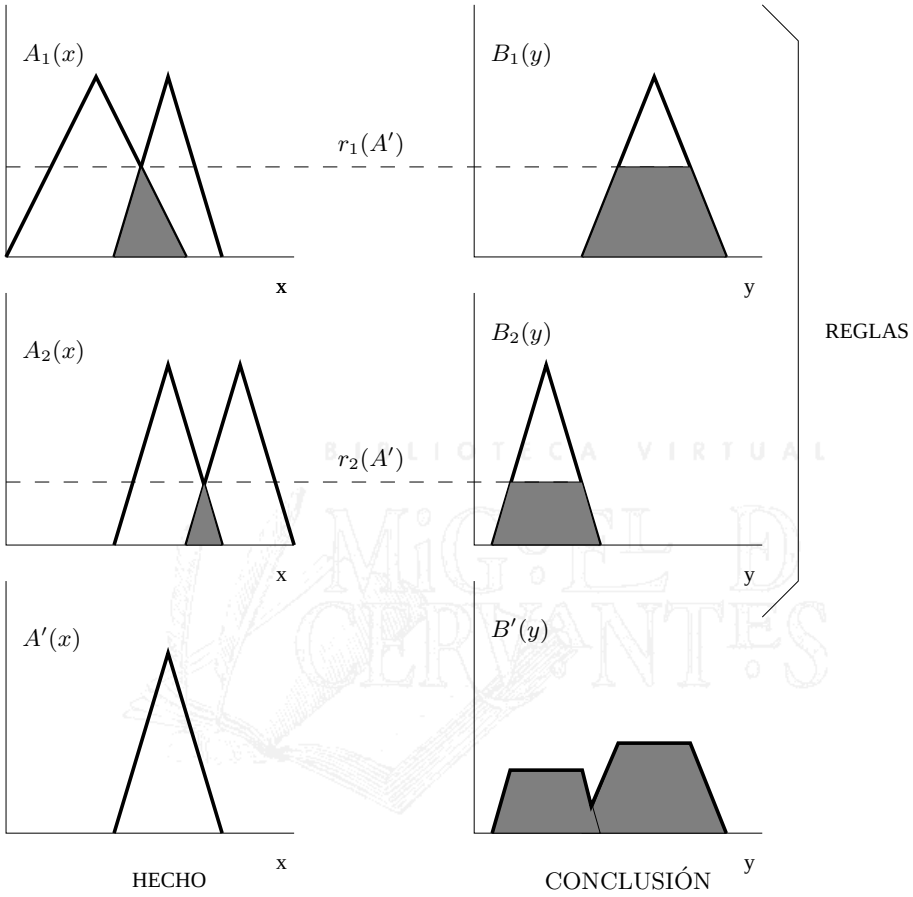


Figura 2.10 Representación Gráfica del *método de interpolación*

2. Calcular la conclusión B' truncando cada conjunto B_j por el valor $r_j(A')$ y tomar la unión de los conjuntos truncados (en caso de que las reglas sean disyuntivas)

$$B'(y) = \bigcup_j (dc_j(A') \bigcap B_j(y)) \ \forall y \in Y \tag{2.56}$$

que usando las operaciones estándar resulta ser

$$B'(y) = \max_j \min[dc_j(A'), B_j(y)] \ \forall y \in Y \tag{2.57}$$

Para el caso de dos entradas, la base de reglas tendrá dos antecedentes, $A_j(x)$ y $B_j(y)$, por lo que el *dc* debe ser calculado de otra forma; se asumirá que el *dc* es el mínimo (o de manera más general, otra t-norma) de los dos niveles de activación de cada antecedente por cada entrada. Específicamente

$$dc_j(X, Y) = dc_j(X) \bigcap dc_j(Y) \tag{2.58}$$

que usando el mínimo resulta

$$dc_j(X,Y) = \min(dc_j(X), dc_j(Y))$$

Esta expresión puede extenderse para el caso de múltiples antecedentes, que para una t-norma genérica resulta

$$(dc_i)_j = \bigcap_i (dc_j(Y_i)) \tag{2.59}$$

donde Y_i son las entradas al sistema.

2.2.4 Entradas al sistema y activación de reglas

BIBLIOTECA VIRTUAL MIGUEL DE CERVANTES

Cuando en un razonamiento aproximado, como los que se describen anteriormente, la entrada es un valor numérico x_0, la activación de cada conjunto puede hacerse de forma directa como

$$dc_j(x_0) = A_j(x_0)$$

Esto es lo que ocurre en muchas de las aplicaciones que usan razonamientos borrosos, como, por ejemplo, los controladores borrosos.

En cambio, cuando la entrada es un conjunto borroso A', es necesario definir un método para calcular el grado de consistencia de cada antecedente A_j dicha entrada. Dicho método se conoce también como *método de activación de reglas*.

Se pueden considerar varias estrategias para solucionar este problema, basadas en el uso de diversos operadores:

Definición 11 (Método Max-min)
([66, 90]) El grado de consistencia se define, como ya se ha visto en (2.55)

$$dc_j(A') = \max_{x \in X}\{\mu_{A_j \cap A'}\} = \max_{x \in X} \min\{A_j(x), A'(x)\} \tag{2.60}$$

Definición 12 (Método Sum-product)
Kosko [72] propone el siguiente método de activación de reglas:

$$dc_j(A') = \int_{R^n} A'(x) A_j(x) dx \tag{2.61}$$

siendo x una variable continua. Podemos derivar fácilmente su versión discreta [3]:

$$dc_j(A') = \frac{1}{K} \sum_i A_j(x_i) A'(x_i) \tag{2.62}$$

con K un factor de normalización. Este método, de forma opuesta al anterior, toma en cuenta el área encerrada bajo la intersección de ambos conjuntos, no sólo su valor máximo.

Definición 13 (Método Max-product)
[20] Una alternativa a los dos métodos anteriores es este método híbrido, expresado como

$$dc_j(A') = \frac{1}{K} \max \{A_j(x)A'(x)\} \tag{2.63}$$

Este método tiene la ventaja de realizar la intersección mediante un producto, pero no considera todo el área bajo la intersección; el operador *max* reducirá de forma considerable las colas laterales que puedan aparecer en el conjunto borroso de salida.

Definición 14 (Método Sum-min)
Una posibilidad más de calcular la activación de los conjuntos $A_j(x)$ y $A'(x)$ es la siguiente:

$$dc_j(A') = \frac{1}{K} \sum_i \min\{A_j(x_i), A'(x_i)\} \tag{2.64}$$

o para el caso de variable continua:

$$dc_j(A') = \int_{R^n} \min\{A_j(x), A'(x)\}dx \tag{2.65}$$

Pueden definirse otros métodos en función de las normas usadas. En [3] los autores realizan un estudio de la influencia de los métodos de interpolación en el comportamiento de los sistemas borrosos.

2.3 SISTEMAS DE LÓGICA BORROSA

Un sistema de sógica borrosa es la aplicación de la inferencia borrosa a la automatización de procesos. En este capítulo se recogen las principales ideas que subyacen en la aplicación de estas técnicas a sistemas reales.

2.3.1 Tipos de sistemas basados en reglas borrosas

En la literatura especializada se distinguen tres clases de sistemas basados en reglas borrosas (SBRB), de acuerdo con la forma de las reglas y del tipo de entradas y salidas [42]:

Sistemas puros

Estos sistemas tienen como entrada y como salida conjuntos borrosos. Al no realizar ninguna transformación sobre las entradas o sobre las salidas, tienen sólo dos componentes principales: una *base de conocimiento* y un *motor de inferencias* (Fig. 2.11).

Las reglas lingüísticas empleadas son de la forma

$$\text{SI } X_1 \text{ es } A_1 \text{ y } \cdots X_n \text{ es } A_n \text{ ENTONCES } Y \text{ es } B$$

donde X_i e Y son variables lingüísticas, y los A_i y B son etiquetas lingüísticas asociadas a conjuntos borrosos.

Sistemas borrosos tipo Mamdani

Este tipo de sistemas fue propuesto por Mamdani [84], quien fue capaz de traducir las teorías borrosas propuestas por Zadeh en el primer sistema borroso aplicado a un problema de control. Esta clase de sistemas, la más usada dentro de los sistemas borrosos, se conoce también con el nombre de *controladores borrosos*.

BIBLIOTECA VIRTUAL

Un sistema tipo Mamdani se corresponde con la noción más conocida de *sistema borroso*; está compuesto por una base de conocimiento, un motor de inferencias, y unos interfaces de borrosificación y desborrosificación (Fig. 2.11). Estos sistemas tienen una serie de características interesantes [8]:

- Pueden ser usados en aplicaciones del mundo real, ya que manejan con facilidad entradas y salidas reales.
- Proporcionan un marco natural para la inclusión de conocimiento experto en forma de las reglas lingüísticas.
- Tienen gran libertad a la hora de elegir los interfaces de borrosificación y desborrosificación.

Por el contrario presentan también una serie de limitaciones [42]:

- Falta de flexibilidad en el SBRB debido a la forma tan rígida en la que se particionan los espacios de entrada y salida.
- No existe una distinción clara entre el conocimiento experto y la definición de las variables lingüísticas incluidas en las reglas borrosas.
- Cuando las variables de entrada al sistema dependen unas de otras, es muy complicado obtener una partición borrosa adecuada de los espacios de entrada.
- El tamaño de la base de conocimiento depende directamente del número de variables y términos lingüísticos que existan en el sistema.

Las reglas que manejan esta clase de sistemas son de la forma

$$\text{SI } X_1 \text{ es } A_1 \text{ y } \cdots X_n \text{ es } A_n \text{ ENTONCES } Y \text{ es } B$$

donde las entradas X_i y la salida Y son ahora números (no borrosos), en lugar de términos lingüísticos (como en el caso anterior), y por lo tanto, los A_i y B son conjuntos borrosos sin interpretación directa, en lugar de etiquetas lingüísticas.

Sistemas borrosos tipo Takagi-Sugeno-Kang

En lugar de trabajar con reglas lingüísticas como las de la sección anterior, Takagi, Sugeno y Kang [117, 121] propusieron un nuevo modelo basado en reglas donde el antecedente estaba compuesto de variables lingüísticas y el consecuente se representaba como una función de las variables de entrada. La forma más habitual de esta clase de reglas es la siguiente:

$$\text{SI } X_1 \text{ es } A_1 \text{ y } \cdots X_n \text{ es } A_n \text{ ENTONCES } Y = p_1 \cdot X_1 + \cdots p_n \cdot X_n + p_0$$

siendo X_i las variables de entrada, Y la variable de salida, y p_i parámetros reales. Esta clase de reglas se conocen como reglas TSK en referencia a sus creadores.

La salida de un sistema borroso TSK que usa una base de conocimiento con m reglas se obtiene como la media ponderada de las salidas individuales proporcionadas por cada regla, Y_i, $(i = 1, \cdots, m)$, como sigue

$$\frac{\sum_{i=1}^{m} h_i \cdot Y_i}{\sum_{i=1}^{m} h_i}$$

siendo $h_i = T(A_1^i(x_1), \cdots, A_n^i(x_n))$ el grado de emparejamiento entre la parte antecedente de la regla i y las entradas actuales al sistema $x = (x_1, \cdots, x_n)$. T es un operador de conjunción que se modela mediante una t-norma.

Esta clase de sistemas no han sido utilizados en los desarrollos posteriores.

2.3.2 Elementos

En la figura 2.11 aparecen los elementos principales de un Sistema de Lógica Borrosa (SLB). Se ha escogido como modelo un sistema tipo Mamdani por ser éste el más usado en la literatura. Contiene cuatro componentes fundamentales [90]:

Interfaz de Borrosificación que realiza un escalado de los valores de las entradas para adecuarlos a los valores típicos para los que se define el sistema, y una "borrosificación" que convierte los datos de entrada en valores lingüísticos adecuados para la manipulación de éstos como entidades borrosas.

Base de Conocimiento formada por una "base de datos", que recoge la definición de las funciones de pertenencia de las entradas y el sistema, y una "base de reglas", que caracteriza y resume la política y objetivos del control de un experto por medio de un conjunto de reglas lingüísticas de control.

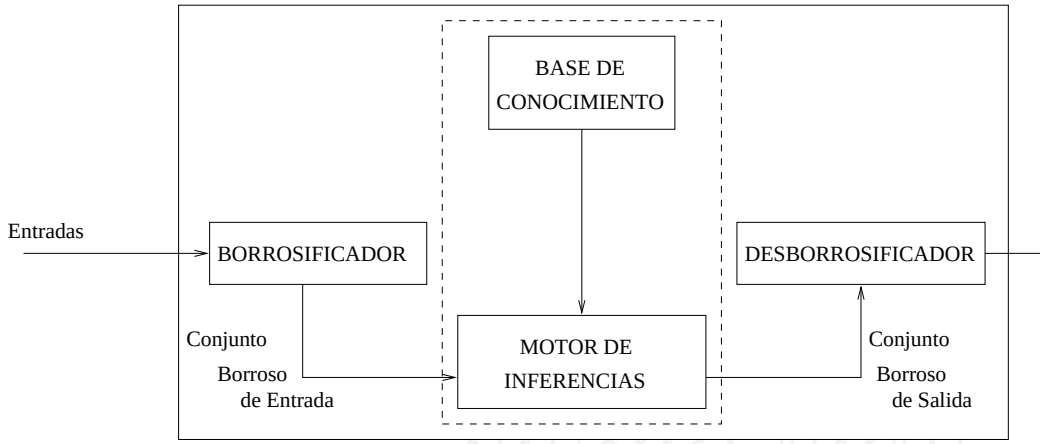


Figura 2.11 Sistema de Lógica Borrosa

Motor de Inferencias que inferirá las acciones del sistema empleando alguna representación de la implicación borrosa, así como de los procedimientos de inferencia en Lógica Borrosa.

Interfaz de Desborrosificación (o de concreción [93]) que convertirá la acción "borrosa" actualmente inferida en una acción concreta susceptible de aplicación sobre el proceso, y realizará un escalado para adecuar los rangos de salida para los que se ha definido el sistema con las entradas del proceso.

2.3.3 Base de conocimiento

Base de reglas

Consideremos como ejemplo ilustrativo un problema genérico que puede ser resuelto en términos de razonamiento aproximado por un ser humano experto en la materia, generalmente siguiendo el esquema de *modus ponens generalizado*. Esto quiere decir que el experto es capaz, mediante un conjunto de reglas o pautas de actuación que constituyen su experiencia sobre el tema de elaborar unas conclusiones o consecuentes a partir de unos hechos observados o antecedentes. El experto podrá transmitir sus conocimientos, al menos parcialmente, mediante un conjunto de N reglas del tipo *si-entonces*, con antecedentes relacionados por el conectivo "Y" en la mayoría de los casos (ver apartado 2.2.2 y siguientes).

La experiencia del ser humano, recogida y almacenada de esta manera, es lo que podemos denominar *base de reglas* (o *base de conocimiento*) por similitud con los métodos de trabajo y la terminología propios de los Sistemas Expertos.

Existen varios modos de derivación de las reglas, siendo éste el punto fuerte de numerosas investigaciones actualmente. Cabe destacar:

- Basados en los conocimientos de un experto, generalmente a través de una verbalización introspectivo de ésta, o a partir de cuestionarios cuidadosamente organizados.
- Basados en las acciones de control de un operador, en función de los datos de entrada-salida observados.
- Basados en un modelo borroso del proceso.
- Basados en aprendizaje (controladores auto-organizados).

Funciones de pertenencia. Base de datos

Los conceptos asociados con la Base de Datos se usan para caracterizar las reglas borrosas y la manipulación de los datos en un sistema borroso. Estos conceptos son definidos subjetivamente y están basados en la experiencia y juicio de un experto sobre el proceso. Algunos de estos aspectos serían la cuantificación y normalización de los universos de discurso, número de conjuntos borrosos o categorías lingüísticas de entradas y control, y la elección de las funciones de pertenencia asociadas a éstas últimas.

La elección de niveles de cuantificación tiene una influencia esencial en la precisión y "finura" del sistema obtenido. El número de categorías lingüísticas determina la granularidad del sistema final.

Existen dos métodos para definir conjuntos borrosos (mediante sus funciones de pertenencia), dependiendo de si los universos de discurso son discretos o continuos:

- Definición numérica: En el caso de universos discretos, la función de pertenencia de un conjunto borroso se representa como un vector de números cuya dimensión depende del grado de discretización.
- Definición funcional: En el caso de universos continuos, las funciones de pertenencia se expresan por medio de funciones regulares, típicamente triangulares o trapezoidales (sección 2.1.2).

La elección de la forma de las funciones de pertenencia se basa en criterios subjetivos o característicos de cada proceso en cuestión. Por ejemplo, si los datos medibles van a estar afectados por ruido, las funciones de pertenencia deberían ser suficientemente anchas para reducir la sensibilidad del sistema frente a éstos.

2.3.4 Motor de inferencias. Inferencia borrosa

El motor de inferencias representa el núcleo del SBRB, y agrupa toda la lógica de inferencia borrosa del sistema, de barrido de las reglas durante ésta, elección refinada de reglas a utilizar, etc.

La inferencia borrosa es el proceso mediante el cual se obtiene como consecuente un conjunto borroso a partir de unos antecedentes también borrosos. Hace uso de lo visto en la sección 2.2.3.

2.3.5 Borrosificación y desborrosificación

En mucho casos, especialmente dentro de la literatura de control, las entradas a un sistema borroso no son conjuntos borrosos sino valores numéricos concretos. Por eso es necesario establecer algún tipo de interfaz entre estos valores numéricos y el motor de inferencia borrosa. Por un lado habrá que elaborar conjuntos borrosos a partir de las entradas no borrosas (borrosificación) y por otro se habrá que calcular un valor numérico de salida a partir del conjunto borroso obtenido en el proceso de inferencia (desborrosificación).

Borrosificación

La primera parte del problema tiene una solución muy sencilla. Si x_0 es el valor concreto de una entrada del sistema borroso, parece lógico definir el correspondiente conjunto borroso de entrada mediante la siguiente función de pertenencia:

$$\mu_A(x) = \begin{cases} 1 & \text{si } x = x_0 \\ 0 & \text{en otro caso} \end{cases} \tag{2.66}$$

Desborrosificación (concreción)

Para calcular la señal de salida que se debe aplicar al proceso se puede recurrir a distintos métodos. Desafortunadamente, no existe ningún procedimiento sistemático para la elección de la estrategia de desborrosificación, siendo las más comúnmente usadas en la actualidad las siguientes:

1. Método de máxima pertenencia: el valor de salida es aquél cuyo grado de pertenencia al conjunto de salida $D(y)$ inferido sea máximo. Es decir:

$$\max_{y \in D(y)} \mu_D(y) = \mu_D(y_0) \tag{2.67}$$

2. Método del centro de gravedad (o centroide). Es sin duda el más usado de todos los métodos, ya que proporciona variaciones suaves y continuas de los valores de salida. Ésta se calcula como el centro de gravedad de la función de pertenencia del conjunto de salida $D(y)$. Esto es:

$$y_0 = \frac{\int y \mu_D(y) dy}{\int \mu_D(y) dy} \tag{2.68}$$

3. Método del máximo indexado: Este método calcula el centro de gravedad del subconjunto borroso del consecuente inferido, correspondiente a los puntos cuyo valor de pertenencia al consecuente es superior a un umbral dado.

Existen otros muchos métodos de desborrosificación [66, 90] siendo éste un tema de constante estudio en la literatura.

2.3.6 Sistemas Borrosos Aditivos

Kosko [70, 72] propone lo que él llama *Sistemas Borrosos Aditivos*. Un caso particular de estos sistemas es el llamado *modelo aditivo estándar* (en inglés *standard additive model*, SAM), cuyo diagrama de bloques se muestra en la Fig. 2.12.

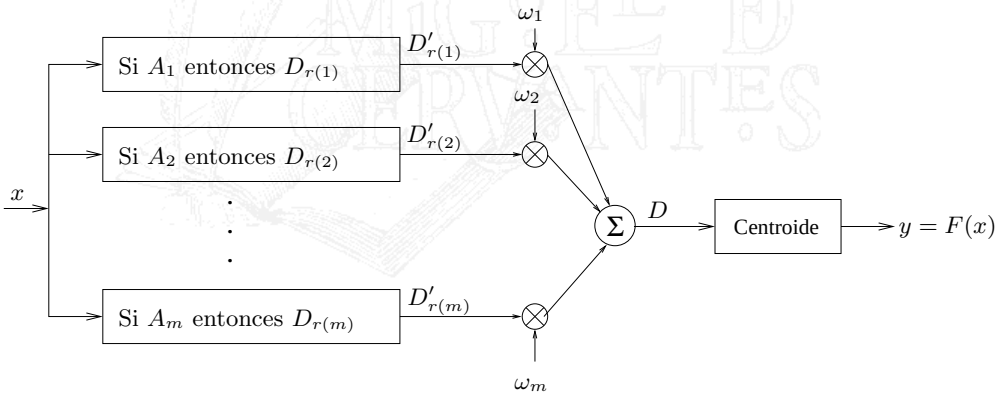


Figura 2.12 Diagrama de bloques de un *modelo aditivo estándar*

A partir de una entrada no borrosa x, el sistema dispara la parte *si* de las reglas, activando los conjuntos $A_{p(j)}$ hasta un cierto nivel dado por el coeficiente de activación $CA_j = A_{p(j)}(x)$. La parte *entonces* de las reglas, correspondiente a los conjuntos de salida $D_{r(j)}$ (siendo $p(j)$ y $r(j)$ funciones de indexado) se escala de acuerdo a este nivel. El escalado se realiza mediante el operador producto. Cada regla proporciona como salida un nuevo conjunto $D'_{r(j)}$, tal que

$$D'_{r(j)} = D_{r(j)} A_{p(j)}(x) \tag{2.69}$$

Estos conjuntos se escalan por un coeficiente ω_j y se suman para obtener el conjunto resultado D. Para obtener el valor de salida se suele usar el centroide de dicho conjunto.

$$F(x) = \text{centroide}(D) = \text{centroide}\left(\sum_j \omega_j D_{r(j)} A_j(x)\right) \tag{2.70}$$

En caso de tener múltiples antecedentes, y por lo tanto múltiples entradas x_1, x_2,... x_L en reglas de la forma

$$\text{Si } x_1 \text{ es } A_{p_1(j)} \text{ y } x_2 \text{ es } A_{p_2(j)} \text{ y ... y } x_L \text{ es } A_{p_L(j)} \text{ entonces } Y \text{ es } D_{r(j)}$$

el nivel de activación de la regla se calcula como el producto de los niveles individuales:

$$\prod_{i=1}^{L}(CA_i)_j = \prod_{i=1}^{L} A_{p_i(j)}(x_i) \tag{2.71}$$

$$D'_{r(j)} = D_{r(j)} \left[\prod_{i=1}^{L}(CA_i)_j \right] \tag{2.72}$$

Los niveles de activación $(CA_i)_j$ son el equivalente en operativa SAM de los grados de consistencia dc_j del caso general (secciones 2.2.3 y 2.2.4).

Kosko defiende esta arquitectura alegando distintas razones. Las más importantes son:

- El cálculo computacional que requiere esta arquitectura es pequeña. Se debe al hecho de que algunas de las variables intermedias que se necesitan para calcular la solución final pueden ser precalculadas y almacenadas. El centroide de salida, por ejemplo, puede escribirse:

$$F(x) = \text{centroide}(D) = \frac{\int yD(y)dy}{\int D(y)dy} \tag{2.73}$$

$$= \frac{\sum_j \omega_j A_j(x) V_j c_j}{\sum_{j=1}^{m} \omega_j A_j(x) V_j} \tag{2.74}$$

  donde los parámetros V_j y c_j son, respectivamente, el *hipervolumen* encerrado por la función de pertenencia D_j y su centroide.

  Como se puede deducir de la expresión, tanto los volúmenes V_j como los centroides c_j pueden ser precalculados. El cálculo real sólo será de los $A_j(x)$ y el uso de la expresión (2.74).

- Kosko propone diferentes operaciones borrosas a las propuestas por otros autores [66] sobre conjuntos borrosos. Específicamente, si se tiene en cuenta el método de *interpolación* (ver Fig. 2.10, en el apartado 2.2.3):

  1. Ya que las entradas que él propone son numéricas (no borrosas) no da a lugar usar la expresión allí apuntada, como un nivel. Los niveles se obtienen directamente de la función de pertenencia de la parte *si* de las reglas en el punto x.

     En el caso de que la entrada sea un vector (es decir, en el caso de que existan varias condiciones *si* en las reglas), Kosko sugiere la multiplicación de los niveles de disparo individuales, tal y como se muestra en la ecuación (2.71), en oposición al operador *min*. La razón para esto es que el multiplicador reduce el nivel de proposiciones en los que los *hechos* no coinciden con la parte *si* de la reglas. Sin embargo, el operador *min* no puede realizar esto.

Si las entradas son borrosas, los niveles se calculan como

$$A_j(A) = A_j \circ A = \int_{R^n} A(x)A_j(x)dx \tag{2.75}$$

2. Respecto a la combinación aditiva de la parte *entonces* de las reglas, Kosko dice que el operador *max* tiende a crear funciones de pertenencia que se aproximan más a pulsos rectangulares cuanto más reglas se combinan.

   Por otro lado, el operador *suma* tiene la tendencia a crear curvas con forma de campana, lo que sugiere la posibilidad de la existencia de un teorema del límite similar al teorema del límite central.

BIBLIOTECA VIRTUAL MIGUEL DE CERVANTES

3

CÁLCULO LINGÜÍSTICO

BIBLIOTECA VIRTUAL
MIGUEL DE CERVANTES

El lenguaje es una parte de nuestro organismo,
no menos complicada que éste.

-Ludwig Wittgenstein

3.1 INTRODUCCIÓN

Muchos problemas con los que nos encontramos en la vida cotidiana no son susceptibles de ser resueltos de manera numérica o precisa, y en muchas ocasiones una excesiva precisión puede hacer imposible una correcta solución. Tómese como ejemplo el caso de aparcar un automóvil. La mayoría de la gente es capaz de aparcar un automóvil fácilmente porque la posición final y la orientación de éste no se especifica con precisión. Si se hiciese, la dificultad de aparcar se incrementaría enormemente con la precisión, y llegaría un punto en que sería imposible para los humanos realizar esa actividad. El problema es más fácil de resolver cuando se formula de forma imprecisa y no se usan métodos de cálculo tradicionales.

Este tipo de cálculo recibe el nombre de *Razonamiento Aproximado* [32, 161, 164] (sección 2.2.3), dentro del cual se encuentra un área en el que los problemas se resuelven usando información de carácter lingüístico. Estas técnicas se reúnen bajo el nombre de *Computación con palabras*, o *Cálculo lingüístico* (CWW del inglés *Computing with Words* [166]), introducido por Zadeh [160] en 1973.

Dos son los imperativos para usar CWW: primero, cuando la información sea demasiado imprecisa como para trabajar con números, y segundo, cuando exista una cierta tolerancia a la imprecisión que pueda ser explotada para nuestros fines. A pesar de que una aproximación lingüística sea menos precisa que una aproximación numérica, hay ocasiones en que esto supone una ventaja: las descripciones basadas en texto van a ser mejor entendidas por los seres humanos, por ejemplo. Este lenguaje natural, dentro de CWW ha sido típicamente modelado usando conjuntos borrosos.

En el presente capítulo se realizará una aproximación al campo de CWW, enmarcándolo dentro de la Teoría Computacional de las Percepciones (sección 3.2), de

acuerdo con la Fig. 3.1. A continuación se revisarán los principales aspectos de esta teoría, tales como las variables lingüísticas (sección 3.3), las lógicas borrosas (sección 3.4), los grafos borrosos (sección 3.5), la granulación de la información (sección 3.6), las restricciones generalizadas (sección 3.7) y una pequeña introducción de la Teoría de la Posibilidad de Dubois y Prade (sección 3.8).

Aproximación histórica

En 1971 Zadeh [159] introduce la noción de semánticas borrosas cuantitativas, explicando cómo se pueden expresar términos lingüísticos como conjuntos borrosos en universos de discurso, y cómo una combinación lógica de términos lingüísticos puede tener una contrapartida numérica en términos de operaciones borrosas. El modo en que Zadeh presenta estos universos de discurso es básicamente en forma de escalas numéricas, de modo que lo que las semánticas cuantitativas borrosas proporcionan es esencialmente una herramienta que hace de interfaz entre números y símbolos. Esta herramienta respeta la continuidad de los universos de discurso y la gradación de los conceptos al manipular los símbolos.

El primer intento de modelar los procesos deductivos con categorías borrosas fue el *modus ponens generalizado* publicado en 1973 (Zadeh [160]). Este artículo tuvo un impacto inmediato, considerándose el origen de la literatura sobre controladores borrosos, desarrollados por Mamdani en los años setenta (por ejemplo [84]). Estas investigaciones dieron lugar posteriormente al cálculo de variables lingüísticas, a los números borrosos, a la teoría de posibilidades (en [164] las restricciones borrosas son vistas por primera vez como distribuciones de probabilidad) y en una teoría general para el razonamiento aproximado [162] que sistematiza el principio de combinación/proyección del modus ponens generalizado.

En los 80, el campo de la Inteligencia Artificial opta por el uso de métodos denominados *sintácticos* (término que hace referencia a lógicas multivaluadas y factores de certidumbre) frente a la aproximación semántica, que se basa en la combinación y proyección de relaciones borrosas en universos (a menudo numéricos) de discurso. Estos últimos métodos parecen recuperar influencia a partir del comienzo de los 90.

En los 80, Zadeh desarrolla diversos aspectos de la teoría de CWW, como son los cuantificadores, los silogismos borrosos, el concepto de *habitualidad* y disposiciones. En los mismos años, Dubois y Prade desarrollan el núcleo de la Teoría de Posibilidades [31, 30, 32] que ampliarán en los años siguientes [27, 48].

Se ha asistido en los últimos años a un interés creciente sobre el tema, que ha llevado a formular una Teoría Computacional de las Percepciones [170], que pretende ser el marco de trabajo bajo el cual se encuentran diversos aspectos, como CWW, la teoría de Posibilidades, la granulación de la información o las restricciones generalizadas.

3.2 DEFINICIÓN DE CWW

La expresión *Calcular con palabras* fue acuñada por Zadeh [160]. En su sentido tradicional, el término *cálculo* indica una manipulación de números y de símbolos[1]. En contraste, los seres humanos usamos sobre todo palabras en nuestros razonamientos y cálculos, llegando de este modo a conclusiones expresadas con palabras a partir de premisas descritas en lenguaje natural o con la forma de percepciones mentales. Las palabras, tal y como las usamos los seres humanos tienen connotaciones borrosas; lo mismo se puede decir del papel que juegan éstas en CWW.

Según [137], CWW proporciona un modelo matemático para la teoría del lenguaje natural. Captura las características y las propiedades más destacadas de los aspectos semánticos del lenguaje natural, siendo especialmente efectiva en el manejo de los aspectos pragmáticos. Los fundamentos de CWW se basan en la teoría de conjuntos borrosos y en la lógica borrosa. Por medio de éstos es posible desarrollar una aplicación tal que una máquina pueda extraer conclusiones a partir de un conjunto de entradas expresadas en lenguaje natural o que contengan una cierta dosis de ambigüedad.

La idea básica que subyace al concepto de CWW es que, en general, la información se transfiere restringiendo los valores que pueden tomar las variables lingüísticas [69]:

1. La información dada se representa como una colección de proposiciones expresadas en lenguaje natural.
2. Cada proposición se ve como una restricción implícita sobre una variable implícita: la proposición expresada en lenguaje natural limita los significados posibles de la variable.
3. Por necesidades del cálculo, es necesario hacer explícitas esas restricciones.
4. Esto lleva al concepto de *forma canónica* (sección 3.7.1) que sirve para explicitar la variable restringida y la relación de restricción.
5. Una vez que las restricciones implícitas en las premisas de partida se hacen explícitas mediante el uso de formas canónicas, se pueden usar las reglas de inferencia de la lógica borrosa para propagar las restricciones borrosas de las premisas a las conclusiones.
6. Finalmente, las restricciones inducidas se traducen a un lenguaje natural.

Algunos de los problemas más representativos que se tratan en este campo son los siguientes [170]:

[1] El término usado en la bibliografía anglosajona, *computing* tiene connotaciones más amplias, ya que sugiere de alguna manera que estas operaciones son susceptibles de hacerse en un ordenador mediante algoritmos automatizables

1. Solución de un sistema de ecuaciones lineales con coeficientes lingüísticos en lugar de numéricos.

2. Operaciones sobre funciones definidas mediante reglas borrosas *si-entonces* con valores no numéricos. Por ejemplo, interpolación o maximización de una función definida mediante conjuntos borrosos de la siguiente forma:

   Si X es *pequeña* entonces Y es *pequeña*
   Si X es *mediana* entonces Y es *grande*
   Si X es *grande* entonces Y es *pequeña*

3. Cálculo de probabilidades lingüísticas. Por ejemplo: Una caja contiene bolas de diversos tamaños. La mayoría de ellas son grandes y unas pocas son pequeñas. ¿Cuál es la probabilidad de que al sacar una bola al azar ésta no sea ni grande ni pequeña?

4. Razonamientos con silogismos:

   La mayor parte de los hombres jóvenes tienen buena salud
   Roberto es joven
   ¿Cuál es la probabilidad de que Roberto tenga buena salud?

Los problemas de esta clase no llevan por sí mismos a una solución si empleamos los métodos tradicionales de la lógica, la teoría de probabilidades o el análisis numérico. Caen dentro del dominio del cálculo con palabras. Tal dominio viene definido sobre cuatro posibles ejes [170]:

1. *El eje "no sé":* En este caso, los valores de las variables y/o de los parámetros no se conocen con la suficiente precisión para justificar el uso de métodos convencionales de cálculo numérico. Un ejemplo es la toma de decisiones en entornos en que las probabilidades no puedan estimarse con precisión.

2. *El eje "no hace falta":* En este caso existe una tolerancia a la imprecisión que puede ser explotada para lograr un método más robusto, más tratable, una solución de bajo coste o una mayor semejanza con la realidad. Un ejemplo es el problema de aparcar un coche.

3. *El eje "no puedo solucionarlo":* En este caso el problema no puede ser resuelto usando técnicas numéricas. Un ejemplo es el problema de conducción automática de un vehículo entre el tráfico de una ciudad.

4. *El eje "no puedo definirlo":* En este caso, alguno de los conceptos que se desean definir es demasiado complejo para ser definido usando criterios numéricos.

3.2.1 Teoría Computacional de las Percepciones

Según Zadeh [170], CWW proporciona una metodología para implementar lo que podemos llamar *Teoría Computacional de las Percepciones* (TCP), una teoría general que

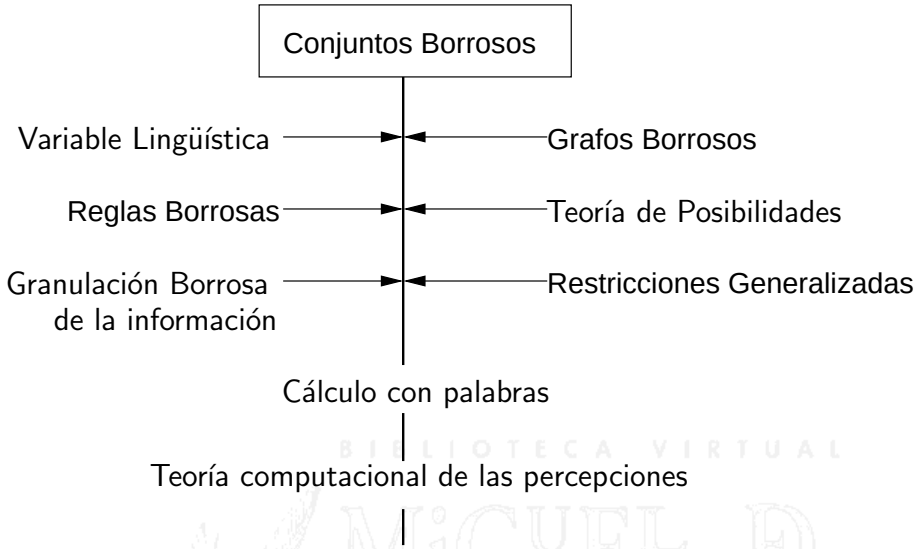


Figura 3.1 Estructura conceptual de la Teoría Computacional de Percepciones

abarcaría diversos aspectos del razonamiento humano en relación con las percepciones. Las principales ramas que forman parte de esta teoría son las siguientes (Fig. 3.1):

Variable lingüística: Noción introducida por Zadeh [161], ampliamente conocida (ver sección 2.2.1).

Reglas Borrosas: Planteamiento del conocimiento en forma de reglas *si-entonces* (capítulo 2).

Granulación Borrosa de la Información: división de la información en gránulos borrosos, siendo un gránulo una agrupación de objetos fusionados siguiendo algún criterio.

Grafos Borrosos: dependencia funcional $f : X \rightarrow Y$, donde X e Y son variables lingüísticas definidas sobre dos espacios U y V.

Restricciones Generalizadas: generalización del concepto de restricción de variables en la lógica borrosa.

Teoría de Posibilidades: metodología para representar el conocimiento vago o incompleto en términos de distribuciones de posibilidad [162].

La idea que subyace a la relación entre CWW y TCP es conceptualmente simple. En la TCP las percepciones y las preguntas se formulan mediante proposiciones de lenguaje natural. Estas proposiciones se procesan usando métodos basados en CWW que conducen a respuestas o a resultados. Algunos ejemplos de formulación de percepciones mediante proposiciones expresadas en lenguaje natural, serían los siguientes:

- David es bastante inteligente
- Comer demasiado causa obesidad
- La mayoría de los suecos son altos
- Es probable que llueva mañana
- Es muy improbable que el precio del petróleo suba de una manera significativa en un futuro próximo

A partir de las percepciones, usando técnicas basadas en CWW se pueden extraer las conclusiones:

(Percepción)	María es joven
(Percepción)	Marta es unos pocos años mayor que María
(Conclusión)	Marta tiene (joven+poco más) años

Un punto importante que debe ser tenido en cuenta es que los sistemas de lógica clásica, tales como la lógica proposicional, la lógica de predicados, la lógica modal e incluso las técnicas basadas en Inteligencia Artificial [145] para el procesado de lenguaje natural y la representación del conocimiento están concebidas de tal manera que tienen una relación fundamental con las proposiciones expresadas en lenguaje natural. La principal diferencia de estas aproximaciones con CWW según Zadeh [170] es que la metodología de CWW -que está basada en la lógica borrosa- proporciona un lenguaje mucho más expresivo para la representación del conocimiento y una maquinaria mucho más versátil para el razonamiento y el cálculo.

En las secciones siguientes se repasarán las distintas ramas que forman en árbol de la TCP, haciendo hincapié en las partes menos conocidas de éste.

3.3 VARIABLE LINGÜÍSTICA

Un concepto que juega un papel fundamental dentro de las aplicaciones de CWW es el de *variable lingüística* [161, 66] ya definido en la sección 2.2.1 dentro del marco de la lógica borrosa. Una nueva definición formal más próxima al campo del cálculo lingüístico sería la siguiente:

Definición 15 (Variable lingüística)
Una variable lingüística se caracteriza por una quíntupla $(H, T(H), U, G, M)$ en la que H es el nombre de la variable; $T(H)$ (o simplemente T) denota el conjunto de términos de H, es decir, el conjunto de nombres de los valores lingüísticos de H, siendo cada valor una variable borrosa denotada generalmente por X, que toma valores en un universo de discurso U asociada a una variable base u; G es una regla sintáctica (que suele tomar la forma de una gramática) para los nombres de los valores de H; y M es una regla

semántica para asociar su significado a cada H, $M(X)$, que es un subconjunto borroso de U.

La noción general de variable lingüística ha sido alterada por diversos autores tratando de conseguir una representación más acorde con las herramientas que cada uno utiliza. En [76], Lawry plantea una interpretación alternativa junto con la noción de *descripción lingüística* de un valor o conjunto de valores. Una descripción lingüística es un conjunto borroso expresado en palabras, donde los valores de pertenencia cuantifican la idoneidad de una determinada palabra como etiqueta para el valor que se considere. Este concepto se emplea en razonamiento con cuantificadores lingüísticos [94, 128], siendo éstos definidos como descripciones lingüísticas de valores de probabilidad (en el sentido Bayesiano del término). Yager matiza el concepto de variable lingüística [141, 143] y amplía la familia de variables introduciendo conceptos como *variables verísticas* [151] o *variables de incertidumbre* [148].

3.4 LÓGICAS BORROSAS

En la toma de decisiones se hace necesario una extensión de los conjuntos y las lógicas de decisión a un dominio borroso. En muchas ocasiones la decisión a tomar no puede ser simplemente *cierto* o *falso*, sino que existen una multitud de matices y estados intermedios. Este hecho es capturado convenientemente por los conjuntos borrosos (ver capítulo 2).

Cuatro características de los conjuntos borrosos son de gran importancia en el campo de CWW [67]:

1. Los conjuntos y las lógicas borrosas poseen una mayor capacidad para capturar medidas de incertidumbres que las técnicas clásicas [33, 75]. Esto supone un mejor puente entre los modelos matemáticos y la realidad física asociada.

2. La capacidad de los conjuntos borrosos de capturar las transiciones graduales de un conjunto a otro mejora su capacidad expresiva. De este modo se puede tratar un rango más amplio de problemas.

3. Otra importante característica es su potencial para capturar la vaguedad inherente a los términos lingüísticos de las proposiciones en lenguaje natural, proporcionando una mejor capacidad para modelar de un modo realista el razonamiento humano "de sentido común" [165], la toma de decisiones y otros aspectos del conocimiento humano.

4. Las lógicas y los conjuntos borrosos son potentes herramientas para manejar la complejidad y controlar costes computacionales.

3.5 GRAFOS BORROSOS

El concepto de *grafo* borroso se introdujo por primera vez en 1971 por Zadeh, y de forma más explícita en 1974 [166]. En su forma más primitiva, sirvió de base a los trabajos de Mamdani y Assilian sobre control borroso [84].

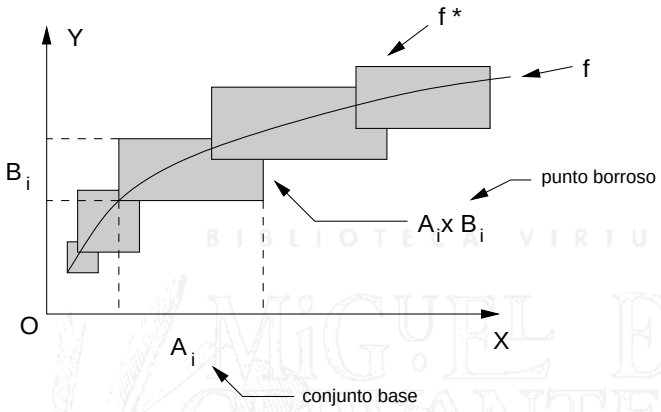


Figura 3.2 Representación de una función y su grafo borroso.

Un *grafo borroso* [78, 166] f^* (tal y como se puede ver en la Fig. 3.2), de una dependencia funcional $f : X \rightarrow Y$, donde X e Y son variables lingüísticas en U y en V, respectivamente, proporciona una representación aproximada y comprimida de f de la forma

$$f^* = A_1 \times B_1 + A_2 \times B_2 + \cdots + A_n \times B_n$$

o de forma más compacta

$$f^* = \sum_{i=1}^{n} A_i \times B_i \tag{3.1}$$

donde los A_i y los B_i son subconjuntos borrosos contiguos de U y V respectivamente, $A_i \times B_i$ es el producto cartesiano de A_i y B_i, y $+$ es el operador disyunción, para el que se suele tomar la unión. Expresado de forma más explícita usando las funciones de pertenencia de f^*, A_i y B_i, se llega a

$$\mu_{f^*}(u, v) = \vee_i(\mu_{A_i}(u) \wedge \mu_{B_i}(v)), \tag{3.2}$$

donde $\wedge = \min$, $\vee = \max$, $u \in U$ y $v \in V$. Generalizando, en lugar de los operadores $\vee$ y $\wedge$ podrían usarse t-normas y s-normas[66].

Alternativamente, un grafo borroso puede representarse como una relación f^*

f^*	A	B
	A_1	B_1
	A_2	B_2
	$\vdots$	$\vdots$
	A_n	B_n

o como una colección de reglas *Si-entonces* borrosas

$$\begin{array}{ll} f^* & \text{Si } X \text{ es } A_1 \text{ entonces } Y \text{ es } B_1 \\ & \text{Si } X \text{ es } A_2 \text{ entonces } Y \text{ es } B_2 \\ & \cdots \\ & \text{Si } X \text{ es } A_n \text{ entonces } Y \text{ es } B_n \end{array}$$

entendiendo que la regla "Si X es A_i entonces Y es B_i" se interpreta mediante la restricción conjunta de X e Y definida por

$$(X, Y) \ es \ (A_i \times B_i)$$

3.5.1 Operaciones sobre grafos borrosos

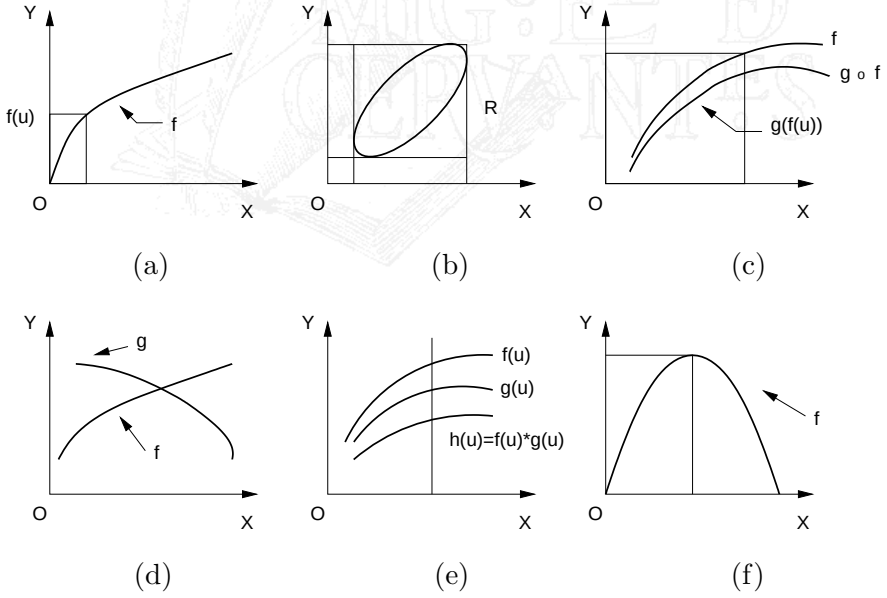


Figura 3.3 Algunas operaciones básicas sobre funciones y relaciones. (a) Evaluación y asignación. (b) Proyección. (c) Composición. (d) Intersección. (e) Combinación. (f) Maximización

Las operaciones que se pueden realizar sobre grafos borrosos son una generalización de las operaciones correspondientes que se realizan sobre funciones y relaciones no borrosas. Algunas de las operaciones básicas aparecen en la figura 3.3.

Las operaciones necesarias pueden ser simplificadas de una manera notable si un operador $*$ es monótonamente no-decreciente, es decir, si a, b, a' y b' son números reales, entonces

$$\begin{array}{lll} a' \geq a, \ b' \geq b & \rightarrow & a' * b' \geq a * b \\ a' \leq a, \ b' \leq b & \rightarrow & a' * b' \leq a * b \end{array} \tag{3.3}$$

Para estas operaciones se puede demostrar que $*$ es distributivo respecto a $\vee$ (max) y a $\wedge$ (min). Por lo tanto

$$a * (b \vee c) = (a * b) \vee (a * c) \tag{3.4}$$
$$a * (b \wedge c) = (a * b) \wedge (a * c) \tag{3.5}$$

Esto implica que si $f^* = \sum_i A_i \times B_i$ es un grafo borroso y C es un conjunto borroso, entonces

$$C * \left(\sum_i A_i \times B_i \right) = \sum_i C * (A_i \times B_i) \tag{3.6}$$

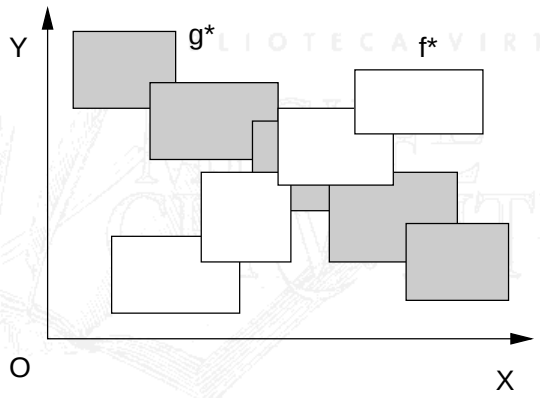


Figura 3.4 Intersección de los grafos borrosos f^* y g^*.

Considérese el problema de encontrar la intersección de los grafos borrosos f^* y g^* (ver Fig. 3.4), donde $f^* = \sum_i A_i \times B_i$ y $g^* = \sum_j C_j \times D_j$. Para este caso

$$f^* \cap g^* = \sum_{i,j} (A_i \times B_i) \cap (C_j \times D_j)$$

que usando la distributividad de $\cap$ se reduce a

$$f^* \cap g^* = \sum_{i,j} (A_i \cap C_j) \times (B_i \cap D_j)$$

Este resultado tiene una aplicación inmediata a la interpolación de reglas *Si-entonces* borrosas. El problema de la interpolación se puede expresar como una pregunta de inferencia

$$\frac{\begin{array}{rl}(X, Y) & \text{es } \sum_i A_i \times B_i \\ X \text{ es } A & \end{array}}{Y \text{ es } ?B}$$

Representando la premisa mayor como un grafo borroso $f^* = \sum_i A_i \times B_i$ y la premisa menor como una extensión cilíndrica, $\overline{A}$, del conjunto A (ver Fig. 3.5), el cálculo de $?B$

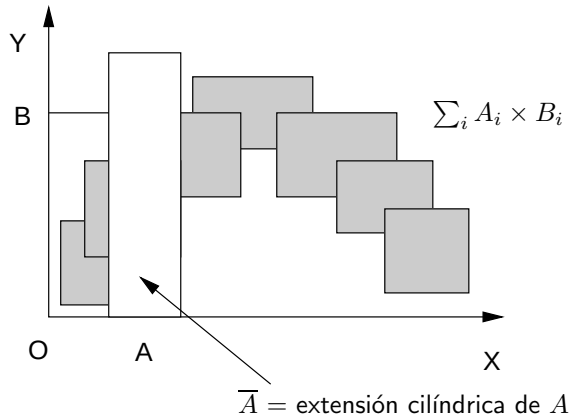


Figura 3.5 Intersección de f^* y $\overline{A}$

se reduce a encontrar la intersección de f^* con $\overline{A}$, y proyectar el conjunto borroso de resultado sobre V, el dominio de Y. Esto es,

$$B = projV\left(\overline{A} \cap \left(\sum_i A_i \times B_i\right)\right) \tag{3.7}$$

$$= projV\left(\sum_i (A \cap A_i) \times B_i\right) \tag{3.8}$$

$$= \sum_i m_i \wedge B_i \tag{3.9}$$

donde $m_i = sup(A \cap A_i)$ representa el grado de solapamiento entre A y A_i.

Un ejemplo más avanzado consiste en la combinación de f^* y g^* mediante el operador *mínimo* (min), siendo $f^* = \sum_i A_i \times B_i$ y $g^* = \sum_i A_i \times C_i$, lo que resulta

$$f^* \min g^* = \sum_i A_i \times (B_i \times C_i)$$

donde $B_i \times C_i$ es el mínimo de B_i y C_i calculado a partir del principio de extensión [161, 66]. Este resultado hace posible calcular la intersección $F \cap G$ de conjuntos borrosos F y G cuyas funciones de pertenencia μ_F y μ_G se representen cualitativamente en forma de reglas *Si-entonces*:

$$\begin{array}{ll} F\text{: Si } X \text{ es } A_i \text{ entonces } \mu_F \text{ es } B_i & i = 1, \cdots, n \\ G\text{: Si } X \text{ es } A_i \text{ entonces } \mu_G \text{ es } C_i & i = 1, \cdots, n \end{array}$$

El concepto de grafo borroso tiene una importante conexión con la representación de relaciones borrosas. Si R es una relación borrosa con atributos que toman valores lingüísticos (ver Fig. 3.6) entonces R se puede representar como un grafo borroso

$$R = R_{11} \times R_{12} \times R_{13} + R_{21} \times R_{22} \times R_{23} + R_{31} \times R_{32} \times R_{33}$$

R	A_1	A_2	A_3
	R_{11}	R_{12}	R_{13}
	R_{21}	R_{22}	R_{23}
	R_{31}	R_{32}	R_{33}

atributos

valores lingüísticos

Figura 3.6 Relación con atributos borrosos

La representación de las relaciones borrosas como grafos borrosos se puede aplicar a la representación de tablas de diagnóstico en las que las entradas son los valores lingüísticos de los test y las faltas correspondientes [166].

3.6 GRANULACIÓN DE LA INFORMACIÓN

3.6.1 Gránulos

El punto de partida en CWW según Zadeh [167] es el concepto de *gránulo*. En esencia, un gránulo es un conjunto borroso de puntos con la forma de una agrupación de elementos que se fusionan siguiendo diversos criterios, tales como indistinguibilidad, similitud, proximidad o funcionalidad (Fig. 3.7). Una palabra w es la etiqueta de un gránulo g, y respectivamente g es la representación de w. Una palabra puede ser *atómica* (como en el caso de *joven*) o *compuesta* (como en *no muy joven*).

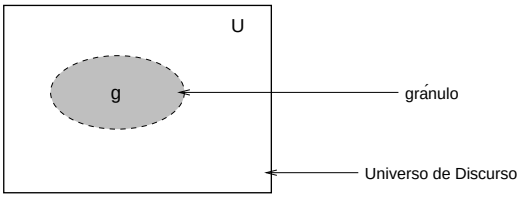


Figura 3.7 Representación de un gránulo

En CWW, un gránulo g que denota a w se ve como una *restricción borrosa* sobre una variable. Como ejemplo, considérese la proposición *María es joven*. En este caso, *joven* es la etiqueta del gránulo *joven* (nótese que por simplicidad se usa el mismo símbolo para una palabra y su gránulo). El conjunto borroso *joven* juega el papel de una restricción borrosa sobre la edad de María.

Una suposición básica en CWW es que la información se transmite por medio de restricciones sobre los valores de las variables. La información se asume que consiste en una colección de proposiciones expresadas en un lenguaje, bien natural o bien sintético.

Un problema genérico básico en CWW sería el siguiente: Se dispone de una colección dada de proposiciones expresadas en lenguaje natural que constituyen el conjunto inicial de datos (CID). A partir del CID se desea inferir una respuesta a una pregunta realizada en lenguaje natural. A la respuesta, que también será expresada en lenguaje natural, nos referiremos como conjunto terminal de datos (CTD). El problema consiste en obtener CTD a partir de CID.

A continuación se exponen unos problemas que servirán para ilustrar estos conceptos:

1. Supóngase que una función f, $f : U \rightarrow V$, $X \in U$, $Y \in V$está descrita en palabras mediante las siguientes reglas *Si-entonces* borrosas

   f: Si X es *pequeña* entonces Y es *pequeña*
   Si X es *mediana* entonces Y es *grande*
   Si X es *grande* entonces Y es *pequeña*

   Esto implica que la función f se aproxima por el grafo borroso f^* donde

   $$f^* = \text{ pequeño} \times \text{pequeño} + \text{medio} \times \text{grande} + \text{grande} \times \text{pequeño}$$

   donde + y $\times$ denotan respectivamente la disyunción y el producto Cartesiano. A una expresión de la forma $A \times B$, donde A y B son palabras, nos referiremos como a un *gránulo cartesiano*. En este sentido, un grafo borroso se puede ver como una disyunción de gránulos cartesianos. En esencia, un grafo borroso proporciona una aproximación a una función o a una relación [166].

   En el ejemplo actual, el CID consiste en un conjunto de reglas borrosas, f. La pregunta es "¿Cuál es el máximo valor de f^*?".

2. Una caja contiene bolas de diversos tamaños. La mayoría de ellas son grandes y unas pocas son pequeñas. ¿Cuál es la probabilidad de que al sacar una bola al azar ésta no sea ni grande ni pequeña? En este caso el CID es una descripción verbal del contenido de la caja; el CTD será la probabilidad deseada.

3. Sean X e Y variables aleatorias independientes que toman valores en un conjunto finito $V = \{v_1, \cdots, v_n\}$ con probabilidades $p_1, \cdots, p_n$ y $q_1, \cdots, q_n$ respectivamente. Para simplificar la notación se usarán los mismos símbolos para denotar X, Y, y sus valores concretos; p y q denotarán las probabilidades de X e Y respectivamente. Supóngase que las distribuciones de probabilidad de X e Y se describen mediante palabras a través de reglas *Si-entonces*:

   P: Si X es *pequeña* entonces p es *pequeña*
   Si X es *mediana* entonces p es *grande*
   Si X es *grande* entonces p es *pequeña*

   y

   Q: Si Y es *pequeña* entonces q es *grande*
   Si Y es *mediana* entonces q es *pequeña*
   Si Y es *grande* entonces q es *grande*

donde los gránulos *pequeña*, *mediana* y *grande* son los valores de las variables lingüísticas X e Y en sus respectivos universos de discurso. Estos conjuntos de reglas representan el CID en este ejemplo. La pregunta es "¿Cómo se puede describir en palabras la distribución de probabilidad conjunta de X e Y?". Esta distribución de probabilidad será el CTD.

3.6.2 Caracterización de la granularidad

La noción de granulación, según Zadeh [168, 169], es la actividad por la que un objeto A produce una colección de gránulos de A. La *granulación borrosa de la información* no se refiere a un solo gránulo sino a una colección de gránulos borrosos que resultan de la granulación (borrosa o no) de un objeto.

Según T.Y. Lin la noción de granulación de Zadeh lleva implícita una cierta simetría innecesaria. Alternativamente propone en [79] una nueva visión que lleva a una formalización de la noción de granulación borrosa. Para Lin, cada gránulo representa un concepto primitivo y la granulación en su conjunto representa un *conocimiento*.

Los principios organizativos que conforman los gránulos tienen que ver con las nociones de *similitud* y *funcionalidad* [69]. En términos relacionales, la similitud es una relación borrosa reflexiva, simétrica y transitiva. Puede formarse una familia entera de relaciones de similitud borrosa a partir de las funcionalidades. Dado que se pueden relacionar diferentes funcionalidades, se pueden relacionar también diferentes familias de gránulos por correspondencias funcionales.

Por lo tanto, la estructura de los gránulos está caracterizada por dos clases de relaciones [69]:

1. *Relativas a la similitud:* tolerancias borrosas (reflexivas y simétricas) y equivalencias borrosas (relaciones de tolerancia transitivas).
2. *Relativas al orden:* preorden y orden (parcial).

Las estructuras relacionales relativas al orden son de particular interés debido a que están ligadas al conocimiento humano. El análisis de problemas complejos requieren que las personas empleemos múltiples niveles de abstracción, intercalando simultáneamente capas de significado y representación.

3.6.3 Captura de la granularidad. Modelos de semiótica relacional

La granulación es un principio de organización que está relacionado con la abstracción. Cuando se forman los gránulos a un determinado nivel de resolución, algunas

características no se pueden distinguir debido a la falta de información, y algunas características son irrelevantes en el contexto dado. Para explorar las estructuras granulares en CWW se necesitan algunos conceptos de agregación, adquisición representación y procesado computacional de las estructuras granulares. Las estructuras semióticas[2] son ese medio de representación [69, 110].

Esquema para la construcción de modelos semióticos relacionales. Para la construcción de modelos que capturen la relevancia de una situación dada es necesario aplicar los siguientes pasos:

1. Selección de las características que se suponen significativas: descartar las irrelevantes.
2. Reducción de los detalles agrupando los casos similares juntos (partición de los datos en clases o bloques equivalentes).
3. Representación de los resultados del punto anterior usando un modelo.

Aproximación del significado en modelos semióticos borrosos. Las relaciones borrosas son una importante herramienta para la aproximación del significado, ya que una propiedad borrosa puede estar presente totalmente o en un cierto grado. Esto hace que el cálculo de relaciones pueda generar significados aproximados.

En combinación con el modelado anteriormente descrito, las relaciones borrosas también proporcionan una representación del lenguaje de alto nivel de conocimiento, por lo que se podrán usar procedimientos computacionales para la identificación y representación de los gránulos que contienen de forma intrínseca los datos empíricos.

Existen otras aproximaciones al tema de la captura de la granularidad y la formación de gránulos a partir de unos datos dados: Lin [79] lo hace a partir de *agrupaciones*, tanto en un nivel como multinivel; Polkowski y Skowron [107] realizan una granulación adaptativa usando un álgebra *mereológica*; Pedrycz y otros en [101] plantean un método para la formación de conceptos a partir de la información dada, usando los datos disponibles y una semántica para generar funciones de pertenencia.

3.7 RESTRICCIONES GENERALIZADAS

3.7.1 Restricciones borrosas y formas canónicas

Una proposición p en un lenguaje natural se puede ver como una red de restricciones borrosas [167]. Las restricciones que contiene p, por medio de agregación resultan en una restricción borrosa general que se puede representar como una expresión de la forma

$$X \text{ es } R$$

[2]La semiótica es el estudio general de los signos.

donde R es una relación borrosa restrictora y X es la variable restringida. A esta expresión se le llama *forma canónica* [167]. La función de una forma canónica es poner en evidencia las restricciones borrosas que se encuentran implícitas en p. Esto se representa de forma esquemática como

$$p \rightarrow X \text{ es } R$$

donde la flecha $\rightarrow$ denota *explicitación* La variable X puede ser condicionada y/o tomar un valor vectorial.

Desde esta perspectiva, el significado de p se define mediante dos procedimientos. El primero actúa sobre una base de datos explicativa (BDE) y devuelve la variable restringida X. El segundo actúa sobre la BDE y devuelve la relación restrictora R.

Una BDE es una colección de relaciones de modo tal que se defina el significado de p. Las relaciones están vacías, esto es, consisten en una serie de nombres, relaciones, atributos y dominios de los atributos sin entrada en las relaciones. Cuando hay entradas en la BDE, se dice que la información se *inicializa* [153], y se denota por BDEI. La BDEI puede ser vista como una descripción de un mundo posible en una semántica de un mundo posible, mientras que la BDE define una colección de posibles mundos, correspondiendo cada posible mundo de la colección a una entrada instantánea particular de BDE.

Como ejemplo, considérese la siguiente proposición

$$p = \text{María no es muy joven}$$

Supóngase que la base de datos explicativa es de la forma

$$\text{BDE} = \textsc{Población}[\text{Nombre; Edad}] + \textsc{Joven}[\text{Edad;}\mu]$$

donde Población es una relación con argumentos Nombre y Edad, Joven es una relación con argumentos Edad y μ; + es una disyunción. En este caso la variable restringida es la edad de María, que puede ser expresada en términos de la BDE como

$$X = \text{Edad(María)} =_{Edad} \textsc{Población}[\text{Nombre=María}]$$

Esta expresión especifica el procedimiento que actúa sobre la BDE y devuelve X. Más específicamente, en este procedimiento Nombre se inicializa a María y la relación resultante se proyecta sobre *Edad*, lo que nos conduce a la edad de María.

La relación restrictora R viene dada por

$$R = ({}^2\textsc{Joven})'$$

lo que implica que el operador *muy* se interpreta como la operación cuadrado (^{2}), y la negación *no* como el operador complemento ($'$).

Equivalentemente, R se puede expresar

$$R = \text{JOVEN}[\text{Edad}; 1 - \mu^2].$$

Un ejemplo más complejo sería de la forma

$$p = \text{Pedro vive en una pequeña ciudad cerca de Madrid}$$

y supóngase que la base de datos explicativa es

$$\begin{aligned}\text{DBE} = \quad & \text{POBLACIÓN}[\text{Nombre; Residencia}] + \text{PEQUEÑA}[\text{Ciudad; } \mu] \\ & + \text{CERCA}[\text{Ciudad1; Ciudad 2; } \mu]\end{aligned}$$

En este caso

$$X = \text{Residencia(Pedro)} =_{Residencia} \text{POBLACIÓN}[\text{Nombre} = \text{Pedro}]$$

y

$$R = \text{PEQUEÑA}[\text{Ciudad;}\mu] \cap_{Ciudad1} \text{CERCA}[\text{Ciudad2} = \text{Madrid}]$$

En R, el primer término es el conjunto borroso de todas las ciudades pequeñas, el segundo es el conjunto borroso de todas las ciudades que están cerca de Madrid, y $\cap$ denota la intersección entre estos conjuntos.

Hasta ahora solamente se han tenido en cuenta restricciones de la forma "X es R", pero pueden tener una amplia variedad de formas. Por ejemplo, una restricción condicional será de la forma

$$\text{Si } X \text{ es } R \text{ entonces } Y \text{ es } S$$

lo que también se puede escribir como

$$Y \text{ es } S \text{ si } X \text{ es } R$$

Según Klir [68], en la aritmética borrosa estándar no se tienen en cuenta las restricciones, que representan información adicional, lo que origina en general una deficiencia de información a la salida. En [68] propone una aritmética borrosa con restricciones de requisitos.

3.7.2 Restricciones generalizadas

Las restricciones vistas anteriormente se conocen como *restricciones básicas*. La riqueza de los lenguajes naturales, por razones de representación de significado, necesita una variedad mucho más amplia de restricciones, de las que las *básicas* son una clase especial.

Una restricción generalizada [167, 171] se representa por

$$X \text{ esr } R$$

donde *esr* define el modo en que R restringe a X. Más concretamente, el papel de R en relación con X viene definido por el valor de la variable discreta r.

Las principales restricciones son las siguientes[3] [171]:

$r :=$	restricción de igualdad; $X = R$
$r :$(en blanco)	restricción disyuntiva (posibilística); X es R.
	R es la distribución de posibilidad de X.
$r : v$	restricción verística; X es$v$ R
	R es la distribución de vericidad de X
$r : p$	restricción probabilística; X es$p$ R.
	R es la distribución de probabilidad de X
$r : pv$	restricción de valor de probabilidad; X espv R.
	X es la probabilidad de un suceso borroso y R es su valor.
$r : rs$	restricción de conjunto aleatorio; X es rs R.
	R es la distribución de probabilidad de R evaluada
	en un conjunto borroso
$r : fg$	restricción de grafo borroso; X esfg R.
	X es una función y R es su grafo borroso.
$r : u$	restricción de habitualidad; X es$u$ R.
	Habitualmente (X es R).

Restricción disyuntiva: Puede ser representada por un espacio en blanco o por la variable d. La expresión a la que da lugar es de la forma

$$X \text{ es } R \tag{3.10}$$

en la que R es una relación borrosa que restringe a X jugando el papel de distribución de posibilidad de X. Si X toma valores en un universo de discurso $U = \{u\}$, entonces

$$\text{Poss}\{X = u\} = \mu_R(u) \tag{3.11}$$

[3]Por mantener la unidad de notación con la usada en la bibliografía, los valores de r seguirán la notación inglesa.

donde μ_R es la función de pertenencia de R. Por ejemplo en la proposición

Eva es joven

donde *joven* es un conjunto borroso con función de pertenencia μ_{joven}, la distribución de posibilidad de X viene dada a través de la expresión $\text{Poss}\{X = u\} = \mu_{joven}(u)$, donde u es el valor numérico de la edad y $\text{Poss}\{X = u\}$ es la posibilidad de que la edad de Eva sea u.

Restricción verística: Se representa con la variable v, y da lugar a una expresión de la forma

$$X \text{ es}v \ R \tag{3.12}$$

Esta restricción significa que la vericidad de la proposición $\{X = u\}$ es igual al valor de la distribución de vericidad R en u. Por ejemplo, en la proposición "Alan es medio alemán, un cuarto francés y un cuarto italiano", la vericidad de la proposición "Alan es alemán" es 0.5. (Nótese que los números 0.5 y 0.25 no son probabilidades).

Restricción probabilística: Se representa con la variable p, y da lugar a una expresión de la forma

$$X \text{ es}p \ R \tag{3.13}$$

donde R es la distribución de probabilidad de X. Por ejemplo

$$X \text{ es}p \ N(m, \sigma^2)$$

significa que X tiene una distribución normal con media m y varianza σ^2.

Restricción de valor de probabilidad: La proposición

p: Es probable que Pedro sea joven

se puede expresar como una restricción con valor de probabilidad:

$$\text{Prob}\{\text{Edad}(\textit{Pedro}) \text{ es } \textit{joven}\} \text{ es } \textit{probable}$$

En esta expresión la variable restringida es X: $\text{Prob}\{\text{Edad}(\textit{Pedro}) \text{ es } \textit{joven}\}$ y la restricción

$$X \text{ es } \textit{probable}$$

es una restricción probabilística en la que *probable* es una probabilidad borrosa, definida mediante una función de pertenencia.

Restricción de conjunto aleatorio: X es una variable aleatoria que toma valores en un conjunto borroso. Suponiendo que los valores de X son los conjuntos borrosos $\{A_i, i = 1, \cdots, n\}$ con probabilidades $p1, \cdots, p_n$ respectivamente, la restricción de conjunto aleatorio sobre X se expresa simbólicamente como

$$X \text{ es}rs \ (p_1 \setminus A_1 + \cdots + p_n \setminus A_n) \tag{3.14}$$

Nótese que una restricción de conjunto aleatorio puede verse como una combinación de una restricción probabilística

$$X \text{ es}p\ (p_1 \setminus u_1 + \cdots + p_n \setminus u_n),\ u_i \in U$$

y otra posibilística

$$(X, Y) \text{ es } R$$

donde R es una relación borrosa definida sobre $U \times V$, con función de pertenencia $\mu_R : U \times V \to [0, 1]$. Si A_i es una sección de R (definida como $\mu_{A_i}(v) = \mu_R(u_i, V)$ [168]) la restricción sobre Y es una restricción de conjunto aleatorio expresada como

$$Y \text{ es}rs\ (p_1 \setminus A_1 + \cdots + p_n \setminus A_n).$$

De forma más esquemática puede representarse

$$\frac{\begin{array}{l} X \text{ es}p\ P \\ (X, Y) \text{ es } Q \end{array}}{Y \text{ es}rs\ R}$$

donde Q es una restricción de posibilidad conjunta de X e Y, y R es un conjunto aleatorio.

Restricción de grafo borroso: Estas restricciones son de la forma

$$X \text{ es}fg\ \ R \tag{3.15}$$

donde X es una función f y R es un grafo borroso que hace el papel de distribución de posibilidad de X. Más específicamente, si $f : U \times V \to [0, 1]$ y A_i, $i = 1, \cdots n$, son respectivamente gránulos borrosos en U y V (ver Fig. 3.8), entonces el grafo borroso de f es la disyunción de los productos cartesianos (gránulos) $U_i \times V_j$, expresados como

$$f^* = \sum_{i=1,j=1}^{m,n} U_i \times V_j \tag{3.16}$$

donde el símbolo $\sum$ se debe interpretar como la unión, en lugar de como la suma algebraica, y U_i y V_j toman valores en los conjuntos $\{A_1, \cdots, A_m\}$ y $\{B_1, \cdots, B_m\}$ respectivamente.

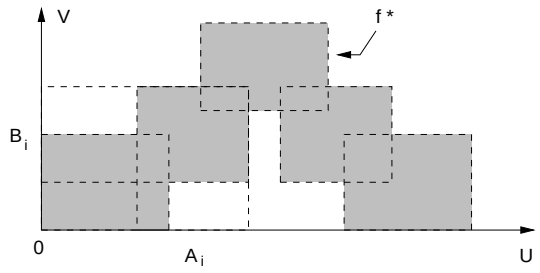


Figura 3.8 Restricción de grafo borroso.

Restricción de habitualidad: es un caso especial de la restricción de valor de probabilidad. Es de la forma

$$X \text{ es}u \ A \tag{3.17}$$

y debe interpretarse como una abreviatura de

$$\text{habitualmente } (X \text{ es } A) \tag{3.18}$$

que también puede interpretarse como

$$\text{Prob}\{X \text{ es } A\} \text{ es } \textit{habitualmente} \tag{3.19}$$

con *habitualmente* haciendo el papel de una probabilidad borrosa con valor cercano a 1. En este sentido, A es un valor *habitual* de X.

En la exposición anterior se ha centrado la atención en las restricciones generalizadas incondicionales. Si generalizamos aún más, una restricción generalizada puede ser también condicional, en cuyo caso se expresa de forma genérica como una regla *si-entonces*

$$\text{Si } X \text{ es}r \ R \text{ entonces } Y \text{ es}s \ S$$

o de manera equivalente

$$Y \text{ es}s \ S \text{ si } X \text{ es}r \ R.$$

Una restricción generalizada puede tener también la forma de una *excepción cualificada*

$$X \text{ es}r \ R \text{ excepto si } Y \text{ es}s \ S$$

3.7.3 Propagación de restricciones borrosas y reglas de inferencia

Las reglas que gobiernan la propagación de las restricciones borrosas son las mismas reglas que para la inferencia borrosa, añadiendo aquellas que gobiernan la modificación de las restricciones.

Suponiendo A y B relaciones borrosas, y definiendo la disyunción y la conjunción a través de t-normas y t-conormas respectivamente, las reglas serían las siguientes [167, 170]:

Reglas directas

1. *Regla Conjuntiva 1:*

$$\frac{\begin{array}{c} X \text{ es } A \\ X \text{ es } B \end{array}}{X \text{ es } A \cap B}$$

2. *Regla Conjuntiva 2:* $(X \in U,\, Y \in V,\, A \subset U,\, B \subset V)$

$$\frac{\begin{array}{c} X \text{ es } A \\ Y \text{ es } B \end{array}}{(X, Y) \text{ es } A \times B}$$

3. *Regla Disyuntiva 1:*

$$\frac{\begin{array}{c} X \text{ es } A \\ \text{ó} \\ X \text{ es } B \end{array}}{X \text{ es } A \cup B}$$

4. *Regla Disyuntiva 2:* $(X \in U,\, Y \in V,\, A \subset U,\, B \subset V)$

$$\frac{\begin{array}{c} X \text{ es } A \\ \text{ó} \\ Y \text{ es } B \end{array}}{(X, Y) \text{ es } A \times V \cup U \times B}$$

   donde $A \times V$ y $U \times B$ son extensiones cilíndricas de A y B respectivamente.

5. *Regla Proyectiva:*

$$\frac{(X, Y) \text{ es } A}{Y \text{ es } \mathrm{proj}_V A}$$

   donde $\mathrm{proj}_V A = \sup_u A$.

6. *Regla Suprayectiva:* $(X \in U,\, Y \in V,\, A \subset U)$

$$\frac{X \text{ es } A}{(X, Y) \text{ es } A \times V}$$

Reglas derivadas

1. *Regla Composicional:*

$$\frac{\begin{array}{c} X \text{ es } A \\ (X, Y) \text{ es } B \end{array}}{Y \text{ es } A \circ B}$$

donde $A \circ B$ denota la composición de A y B.

2. *Principio de Extensión (Regla de Transformación):*

$$\frac{X \text{ es } A}{f(X) \text{ es } f(A)}$$

donde $f : U \to V$ y $f(A)$ se definen por

$$\mu_{f(A)}(v) = \sup_{u|v=f(u)} \mu_A(u)$$

3. *Regla de Transformación Inversa:*

$$\frac{f(X) \text{ es } A}{X \text{ es } f^{-1}(A)}$$

donde $\mu_{f^{-1}(A)}(u) = \mu_A(f(u))$.

4. *Modus Ponens Generalizado:*

$$\frac{\begin{array}{l} X \text{ es } A \\ \text{si } S \text{ es } B \text{ entonces } Y \text{ es } C \end{array}}{Y \text{ es } A \circ ((\neg B) \oplus C)}$$

donde la suma acotada $(\neg B) \oplus C$ representa la definición de implicación de Lukasiewicz.

5. *Principio de Extensión Generalizado:*

$$\frac{f(X) \text{ es } A}{q(X) \text{ es } q(f^{-1}(A))}$$

donde

$$\mu_q(v) = \sup_{u|v=f(u)} \mu_A(g(u))$$

6. *Regla silogística:*

$$\frac{\begin{array}{l} Q_1 A\text{'s son } B's \\ Q_2(AyB)\text{'s son } C\text{'s} \end{array}}{(Q_1 \otimes Q_2)A\text{'s son } (ByC)\text{'s}}$$

donde Q_1 y Q_2 son cuantificadores borrosos, A, B, y C son relaciones borrosas y $Q_1 \otimes Q_2$ es el producto de Q_1 y Q_2 en aritmética borrosa.

7. *Reglas de Modificación de Restricciones:*

$$X \text{ es } mA \to X \text{ es } f(A)$$

donde m es un modificador tal como *no*, *muy*, *más o menos*, y $f(A)$ define el modo en que m modifica a A. Por ejemplo

si $m = no$ entonces $f(A) = A'$ (complemento)
si $m = muy$ entonces $f(A) =^2 A$ (cuadrado izquierdo)

donde $\mu_{2_A}(u) = (\mu_A(u))^2$.

8. *Regla de Cualificación de Probabilidad:*

$$(X \text{ es } A) \text{ es } \Lambda \rightarrow P \text{ es } \Lambda$$

donde X es una variable aleatoria que toma valores en U con densidad de probabilidad $p(u)$, Λ es una probabilidad lingüística expresada en palabras tales como *probable*, *no muy probable*, etc., y P es la probabilidad del suceso borroso "X es A", expresada como

$$P = \int_U \mu_A(u)p(u)du$$

Reglas en función de restricciones generalizadas

Las reglas anteriores han sido expresadas usando restricciones de la forma X es A, dado que es la forma habitual de encontrarlas. Sin embargo, pueden ser reformuladas usando restricciones generalizadas [171] del siguiente modo:

1. *Regla Conjuntiva 1:*

$$\frac{\begin{array}{c} X \text{ es}r\ R \\ X \text{ es}s\ S \end{array}}{X \text{ est } T}$$

El uso de diferentes símbolos (r, s, t) para indicar las relaciones se debe a que estas no tienen por qué ser del mismo tipo.

2. *Regla Conjuntiva 2:*

$$\frac{\begin{array}{c} X \text{ es}r\ R \\ Y \text{ es}s\ S \end{array}}{(X, Y) \text{ est } T}$$

3. *Regla Disyuntiva 1:*

$$\frac{\begin{array}{c} X \text{ es}r\ R \\ \text{ó} \\ X \text{ es}s\ S \end{array}}{X \text{ est } T}$$

4. *Regla Disyuntiva 2:*

$$\frac{\begin{array}{c} X \text{ es}r\ R \\ Y \text{ es}s\ S \end{array}}{(X, Y) \text{ est } T}$$

5. *Regla Proyectiva:*

$$\frac{(X,Y) \text{ es}r\ R}{Y \text{ es}s\ S}$$

6. *Regla Surjectiva:*

$$\frac{X \text{ es}r\ R}{(X,Y) \text{ es}s\ S}$$

7. *Regla Inversiva*

$$\frac{f(X) \text{ es}r\ R}{X \text{ es}s\ S}$$

donde $f(X)$ es una función de X.

8. *Regla Composicional:*

$$\frac{\begin{array}{c}X \text{ es}r\ R \\ (X,Y) \text{ es}s\ S\end{array}}{(Y \text{ est}\ T}$$

9. *Principio de Extensión Generalizado:*

$$\frac{f(X) \text{ es}r\ R}{g(X) \text{ es}s\ S}$$

donde f y g son funciones dadas.

A pesar de que esta formulación es mucho más general, su utilidad práctica es menor.

3.8 TEORÍA DE LA POSIBILIDAD

En [162] Zadeh plantea una teoría de la posibilidad que tiene como base a los conjuntos borrosos. Es, en esencia, una metodología para representar el conocimiento vago o incompleto en términos de distribuciones de posibilidad, infiriendo los valores de las variables de interés por medio de la aplicación de las reglas de la teoría de la posibilidad. Esta teoría será ampliada principalmente por el propio Zadeh [171], por las aportaciones de Dubois y Prade [30, 32, 35] y por los trabajos de Yager [40, 145, 147, 152, 153].

3.8.1 El principio del mínimo específico

Sea U un conjunto que representa el rango de variabilidad de una variable x. Dicha variable no representa un objeto específico, sino un valor desconocido que toma cierto

atributo, como por ejemplo la edad de un coche, la altura de un edificio, etc. Una densidad de distribución π_x en U es una transformación de U en el intervalo $[0, 1]$ sujeto a la variable x. La función π_x representa una restricción flexible de los valores de x de la siguiente forma:

$$\begin{array}{ll} \pi_x(u) = 0 & \text{significa que } x = u \text{ es imposible,} \\ \pi_x(u) = 1 & \text{significa que } x = u \text{ es permitido.} \end{array}$$

La flexibilidad en la restricción se modela permitiendo a $\pi_x(u)$ variar en el intervalo $[0, 1]$. La cantidad $\pi_x(u)$ representa entonces el grado de posibilidad de la asignación $x = u$, siendo unos valores de u más posibles que otros. Si se trabaja en todo el rango de U, existirá un valor para el que $\pi_x(u) = 1$ (normalización).

Las distribuciones de posibilidad no son otra cosa que funciones de características de los conjuntos generalizadas, que se interpretan como restricciones sobre los valores de la variable. De este modo, si lo único que se conoce de la variable x es que "x cae en A", donde $A \subseteq U$, la distribución de posibilidad tal y como la define Zadeh [162] será

$$\pi_x(u) = \mu_A(u), \ \ \forall u \in U \tag{3.20}$$

donde μ_A es la función característica del conjunto A de los posibles valores de x. La ecuación 3.20 no significa que las funciones de pertenencia y las distribuciones de probabilidad sean lo mismo. Significa que "dado que el único conocimiento disponible es que x cae en A, el grado de posibilidad de $x = u$ lo evaluamos mediante el grado de pertenencia $\mu_A(u)$" [32].

Cuando se encuentran disponibles diversas distribuciones de probabilidad para x (de n fuentes) π_x^1, π_x^2, ..., π_x^n, el principio del mínimo específico lleva a asumir que

$$\pi_x = \min_{i=1,\cdots,n} \pi_x^i \tag{3.21}$$

Si se supone que existen dos variables, x e Y, con distribuciones de posibilidad π_x y π_y respectivamente, el principio del mínimo específico lleva a la definición de la distribución de posibilidad conjunta $\pi_{x,y}$ como

$$\pi_{x,y} = \min(\pi_x, \pi_y) \tag{3.22}$$

3.8.2 El mecanismo de combinación/proyección

Dado un estado de conocimiento descrito en términos de distribución de posibilidad conjunta por $\pi_{x_1,x_2,\cdots,x_n}$, un problema básico consiste en realizar deducciones de la forma $(x_1, \cdots, x_n) \in B$, siendo B un subconjunto *ordinario*. Debido a la imprecisión del conocimiento disponible, $(x_1, \cdots, x_n) \in B$ puede ser cierto, falso o parcialmente desconocido. Este estado de incertidumbre se puede evaluar usando diferentes medidas:

Medida de posibilidad: $\Pi(B)$ definida como

$$\Pi(B) = \sup_{(u_1,\cdots,u_n)\notin B} \pi_{x_1,\cdots,x_n}(u_1,\cdots,u_n) \tag{3.23}$$

Evalúa hasta qué punto B es consistente con $\pi_{x_1,\cdots,x_n}$.

Medida de certeza: (también llamada de necesidad), $N(B)$ se define

$$N(B) = 1 - \Pi(\overline{B}) = \inf_{(u_1,\cdots,u_n)\notin B} [1 - \pi_{x_1,\cdots,x_n}(u_1,\cdots,u_n)] \tag{3.24}$$

Evalúa hasta que punto el conjunto borroso con función de pertenencia $\pi_{x_1,\cdots,x_n}$ está contenido en B

Las medidas de posibilidad son funciones de conjuntos que cumplen los siguientes axiomas [32]:

$$\forall I, \quad \Pi\left(\cup_{i\in I} B_i\right) = \sup_{i\in I} \Pi(B_i) \tag{3.25}$$

$$\forall I, \quad N\left(\cup_{i\in I} B_i\right) = \inf_{i\in I} N(B_i) \tag{3.26}$$

Si $x_1,\cdots,x_n$ son no-interactivas (es decir, $\pi_{x_1,\cdots,x_n} = \min_{i=1,\cdots,n} \pi_{x_1}$) y si para todo $i = 1,\cdots,n$ B_i es un subconjunto del universo U_i se tiene que

$$\Pi(B_1 \times \cdots \times B_n) \quad = \min_{i=1,\cdots,n} \Pi(B_i) \quad = \min_{i=1,\cdots,n} \Pi_i(B_i) \tag{3.27}$$

$$N(B_1 + \cdots + B_n) \quad = \max_{i=1,\cdots,n} N(B_i) \quad = \max_{i=1,\cdots,n} N_i(B_i) \tag{3.28}$$

De un modo más general, incluso cuando las x_i están relacionadas, la distribución de posibilidad de π_{x_i} se puede calcular como la *proyección* de $\pi_{x_1,\cdots,x_n}$ sobre U_i, definida como

$$\begin{aligned} \pi_{x_i}(u_i) &= \Pi(U_1 \times \cdots \times U_{i-1} \times \{u_i\} \times U_{i+1} \times \cdots \times U_n) \\ &= \sup_{u_1,\cdots,u_{i-1},u_{i+1},\cdots,u_n} \pi_{x_1,\cdots,x_n}(u_1,\cdots,u_n) \end{aligned} \tag{3.29}$$

y generalmente $\pi_{x_1,\cdots,x_n} \leq \min(\pi_{x_1},\cdots,\pi_{x_n})$. A partir de esto ya se puede definir el principio de proyección combinación.

Principio de combinación/proyección: Dado un conjunto de m distribuciones de posibilidad $\pi_1,\cdots,\pi_m$, donde π_j es una restricción borrosa de un conjunto de variables $\{x_i | i \in I_j\}$, sea x_k una variable cuyo valor queremos averiguar, con $k \in K = \cup_{j=1,\cdots,m} I_j$. La distribución de posibilidad π_k se define por medio de dos pasos básicos [161, 164]:

1. *Combinación:* en virtud del principio del mínimo específico, la distribución de posibilidad total será
$$\pi = \min_{j=1\cdots,m} \pi_j$$
2. *Proyección:* el cálculo de la proyección de π en el dominio U_k de x_k se hace usando la eq.(3.29).

3.9 CONCLUSIONES

El problema del cálculo aproximado usando palabras ha dado lugar a muchas y diversas aproximaciones, como se ha podido ver en el presente capítulo. Mucha de la teoría aquí expuesta es reciente, por lo que todavía adolece de una falta de unidad. Algunos conceptos (como por ejemplo el de *grafo borroso*) parecen simples artificios matemáticos que poco aportan a la teoría general. Otros, como el concepto de gránulo, aportan un nuevo punto de vista respecto al procesado de la información, pero todavía no existen muchas aplicaciones basadas en estos conceptos.

El propósito de esta tesis no es simplemente contribuir a incrementar el poso teórico de CWW, sino encontrar un método que permita la implementación real de los conceptos aquí citados. Por ello, en lo sucesivo, se hará uso de las variables lingüísticas, las lógicas borrosas, los gránulos y parte de la teoría de las restricciones.

MIGUEL DE CERVANTES BIBLIOTECA VIRTUAL

4

NUEVO MARCO DE TRABAJO MATRICIAL

BIBLIOTECA VIRTUAL MIGUEL DE CERVANTES

4.1 REPRESENTACIÓN GRANULAR DE LA INFORMACIÓN

4.1.1 Discusión previa

Diferentes autores plantean la posibilidad de representar la información lingüística en modo de gránulos borrosos (sec. 3.6). Sin embargo, a la hora de definir dichos gránulos se hace uso de las conocidas funciones de pertenencia, ya sean triangulares, trapezoidales, Gaussianas... (sec. 2.1.2). Por lo tanto, un concepto que a priori es lingüístico, es adaptado a a unos ejes matemáticos y a unas funciones de pertenencia para poder trabajar con él.

La mayoría de los autores ligan la existencia de los conjuntos borrosos a un *universo de discurso* numérico [66], sobre el que los conjuntos borrosos realizan una partición (también borrosa). En ocasiones esta partición se realiza sobre un eje ficticio que nada tiene que ver con la entidad representada.

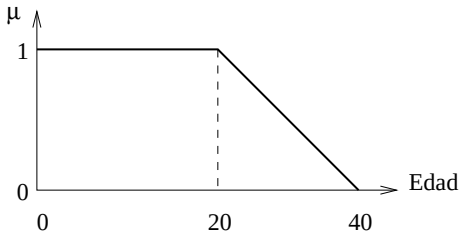


Figura 4.1 Definición de la función de pertenencia del término *Joven*

Trabajando con conceptos tales como *Edad*, parece claro que la asignación de valores es casi inmediata y necesaria. Puede definirse la función de pertenencia del término *Joven* mediante una función PTS (sección 2.1.2) como la que se muestra en la Fig. 4.1. A los valores comprendidos entre 0 y 20 se les asigna un valor de pertenencia máxima. Este valor de pertenencia cae linealmente hasta los 40 años. Como se puede ver, existe una relación directa entre la información numérica (edad) y la información lingüística (*Joven*).

En otras aplicaciones sin embargo, puede hacerse uso de términos que no tengan asociada una dimensión numérica. Es el caso de la aplicación de seguimiento de vehículos basada en texto que se describe en [7]. Una de las entradas al sistema es la variable lingüística (VL) *Vehículo*, que toma valores como *coche*, *furgoneta*, *camión*. Puede intuirse que estos términos van a solaparse de alguna manera, pero no parece tan claro que lleven asociada una dimensión medible.

Surge por lo tanto la necesidad de definir los términos de una variable lingüística mediante las relaciones entre dichos términos (como por ejemplo, el nivel de *solapamiento* semántico), sin que sea necesario crear un eje numérico ficticio.

En la secciones posteriores se partirá de esta idea para desarrollar una nueva forma de definir aquellos conjuntos borrosos que no necesiten de una variable base. Esta definición se denominará *gránulo borroso* por las similitudes con el concepto de gránulo planteado en la literatura de CWW (sec. 3.6).

4.1.2 Definición de conjuntos borrosos mediante el grado de solapamiento

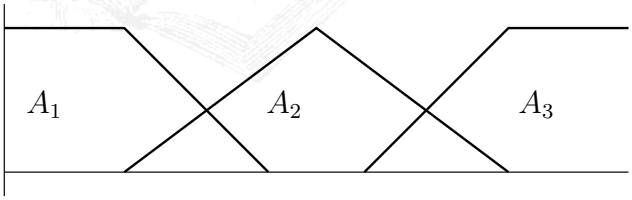


Figura 4.2 Definición clásica de 3 conjuntos borrosos

Supóngase que en una determinada aplicación las entradas a un motor de inferencia borrosa son términos lingüísticos expresados en *lenguaje natural*. En la aplicación de seguimiento de vehículos citada anteriormente, la entrada es el nombre de los vehículos observados. La técnica habitual para modelar estas entradas como conjuntos borrosos consiste en definir un universo de discurso U sobre el que se definen, a su vez, una serie de conjuntos A_i que dividen dicho espacio. En lo que sigue, por simplicidad, nos referiremos a un conjunto borroso y a su función de pertenencia de la misma manera, usando la notación del conjunto. En la figura 4.2 se muestra un ejemplo con tres conjuntos. Se puede observar:

1. Es necesario adoptar una forma para los conjuntos. En la figura se ha elegido que éstos sean trapezoidales o triangulares.
2. Es necesario realizar una partición del espacio U.
3. Es necesario proporcionar valores al eje de abscisas.
4. Habrá que definir un método de activación de reglas, de modo que cuando la entrada sea A_i, el conjunto A_j se active $A_i \circ A_j$ [4, 66].

Si la entrada a un sistema es el conjunto borroso X, será necesario recurrir a un método de activación de reglas (ver sección 2.2.4) para calcular $A_i(X)$, siendo $A_i(X)$ el nivel de activación del antecedente A_i cuando la entrada es X. Independientemente del método elegido, esta activación puede escribirse $A_i \circ X$.

Si la entrada X toma valores únicamente dentro de A_i (esta suposición se estudiará con más profundidad en la sección 4.2.2), los posibles valores de entrada quedan reducidos a

$$A_i \circ A_j \qquad i = 1, ..., N; \qquad j = 1, ..., N \tag{4.1}$$

o a una combinación de dichos valores.

A partir de este razonamiento, se puede definir una *matriz de activación de conjuntos*, o *matriz de intersecciones* del siguiente modo:

$$\mathbf{R_A} = \begin{pmatrix} A_1 \cap_{gs} A_1 & \cdots & A_M \cap_{gs} A_1 \\ A_1 \cap_{gs} A_2 & \cdots & A_M \cap_{gs} A_2 \\ \vdots & \cdots & \vdots \\ A_1 \cap_{gs} A_M & \cdots & A_M \cap_{gs} A_M \end{pmatrix} = \begin{pmatrix} A_1 \circ A_1 & \cdots & A_M \circ A_1 \\ A_1 \circ A_2 & \cdots & A_M \circ A_2 \\ \vdots & \cdots & \vdots \\ A_1 \circ A_M & \cdots & A_M \circ A_M \end{pmatrix} \tag{4.2}$$

siendo $\cap_{gs}$ el *grado de solapamiento* entre conjuntos. No debe confundirse con la intersección entre conjuntos borrosos, generalmente llevada a cabo por un operador t-norma. El grado de solapamiento, en este caso, se corresponde con el operador $\circ$, que puede llevarse a cabo mediante un operador t-norma y un operador t-conorma sucesivos.

Nótese que en la matriz de la ecuación (4.2), generalmente se cumple que

$$\begin{cases} A_i \cap_{gs} A_j = A_j \cap_{gs} A_i & i \neq j \\ A_i \cap_{gs} A_j = 1 & i = j \end{cases}$$

Con esta matriz es suficiente para definir el comportamiento de las entradas al sistema, siempre y cuando las entradas sean los A_i o una combinación de ellos.

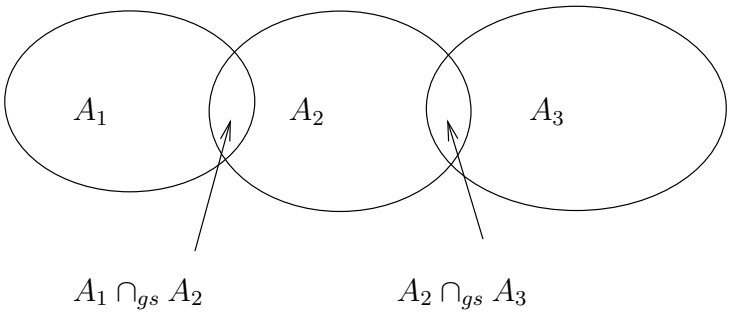


Figura 4.3 Definición de 3 conjuntos mediante intersecciones.

Por lo tanto, se plantea la posibilidad de definir los conjuntos borrosos pertenecientes a una variable lingüística, simplemente mediante sus grados de solapamiento. Como

ejemplo, en la Fig. 4.3 se representan los mismos 3 conjuntos borrosos que en la Fig. 4.2. Mediante la nueva representación no es necesario definir una partición en U, ni dar valores a ordenadas ni a abscisas, ni definir un método de activación, ni siquiera definir la forma de los conjuntos o la función a utilizar. Basta con definir los conjuntos de forma abstracta con un nombre, A_i, y definir el valor del solapamiento entre ellos. En este caso, la matriz de intersecciones sería:

$$\begin{aligned}\mathbf{R_A} &= \begin{pmatrix} 1 & A_2 \cap_{gs} A_1 & 0 \\ A_1 \cap_{gs} A_2 & 1 & A_3 \cap_{gs} A_2 \\ 0 & A_2 \cap_{gs} A_3 & 1 \end{pmatrix} \\ &= \begin{pmatrix} 1 & A_2 \circ A_1 & 0 \\ A_1 \circ A_2 & 1 & A_3 \circ A_2 \\ 0 & A_2 \circ A_3 & 1 \end{pmatrix}\end{aligned}$$

Un caso más complejo se recoge en la figura 4.4. Aquí, las intersecciones no son sólo con los conjuntos adyacentes (siguiendo la idea de los conjuntos definidos sobre una recta), sino que existen múltiples intersecciones, lo que no supone ningún problema a la hora de definir la matriz $\mathbf{R_A}$.

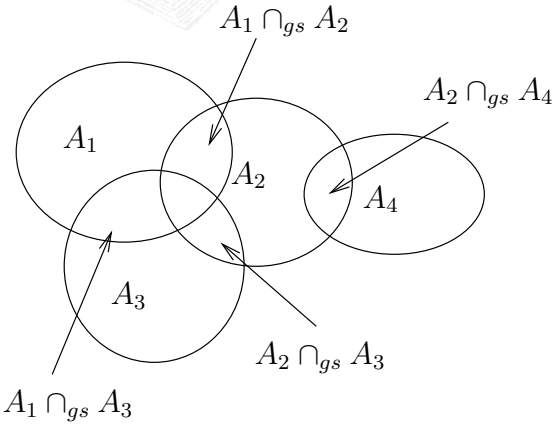


Figura 4.4 Múltiples intersecciones en la definición de conjuntos borrosos.

Generalizando, puede decirse que esta nueva manera de definir los conjuntos borrosos de una variable lingüística es una extensión natural del concepto de *gránulo borroso* [166]. Los gránulos son conceptos que por sí mismos no tienen representación numérica, sino que la tienen cuando entran en contacto con los demás.

Respecto a la matriz $\mathbf{R_A}$, ésta puede interpretarse como un caso concreto de una relación borrosa $R(X, X)$ puesta en forma matricial $\mathbf{R}$, en la que la relación que se establece es de compatibilidad entre conjuntos borrosos [66]. Los distintos valores de X van a ser los diversos conjuntos que forman la variable lingüística.

Ejemplo de aplicación

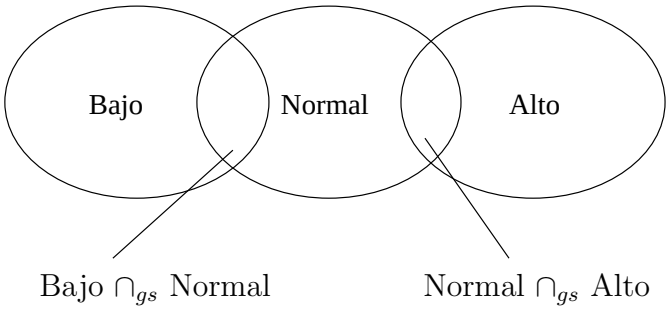


Figura 4.5 Definición mediante gránulos de la variable lingüística *Altura*

En el siguiente ejemplo se considerará la variable lingüística *altura*. Suponiendo que las entradas al sistema siempre van a ser de tipo lingüístico y no numérico, se define mediante gránulos borrosos tal y como se muestra en la Fig. 4.5. La matriz de intersecciones se definirá:

$$\mathbf{R_A} = \begin{pmatrix} Bajo \circ Bajo & Bajo \circ Normal & Bajo \circ Alto \\ Normal \circ Bajo & Normal \circ Normal & Normal \circ Alto \\ Alto \circ Bajo & Alto \circ Normal & Alto \circ Alto \end{pmatrix} = \begin{pmatrix} 1 & \vartheta_1 & 0 \\ \vartheta_1 & 1 & \vartheta_2 \\ 0 & \vartheta_2 & 1 \end{pmatrix}$$

Basta por lo tanto con definir dos cantidades ϑ_1 y ϑ_2 (que se corresponden con el grado de solapamiento entre conceptos) para tener caracterizado el sistema ($\vartheta_1, \vartheta_2 \in [0,1]$). Si la base de reglas es de la forma:

Si X es Bajo	entonces Y es B_1
Si X es Normal	entonces Y es B_2
Si X es Alto	entonces Y es B_3

el conjunto de salida Y usando una t-norma ($\bigcap$) y una t-conorma ($\bigcup$) genéricas será (ecuación 2.56)

$$Y = \Big((Bajo \circ X) \bigcap B_1\Big) \bigcup \Big((Normal \circ X) \bigcap B_2\Big) \bigcup \Big((Alto \circ X) \bigcap B_3\Big)$$

si la entrada es $X = Normal$ esto equivale[1] a

$$Y = \Big(\vartheta_1 \bigcap B_1\Big) \bigcup \Big(1 \bigcap B_2\Big) \bigcup \Big(\vartheta_2 \bigcap B_3\Big)$$

[1] Como se verá a continuación esto es cierto si se cumple que las normas usadas son distributivas

4.1.3 Matrices de activación para conjuntos definidos

La formulación empleada en el caso anterior puede ser extrapolada a conjuntos definidos de forma clásica, es decir sobre un eje, asignando valores a sus puntos (Fig. 4.2). En este caso, en lugar de definir directamente los valores de la matriz $\mathbf{R_A}$, ésta puede calcularse a partir de los conjuntos, tal como se muestra en la ecuación (4.2).

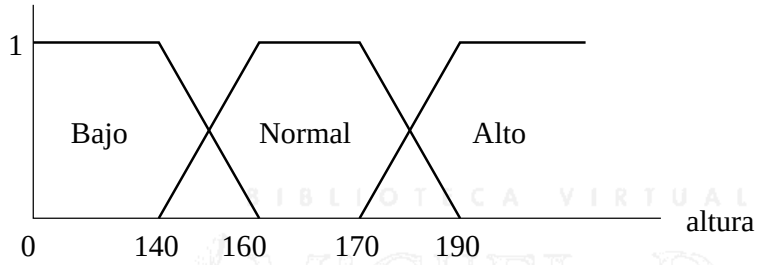


Figura 4.6 Definición de la variable lingüística *Altura* mediante conjuntos trapezoidales

Si se toma el ejemplo del apartado anterior, los conjuntos de la variable altura podrían definirse usando funciones trapezoidales, tal como se muestra en la figura 4.6. De este modo, los conjuntos quedan definidos por la 4-tupla *(a,b,c,d)*, donde b y c indican el intervalo en el que el valor de pertenencia es 1, y a y d indican los límites de la función:

$$\begin{aligned} \text{Bajo} &= (0,0,140,160) \\ \text{Normal} &= (140,160,170,190) \\ \text{Alto} &= (170,190,250,250) \end{aligned}$$

La matriz $\mathbf{R_A}$ quedaría entonces de la siguiente forma:

$$\mathbf{R_A} = \begin{pmatrix} Bajo \circ Bajo & Bajo \circ Normal & Bajo \circ Alto \\ Normal \circ Bajo & Normal \circ Normal & Normal \circ Alto \\ Alto \circ Bajo & Alto \circ Normal & Alto \circ Alto \end{pmatrix} \tag{4.3}$$

Será necesario definir un método para calcular $A_i \circ A_j$, como *max-min* o *suma-producto* (sección 2.2.4).

4.1.4 Modificadores lingüísticos

Como condición inicial se ha asumido que la entrada al sistema borroso va a ser una combinación de los valores de la variable lingüística de entrada. Aunque esta limitación no es importante en un número amplio de aplicaciones, sin duda alguna restringe su uso en aplicaciones más generales. Mediante el uso de modificadores lingüísticos (sección 2.1.9) el vocabulario de entrada puede ser extendido fácilmente.

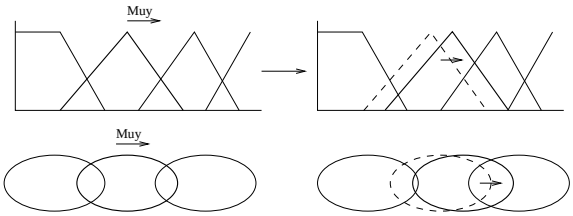


Figura 4.7 El modificador lingüístico *Muy* aplicado sobre un conjunto borroso (arriba) y sobre un gránulo borroso (abajo)

Supóngase que se quiere hacer uso del modificador *Muy*. El efecto de este operador puede ser el de desplazar algunos conjuntos hacia la derecha (y otros a la izquierda), tal y como se muestra en la Fig. 4.7. Cuando se aplica un modificador sobre conjuntos definidos, éste modifica la forma y/o la localización de dicho conjunto (sec.2.1.9). El efecto final es la modificación de la interacción de este conjunto con el resto de elementos de la variable lingüística. Este efecto puede ser capturado mediante la definición de una matriz de incrementos, a la que llamaremos $\mathbf{\Delta}_H$. Esta matriz puede definirse como

$$\mathbf{\Delta}_H = \begin{pmatrix} \delta_{11} & \delta_{12} & \cdots & \delta_{1M} \\ \delta_{21} & \delta_{22} & \cdots & \delta_{2M} \\ \vdots & \vdots & \vdots & \vdots \\ \delta_{M1} & \delta_{M2} & \cdots & \delta_{MM} \end{pmatrix} \tag{4.4}$$

siendo δ_{ij} la alteración introducida por el modificador *Muy* en la interacción original $A_i \cap_{gs} A_j$ de la matriz $\mathbf{R_A}$, la cual puede ser positiva o negativa.

Como ejemplo, se hará uso del modificador *Muy* sobre el ejemplo de la sección anterior. Supóngase que este modificador afecta sólo a los términos *Bajo* y *Alto*, reduciendo el grado de solapamiento en una cantidad δ_v. La matriz de incrementos para *Muy* es:

$$\mathbf{\Delta}_{H(Muy)} = \begin{pmatrix} 0 & -\delta_v & 0 \\ -\delta_v & 0 & -\delta_v \\ 0 & -\delta_v & 0 \end{pmatrix}$$

Este modificador desplaza el gránulo *Alto* hacia la derecha, y el gránulo *Bajo* a la izquierda, disminuyendo el grado de solapamiento con el gránulo del conjunto *Medio*. La matriz total del sistema será (suponiendo que el grado de solapamiento entre *Alto* y *Muy alto* es la unidad):

$$\mathbf{R_A} + \mathbf{\Delta}_{H(Muy)} = \begin{pmatrix} 1 & \delta_1 - \delta_v & 0 \\ \delta_1 - \delta_v & 1 & \delta_2 - \delta_v \\ 0 & \delta_2 - \delta_v & 1 \end{pmatrix}$$

4.1.5 Generalización de los gránulos

La definición de *gránulo* introducida en este capítulo se basa en el grado de solapamiento existente entre conceptos borrosos. Este grado de solapamiento se ha rela-

cionado con el operador $\circ$ actuando sobre dos conjuntos borrosos. Sin embargo, dentro del campo del razonamiento aproximado, una relación R entre dos conjuntos borrosos no sólo se limita a este operador.

Una definición más amplia del concepto de gránulo conllevaría la creación de una serie de matrices de transición $\mathbf{R}_i$ en las que se codifique la información de las diferentes relaciones que afectan a los conceptos representados por dichos gránulos.

En [27] por ejemplo, los autores definen la "aproximación superior" y la "aproximación inferior" de un conjunto borroso que serán usadas en un nuevo proceso de inferencia. Para su cálculo necesitan de dos operadores, definidos como

$$\text{cons}(A, B) = \max_{u \in U} \min(\mu_A(u), \mu_B(u))$$

y

$$\text{inc}(A, B) = \min_{u \in U} \max(1 - \mu_A(u), \mu_B(u))$$

El primer operador es similar al operador $\circ$ usado para definir los grados de solapamiento entre gránulos. El segundo, en cambio, no puede ser calculado a partir de una representación granular como la planteada en esta tesis. Si se quisiera llevar a cabo algún tipo de inferencia en que se hiciera uso de él, haría falta definir una nueva matriz $\mathbf{R'_A}$ que contuviera estas relaciones.

En lo sucesivo, sin embargo, se trabajará considerando sólo la relación *grado de solapamiento*, dejando la generalización a otras relaciones como una línea futura de trabajo.

4.2 INFERENCIA BORROSA MATRICIAL

En la presente sección se pretende explotar la representación de conjuntos borrosos mediante una matriz de activaciones $\mathbf{R_A}$. Partiendo de estas matrices, y representando los conjuntos de entrada de forma vectorial, va a ser posible sustituir la inferencia borrosa tradicional por una operativa matricial que, asu vez, va a permitir precalcular diversos valores, de tal modo que el número de operaciones a realizar durante la inferencia pueda verse reducido considerablemente. A este tipo de operativa se le ha denominado FITM (*Fast Inference using Transition Matrices*, inferencia rápida usando matrices de transición) [2].

4.2.1 Estado del arte

Es bien sabido que los sistemas borrosos padecen lo que se ha llamado *maldición de la dimensionalidad* [72]: necesitan un gran número de reglas para aproximar funciones y el número de reglas crece exponencialmente con el número de variables de entrada y

salida. Un número mayor de variables puede hacer el modelo más realista, pero el precio a pagar es el alto coste computacional debido a la explosión de las reglas.

El procedimiento típicamente empleado para reducir este coste computacional ha sido la reducción de la base de reglas, eligiendo aquéllas que se consideran *óptimas*. Sin embargo, esto es sólo una parte del problema, dado que en cada paso de la inferencia se han de calcular un gran número de t-normas y t-conormas sobre cada conjunto, y este número será tanto mayor cuanto mayor sea la resolución del universo sobre el que el conjunto está definido. De este modo, en la literatura pueden encontrarse dos tipos de aproximaciones para tratar de reducir el coste computacional de las inferencias: por un lado la que trata de reducir el número de reglas de la base; por otro, la que trata de reducir el número de operaciones en la inferencia.

BIBLIOTECA VIRTUAL

Respecto a los procedimientos de reducción de reglas, la idea es seleccionar sólo aquellas reglas de la base que tengan una alta contribución sobre el conjunto de salida, o bien crear un nuevo conjunto de reglas de cardinal inferior al original, que aproxime la misma función. En [53], por ejemplo, la búsqueda de las reglas óptimas se plantea como un problema de optimización; la solución se obtiene usando algoritmos genéticos. Otra contribución es [156], en la que los autores reducen el número de reglas mediante una descomposición en valores singulares de los consecuentes de las reglas, y calculan combinaciones lineales de las funciones de pertenencia originales para encontrar las nuevas funciones para el nuevo conjunto de datos. En [126] se propone un método alternativo al anterior basado en combinación de reglas. Este método sólo es válido para inferencias SAM o max-min. Finalmente, en [36] se propone una metodología neuronal para la extracción y optimización de las reglas.

Respecto a los algoritmos que se ocupan de reducir el coste computacional, deben mencionarse algunas contribuciones. Comb y Andrews proponen en [21] una nueva configuración de las reglas que simplifica la estructura del problema, ahorrando tiempo de computación. Este trabajo, ha encontrado cierta oposición atendiendo a las respuestas que han aparecido a continuación [92, 23].

En [80] los autores parten del hecho de que el conjunto solución de un sistema borroso basado en ecuaciones relacionales (usando la composición max-min) puede encontrarse por medio de una solución máxima y un conjunto de soluciones mínimas. Estas última solución se ve afectada de explosión combinatoria. El esfuerzo de los autores se centra en plantear el problema de tal modo que el conjunto de soluciones mínimas se reduzca. Proponen un algoritmo eficiente para encontrar dicho conjunto. Por último, en [77] se plantea que la complejidad de un sistema borroso de múltiples entradas y una salida es mucho mayor que la de una colección equivalente de sistemas borrosos de una entrada y una salida, por lo que los autores plantean una forma de combinar las salidas de estos últimos sistemas para encontrar de manera eficiente la salida del primero.

Como hemos indicado, los gránuos borrosos y las matrices de activación prporcionan una vía alternativa para reducir complejidad. En lo que sigue se indica la forma de llevar esto a cabo. Se asumirá una base de reglas completa, es decir, la base contiene una regla

para cada posible combinación de valores de los antecedentes. En caso contrario, la base de reglas debe completarse, bien con un consecuente del tipo "no importa", o bien a partir de los valores implícitos en las otras reglas (ver apéndice C.4).

4.2.2 Definición vectorial de entradas borrosas. Modelo SAM

Un modelo aditivo estándar (sec. 2.3.6) tiene una salida de la forma[2]:

$$\text{Salida} = \sum_j D_{r(j)} \left[\prod_i (CA_i)_j \right] \tag{4.5}$$

con $D_{r(j)}$ la salida borrosa de la regla j-ésima, $r(j)$ una función de indexado para la que $1 \leq r(j) \leq L$ si la variable de salida consiste en L conjuntos borrosos, y $(CA_i)_j$ es el coeficiente del antecedente i-ésimo para la regla j-ésima que representa el grado de consistencia entre el antecedente i-ésimo y el hecho i-ésimo para la regla j-ésima, en otras palabras, cómo se activa el antecedente en función de la entrada al sistema.

Considérese en un sistema SAM una entrada borrosa $X \subset U$, siendo U el universo de discurso. Como condición inicial se tomará la hipótesis de que esta entrada va a ser uno de los conjuntos A_i que forman la variable lingüística o una combinación lineal de éstos.

En aplicaciones reales basadas en CWW (cap. 3) las entradas al sistema son palabras que han sido previamente modeladas usando conjuntos borrosos sobre una variable lingüística. Por lo tanto, en esos casos, el vocabulario de entrada se restringe a los valores de dicha variable. Consecuentemente, una entrada podría definirse

$$X = \sum_{k=1}^{M} \beta_k A_k \tag{4.6}$$

donde A_k son los distintos conjuntos que forman la variable lingüística, y β_j un coeficiente que vale 1 si la entrada es A_j y cero para el resto.

Tal y como indica la ecuación (4.5), la salida de un SAM es una combinación lineal de los conjuntos de salida. Si esta salida es la entrada de otro sistema (Fig. 4.8), esta entrada no será un sólo conjunto borroso, sino una combinación lineal de conjuntos borrosos. En este caso, los coeficientes β_j ya no serán necesariamente binarios.

Si se expresa la entrada usando la ecuación (4.6) el coeficiente $(CA_i)_j$ de la ecuación (4.5) se puede reescribir, asumiendo que la operación $\circ$ representa la composición *suma-producto*,

$$(CA_i)_j = A_{p_i(j)} \circ X = A_{p_i(j)} \circ \left(\sum_k \beta_k A_k \right) = \sum_k \beta_k (A_{p_i(j)} \circ A_k) \tag{4.7}$$

[2]Se va a trabajar con la hipótesis de que todas las reglas van a tener el mismo peso, por lo que $\omega_1 = \omega_2 = \cdots = \omega_M = 1$.

con $p_i(j)$ una función de indexado que proporciona el índice del conjunto borroso usado como antecedente i-ésimo en la regla j-ésima. De este modo, la expresión de la ecuación (4.5) se convierte en

$$\text{Salida} = \sum_j D_{r(j)} \left[\prod_i \sum_k \beta_k (A_{p_i(j)} \circ A_k) \right] \tag{4.8}$$

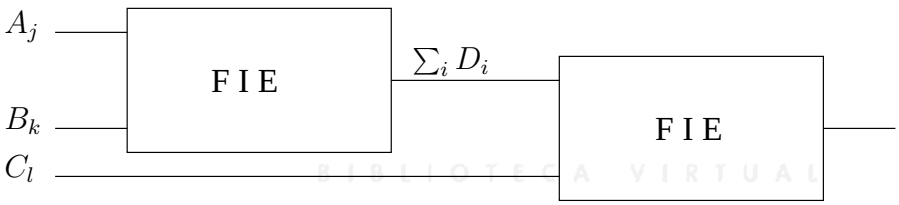


Figura 4.8 Sistema borroso de dos bloques

La ecuación muestra que la operación de composición $\circ$ es independiente de la entrada X, ya que la dependencia de la entrada ahora recae sobre los coeficientes β_k. Consecuentemente, esta operación podrá ser precalculada y almacenada, tal y como ocurría en el caso de los gránulos. Además, se vuelve irrelevante que los conjuntos borrosos tengan funciones de pertenencia continuas o discretas, así como la resolución del universo de discurso. Toda está información se encuentra oculta en la operación de composición.

Por lo tanto, para un SAM puede definirse la entrada al sistema como un vector de coeficientes de la forma

$$X = [\beta_1 \;\; \beta_2 \;\; \cdots \;\; \beta_N]^T \tag{4.9}$$

En el ejemplo de las alturas de la sección 4.1.2, si la entrada es el conjunto *Normal*, el vector correspondiente sería

$$X = [0 \; 1 \; 0]^T$$

Este modo de representar las entradas se puede utilizar tanto para conjuntos definidos de forma granular, como para conjuntos definidos sobre un eje numérico.

4.2.3 SAM con una entrada y una salida

Supóngase un sistema borroso con una entrada X y una salida Z, ambas conjuntos borrosos definidos sobre variables lingüísticas. La VL de entrada consta de M posibles conjuntos borrosos A_k definidos en el universo U y la VL de salida consta de L posibles conjuntos borrosos D_n definidos en el universo W. En caso de trabajar con conjuntos borrosos definidos sobre un eje numérico real, el sistema realizará una transformación $f : U \subset R \mapsto W \subset R$.

La hipótesis de la ecuación (4.6) puede reescribirse en forma vectorial como

$$X = \sum_{k=1}^{M} \beta_k A_k = \left[\begin{array}{cccc} \beta_1 & \beta_2 & \cdots & \beta_M \end{array} \right] \left[\begin{array}{cccc} A_1 & A_2 & \cdots & A_M \end{array} \right]^T = \boldsymbol{\beta}^T \mathbf{A}$$

y la salida, de acuerdo con la ecuación (4.5) se puede escribir como una combinación lineal de los conjuntos $D_{r(j)}$ (con $1 \leq r(j) \leq L$)

$$Z = \sum_{n=1}^{L} \gamma_n D_n = \left[\begin{array}{cccc} \gamma_1 & \gamma_2 & \cdots & \gamma_L \end{array} \right] \left[\begin{array}{cccc} D_1 & D_2 & \cdots & D_L \end{array} \right]^T = \boldsymbol{\gamma}^T \mathbf{D}$$

A partir de la ecuación (4.8) se puede escribir

$$\begin{aligned} \boldsymbol{\gamma}^T \mathbf{D} &= \sum_{j=1}^{M} D_{r(j)} \left[\sum_{k=1}^{M} \beta_k (A_{p(j)} \circ A_k) \right] \\ &= \sum_{j=1}^{M} D_{r(j)} \left[\begin{array}{cccc} \beta_1 & \beta_2 & \cdots & \beta_M \end{array} \right] \left[\begin{array}{c} (A_{p(j)} \circ A_1) \\ (A_{p(j)} \circ A_2) \\ \vdots \\ (A_{p(j)} \circ A_M) \end{array} \right] \end{aligned} \tag{4.10}$$

donde $r(j)$ y $p(j)$ son operaciones de indexado apropiadas que seleccionan los conjuntos borrosos antecedente y consecuente correctos respectivamente para la regla j-ésima.

La función $D_r(j)$ se puede escribir en forma matricial para $j = \{1, \ldots, M\}$. Dado que esta operación determina el consecuente para cada regla, puede ser expresada con la operación matricial

$$\mathbf{CD} = \mathbf{C} \left[\begin{array}{c} D_1 \\ D_2 \\ \vdots \\ D_L \end{array} \right]$$

donde la matriz $\mathbf{C}$ (de dimensión $M \times L$) es una matriz de selección de consecuentes con tantas filas como reglas hay en la base de conocimiento; para cada fila todos los valores serán cero excepto el que se encuentre en la columna i cuando el consecuente de salida para esta regla sea D_i. La ecuación (4.10) se puede expresar de forma matricial mediante

$$\boldsymbol{\gamma}^T \mathbf{D} = \boldsymbol{\beta}^T \mathbf{G}^T \mathbf{CD}$$

donde la matriz $\mathbf{G}^T$ se obtiene apilando columnas como la expresada en la ecuación (4.10), siendo j un índice creciente:

$$\mathbf{G}^T = \left[\begin{array}{ccc} (A_{p(1)} \circ A_1) & \cdots & (A_{p(M)} \circ A_1) \\ (A_{p(1)} \circ A_2) & \cdots & (A_{p(M)} \circ A_2) \\ \vdots & \vdots & \vdots \\ (A_{p(1)} \circ A_M) & \cdots & (A_{p(M)} \circ A_M) \end{array} \right]$$

y su traspuesta

$$\mathbf{G} = \begin{bmatrix} (A_{p(1)} \circ A_1) & (A_{p(1)} \circ A_2) & \cdots & (A_{p(1)} \circ A_M) \\ (A_{p(2)} \circ A_1) & (A_{p(2)} \circ A_2) & \cdots & (A_{p(2)} \circ A_M) \\ \vdots & \vdots & \vdots & \vdots \\ (A_{p(M)} \circ A_1) & (A_{p(M)} \circ A_2) & \cdots & (A_{p(M)} \circ A_M) \end{bmatrix}$$

Si se elige una determinada ordenación de los datos, esto es, haciendo que $p(j) = j$, la matriz pasa a ser

$$\begin{aligned} \mathbf{G} &= \begin{bmatrix} (A_1 \circ A_1) & (A_1 \circ A_2) & \cdots & (A_1 \circ A_M) \\ (A_2 \circ A_1) & (A_2 \circ A_2) & \cdots & (A_2 \circ A_M) \\ \vdots & \vdots & \vdots & \vdots \\ (A_M \circ A_1) & (A_M \circ A_2) & \cdots & (A_M \circ A_M) \end{bmatrix} \\ &= [A_1 \dots A_M]^T \circ [A_1 \dots A_M] = \mathbf{A} \circ \mathbf{A}^T = \mathbf{R_A} \end{aligned} \tag{4.11}$$

Esta matriz $\mathbf{R_A}$ se corresponde con la matriz de activación de conjuntos definida en la ecuación (4.2), sección 4.1. La expresión (4.10) puede expresarse finalmente como

$$\begin{aligned} \boldsymbol{\gamma}^T \mathbf{D} = \mathbf{D}^T \boldsymbol{\gamma} &= \mathbf{D}^T \mathbf{C}^T \mathbf{G} \boldsymbol{\beta} \\ \boldsymbol{\gamma} &= \mathbf{C}^T \mathbf{G} \boldsymbol{\beta} \\ &= \mathbf{C}^T \mathbf{R_A} \boldsymbol{\beta} \\ &= \boldsymbol{\Omega} \boldsymbol{\beta} \end{aligned} \tag{4.12}$$

La matriz $\boldsymbol{\Omega} = \mathbf{C}^T \mathbf{G} = \mathbf{C}^T \mathbf{R_A}$ que define la relación entrada-salida se ha denominado en la literatura *matriz de transición* en referencia a su uso inicial en sistemas realimentados [3]. Esta matriz es independiente de la entrada, por lo que puede ser precalculada. Además, una vez hallado su valor, las otras matrices son innecesarias, lo que supone un ahorro en espacio de almacenamiento.

La ecuación (4.12) puede reescribirse finalmente

$$\begin{bmatrix} \gamma_1 \\ \gamma_2 \\ \vdots \\ \gamma_L \end{bmatrix} = \boldsymbol{\Omega} \begin{bmatrix} \beta_1 \\ \beta_2 \\ \vdots \\ \beta_M \end{bmatrix}$$

4.2.4 SAM de 2 entradas y una salida

Supóngase una situación como la que se representa en la Fig. 4.9; el sistema aquí considerado es similar al de la sección anterior excepto por la presencia de una segunda entrada Y, una VL consistente en N posibles conjuntos borrosos B_l definidos en el universo de discurso V. En caso de trabajar con conjuntos definidos sobre un eje real, el sistema realizará una transformación $f : U \times V \subset R^2 \mapsto W \subset R$. Las entradas en forma

Figura 4.9 Sistema borroso genérico con dos entradas y una salida

vectorial son de la forma

$$\begin{aligned} X &= \sum_{k=1}^{M} \beta_k A_k = \boldsymbol{\beta}^T \mathbf{A} \\ Y &= \sum_{l=1}^{N} \alpha_l B_l = \left[\begin{array}{cccc} \alpha_1 & \alpha_2 & \cdots & \alpha_N \end{array} \right] \left[\begin{array}{cccc} B_1 & B_2 & \cdots & B_N \end{array} \right]^T = \boldsymbol{\alpha}^T \mathbf{B} \end{aligned}$$

y la salida es

$$Z = \sum_{n=1}^{L} \gamma_n D_n = \boldsymbol{\gamma}^T \mathbf{D}$$

La expresión (4.8) puede escribirse[3]

$$\begin{aligned} \boldsymbol{\gamma}^T \mathbf{D} &= \sum_{j=1}^{MN} D_{r(j)} \left[\sum_{k=1}^{M} \beta_k (A_{p(j)} \circ A_k) \right] \left[\sum_{l=1}^{N} \alpha_l (B_{q(j)} \circ B_l) \right] \\ &= \sum_{l=1}^{N} \alpha_l \left(\sum_{j=1}^{MN} D_{r(j)} \left[\sum_{k=1}^{M} \beta_k (A_{p(j)} \circ A_k) \right] (B_{q(j)} \circ B_l) \right) \\ &= \sum_{l=1}^{N} \alpha_l \left(\sum_{j=1}^{MN} D_{r(j)} \left[\begin{array}{cccc} \beta_1 & \beta_2 & \cdots & \beta_M \end{array} \right] \left[\begin{array}{l} (A_{p(j)} \circ A_1)(B_{q(j)} \circ B_l) \\ (A_{p(j)} \circ A_2)(B_{q(j)} \circ B_l) \\ \vdots \\ (A_{p(j)} \circ A_M)(B_{q(j)} \circ B_l) \end{array} \right] \right) \end{aligned} \tag{4.13}$$

donde $p(j)$, $q(j)$ y $r(j)$ son operaciones de indexado que seleccionan los conjuntos borrosos adecuados en los antecedentes y el consecuente de la regla j-ésima.

La función $D_{r(j)}$ puede escribirse de forma matricial para $j = \{1, \ldots, MN\}$ multiplicando la matriz $\mathbf{C}$ (de dimensiones $MN \times L$) y el vector $\mathbf{D}$, de la misma forma que en la sección anterior. La ecuación (4.13) puede expresarse en forma matricial por

$$\boldsymbol{\gamma}^T \mathbf{D} = \sum_{l=1}^{N} \alpha_l \boldsymbol{\beta}^T \mathbf{G}_l^T \mathbf{C} \mathbf{D} \tag{4.14}$$

[3]Para simplificar la notación se evitarán los subíndices innecesarios en la ecuación (4.8), renombrando $p_1(j) = p(j)$ y $p_2(j) = q(j)$.

donde la matriz $\mathbf{G}_l^T$ se obtiene apilando columnas como las que aparecen en la ecuación (4.13)

$$\mathbf{G}_l^T = \begin{bmatrix} (A_{p(1)} \circ A_1)(B_{q(1)} \circ B_l) & \cdots & (A_{p(MN)} \circ A_1)(B_{q(MN)} \circ B_l) \\ (A_{p(1)} \circ A_2)(B_{q(1)} \circ B_l) & \cdots & (A_{p(MN)} \circ A_2)(B_{q(MN)} \circ B_l) \\ \vdots & \vdots & \vdots \\ (A_{p(1)} \circ A_M)(B_{q(1)} \circ B_l) & \cdots & (A_{p(MN)} \circ A_M)(B_{q(MN)} \circ B_l) \end{bmatrix}$$

y su traspuesta

$$\mathbf{G}_l = \begin{bmatrix} (A_{p(1)} \circ A_1)(B_{q(1)} \circ B_l) & (A_{p(1)} \circ A_2)(B_{q(1)} \circ B_l) & \cdots & (A_{p(1)} \circ A_M)(B_{q(1)} \circ B_l) \\ (A_{p(2)} \circ A_1)(B_{q(2)} \circ B_l) & (A_{p(2)} \circ A_2)(B_{q(2)} \circ B_l) & \cdots & (A_{p(2)} \circ A_M)(B_{q(2)} \circ B_l) \\ \vdots & \vdots & \vdots & \vdots \\ (A_{p(MN)} \circ A_1)(B_{q(MN)} \circ B_l) & (A_{p(MN)} \circ A_2)(B_{q(MN)} \circ B_l) & \cdots & (A_{p(MN)} \circ A_M)(B_{q(MN)} \circ B_l) \end{bmatrix}$$

Usando una ordenación apropiada para $p(j)$ y $q(j)$ es fácil deducir (ver apéndice A.1.1) que esta matriz puede ser definida de forma compacta en términos del producto tensorial de Kronecker[4] mediante

$$\mathbf{G}_l = \mathbf{R_A} \otimes_{\mathcal{K}} [\mathbf{R_B}\mathbf{E}_l] \tag{4.15}$$

con $\mathbf{R_A}$ y $\mathbf{R_B}$ matrices de activación de reglas definidas

$$\mathbf{R_A} = \begin{pmatrix} A_1 \circ A_1 & \cdots & A_1 \circ A_M \\ A_2 \circ A_1 & \cdots & A_2 \circ A_M \\ \vdots & \cdots & \vdots \\ A_M \circ A_1 & \cdots & A_M \circ A_M \end{pmatrix} = [A_1 \dots A_M]^T \circ [A_1 \dots A_M] = \mathbf{A} \circ \mathbf{A}^T \tag{4.16}$$

y

$$\mathbf{R_B} = \begin{pmatrix} B_1 \circ B_1 & \cdots & B_1 \circ B_N \\ B_2 \circ B_1 & \cdots & B_2 \circ B_N \\ \vdots & \cdots & \vdots \\ B_N \circ B_1 & \cdots & B_N \circ B_N \end{pmatrix} = [B_1 \dots B_N]^T \circ [B_1 \dots B_N] = \mathbf{B} \circ \mathbf{B}^T \tag{4.17}$$

y $\mathbf{E}_l$ un vector de selección de columna, esto es, un vector columna con todas las entradas a cero excepto la que se encuentra en la fila l, cuyo valor es la unidad. Este vector tiene el propósito de extraer la columna l de la matriz $\mathbf{R_B}$.

De este modo, la ecuación (4.13) finalmente puede expresarse

$$\begin{aligned} \boldsymbol{\gamma}^T\mathbf{D} = \mathbf{D}^T\boldsymbol{\gamma} &= \sum_{l=1}^{N} \alpha_l \mathbf{D}^T\mathbf{C}^T\mathbf{G}_l\boldsymbol{\beta} \\ \boldsymbol{\gamma} &= \mathbf{C}^T\left(\sum_{l=1}^{N} \alpha_l \mathbf{G}_l\right)\boldsymbol{\beta} \\ &= \left(\sum_{l=1}^{N} \alpha_l \boldsymbol{\Omega}_l\right)\boldsymbol{\beta} \end{aligned} \tag{4.18}$$

[4]Aunque en la literatura [41] el símbolo $\otimes$ denota habitualmente el producto tensorial de Kronecker, se ha preferido denotarlo aquí por $\otimes_{\mathcal{K}}$ para evitar confusiones con el operador t-norma, representado por algunos autores por $\otimes$.

con $\mathbf{\Omega}_l = \mathbf{C}^T\mathbf{G}_l$ la matriz de transición del sistema para la entrada $Y = B_l$. Al igual que en la sección anterior, la expresión (4.18) puede escribirse:

$$\begin{bmatrix} \gamma_1 \\ \gamma_2 \\ \vdots \\ \gamma_L \end{bmatrix} = \left(\sum_{l=1}^{N} \alpha_l \mathbf{\Omega}_l \right) \begin{bmatrix} \beta_1 \\ \beta_2 \\ \vdots \\ \beta_M \end{bmatrix}$$

Ejemplo: Construcción de una matriz de transición a partir de la base de reglas

Supóngase que la entrada X toma valores en el rango $\{A_1, \ldots, A_4\} \in U$ $(M = 4)$, la entrada Y en $\{B_1, B_2, B_3\} \in V$ $(N = 3)$ y la salida Z tiene como posibles valores $\{D_1, \ldots, D_5\} \in W$ $(L = 5)$. Para simplificar la notación se supondrá que

$$A_i \circ A_j = \begin{cases} = 1 & si \quad i = j \\ = \delta_A & si \quad |i - j| = 1 \\ = 0 & si \quad |i - j| > 1 \end{cases} \tag{4.19}$$

y respecto a la segunda variable

$$B_i \circ B_j = \begin{cases} = 1 & si \quad i = j \\ = \delta_B & si \quad |i - j| = 1 \\ = 0 & si \quad |i - j| > 1 \end{cases} \tag{4.20}$$

La base de reglas se muestra en las tres columnas de la izquierda de la tabla 4.1. La columna de la derecha muestra cómo se activa cada regla cuando $Y = B_2$ y $X = \beta_1 A_1 + \beta_2 A_2 + \beta_3 A_3 + \beta_4 A_4$. En primer lugar se construyen las matrices $\mathbf{R_A}$ y $\mathbf{R_B}$ (ecuaciones (4.16) y (4.17)), que en este caso serán

$$\mathbf{R_A} = \begin{pmatrix} 1 & \delta_A & 0 & 0 \\ \delta_A & 1 & \delta_A & 0 \\ 0 & \delta_A & 1 & \delta_A \\ 0 & 0 & \delta_A & 1 \end{pmatrix} \tag{4.21}$$

y

$$\mathbf{R_B} = \begin{pmatrix} 1 & \delta_B & 0 \\ \delta_B & 1 & \delta_B \\ 0 & \delta_B & 1 \end{pmatrix} \tag{4.22}$$

A_j	B_j	D_j	R_j
A_1	B_1	D_1	$(\beta_1 + \delta_A\beta_2)\delta_B D_1$
A_1	B_2	D_2	$(\beta_1 + \delta_A\beta_2)D_2$
A_1	B_3	D_2	$(\beta_1 + \delta_A\beta_2)\delta_B D_2$
A_2	B_1	D_2	$(\delta_A\beta_1 + \beta_2 + \delta_A\beta_3)\delta_B D_2$
A_2	B_2	D_3	$(\delta_A\beta_1 + \beta_2 + \delta_A\beta_3)D_3$
A_2	B_3	D_4	$(\delta_A\beta_1 + \beta_2 + \delta_A\beta_3)\delta_B D_4$
A_3	B_1	D_2	$(\delta_A\beta_2 + \beta_3 + \delta_A\beta_4)\delta_B D_2$
A_3	B_2	D_3	$(\delta_A\beta_2 + \beta_3 + \delta_A\beta_4)D_3$
A_3	B_3	D_3	$(\delta_A\beta_2 + \beta_3 + \delta_A\beta_4)\delta_B D_3$
A_4	B_1	D_4	$(\delta_A\beta_3 + \beta_4)\delta_B D_4$
A_4	B_2	D_4	$(\delta_A\beta_3 + \beta_4)D_4$
A_4	B_3	D_5	$(\delta_A\beta_3 + \beta_4)\delta_B D_5$

Tabla 4.1 Base de reglas para el ejemplo (tres columnas de la izquierda) y activación de reglas cuando las entradas son $X = \beta_1 A_1 + \beta_2 A_2 + \beta_3 A_3 + \beta_4 A_4$ y $Y = B_2$ (columna de la derecha).

respectivamente. El siguiente paso es construir las matrices $\mathbf{G}_l$. (ecuación (4.15)) Como ejemplo se construirá solamente $\mathbf{G}_2$

$$\mathbf{G}_2 = \mathbf{R_A} \otimes_{\mathcal{K}} [\mathbf{R_B}\mathbf{E}_2] = \begin{pmatrix} \delta_B & \delta_A\delta_B & 0 & 0 \\ 1 & \delta_A & 0 & 0 \\ \delta_B & \delta_A\delta_B & 0 & 0 \\ \delta_A\delta_B & \delta_B & \delta_A\delta_B & 0 \\ \delta_A & 1 & \delta_A & 0 \\ \delta_A\delta_B & \delta_B & \delta_A\delta_B & 0 \\ 0 & \delta_A\delta_B & \delta_B & \delta_A\delta_B \\ 0 & \delta_A & 1 & \delta_A \\ 0 & \delta_A\delta_B & \delta_B & \delta_A\delta_B \\ 0 & 0 & \delta_A\delta_B & \delta_B \\ 0 & 0 & \delta_A & 1 \\ 0 & 0 & \delta_A\delta_B & \delta_B \end{pmatrix} \tag{4.23}$$

La matriz de selección de consecuente $\mathbf{C}$ pasa a ser

$$\mathbf{C} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix} \tag{4.24}$$

BIBLIOTECA VIRTUAL MIGUEL DE CERVANTES

Nótese que estas dos matrices son una transcripción casi directa de los contenidos de la tabla 4.1. El último paso es la construcción de las matrices de transición $\mathbf{\Omega}_l$. Para el caso de $\mathbf{\Omega}_2$ se obtiene

$$\mathbf{\Omega}_2 = \mathbf{C}^T\mathbf{G_2} = \begin{pmatrix} \delta_B & \delta_A\delta_B & 0 & 0 \\ 1+\delta_B+\delta_A\delta_B & \delta_A+2\delta_A\delta_B+\delta_B & (1+\delta_A)\delta_B & \delta_A\delta_B \\ \delta_A & 1+\delta_A+\delta_A\delta_B & 1+\delta_A+\delta_B & \delta_A(1+\delta_B) \\ \delta_A\delta_B & \delta_B & \delta_A+2\delta_A\delta_B & 1+\delta_B \\ 0 & 0 & \delta_A\delta_B & \delta_B \end{pmatrix} \tag{4.25}$$

4.2.5 SAM de múltiples entradas y una salida

En el caso en que el sistema tenga múltiples entradas, las expresiones de las secciones anteriores han de ser ampliadas. Se estudiará en primer lugar el caso particular de tres entradas para luego hacer una extensión a un sistema genérico con múltiples entradas.

Caso particular: SAM de tres entradas

Supóngase un sistema con tres entradas borrosas que pueden escribirse

$$\begin{aligned} X &= \sum_{k=1}^{M} \beta_k A_k = \boldsymbol{\beta}^T \mathbf{A} \\ Y_1 &= \sum_{l=1}^{N_1} \alpha_l^1 B_l^1 = (\boldsymbol{\alpha}^1)^T \mathbf{B}^1 \\ Y_2 &= \sum_{t=1}^{N_2} \alpha_t^2 B_t^2 = (\boldsymbol{\alpha}^2)^T \mathbf{B}^2 \end{aligned}$$

y la salida

$$Z = \sum_{n=1}^{L} \gamma_n D_n = \boldsymbol{\gamma}^T \mathbf{D}$$

La diferencia en el modo de escribir las entradas X e Y_i se debe sólo a conveniencia matemática, como se verá a continuación.

La ecuación (4.8) puede reescribirse, en este caso:

$$\begin{aligned}
\boldsymbol{\gamma}^T \mathbf{D} &= \sum_{j=1}^{MN_1N_2} D_{r(j)} \left[\sum_{k=1}^{M} \beta_k (A_{p(j)} \circ A_k)\right] \left[\sum_{l=1}^{N_1} \alpha_l^1 (B_{q(j)}^1 \circ B_l^1)\right] \left[\sum_{t=1}^{N_2} \alpha_t^2 (B_{s(j)}^2 \circ B_t^2)\right] \\
&= \sum_{l=1}^{N_1}\sum_{t=1}^{N_2} \alpha_l^1 \alpha_t^2 \left(\sum_{j=1}^{MN_1N_2} D_{r(j)} \left[\sum_{k=1}^{M} \beta_k (A_{p(j)} \circ A_k)\right] (B_{q(j)}^1 \circ B_l^1)(B_{s(j)}^2 \circ B_t^2)\right) \quad (4.26) \\
&= \sum_{l=1}^{N_1}\sum_{t=1}^{N_2} \alpha_l^1 \alpha_t^2 \left(\sum_{j=1}^{MN_1N_2} D_{r(j)} \left[\sum_{k=1}^{M} \beta_k (A_{p(j)} \circ A_k)\right] B_{q(j)s(j)}^{12}(l,t)\right) \quad (4.27) \\
&= \sum_{l=1}^{N_1}\sum_{t=1}^{N_2} \alpha_l^1 \alpha_t^2 \left(\sum_{j=1}^{MN_1N_2} D_{r(j)} \begin{bmatrix} \beta_1 & \beta_2 & \cdots & \beta_M \end{bmatrix} \begin{bmatrix} (A_{p(j)} \circ A_1) B_{q(j)s(j)}^{12}(l,t) \\ (A_{p(j)} \circ A_2) B_{q(j)s(j)}^{12}(l,t) \\ \vdots \\ (A_{p(j)} \circ A_M) B_{q(j)s(j)}^{12}(l,t) \end{bmatrix}\right) \quad (4.28)
\end{aligned}$$

donde $p(j)$, $q(j)$, $s(j)$ y $r(j)$ son operaciones de indexado que permiten seleccionar los antecedentes y consecuentes adecuados para la regla j-ésima.

De nuevo la función $D_{r(j)}$ se puede escribir en forma matricial para $j = \{1, \ldots, MN_1N_2\}$ multiplicando la matriz $\mathbf{C}$ (de dimensión $MN_1N_2 \times L$) por el vector $\mathbf{D}$, tal y como se han definido en las secciones anteriores. La ecuación (4.28) puede expresarse en forma matricial como

$$\boldsymbol{\gamma}^T \mathbf{D} = \sum_{l=1}^{N_1}\sum_{t=1}^{N_2} \alpha_l^1 \alpha_t^2 \boldsymbol{\beta}^T \mathbf{G}_{l,t}^T \mathbf{C}\mathbf{D} \quad (4.29)$$

donde la matriz $\mathbf{G}_{l,t}^T$ se obtiene apilando columnas como las que aparecen en la ecuación (4.28) con j un índice creciente. Los detalles del proceso subsiguiente aparecen en el apéndice A.1.2. De este modo se llega a

$$\begin{aligned}
\mathbf{G}_{l,t} &= \mathbf{R}_{\mathbf{A}} \otimes_{\mathcal{K}} [(\mathbf{R}_{\mathbf{B}^1} \otimes_{\mathcal{K}} (\mathbf{R}_{\mathbf{B}^2} \mathbf{E}_t))\mathbf{E}_l] \\
&= \mathbf{R}_{\mathbf{A}} \otimes_{\mathcal{K}} [(\mathbf{R}_{\mathbf{B}^1} \otimes_{\mathcal{K}} ((\mathbf{R}_{\mathbf{B}^2} \otimes_{\mathcal{K}} 1)\mathbf{E}_t))\mathbf{E}_l] \quad (4.30)
\end{aligned}$$

con $\mathbf{R}_{\mathbf{A}}$, $\mathbf{R}_{\mathbf{B}^1}$ y $\mathbf{R}_{\mathbf{B}^2}$ matrices de activación de reglas definidas como en la sección 4.2.4 y $\mathbf{E}_h$ un vector de elección de columna con todas las entradas cero excepto la correspondiente a la fila h, cuyo valor es la unidad. El propósito de este vector es la extracción de la columna h de la matriz a la que multiplica.

Finalmente se puede reescribir la ecuación (4.28)

$$\begin{aligned}
\boldsymbol{\gamma}^T\mathbf{D} = \mathbf{D}^T\boldsymbol{\gamma} &= \mathbf{D}^T\mathbf{C}^T\left(\sum_{l=1}^{N_1}\sum_{t=1}^{N_2}\alpha_l^1\alpha_t^2\mathbf{G}_{l,t}\right)\boldsymbol{\beta} \\
\boldsymbol{\gamma} &= \mathbf{C}^T\left(\sum_{l=1}^{N_1}\sum_{t=1}^{N_2}\alpha_l^1\alpha_t^2\mathbf{G}_{l,t}\right)\boldsymbol{\beta} \\
&= \sum_{l=1}^{N_1}\sum_{t=1}^{N_2}\alpha_l^1\alpha_t^2\left(\mathbf{C}^T\mathbf{G}_{l,t}\right)\boldsymbol{\beta} \\
&= \left(\sum_{l=1}^{N_1}\sum_{t=1}^{N_2}\alpha_l^1\alpha_t^2\boldsymbol{\Omega}_{l,t}\right)\boldsymbol{\beta} \qquad (4.31)
\end{aligned}$$

con $\boldsymbol{\Omega}_{l,t} = \mathbf{C}^T\mathbf{G}_{l,t}$ la matriz de transición del sistema para las entradas $Y_1 = B_l^1$ y $Y_2 = B_t^2$.

Caso general

Figura 4.10 Sistema borroso con múltiples entradas y una salida

Considérese el sistema borroso con $F+1$ entradas y una salida que aparece en la figura 4.10. En el caso de que los conjuntos se definan sobre un eje numérico real el sistema realiza una transformación $f : U \times V \subset R^{F+1} \mapsto W \subset R$, donde $V = V_1 \times V_2 \times \cdots \times V_F \subset R^F$. Los conjuntos borrosos de entrada se pueden escribir como

$$\begin{aligned}
Y_1 &= \sum_{i_1=1}^{N_1}\alpha_{i_1}^1 B_{i_1}^1 = (\boldsymbol{\alpha}^1)^T\mathbf{B}^1 \\
Y_2 &= \sum_{i_2=1}^{N_2}\alpha_{i_2}^2 B_{i_2}^2 = (\boldsymbol{\alpha}^2)^T\mathbf{B}^2 \\
&\vdots \quad\quad \vdots \\
Y_F &= \sum_{i_F=1}^{N_F}\alpha_{i_F}^F B_{i_F}^F = (\boldsymbol{\alpha}^F)^T\mathbf{B}^F \\
X &= \sum_{k=1}^{M}\beta_k A_k = (\boldsymbol{\beta})^T\mathbf{A}
\end{aligned}$$

Se definen la matrices de activación para las entradas:

$$\begin{aligned}
\mathbf{R}_{\mathbf{B}^i} &= \left[B_1^i \ldots B_{N_i}^i\right]^T \circ \left[B_1^i \ldots B_{N_i}^i\right] = \mathbf{B}^i \circ (\mathbf{B}^i)^T \qquad (\text{tamaño } N_i \times N_i, i = 1, \ldots, F) \\
\mathbf{R}_{\mathbf{A}} &= [A_1 \ldots A_M]^T \circ [A_1 \ldots A_M] = \mathbf{A} \circ (\mathbf{A})^T \qquad (\text{tamaño } M \times M)
\end{aligned}$$

Las matrices de composición se definen como:

$$\begin{array}{lll}
\mathbf{\Lambda}_{i_F} = & (\mathbf{R}_{\mathbf{B}^F} \otimes_{\mathcal{K}} 1)\mathbf{E}_{i_F} & i_F = 1, \ldots, N_F \\
\mathbf{\Lambda}_{i_{F-1} \cdot i_F} = & (\mathbf{R}_{\mathbf{B}^{F-1}} \otimes_{\mathcal{K}} \mathbf{\Lambda}_{i_F})\mathbf{E}_{i_{F-1}} & i_{F-1} = 1, \ldots, N_{F-1} \\
\mathbf{\Lambda}_{i_{F-2} \cdot i_{F-1} \cdot i_F} = & (\mathbf{R}_{\mathbf{B}^{F-2}} \otimes_{\mathcal{K}} \mathbf{\Lambda}_{i_{F-1} \cdot i_F})\mathbf{E}_{i_{F-2}} & i_{F-2} = 1, \ldots, N_{F-2} \\
\vdots & \vdots & \vdots \\
\mathbf{\Lambda}_{\prod_{j=1}^{F} i_j} = & (\mathbf{R}_{\mathbf{B}^1} \otimes_{\mathcal{K}} \mathbf{\Lambda}_{\prod_{j=2}^{F} i_j})\mathbf{E}_{i_1} & i_1 = 1, \ldots, N_1
\end{array}$$

Con estas definiciones y siguiendo la metodología usada anteriormente para llegar a la expresión (4.30) se puede escribir

$$\mathbf{G}_{\prod_{j=1}^{F} i_j} = \mathbf{R}_{\mathbf{A}} \otimes_{\mathcal{K}} \mathbf{\Lambda}_{\prod_{j=1}^{F} i_j} \tag{4.32}$$

En estas ecuaciones, $\mathbf{E}_{i_j}$ $(j = 1, \ldots, F)$ es un vector columna de la longitud apropiada, que se encarga de extraer la columna i_j de la matriz a la que multiplica. Todas sus entradas serán cero excepto el elemento i_j, que será la unidad. Finalmente, la matriz de transición para las entradas seleccionadas será

$$\mathbf{\Omega}_{\prod_{j=1}^{F} i_j} = \mathbf{C}^T \mathbf{G}_{\prod_{j=1}^{F} i_j} \qquad i_j = 1, \cdots, N_j \tag{4.33}$$

La relación entre coeficientes viene dada ahora por

$$\boldsymbol{\gamma} = \left(\sum_{i_1=1}^{N_1} \sum_{i_2=1}^{N_2} \sum_{i_3=1}^{N_3} \cdots \sum_{i_F=1}^{N_F} \alpha_{i_1}^1 \alpha_{i_2}^2 \alpha_{i_3}^3 \cdots \alpha_{i_F}^F \mathbf{\Omega}_{\prod_{j=1}^{F} i_j} \right) \boldsymbol{\beta} \tag{4.34}$$

Debe tenerse en cuenta de nuevo que estas operaciones son independientes de la resolución del universo de discurso de cada uno de los términos lingüísticos. Esta resolución sólo es necesaria para el cálculo de $\mathbf{R}_{\mathbf{A}}$ y $\mathbf{R}_{\mathbf{B}^i}$, que pueden ser olvidadas (junto con $\mathbf{C}$ y $\mathbf{G}_{\prod_{j=1}^{F} i_j}$) cuando se hayan creado las matrices de transición. Por lo tanto, es posible incluso manejar conjuntos borrosos definidos sobre un universo continuo.

4.2.6 Extensión a entradas numéricas

Existen casos en los que en un sistema borroso conviven entradas de diversas clases [45], como, por ejemplo, entradas numéricas y entradas lingüísticas. En la inferencia en modo matricial, hasta ahora, no se ha contemplado esta posibilidad. Será necesario por lo tanto extender el sistema para que acepte distintos tipos de información.

Un ejemplo sería un sistema que mida características de personas. El color del pelo sería un término lingüístico, perfectamente definible mediante gránulos (*rubio*). En cambio, la altura podría ser un valor numérico.

Un sistema de esta clase, por ejemplo podría tener reglas de la forma

Si x es A_i e Y es B_j entonces Z es C_{ij}

donde la entrada $x \in R$ es un número y la entrada $Y \subset V$ un conjunto borroso.

Supóngase que se tiene una variable lingüística $\mathcal{L}$, definida mediante M conjuntos borrosos A_i. Los valores que van a entrar al sistema son numéricos (no borrosos). Dicho valor se denota por x_0. La forma más habitual para calcular la activación de esta entrada no borrosa es calcular $A_i(x_0)$ (secc 2.3.6). Se define por lo tanto un vector de entrada, equivalente a los que se usaban en el caso de entradas borrosas. Este vector va a tener M componentes, y se define de la forma:

$$X = \begin{bmatrix} A_1(x_0) \\ A_2(x_0) \\ \vdots \\ A_M(x_0) \end{bmatrix} = \begin{bmatrix} \beta_1 \\ \beta_2 \\ \vdots \\ \beta_M \end{bmatrix} \tag{4.35}$$

Nótese que para calcular $A_k(x_0)$ se necesita que el conjunto esté definido sobre un eje numérico.

Se define a continuación la matriz de activación de reglas $\mathbf{R_A}$. Dado que la activación va implícita en el vector X, en caso de entradas numéricas la matriz pasa a ser la identidad de tamaño $M \times M$.

$$\mathbf{R_A} = \begin{pmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \cdots & \vdots \\ 0 & 0 & \cdots & 1 \end{pmatrix} \tag{4.36}$$

Puede verse que esto es así razonando a partir de la expresión (4.13). La activación de cada antecedente se expresa mediante $\sum_k \beta_k (A_{p(j)} \circ A_k)$. Esta activación en el caso de entradas numéricas viene dada por los coeficientes $\beta_i = A_i(x_0)$, por lo que el valor de $(A_{p(j)} \circ A_k)$ debe ser binario; uno cuando $A_{p(j)} = A_k$ y cero en el resto de los casos. De aquí, la matriz $\mathbf{R_A}$ pasa a ser la matriz identidad.

A partir de este punto se opera de igual modo que si las entradas fueran conjuntos borrosos. Nótese que el uso de matrices de transición no está justificado plenamente si todas las entradas son numéricas, ya que al tener que calcular los valores de $A_i(x)$ no se ahorra operaciones. Sí tiene sentido en el caso de convivencia de distintas clases de datos.

Ejemplo: entradas numéricas y borrosas

Como ejemplo de un sistema con entradas de distintos tipos se va a plantear una versión reducida del sistema descrito en [7] y que será estudiado con profundidad en

el capítulo 5. Supóngase un bloque borroso dentro de un sistema de seguimiento de vehículos basado en texto. El sistema va a tener dos entradas

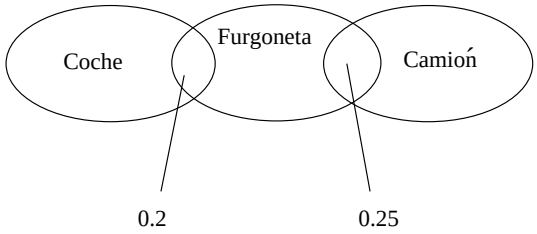


Figura 4.11 Gránulos de la variable *Vehículo*

BIBLIOTECA VIRTUAL

1. Nombre del vehículo observado. Va a definirse como una variable lingüística $\mathcal{V}$ formada por 3 gránulos borrosos, *Coche*, *Furgoneta*, *Camión* (Fig. 4.11) definidos por su matriz de activación

$$\mathbf{R_A} = \begin{pmatrix} 1 & 0.2 & 0 \\ 0.2 & 1 & 0.25 \\ 0 & 0.25 & 1 \end{pmatrix}$$

   La entrada al sistema será el nombre del vehículo observado.

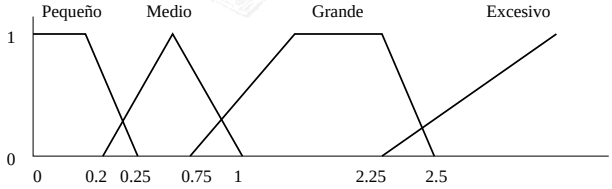


Figura 4.12 Conjuntos de la variable *Error*

2. Error de predicción de la posición del vehículo. Se define una variable lingüística $\mathcal{E}$ formada por cuatro conjuntos borrosos $E_i(y)$, definidos sobre un eje numérico, de valores *Pequeño*, *Medio*, *Grande*, *Excesivo* (Fig. 4.12). La matriz de activación en este caso será la identidad de orden 4.

$$\mathbf{R_B} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

   La entrada al sistema será el error medido (numérico).

La salida será un conjunto de la variable lingüística *Probabilidad*, con valores *Improbable*, *Posible* y *Probable*. Las reglas serán de la forma:

Regla: Si X es *Camión* e y es *Pequeño* entonces Z es *Probable*

La operativa matricial de este sistema quedará de la forma:

$$\begin{bmatrix} \gamma_1 \\ \gamma_2 \\ \gamma_3 \end{bmatrix} = \left(\sum_{l=1}^{4} E_l(y_0) \mathbf{\Omega}_l \right) \begin{bmatrix} \beta_1 \\ \beta_2 \\ \beta_3 \end{bmatrix} \tag{4.37}$$

donde γ_i son los coeficientes del vector de salida (*probabilidad*), β_i los coeficientes del vector de la variable *Vehículo* y $E_i(y_0)$ la evaluación del valor del error de entrada y_0 sobre los conjuntos de la variable *Error*.

4.2.7 Reglas con múltiples consecuentes

BIBLIOTECA VIRTUAL

En lo visto hasta ahora, los elementos de cada fila de la matriz de extracción de reglas $\mathbf{C}$ han tomado valor 1 ó 0, siendo sólo un elemento el que tomaba valor unidad. Esto se debe a que se ha trabajado con reglas de la forma

$$\text{Si } X \text{ es } A_k \text{ entonces } Z \text{ es } D_n$$

en las que sólo existe un consecuente de salida, que se activa totalmente. En algunos casos (como los árboles de transición, apéndice C) una regla puede tener como salida varios consecuentes, ponderados por un cierto peso.

Supóngase un SAM, cuya salida Z es una VL consistente en L conjuntos borrosos D_n. Las base de reglas va a poder formularse

$$\text{Si } X_1 \text{ es } A_{p_1(j)} \text{ y } \cdots \text{ y si } X_M \text{ es } A_{p_M(j)} \text{ entonces } Z \text{ es } \omega_1^j D_1 + \omega_2^j D_2 + \cdots + \omega_L^j D_L$$

Los ω_i^j representan los pesos por los que se pondera cada uno de los conjuntos de salida. El caso de tener un solo consecuente será por lo tanto una particularización en que todos los pesos valen cero, excepto uno que vale la unidad.

Para codificar los distintos pesos dentro de la matriz de reglas $\mathbf{C}$ simplemente hay que colocar el valor ω_i de la regla j-ésima en la columna i de la fila j de la matriz. Suponiendo N reglas:

$$\mathbf{C} = \begin{array}{c} \\ R_1 \\ R_2 \\ \vdots \\ R_N \end{array} \begin{array}{c} \begin{array}{cccc} D_1 & D_2 & \cdots & D_L \end{array} \\ \begin{pmatrix} \omega_1^1 & \omega_2^1 & \cdots & \omega_L^1 \\ \omega_1^2 & \omega_2^2 & \cdots & \omega_L^2 \\ \cdots & \cdots & & \cdots \\ \omega_1^N & \omega_2^N & \cdots & \omega_L^N \end{pmatrix} \end{array}$$

4.2.8 Cálculo de centroides

Dado un sistema borroso en el que la inferencia se realiza usando una serie de matrices de transición $\mathbf{\Omega}_i$ y en el que los conjuntos de salida llevan asociada una dimensión numérica, el valor del centroide de salida puede calcularse a partir de la siguiente expresión [3]:

$$C_k = \frac{[c_1 \;\; c_2 \cdots c_L]\boldsymbol{\gamma}}{[1 \;\; 1 \cdots 1]\boldsymbol{\gamma}} = \frac{\mathbf{c}^T\boldsymbol{\gamma}}{||\boldsymbol{\gamma}||_{L_1}} \tag{4.38}$$

siendo $\mathbf{c}$ el vector de centroides c_i de los conjuntos de salida y $\boldsymbol{\gamma}$ el vector de de salida. La ecuación 4.38 es genérica para cualquier sistema borroso (independientemente del número de entradas)y es totalmente equivalente a calcular el centroide tal como lo expresa Kosko [72].

BIBLIOTECA VIRTUAL

4.2.9 Modificadores lingüísticos

Tal y como se apuntó en la sección 4.1.4, la restricción impuesta al sistema de que la entrada sea una combinación de los conjuntos de la VL de entrada puede ser superada mediante el uso de modificadores lingüísticos.

Como ya se indicó en dicha sección, el modificador en el nuevo marco de trabajo resultará en una matriz $\mathbf{\Delta}_H$ (ecuación (4.4)) que modifica los valores de la matriz $\mathbf{R_A}$ original. En lo que sigue se proporcionan los detalles.

Matriz de transición del modificador

Considérese un modificador $H()$ aplicado sobre un conjunto A_k, dando como resultado el conjunto de entrada $A = H(A_k)$. La salida de un sistema SAM con una entrada y una salida puede escribirse como

$$\boldsymbol{\gamma}^T\mathbf{D} = \sum_{j=1}^{M} D_{r(j)} \left[A_{p(j)} \circ H(A_k) \right]$$

De acuerdo con lo visto en la sección 4.1.4, un modificador altera las intersecciones entre gránulos o conjuntos en una determinada cantidad. Puede escribirse entonces

$$A_{p(j)} \circ H(A_k) = A_{p(j)} \circ A_k + \delta_{p(j)k}$$

La cantidad $\delta_{p(j)k}$ es el cambio que experimenta la composición $A_{p(j)} \circ A_k$ al aplicar el modificador $H()$ sobre A_k. El proceso completo de inferencia quedará

$$\begin{aligned} \boldsymbol{\gamma}^T\mathbf{D} &= \sum_{j=1}^{M} \left[(A_{p(j)} \circ A_k) + \delta_{p(j)k} \right] D_{r(j)} \\ &= \left[\; (A_{p(1)} \circ A_k) + \delta_{p(1)k} \quad \cdots \quad (A_{p(M)} \circ A_k) + \delta_{p(M)k} \; \right] \mathbf{CD} \end{aligned} \tag{4.39}$$

Se asume por simplicidad que $p(j) = j$. Definiendo la matriz

$$\boldsymbol{\Delta}_H = \begin{bmatrix} \delta_{11} & \delta_{12} & \cdots & \delta_{1M} \\ \delta_{21} & \delta_{22} & \cdots & \delta_{2M} \\ \vdots & \vdots & \ddots & \vdots \\ \delta_{M1} & \delta_{M2} & \cdots & \delta_{MM} \end{bmatrix}$$

y $\boldsymbol{\beta}_k$ un vector columna con todas las entradas cero excepto la fila k, se puede escribir

$$\begin{aligned} \boldsymbol{\gamma}^T\mathbf{D} &= \boldsymbol{\beta}_k^T(\mathbf{R_A} + \boldsymbol{\Delta}_H)^T\mathbf{CD} = \\ \mathbf{D}^T\boldsymbol{\gamma} &= \mathbf{D}^T\mathbf{C}^T(\mathbf{R_A} + \boldsymbol{\Delta}_H)\boldsymbol{\beta}_k \\ \boldsymbol{\gamma} &= \mathbf{C}^T(\mathbf{R_A} + \boldsymbol{\Delta}_H)\boldsymbol{\beta}_k \\ &= \left(\boldsymbol{\Omega} + \mathbf{C}^T\boldsymbol{\Delta}_H\right)\boldsymbol{\beta}_k \end{aligned}$$

Si se renombra $\boldsymbol{\Delta\Omega}_H = \mathbf{C}^T\boldsymbol{\Delta}_H$ la expresión da lugar a

$$\boldsymbol{\gamma} = (\boldsymbol{\Omega} + \boldsymbol{\Delta\Omega}_H)\,\boldsymbol{\beta}_k \tag{4.40}$$

De este modo un modificador aplicado sobre los conjuntos de entrada puede verse como una matriz que modifica los valores de la matriz de transición del sistema.

Ejemplo de modificador lingüístico

Supóngase un sistema SAM con dos entradas borrosas cuyas matrices de activación sean

$$\mathbf{R_A} = \mathbf{R_B} = \begin{pmatrix} 1 & 0.4 & 0 \\ 0.4 & 1 & 0.5 \\ 0 & 0.5 & 1 \end{pmatrix}$$

y la matriz de reglas $\mathbf{C}$ es

$$\mathbf{C} = \begin{pmatrix} 1 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

de modo que las matrices de transición del sistema son

$$\boldsymbol{\Omega}_1 = \begin{pmatrix} 1.8 & 1.56 & 0.5 \\ 0.16 & 1.1 & 1.6 \\ 0 & 0 & 0 \end{pmatrix} \quad \boldsymbol{\Omega}_2 = \begin{pmatrix} 1.56 & 0.96 & 0.2 \\ 1.1 & 2.4 & 2.15 \\ 0 & 0.25 & 0.5 \end{pmatrix} \quad \boldsymbol{\Omega}_3 = \begin{pmatrix} 0.5 & 0.2 & 0 \\ 1.6 & 2.15 & 1.25 \\ 0 & 0.5 & 1 \end{pmatrix}$$

La extensión de lo visto en el apartado anterior a sistemas de más entradas es casi inmediato teniendo en cuenta que el operador $\otimes_{\mathcal{K}}$ es lineal. El caso de 2 entradas y una salida se encuentra en el apéndice A.2.

Se define un modificador lingüístico que actuará sobre $\mathbf{R_A}$ y que tiene por matriz de activación

$$\mathbf{\Delta}_H = \begin{pmatrix} 0 & \alpha_1 \cdot 0.2 & 0 \\ \alpha_1 \cdot 0.2 & 0 & \alpha_2 \cdot 0.2 \\ 0 & \alpha_2 \cdot 0.2 & 0 \end{pmatrix}$$

Nótese que la expresión final queda en función de α_i, de modo que pueda admitira distintos valores para estos parámetros. Las matrices de transición de este modificador se calcularán (ecuación (A.1))

$$\mathbf{\Delta\Omega}_i = \mathbf{C}^T \left(\mathbf{\Delta}_H \otimes_{\mathcal{K}} [\mathbf{R_B E}_i]\right)$$

dando lugar a las siguientes matrices

$$\mathbf{\Delta\Omega}_1 = \begin{pmatrix} 1/5 \cdot \alpha_1 & 7/25 \cdot \alpha_1 & 1/5 \cdot \alpha_2 \\ 2/25 \cdot \alpha_1 & 7/25 \cdot \alpha_2 & 2/25 \cdot \alpha_2 \\ 0 & 0 & 0 \end{pmatrix}$$

$$\mathbf{\Delta\Omega}_2 = \begin{pmatrix} 2/25 \cdot \alpha_1 & 7/25 \cdot \alpha_1 & 2/25 \cdot \alpha_2 \\ 3/10 \cdot \alpha_1 & 1/10 \cdot \alpha_1 + 7/25 \cdot \alpha_2 & 3/10 \cdot \alpha_2 \\ 0 & 1/10 \cdot \alpha_2 & 0 \end{pmatrix}$$

$$\mathbf{\Delta\Omega}_3 = \begin{pmatrix} 0 & 1/10 \cdot \alpha_1 & 0 \\ 3/10 \cdot \alpha_1 & 1/5 \cdot \alpha_1 + 1/10 \cdot \alpha_2 & 3/10 \cdot \alpha_2 \\ 0 & 1/5 \cdot \alpha_2 & 0 \end{pmatrix}$$

4.2.10 Evaluación de la complejidad

A lo largo del capítulo se ha aludido al ahorro operacional que resulta del hecho de usar matrices de transición. En este apartado se cuantificará esta ganancia computacional para el caso de los sistemas SAM.

Sistema de dos entradas y una salida

Para un sistema borroso con dos entradas y una salida (acorde con la expresión 4.8), la relación entre éstas viene dada por

$$Z = \sum_{j=1}^{MN} D_{r(j)} \left[A_{p(j)} \circ X\right] \left[B_{q(j)} \circ Y\right]$$

Supóngase que los conjuntos borrosos de la primera entrada se definen con una resolución de P_1 puntos, P_2 para la segunda entrada y P_3 para la salida. El número de operaciones realizadas en un sistema SAM para la regla j-ésima es:

- $A_{p(j)}(X) = A_{p(j)} \circ X = \sum_{i=1}^{P_1} A_{p(j)}(x_i)X(x_i)$: P_1 productos y $(P_1 - 1)$ sumas.
- $B_{q(j)}(Y) = B_{q(j)} \circ Y = \sum_{i=1}^{P_2} B_{p(j)}(x_i)X(x_i)$: P_2 productos y $(P_2 - 1)$ sumas.
- $D_{r(j)}A_{p(j)}(X)B_{q(j)}(Y)$: $P_3 + 1$ productos.

El número total de operaciones será de $MN \times (P_1 + P_2 + P_3 + 1)$ productos y $MN \times (P_1 + P_2 + P_3 - 2) - P_3$ sumas; definiendo $R = MN$ como el número de reglas en la base, entonces $O_p^{SAM} = [R \times (P_1 + P_2 + P_3 + 1)]_\times + [R \times (P_1 + P_2 + P_3 - 2) - P_3]_+$.

Usando matrices de transición para $X = \boldsymbol{\beta}^T \mathbf{A}$ e $Y = \boldsymbol{\alpha}^T \mathbf{B}$, las operaciones que se llevan a cabo son únicamente las correspondientes a la ecuación (4.18), es decir, $L \times M \times (N + 1)$ productos y $L \times M \times N - L$ sumas. Por lo tanto $O_p^{FITM} = [L \times (R + M)]_\times + [L \times R - L]_+$:

SAM	FITM
$O_\times^{SAM} \approx R \times [P_1 + P_2 + P_3]$	$O_\times^{FITM} \approx R \times L$
$O_+^{SAM} \approx R \times [P_1 + P_2 + P_3]$	$O_+^{FITM} \approx R \times L$

El número de conjuntos en el espacio de salida (L) será siempre mucho menor que el número de puntos de resolución de dicho espacio (P_3), por lo que el número de operaciones llevadas a cabo con el método FITM será mucho menor que las llevadas a cabo por el método tradicional.

En un caso particular implementado en MATLAB, con $P1 = P2 = P3 = 100$, $M = 3$, $N = 5$ ($R = 15$) y $L = 4$, se midieron las operaciones en punto flotante, con este resultado:

SAM	FITM
15015	132

Caso general

Para un sistema borroso con $k + 1$ entradas y una salida, se puede aproximar el número de operaciones de una inferencia SAM por $O_p^{SAM} \approx 2 \times R \times [\sum_i P_i]$, siendo P_i el número de puntos de resolución de cada espacio de entrada. Se asumirá por simplicidad que $P_1 = P_2 = \cdots = P_{k+1} = P_p$, con lo que $O_p^{SAM} \approx 2 \times R \times (k + 1) \times P_p$.

Para el caso FITM el número de operaciones asociadas (ecuación (4.34)) es aproximadamente $O_p^{FITM} \approx 2 \times L \times \prod_{i=1}^{k+1} N_i$, siendo N_i el número de conjuntos de cada VL. Si se hace que $R = \prod_{i=1}^{k+1} N_i$ puede escribirse $O_p^{FITM} \approx 2 \times R \times L$.

SAM	FITM
$O(R \times k \times P_p)$	$O(R \times L)$

De nuevo se puede observar que P_p en las aplicaciones reales va a ser siempre mucho mayor que L, con lo que el número de operaciones de la inferencia tradicional será mucho mayor que el correspondiente al uso de matrices de transición.

4.3 EXTENSIÓN A SISTEMAS NO-LINEALES

Todo lo expuesto hasta ahora es exclusivo para sistemas que usen operativa lineal SAM [72]. El paso siguiente consiste en extender esta operativa matricial a otros sistemas borrosos que puedan usar operadores no lineales (como *max-min*) para llevar a cabo la inferencia.

BIBLIOTECA VIRTUAL

En el apéndice B se realiza una extensión de la operativa matricial vista hasta ahora para los operadores unión e intersección genéricos. Los desarrollos corresponden al caso de un sistema con una entrada y una salida, y al de un sistema de dos entradas y una salida. Estos resultados son aplicables a t-normas y t-conormas genéricas si éstas cumplen las propiedades conmutativa, asociativa y distributiva, y si se usan las mismas normas para el cálculo de las cuatro operaciones básicas relacionadas con la inferencia (altura, truncado, unión e intersección, sec. 2.2.3).

Definición vectorial de las entradas

Al igual que se ha hecho para el SAM (sección 4.2.2), es posible definir las entradas a un sistema borroso genérico de forma vectorial. De acuerdo con lo visto en la sección 2.2.3, para un número genérico de entradas y para una t-norma y una t-conorma que cumplan las propiedades antes mencionadas, la salida se puede escribir como

$$\text{Salida} = \bigcup_j \left[(dc_i)_j \bigcap D_{r(j)} \right] \tag{4.41}$$

con $D_{r(j)}$ la salida borrosa de la regla j-ésima, $r(j)$ una función de indexado para la que $1 \leq r(j) \leq L$ si la variable de salida consiste en L conjuntos borrosos, y

$$(dc_i)_j = \bigcap_i (dc_j(Y_i))$$

el grado de consistencia para la regla j-ésima, formado por la intersección entre los grados de consistencia de todos los antecedentes de la regla para las entradas Y_i.

En aplicaciones de CWW va a poderse restringir el vocabulario de entrada a los valores de la VL definida en ese espacio. Consecuentemente, una entrada puede definirse

$$X = \bigcup_{k=1}^{M} \left(\beta_k \bigcap A_k \right) \tag{4.42}$$

donde A_k son los distintos conjuntos que forman la variable lingüística, y β_j un coeficiente que vale 1 si la entrada es A_j y cero para el resto.

De acuerdo con la ecuación (4.41) la salida de un sistema borroso es una combinación (basada en normas) de los conjuntos de salida. Si esta salida es a su vez la entrada de otro sistema (tal y como aparece en la Fig. 4.8, sustituyendo los operadores suma y producto por las respectivas normas), esta entrada no será un solo conjunto borroso, sino una combinación de conjuntos borrosos, en cuyo caso los coeficientes β_k ya no serán necesariamente binarios.

De este modo, un conjunto borroso puede verse como un vector de coeficientes, esto es

$$X = [\beta_1 \ \beta_2 \ \cdots \ \beta_M]^T$$

Este modo de representar las entradas se puede utilizar tanto para conjuntos definidos de forma granular, como para conjuntos definidos sobre un eje numérico.

4.3.1 Matrices de transición para un sistema genérico

A partir de los resultados del apéndice B, se va a plantear la algorítmica genérica que va a permitir realizar una inferencia borrosa usando matrices de transición, usando t-normas y t-conormas genéricas, siempre que cumplan los requisitos ya expuestos.

Se va a ilustrar el método para un sistema con dos entradas ($X \subset U$ e $Y \subset V$) y de una salida ($Z \subset W$), todas ellas conjuntos borrosos pertenecientes a sus respectivas variables lingüísticas (la extensión a sistemas con más entradas es prácticamente inmediata). Se pueden escribir de forma vectorial como

$$\begin{aligned} X &= \bigcup_{k=1}^{M} \beta_k \bigcap A_k \\ Y &= \bigcup_{l=1}^{N} \alpha_l \bigcap B_l \\ Z &= \bigcup_{n=1}^{L} \gamma_n \bigcap C_n \end{aligned}$$

El proceso a seguir a continuación es similar a lo visto para el caso SAM:

1. Se definen las matrices $\mathbf{R_A} = [A_1 \dots A_M]^T \circ [A_1 \dots A_M]$ y $\mathbf{R_B} = [B_1 \dots B_N]^T \circ [B_1 \dots B_N]$, siendo ahora el operador $\circ$ la composición basada en las normas elegidas (definición 16, apéndice B.1). Nótese que esta operación es la única que depende de la resolución del universo de discurso en el que las variables lingüísticas están definidas.
2. Se generan las matrices $\mathbf{G}_l$:

$$\mathbf{G}_l = \mathbf{R_A} \otimes_{\cap} [\mathbf{R_B} \mathbf{E}_l] \quad l = 1, ...N \tag{4.43}$$

siendo $\otimes_{\cap}$ la intersección tensorial de Kronecker (definición 22 apéndice B.1), y $\mathbf{E}_l$ un vector columna que extrae la columna l de la matriz $\mathbf{R_B}$, tal como se vio en el caso SAM.

3. Se define la matriz de reglas $\mathbf{C}$ tal como se hizo para el caso lineal.

4. Se calculan las matrices de transición:

$$\mathbf{\Omega}_l = \mathbf{C}^T \times_{\cup\cap} \mathbf{G}_l \qquad l = 1, 2, \cdots, N \tag{4.44}$$

con $\times_{\cup\cap}$ definida en B.1. La inferencia se lleva ahora a cabo mediante la siguiente expresión

$$\begin{bmatrix} \gamma_1 \\ \gamma_2 \\ \cdots \\ \gamma_L \end{bmatrix} = \left(\bigcup_{l=1}^{N} \left(\alpha_l \bigcap \mathbf{\Omega}_l \right) \right) \times_{\cup\cap} \begin{bmatrix} \beta_1 \\ \beta_2 \\ \cdots \\ \beta_M \end{bmatrix} \tag{4.45}$$

4.4 SISTEMAS BORROSOS REALIMENTADOS (FFS)

Un caso particular dentro de los sistemas borrosos que pueden ser implementados mediante matrices de transición es el de los sistemas borrosos realimentados [3]. A pesar de que la algorítmica usada en estos sistemas es muy similar a la usada en los sistemas no realimentados, las propiedades que de ellos se derivan son suficientemente interesantes como para justificar un estudio especial.

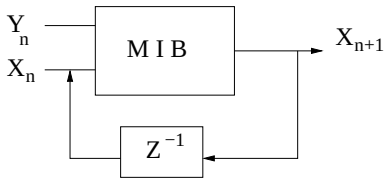


Figura 4.13 FFS de dos entradas y una salida.

Como punto de partida considérese un sistema realimentado genérico con una entrada exterior y una entrada realimentada desde la salida (Fig. 4.13). Tanto las entradas como las salidas son conjuntos borrosos. Dichos conjuntos pueden estar definidos sobre un eje numérico o granularmente.

En caso de estar definidos sobre un eje numérico, se puede definir el sistema como una transformación $f : U \times V \subset R^2 \mapsto U \subset R$. $X_n \subset U$ e $Y \subset V$ denotan las variables de entrada y $X_{n+1} \subset U$ denota la variable de salida. El subíndice en la primera entrada enfatiza la naturaleza recursiva del FFS.

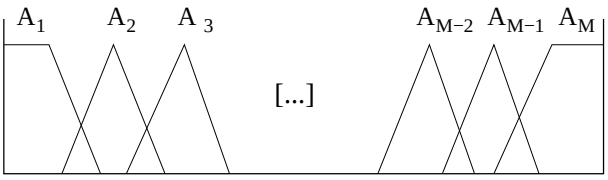


Figura 4.14 Conjuntos borrosos de la variable lingüística $\mathcal{L}$

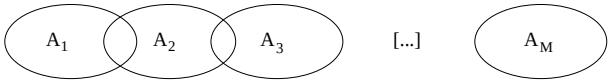


Figura 4.15 Gránulos borrosos de la variable lingüística $\mathcal{L}$

BIBLIOTECA VIRTUAL

4.4.1 Matrices de transición para un sistema borroso realimentado

El desarrollo de las matrices de transición para un sistema realimentado se basa en los desarrollos planteados en la sección 4.2.4, pero en este caso, la segunda entrada en el instante n se corresponde con la salida del sistema en el instante anterior.

Considérese que la entrada realimentada y la salida toman valores sobre una variable lingüística $\mathcal{L}$, formada por M conjuntos borrosos A_i en U. La otra entrada toma valores sobre una VL formada por N conjuntos B_j. Estos conjuntos se pueden definir sobre un eje numérico, usando funciones consistentes y normales, tal como se muestra en la figura 4.14, o bien mediante gránulos borrosos, tal como se muestra en la figura 4.15.

Si se trabaja con un sistema SAM que usa operadores lineales, X_n e Y se pueden escribir como una combinación lineal de los conjuntos A_i y B_j respectivamente:

$$X_n = \beta_1^n A_1 + \beta_2^n A_2 + ... + \beta_M^n A_M \tag{4.46}$$

$$Y = \alpha_1 B_1 + \alpha_2 B_2 + ... + \alpha_N B_N \tag{4.47}$$

con β_i^n y α_i coeficientes que dependen del método de activación de reglas. La salida X_{n+1} puede también definirse sobre U, y escribirse:

$$X_{n+1} = \beta_1^{n+1} A_1 + \beta_2^{n+1} A_2 + ... + \beta_M^{n+1} A_M \tag{4.48}$$

Para unas reglas genéricas de la forma

$$R_{ij}: \quad \text{Si } X_n \text{ es } A_i \quad \text{e } Y \text{ es } B_j \quad \text{entonces } X_{n+1} \text{ es } A_{p(ij)}$$

($i = 1, \cdots, M$, $j = 1, \cdots, N$) se puede definir una base de reglas como la de la tabla 4.2 ($A_{p(ij)}$ toma valores dentro de la variable $\mathcal{L}$).

R_{ij}	A_i	B_j	$A_{p(ij)}$
R_{11}	A_1	B_1	$A_{p(11)}$
R_{12}	A_1	B_2	$A_{p(12)}$
$\cdots$	$\cdots$	$\cdots$	$\cdots$
R_{MN-1}	A_M	B_{N-1}	$A_{p(MN-1)}$
R_{MN}	A_M	B_N	$A_{p(MN)}$

Tabla 4.2 Base de reglas genérica para un FFS

La creación de las matrices de transición para el FFS es similar a la realizada para el caso no realimentado. En primer lugar se definen las matrices de activación correspondientes a las dos entradas:

$$\mathbf{R_A}^{X_n} = \begin{pmatrix} A_1 \circ A_1 & \cdots & A_1 \circ A_M \\ A_2 \circ A_1 & \cdots & A_2 \circ A_M \\ \vdots & \cdots & \vdots \\ A_M \circ A_1 & \cdots & A_M \circ A_M \end{pmatrix} = [A_1 \cdots A_M]^T \circ [A_1 ... A_M] \tag{4.49}$$

$$\mathbf{R_B}^{Y_n} = \begin{pmatrix} B_1 \circ B_1 & \cdots & B_1 \circ B_N \\ B_2 \circ B_1 & \cdots & B_2 \circ B_N \\ \vdots & \cdots & \vdots \\ B_N \circ B_1 & \cdots & B_N \circ B_N \end{pmatrix} = [B_1 \cdots B_N]^T \circ [B_1 ... B_N] \tag{4.50}$$

siendo $\mathbf{R_A}^{X_n}$ una matriz de dimensiones $M \times M$, y $\mathbf{R_B}^{Y}$ una matriz de dimensiones $N \times N$. El siguiente paso es definir un conjunto de matrices $\mathbf{G}_l$ (de tamaño $MN \times M$):

$$\mathbf{G}_l = \mathbf{R_A}^{X_n} \otimes_{\mathcal{K}} \left[\mathbf{R_B}^{Y} \mathbf{E}_l\right] \quad l = 1, ...N \tag{4.51}$$

y la matriz de selección de consecuente $\mathbf{C}$. Se va a trabajar suponiendo que se cuenta con una base de reglas completa y ordenada según se representa en la tabla 4.2.

$$\mathbf{C} = \begin{array}{c} \\ R_1 \\ R_2 \\ \vdots \\ R_{M\times N-1} \\ R_{M\times N} \end{array} \begin{array}{c} \begin{array}{ccccccc} A_1 & A_2 & \cdots & A_k & \cdots & \cdots & A_M \end{array} \\ \begin{pmatrix} 1 & 0 & \cdots & \cdots & \cdots & \cdots & 0 \\ 0 & 1 & 0 & \cdots & \cdots & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & \cdots & \cdots & \cdots & 1 & 0 & 0 \\ 0 & \cdots & \cdots & \cdots & \cdots & \cdots & 1 \end{pmatrix} \end{array} \tag{4.52}$$

El último paso es calcular la matriz de transición $\mathbf{\Omega}_l$, esto es, la relación entre los coeficientes de entrada (los pertenecientes a X_n e Y, ecuaciones (4.46) y (4.47)) y los coeficientes de salida (los de X_{n+1}, ecuación 4.48) cuando la entrada Y es B_l. La matriz se calcula

$$\mathbf{\Omega}_l = \mathbf{C}^T \mathbf{G}_l \qquad l = 1, 2, \cdots, N \tag{4.53}$$

Con este conjunto de matrices, la inferencia pasa a ser una operativa de suma y multiplicación de matrices. El conjunto de salida X_{n+1} con coeficientes β_k^{n+1} se calcula como sigue:

$$\begin{bmatrix} \beta_1^{n+1} \\ \beta_1^{n+1} \\ \vdots \\ \beta_M^{n+1} \end{bmatrix} = \left(\sum_{l=1}^{N} \alpha_l \mathbf{\Omega}_l \right) \begin{bmatrix} \beta_1^{n} \\ \beta_2^{n} \\ \vdots \\ \beta_M^{n} \end{bmatrix} \tag{4.54}$$

Estas operaciones son independientes de la resolución del universo de discurso y de la forma de los conjuntos allí definidos. Esta resolución (incluso considerando un universo continuo) sólo es importante a la hora de determinar las matrices $\mathbf{R_A}$ y $\mathbf{R_B}$, que pueden ser precalculadas y almacenadas.

4.4.2 Análisis de un sistema realimentado en régimen estacionario

El uso de matrices de transición en los sistemas borrosos realimentados simplifica el estudio de éstos, particularmente la convergencia en régimen estacionario. En la presente sección se estudiará el comportamiento de un sistema borroso con una entrada realimentada cuando la otra entrada permanece constante a un valor determinado.

Tal y como se ha visto en la sección anterior, un SAM realimentado se puede expresar de forma matricial del siguiente modo

$$\begin{bmatrix} \beta_1^{n+1} \\ \beta_2^{n+1} \\ \vdots \\ \beta_M^{n+1} \end{bmatrix} = \left(\sum_{l=1}^{N} \alpha_l \mathbf{\Omega}_l \right) \begin{bmatrix} \beta_1^{n} \\ \beta_2^{n} \\ \vdots \\ \beta_M^{n} \end{bmatrix}$$

Si se mantiene constante la entrada no realimentada, los coeficientes α_i serán constantes en el tiempo, con lo que la expresión

$$\sum_{l=1}^{N} \alpha_l \mathbf{\Omega}_l$$

dará lugar a una matriz $M \times M$ constante:

$$\mathbf{\Omega} = \sum_{l=1}^{N} \alpha_l \mathbf{\Omega}_l \tag{4.55}$$

El sistema en régimen estacionario estará representado por

$$\boldsymbol{\beta}^{n+1} = \mathbf{\Omega} \boldsymbol{\beta}^{n} \tag{4.56}$$

Dado que el sistema es lineal se cumple que

$$\begin{aligned}\boldsymbol{\beta}^k &= \boldsymbol{\Omega}\boldsymbol{\beta}^{k-1} \\ &= \boldsymbol{\Omega}^2\boldsymbol{\beta}^{k-2} \\ &\cdots \\ &= \boldsymbol{\Omega}^{k-1}\boldsymbol{\beta}^1 \\ &= \boldsymbol{\Omega}^k\boldsymbol{\beta}^0\end{aligned} \tag{4.57}$$

Por lo tanto, en estado estacionario es posible relacionar la salida en un instante arbitrario de tiempo con la entrada en el instante inicial.

La matriz $\boldsymbol{\Omega}$ va a ser diagonalizable, ya que es la matriz de un sistema para el que siempre existe solución [3, 41], por lo que se puede escribir

$$\boldsymbol{\Omega}^k = \mathbf{B}\boldsymbol{\Lambda}^k\mathbf{B}^{-1} \tag{4.58}$$

donde $\boldsymbol{\Lambda}$ es una matriz diagonal cuyos elementos son los autovalores de $\boldsymbol{\Omega}$ y $\mathbf{B}$ es su matriz de autovectores asociados. La expresión (4.57) se convierte en

$$\boldsymbol{\beta}^k = \mathbf{B}\boldsymbol{\Lambda}^k\mathbf{B}^{-1}\boldsymbol{\beta}^0 \tag{4.59}$$

El centroide de salida puede ser calculado fácilmente a partir de la expresión

$$C_k = \frac{\left[\begin{array}{cccc} c_1 & \cdots & c_{M-1} & c_M \end{array}\right]\mathbf{B}\boldsymbol{\Lambda}^k\mathbf{B}^{-1}\left[\begin{array}{ccc} \beta_1^0 & \beta_2^0 & \cdots \beta_M^0 \end{array}\right]^T}{\left[\begin{array}{cccc} 1 & \cdots & 1 & 1 \end{array}\right]\mathbf{B}\boldsymbol{\Lambda}^k\mathbf{B}^{-1}\left[\begin{array}{ccc} \beta_1^0 & \beta_2^0 & \cdots \beta_M^0 \end{array}\right]^T} \tag{4.60}$$

donde los c_i son los centroides de los conjuntos A_i. El límite de esta expresión cuando k tiende a infinito, dará el comportamiento en régimen estacionario del sistema. Si se asume que el máximo autovalor de la matriz $\boldsymbol{\Omega}$ se encuentra en la fila h-ésima de la matriz $\boldsymbol{\Lambda}$ (esto es, $|\lambda_{MAX}| = |\lambda_h| > |\lambda_p|, \forall p \neq h$) se puede escribir

$$\begin{aligned}\boldsymbol{\Omega}^k = &\begin{pmatrix} b_{11} & b_{12} & \cdots & b_{1M} \\ b_{21} & b_{22} & \cdots & b_{2M} \\ \vdots & \vdots & \ddots & \vdots \\ b_{M1} & b_{M2} & \cdots & b_{MM} \end{pmatrix} \lambda_h^k \begin{pmatrix} \left(\frac{\lambda_1}{\lambda_h}\right)^k & & & & \\ & \ddots & & & \\ & & 1 & & \\ & & & \ddots & \\ & & & & \left(\frac{\lambda_m}{\lambda_h}\right)^k \end{pmatrix} \\ &\cdot \begin{pmatrix} b'_{11} & b'_{12} & \cdots & b'_{1M} \\ b'_{21} & b'_{22} & \cdots & b'_{2M} \\ \vdots & \vdots & \ddots & \vdots \\ b'_{M1} & b'_{M2} & \cdots & b'_{MM} \end{pmatrix}\end{aligned} \tag{4.61}$$

Cuando k tiende a infinito, todos los elementos de la diagonal desaparecen excepto el correspondiente a la fila h-ésima, que vale la unidad. De este modo, el centroide de salida

resulta ser

$$C_k = \frac{\begin{bmatrix} c_1 & \cdots & c_{M-1} & c_M \end{bmatrix} \lambda_h^k \begin{bmatrix} b_{1h} \sum_{i=1}^{M} b'_{hi} \beta_i^0 \\ b_{2h} \sum_{i=1}^{M} b'_{hi} \beta_i^0 \\ \vdots \\ b_{Mh} \sum_{i=1}^{M} b'_{hi} \beta_i^0 \end{bmatrix}}{\begin{bmatrix} 1 & \cdots & 1 & 1 \end{bmatrix} \lambda_h^k \begin{bmatrix} b_{1h} \sum_{i=1}^{M} b'_{hi} \beta_i^0 \\ b_{2h} \sum_{i=1}^{M} b'_{hi} \beta_i^0 \\ \vdots \\ b_{Mh} \sum_{i=1}^{M} b'_{hi} \beta_i^0 \end{bmatrix}} \tag{4.62}$$

esto es,

$$C_k = \frac{\sum_{j=1}^{M} c_j b_{jh}}{\sum_{j=1}^{M} b_{jh}} \tag{4.63}$$

con c_j los centroides de los conjuntos borrosos del espacio de salida, y $b_h = b_{MAX} = [b_{1h} \cdots b_{Mh}]^T$ el autovector asociado a λ_{MAX}. De este resultado se pueden extraer varias conclusiones:

- El valor de convergencia del sistema para entrada estacionaria es independiente del estado inicial del sistema, y sólo depende del valor de la entrada (α_l) y de la base de reglas.
- El valor de convergencia es la suma promediada de los centroides de los conjuntos del espacio de salida. Los pesos se corresponden con el autovector asociado al autovalor máximo de la matriz de transición.

La ecuación (4.63) es en realidad la operación que calcula el centroide de un determinado conjunto borroso. Se puede por lo tanto asociar dicho centroide con un conjunto borroso de salida, que tendría la forma $H_\infty = \sum_j b_{MAXj} A_j$. Consecuentemente, el conjunto borroso de salida con entrada estacionaria será

$$H_\infty = \beta_1^\infty A_1 + \beta_2^\infty A_2 + \ldots + \beta_M^\infty A_M \tag{4.64}$$

con β_j^∞ las componentes del autovector asociado al autovalor máximo de la matriz del sistema $\mathbf{\Omega}$.

Como conclusión, se puede decir que es posible conocer el estado final de un sistema borroso realimentado con entrada estacionaria, estudiando los autovalores y autovectores de la matriz de transición. Esto es de enorme valor en determinadas ocasiones. Por ejemplo, en la aplicación TextTrack [3, 7] que se estudia en el capítulo 5, se utilizó para predecir el valor final hacia el que tiende un sistema que actualiza etiquetas de probabilidad cuando la entrada se fija al valor máximo. En [3] se estudia el valor de convergencia para diversos métodos de activación, observando que para la misma base de reglas algunos métodos tienden con el tiempo hacia valores intermedios, disminuyendo así la precisión del sistema.

4.4.3 Generalización de los resultados

Supóngase un sistema borroso realimentado con $F+1$ entradas borrosas, de las cuales F son exteriores y 1 es la realimentada. En el instante n las entradas del sistema son

$$\begin{aligned} Y_1 &= \{\alpha_1^1, \cdots, \alpha_{N_1}^1\} \\ Y_2 &= \{\alpha_1^2, \cdots, \alpha_{N_2}^2\} \\ \vdots &\quad \vdots \\ Y_F &= \{\alpha_1^K, \cdots, \alpha_{N_K}^F\} \\ X_n &= \{\beta_1^n, \cdots, \beta_M^n\} \end{aligned}$$

Se define una inferencia matricial de la forma

$$\begin{bmatrix} \beta_1^{n+1} \\ \beta_2^{n+1} \\ \vdots \\ \beta_M^{n+1} \end{bmatrix} = \left(\sum_{i_1=1}^{N_1} \sum_{i_2=1}^{N_2} \sum_{i_3=1}^{N_3} \cdots \sum_{i_F=1}^{N_F} \alpha_{i_1}^1 \alpha_{i_2}^2 \alpha_{i_3}^3 \cdots \alpha_{i_F}^F \mathbf{\Omega}_{\prod_{j=1}^F i_j} \right) \begin{bmatrix} \beta_1^n \\ \beta_2^n \\ \vdots \\ \beta_M^n \end{bmatrix} \tag{4.65}$$

Si se mantienen constantes las entradas Y_i, la salida evolucionará hacia un estado estacionario que se corresponde con el autovector asociado al autovalor máximo de la matriz

$$\mathbf{\Omega} = \left(\sum_{i_1=1}^{N_1} \sum_{i_2=1}^{N_2} \sum_{i_3=1}^{N_3} \cdots \sum_{i_F=1}^{N_F} \alpha_{i_1}^1 \alpha_{i_2}^2 \alpha_{i_3}^3 \cdots \alpha_{i_F}^F \mathbf{\Omega}_{\prod_{j=1}^F i_j} \right) \tag{4.66}$$

independientemente de cuál fuera el estado inicial (es decir, el valor de X_0).

Ejemplo de aplicación

Supóngase una aplicación borrosa de filtrado recursivo de imagen, como por ejemplo la citada en [4]. Allí se realiza una difusión anisótropa basada en reglas borrosas. Para el filtrado correspondiente a la difusión en la dirección *Este* los pixels que entran al decisor son los que se encuentran en la figura 4.16 (P_3, P_6, P_9 y el pixel central P_5).

El sistema borroso al que da lugar es el que se muestra en la figura 4.17. El pixel central entra al motor de inferencias y es a la vez la salida de éste. Si el filtrado es recursivo y en un momento dado los pixels del entorno (es decir, P_3, P_6 y P_9) tienen valores estables, se podrá usar el valor del límite (dado por el autovalor asociado al autovalor máximo de la matriz de transición) como criterio de parada del filtrado. Aunque en una aplicación de imagen es difícil que los valores de los píxeles permanezcan estables, si dichos valores varían poco, y son borrosificados antes de entrar al sistema, la suposición de estabilidad sería aceptable.

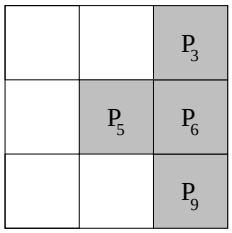


Figura 4.16 Pixels de entrada

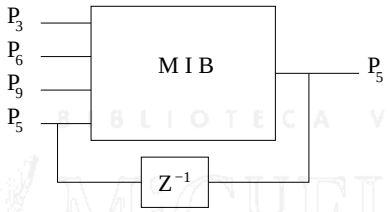


Figura 4.17 Sistema borroso realimentado

Por otro lado, el conocimiento del valor límite ante diferentes entradas, puede ser de gran utilidad a la hora de definir la base de reglas que gobierne el filtrado.

4.4.4 Inferencia recursiva

Por extensión de lo visto anteriormente, se puede concluir que si en un FFS se mantiene la entrada no realimentada constante, y se realizan k inferencias, esto es equivalente a

$$\begin{bmatrix} \beta_1^{n+k} \\ \beta_2^{n+k} \\ \vdots \\ \beta_M^{n+k} \end{bmatrix} = \left(\sum_{l=1}^{N} \alpha_l \mathbf{\Omega}_l \right)^k \begin{bmatrix} \beta_1^{n} \\ \beta_2^{n} \\ \vdots \\ \beta_M^{n} \end{bmatrix} \tag{4.67}$$

4.4.5 Sistema realimentado no lineal

Partiendo de un sistema borroso realimentado como el visto en la sección 4.4.2 se estudiará a continuación su comportamiento cuando se utilizan los t-normas y t-conormas genéricas. Un sistema de estas características realizará una inferencia matricial del tipo

$$\begin{bmatrix} \beta_1^{n+1} \\ \beta_2^{n+1} \\ \vdots \\ \beta_M^{n+1} \end{bmatrix} = \left(\bigcup_l (\alpha_l \bigcap \mathbf{\Omega}_l) \right) \times_{\cup\cap} \begin{bmatrix} \beta_1^{n} \\ \beta_2^{n} \\ \vdots \\ \beta_M^{n} \end{bmatrix} \tag{4.68}$$

Si la entrada al sistema permanece constante, tal como se veía en la sección 4.4.2 se escribirá una matriz $\mathbf{\Omega} = (\bigcup_l (\alpha_l \bigcap \mathbf{\Omega}_l))$ de tal modo que el sistema ahora queda

$$\begin{bmatrix} \beta_1^{n+1} \\ \beta_2^{n+1} \\ \vdots \\ \beta_M^{n+1} \end{bmatrix} = \mathbf{\Omega} \times_{\cup\cap} \begin{bmatrix} \beta_1^{n} \\ \beta_2^{n} \\ \vdots \\ \beta_M^{n} \end{bmatrix}$$

que en modo compacto puede escribirse

$$\boldsymbol{\beta}^{n+1} = \mathbf{\Omega} \times_{\cup\cap} \boldsymbol{\beta}^{n} \tag{4.69}$$

Se puede definir la salida en el instante arbitrario k en función del instante inicial como

$$\boldsymbol{\beta}^{k} = \mathbf{\Omega}^{k} \times_{\cup\cap} \boldsymbol{\beta}^{0} \tag{4.70}$$

entendiendo

$$\mathbf{\Omega}^{k} = \mathbf{\Omega} \times_{\cup\cap} \mathbf{\Omega} \times_{\cup\cap} \cdots \times_{\cup\cap} \mathbf{\Omega} \tag{4.71}$$

Experimentalmente, para los operadores máximo y mínimo se ha podido comprobar que la operación $\mathbf{\Omega}^k$ en la mayoría de los casos llega a un estado estacionario para un valor bajo de k. Para matrices 3×3 se ha observado que es frecuente que este valor sea $k = 2$. (Bajo determinadas condiciones, se han encontrado matrices en las que el estado final no es estacionario, sino que oscila entre dos valores).

Por ello, a diferencia del sistema descrito en la sección 4.4.2, en el que el sistema tiende a un valor estable asintóticamente, cuando se cambia la operativa SAM por *max-min* el sistema suele llegar a un estado estacionario en pocas iteraciones. De este modo, el estado final se puede calcular como

$$\begin{bmatrix} \beta_1^{\infty} \\ \beta_2^{\infty} \\ \vdots \\ \beta_M^{\infty} \end{bmatrix} = \mathbf{\Omega}^{K} \times_{\cup\cap} \begin{bmatrix} \beta_1^{0} \\ \beta_2^{0} \\ \vdots \\ \beta_M^{0} \end{bmatrix} \tag{4.72}$$

siendo K el valor de k a partir del cual $\mathbf{\Omega}^k = \mathbf{\Omega}^{k+1}$. Se puede observar que en este caso, al contrario que ocurre con el SAM, el estado final sí depende del estado inicial en que se encuentre el sistema.

4.5 REGLAS DE INFERENCIA Y COMPOSICIÓN DE MATRICES

En la sección 3.7.3 se exponen las reglas que gobiernan la inferencia borrosa (o en ese caso la propagación de restricciones borrosas [171]). Esas reglas están enunciadas para cada relación borrosa individual. Si se usa el método matricial anteriormente expuesto, las distintas relaciones borrosas pueden ser consideradas en su conjunto, y no una a una, por lo que es necesario redefinir dichas propiedades para el nuevo método de inferencia.

De este modo, en vez de trabajar con relaciones de la forma

$$X \text{ es } A$$

se va a trabajar con un conjunto de relaciones de la forma

$$X \text{ es } A_i \qquad i = 1, \ldots, M$$

Como punto de partida, considérese una relación borrosa como la anterior, X es A_i, $i = 1, \ldots, M$. En forma matricial, para todos los posibles valores de i, queda representada por la matriz $\mathbf{R_A}$. Para un determinado valor de X, la relación queda explicitada por $\mathbf{R_A} \times_{\cup\cap} X$. A partir de una relación en forma matricial, puede pasarse fácilmente a una proposición de la forma

$$\text{Si } [X \text{ es } A_i] \text{ entonces } Y \text{ es } B_{p(i)} \qquad i = 1, \ldots, M$$

mediante el uso de una matriz de selección de consecuente $\mathbf{C}$:

$$\mathbf{C}^T \times_{\cup\cap} \mathbf{R_A}$$

En los desarrollos siguientes se hará uso de los operadores $\times_{\cup\cap}$, $\otimes_{\cap}$ (intersección tensorial de Kronecker) y $\otimes_{\cup}$ (unión tensorial de Kronecker), definidos en el apéndice B.1.

Las principales reglas de inferencia y su composición matricial son las siguientes:

1. *Regla Conjuntiva 1:* ($X \subset U$, $A_i \subset U$)

$$\frac{\begin{array}{c} X \text{ es } A_j \\ X \text{ es } A_l \end{array}}{X \text{ es } A_j \cap A_l}$$

Esta regla afecta a un conjunto de relaciones del tipo

$$X \text{ es } A_j \quad \text{y} \quad X \text{ es } A_l$$

con $\{j,l\} = 1,\ldots,M$. Los conjuntos A_i $(i = 1,\ldots,M)$ se definen en función de su matriz de activación $\mathbf{R_A}$. El conjunto de salida $A_j \cap A_l$ queda definido por las matrices $\mathbf{G}_l$ de la siguiente forma:

$$\mathbf{G}_l(X) = \mathbf{R}_{(\mathbf{A}\cap\mathbf{A})} = \mathbf{R_A} \otimes_\cap [\mathbf{R_A}\mathbf{E}_l] \quad l = 1,\ldots,M$$

Las matrices $\mathbf{G}_l$ tendrán un tamaño $M^2 \times M$.

2. *Regla Conjuntiva 2:* $(X \subset U,\, Y \subset V,\, A_i \subset U,\, B_l \subset V)$

$$\frac{\begin{array}{c} X \text{ es } A_i \\ Y \text{ es } B_l \end{array}}{(X,Y) \text{ es } A_i \times B_l}$$

Los conjuntos A_i $(i = 1,\ldots,M)$ y B_l $(l = 1,\ldots,N)$ se definen en función de sus matrices de activación $\mathbf{R_A}$ y $\mathbf{R_B}$. El conjunto de salida $A_i \times B_l$ queda definido por las matrices $\mathbf{G}_l$ de la siguiente forma:

$$\mathbf{G}_l(X,Y) = \mathbf{R}_{(\mathbf{A}\times\mathbf{B})} = \mathbf{R_A} \otimes_\cap [\mathbf{R_B}\mathbf{E}_l] \quad l = 1,\ldots,N$$

Las matrices $\mathbf{G}_l$ tendrán un tamaño $MN \times M$.

Ejemplo: Supóngase un conjunto de M relaciones borrosas que vienen dadas por la expresión

$$X \text{ es } A_i \qquad i = 1,\ldots,M$$

Como ya se ha comentado, esta relación está representada por la matriz

$$\mathbf{R_A} = [A_1 \ldots A_M]^T \circ [A_1 \ldots A_M]$$

A su vez, existe otro conjunto de N relaciones borrosas según la expresión

$$Y \text{ es } B_j \qquad j = 1,\ldots,N$$

que están representadas por la matriz $\mathbf{R_B}$. Puede formarse un nuevo conjunto de $M \times N$ relaciones a partir de la conjunción de los dos anteriores, de la forma

$$X \text{ es } A_i \quad \text{e} \quad Y \text{ es } B_j$$

La conjunción 'y'('e' en la relación anterior por motivos fonéticos) en operativa matricial se lleva a cabo mediante la intersección de Kronecker. Las matrices correspondientes a las nuevas relaciones se calcularán

$$\mathbf{G}_l(X,Y) = \mathbf{R}_{(\mathbf{A}\times\mathbf{B})} = \mathbf{R_A} \otimes_\cap [\mathbf{R_B}\mathbf{E}_l] \quad l = 1,\ldots,N$$

Si a partir de esta relación se plantea un conjunto de proposiciones borrosas de la forma

$$\text{Si } [X \text{ es } A_i \text{ y si } Y \text{ es } B_l] \text{ entonces } Z \text{ es } C_{p(i,l)}$$

será necesario definir una matriz $\mathbf{C}$ que seleccione el consecuente $C_{p(i,l)}$ para cada regla. De este modo se crean las matrices $\mathbf{\Omega}_l$:

$$\mathbf{\Omega}_l = \mathbf{C}^T \times_{\cup\cap} \mathbf{G}_l(X,Y)$$

3. *Regla Disyuntiva 1:* $(X \subset U,\ A_i \subset U)$

$$\frac{\begin{array}{c} X \text{ es } A_j \\ \text{ó} \\ X \text{ es } A_l \end{array}}{X \text{ es } A_j \cup A_l}$$

Esta regla afecta a un conjunto de relaciones del tipo

$$X \text{ es } A_j \quad \text{ó} \quad X \text{ es } A_l$$

con $\{j,l\} = 1,\ldots,M$.

Los conjuntos A_i $(i = 1,\ldots,M)$ se definen en función de su matriz de activación $\mathbf{R_A}$. El conjunto de salida $A_j \cup A_l$ queda definido por las matrices $\mathbf{G}_l$ de la siguiente forma:

$$\mathbf{G}_l(X) = \mathbf{R}_{(\mathbf{A}\cup\mathbf{A})} = \mathbf{R_A} \otimes_{\cup} [\mathbf{R_A}\mathbf{E}_l] \quad l = 1,\ldots,M$$

Las matrices $\mathbf{G}_l$ tendrán un tamaño $M^2 \times M$.

4. *Regla Disyuntiva 2:* $(X \subset U,\ Y \subset V,\ A_i \subset U,\ B_j \subset V)$

$$\frac{\begin{array}{c} X \text{ es } A_i \\ \text{ó} \\ Y \text{ es } B_l \end{array}}{(X,Y) \text{ es } A_i \times V \cup U \times B_l}$$

Los conjuntos A_i $(i = 1,\ldots,M)$ y B_l $(l = 1,\ldots,N)$ se definen en función de sus matrices de activación $\mathbf{R_A}$ y $\mathbf{R_B}$. El conjunto de salida $A_i \times V \cup U \times B_l$ queda definido por las matrices $\mathbf{G}_l$ de la siguiente forma:

$$\mathbf{G}_l(X,Y) = \mathbf{R}_{(\mathbf{A}\times V\cup U\times\mathbf{B})} = \mathbf{R_A} \otimes_{\cup} [\mathbf{R_B}\mathbf{E}_l] \quad l = 1,\ldots,N$$

Las matrices $\mathbf{G}_l$ tendrán un tamaño $MN \times M$.

Ejemplo: Supóngase un conjunto de M relaciones borrosas que vienen dadas por la expresión

$$X \text{ es } A_i \qquad i = 1,\ldots,M$$

representada por la matriz $\mathbf{R_A}$. A su vez, existe otro conjunto de N relaciones borrosas según la expresión

$$Y \text{ es } B_l \qquad l = 1,\ldots,N$$

que están representadas por la matriz $\mathbf{R_B}$. Puede formarse un nuevo conjunto de $M \times N$ relaciones a partir de la conjunción de los dos anteriores, de la forma

$$X \text{ es } A_i \quad \text{ó} \quad Y \text{ es } B_l$$

La disyunción 'o' en operativa matricial se lleva a cabo mediante la unión de Kronecker. Las matrices correspondientes a las nuevas relaciones se calcularán

$$\mathbf{G}_l(X,Y) = \mathbf{R}_{(\mathbf{A}\times\mathbf{B})} = \mathbf{R_A} \otimes_{\cup} [\mathbf{R_B}\mathbf{E}_l] \;\; l = 1, \ldots, N$$

Si a partir de esta relación se plantea un conjunto de proposiciones borrosas de la forma

$$\text{Si } [X \text{ es } A_i \text{ ó si } Y \text{ es } B_l] \text{ entonces } Z \text{ es } C_{p(i,l)}$$

será necesario definir una matriz $\mathbf{C}$ que seleccione el consecuente $C_{p(i,l)}$ para cada regla. De este modo se crean las matrices $\mathbf{\Omega}_l$:

$$\mathbf{\Omega}_l = \mathbf{C}^t \times_{\cup\cap} \mathbf{G}_l(X,Y)$$

5. *Regla Proyectiva:* $(X \subset U,\, Y \subset V,\, A_i \subset U \times V)$

$$\frac{(X,Y) \text{ es } A_i}{X \text{ es } \text{proj}_U A_i}$$

De acuerdo con la regla conjuntiva, "(X,Y) es A_i" está representado por una matriz $\mathbf{G}_l(X,Y)$ con $l = 1, \cdots, N$. La proyección sobre el espacio U queda definida por

$$\mathbf{G}(X) = \bigcup_{l=1}^{N} \mathbf{G}_l(X,Y)$$

donde $\bigcup$ es la t-conorma.

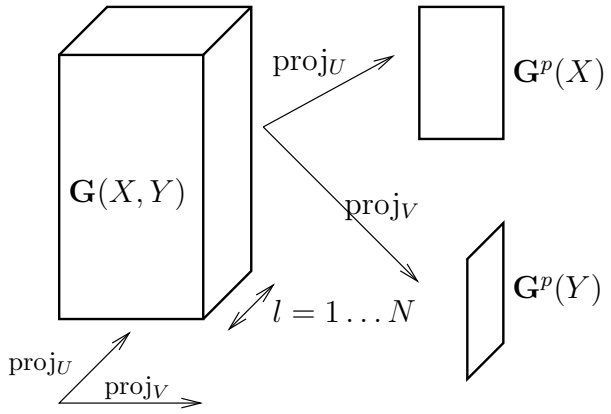


Figura 4.18 Proyecciones sobre la matriz $\mathbf{G}$

Si la regla se formulara

$$\frac{(X,Y) \text{ es } A_i}{Y \text{ es } \text{proj}_V A_i}$$

sería necesario hace una proyección sobre el espacio V. En tal caso si se representa $\mathbf{G}(X,Y)$ como una matriz tridimensional (con la tercera coordenada el índice l), tal y como se muestra en la Fig. 4.18, es fácil ver que la proyección necesaria en este caso es similar a la realizada anteriormente. Concretamente, si

$$\mathbf{G}_l(X,Y) = \begin{bmatrix} g^l_{11} & g^l_{12} & \cdots & g^l_{1M} \\ \vdots & \vdots & \vdots & \\ g^l_{(NM)1} & g^l_{(NM)2} & g^l_{(NM)M} & \end{bmatrix} = [g^l_{jk}]$$

la proyección sobre el espacio V es

$$\mathbf{G}(Y) = bigcup_{k=1}^{M}[g^l_{jk}]$$

Ejemplo: Supóngase un conjunto de $M \times N$ relaciones borrosas que vienen dadas por la expresión

$$X \text{ es } A_i \quad \text{e} \quad Y \text{ es } B_l, \qquad i = 1,\ldots,M \; l = 1,\ldots,N$$

representada por las matrices $\mathbf{G}_l(X,Y)$, $l = 1,\ldots,N$. El conjunto de M relaciones borrosas dado por

$$X \text{ es } A_i \qquad i = 1,\ldots,M$$

vendrá representado por la matriz $\mathbf{G}(X)$ de acuerdo con la expresión

$$\mathbf{G}(X) = \bigcup_{l=1}^{N} \mathbf{G}_l(X,Y)$$

6. *Regla Suprayectiva:*

$$\frac{X \text{ es } A_i}{(X,Y) \text{ es } A_i \times V}$$

La restricción"X es A_i" está representada por una matriz $\mathbf{R_A}$, de tamaño $M \times M$. El espacio de Y está formado por N conjuntos B_j. El conjunto de salida $A_i \times V$ queda definido por las matrices $\mathbf{G}_l$ de la siguiente forma:

$$\mathbf{G}_l(X,Y) = \mathbf{R_A}(X) \;\; l = 1,\ldots,N$$

Esta regla afecta a un conjunto de relaciones del tipo

$$X \text{ es } A_i$$

con $i = 1,\ldots,M$, que quiere convertirse en un conjunto de relaciones del tipo

$$X \text{ es } A_i \quad \text{e} \quad Y \text{ es } B_l$$

con $l = 1, \ldots, M$.

Ejemplo: Supóngase un conjunto de M relaciones borrosas que vienen dadas por la expresión

$$X \text{ es } A_i \qquad i = 1, \ldots, M$$

representada por la matriz $\mathbf{R_A}$. El conjunto de relaciones borrosas dado por

$$(X, Y) \text{ es } A_i \times V, \qquad i = 1, \ldots, M$$

(siendo V el espacio en que están definidos los conjuntos B_l $l = 1, \ldots, N$) vendrá representado por la matrices $\mathbf{G}_l(X, Y)$ de acuerdo con la expresión

$$\mathbf{G}_l(X, Y) = \mathbf{R_A}(X) \qquad l = 1, \ldots, N$$

Nótese que una proposición borrosa de la forma

$$\text{Si } [(X, Y) \text{ es } A_i \times V] \text{ entonces } Z \text{ es } C_{p(i)}$$

dará lugar a N matrices $\mathbf{\Omega}_l = \mathbf{C}^T \times_{\cup\cap} \mathbf{G}_l$ iguales entre sí. A la hora de realizar una inferencia, la componente correspondiente a la entrada Y no va a tener influencia sobre el resultado final.

5

APLICACIÓN A ESQUEMAS DE SEGUIMIENTO

BIBLIOTECA VIRTUAL

En el presente capítulo se describen dos implementaciones reales sobre esquemas de seguimiento de la teoría expuesta anteriormente. En primer lugar, un sistema de seguimiento basado en información textual es modelado usando para ello diferentes fuentes de información que se van a representar mediante gránulos, conjuntos borrosos y números reales. La inferencia global del sistema puede ser descrita usando la notación matricial introducida en el capítulo anterior.

Dado que las entradas a esta aplicación van a estar expresadas en lenguaje natural parece clara la utilización de técnicas basadas en CWW. No parece tan claro en la segunda aplicación descrita. En ella se modela un detector de maniobra para un filtro de Kalman en un entorno de seguimiento específico. En ese caso la ambigüedad no va a ser lingüística, sino que va a encontrarse en la dificultad de detectar un cambio de maniobra. Para ello se empleará un sistema realimentado similar al descrito en la sección 4.4.

5.1 ALGORITMO MHT BORROSO

5.1.1 Introducción

Los mensajes en lenguaje natural están presentes en muchas aplicaciones de procesado y análisis de información. Hasta la fecha la mayoría de estos sistemas se han utilizado para búsqueda en bases de datos o traducción automática. Es necesario que se desarrollen nuevas y más potentes técnicas de procesado de texto para manejar otras aplicaciones que puedan requerir correlación de mensajes basados en texto, tales como análisis inteligente, seguridad informática, bases de datos de incidentes etc.

Estas aplicaciones tienen varias características comunes: necesitan un seguimiento probabilístico de la información ambigua generada por diversos observadores en el tiempo (en este contexto, *seguimiento* significa encontrar qué mensajes tienen que ver con las mismas partes de la información, y por lo tanto, deben ser correlados de algún modo en el tiempo). Estas áreas de aplicación necesitan cada vez más de técnicas de análisis automático dado el incremento de la información basada en texto disponible.

La aplicación TextTrack, descrita en [3, 7], es un software cuyo propósito es la aplicación de conceptos de seguimiento y procesado avanzado de señal al procesado de lenguaje natural. TextTrack maneja problemas de correlación y seguimiento a partir observaciones de múltiples vehículos móviles. Dichas observaciones llegan en forma de mensajes expresados en lenguaje natural generados asíncronamente por múltiples observadores a lo largo del tiempo. El sistema ha demostrado que estos problemas se pueden tratar usando conceptos relativamente maduros de procesado de señal de RADAR, concretamente el algoritmo de seguimiento mediante múltiples hipótesis (MHT) [108]. El prototipo acepta mensajes sencillos en lenguaje natural sobre tipos y localizaciones de vehículos, correla estos mensajes y asocia grupos de mensajes en la pista más probable, basada en la sucesión de posiciones. Se proponen dos soluciones alternativas: primero, se usa una operativa Bayesiana para manejar la ambigüedad en las descripciones en lenguaje natural. Un teorema formal [7] demuestra que bajo unas restricciones leves, se alcanza la solución correcta. El segundo paso usa un motor de inferencias borroso, concretamente una versión borrosa del algoritmo MHT de Reid [108].

En [3] se plantea un análisis más riguroso del sistema, presentando un fundamento teórico sólido, mediante la caracterización analítica del motor de inferencias borroso en que se basa el algoritmo. La parte principal de este sistema es una variante de un FFS (sección 4.4). Tradicionalmente estos sistemas se han aplicado a teoría de control, por lo que las investigaciones previas se han centrado en temas específicos de dicha teoría. Para este problema las cantidades realimentadas al sistema van a ser probabilidades acumuladas a lo largo del tiempo.

El estudio de este sistema realimentado lleva a los autores a plantear en [3] un estudio basado en matrices de transición, muy similares a las planteadas en el capítulo 4, aunque no se propone un método para su construcción.

En este capítulo se modelará el algoritmo MHT borroso usando matrices de transición y gránulos borrosos en aquellas entradas en que esto sea conveniente. Se comparará este modelado con el modelado borroso estándar que inicialmente se utilizó para definir el sistema [7].

5.1.2 El algoritmo de Seguimiento mediante Múltiples Hipótesis (MHT)

Este algoritmo es una aproximación Bayesiana al problema del seguimiento de múltiples blancos, y está basado en el hecho de que las decisiones han de ser aplazadas hasta que haya suficiente evidencia de que la elección es correcta. La idea clave es simplemente almacenar en memoria (idealmente) todas las hipótesis que se hayan podido realizar a partir de las observaciones obtenidas hasta ese momento, así como una estimación de la probabilidad de tales hipótesis. Las hipótesis más probables son las consideradas como "verdaderas". Donald B. Reid planteó las ideas básicas de este algoritmo a finales de los 70 en [108].

El algoritmo MHT se usa típicamente en aplicaciones de radar con varios sensores. Los sensores devuelven una gran cantidad de información en intervalos discretos de tiempo, produciendo un barrido. Las informaciones obtenidas de diferentes barridos se organizan en pistas, de acuerdo con un cálculo probabilístico basado en la dinámica de los objetos detectados. A una colección de pistas consistentes (es decir, que diferentes pistas no compartan información) es lo que se denomina una hipótesis, y cada hipótesis tiene un valor de verosimilitud. El propósito es mantener las hipótesis de mayor verosimilitud, teniendo en cuenta que futuros mensajes pueden cambiar drásticamente las probabilidades de las hipótesis actuales.

Sin embargo, si este algoritmo se implementa directamente como ha sido explicado anteriormente, en muy poco tiempo el sistema se verá afectado de problemas de memoria. Existe una explosión combinatoria incluso con un pequeño número de observaciones. Por lo tanto, habrá que usar algún procedimiento para controlar el número de hipótesis que se mantienen en memoria.

De manera más rigurosa, supóngase que en un instante k se reciben M_k medidas que están almacenadas en el vector $Z(k)$. Se puede suponer que antes de recibir estas medidas ya existe un número de pistas establecidas. Cada una de las M_k observaciones puede ser

1. Una nueva observación correspondiente a una de las pistas existentes
2. Un nuevo blanco que aparece en el campo de visión del sensor.
3. Una falsa alarma debido a un error por *clutter* o a ruido térmico.

El propósito del algoritmo MHT es obtener de una manera efectiva la clasificación de estas nuevas M_k observaciones. Para ese fin, se almacenan distintas hipótesis derivadas de posibles asociaciones de observaciones a las rutas, se calcula su probabilidad, y se eliminan algunas de las hipótesis menos probables.

Sea Θ_m^k la m-ésima hipótesis activa después de recibir y procesar la observación k-ésima $Z(k)$. Esta hipótesis está construida a partir de la hipótesis $\Theta_{l(m)}^{k-1}$ en el anterior instante de tiempo la cual es actualizada por la asociación datos-a-ruta $\theta_m(k)$. Entonces

$$\Theta_m^k = \left\{\Theta_{l(m)}^{k-1}, \theta_m(k)\right\} \tag{5.1}$$

Se denota por Z^k el conjunto de medidas hasta el instante de tiempo k (inclusive), y consistirá en

$$Z^k = \left\{Z^{k-1}, Z(k)\right\} \tag{5.2}$$

Se pretende poder obtener

$$P\left\{\Theta_m^k | Z^k\right\} \tag{5.3}$$

lo que, tras usar el Teorema de Bayes, se puede escribir como

$$\begin{aligned} P\left\{\Theta_m^k|Z^k\right\} &= P\left\{\Theta_{l(m)}^{k-1}, \theta_m(k)|Z^k\right\} \\ &= \frac{1}{c}P\left\{Z(k)|\Theta_{l(m)}^{k-1}, \theta_m(k), Z^{k-1}\right\} \\ &\quad P\left\{\theta_m(k)|\Theta_{l(m)}^{k-1}, Z^{k-1}\right\} P\left\{\Theta_{l(m)}^{k-1}|Z^{k-1}\right\} \end{aligned} \tag{5.4}$$

donde c es una constante de normalización. El último término de la ecuación es la probabilidad de la hipótesis global anterior, por lo que está disponible de la iteración anterior. El primer factor es la verosimilitud de las medidas dada una asociación de medidas a pistas. El segundo factor es la probabilidad de esta asignación concreta de la observación a las pistas preexistentes.

5.1.3 MHT con entradas de texto

El algoritmo MHT borroso surge ante la necesidad de combinar las ideas clásicas de seguimiento (descritas brevemente en la sección anterior) con el procesado de lenguaje natural. Su propósito es el de desarrollar un sistema capaz de recibir mensajes orales o escritos sobre observaciones de múltiples blancos y ser capaz de correlar dichas observaciones. Los mensajes contienen la siguiente información:

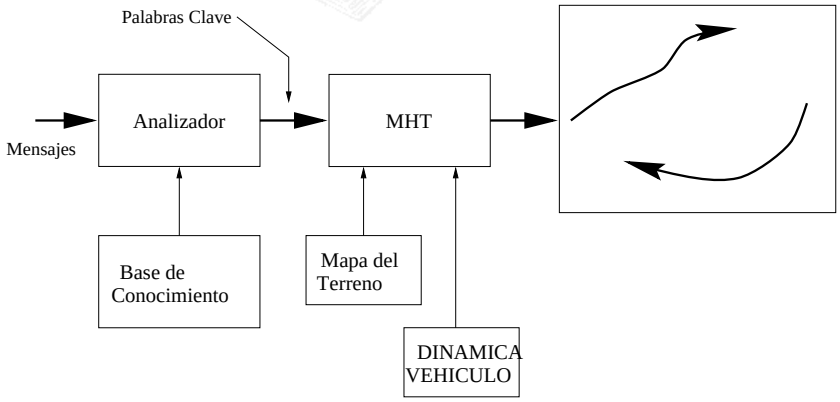


Figura 5.1 Propuesta de arquitectura del sistema de seguimiento con entradas de texto

- Tiempo del mensaje
- Localización del vehículo observado
- Descripción de algunas características del objeto (color, tamaño...)
- Dirección del objeto

El sistema analizará el mensaje para buscar las palabras clave (ver figura 5.1) y hará uso tanto de la información sobre la dinámica de los objetos como de la información de la

infraestructura disponible (carreteras, ríos, etc). Esta información permitirá al sistema estimar las posiciones futuras de los objetos así como construir las ventanas adecuadas para la asociación de datos, pero sólo en aquello sitios en los que sea razonable encontrar el objeto en el futuro (si el objeto es un vehículo de carretera, la búsqueda se realizará sólo a lo largo de las carreteras, si es un barco, sólo en aquellos sitios en que pueda navegar). De manera más concreta, la información contenida en los mensajes se usará de la siguiente forma:

- El *tiempo del mensaje* se usa para calcular el tiempo transcurrido entre dos observaciones. Este tiempo va a ser usado posteriormente para predecir la posición del vehículo en función de la velocidad del mismo.
- A partir de la *posición del vehículo observado* se comprueba si la observación cae dentro de la ventana de cada pista. La diferencia entre la posición predicha y la observada constituye el *error de predicción*
- La descripción del objeto se usa como medida de la *compatibilidad* entre la historia de la pista y la observación. Esto quiere decir que para correlar una medida con una ruta necesitamos que se cumplan dos condiciones:
  1. La observación debe estar cercana físicamente a la predicción, lo cual se cuantifica por medio de la ventana de correlación ya descrita.
  2. La observación debe estar cercana lógicamente a la predicción, lo que significa que sólo se correlan objetos similares. En otras palabra, no tiene sentido correlar una camión con una bicicleta.

  Así que el factor
  $$P\left\{\theta_m(k)|\Theta_{l(m)}^{k-1}, Z^{k-1}\right\}$$
  en la expresión (5.4) se considera como la medida de compatibilida entre objetos, de la siguiente manera:

$$P\left\{\theta_m(k)|\Theta_{l(m)}^{k-1}, Z^{k-1}\right\} = \begin{cases} P\left\{Vh(k)|Vh(k-1)\right\}(1-P_{new_t}) & \text{si blanco en ruta} \\ P_{new_t} & \text{si nuevo blanco} \end{cases} \tag{5.5}$$

  donde $Vh(k)$ es el vehículo observado en el instante k y P_{new_t} es la probabilidad de que el mensaje describa un blanco que no existía anteriormente.

5.1.4 Implementación borrosa

El sistema con las necesidades descritas anteriormente ha sido modelado en [3, 7] usando un sistema borroso tipo SAM [72] y definiendo los términos lingüísticos mediante el uso de conjuntos borrosos triangulares o trapezoidales. En la presente sección se mostrará que es este caso es más sencillo y conveniente un modelado basado en gránulos. Además, este sistema es ideal para su implementación usando matrices de transición.

Fuentes de ambigüedad

Dado que se va a trabajar con entradas de lenguaje natural, basadas en mensajes de texto, aparecen numerosas fuentes de ambigüedad, debidas a la imprecisión de las entradas o bien a la propia de este tipo de lenguaje. Las principales fuentes de ambigüedad van a ser:

Ambigüedad en la descripción de los objetos: Cuando llega un mensaje sobre un tipo de vehículo, éste lleva implícito cierta clase de ambigüedad. Por ejemplo, un observador puede informar del paso de un *camión*, mientras que a otro observador el mismo vehículo le puede parecer una *furgoneta*, o incluso puede parecerle un *coche grande* Lo que para un observador es un *coche*, para otro puede ser un *monovolumen*, o simplemente un *vehículo pequeño*.

Por otro lado, la cuestión de si un vehículo es pequeño o es grande es también algo sujeto a variabilidad. Las categorías de *grande, muy grande, pequeño, muy pequeño,* etc, corresponden a una impresión subjetiva de cada uno. Alguien acostumbrado a conducir un coche grande, verá el resto de los coches como pequeños, y viceversa.

A la ambigüedad propia de cada vehículo, se pueden añadir las propias de su descripción. La aplicación podría aceptar descripciones de dicho vehículo, tales como el *tamaño* o el *color*.

Ambigüedad en la descripción de la localización de los objetos: La posición de los objetos se puede describir de diversos modos. Un observador puede tener medios automáticos para informar de la posición de los objetos de una manera muy precisa, por ejemplo por coordenadas absolutas usando un equipo GPS. En este caso la ambigüedad no existe, salvo por el escaso ruido de medida.

Sin embargo, los objetos pueden referenciarse diciendo que se encuentran *no muy lejos* de un sitio, que están *cerca* de algún punto, o que están al *otro lado* de algún sitio conocido. En este caso, habría que modelar convenientemente la ambigüedad introducida por estos informes. De nuevo, la subjetividad del informador juega un papel importante. Haría falta también conocer de antemano la localización de ciertos puntos, para tomarlos como lugares de referencia.

Incertidumbre en la predicción de futuras posiciones del objeto: Esta parte no es consecuencia directa de la ambigüedad en el lenguaje, sino que es consecuencia de la dinámica de los objetos. Típicamente, sólo será útil una descripción estadística de la cinemática de los objetos, y por lo tanto las predicciones de posiciones futuras están sujetas a errores, cuyas varianzas dependen tanto del modelo usado como de el tiempo transcurrido entre observación y predicción.

El error en la predicción, es decir, la diferencia entre la predicción y la siguiente observación es una medida concreta, pero puede ser interpretada en términos de *error pequeño* o *error grande*.

Definición borrosa de las entradas

A partir de las fuentes de ambigüedad planteadas, surge por lo tanto la necesidad de definir las entradas al sistema usando métodos borrosos. Concretamente, se van a definir usando variables lingüísticas.

Tipo de Vehículo: La aplicación original plantea 6 posibles tipos de vehículos: *motocicleta*, *coche*, *jeep*, *lanzadera*, *furgoneta* y *camión*. Dichos vehículos en [7] son modelados como conjuntos borrosos pertenecientes a una variable lingüística $\mathcal{V}$. Los vehículos así descritos aparecen en la Fig 5.2[1].

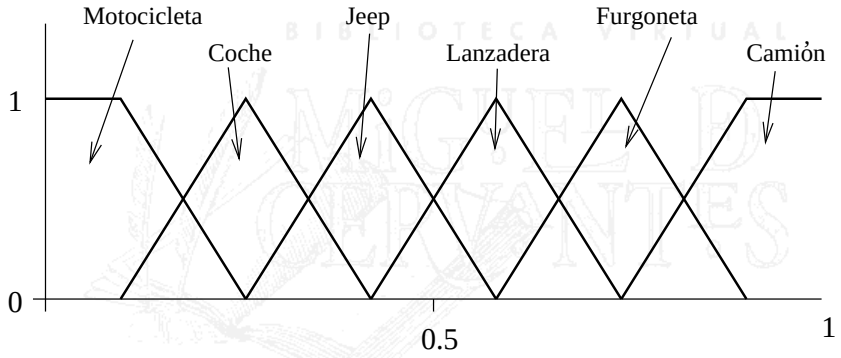


Figura 5.2 Interpretación de la variable lingüística *Tipo de Vehículo* usando conjuntos borrosos

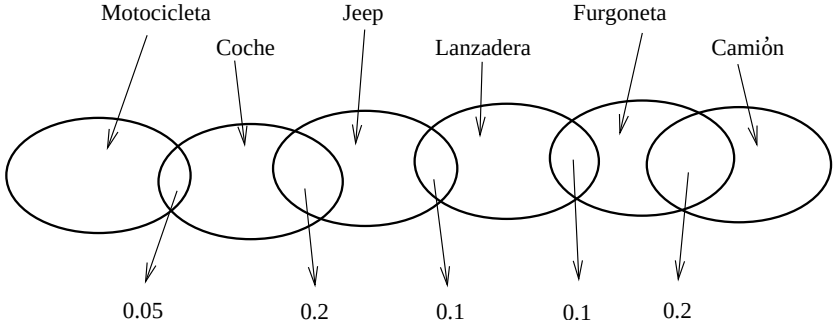


Figura 5.3 Interpretación de la variable lingüística *Tipo de Vehículo* usando gránulos borrosos

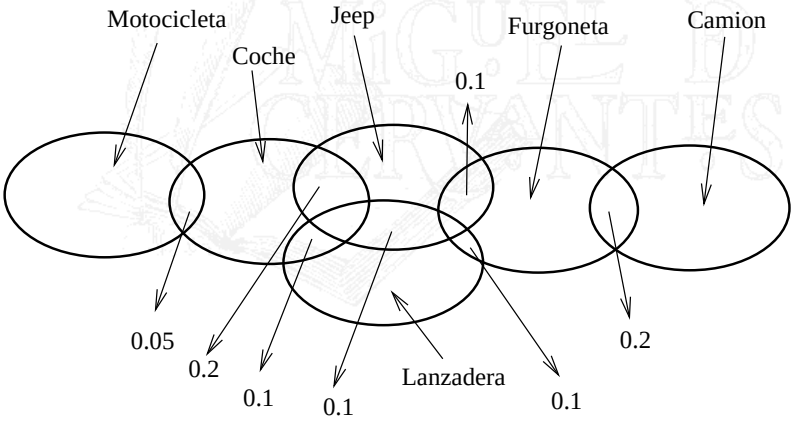
Nótese que ha tenido que crearse una variable base ficticia sobre la que se ha realizado una partición borrosa que ha dado lugar a los distintos conjuntos. Dado que los conceptos que hacen referencia a la VL Vehículo no tienen asociada de forma natural esta variable base, pueden definirse de forma más sencilla usando gránulos borrosos, tal y como se muestra en la Fig. 5.3. En este caso, la representación gráfica es sólo por motivos de facilidad de comprensión, ya que la variable queda totalmente

[1]Los valores usados para los grados de solapamiento de los distintos gránulos han sido elegidos de forma intuitiva, con una intención meramente ilustrativa. Existen técnicas que permiten el aprendizaje de estos valores a partir de los datos [136].

representada por su matriz de intersecciones:

$$\mathbf{R}_{\mathcal{V}} = \begin{bmatrix} \text{moto} \circ \text{moto} & \text{moto} \circ \text{coche} & \cdots & \text{moto} \circ \text{camión} \\ \text{coche} \circ \text{moto} & \text{coche} \circ \text{coche} & \cdots & \text{coche} \circ \text{camión} \\ \vdots & \vdots & & \vdots \\ \text{camión} \circ \text{moto} & \text{camión} \circ \text{coche} & \cdots & \text{camión} \circ \text{camión} \end{bmatrix}$$

$$= \begin{bmatrix} 1 & 0.05 & 0 & 0 & 0 & 0 \\ 0.05 & 1 & 0.2 & 0 & 0 & 0 \\ 0 & 0.2 & 1 & 0.1 & 0 & 0 \\ 0 & 0 & 0.1 & 1 & 0.1 & 0 \\ 0 & 0 & 0 & 0.1 & 1 & 0.2 \\ 0 & 0 & 0 & 0 & 0.2 & 1 \end{bmatrix} \tag{5.6}$$



,

Figura 5.4 Interpretación alternativa de la variable lingüística *Tipo de Vehículo* usando gránulos borrosos

Se ha usado la misma distribución que tenía el sistema original, pero la definición mediante gránulos permite que los solapamientos entre conjuntos sean más complejos, acercándose más al caso real. Como ejemplo, se podría haber definido la variable tal y como se muestra en la Fig. 5.4, con matriz de intersecciones

$$\mathbf{R}_{\mathcal{V}} = \begin{bmatrix} 1 & 0.05 & 0 & 0 & 0 & 0 \\ 0.05 & 1 & 0.2 & 0.1 & 0 & 0 \\ 0 & 0.2 & 1 & 0.1 & 0.1 & 0 \\ 0 & 0.1 & 0.1 & 1 & 0.1 & 0 \\ 0 & 0 & 0 & 0.1 & 1 & 0.2 \\ 0 & 0 & 0 & 0.1 & 0.2 & 1 \end{bmatrix} \tag{5.7}$$

En lo sucesivo se considerará la matriz definida en la ecuación (5.6).

Error de Predicción: El error de predicción es la diferencia entre la posición predicha y la posición en que se encuentra realmente el objeto. Este error es un número concreto, que en la aplicación original se discretiza en seis niveles que se corresponden con los seis conjuntos que se definen dentro de la variable lingüística $\mathcal{E}$. Los valores

serán *casi cero, pequeño, medio, grande, muy grande* y *excesivo*, tal y como aparecen en la figura 5.5.

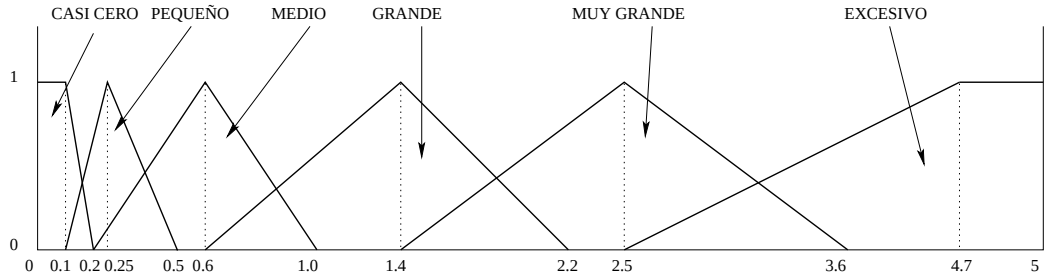


Figura 5.5 Variable Lingüística *Error de Predicción*

El error está normalizado por la desviación típica del mismo, de ahí que un error de 5 se considere excesivo.

Nótese que ahora la existencia de una variable base está plenamente justificada, ya que el *error* es una magnitud medible, y por lo tanto tiene sentido describir los conjuntos en términos numéricos. La entrada, en este caso será el valor del error, que es un valor no borroso, que activará a alguno de los conjuntos definidos en la VL. Por lo tanto, de acuerdo con lo visto en la sección 4.2.6, la matriz de activación de reglas será la matriz identidad

$$\mathbf{R}_{\mathcal{E}} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \tag{5.8}$$

Probabilidad: Al comparar las compatibilidades entre los vehículos observados y los errores de predicción se obtendrá una probabilidad de que el vehículo observado pertenezca a una pista. Dado que todo el proceso se realiza dentro de un motor de inferencia borrosa, se define una nueva variable lingüística llamada *Probabilidad*, que va a asignar etiquetas de probabilidad a los distintos eventos y a la pista.

La variable lingüística $\mathcal{P}$ se define por medio de 5 valores: *Muy Probable, Probable, Posible, Improbable* y *Muy Improbable*. Estas etiquetas corresponden a los consiguientes conjuntos borrosos, que tienen como funciones de pertenencia las que aparecen en la figura 5.6.

Por tratarse de una probabilidad se le asigna una variable base cuyos valores están contenidos entre 0 y 1. En el proceso la probabilidad se maneja siempre como conjunto borroso, (excepto el valor final de salida, del que se obtendrá el centroide), por lo que se puede definir una matriz de activación para esta variable, suponiendo

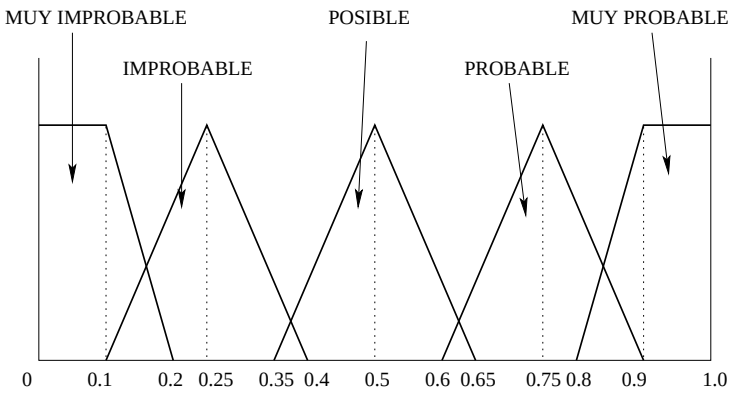


Figura 5.6 Variable Lingüística *probabilidad*

una composición de conjuntos usando operadores suma y producto:

$$\mathbf{R}_{\mathcal{P}} = \begin{pmatrix} 1 & 0.1 & 0 & 0 & 0 \\ 0.1 & 1 & 0.01 & 0 & 0 \\ 0 & 0.01 & 1 & 0.01 & 0 \\ 0 & 0 & 0.01 & 1 & 0.1 \\ 0 & 0 & 0 & 0.1 & 1 \end{pmatrix} \tag{5.9}$$

Modificadores: Es posible incluir modificadores como términos de entrada, de tal manera que cambien los valores de los conjuntos. Para esta aplicación se ha optado por un conjunto de modificadores bajo el nombre de *Tamaño* que afectarán a la entrada *Tipo de vehículo* cambiando las intersecciones entre los gránulos. Definimos cinco modificadores:

1. *Muy Grande:* desplaza los conjuntos hacia la derecha, disminuyendo la intersección con los conjuntos que consideramos de tamaño inferior, y aumentando la intersección con los que se consideran mayores. La matriz de modificación para el término *Coche* será de la siguiente forma

$$\mathbf{\Delta R}_{\text{Muy Grande}}(Coche) = \begin{pmatrix} 0 & -0.05 & 0 & 0 & 0 & 0 \\ -0.05 & -0.1 & 0.2 & 0.1 & 0.05 & 0 \\ 0 & 0.2 & 0 & 0 & 0 & 0 \\ 0 & 0.1 & 0 & 0 & 0 & 0 \\ 0 & 0.05 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \tag{5.10}$$

2. *Grande:* desplaza los conjuntos hacia la derecha, disminuyendo la intersección con los conjuntos que consideramos de tamaño inferior, y aumentando la intersección con los que se consideran mayores, pero en menor término que el modificador anterior. La matriz de modificación para el término *Coche* será de

la siguiente forma

$$\mathbf{\Delta R}_{\text{Grande}}(Coche) = \begin{pmatrix} 0 & -0.04 & 0 & 0 & 0 & 0 \\ -0.04 & 0 & 0.1 & 0.05 & 0 & 0 \\ 0 & 0.1 & 0 & 0 & 0 & 0 \\ 0 & 0.05 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \tag{5.11}$$

3. *Normal:* juega el papel del elemento neutro dentro de los modificadores. No cambia los conjuntos. Su matriz de modificación será de la forma

$$\mathbf{\Delta R}_{\text{Normal}}(Coche) = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \tag{5.12}$$

4. *Pequeño:* desplaza los conjuntos en sentido inverso que el modificador *Grande*. La matriz de modificación para el término *Coche* será de la siguiente forma

$$\mathbf{\Delta R}_{\text{Pequeño}}(Coche) = \begin{pmatrix} 0 & 0.01 & 0 & 0 & 0 & 0 \\ 0.01 & 0 & -0.05 & 0 & 0 & 0 \\ 0 & -0.05 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \tag{5.13}$$

5. *Muy Pequeño:* desplaza los conjuntos en sentido inverso que el modificador *Grande*. La matriz de modificación para el término *Coche* será de la siguiente forma

$$\mathbf{\Delta R}_{\text{MuyPequeño}}(Coche) = \begin{pmatrix} 0 & 0 & 0.02 & 0 & 0 & 0 \\ 0.02 & 0 & -0.15 & 0 & 0 & 0 \\ 0 & -0.15 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \tag{5.14}$$

Sistema borroso

Si se acepta que toda la ambigüedad en los mensajes ha sido capturada por los conjuntos borrosos descritos anteriormente, el siguiente paso será cuantificar recursivamente la probabilidad de todas las hipótesis, esto es, de todas las posibles asignaciones de medidas a pistas. El proceso de asignación se puede hacer mediante una serie de reglas borrosas (tipo Mamdani) *si-entonces*:

	Moto	Coche	Jeep	Lanzadera	Furgoneta	Camión
Moto	Muy Prob.	Probable	Improbable	Muy Improb.	Muy Improb.	Muy Improb.
Coche	Probable	Muy Prob.	Probable	Improbable	Muy Improb.	Muy Improb.
Jeep	Improbable	Probable	Muy Prob.	Probable	Posible	Improbable
Lanzadera	Muy Improb.	Improbable	Probable	Muy Prob.	Probable	Posible
Furgoneta	Muy Improb.	Muy Improb.	Posible	Probable	Muy Prob.	Probable
Camión	Muy Improb.	Muy Improb.	Improbable	Posible	Probable	Muy Prob.

Tabla 5.1 Tabla de reglas para la compatibilidad entre vehículos

1. En primer lugar se crea un conjunto de reglas que tengan en cuenta la *compatibilidad* entre vehículos en la pista. Si se considera la compatibilidad como un proceso de un solo paso (en otras palabras, se considera simplemente compatibilidad entre el último vehículo de la pista y la observación) el análisis se reduce a una serie de reglas de la forma[2]

Regla:	Si X es $Coche$,	e Y es $Coche$	entonces Z es $Muy\ \ Probable$
Regla:	Si X es $Coche$,	e Y es $Furgoneta$	entonces Z es $Probable$
...	...	...	...
Regla:	Si X es $Moto$,	e Y es $Camión$	entonces Z es $Muy\ \ Improbable$

   donde Y representa al último vehículo en la pista y X representa al vehículo observado. El sistema es una transformación $f : X \in \mathcal{V} \times Y \in \mathcal{V} \rightarrow Z \in \mathcal{P}$. Obviamente lo que se trata de encontrar es el grado de compatibilidad (expresado aquí por la variable lingüística *probabilidad*) de la asignación de esta observación a esa pista (usando sólo el tipo de vehículo, es decir, en términos únicamente de compatibilidad "lógica").

   Este tipo de reglas dan lugar una base de conocimiento completa con 6×6 reglas (representada en la tabla 5.1), y a una matriz de reglas $\mathbf{C}_{\mathcal{V}}$ de tamaño 36×5, de la forma:

   $$\mathbf{C}_{\mathcal{V}} = \begin{pmatrix} 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix} \tag{5.15}$$

   correspondiendo cada fila a la regla $R_{i \times j}$ y cada columna a los conjuntos de salida.

2. En una segunda fase, se ha de comparar la compatibilidad lógica (vehículo observado con vehículo anterior) con la física (error de predicción). En este caso, la variable $\mathcal{C}$ representa la compatibilidad lógica obtenida anteriormente, y la variable $\mathcal{E}$ es el error cometido al predecir la posición del vehículo. Las reglas a las que esto da lugar son de la forma

Regla:	Si C es $Muy\ \ Probable$,	y E es $Casi\ \ Cero$	entonces Z es $Muy\ \ Probable$
Regla:	Si C es $Muy\ \ probable$,	y E es $Pequeño$	entonces Z es $Probable$
...	...	...	...
Regla:	Si C es $Muy\ \ Improbable$,	y E es $Excesivo$	entonces Z es $Muy\ \ Improbable$

[2] Si la compatibilidad tuviera más memoria, haría falta definir un sistema borroso de múltiples entradas y una salida.

	Casi 0	Pequeño	Medio	Grande	Muy Grande	Excesivo
Muy Improb.	Improbable	Improbable	Improbable	Muy Improb.	Muy Improb.	Muy Improb.
Improbable	Posible	Posible	Improbable	Muy Improb.	Muy Improb.	Muy Improb.
Posible	Probable	Posible	Posible	Improbable	Improbable	Muy Improb.
Probable	Muy Prob.	Probable	Probable	Posible	Improbable	Muy Improb.
Muy prob.	Muy Prob.	Muy Prob.	Probable	Posible	Improbable	Muy Improb.

Tabla 5.2 Tabla de reglas para la comparación de compatibilidad lógica y física.

	Muy Improb.	Improbable	Posible	Probable	Muy prob.
Muy Improb.	Muy Improb.	Muy Improb.	Improbable	Improbable	Improbable
Improbable	Muy Improb.	Muy Improb.	Improbable	Improbable	Improbable
Posible	Improbable	Improbable	Posible	Posible	Probable
Probable	Improbable	Improbable	Posible	Probable	Muy prob.
Muy prob.	Improbable	Improbable	Probable	Muy prob.	Muy prob.

Tabla 5.3 Tabla de reglas para la actualización de probabilidades

El sistema es una transformación $f : C \in \mathcal{P} \times E \in \mathcal{E} \rightarrow Z \in \mathcal{P}$. Las entradas serán un conjunto borroso C que viene de la inferencia anterior y un valor numérico E que es el error de predicción medido.

Este tipo de reglas dan lugar una base de conocimiento completa con 7×6 reglas (representada en la tabla 5.2), y a una matriz de reglas $\mathbf{C}_{\mathcal{E}}$ de tamaño 42×5, donde la columna j de la fila i es uno si la regla i-ésima tiene de salida el conjunto j-ésimo.

3. Hasta este punto se ha obtenido el *grado de verdad* de la asignación de una observación a una pista (o generalizando, a una hipótesis). Sin embargo, el algoritmo MHT es recursivo por naturaleza y por lo tanto es necesario actualizar la probabilidad de todas las hipótesis como una función de las probabilidades tanto de las asignaciones obtenidas en este instante de tiempo como de la historia de la hipótesis.

   Esto puede hacerse de una manera sencilla por medio de una última inferencia en la que las asignaciones probables refuerzan las hipótesis probables, y las asignaciones improbables hacen más improbables a las hipótesis. Una posible implementación de esta tabla de verdad sería la siguiente:

Regla:	Si P es $Muy\ \ Probable$,	y H_t es $Muy\ \ Probable$	entonces H_{t+1} es $Muy\ \ Probable$
Regla:	Si P es $Muy\ \ Probable$,	y H_t es $Posible$	entonces H_{t+1} es $Probable$
...	...	...	...
Regla:	Si P es $Muy\ \ Improbable$,	y H_t es $Posible$	entonces H_{t+1} es $Improbable$

   donde H_t es en este caso la probabilidad de la hipótesis actual en el instante t, y P la probabilidad de la asignación. El sistema es una transformación $f : P \in \mathcal{P} \times H_t \in \mathcal{P} \rightarrow H_{t+1} \in \mathcal{P}$. Cabe resaltar que la tabla de verdad no tiene por qué ser simétrica, pudiendo así ajustarse la importancia relativa de la historia o la asignación en la probabilidad global de las hipótesis. Nótese también que esta clase de reglas dan lugar a un FFS (véase tercer bloque de la Fig.5.7).

   La base de conocimiento completa con 5×5 reglas (representada en la tabla 5.3) tiene como matriz de reglas $\mathbf{C}_{\mathcal{P}}$ de tamaño 25×5, definida del mismo modo que para los otros bloques.

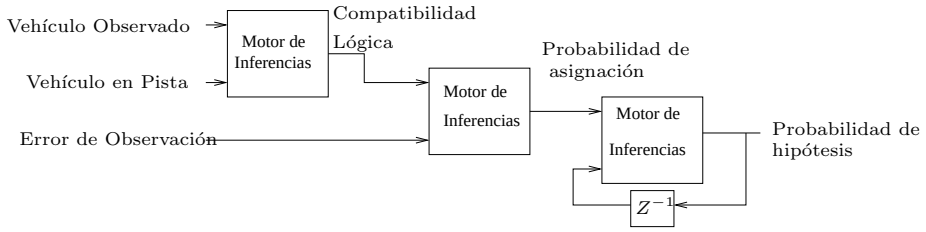


Figura 5.7 Esquema del proceso de obtención de *etiquetas de probabilidad*

5.1.5 Implementación del sistema borroso usando matrices de transición

BIBLIOTECA VIRTUAL

Siguiendo lo visto en el capítulo 4 se procederá a implementar el sistema borroso descrito anteriormente usando matrices de transición.

Bloque de compatibilidad lógica

El primer bloque de la Fig. 5.7 se implementará a partir de la matriz $\mathbf{R}_{\mathcal{V}}$ (ecuación (5.7)). Dado que la variable lingüística está formada por 6 elementos (gránulos borrosos), esto dará lugar a 6 matrices de transición $\mathbf{\Omega}_i^{\mathcal{V}}$, $i = 1, \ldots, 6$. El proceso de construcción es el siguiente:

$$\mathbf{G}_i^{\mathcal{V}} = \mathbf{R}_{\mathcal{V}} \otimes_{\mathcal{K}} [\mathbf{R}_{\mathcal{V}} \mathbf{E}_i] \;\; i = 1, \ldots, 6$$

$$\mathbf{\Omega}_i^{\mathcal{V}} = \mathbf{C}_{\mathcal{V}}^T \mathbf{G}_i^{\mathcal{V}} \;\; i = 1, \ldots, 6$$

De este modo, el proceso de inferencia queda definido por la ecuación

$$\mathbf{p}^1 = \left[\sum_{i=1}^{6} v_i^p \mathbf{\Omega}_i^{\mathcal{V}} \right] \mathbf{v}^o$$

$$\begin{bmatrix} p_1^1 \\ p_2^1 \\ p_3^1 \\ p_4^1 \\ p_5^1 \end{bmatrix} = \left[\sum_{i=1}^{6} v_i^p \mathbf{\Omega}_i^{\mathcal{V}} \right] \begin{bmatrix} v_1^o \\ v_2^o \\ v_3^o \\ v_4^o \\ v_5^o \\ v_6^o \end{bmatrix}$$

siendo $\mathbf{p}^1 \in \mathcal{P}$ el vector de salida, $\mathbf{v}^o \in \mathcal{V}$ el vector de entrada correspondiente al vehículo observado y $\mathbf{v^P} \in \mathcal{V}$ el vector de entrada correspondiente al último vehículo en la pista.

Toda la información del sistema se encuentra recogida en 6 matrices de transición de tamaño 5×6. Los valores de las matrices se encuentran recogidos en el apéndice D.3.

Bloque de compatibilidad física y lógica

El segundo bloque de la Fig. 5.7 se implementará a partir de las matrices $\mathbf{R}_{\mathcal{E}}$ (asociada al error, ecuación (5.8)) y $\mathbf{R}_{\mathcal{P}}$ (correspondiente a las probabilidades, ecuación (5.9)). Dado que la variable lingüística $\mathcal{E}$ está formada por 6 elementos (conjuntos borrosos), esto dará lugar nuevamente a 6 matrices de transición $\mathbf{\Omega}_j^{\mathcal{E}}$, $j = 1, \ldots, 6$. El proceso de construcción es el siguiente:

$$\mathbf{G}_j^{\mathcal{E}} = \mathbf{R}_{\mathcal{P}} \otimes_{\mathcal{K}} [\mathbf{R}_{\mathcal{E}} \mathbf{E}_j] \;\; j = 1, \ldots, 6$$

$$\mathbf{\Omega}_j^{\mathcal{E}} = \mathbf{C}_{\mathcal{E}}^T \mathbf{G}_j^{\mathcal{E}} \;\; j = 1, \ldots, 6$$

De este modo, el proceso de inferencia queda definido por la ecuación

$$\mathbf{p}^2 = \left[\sum_{j=1}^{6} e_j \mathbf{\Omega}_j^{\mathcal{E}} \right] \mathbf{p}^1$$

$$\begin{bmatrix} p_1^2 \\ p_2^2 \\ p_3^2 \\ p_4^2 \\ p_5^2 \end{bmatrix} = \left[\sum_{j=1}^{6} e_j \mathbf{\Omega}_j^{\mathcal{E}} \right] \begin{bmatrix} p_1^1 \\ p_2^1 \\ p_3^1 \\ p_4^1 \\ p_5^1 \end{bmatrix}$$

siendo $\mathbf{p}^1 \in \mathcal{P}$ el vector de entrada, procedente del bloque anterior y $\mathbf{p}^2 \in \mathcal{P}$ el vector de salida. Los términos e_j se corresponden con la activación del conjunto $E_j(x)$ de la variable lingüística $\mathcal{E}$ por el error de predicción. Si este error se denota por e_{x_0}, los términos e_j equivaldrían a $E_j(e_{x_0})$:

$$\mathbf{p}^2 = \left[\sum_{j=1}^{6} E_j(e_{x_0}) \mathbf{\Omega}_j^{\mathcal{E}} \right] \mathbf{p}^1$$

Nótese que para este caso, sí es necesario almacenar los conjuntos que forman la variable *Error de Predicción*, ya que la entrada es numérica. Las matrices $\mathbf{\Omega}_j^{\mathcal{E}}$ serán de tamaño 5×5. Sus valores se encuentran recogidos en el apéndice D.3.

Bloque de actualización de probabilidades

El último bloque de la Fig. 5.7 se implementará a partir de la matriz $\mathbf{R}_{\mathcal{P}}$, correspondiente a las probabilidades (ecuación (5.9)). Dado que la variable lingüística $\mathcal{P}$ está formada por 5 elementos (conjuntos borrosos), esto dará lugar a 5 matrices de transición $\mathbf{\Omega}_k^{\mathcal{P}}$, $k = 1, \ldots, 5$. El proceso de construcción es el siguiente:

$$\mathbf{G}_k^{\mathcal{P}} = \mathbf{R}_{\mathcal{P}} \otimes_{\mathcal{K}} [\mathbf{R}_{\mathcal{P}} \mathbf{E}_k] \;\; k = 1, \ldots, 5$$

$$\mathbf{\Omega}_k^{\mathcal{P}} = \mathbf{C}_{\mathcal{P}}^T \mathbf{G}_k^{\mathcal{P}} \;\; k = 1, \ldots, 5$$

De este modo, el proceso de inferencia queda definido por la ecuación

$$\begin{aligned} \mathbf{h}^{t+1} &= \left[\sum_{k=1}^{5} h_k^t \mathbf{\Omega}_k^{\mathcal{P}}\right] \mathbf{p}^2 \\ \begin{bmatrix} h_1^{t+1} \\ h_2^{t+1} \\ h_3^{t+1} \\ h_4^{t+1} \\ h_5^{t+1} \end{bmatrix} &= \left[\sum_{k=1}^{5} h_k^t \mathbf{\Omega}_k^{\mathcal{P}}\right] \begin{bmatrix} p_1^2 \\ p_2^2 \\ p_3^2 \\ p_4^2 \\ p_5^2 \end{bmatrix} \end{aligned}$$

siendo $\mathbf{p}^2 \in \mathcal{P}$ el vector de entrada, procedente del bloque anterior y $\mathbf{h}^t \in \mathcal{P}$ el vector de probabilidad acumulada de la hipótesis en el instante t. Las matrices de transición tendrán un tamaño 5×5. Sus valores se encuentran de nuevo recogidos en el apéndice D.3.

Sistema completo

Los distintos bloques descritos anteriormente tienen la propiedad de que la salida de cada uno de ellos es la entrada de otro. Por lo tanto, pueden fusionarse en un sólo sistema, evitando así variables intermedias:

$$\mathbf{h}^{t+1} = \left[\sum_{k=1}^{5} h_k^t \mathbf{\Omega}_k^{\mathcal{P}}\right] \left[\sum_{j=1}^{6} E_j(e_{x_0}) \mathbf{\Omega}_j^{\mathcal{E}}\right] \left[\sum_{i=1}^{6} v_i^p \mathbf{\Omega}_i^{\mathcal{V}}\right] \mathbf{v}^o \tag{5.16}$$

Evolución del sistema para entrada estacionaria

Como ya se vio en la sección 4.4.2 es posible calcular el estado final de un sistema realimentado definido mediante matrices de transición cuando la entrada permanece estacionaria. Si se supone que la salida del segundo bloque (Fig. 5.7) es constante (por ejemplo, para analizar el comportamiento del sistema en fase de diseño), la expresión (5.16) pasa a ser

$$\mathbf{h}^{t+1} = \left[\sum_{k=1}^{5} h_k^t \mathbf{\Omega}_k^{\mathcal{P}}\right] \mathbf{p}^e$$

siendo $\mathbf{p}^e$ el vector de salida del segundo bloque, supuesto constante. El sistema así descrito corresponde a un sistema borroso realimentado. Dado que la base de reglas para la construcción de las matrices $\mathbf{\Omega}_k^{\mathcal{P}}$ es simétrica, podemos reescribir la expresión de la siguiente forma

$$\mathbf{h}^{t+1} = \left[\sum_{k=1}^{5} p_k^e \mathbf{\Omega}_k^{\mathcal{P}}\right] \mathbf{h}^t$$

y tal como se ha demostrado anteriormente (sec. 4.4.2), este sistema evolucionará hacia el valor final

$$C = \frac{\sum_{j=1}^{5} c_j b_{jMAX}}{\sum_{j=1}^{5} b_{jMAX}}$$

siendo c_j los centroides de los conjuntos del espacio de salida, y $\mathbf{b}_{MAX}$ el autovector asociado al autovalor máximo de la matriz $\mathbf{\Omega} = \sum_{k=1}^{5} p_k^e \mathbf{\Omega}_k^{\mathcal{P}}$.

5.1.6 Ejemplo de funcionamiento

Como ejemplo, supóngase que en un determinado instante, se dispone de los siguientes datos para una hipótesis:

- Último vehículo observado: *Furgoneta*
- Probabilidad de la hipótesis: *Probable*

Tras una observación, los datos que llegan son los siguientes:

- Vehículo observado: *Camión*
- Error de observación: 0.45

Estos datos se pueden interpretar de forma vectorial del siguiente modo:

- Vehículo en pista $\mathbf{v}^p = [0,0,0,0,1,0]^T$ (correspondiente al gránulo *Furgoneta*).
- Vehículo observado $\mathbf{v}^o = [0,0,0,0,0,1]^T$ (correspondiente al gránulo *Camión*).
- Probabilidad de la hipótesis: $\mathbf{h}^t = [0,0,0,1,0]^T$ (correspondiente al conjunto *Probable*).
- Error de observación: $\mathbf{E}(e_{x_0}) = [0, 0.2, 0.625, 0, 0, 0]^T$

De este modo la matriz de la ecuación (5.16) pasa a ser

$$\mathbf{h}^{t+1} = \left[h_4^t \mathbf{\Omega}_4^{\mathcal{P}}\right] \left[E_2(\Delta e)\mathbf{\Omega}_2^{\mathcal{E}} + E_3(\Delta e)\mathbf{\Omega}_3^{\mathcal{E}}\right] \left[v_5^p \mathbf{\Omega}_5^{\mathcal{V}}\right] \mathbf{v}^o$$

y sustituyendo por los valores correspondientes (las matrices $\mathbf{\Omega}_i^{\mathcal{K}}$ $\mathcal{K} = \{\mathcal{P}, \mathcal{E}, \mathcal{V}\}$ se pueden consultar en el apéndice D.3).

$$\mathbf{h}^{t+1} = \left[\mathbf{\Omega}_4^{\mathcal{P}}\right] \left[0.2\mathbf{\Omega}_2^{\mathcal{E}} + 0.625\mathbf{\Omega}_3^{\mathcal{E}}\right] \left[\mathbf{\Omega}_5^{\mathcal{V}}\right] [0\ \ 0\ \ 0\ \ 0\ \ 0\ \ 1]^T$$

Operando se obtiene que $\mathbf{h}^{t+1} = [0, 0.0018, 0.1171, 1.2482, 0.3696]^T$, y normalizando por el valor máximo se convierte en $\mathbf{h}^{t+1} = [0, 0.0014, 0.0939, 1, 0.2961]^T$, que se corresponde con el conjunto de etiqueta *Probable*. Si se calcula un valor numérico usando el método de los centroides y tomando como vector de centroides $\mathbf{c}_{\mathcal{P}}$, se obtendrá una medida no borrosa de la probabilidad de esa pista:

$$P^{t+1} = \frac{[0.05\ \ 0.25\ \ 0.5\ \ 0.75\ \ 0.95]\mathbf{h}^{t+1}}{[1\ \ 1\ \ 1\ \ 1\ \ 1]\mathbf{h}^{t+1}} = 0.7752$$

Si las entradas permanecen constantes a los valores observados, es decir, como vehículo observado *Camión*, y como error 0.45, puede calcularse fácilmente el estado final del sistema. Para ello, primero se calcula la salida del bloque de compatibilidad física y lógica

$$\begin{aligned}\mathbf{p}^e &= \left[0.2\mathbf{\Omega}_2^{\mathcal{E}} + 0.625\mathbf{\Omega}_3^{\mathcal{E}}\right]\left[\mathbf{\Omega}_6^{\mathcal{V}}\right][0\ \ 0\ \ 0\ \ 0\ \ 0\ \ 1]^T \\ \mathbf{p}^e &= [0\ \ 0\ \ 0.0033\ \ 1.0908\ \ 0.2160]^T\end{aligned}$$

La matriz $\mathbf{\Omega}$ se calculará como

$$\begin{aligned}\mathbf{\Omega} &= \sum_{k=1}^{5} p_k^e \mathbf{\Omega}_k^{\mathcal{P}} \\ &= 0.0033\mathbf{\Omega}_3^{\mathcal{P}} + 1.0908\mathbf{\Omega}_4^{\mathcal{P}} + 0.2160\mathbf{\Omega}_5^{\mathcal{P}}\end{aligned}$$

El autovector asociado a su autovalor máximo vale

$$\mathbf{b}_{MAX} = [0.0001\ 0.4351\ 0.0527\ 0.2944\ 0.8493]^T$$

con lo que el valor hacia el que tiende el sistema cuando mantenemos las entradas constantes será

$$P^{\infty} = \frac{[0.05\ \ 0.25\ \ 0.5\ \ 0.75\ \ 0.95]\mathbf{b}_{MAX}}{[1\ \ 1\ \ 1\ \ 1\ \ 1]\mathbf{b}_{MAX}} = 0.7127$$

5.2 DETECTOR BORROSO DE MANIOBRA PARA UN FILTRO DE KALMAN EN ENTORNO DE SEGUIMIENTO ESTÉREO

5.2.1 Planteamiento del problema

En [5] puede encontrarse una aplicación de los sistemas realimentados expuestos en la sección 4.4 para un caso en el que no existen términos lingüísticos ni información basada en texto. Sin embargo, en este sistema se va a dar también una situación de ambigüedad en la información que va a hacer posible la aplicación de un filtro borroso realimentado.

Se desea controlar el valor de la matriz de covarianza de ruido de planta de un filtro de Kalman en un entorno de seguimiento visual estéreo. Por motivos de separación de las coordenadas espaciales se optó por trabajar con un esquema concreto: el Filtro de Kalman en Coordenadas Principales (FKCP). Este filtro está inspirado en la literatura de RADAR; se basa en la idea de realizar los cálculos para la ganancia Kalman y la actualización de la matriz de covarianza en el marco de referencia de las *direcciones principales* de la covarianza de las posiciones 3D medidas, en donde las coordenadas reconstruidas están incorreladas. La ganancia de Kalman se calcula en *coordenadas principales* y por lo tanto, la innovación debe transformarse a este nuevo marco de

referencia. Una vez multiplicada, se devuelve a coordenadas cartesianas 3D, que son usadas para extrapolar el estado. Estos cambios de base se realizan mediante dos matrices de transición **C2P** y **P2C**, construidas a partir de los autovectores de la matriz de covarianza de ruido del ruido de observación $\mathbf{R}(x, y, z)$ (los detalles se pueden consultar en el apéndice E).

Una aplicación que use el filtro de Kalman debe tener en cuenta dos cosas; por un lado la correcta estimación de la matriz de covarianza de ruido de planta y por otro la robustez del filtro en presencia de discrepancias respecto al modelo, como en el caso de que el objeto realice una maniobra [12, 19]. La elección de un valor demasiado alto para la varianza del ruido de planta hace que el filtro sea más impulsivo: responde mejor a las observaciones a costa de perder capacidad de suavizado. Por otro lado, un valor demasiado bajo hace que el filtro pierda la pista tan pronto como el móvil realiza un movimiento inesperado.

Para estimar la varianza de ruido de planta se ha optado por la filosofía ya usada en [26, 74], haciendo que ésta (y por consiguiente su matriz de covarianza, Q) sea variable y adaptativa al escenario, en función del error de predicción en el instante anterior y del incremento de dicho error. Esto permite al filtro tomar el valor de ruido más adecuado para cada caso. Puede encontrarse un estudio detallado del tema en [5, 112] y en el apéndice E.

Respecto a la robustez frente a maniobras, se ha optado por seguir la filosofía clásica de ajuste de parámetros [12]. Se monitorizará la historia del error de predicción del filtro, y, si los errores llegan a un estado estacionario, el filtro se reinicia. Para ello se empleará un FFS similar al usado en la aplicación de seguimiento de información basada en texto ya descrita (sección 5.1). En este caso la ambigüedad no es lingüística, sino que procede de la dificultad en la determinación precisa de la existencia de desacople en el filtro.

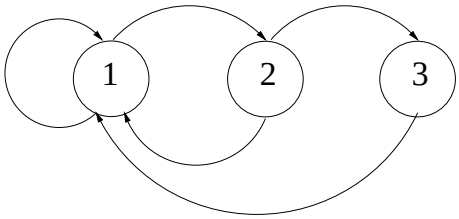
El desarrollo completo del filtro está fuera de la intención ilustrativa de este capítulo, por lo que a continuación se expondrá solamente la parte correspondiente al detector de maniobras, relegando el resto, como ya se ha comentado, al apéndice E.

5.2.2 Detector borroso de maniobra

En caso de que una maniobra repentina haga desacoplarse al filtro de tal manera que no sea posible volver a un buen funcionamiento en un espacio corto de tiempo ampliando al máximo la varianza de ruido, será necesario proveer al sistema de mecanismos de corrección. Siguiendo la filosofía clásica de control de maniobras [12] se ha optado por lo siguiente: en caso de sutil cambio de la trayectoria, el filtro se recupera modificando su matriz de covarianza (apéndice E). En caso de que el error sea excesivo, el filtro debe reiniciarse.

Para este control se ha usado un FFS (véase diagrama de estados en la Fig. 5.8) que almacena una historia del error, de tal manera que si un error excesivo permanece, el

filtro se reinicie, pero no lo haga en caso de errores grandes aislados. Esta filosofía de diseño se basa en el bloque de historia de probabilidades de la aplicación descrita en la sección 4.4.2 y en [3]. La posibilidad de que el filtro de Kalman se desacople debido a un cambio de trayectoria, puede modelarse como un proceso borroso que da lugar a un FFS con entradas borrosas, del que es posible conocer su comportamiento a priori.



BIBLIOTECA VIRTUAL

Figura 5.8 Diagrama de estados para el reinicio

El funcionamiento del filtro está resumido en el diagrama de estados de la figura 5.8. El primer estado corresponde a *no-reinicio* y el tercer estado a *reinicio*. El segundo es un estado de transición. Para realizar el paso entre estados se ha usado un sistema borroso realimentado como el de la figura 5.9.

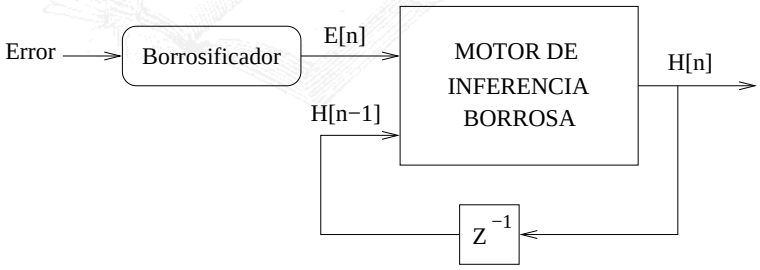


Figura 5.9 Sistema borroso realimentado

Tanto las entradas como las salidas son conjuntos borrosos. Se ha usado un sistema de activación de reglas basado en suma y producto y una operativa SAM para la inferencia. De este modo se asegura un correcto funcionamiento del sistema, tal como se demostró en [4]. Las variables de entrada serán el Error de predicción normalizado (figura 5.10) y la Historia de predicciones hasta ese instante (figura 5.11); la salida será la Historia en el instante siguiente. Como se ha indicado, los detalles pueden encontrarse en el apéndice E.

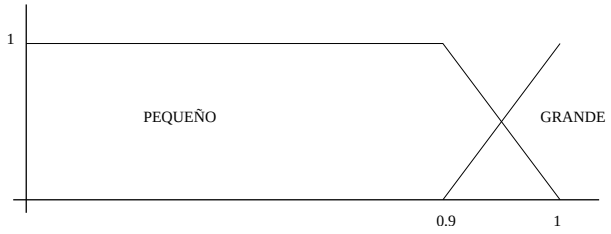


Figura 5.10 Funciones de pertenencia de la variable *Error*

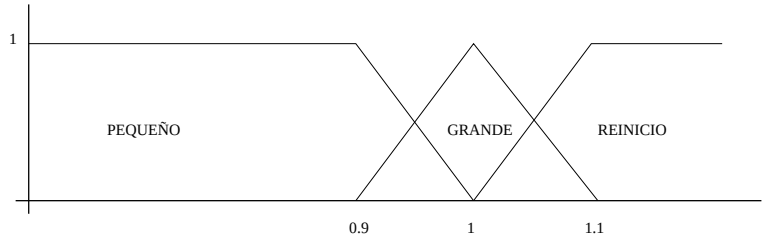


Figura 5.11 Funciones de pertenencia de la variable *Historia*

La matriz que contiene las reglas del sistema es la siguiente:

		Historia		
		Pequeño	Grande	Reinicio
Error	Pequeño	Pequeño	Pequeño	Grande
	Grande	Grande	Reinicio	Reinicio

Cuando el valor no borroso correspondiente al centroide de salida (historia del error) alcanza un cierto umbral, el filtro se reinicializa. El umbral se ha fijado en la unidad, que se corresponde con el valor en el que el conjunto de salida *Reinicio* empieza a tomar valores no nulos (Fig. 5.11).

5.2.3 Ejemplos prácticos y resultados

Se han realizado 4 experimentos. Se supondrá que el móvil se desplaza con movimiento rectilíneo uniforme, y que, en un momento dado (que en todos los casos coincidirá con el instante de tiempo $t = 0.52s$.) el móvil realiza un cambio de dirección o de velocidad abrupto. Para todos los experimentos se ha considerado un ruido de observación con desviación típica $\sigma_v = \sigma_u = \sigma = 10^{-4}m$, y unas distancias $b = 0.1m$ y $f = 0.0505m$ (ver figura E.1 en el apéndice E) según las restricciones planteadas en [5, 112]. En todos los experimentos se proporcionan estimaciones del error de filtrado a la salida del filtro obtenidas a partir de 1000 iteraciones de Monte Carlo. Los filtros *convencionales* utilizan por defecto una desviación típica de ruido de maniobra $\sigma_w = 0.05m/s^2$. El movimiento se registra con una frecuencia de muestreo de 50 *Hz*. La eficiencia del filtro en cada instante de tiempo se mide por medio de la raíz del error cuadrático medio total (TRMSE), que se calcula como la raíz cuadrada de la suma del error cuadrático medio de cada iteración de Monte Carlo para cada instante de tiempo.

Las cuatro trayectorias son (Fig. 5.12):

- *Trayectoria T1:* Maniobra leve. El punto inicial es $\mathbf{r_0} = (0.1\ 0.1\ 0.1)$. El móvil se mueve con movimiento rectilíneo uniforme con velocidad $\mathbf{v_0} = (1\ 1\ 0)$. En el instante de tiempo $t = 0.52$, el móvil realiza una maniobra y la nueva velocidad es $\mathbf{v_1} = (0.93\ 0.93\ 0.1)$.

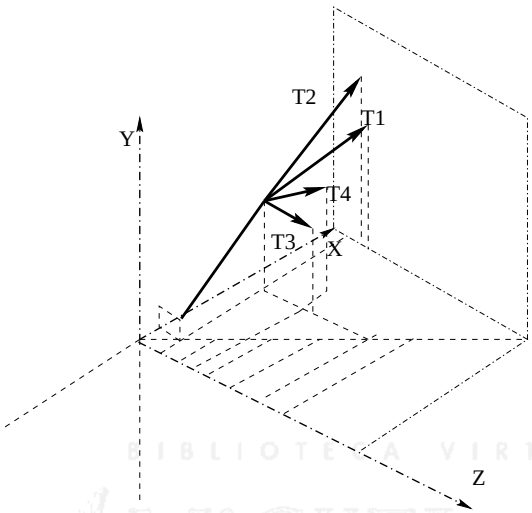


Figura 5.12 Trayectorias del experimento

BIBLIOTECA VIRTUAL MIGUEL DE CERVANTES

- *Trayectoria T2:* Maniobra muy leve. El punto inicial es $\mathbf{r_0} = (0.1\ 0.1\ 0.1)$. El móvil se mueve con movimiento rectilíneo uniforme con velocidad $\mathbf{v_0} = (1\ 1\ 0)$. En el instante de tiempo $t = 0.52$, el móvil realiza una maniobra y la nueva velocidad es $\mathbf{v_1} = (0.95\ 0.95\ 0.05)$.
- *Trayectoria T3:* Maniobra abrupta. El punto inicial es $\mathbf{r_0} = (0.1\ 0.1\ 0.1)$. El móvil se mueve con movimiento rectilíneo uniforme con velocidad $\mathbf{v_0} = (1\ 1\ 0)$. En el instante de tiempo $t = 0.52$, el móvil realiza una maniobra y la nueva velocidad es $\mathbf{v_1} = (0\ 0\ 1)$.
- *Trayectoria T4:* Maniobra abrupta. El punto inicial es $\mathbf{r_0} = (0.1\ 0.1\ 0.1)$. El móvil se mueve con movimiento rectilíneo uniforme con velocidad $\mathbf{v_0} = (1\ 1\ 0)$. En el instante de tiempo $t = 0.52$, el móvil realiza una maniobra y la nueva velocidad es $\mathbf{v_1} = (0.5\ 0.5\ 0.5)$.

Los resultados se recogen en la figura 5.13. En *T1* se puede ver que la adaptación de la matriz de covarianza hace que el filtro sea capaz de seguir esa pequeña variación, que a la larga se va a convertir en un error de predicción grande. La mejora del error de salida es muy considerable. En *T2*, a pesar de que el filtro borroso también alcanza un nivel de error alto, la adaptación de la matriz de covarianza hace que el filtro se vuelva a acoplar.

En los dos últimos casos, *T3* y *T4* el cambio de trayectoria hace que el filtro se desacople totalmente, y que el error aumente considerablemente. Sin embargo, en el caso del filtro con control borroso, al detectar este aumento, el filtro se reinicia, logrando así engancharse de nuevo.

Para ver con mayor exactitud qué ocurre alrededor del punto $t = 0.52$, se representa en la figura 5.14 el error de predicción y el error de salida para una iteración aleatoria.

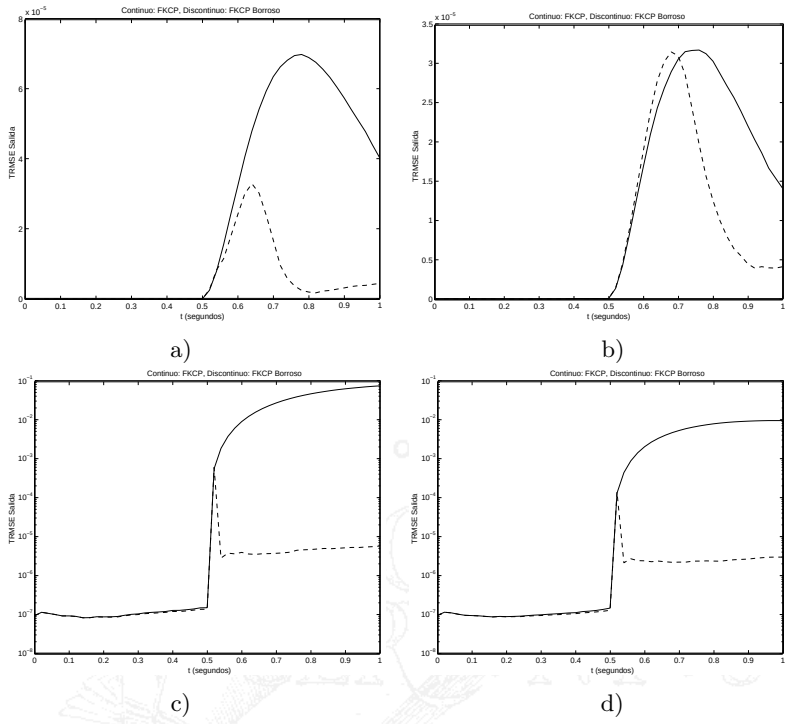


Figura 5.13 Comparación del FKCP convencional con el filtro con control borroso. a) Trayectoria T1 b) Trayectoria T2 c) Trayectoria T3 d) Trayectoria T4

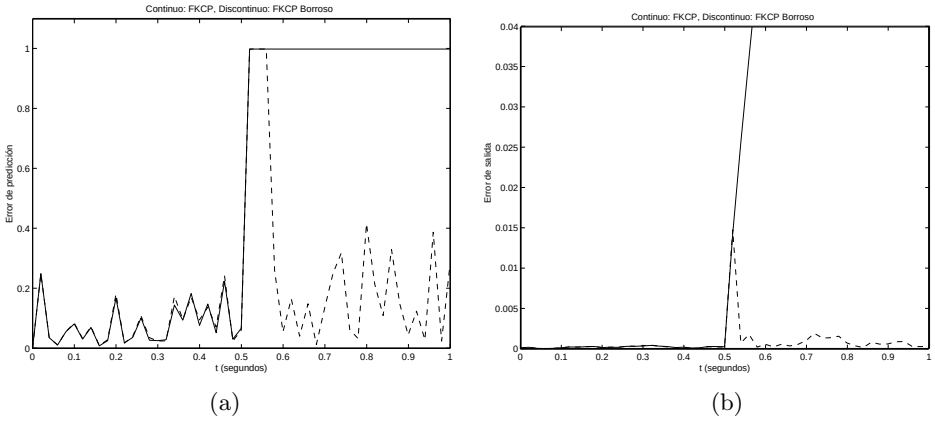


Figura 5.14 Comparación del FKCP convencional con el filtro con control borroso. a) Error de Predicción b) Error de Filtrado

En la figura 5.14-a, se puede ver que al realizar la maniobra, en el filtro convencional el error de predicción sube a 1, y a partir de aquí el error de salida (figura 5.14-b) se incrementa cada vez más. Sin embargo, en el filtro con control borroso, al detectar un error grande persistente, el filtro de reinicia, con lo que el error de predicción disminuye, y el error de salida también.

Comparación con un detector de maniobra clásico

Dotamos ahora al filtro de Kalman clásico con un detector de maniobra consistente en monitorizar el error, y cuando éste excede un cierto valor umbral, el filtro se reinicializa. Para poder comparar mejor en los experimentos posteriores se ha fijado el valor del umbral en el mismo valor que el fijado en el filtro borroso, es decir, el filtro se reinicia cuando el error normalizado $|\mathbf{E}_p[n]| \geq 1$.

Los filtros se han comparado con la trayectoria $T4$. Los resultados se muestran en la figura 5.15.

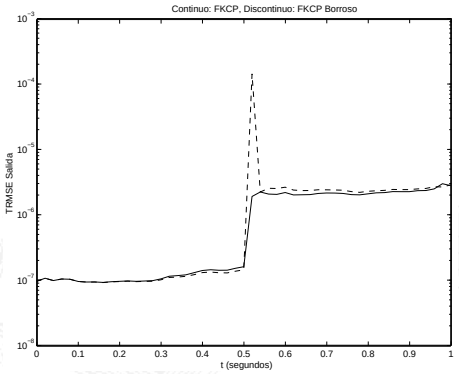


Figura 5.15 Comparación de los decisores de maniobra borroso y clásico. Trayectoria T4 (log)

En la figura se puede ver la consecuencia del empleo de un FFS en el diseño: el filtro borroso tiene una cierta inercia debida al tiempo de permanencia en el estado 2 antes de ser reiniciado. Esto hace que el error presente un pico que no aparece en el filtro clásico. Tras la reinicialización, ambos filtros tienen un comportamiento parecido. Pudiera parecer que este "pico" hace al filtro con control borroso peor. Entendemos que no es así; esta *inercia* es el precio que hay que pagar para hacer más robusto al filtro y evitar constantes reinicios de éste.

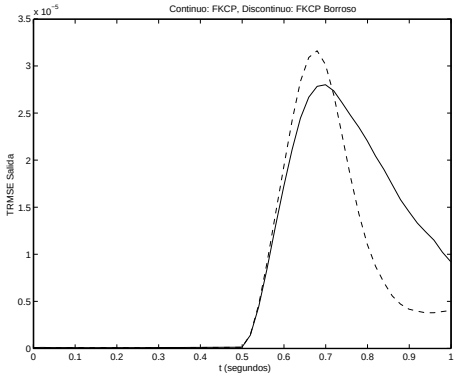
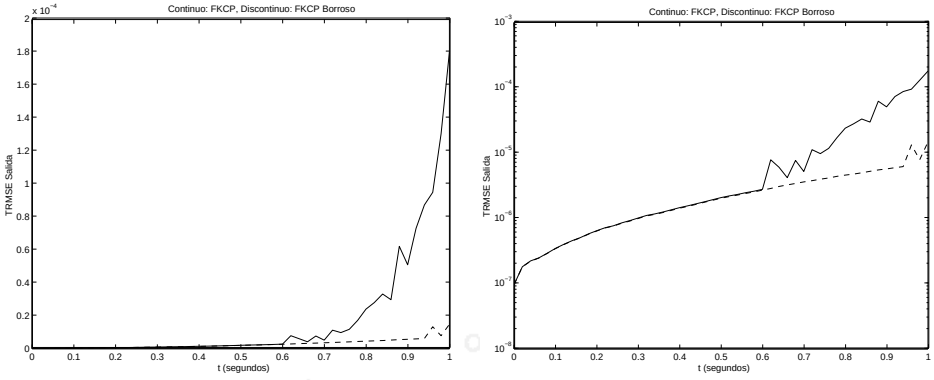


Figura 5.16 Comparación de los decisores de maniobra borroso y clásico. Trayectoria T2

La Fig. 5.16 muestra una comparativa en caso de una maniobra ligera. De nuevo el filtro borroso presenta cierta inercia, pero se recupera mejor que el filtro clásico.



a) b)

Figura 5.17 Comparación de los decisores de maniobra borroso y clásico. a) Trayectoria T6 b) Trayectoria T6 (log)

Como ya se ha comentado, esta inercia que aparece en el filtro borroso es necesaria para evitar reinicios innecesarios en caso de errores aislados. En la figura 5.17 se muestra un caso en el que no hay maniobra, pero hay un alto nivel de ruido (la trayectoria T6 está definida en el apéndice E). El filtro clásico reinicia el filtro una y otra vez, mientras que el controlador borroso detecta estos errores como aislados, y no llega a reiniciar, realizando exclusivamente una adaptación de la matriz de covarianza.

6

APLICACIÓN AL CÁLCULO AUTOMÁTICO DEL GRADO DE MADURACIÓN ÓSEA EN LA INFANCIA

BIBLIOTECA VIRTUAL MIGUEL DE CERVANTES

6.1 INTRODUCCIÓN

La determinación del estado de maduración esquelética en la infancia constituye en la actualidad un procedimiento de gran importancia en la detección y seguimiento de trastornos relacionados con el crecimiento. Habitualmente, esta determinación se realiza a partir de radiografías de la mano y muñeca no dominantes. Debido a la importancia de este procedimiento y a que los métodos actuales requieren la intervención de un especialista, es de gran interés el desarrollo de un método automático para la determinación de la edad ósea, el cual pueda ayudar a los especialistas en su tarea.

El problema de automatizar un método, de tal forma que a partir de una radiografía digital un sistema experto sea capaz de calcular la edad ósea, no es tarea fácil. De todas las posibles fases de implementación en el presente capítulo se mostrará la parte final, correspondiente a un clasificador: a partir de una serie de características de entrada que se supone se pueden conseguir, se obtiene un valor para el grado de maduración ósea del paciente.

Existen dos métodos para realizar este cálculo: el de Greulich-Pyle [43] y el de Tanner-Whitehouse [124]. Este último es más preciso, pero también más laborioso, lo que limita su utilización por parte de los especialistas. Como filosofía de diseño se ha decidido implementar directamente los conceptos *lingüísticos* (no numéricos) del método Tanner-Whitehouse (que se expondrá a continuación) haciendo uso de las herramientas descritas en el capítulo 4.

6.2 CÁLCULO DE LA MADURACIÓN ÓSEA EN LA INFANCIA

La valoración del grado de maduración ósea es un proceso usado con frecuencia dentro del campo de la radiología pediátrica. Basado en un examen radiológico del desarrollo de los huesos de la mano y la muñeca izquierdas (no dominantes, en general), se puede obtener así la edad ósea del paciente y compararla con la edad cronológica. Una

discrepancia entre estos dos valores indica anormalidades en el desarrollo óseo de dicho paciente.

Sin embargo no hay un método aceptado por todos los radiólogos como estándar. La mayoría (un 76%) usa el método de Greulich-Pyle [43], basado en la comparación con patrones incluidos en un atlas. Por lo tanto es de naturaleza subjetiva y puede producir diferentes resultados para distintos observadores. Otra técnica muy usada es el método Tanner-Whitehouse (TW2)[1] [124], basado en un análisis detallado de cada hueso, el cual clasifica en un estadio de desarrollo. De este modo, cada hueso obtiene unas puntuaciones, que ponderadas debidamente, permiten obtener la edad ósea total del paciente. Aunque la clasificación de cada hueso puede seguir siendo algo subjetiva, debido a la naturaleza visual del proceso, en su conjunto proporciona una mayor precisión que el anterior. Tiene el problema de ser mucho más complicado y laborioso, por lo que sólo se usa en el 20% de los casos. El tiempo empleado en la inspección manual de las radiografías es excesivo y, además, requiere de un facultativo especialista, habida cuenta de la especifidad del proceso.

De todo lo anterior surge la necesidad de un método de diagnóstico apoyado en ordenador. Este método bien podría ser una implementación práctica del método TW2 o un nuevo planteamiento basado en reconocimiento de patrones [55] a partir de las características de cada radiografía. Varios esfuerzos en esta dirección se han realizado en los últimos años, tal como se verá posteriormente.

6.2.1 Radiografía de la mano

Debido a que la gran mayoría de los métodos de determinación de la edad ósea utilizan como fuente de información la radiografía de la mano, es de interés un pequeño estudio óseo de ésta.

En primer lugar, es importante aclarar el significado de dos términos que serán de gran importancia en adelante. Se trata de la *epífisis* y la *metáfisis* de los huesos. Como puede verse en la Fig. 6.1, en los huesos largos se puede distinguir entre la diáfisis, la metáfisis y la epífisis. La diáfisis es la parte central del hueso, la cual no es de especial interés en este caso. La epífisis (en el extremo de cada hueso) parte de un centro de osificación independiente, y una vez completado el proceso de crecimiento acabará por fusionarse con la metáfisis (parte ancha al final del hueso).

En cuanto a la mano, pueden distinguirse tres zonas principales: la muñeca, una parte cuadrangular intermedia y los cinco dedos. El esqueleto óseo de estas tres partes está constituido por el carpo, el metacarpo y las falanges, respectivamente. En la Fig. 6.3 puede verse un esquema del esqueleto de la mano. La estructura ósea de cada una de estas zonas es como sigue:

[1] El nombre TW2 viene de ser la segunda edición del citado método. Recientemente se ha publicado la tercera edición, conocida como TW3 [122], con ligeros cambios respecto a las anteriores, pero manteniendo el mismo método.

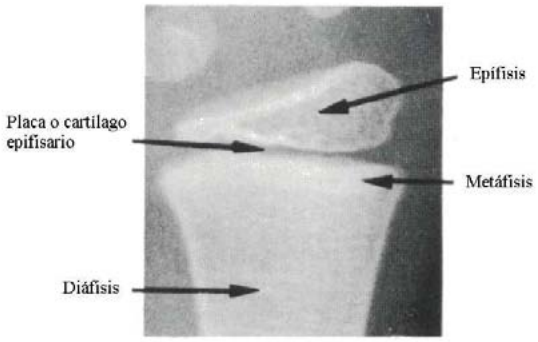


Figura 6.1 Nomenclatura de las áreas epifisarias

BIBLIOTECA VIRTUAL MIGUEL DE CERVANTES

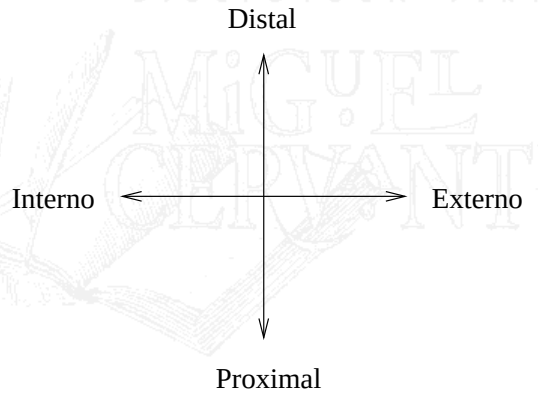


Figura 6.2 Nomenclatura de las direcciones para los huesos de la mano

- En el carpo hay 8 huesos dispuestos en dos filas de cuatro huesos cada una: escafoides, semilunar, piramidal y pisiforme en la fila proximal, y trapecio, trapezoide, hueso grande y hueso ganchoso en la fila distal (ver direcciones en la Fig.6.2). Los huesos del carpo no están situados todos en el mismo plano, sino que forman en conjunto una superficie en forma de canal abierto en sentido distal. El espacio que queda así configurado recibe el nombre de túnel carpiano.
- El metacarpo está formado por los 5 metacarpianos, que parten divergentes desde el carpo. Todos ellos tienen la morfología de un hueso largo, con una epífisis proximal, una diáfisis casi cilíndrica y una epífisis distal o cabeza que sirve de articulación al esqueleto de cada uno de los dedos.
- En los dedos hay 14 huesos llamados falanges. Excepto el pulgar, que tiene dos, los demás dedos tienen tres falanges cada uno, que se denominan falange proximal, media y distal

Es necesario asimismo mencionar los huesos del antebrazo, radio y cúbito, que también aparecen en las radiografías de la mano y son tenidos en cuenta en el estudio de la maduración ósea. En el extremo distal de ambos huesos, esto es, en la muñeca,

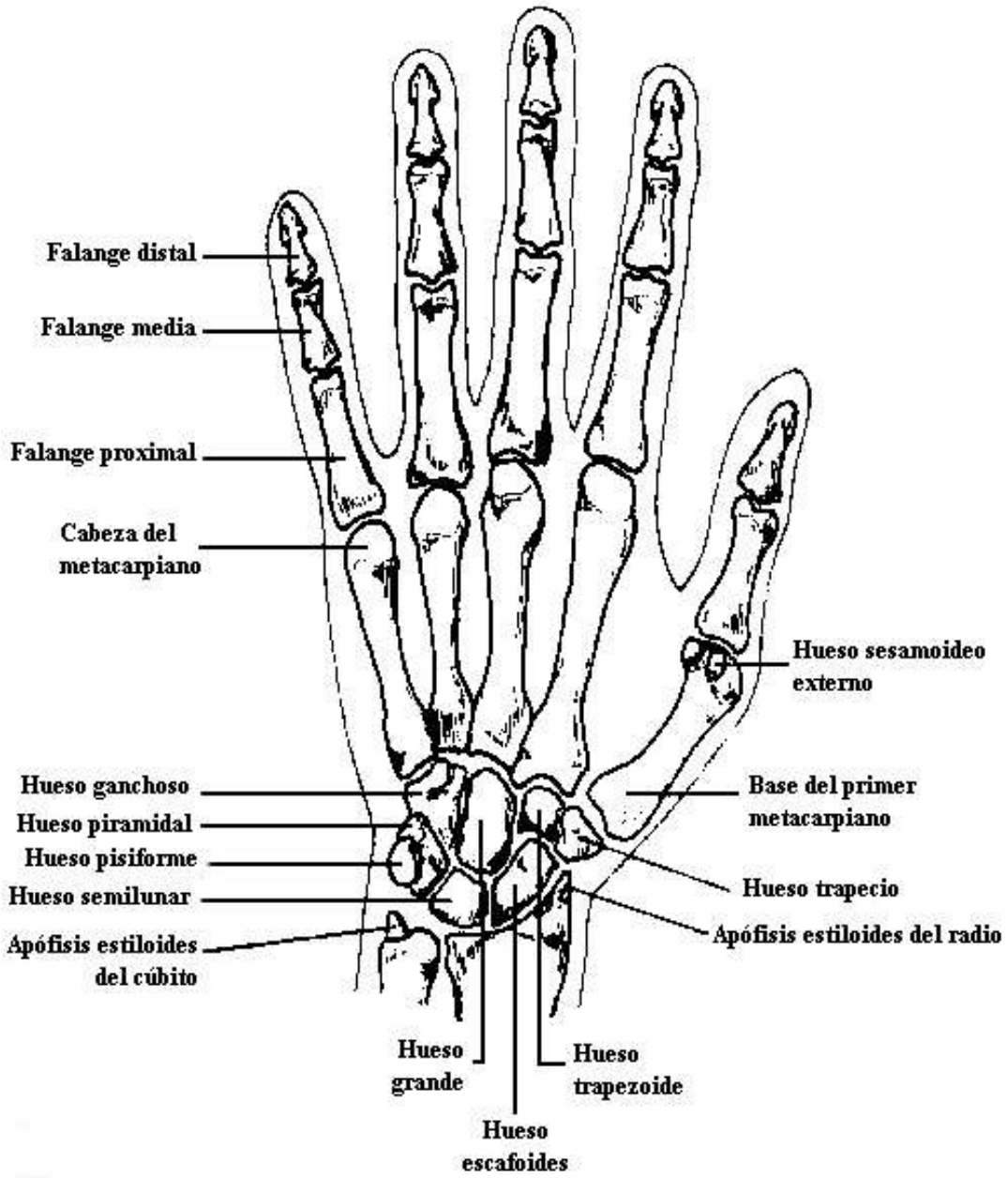


Figura 6.3 Esquema del esqueleto de la mano y muñeca.

radio y cúbito presentan las llamadas apófisis estiloides del radio y del cúbito, que son dos salientes afilados.

6.3 AUTOMATIZACIÓN DEL MÉTODO TW2

6.3.1 Trabajos previos

A principios de los 80 surgen los primeros intentos de lograr un proceso automatizado, que apoye de distintas maneras el trabajo de los especialistas. A. Pathak, S. K. Pal y R.A. King plantean el problema como un problema de clasificación. Paralelamente, desarrollan una serie de trabajos sobre el análisis de radiografías [73, 95, 96, 97] y clasificadores borrosos y sintácticos aplicados al campo de la maduración ósea [99, 100]. En [100] se presenta la implementación de un clasificador basado en Gramáticas Borrosas. Parten de la hipótesis de que se ha realizado un preprocesado de las imágenes y existe un conjunto de primitivas ya extraídas (puntos, curvas y rectas). Como características de entrada al clasificador usan los contornos de las epífisis y metáfisis. Estas características sirven para clasificar los huesos dentro de un estadio de desarrollo, siguiendo la filosofía del método TW2. La elección de un clasificador basado en gramáticas se debe, según los autores, a que los patrones a estudiar poseen más información en las estructuras que los forman (curvas, segmentos...) que en los valores numéricos (tamaños, distancias). De este modo, el clasificador tiene en cuenta las formas más que los tamaños, dando especial importancia al desarrollo de la epífisis. Este trabajo es de los pocos en este campo que justifica la elección de las características y del clasificador, adaptándose al problema real. Pese a ello, como punto débil tiene el hecho de usar la información de un solo hueso (el radio).

Existe otra línea de investigación sobre el tema, desarrollada por Ewa Pietka y sus colaboradores. Teniendo como fondo también el método TW2, su aproximación es ligeramente distinta. El planteamiento del problema está enfocado como un clásico problema de reconocimiento de patrones [55], en el que realiza una etapa de *preprocesado*, selección y extracción de características [81] y por último clasificación. Dada la naturaleza del patrón a analizar (imágenes de la mano izquierda obtenidas a partir de radiografías) divide su trabajo en dos partes claramente diferenciadas: estudio de los huesos de las falanges y estudio de los huesos del carpo. Esta división se debe a las diferencias importantes entre las dos zonas a estudiar, y ha sido tenida en cuenta por la mayoría de los autores.

El primer trabajo publicado [106] se centra en la extracción de características de los huesos de las falanges. En una primera etapa de procesado, se elimina el fondo no expuesto de la radiografía (mediante el cálculo de la desviación estándar) y se rota un cierto ángulo (permintiendo sólo los valores $\{\pi/2, -\pi/2, \pi\}$) para lograr tener todas las radiografías con una misma orientación. De una forma sencilla obtiene la región de interés y mediante técnicas de procesado de imágenes (gradiente de *Sobel* y umbrales) realiza una extracción de bordes de los huesos. Las medidas en este primer artículo se realizan únicamente sobre el dedo medio (falange III) pero en una versión definitiva [102] se realizan sobre las falanges II, III y IV (dedos índice, medio y anular). Las características a extraer son una serie de cocientes basados en medidas de las epífisis (3 falanges, 3 zonas por falange, 1 cociente por zona; 9 características).

El análisis de la zona del carpo [105] es ligeramente más complejo. Además, estos huesos sólo proporcionan información útil de la edad ósea hasta los 9-12 años. El preprocesado realizado es similar al que se realiza para la zona de las falanges (eliminación del fondo no expuesto y giro de la radiografía). La complejidad aparece en la extracción de la zona de interés. En un primer lugar la zona viene marcada por un rectángulo, calculado a partir de ciertos límites (unión del tejido del pulgar y el índice, etc). Se realiza una segmentación por umbrales (*thresholding*) en dos etapas (una grosera y otra dinámica) para separar los huesos del tejido blando. El siguiente paso es la eliminación de los huesos que no entran en el análisis. Para ello se usan ciertas técnicas morfológicas. En un primer momento se eliminan los huesos que tocan los bordes de la región de interés (cúbito, radio y parte de las falanges). Tras un cierto post-procesado (que incluye operaciones morfológicas de dilatación-intersección) se logra un conjunto de objetos independientes sobre fondo negro. Sobre ellos se realizan una serie de medidas (área, perímetro, tasa de compacidad, centro de gravedad, coeficiente de convexidad...) que permitirán saber cuáles de estos objetos pertenecen a huesos del carpo y cuáles deben ser eliminados.

Una vez seleccionados los huesos de interés es el momento de seleccionar el conjunto óptimo de características. Siguiendo un criterio estadístico que cuantifica la importancia de cada una de ellas, las escogidas por los autores son: área, perímetro, coeficiente de convexidad, número de huesos del carpo y *cociente 2* (ponderación del área del hueso por el área de todos los huesos del carpo). A pesar de todo un desarrollo en el que justifica la utilización de estas 5 características, en [102] se decide utilizar sólo 3 por hueso.

En [102] se parte de todo el trabajo previo para definir un clasificador basado en lógica borrosa que sea el encargado de realizar la clasificación. Se realiza un clasificador para cada región de interés; de este modo se obtiene una medida de la edad ósea para la zona del carpo y otra para la zona de las falanges. Es un radiólogo el que debe decidir la edad final. Para la definición de estos clasificadores, se hace mucho uso del conocimiento del dominio en que se encuentra, tendiendo hacia una solución demasiado heurística. Finalmente, en [14] aparece una descripción de una implementación real de este sistema. La tasa de aciertos (según [102]) es de 75% para la zona de las falanges y del 63% para el carpo.

Existen otras aproximaciones al tema, como [118]. Basándose en los trabajos de Pietka, plantean un método de análisis basado en visión artificial (extracción de bordes por Sobel y posterior adelgazamiento). Se limitan a la zona de las falanges, eligiendo como conjunto de características la longitud, anchura, perímetro y área de cada hueso (para esto último asumen que los huesos tienen forma elíptica). La clasificación la hacen interpolando los valores obtenidos en unas tablas. Al-Taani *et al.* [6] plantean un clasificador basado en modelos de distribución de puntos, mientras que Wastl y Dickhaus [140] desarrollan un sistema automatizado basado en prototipos de la representación espectral, obteniendo aciertos en un rango del 65% al 80%. Efford [37] propone una automatización directa del método TW2, dedicando la mayor parte de su esfuerzo a lograr una segmentación basada en modelos de cada uno de los huesos de interés.

Existe también una corriente dentro de los especialistas médicos que pretende encontrar un método automatizado para el cálculo de la maduración ósea. Es el caso de Suanes y Cañete [115], que realizan una clasificación estadística basada en descriptores de Fourier; Hill y Pynsent [58], quienes realizan un TW2 automatizado (*Royal Orthopedic Hospital Skeletal Ageing System*), con un rechazo del 25%, y de Tanner y Cols con el método CASAS [125] (*Computer Assisted Skeletal Assessment Scores*), método semiautomático basado también en Fourier. El problema de todos estos métodos (además de usar aproximaciones muy simplistas) es que tienen una reproductibilidad baja y precisan de radiografías de muy buena calidad para poder funcionar.

6.3.2 Una nueva aproximación

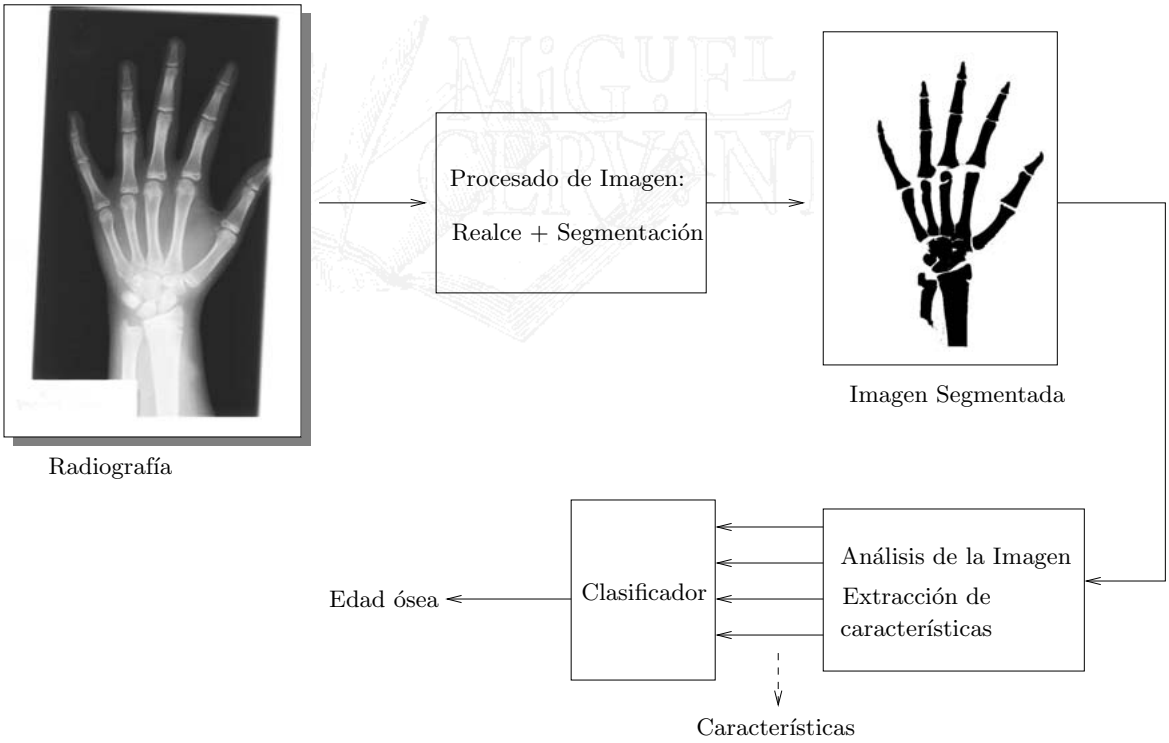


Figura 6.4 Esquema del proceso de automatización del método TW2

Las soluciones técnicas comentadas en el apartado anterior tienen en común que primero extraen un conjunto de características a partir de la radiografía de la mano, para luego crear un clasificador a partir de estas características, usando técnicas clásicas de aprendizaje (Fig. 6.4). De este modo, la construcción del clasificador estará limitada por las características elegidas.

Por otro lado, para lograr un correcto aprendizaje es necesario disponer de una gran cantidad de datos de entrenamiento. Sin embargo, en entornos clínicos reales, esta gran cantidad de datos no siempre está disponible para todas las edades. Otro problema inherente al cálculo de la maduración ósea es la precisión. Se necesita una gran precisión

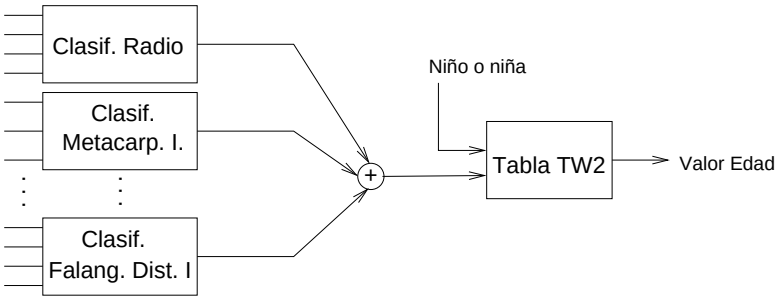


Figura 6.5 Clasificador global formado por un microclasificador para cada hueso

en el proceso de estimación de la edad ósea, lo que lleva a una gran cantidad de posibles estados de salida en el clasificador, si se suponen valores discretos de salida. Un proceso alternativo va a ser considerar *microclasificadores*, en los que se calcula el estado de maduración ósea de cada hueso, de manera similar a como se hace en el TW2 (Fig. 6.5).

Por lo tanto, en el presente capítulo se va a proponer la construcción de un clasificador para TW2 basado en el propio método. Dado que las descripciones que allí aparecen son de carácter lingüístico, se hará uso de la teoría para el procesado de información no numérica descrita en los capítulos 3 y 4. El uso directo de las conclusiones del método TW2 implica que se está usando toda la información contenida en el método, es decir, las 6000 radiografías empleadas por los autores. De este modo no es necesario el uso de un número muy alto de ejemplos para el entrenamiento, simplemente pueden necesitarse un pequeño número de radiografías para validar o refinar los resultados. En este caso se han usado 57 radiografías de niños para refinar y 85 de niñas para validar.

6.4 CONSTRUCCIÓN DE UN CLASIFICADOR PARA TW2

Lo primero a tener en cuenta a la hora de construir el clasificador es el número de muestras de las que se dispone para crear el conjunto de ejemplos. Dado que el número de radiografías disponibles es mucho menor que el que sería necesario para entrenar cualquier clasificador, se va a hacer uso de la experiencia acumulada en el método TW2. Para la creación de éste, se ha trabajado con 3000 niños y 3000 niñas. Si de alguna manera se utilizan las conclusiones por ellos extraídas, sería equivalente a la utilización de ese número de muestras en nuestro conjunto de test.

Por ello, la filosofía seguida ha sido la de clasificar cada hueso por separado en su estadio de desarrollo (A, B,... I). De este modo se tendrá un número discreto y pequeño de clases de salida (9 en la mayoría de los casos) haciendo el método universal y adaptable a niños y niñas (sólo variará la puntuación que se asigna a cada estadio). Los distintos huesos son ponderados según el esquema utilizado y luego, por medio de una función, se asigna su edad ósea.

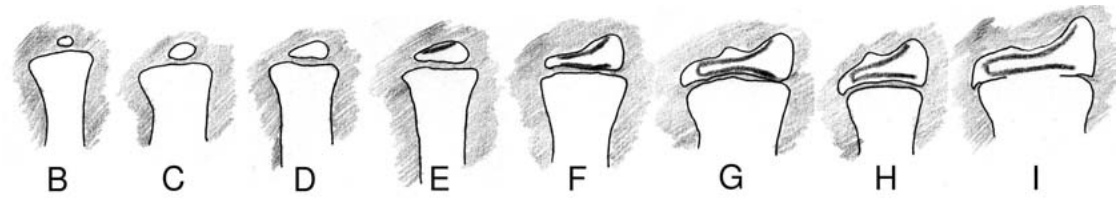


Figura 6.6 Estadios TW2 para el radio

Dentro del TW2 hay dos posibles esquemas de análisis. Uno de ellos, el RUS, hace uso de 13 huesos (las falanges, el cúbito y el radio), mientras que el otro, el TW2 propiamente dicho, usa 20 huesos (los 13 anteriores más los 7 huesos del carpo). Se ha optado por implementar el método RUS, por no usar los huesos del carpo, de los que es difícil extraer características de forma automática y que, por otro lado, dejan de tener validez en el método a partir de cierta edad.

6.4.1 Definición de características. Ejemplo del radio

A modo de ejemplo, se va a realizar un estudio detallado de las características presentes en el método para la clasificación del radio en su estado de maduración. Los demás huesos han seguido un razonamiento similar.

Según el método TW2 [124] los estadios en los que se clasifica el radio son los siguientes (Fig. 6.6):

Estadio A 1. No existe centro epifisario

Estadio B 1. El centro es apenas visible como un único depósito de calcio o más raramente en forma de depósitos múltiples. El borde se halla mal definido.

Estadio C 1. El centro tiene un aspecto distinto y es de forma oval con un borde liso y continuo.

Estadio D 1. El diámetro máximo tiene una anchura igual o superior a la mitad de la epífisis.

2. La epífisis se ha ensanchado principalmente en su cara lateral, con lo que esta porción es más ancha y redondeada y la porción interna está más ribeteada.
3. El tercio medio de la superficie proximal es plano y ligeramente ensanchado, y la discontinuidad entre éste y la metáfisis radial se ha estrechado aproximadamente 1mm.

Estadio E 1. Ha aparecido una línea blanca densa justo dentro del borde distal de la epífisis. Ello representa el borde de la superficie palmar y el hueso que aparece de nuevo en situación distal a lo anterior es el borde de la superficie dorsal.

Estadio F 1. El borde proximal de la epífisis se halla ahora diferenciado en las superficies palmar y dorsal. La superficie palmar se halla visible formando una línea blanca amplia e irregularmente densa en el borde proximal de la epífisis.

2. Los extremos de la epífisis, aunque de modo especial el interno, han crecido hacia afuera y aproximadamente desde el último estadio, con lo que el borde proximal se adapta ahora a la forma de la metáfisis en la mayor parte de su recorrido.

Estadio G 1. En la superficie dorsal se distinguen ahora los bordes articulares diferenciados del escafoides y el semilunar unidos a una pequeña giba.

2. El borde interno de la epífisis ha desarrollado superficies palmar y dorsal para la articulación con la epífisis cubital. La superficie que se proyecta internamente puede ser tanto la palmar como la dorsal, dependiendo de la posición de la muñeca.

3. El borde proximal de la epífisis es ahora ligeramente cóncavo.

Estadio H 1. La epífisis ahora corona la metáfisis en un punto (generalmente interno) a ambos lados.

Estadio I 1. Ha empezado la fusión de la epífisis y metáfisis. Puede ser todavía visible una línea constituida en parte por áreas oscuras en las que aún existe cartílago epifisario y en parte por áreas blancas densas en las que tiene lugar la fusión, o bien, la línea puede haber desaparecido.

De esta descripción textual se pueden obtener 6 características, que cubren todo el espectro del método:

Presencia: Existencia o no del centro epifisario. En caso de no existir, el estadio es A. Si existe pero es pequeño y apenas visible, el estadio es el B. Si existe y es bien visible, los estadios son C, D, E, F, G, H, I.

Separación: Posición relativa entre epífisis y metáfisis: pueden encontrarse separadas (estadios B, C, D, E, F, G), puede ocurrir que la epífisis corone a la metáfisis en un punto (estadio H) y puede haber comenzado la fusión (estadio I).

Forma I: En caso de existir la epífisis, su forma puede ser oval (estadio C) o "picuda" (estadios D, E, F, G, H, I).

Diámetros: Relación entre el diámetro de la epífisis y la metáfisis.

Forma II: En caso de ser "picuda", la epífisis puede tener el borde liso (estadios D y E), adaptado a la forma de la metáfisis (estadio F), o con una pequeña giba producida por las articulaciones del escafoides y semilunar (estadios G, H, I).

Superficies: Esta característica trata de representar la aparición de superficies interiores (palmar y dorsal) en la epífisis. Pueden estar ausentes (estadios B, C, D), aparecer como una línea densa (estadio E), como dos líneas (estadio F) o pueden apreciarse ya las superficies como una línea en forma de c (estadios G, H, I).

Estas características, puestas en forma de tabla para cada estadio quedan de la siguiente forma:

Estadio	Presencia	Separación	Forma I	Diámetros	Forma II	Superficies
A	No					
B	Poco	(sí)	(oval)	> 2		(no)
C	Sí	sí	oval	> 2		no
D	Sí	sí	picuda	≤ 2	plano	no
E	Sí	sí	picuda	≤ 2	plano	1 línea
F	Sí	sí	picuda	≤ 2	adaptada	2 líneas
G	Sí	sí	picuda	≤ 2	bases	forma de c
H	Sí	corona	picuda	≤ 2	bases	forma de c
I	Sí	fusión	picuda	≤ 2	bases	forma de c

Se puede observar que en la tabla hay características que son excluyentes entre sí. Debido a que técnicamente es siempre deseable tener el mínimo número de características (por razones de eficiencia y precisión) se va a realizar una *fusión* de algunas de éstas. Por ejemplo, se podría pensar en crear una característica llamada *Epífisis*, que agrupe las dos primeras (*Presencia* y *Separación*). Esta nueva característica podría tomas ahora los siguientes valores: Ausencia (estadio A), presencia apenas visible (estadio B), presencia visible separada (estadios C, D, E, F, G), epífisis corona metáfisis (estadio H) y comienzo de la fusión (estadio I). Se fusionan también las características *Forma I* y *Forma II*. Las nuevas características del radio serán las siguientes:

Epífisis: Existencia o no de la epífisis, y su posición respecto a la metáfisis. Los valores que toma son: *Ausencia* (estadio A), *Presencia a penas visible* (estadio B), *Presencia separada* (estadios C, D, E, F, G), *epífisis corona metáfisis* (estadio H) y *comienzo de la fusión* (estadio I).

Diámetros: Relación entre el diámetro de la epífisis y la metáfisis.

Forma: Forma del contorno de la epífisis. Puede ser *Oval* (estadio C), *Picuda con la base plana* (estadios D y E), *Picuda con la base adaptada* (estadio F), y *Picuda con una pequeña giba* (estadios G, H, I).

Superficies: Representa la aparición de superficies interiores (palmar y dorsal). Pueden estar *ausentes* (estadios B, C, D), aparecer como *una línea densa* (estadio E), como *dos líneas* (estadio F) o pueden apreciarse ya las *superficies* (estadios G, H, I).

La nueva tabla de valores será:

Estadio	Epífisis	Diámetros	Forma	Superficies
A	Ausente	∞	(oval)	(plano)
B	Pequeña	$\gg 2$	(oval)	(plano)
C	Separada	> 2	oval	plano
D	Separada	≤ 2	picuda plana	plano
E	Separada	< 2	picuda plana	1 línea
F	Separada	< 2	picuda adaptada	2 líneas
G	Separada	≈ 1	articulaciones	forma c
H	Corona	≈ 1	articulaciones	forma c
I	Fusión	≈ 1	articulaciones	forma c

Se ha procedido con el resto de los huesos del mismo modo que con el radio. Las características usadas pueden encontrarse en el apéndice F.

6.4.2 Modelado borroso de las características

Para construir un clasificador automático es necesario modelar las características descritas en la sección anterior más formalmente, de modo que sean susceptibles de ser procesadas en un ordenador.

Algunos estudios muestran que puede existir una discrepancia entre los resultados de la clasificación llevada a cabo por diferentes radiólogos, e incluso entre las clasificaciones de un mismo radiólogo en distintos instantes de tiempo; es lo que se conoce como variaciones inter e intra-observador [135]. La razón de esto es que algunos estadios pueden confundirse fácilmente con los más cercanos y, en ocasiones, los huesos presentan características pertenecientes a distintos estadios de clasificación. Por lo tanto, las características con las que se va a trabajar no están perfectamente definidas, pudiendo (e incluso debiendo) existir un cierto grado de solapamiento entre ellas. En la práctica no hay una frontera clara entre una epífisis oval y una que empieza a ser apuntada, ya que el proceso de evolución de la forma no es un proceso discreto sino gradual.

La naturaleza intrínseca del proceso y la falta de transiciones claras dentro de las características han hecho que se opte por un modelado basado en *variables lingüísticas* (cap. 3). En función de la naturaleza numérica –o no– de las características, las variables lingüísticas estarán formadas por conjuntos borrosos definidos sobre una variable base, o bien por gránulos borrosos.

Las cuatro características del radio definidas en la sección anterior se modelan de la siguiente forma:

Epífisis: Variable lingüística formada por 5 gránulos (*Ausente*, *Pequeña*, *Presente*, *Corona*, *Fusión*) como se muestra en la figura 6.7. El grado de solapamiento entre gránulos se ha estimado a partir de las discrepancias inter e intra-observador.

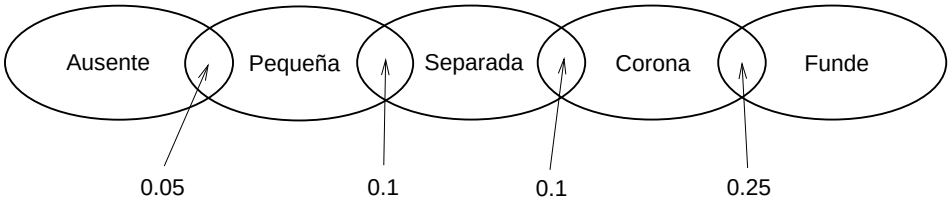


Figura 6.7 Variable lingüística *Epífisis*

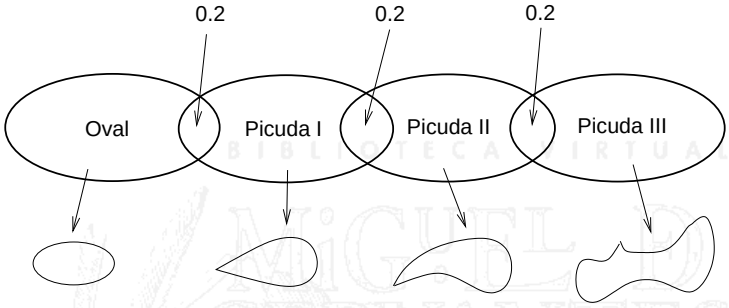


Figura 6.8 Variable lingüística *Forma*

Forma: Variable lingüística formada por 4 gránulos (*Oval*, *Picuda I*, *Picuda II*, *Picuda III*) como se muestra en la figura 6.8. El grado de solapamiento entre gránulos se ha estimado a partir de las discrepancias inter e intra-observador.

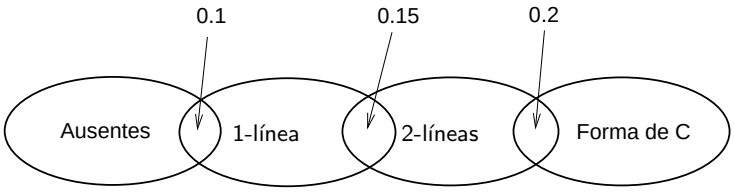


Figura 6.9 Variable lingüística *Superficies*

Superficies: Variable lingüística formada por 4 gránulos (*Ausentes*, *1 línea*, *2 líneas*, *superficie*) como se muestra en la figura 6.9. El grado de solapamiento entre gránulos se ha estimado a partir de las discrepancias inter e intra-observador.

Diámetros: En este caso la entrada va a ser un número, que es la relación entre el diámetro de la metáfisis (D) y de la epífisis (d), esto es $R = \frac{D_{metaf}}{d_{epif}}$. Se definen 4 conjuntos, correspondientes a los números borrosos *menor que uno*, *más o menos uno*, *más o menos 2* y *mayor que dos*, como se muestra en la figura 6.10.

6.4.3 Elección del clasificador. ID3 borroso

Se ha observado que entre las características elegidas, existen algunas que son excluyentes: si la *epífisis* está *ausente*, el resto de características no tomarán ningún valor;

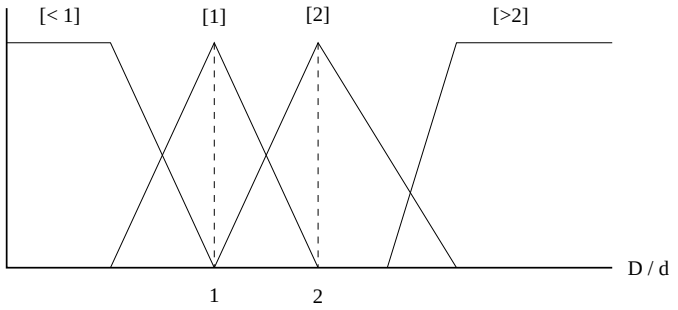


Figura 6.10 Variable lingüística *Diámetros*

si la epífisis corona la metáfisis es muy extraño que la forma de la primera sea *oval*. Este hecho es la causa principal de la elección de un clasificador en forma de árbol, concretamente, un árbol de clasificación borroso, para poder tratar con las variables lingüísticas anteriormente definidas.

Puesto que el dominio de salida es discreto y particionado de antemano, se ha optado por un árbol del tipo ID3 [134]. Siguiendo lo propuesto en el método TW2, se generarán 13 árboles de decisión borrosos, uno para cada hueso del método RUS. Cada hueso será clasificado en un estadio de desarrollo (A, B,..., I). Este esquema de clasificación es aplicable tanto a radiografías de niños como de niñas (que en el método reciben distinta puntuación para un mismo estadio).

ID3 [9, 134] es un método de clasificación basado en árboles que asume dominios discretos de baja cardinalidad. Requiere una partición *a priori* de los diferentes dominios. Umano y otros en [134] introducen la versión borrosa de este algoritmo. *Fuzzy ID3* es una extensión de ID3 para aplicarse sobre conjuntos de datos borrosos de entrada y genera un árbol de decisión borroso usando los conjuntos borrosos definidos *a priori* por el usuario para cada atributo. Este algoritmo genera el árbol de decisión de mínimo tamaño para unos ciertos parámetros de clasificación, eligiendo el orden óptimo (en términos de ganancia de información) de estos parámetros a lo largo del árbol.

Un árbol de decisión borroso consiste en una serie de *nodos* donde se evalúan los atributos, *ramas* para los distintos valores de cada símbolo y *hojas* donde se toman las decisiones. Para generar un árbol se necesita un conjunto de datos de entrenamiento. Si el árbol es borroso, en cada hoja no sólo aparecerá el valor de un conjunto de salida, sino el grado de pertenencia asociado a cada una de las clases de salida.

En el apéndice C se recoge de forma más detallada la teoría que concierne a los árboles de decisión, así como cómo se puede trabajar con ellos usando matrices de transición como las descritas en el capítulo 4.

6.4.4 Definición del conjunto de entrenamiento

El número de datos (radiografías) etiquetados necesarios para entrenar correctamente un árbol de decisión o alguna otra clase de clasificador es muy alto, y estas radiografías han de cubrir todos los posibles estados de cada hueso; sin embargo, como ya se ha indicado, el empleo de esta clase de conjunto de entrenamiento sería equivalente a la construcción de un nuevo método de cálculo de la maduración ósea desde cero. Una forma alternativa de actuar, sería utilizar el conocimiento ya existente y la experiencia acumulada que se puede encontrar dentro del TW2. Para crearlo, los autores utilizaron radiografías procedentes de 3000 niños y 3000 niñas. Al trabajar con las conclusiones extraídas por Tanner y Whitehouse se estarían empleando implícitamente las 6000 radiografías para entrenar el árbol.

BIBLIOTECA VIRTUAL

Aunque el conjunto de entrenamiento no puede extraerse directamente del método, sí puede hacerse a partir de las características anteriormente definidas y cierta información complementaria suministrada por radiólogos. El proceso de construcción del conjunto de entrenamiento es el siguiente:

CERVANTES

1. En primer lugar se extraen 9 *prototipos* directamente del método. Estos prototipos contienen las características "canónicas" de cada uno de los 9 posibles estadios en que se puede clasificar cada hueso. Se les asigna valores de pertenencia iguales a 1.

2. En segundo lugar se han de estudiar los casos "no canónicos", es decir, aquellos que presentan características que según el método pueden pertenecer a diversos estadios de desarrollo. Dado que los radiólogos habitualmente se encuentran con estos casos, su experiencia les convierte en los agentes ideales para clasificar tales situaciones, y asignarles así un valor de pertenencia que oscilará entre 0 y 1. Para ello se les proporcionó una plantilla para cada hueso, como la que aparece en la tabla 6.1. Los resultados devueltos por los expertos[2] para el caso del radio pueden verse en la tabla 6.2.

3. Los datos etiquetados de los que se dispone serán usados *a posteriori* para refinar el árbol y realizar diversas pruebas.

6.4.5 Construcción de los árboles de decisión

Usando como datos de entrenamiento los conjuntos descritos en la sección anterior y un algoritmo ID3 borroso (apéndice C) se llega a un árbol de clasificación borroso para cada uno de los 13 huesos del método RUS. A modo de ejemplo en la figura 6.12 se muestra el árbol de clasificación para el radio. En las *hojas* de árbol sólo se ha representado el conjunto que tiene la máxima pertenencia, pero realmente estarían las 9 clases de salida, cada una con un valor distinto de pertenencia.

[2] Ver agradecimientos.

Plantilla para el Método TW-2

Hueso: **Radio**

Epífisis	Forma	Superficies	Diámetros	Estadio	Confianza
1 2 3 4 5	1 2 3 4	1 2 3 4	D/d	A B C D E F G H I	%
□□□□□□	□ □ □ □	□ □ □ □	——	□ □ □ □ □ □ □ □ □	——
□□□□□□	□ □ □ □	□ □ □ □	——	□ □ □ □ □ □ □ □ □	——
□□□□□□	□ □ □ □	□ □ □ □	——	□ □ □ □ □ □ □ □ □	——
⋮	⋮	⋮	⋮	⋮	⋮
□□□□□□	□ □ □ □	□ □ □ □	——	□ □ □ □ □ □ □ □ □	——

Tabla 6.1 Modelo de plantilla dada a los radiólogos

Siguiendo el esquema de la figura 6.4, cada árbol tendrá como entrada las características propias de cada hueso y como salida un estadio de maduración ósea, que será ponderado numéricamente según el método TW2, siendo el valor de ponderación diferente para cada hueso y diferente para niños y niñas. La suma de los valores finales, transformados convenientemente según indica el método, proporciona como salida la edad ósea del paciente, con la misma resolución que el método original[3].

6.5 IMPLEMENTACIÓN MATRICIAL

Una vez confeccionados los árboles, es posible hacer una traducción de éstos a matrices de transición. Siguiendo las técnicas que se explican en el apéndice C, puede crearse una matriz completa de reglas, **C** a partir de los valores del árbol. Las entradas granulares se definen mediante sus correspondientes matrices de activación:

$$\mathbf{R_{ep}} = \begin{pmatrix} 1 & 0.05 & 0 & 0 & 0 \\ 0.05 & 1 & 0.1 & 0 & 0 \\ 0 & 0.1 & 1 & 0.1 & 0 \\ 0 & 0 & 0.1 & 1 & 0.25 \\ 0 & 0 & 0 & 0.25 & 1 \end{pmatrix}$$

$$\mathbf{R_{for}} = \begin{pmatrix} 1 & 0.2 & 0 & 0 \\ 0.2 & 1 & 0.2 & 0 \\ 0 & 0.2 & 1 & 0.2 \\ 0 & 0 & 0.2 & 1 \end{pmatrix}$$

[3]El problema de la resolución del espacio de salida es uno de los cuellos de botella de la implementación del TW2 con otras técnicas, como por ejemplo usando redes neuronales [82]. Cuanto mayor sea la resolución exigida, mayor es el número de ejemplos que se han de usar para el entrenamiento.

Estadio	Pertenencia	Epífisis	Forma	Superficies	Diámetros
A	1	1	1	1	10
B	1	2	1	1	5
B	1	2	1	1	3
C	1	3	1	1	2.5
C	0.8	3	1	1	2
D	0.2	3	1	2	2.5
C	0.2	3	1	2	1.9
C	0.5	3	2	1	2.5
C	0.5	3	2	1	2.2
D	0.9	3	2	1	2
D	1	3	2	1	1.9
E	0.9	3	2	2	2
E	1	3	2	2	1.5
E	0.5	3	2	3	2
F	0.9	3	2	3	1.5
F	0.5	3	3	2	1.8
F	0.8	3	3	2	1.10
F	0.5	3	3	3	2
F	1	3	3	3	1.11
F	0.8	3	3	3	1
F	0.8	3	3	4	1
G	0.7	3	4	3	1
G	1	3	4	4	1
H	0.2	4	3	3	1
H	0.8	4	3	4	1
G	0.6	4	4	3	0.95
H	1	4	4	4	1
I	0.7	5	3	4	0.88
I	0.3	5	4	3	0.90
I	0.2	5	3	3	0.91
I	1	5	4	4	0.8

Tabla 6.2 Datos de entrenamiento para el radio

$$\mathbf{R_{sup}} = \begin{pmatrix} 1 & 0.1 & 0 & 0 \\ 0.1 & 1 & 0.15 & 0 \\ 0 & 0.15 & 1 & 0.2 \\ 0 & 0 & 0.2 & 1 \end{pmatrix}$$

La entrada relativa a la relación entre diámetros no es granular, por lo tanto la entrada al sistema será la activación de cada uno de los conjuntos por la entrada. Para crear las matrices de transición, usamos la identidad 4×4.

De este modo, el sistema queda representado por 64 matrices $\mathbf{\Omega}_{ijk}$ de tamaño 9×5. Si se usa el método SAM para realizar la inferencia (como suele ser habitual en los árboles de decisión), la inferencia se reduce a calcular

$$\begin{bmatrix} A \\ B \\ C \\ D \\ E \\ F \\ G \\ H \\ I \end{bmatrix} = \sum_{k=1}^{4} d_k \sum_{j=1}^{4} s_j \sum_{i=1}^{4} f_i \mathbf{\Omega}_{ijk} \mathbf{ep}$$

BIBLIOTECA VIRTUAL

donde d_k son las componentes del vector de entrada de la relación de diámetros, s_j las componentes del vector de *Superficies*, f_i las componentes del vector *Forma* y **ep** el vector de *Epífisis*.

6.5.1 Inclusión del conocimiento a posteriori

A la hora de determinar el estado de maduración ósea de un hueso, muchas veces los expertos tienen dudas entre un estadio y el siguiente, dudas que despejan tras analizar el resto de los huesos. El conocer el estado general de la mano, les hace optar *a posteriori* por uno de los dos estados dudosos.

Lo mismo ocurre a la hora de clasificar un hueso automáticamente. Como salida se va a obtener una serie de valores de pertenencia del hueso a diferentes estadios. Puede ocurrir que dos estadios tengan un valor de pertenencia muy parecido y muy superior al resto. El árbol tal y como se ha expuesto hasta ahora, en un determinado momento toma una decisión *dura*, por lo que esta información se pierde.

Como línea complementaria (todavía en fase de desarrollo) se pretende incluir este conocimiento *a posteriori* usando para ello un sistema realimentado como el que se muestra en la Fig. 6.11. De acuerdo con lo visto en la sección 4.4, esta clase de sistema será fácilmente modelable usando matrices de transición, incluso es posible predecir estados futuros para entradas constantes, siendo la única variable la rama realimentada.

6.6 EXPERIMENTOS Y RESULTADOS

Una vez confeccionados los distintos árboles de decisión se realizó una primera sesión de *test* con las radiografías etiquetadas de los niños. Las características se extrajeron mediante estimación visual directa por parte de un no-especialista (concretamente, del autor de esta tesis). De las 57 radiografías hubo que descartar 9, debido a su mala

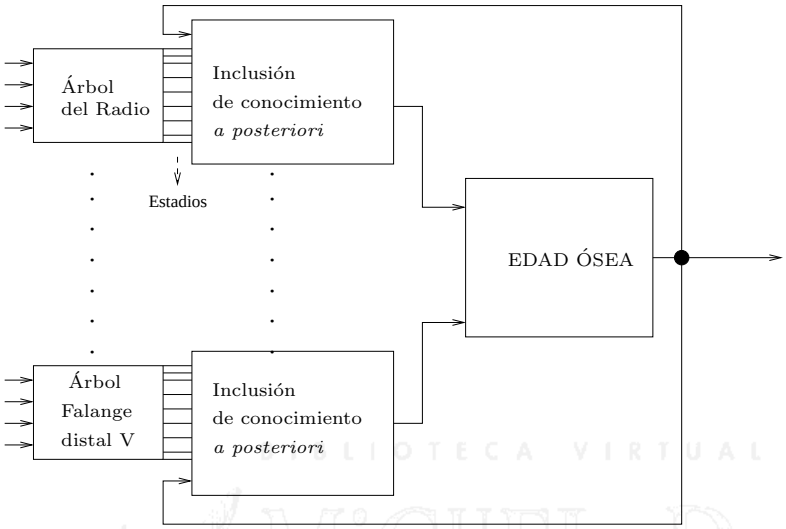


Figura 6.11 Inclusión de la información a posteriori dentro del método

calidad, que hacía imposible distinguir epífisis y metáfisis. A partir de las desviaciones observadas en cada hueso, los árboles fueron reajustados manualmente.

Usando como conjunto de test de nuevo las 48 radiografías útiles de los niños, la clasificación individual de cada hueso realizada por los árboles obtiene de media un 94% de acuerdo con la clasificación de los radiólogos. El test se repitió para un conjunto de datos distinto del de entrenamiento, en este caso las 85 radiografías etiquetadas de las niñas. Los resultados son bastante similares:

Hueso	Aciertos	%
Cúbito	83	97,6%
Falange proximal I	81	95,3%
Falange Media III	83	97,6%

La buena calidad de los resultados se debe sin duda a que el clasificador ha sido creado directamente a partir del método, por lo que el proceso es prácticamente el mismo, y los resultados tienen que ser muy similares. Como es natural, las prestaciones se reducirán al emplear características obtenidas automáticamente, pero es posible que mejoren si los grados de solapamiento entre gránulos se estiman a partir de datos de entrenamiento.

6.7 CONCLUSIONES

En el presente capítulo se ha mostrado una implementación en un caso real de la teoría descrita en el capítulo 4. El uso de gránulos borrosos para definir las características usadas en el clasificador, ha permitido traducir un método lingüístico, como es el TW2, a

un método programable. A la hora de refinar el árbol obtenido, el uso de éstos gránulos (y por consiguiente de las matrices de activación $\mathbf{R_A}$) ha supuesto una ventaja añadida; no ha sido necesario retocar formas ni distribución de conjuntos, como suele ser usual en las aplicaciones borrosas. Con la variación de los grados de solapamiento entre conceptos ha sido suficiente.

Con vistas a la implementación futura de la aplicación completa, el uso de matrices de transición en la parte operativa puede acelerar considerablemente el proceso, sobre todo teniendo en cuenta que si se incluye el módulo que evalúa el conocimiento a posteriori va a ser necesario realizar las inferencias varias veces de manera recursiva para cada radiografía.

En este capítulo se ha asumido que las características que requiere el método pueden obtenerse de forma automática. Si bien en el momento actual no disponemos de una solución cerrada para ello, sí disponemos de resultados preliminares [82, 83, 87] en esta dirección.

BIBLIOTECA VIRTUAL MIGUEL DE CERVANTES

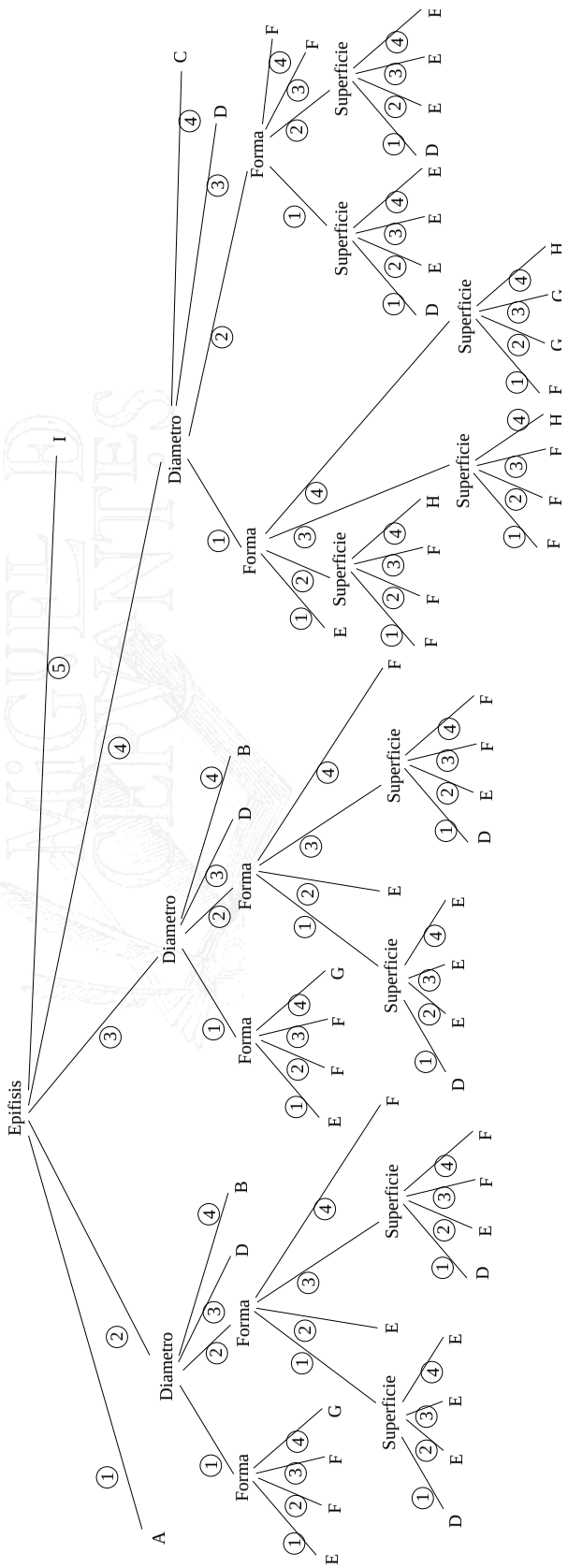


Figura 6.12 Árbol del radio

7

CONCLUSIONES, APORTACIONES Y LÍNEAS FUTURAS

BIBLIOTECA VIRTUAL

7.1 CONCLUSIONES

Como idea central, en el presente trabajo se ha propuesto un método matricial inédito que llevar a cabo inferencias borrosas de un modo rápido y compacto. Dicho método es especialmente eficiente cuando las entradas al sistema son conjuntos borrosos o bien gránulos borrosos, aunque, como se ha descrito, acepta también entradas numéricas. Este nuevo marco de trabajo se ha aplicado sobre diferentes problemas reales, como se ha visto en los capítulos 5 y 6.

En términos más concretos, podemos decir que el trabajo se ha vertebrado acorde con los siguientes extremos:

- Basándonos en el concepto de *gránulo* existente en la literatura, se ha planteado una forma de representar conceptos lingüísticos considerando únicamente las relaciones entre dichos conceptos. A esta representación se le ha denominado en el presente trabajo *gránulo borroso*. Los conceptos representados de esta forma no precisan de una variable base ni de unas funciones de pertenencia determinadas, (como sucede en el caso de la representación usando conjuntos borrosos). Tan sólo es necesario definir el grado de solapamiento entre conceptos.

  Los gránulos borrosos son susceptibles de ser procesados mediante un proceso de inferencia borrosa estándar, pero dado que los solapamiento se pueden expresar en forma de matriz, va a ser más eficiente un tipo de inferencia basado en operativa enteramente matricial.

- El trabajo parte de la hipótesis de que, ya sea a partir de una definición granular ya sea usando conjuntos borrosos, la entrada a un sistema va a poderse representar de forma vectorial. En caso de que la entrada no sea una combinación de los conjuntos (o gránulos) que forman la variable lingüística del espacio de entrada, se asume que ésta podrá ser representada como alguno de esos conjuntos afectado por un modificador. Esta hipótesis, como se ha indicado en el capítulo 4, es perfectamente realista en aplicaciones CWW.

- Para un sistema que tenga como entradas gránulos o conjuntos borrosos, va a ser posible definir un proceso de inferencia basado en matrices. Las peculiaridades de este tipo de inferencia van a permitir que un gran número de operaciones sean precalculadas, ahorrando de este modo tiempo de cálculo y espacio de almacenamiento. Toda la información de un sistema borroso queda reducida a un conjunto de matrices de transición, cuyo número y tamaño van a estar relacionados con el número de entradas en el sistema y por el número de conjuntos definidos sobre las VL de entrada y salida.

  Hay que destacar que estas matrices son totalmente independientes de la resolución de los espacios de entrada o salida, incluso se podría trabajar con espacios continuos. Esto conlleva una considerable disminución del número de operaciones.

- En un primer momento se ha desarrollado un método basado en matrices de transición para inferencias borrosas del tipo SAM, el cual usa operadores lineales. No obstante, estos resultados han sido extendidos a sistemas borrosos que usen t-normas y t-conormas genéricas, siempre y cuando éstas cumplan las propiedades conmutativa, asociativa y distributiva, y se usen las mismas normas durante todo el proceso de inferencia borrosa. De este modo, una operativa que utilice los operadores máximo y mínimo (los más usados en sistemas borrosos) puede ser implementada dentro de este nuevo marco de trabajo.

- La metodología admite también entradas numéricas. Sin embargo, en este caso, el uso de matrices no está justificado, ya que no existe un ahorro computacional. Se ha pretendido extender el marco matricial para albergar también este caso con el objetivo de que pueda utilizarse en sistemas en que convivan entradas procedentes de fuentes de información heterogéneas.

- Se ha realizado un estudio especial de los sistemas borrosos realimentados, ya que el uso de matrices de transición permite comprender mejor su evolución temporal. Para el caso de un sistema realimentado tipo SAM, se ha visto que es posible calcular el estado final hacia el que tiende el sistema en régimen estacionario, realizando un estudio de autovalores y autovectores.

  También se ha visto que en un sistema realimentado SAM en régimen estacionario el valor final de la salida no depende del valor de entrada en el instante inicial.

- Es posible también interpretar las matrices de activación $\mathbf{R_A}$ como una representación de un conjunto de relaciones borrosas. De este modo, las reglas que gobiernan la inferencia y la propagación de restricciones han sido reinterpretadas para conjuntos de relaciones, asociando los diversos operadores matriciales con las conjunciones y disyunciones.

- Este nuevo marco de trabajo ha sido aplicado a varios problemas reales, por un lado a esquemas de seguimiento y por otro a un caso de análisis de imagen médica. A este respecto:

  – A partir de un sistema borroso ya existente, el MHT borroso, se han redefinido algunas de las variables lingüísticas de entrada de tal forma que en el sistema

conviven entradas granulares, borrosas y numéricas. Dicho sistema se ha implementado mediante matrices de transición.

– El segundo caso de seguimiento no está directamente basado en términos lingüísticos, lo cual muestra que algunos de los procedimientos descritos en esta tesis no se limitan a entornos CWW. Se trata de un detector de maniobra para un filtro de Kalman. La imprecisión a la hora de determinar si el filtro está desacoplado puede procesarse mediante sistema borroso realimentado. Este FFS es el encargado de reinicializar el filtro en caso de desacople, pero no debe hacerlo en caso de errores grandes aislados. El diseño ha sido posible gracias a las propiedades conocidas de los FFS.
– Por último, se ha propuesto una automatización del método TW2, para el cálculo del estado de maduración ósea en la infancia. Se ha procedido a implementar un clasificador directamente a partir del texto del método original, por lo que el uso de gránulos para manejar conceptos ha sido imprescindible. Por otro lado, dado que el clasificador final consta de 13 árboles de decisión, y puede que sea necesario realimentar las decisiones del clasificador (en caso de usar conocimiento *a posteriori*) la conversión del árbol en un sistema de matrices de transición puede acelerar considerablemente el proceso.

7.2 APORTACIONES

Para mayor claridad expositiva, aunque si bien se encuentra implícito en el apartado anterior, podemos decir que la aportación principal de este trabajo es la concepción, análisis e implementación de un nuevo marco de trabajo para llevar a cabo inferencias borrosas, basado en matrices de transición (FITM). En particular:

1. Se ha propuesto el representar la información mediante *gránulos borrosos*. Estos gránulos por sí mismos no tienen representación numérica, sino que la tienen cuando entran en relación con los demás.
2. El nuevo *marco de trabajo matricial* se ha propuesto inicialmente para sistemas tipo SAM, si bien se ha extendido posteriormente a t-normas y s-normas genéricas (que cumplan determinadas propiedades). Asimismo, se ha propuesto cómo extender esta metodología a entradas no borrosas.
3. Se ha hecho un profundo estudio de la convergencia de los *sistemas borrosos realimentados*, basado, precisamente, en las matrices de transición.

A partir del núcleo central de la tesis, constituido por las aportaciones que acabamos de señalar, se han realizado aportaciones en forma de aplicaciones concretas de dicha teoría en los ámbitos que siguen:

1. Árboles de decisión borrosos, concretamente árboles tipo ID3.

2. Algoritmo MHT borroso.
3. Detección de maniobra en entorno de seguimiento estereoscópico.
4. Análisis automático de estado de maduración ósea en la infancia a partir del método TW2.

Finalmente, como soporte y/o extensión de la metodología FITM:

1. Se han extendido las reglas de inferencia borrosas a conjuntos de relaciones borrosas.
2. Se han definido un conjunto de operadores matriciales basados en intersecciones y uniones: $\times_{\cup\cap}$, unión tensorial de Kronecker ($\otimes_{\cup}$) e intersección tensorial de Kronecker ($\otimes_{\cap}$).

7.3 LÍNEAS FUTURAS

La metodología presentada en este trabajo puede ser, por supuesto, extendida y generalizada. Destacamos algunas posibles direcciones:

- La definición de gránulo borroso que aparece en la tesis se basa exclusivamente en el grado de solapamiento entre conceptos, que se relaciona con el grado de consistencia entre dos conjuntos borrosos. Existen otras muchas relaciones entre conjuntos borrosos que pueden ser reinterpretadas como nuevas relaciones entre gránulos.
- Toda la metodología propuesta esta basada en asunción de que la relación entre todas las reglas es disyuntiva. Podría ser de interés deducir la operativa para otro tipo de relaciones, por ejemplo, conjuntiva.
- Dentro del clasificador basado en TW2 será necesario incluir el conocimiento *a posteriori*. Para ello, naturalmente, deberá contarse con la ayuda de los expertos médicos.
- En sistemas borrosos de dos o más entradas la inferencia se realiza a partir de conjuntos de matrices bidimensionales. Estos conjuntos de matrices pueden ser representados por una sola matriz N-dimensionales (para sistemas con $N-1$ entradas), permitiendo así reformular todo el proceso en notación tensorial.
- En [91], Mendel plantea que la representación de palabras mediante conjuntos borrosos estándar puede ser *científicamente incorrecta*. Como solución propone el uso de sistemas borrosos de tipo-2 [63]. Siguiendo esta filosofía, sería interesante plantear una extensión de los gránulos borrosos a gránulos de tipo-2.

GLOSARIO DE ABREVIATURAS

BDE: Base de datos explicativa.

BDEI: Base de datos explicativa inicializada.

CID: Conjunto inicial de datos.

CTD: Conjunto terminal de datos.

CWW: Cálculo lingüístico (Computing with Words).

FEK: Filtro extendido de Kalman.

FFS: Sistema borroso realimentado (Fuzzy Feedback System).

FITM: Inferencia rápida usando matrices de transición (Fast Inference using Transition Matrices).

FKA: Filtro de Kalman acoplado.

FKCP: Filtro de Kalman en coordenadas principales.

FKD: Filtro de Kalman desacoplado.

FK: Filtro de Kalman.

FLS: Fuzzy Logic System.

MHT: Seguimiento mediante múltiples hipótesis (Multiple Hypothesis Tracking).

MIB: Motor de inferencia borrosa.

MPG: Modus ponens generalizado.

MP: Modus ponens.

MTG: Modus tollens generalizado.

MT: Modus tollens.

PTS: Forma pseudo-trapezoidal (pseudo-trapezoid shaped).

SAM: Modelo aditivo estándar (standard additive model).

SBRB: Sistema basado en reglas borrosas.

TCP: Teoría computacional de las percepciones.

TSK: Sistema borroso tipo Takagi-Sugeno-Kang.

TW2: Método de Tanner-Whitehouse (segunda edición).

VL: Variable lingüística.

BIBLIOTECA VIRTUAL MIGUEL DE CERVANTES

BIBLIOGRAFÍA

[1] S. Aja-Fernández y C. Alberola-López, "Inference with fuzzy granules for computing with words: A practical viewpoint," en *Proc. of FUZZ-IEEE'03*, St. Louis, MO, Mayo 2003, pp. 566–571.

[2] S. Aja-Fernández y C. Alberola-López, "Fast inference in SAM fuzzy systems using transition matrices," *IEEE Trans. Fuzzy Systems*, vol. 12, no. 2, pp. 170–182, Abr. 2004.

[3] S. Aja-Fernández, C. Alberola-López y G. Cybenko, "A fuzzy MHT algorithm applied to text-based information tracking," *IEEE Trans. Fuzzy Systems*, vol. 10, no. 3, pp. 360–374, Jun. 2002.

[4] S. Aja-Fernández, C. Alberola-López y J. Ruiz-Alzola, "Fuzzy anisotropic diffusion for speckle filtering," en *Proceedings of the ICASSP 2001*, Salt Lake City, Utah, Mayo 2001.

[5] S. Aja-Fernández, C. Alberola-López y J. Ruiz-Alzola, "A fuzzy controlled Kalman filter applied to stereo-visual tracking schemes," *Signal Processing*, vol. 83, pp. 101–120, 2003.

[6] A. T. Al-Taani, I. W. Ricketts y A. Y. Cairns, "Classification of hand bones for bone age assessment," en *Proc. of the Third IEEE International Conference on Electronics, Circuits, and Systems, ICECS'96*, vol. 2, Oct. 1996, pp. 1088–1091.

[7] C. Alberola y G. Cybenko, "Tracking with text-based messages," *IEEE Intelligent Systems*, vol. 14, no. 4, pp. 70–78, Ago. 1999.

[8] R. Alcalá, J. Casilla, O. Gordón, F. herrera y S. Zwir, cap. Techniques for learning and tuning fuzzy rule-based systems for linguistic modeling and their application, *Knowledge engineering. Systems techniques and applications*, C. Leondes, Ed. Accademic Press, 1999.

[9] L. I. Arranz, "Modificaciones sobre ID3 para el tratamiento de incertidumbre," en *II Congreso español sobre tecnologías y lógica fuzzy*, Facultad de Informática, UPM, Madrid, Spain, Nov. 1992, pp. 193–201.

[10] A. Bagga, J. Chai y A. Biermann, cap. Extracting information from text, *Computing with words.*, P. P. Wang, Ed. John Wiley and Sons Inc., 2001, pp. 209–234.

[11] H. Becker, "Computing with words and machine learning in medical diagnosis," *Information Sciences*, vol. 134, pp. 53–69, 2001.

[12] S. Blackman, *Multiple-target tracking with RADAR applications.* Dedham, MA: Artech House, Inc, 1986.

[13] F. Cao, H. Huang, E. Pietka y V. Gilsanz, "Digital atlas and web-based bone age assessment: system design and implementation," *Computerized Medical Imaging and Graphics*, vol. 24, pp. 297–307, 2000.

[14] F. E. I. Cao, H. K. Huang, E. Pietka, V. Gilsanz y S. Omisky, "Diagnostic workstation for digital hand atlas in bone age assessment," *Proc. SPIE Medical Imaging*, pp. 608–614, Feb. 1998.

[15] S. Cao, N. Rees y G. Feng, "Stability analysis of fuzzy control systems," *IEEE Transactions on System, Man and Cybernetics - Part B: Cybernetics*, vol. 26, no. 1, pp. 201–204, Feb. 1996.

[16] K. Chan, V. Lee y H. Leung, "RADAR tracking for air surveillance in a stressful enviroment using a fuzzy-gain filter," *IEEE Trans. Fuzzy Systems*, vol. 5, no. 1, pp. 80–88, Feb. 1997.

[17] G. Chen, Q. Xie y L. Shieh, "Fuzzy Kalman filtering," *Journal of Information Sciences*, vol. 109, pp. 197–209, 1998.

[18] Y. Chen y T. Tsao, "A description of the dynamical behaviour of fuzzy systems," *IEEE Transactions on System, Man and Cybernetics*, vol. 19, no. 4, pp. 745–755, Jul. 1989.

[19] C. K. Chui y G. Chen, *Kalman Filtering with Real-Time Applications.* Berlin: Springer-Verlag, 1998.

[20] F. Chung y T. Lee, "On fuzzy associative memory with multiple-rule storage capacity," *IEEE Trans. Fuzzy Systems*, vol. 4, no. 3, pp. 375–384, Ago. 1996.

[21] W. Combs y E. Andrews, "Combinatorial rule explosion eliminated by a fuzzy rule configuration," *IEEE Trans. Fuzzy Systems*, vol. 6, no. 1, pp. 1–11, Feb. 1998.

[22] E. Czogała y J. Łęski, cap. Fuzzy implications in approximate reasoning, *Computing with words in information. Intelligent Systems 1*, J. Kacprzyk y L. Zadeh, Eds. Physica-Verlag, 1999, pp. 342–357.

[23] S. Dick y A. Kandel, "Comment on combinational rule explosion eliminated by a fuzzy rule configuration," *IEEE Trans. Fuzzy Systems*, vol. 7, no. 4, pp. 475–477, Ago. 1999.

[24] V. Dimitrov y B. Hodge, cap. Virtual meaning: problems of interpretation in the social sciences, *Computing with words in information. Intelligent Systems 1*, J. Kacprzyk y L. Zadeh, Eds. Physica-Verlag, 1999, pp. 136–153.

[25] M. S. Dinesh, Bhanuprakash y A. Rao, "Vision system for bone measurement from digital hand radiograph," en *14th Conference of the Biomedical Engineering Society of India. An International Meeting, Proc. of the First Regional Confererence, IEEE.* Engineering in Medicine and Biology Society, 1996, pp. SPC9–SPC10.

[26] Z. Ding, H. Leung y K. Chan, "Model-set adaptation using a fuzzy Kalman filter," en *Proc of the third Int. Conf on Information Fusion, FUSION 2000*, Paris, France, Jul. 2000.

[27] D. Dubois, L. Foulloy, S. Galichet y H. Prade, cap. Performing approximate reasoning with words?, *Computing with words in information. Intelligent Systems 1*, J. Kacprzyk y L. Zadeh, Eds. Physica-Verlag, 1999, pp. 24–49.

[28] D. Dubois, E. Hüllermeier y H. Prade, "On the representation of fuzzy rules in terms of crisp rules," *Information sciences*, 2003, en imprenta.

[29] D. Dubois y H. Prade, "A theorem of implication functions defined from triangular norms," *Stochastica*, vol. 8, no. 3, pp. 267–279, 1984.

[30] D. Dubois y H. Prade, "Default reasoning and possibility theory," *Artificial Intelligence*, vol. 35, pp. 243–257, 1988.

[31] D. Dubois y H. Prade, "On fuzzy syllogisms," *Computational Intelligence*, vol. 4, pp. 171–179, 1988.

[32] D. Dubois y H. Prade, "Fuzzy sets in approximate reasoning, part 1: inference with possibility distributions," *Fuzzy Sets and Systems*, vol. 40, pp. 143–202, 1991.

[33] D. Dubois y H. Prade, "Fuzzy sets - a convenient fiction for modeling vagueness and possibility," *IEEE Trans. Fuzzy Systems*, vol. 2, no. 1, pp. 16–21, Feb. 1994.

[34] D. Dubois y H. Prade, "The three semantics of fuzzy sets," *Fuzzy Sets and Systems*, vol. 90, pp. 141–150, 1997.

[35] D. Dubois, H. Prade, F. Esteva, P. García y L. Godo, "A logical approach to interpolation based on similarity relations," *International journal of approximate reasoning*, vol. 17, pp. 1–36, 1997.

[36] W. Duch, R. Adamczak y K. Grabczewski, "New methodology of extraction, optimization and application of crisp and fuzzy logical rules," *IEEE Trans. on Neural Networks*, vol. 12, no. 2, pp. 277–306, Mar. 2001.

[37] N. D. Efford, "Knowledge-based segmentation and feature analysis of hand-wrist radiographs," School of Computer Studies, University of Leeds, Technical Report, Report 94.31, 1994.

[38] E. Ferri, A. Kandel y G. Langholz, cap. Fuzzy Negation, *Computing with words in information. Intelligent Systems 1*, J. Kacprzyk y L. Zadeh, Eds. Physica-Verlag, 1999, pp. 297–325.

[39] M. Figueiredo, F. Gomide, A. Rocha y R. Yager, "Comparison of Yager's level set method for fuzzy logic control with Mamdani's and Larsen's methods," *IEEE Trans. Fuzzy Systems*, vol. 1, no. 2, pp. 156–159, Mayo 1993.

[40] L. Garmendia, R. Yager, E. Trillas y A. Salvador, "On the instantiation of possibility distributions," *Fuzzy Sets and Systems*, vol. 133, pp. 237–248, 2003.

[41] G. Golub y C. V. Loan, *Matrix Computations*, 3^a ed. Baltimore, ME: The Johns Hopkins University Press, 1996.

[42] O. Gordon, "An evolutionary methodology for designing fuzzy rule-based systems," Tesis Doctoral, Department of Computer Science and Artificial Intelligence (DECSAI), Universidad de Granada, 1997.

[43] W. W. Greulich y S. I. Pyle, *Radiographic Atlas of Skeletal Development of Hand Wrist*, 2^a ed. Stanford CA: Standford University Press, 1971.

[44] P. Grim, "Self-reference and chaos in fuzzy logic," *IEEE Trans. Fuzzy Systems*, vol. 1, no. 4, pp. 237–253, Nov. 1994.

[45] R. Hathaway, J. Bezdek y W. Pedrycz, "A parametric model for fusing heterogeneous fuzzy data," *IEEE Trans. Fuzzy Systems*, vol. 4, no. 3, pp. 270–281, Ago. 1996.

[46] F. Herrera y L. Martínez, "On granularity in fuzzy logic: Minimum and maximum are the only absolutely franular t-norm and t-conorm," Mar. 1999, tech. Rep. #DECSAI-990102, ETS de Ingeniería Informática, Universidad de Granada.

[47] F. Herrera y L. Martínez, "A 2-tuple fuzzy linguistic representation model for computing with words," *IEEE Trans. Fuzzy Systems*, vol. 8, no. 6, pp. 746–752, Dic. 2000.

[48] E. Hüllermeier, D. Dubois y H. Prade, "Model adaptation in possibilistic instance-based reasoning," *IEEE Trans. Fuzzy Systems*, vol. 10, no. 3, pp. 333–339, Jun. 2002.

[49] N. Ho y V. Huynh, "An algebtraic approach to linguistic hedges in Zadeh's fuzzy logic," *Fuzzy Sets and Systems*, vol. 129, pp. 229–254, 2002.

[50] C. S. Hsu80, "A theory of cell-to-cell mapping dynamical systems," *ASME Tras. J. Dynam. Syst. Meas. Contr*, vol. 47, pp. 931–939, 1980.

[51] V. Huynh, T. Ho y Y. Nakamori, "A parametric representation of linguistic hedges in Zadeh's fuzzy logic," *International journal of approximate reasoning*, vol. 30, pp. 203–223, 2002.

[52] H. Ishibuchi y T. Nakashima, "Effect of rule weights in fuzzy rule-based classification systems," *IEEE Trans. Fuzzy Systems*, vol. 9, no. 4, pp. 506–515, Ago. 2001.

[53] H. Ishibuchi, K. Nozaki, N. Yamamoto y H. Tanaka, "Selecting fuzzy if-then rules for classification problems using genetic algorithms," *IEEE Trans. Fuzzy Systems*, vol. 3, no. 2, pp. 260–270, Ago. 1995.

[54] A. Jain y D. Zongker, "Feature selection: Evaluation, application, and small sample performance," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 19, no. 2, pp. 153–158, Feb. 1997.

[55] A. K. Jain, R. P. W. Duin y J. Mao, "Statistical pattern recognition: A review," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 1, pp. 4–37, Ene. 2000.

[56] C. Z. Janikow, "Exemplar learning in fuzzy decision trees," en *Proc. of the fifth IEEE conference on fuzzy systems, FUZZ-IEEE'96*, New Orleans, Louisiana, USA, Sept. 1996, pp. 1500–1505.

[57] C. Janikow, "Fuzzy decision trees: issues and methods," *IEEE Trans. on System, Man and Cybernetics - Part B: Cybernetics*, vol. 28, no. 1, pp. 1–14, Feb. 1998.

[58] K. H. K y P. Pynsent, "A fully automated bone-ageing system," *Acta Paediatr.*, vol. 406, pp. 81–83, 1994.

[59] J. Kacprzyk, cap. Fuzzy sets and fuzzy systems: a brief introduction, *Fuzzy systems in medicine*, P. Szczepaniak, G. L. P.J y J. Kacprzyk, Eds. New York: Physica-Verlag, 2000, pp. 3–30.

[60] J. Kacprzyk y L. Zadeh, Eds., *Computing with words in information. Intelligent Systems 1. Foundations.* Physica-Verlag, 1999.

[61] J. Kacprzyk y L. Zadeh, Eds., *Computing with words in information. Intelligent Systems 2. Applications.* Physica-Verlag, 1999.

[62] H. Kang, "Stability and control of fuzzy dynamic system via cell-state transitions in fuzzy hypercubes," *IEEE Trans. Fuzzy Systems*, vol. 1, no. 4, pp. 267–279, Nov. 1993.

[63] N. Karnik, J. Mendel y Q. Liang, "Type-2 fuzzy logic systems," *IEEE Trans. Fuzzy Systems*, vol. 7, no. 6, pp. 643–658, Dic. 1999.

[64] S. M. Kay, *Fundamentals of Statistical Signal Processing. Estimation Theory.* Englewood Cliffs, NJ: Prentice Hall, 1993.

[65] E. Kerre y M. Nachtegael, Eds., *Fuzzy techniques in image processing.* New York: Physica-Verlag, 2000.

[66] G. Klir y B. Yuan, *Fuzzy Sets and Fuzzy Logic.* New Jersey: Prentice-Hall International, 1995.

[67] G. Klir, "Fuzzy logic: its meaning and significance," *IEEE Potential*, vol. 14, no. 4, pp. 10–15, Oct. 1995.

[68] G. Klir, "Fuzzy arithmetic with requisite constraints," *Fuzzy Sets and Systems*, vol. 91, pp. 165–175, 1997.

[69] L. Kohout, B. Granville y E. Kim, cap. Granular relational computing with semiotic descriptors using BK-products of fuzzy relations, *Computing with words.*, P. P. Wang, Ed. John Wiley and Sons Inc., 2001, pp. 89–146.

[70] B. Kosko, *Neural Networks and Fuzzy Systems.* New Jersey: Prentice-Hall International, 1992.

[71] B. Kosko, *Neural Networks and Fuzzy Systems: A Dynamical Systems Approach to Machine Intelligence*. Prentice Hall, 1992.

[72] B. Kosko, *Fuzzy Engineering*. New Jersey: Prentice-Hall International, 1997.

[73] S. Kwabwe, S. K. Pal y R. A. King, "Recognition of bones from X-ray of the hand and wrist," *Int J. Syst. Sci*, vol. 16, no. 4, pp. 403–413, 1985.

[74] S. Kwok y J. Chambers, "2-D frequency domain Kalman filter with fuzzy gearshifting for object velocity estimation in image sequences," *Electronic Letters*, vol. 31, no. 19, pp. 1653–1654, Sept. 1995.

[75] N. Laviolette y J. W. S. Jr, "The efficacy of fuzzy representations of uncertainty," *IEEE Trans. Fuzzy Systems*, vol. 2, no. 1, pp. 4–15, Feb. 1994.

[76] J. Lawry, "A methodology for computing with words," *Int. Journal of Approximate Reasoning*, vol. 28, pp. 51–89, 2001.

[77] B. Lazzerini y F. Marcelloni, "Reducing computation overhead in MISO fuzzy systems," *Fuzzy Sets and Systems*, vol. 113, no. 3, pp. 485–496, 2000.

[78] H. Lee y H. Tanaka, cap. Fuzzy graphs with linguistic input-output by fuzzy approximation models, *Computing with words in information. Intelligent Systems 2*, J. Kacprzyk y L. Zadeh, Eds. Physica-Verlag, 1999, pp. 121–136.

[79] T. Lin, cap. Granular computing: fuzzy logic and rough sets, *Computing with words in information. Intelligent Systems 1*, J. Kacprzyk y L. Zadeh, Eds. Physica-Verlag, 1999, pp. 183–200.

[80] J. Loetamonphong y S. Fang, "An efficient solution procedure for fuzzy relation equations with max-product composition," *IEEE Trans. Fuzzy Systems*, vol. 7, no. 4, pp. 441–445, Ago. 1999.

[81] M. Loew, cap. Feature Extraction, *Handbook of Medical Imaging. Volume 2: Medical Image Processing and Analysis*, M. Sonka y J. M. Fitzpatrick, Eds. Bellingham, WN, USA: SPIE Press, 2000, pp. 273–342.

[82] R. Luis-García, J. I. Arribas, S. Aja-Fernández y C. Alberola-López, "A neural architecture for bone age assessment," en *Proc. of the IASTED International Conference on Signal Processing, Pattern Recognition & Applications*, Creta, Greece, Jun. 2002, pp. 161–166.

[83] R. Luis-García, M. Martín-Fernández, J. I. Arribas y C. Alberola-López, "A fully automatic algorithm for contour detection of bones in hand radiographs using active contours," en *Proc. of the ICIP 2003*, Barcelona, Spain, Sept. 2003.

[84] E. Mamdani, "Applications of fuzzy logic to approximate reasoning using linguistic synthesis," *IEEE Transactions on Computers*, vol. 26, no. 12, pp. 1182–1191, Dic. 1977.

[85] Marimin, M. Umano, I. Hatono y H. Tamura, "Linguistic labels for expressing fuzzy preference relations in fuzzy group decision making," *IEEE Trans. on System, Man and Cybernetics - Part B: Cybernetics*, vol. 28, pp. 205–218, 1998.

[86] C. Marsala, "Apprentissage inductif en présence de données imprécises: construction et utilisation d'arbres de décision flous," Tesis Doctoral, Université Paris 6, Ene. 1998.

[87] M. A. Martín-Fernández, M. Martín-Fernández y C. Alberola-López, "Automatic bone age assessment: a registration approach," en *Proceedings on Medical Imaging, SPIE*, San Diego, CA, Feb. 2003.

[88] S. McGinnity y G. Irwin, "Fuzzy assisted Kalman filtering for target tracking," en *UKACC, Int. Conf. Control'96, IEE Conf Pub*, vol. 1, no. 427, 1996, pp. 614–619.

[89] S. McGinnity y G. Irwin, "Nonlinear Kalman filtering using fuzzy local linear models," en *Proc of the American Control Conf.*, Alburquerque, New Mexico, USA, Jun. 1997.

[90] J. M. Mendel, "Fuzzy logic systems for engineering: A tutorial," *Proceedings of the IEEE*, vol. 83, no. 3, pp. 345–377, Mar. 1995.

[91] J. Mendel, "Fuzzy sets for words: a new beginning," en *Proc. of FUZZ-IEEE'03*, St. Louis, MO, Mayo 2003, pp. 37–42.

[92] J. Mendel y Q. Liang, "Comments on combinational rule explosion eliminated by a fuzzy rule configuration," *IEEE Trans. Fuzzy Systems*, vol. 7, no. 3, pp. 369–371, Jul. 1999.

[93] F. J. Moreno-Velo, "Un entorno de desarrollo para sistemas de inferencia complejos basados en lógica difusa," Tesis Doctoral, Universidad de Sevilla, Abr. 2003.

[94] V. Novák, "Antonyms and linguistic quantifiers in fuzzy logic," *Fuzzy Sets and Systems*, vol. 124, pp. 335–351, 2001.

[95] S. K. Pal y R. A. King, "On edge detection of X-ray images using fuzzy sets," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 5, no. 1, pp. 69–77, Ene. 1983.

[96] S. K. Pal, R. A. King y A. A. Hashim, "Automatic grey level thresholding using index of fuzziness and entropy," *Patt. Recog. Lett.*, vol. 1, pp. 141–146, Mar. 1983.

[97] S. K. Pal, R. A. King y A. A. Hashim, "Image description and primitive extraction using fuzzy sets," *IEEE Trans. on System, Man and Cybernetics*, vol. SMC.-13, no. 1, pp. 94–100, Ene. 1983.

[98] K. Passino y S. Yurkovich, *Fuzzy Control*. California: Addison Wesley, 1998.

[99] A. Pathak, S. Pal y R. King, "Syntactic recognition of skeletal maturity," *Patt. Recog. Lett.*, vol. 1, pp. 141–146, Mar. 1983.

[100] A. Pathak y S. Pal, "Fuzzy grammars in syntactic recognition of skeletal maturity from X-rays," *IEEE Trans. on System, Man and Cybernetics*, vol. SMC-16, no. 5, pp. 657–666, Sept. 1986.

[101] W. Pedrycz, A. Vasilakos y A. Gacek, "Information granulation for concept formation," en *SAC'00*, Como, Italy, Mar. 2000.

[102] E. Pietka, "Computer-assisted bone age assesment based on features automatically extracted from a hand radiograph," *Computerized Medical Imaging and Graphics*, vol. 19, no. 3, pp. 251–259, 1995.

[103] E. Pietka, A. Gertych, S. Pospiech, F. Cao, H. Huang y V. Gilsanz, "Computer-assisted bone age assessment: image preprocessing and epiphyseal/metaphyseal ROI extraction," *IEEE Trans. Med. Imaging*, vol. 20, no. 8, pp. 715–729, Ago. 2001.

[104] E. Pietka y H. Huang, "Epiphyseal fusion assessment based on wavelets decomposition analysis," *Computerized Medical Imaging and Graphics*, vol. 19, no. 6, pp. 465–472, 1995.

[105] E. Pietka, L. Kaabi, M. L. Kuo y H. K. Huang, "Feature extraction in carpal-bone analysis," *IEEE Trans. Med. Imaging*, vol. 12, no. 1, pp. 44–49, Mar. 1993.

[106] E. Pietka, M. F. McNitt-Gray, M. L. Kuo y H. K. Huang, "Computer-assisted phalangeal analysis in skeletal age assessment," *IEEE Trans. Med. Imaging*, vol. 10, no. 4, pp. 616–620, Dic. 1991.

[107] L. Polkowski y A. Skowron, cap. Towards and adaptive calculus of granules, *Computing with words in information. Intelligent Systems 1*, J. Kacprzyk y L. Zadeh, Eds. Physica-Verlag, 1999, pp. 201–228.

[108] D. Reid, "An algorithm for tracking multiple targets," *IEEE Trans. on Automatic Control*, vol. AC-24, no. 6, pp. 843–854, Dic. 1979.

[109] R. Reyero y C. Nicolás, *Sistemas de control basados en Lógica borrosa: fuzzy control*. Madrid: Omron Electronics S.A. Centro de Investigaciones Tecnológicas IKERLAN, 1995.

[110] B. Rieger, cap. Semiotics and computational linguistics. On semiotic cognitive information processing, *Computing with words in information. Intelligent Systems 1*, J. Kacprzyk y L. Zadeh, Eds. Physica-Verlag, 1999, pp. 93–118.

[111] J. Roberts, D. Mills, D. Charnley y C. Harris, "Improved Kalman filter initialisation using neurofuzzy estimation," en *Proc 4th Int. Conf. on Artificial Neural Networks*, 1995, pp. 329–334.

[112] J. Ruiz-Alzola, C. Alberola-López y J. Casar-Corredera, "Model-based stereo-visual tracking: Covariance analysis and tracking schemes," *Signal Processing*, vol. 80, pp. 23–43, 2000.

[113] M. Sonka y J. M. Fitzpatrick, Eds., *Handbook of Medical Imaging. Volume 2: Medical Image Processing and Analysis*. Bellingham, WN, USA: SPIE Press, 2000.

[114] E. Straszecka, cap. Defining membership functions, *Fuzzy systems in medicine*, P. Szczepaniak, G. L. P.J y J. Kacprzyk, Eds. New York: Physica-Verlag, 2000, pp. 32–47.

[115] A. Suanes y R. Cañete, "Sistema automatizado de evaluación de la maduración ósea aplicado al método Tanner-Whitehouse 2," *Anales Españoles de Pediatría*, vol. 50, no. 2, pp. 134–139, 1999.

[116] T. Sudkamp y R. J. H. II, "Interpolation, completion and learning fuzzy rules," *IEEE Trans. on System, Man and Cybernetics*, vol. 24, no. 2, pp. 332–342, Feb. 1994.

[117] M. Sugeno y G. Kang, "Structure identification of fuzzy model," *Fuzzy Sets and Systems*, vol. 28, no. 1, pp. 15–33, 1988.

[118] Y. N. Sun, C. C. Ko, C. W. Mao y C. J. Lin, "A computer system for skeletal growth measurement," *Computers and Bimedical Research*, pp. 2–12, Feb. 1994.

[119] A. Suárez y J. Lutsko, "Globally optimal fuzzy decision trees," *IEEE Trans. Pattern Anal. Mach. Intell.*, no. 12, pp. 1297–1311, Dic. 1999.

[120] P. Szczepaniak, G. L. P.J y J. Kacprzyk, Eds., *Fuzzy systems in medicine*. New York: Physica-Verlag, 2000.

[121] T. Takagi y M. Sugeno, "Fuzzy identification of systems and its applications to modeling and control," *IEEE Trans. on System, Man and Cybernetics*, vol. 15, no. 1, pp. 116–132, Ene. 1985.

[122] J. M. Tanner, H. Goldstein, N. Cameron, W. B. Saunders y M. J. R. Healy, *Assessment of skeletal maturity and Prediction of Adult Height*, 3^a ed. W B Saunders Co, 2001.

[123] J. M. Tanner y R. H. Whitehouse, *Assessment of Skeletal Maturity and Prediction of Adult Height (TW2 Method)*. London, UK: Academic Press, 1975.

[124] J. M. Tanner, R. H. Whitehouse, N. Cameron, W. A. Marshall, M. J. R. Healy y H. Goldstein, *Valoración de la Maduración Esquelética y Predicción de la Talla Adulta (Método TW2)*, 2^a ed. Barcelona: Áncora S. A., 1988.

[125] J. Tanner, O. D., G. Lindgren, J. Grunbaum, R. Elsouki y D. Labarthe, "Reliability and validity of computer-assisted estimates of tanner- whitehouse skeletal maturity (*casas*): comparison with the manual method," *Horm Res.*, vol. 42, pp. 288–294, 1994.

[126] C. W. Tao, "Simplification of a fuzzy mechanism with rule combination," *Soft Computing*, vol. 6, pp. 418–425, 2001.

[127] H. Thiele, "On semantic models for investigating Computing with Words," Abr. 1998, tech. Rep., ISSN 1433-3325, University of Dortmund.

[128] H. Thiele, cap. On the semantics of fuzzy linguistic quantifiers, *Computing with words in information. Intelligent Systems 1*, J. Kacprzyk y L. Zadeh, Eds. Physica-Verlag, 1999, pp. 358–382.

[129] R. Tong, "Some properties of fuzzy feedback systems," *IEEE Trans. on System, Man and Cybernetics*, vol. SMC-10, no. 6, pp. 327–330, Jun. 1980.

[130] E. Trillas, C. Alsina y J. Terricabras, *Introducción a la lógica borrosa*. Barcelona: Ed. Ariel S.A., 1995.

[131] E. Trillas y J. G. Ríos, *Aplicaciones de la Lógica Borrosa*. Madrid: Consejo Superior de Investigaciones Científicas, 1992.

[132] I. Türkşen, cap. Computing with descriptive and veristic words: Knowledge representation and reasoning, *Computing with words.*, P. P. Wang, Ed. John Wiley and Sons Inc., 2001, pp. 297–328.

[133] I. Türkşen, V. Kreinovich y R. Yager, "A new class of fuzzy implications. axioms of fuzzy implication revisited," *Fuzzy Sets and Systems*, vol. 100, pp. 267–272, 1998.

[134] M. Umano, H. Okamoto, I. Hatono y H. Tamura, "Fuzzy decision trees by fuzzy ID3 algorithm and its application to diagnosis systems," en *Proc. of the third IEEE conference on fuzzy systems, FUZZ-IEEE'94*, Orlando, Florida, USA, Jun. 1994, pp. 2113–2118.

[135] B. Viñuela-Rueda y S. Alberola-López, "Variabilidad inter e intra-observador en el estudio de la edad ósea mediante el método TW2-RUS de Tanner," en *26 Congreso Nacional Soc. Esp. Rad. Médica*, Maspalomas, Gran Canaria, Spain, Mayo 2002.

[136] L. X. Wang y J. Mendel, "Generating fuzzy rules by learning from examples," *IEEE Trans. on System, Man and Cybernetics*, vol. 22, no. 6, pp. 1414–1427, Jun. 1992.

[137] P. P. Wang, Ed., *Computing with Words*. John Wiley and Sons Inc., 2001.

[138] P. Wang, cap. Computing with words: introduction, implications and applications, *Computing with words.*, P. P. Wang, Ed. John Wiley and Sons Inc., 2001, pp. 13–33.

[139] X. Wang, B. Chen, G. Qian y F. Ye, "On the optimization at fuzzy decision trees," *Fuzzy sets and systems*, pp. 117–125, 2000.

[140] S. Wastl y H. Dickhaus, "Computerized classification of maturity stages of hand bones of children and juveniles," en *Bridging Disciplines for Biomedicine,, 18th Annual International Conference of the IEEE*, H. Boom, C. Robinson, W. Rutten, M. Neuman y H. Wijkstra, Eds. Engineering in Medicine and Biology Society, Oct. 1996, pp. 1155–1156, vol. 3.

[141] R. Yager, "Some questions related to linguistic variables," *Busefal*, vol. 10, pp. 54–65, 1982.

[142] R. Yager, "Quantified propositions in linguistic logic," *International Journal of Man-Machine Stud.*, vol. 9, pp. 195–227, 1983.

[143] R. Yager, "On different classes of linguistic variables defined via fuzzy subsets," *Kybernetes*, vol. 13, pp. 103–110, 1984.

[144] R. Yager, "Connectives and quantifiers in fuzzy sets," *Fuzzy Sets and Systems*, vol. 40, pp. 39–75, 1991.

[145] R. Yager, "Fuzzy logic and artificial intelligence," *Fuzzy Sets and Systems*, vol. 90, pp. 193–198, 1997.

[146] R. Yager, "On a class of weak triangular norm operators," *Information Sciences*, vol. 96, pp. 47–78, 1997.

[147] R. Yager, cap. Approximate reasoning as a basis for computing with words, *Computing with words in information. Intelligent Systems 1*, J. Kacprzyk y L. Zadeh, Eds. Physica-Verlag, 1999, pp. 50–77.

[148] R. Yager, "Modeling uncertsinty using partial information," *Information sciences*, vol. 121, pp. 271–294, 1999.

[149] R. Yager, "On global requirements for implication operators in fuzzy modus ponens," *Fuzzy Sets and Systems*, vol. 106, pp. 3–10, 1999.

[150] R. Yager, "Fuzzy modeling for intelligent decision making under uncertainly," *IEEE Trans. on System, Man and Cybernetics - Part B: Cybernetics*, vol. 30, no. 1, pp. 60–70, Feb. 2000.

[151] R. Yager, "Veristic variables," *IEEE Trans. on System, Man and Cybernetics - Part B: Cybernetics*, vol. 30, no. 1, pp. 71–84, Feb. 2000.

[152] R. Yager, "Determining equivalent values for possibilistic variables," *IEEE Trans. on System, Man and Cybernetics - Part B: Cybernetics*, vol. 31, no. 1, pp. 19–31, Feb. 2001.

[153] R. Yager, "On the instantiation of possibility distributions," *Fuzzy Sets and Systems*, vol. 128, pp. 261–266, 2002.

[154] R. Yager y V. Kreinovich, "Decision making under interval probabilities," *International Journal of Approximate Reasoning*, vol. 22, pp. 195–215, 1999.

[155] R. Yager y A. Rybalov, "Noncommutative self-identity aggregation," *Fuzzy Sets and Systems*, vol. 85, pp. 73–82, 1997.

[156] Y. Yam, P. Baranyi y C. Yang, "Reduction of fuzzy rule base via singular value decomposition," *IEEE Trans. Fuzzy Systems*, vol. 7, no. 2, pp. 120–132, Abr. 1999.

[157] M. Ying, "A formal model for computing with words," *IEEE Trans. Fuzzy Systems*, vol. 10, no. 5, pp. 640–652, Oct. 2002.

[158] L. A. Zadeh, "Fuzzy sets," *Information and Control*, vol. 8, no. 3, pp. 338–353, 1965.

[159] L. A. Zadeh, "Quantitative fuzzy semantics," *Information Sciences*, vol. 3, pp. 159–176, 1971.

[160] L. A. Zadeh, "Outline o a new approach to the analysis of complex system and decission processes," *IEEE Trans. on System, Man and Cybernetics*, vol. SMC-3, no. 1, pp. 28–44, Ene. 1973.

[161] L. A. Zadeh, "The concept of a linguistic variable and its application to approximate reasoning I," *Information Sciences*, vol. 8, pp. 199–249, 1975.

[162] L. A. Zadeh, "Fuzzy sets as a basis for a theory of possibility," *Fuzzy Sets and Systems*, vol. 1, pp. 3–28, 1978.

[163] L. A. Zadeh, "Pruf - a meaning representation languaje for natural languajes," *International Journal of Man-Machine Studies*, vol. 10, pp. 395–460, 1978.

[164] L. A. Zadeh, cap. A theory of approximate reasoning, *Machine Intelligence, Vol 9*, J. H. ans D. Mitchie y L. Mikulich, Eds. New York: Wiley, 1979, pp. 149–194.

[165] L. A. Zadeh, "Commonsense reasoning based on fuzzy logic," en *Proc. of the 1986 Winter Simulation Conference*, 1986, pp. 445–447.

[166] L. A. Zadeh, "Fuzzy logic, neural networks and soft computing," *Communications of the ACM*, vol. 37, no. 3, pp. 77–84, Mar. 1994.

[167] L. A. Zadeh, "Fuzzy logic = computing with words," *IEEE Trans. Fuzzy Systems*, vol. 4, no. 2, pp. 103–111, Mayo 1996.

[168] L. A. Zadeh, "Towards a theory of fyzzy information granulation and its centrality in human reasoning and fuzzy logic," *Fuzzy Sets and Systems*, vol. 90, pp. 111–127, 1997.

[169] L. A. Zadeh, "Somer reflections on soft computing, granular computing and their roles in the conception, design and utilization of information/intelligence systems," *Soft Computing*, vol. 2, pp. 23–25, 1998.

[170] L. A. Zadeh, "From computing with numbers to computing with words, from manipulation of measurements to manipulation of perceptions," *IEEE Trans. on Circuits and Systems*, vol. 45, no. 1, pp. 105–119, Ene. 1999.

[171] L. A. Zadeh, "Towards a perception-based theory of probabilistic reasoning with imprecise probabilities," *Journal of statistical planning and inference*, vol. 105, pp. 233–264, 2002.

[172] J. Zeidler y M. Schlosser, "Continuous-valued attributed in fuzzy decision trees," en *Proc. Information processing and management of uncertainty in knowledge - based systems IPMU-96*, Granada, Spain, Jul. 1996, pp. 395–400.

[173] J. Zeidler, M. Schlosser, A. Ittner y C. Posthoff, "Fuzzy decision trees and numerical attributes," en *Proc. of the fifth IEEE conference on fuzzy systems, FUZZ-IEEE'96*, New Orleans, Louisiana, USA, Sept. 1996, pp. 985–990.

[174] X. Zeng y M. Singh, "Approximation theory of fuzzy systems- SISO case," *IEEE Trans. Fuzzy Systems*, vol. 2, no. 2, pp. 162–176, Feb. 1994.

A

DESARROLLOS AUXILIARES DEL CAPÍTULO 4

BIBLIOTECA VIRTUAL MIGUEL DE CERVANTES

A.1 ECUACIONES

A.1.1 Sobre la ecuación (4.15)

Si la base de reglas se crea variando el valor del segundo antecedente para cada valor del primero, la matriz $\mathbf{G}_l$ pasa a ser

$$\mathbf{G}_l = \begin{bmatrix} (A_1 \circ A_1)(B_1 \circ B_l) & (A_1 \circ A_2)(B_1 \circ B_l) & \cdots & (A_1 \circ A_M)(B_1 \circ B_l) \\ (A_1 \circ A_1)(B_2 \circ B_l) & (A_1 \circ A_2)(B_2 \circ B_l) & \cdots & (A_1 \circ A_M)(B_2 \circ B_l) \\ \vdots & \vdots & \vdots & \vdots \\ (A_1 \circ A_1)(B_N \circ B_l) & (A_1 \circ A_2)(B_N \circ B_l) & \cdots & (A_1 \circ A_M)(B_N \circ B_l) \\ (A_2 \circ A_1)(B_1 \circ B_l) & (A_2 \circ A_2)(B_1 \circ B_l) & \cdots & (A_2 \circ A_M)(B_1 \circ B_l) \\ (A_2 \circ A_1)(B_2 \circ B_l) & (A_2 \circ A_2)(B_2 \circ B_l) & \cdots & (A_2 \circ A_M)(B_2 \circ B_l) \\ \vdots & \vdots & \vdots & \vdots \\ (A_M \circ A_1)(B_1 \circ B_l) & (A_M \circ A_2)(B_1 \circ B_l) & \cdots & (A_M \circ A_M)(B_1 \circ B_l) \\ (A_M \circ A_1)(B_2 \circ B_l) & (A_M \circ A_2)(B_2 \circ B_l) & \cdots & (A_M \circ A_M)(B_2 \circ B_l) \\ \vdots & \vdots & \vdots & \vdots \\ (A_M \circ A_1)(B_N \circ B_l) & (A_M \circ A_2)(B_N \circ B_l) & \cdots & (A_M \circ A_M)(B_N \circ B_l) \end{bmatrix}$$

de donde se puede extraer directamente la ecuación (4.15).

A.1.2 Sobre la ecuación (4.30)

Como se indicó anteriormente, la matriz $\mathbf{G}_{l,t}^T$ se obtiene apilando columnas de la ecuación (4.27) siendo j un índice creciente

$$\mathbf{G}_{l,t}^T = \begin{bmatrix} (A_{p(1)} \circ A_1)B^{12}_{q(1)s(1)}(l,t) & \cdots & (A_{p(MN_1N_2)} \circ A_1)B^{12}_{q(MN_1N_2)s(MN_1N_2)}(l,t) \\ (A_{p(1)} \circ A_2)B^{12}_{q(1)s(1)}(l,t) & \cdots & (A_{p(MN_1N_2)} \circ A_2)B^{12}_{q(MN_1N_2)s(MN_1N_2)}(l,t) \\ \vdots & \vdots & \vdots \\ (A_{p(1)} \circ A_M)B^{12}_{q(1)s(1)}(l,t) & \cdots & (A_{p(MN_1N_2)} \circ A_M)B^{12}_{q(MN_1N_2)s(MN_1N_2)}(l,t) \end{bmatrix}$$

y su traspuesta

$$\mathbf{G}_{l,t} = \begin{bmatrix} (A_{p(1)} \circ A_1)B^{12}_{q(1)s(1)}(l,t) & \cdots & (A_{p(1)} \circ A_M)B^{12}_{q(1)s(1)}(l,t) \\ (A_{p(2)} \circ A_1)B^{12}_{q(2)s(2)}(l,t) & \cdots & (A_{p(2)} \circ A_M)B^{12}_{q(2)s(2)}(l,t) \\ \vdots & \vdots & \vdots \\ (A_{p(MN_1N_2)} \circ A_1)B^{12}_{q(MN_1N_2)s(MN_1N_2)}(l,t) & \cdots & (A_{p(MN_1N_2)} \circ A_M)B^{12}_{q(MN_1N_2)s(MN_1N_2)}(l,t) \end{bmatrix}$$

Usando la definición de $B^{12}_{q(2)s(2)}(l,t)$ (ecuación 4.28), se puede escribir

$$\mathbf{G}_{l,t} = \begin{bmatrix} (A_{p(1)} \circ A_1)(B^1_{q(1)} \circ B^1_l)(B^2_{s(1)} \circ B^2_t) & \cdots & (A_{p(1)} \circ A_M)(B^1_{q(1)} \circ B^1_l)(B^2_{s(1)} \circ B^2_t) \\ \vdots & \ddots & \vdots \\ (A_{p(MN_1N_2)} \circ A_1)(B^1_{q(MN_1N_2)} \circ B^1_l)(B^2_{s(MN_1N_2)} \circ B^2_t) & \cdots & (A_{p(MN_1N_2)} \circ A_M)(B^1_{q(MN_1N_2)} \circ B^1_l)(B^2_{s(MN_1N_2)} \circ B^2_t) \end{bmatrix}$$

Usando valores apropiados para $p(j)$, $q(j)$ and $s(j)$ la matriz se convierte en

$$\mathbf{G}_{l,t} = \begin{bmatrix} (A_1 \circ A_1)(B^1_1 \circ B^1_l)(B^2_1 \circ B^2_t) & \cdots & (A_1 \circ A_M)(B^1_1 \circ B^1_l)(B^2_1 \circ B^2_t) \\ \vdots & \vdots & \vdots \\ (A_1 \circ A_1)(B^1_1 \circ B^1_l)(B^2_{N_2} \circ B^2_t) & \cdots & (A_1 \circ A_M)(B^1_1 \circ B^1_l)(B^2_{N_2} \circ B^2_t) \\ (A_1 \circ A_1)(B^1_2 \circ B^1_l)(B^2_1 \circ B^2_t) & \cdots & (A_1 \circ A_M)(B^1_2 \circ B^1_l)(B^2_1 \circ B^2_t) \\ \vdots & \vdots & \vdots \\ (A_1 \circ A_1)(B^1_2 \circ B^1_l)(B^2_{N_2} \circ B^2_t) & \cdots & (A_1 \circ A_M)(B^1_2 \circ B^1_l)(B^2_{N_2} \circ B^2_t) \\ \vdots & \vdots & \vdots \\ (A_1 \circ A_1)(B^1_{N_1} \circ B^1_l)(B^2_{N_2} \circ B^2_t) & \cdots & (A_1 \circ A_M)(B^1_{N_1} \circ B^1_l)(B^2_{N_2} \circ B^2_t) \\ (A_2 \circ A_1)(B^1_1 \circ B^1_l)(B^2_1 \circ B^2_t) & \cdots & (A_2 \circ A_M)(B^1_1 \circ B^1_l)(B^2_1 \circ B^2_t) \\ \vdots & \vdots & \vdots \\ (A_2 \circ A_1)(B^1_{N_1} \circ B^1_l)(B^2_{N_2} \circ B^2_t) & \cdots & (A_2 \circ A_M)(B^1_{N_1} \circ B^1_l)(B^2_{N_2} \circ B^2_t) \\ \vdots & \vdots & \vdots \\ (A_M \circ A_1)(B^1_{N_1} \circ B^1_l)(B^2_{N_2} \circ B^2_t) & \cdots & (A_M \circ A_M)(B^1_{N_1} \circ B^1_l)(B^2_{N_2} \circ B^2_t) \end{bmatrix}$$

que puede ser escrita de forma compacta usando en producto tensorial de Kronecker

$$\mathbf{G}_{l,t} = \begin{bmatrix} ((A_1 \circ A_1)(B^1_1 \circ B^1_l)) \otimes_{\mathcal{K}} [\mathbf{R}_{\mathbf{B}^2}\mathbf{E}_t] & \cdots & ((A_1 \circ A_M)(B^1_1 \circ B^1_l)) \otimes_{\mathcal{K}} [\mathbf{R}_{\mathbf{B}^2}\mathbf{E}_t] \\ ((A_1 \circ A_1)(B^1_2 \circ B^1_l)) \otimes_{\mathcal{K}} [\mathbf{R}_{\mathbf{B}^2}\mathbf{E}_t] & \cdots & ((A_1 \circ A_M)(B^1_2 \circ B^1_l)) \otimes_{\mathcal{K}} [\mathbf{R}_{\mathbf{B}^2}\mathbf{E}_t] \\ \vdots & \vdots & \vdots \\ ((A_1 \circ A_1)(B^1_{N_1} \circ B^1_l)) \otimes_{\mathcal{K}} [\mathbf{R}_{\mathbf{B}^2}\mathbf{E}_t] & \cdots & ((A_1 \circ A_M)(B^1_{N_1} \circ B^1_l)) \otimes_{\mathcal{K}} [\mathbf{R}_{\mathbf{B}^2}\mathbf{E}_t] \\ ((A_2 \circ A_1)(B^1_1 \circ B^1_l)) \otimes_{\mathcal{K}} [\mathbf{R}_{\mathbf{B}^2}\mathbf{E}_t] & \cdots & ((A_2 \circ A_M)(B^1_1 \circ B^1_l)) \otimes_{\mathcal{K}} [\mathbf{R}_{\mathbf{B}^2}\mathbf{E}_t] \\ \vdots & \vdots & \vdots \\ ((A_2 \circ A_1)(B^1_{N_1} \circ B^1_l)) \otimes_{\mathcal{K}} [\mathbf{R}_{\mathbf{B}^2}\mathbf{E}_t] & \cdots & ((A_2 \circ A_M)(B^1_{N_1} \circ B^1_l)) \otimes_{\mathcal{K}} [\mathbf{R}_{\mathbf{B}^2}\mathbf{E}_t] \\ \vdots & \vdots & \vdots \\ ((A_M \circ A_1)(B^1_1 \circ B^1_l)) \otimes_{\mathcal{K}} [\mathbf{R}_{\mathbf{B}^2}\mathbf{E}_t] & \cdots & ((A_M \circ A_M)(B^1_1 \circ B^1_l)) \otimes_{\mathcal{K}} [\mathbf{R}_{\mathbf{B}^2}\mathbf{E}_t] \\ \vdots & \vdots & \vdots \\ ((A_M \circ A_1)(B^1_{N_1} \circ B^1_l)) \otimes_{\mathcal{K}} [\mathbf{R}_{\mathbf{B}^2}\mathbf{E}_t] & \cdots & ((A_M \circ A_M)(B^1_{N_1} \circ B^1_l)) \otimes_{\mathcal{K}} [\mathbf{R}_{\mathbf{B}^2}\mathbf{E}_t] \end{bmatrix}$$

y procediendo de manera similar

$$\mathbf{G}_{l,t} = \begin{bmatrix} (A_1 \circ A_1) \otimes_{\mathcal{K}} [(\mathbf{R}_{\mathbf{B}^1} \otimes_{\mathcal{K}} [\mathbf{R}_{\mathbf{B}^2}\mathbf{E}_t])\mathbf{E}_l] & \cdots & (A_1 \circ A_M) \otimes_{\mathcal{K}} [(\mathbf{R}_{\mathbf{B}^1} \otimes_{\mathcal{K}} [\mathbf{R}_{\mathbf{B}^2}\mathbf{E}_t])\mathbf{E}_l] \\ \vdots & \ddots & \vdots \\ (A_M \circ A_1) \otimes_{\mathcal{K}} [(\mathbf{R}_{\mathbf{B}^1} \otimes_{\mathcal{K}} [\mathbf{R}_{\mathbf{B}^2}\mathbf{E}_t])\mathbf{E}_l] & \cdots & (A_M \circ A_M) \otimes_{\mathcal{K}} [(\mathbf{R}_{\mathbf{B}^1} \otimes_{\mathcal{K}} [\mathbf{R}_{\mathbf{B}^2}\mathbf{E}_t])\mathbf{E}_l] \end{bmatrix}$$

finalmente se obtiene

$$\begin{aligned} \mathbf{G}_{l,t} &= \mathbf{R}_{\mathbf{A}} \otimes_{\mathcal{K}} [(\mathbf{R}_{\mathbf{B}^1} \otimes_{\mathcal{K}} (\mathbf{R}_{\mathbf{B}^2}\mathbf{E}_t))\mathbf{E}_l] \\ &= \mathbf{R}_{\mathbf{A}} \otimes_{\mathcal{K}} [(\mathbf{R}_{\mathbf{B}^1} \otimes_{\mathcal{K}} ((\mathbf{R}_{\mathbf{B}^2} \otimes_{\mathcal{K}} 1)\mathbf{E}_t))\mathbf{E}_l] \end{aligned}$$

como aparece en la ecuación (4.30).

BIBLIOTECA VIRTUAL

A.2 MATRIZ DE TRANSICIÓN DE UN MODIFICADOR EN UN SAM DE DOS ENTRADAS

La salida de un SAM de dos entradas y una salida puede escribirse de forma matricial como (ecuación (4.14))

$$\boldsymbol{\gamma}^T\mathbf{D} = \sum_{l=1}^{N} \alpha_l \boldsymbol{\beta}^T \mathbf{G}_l^T \mathbf{C}\mathbf{D}$$

Si se define la matriz modificada $\mathbf{R}'_{\mathbf{A}}$ como la matriz $\mathbf{R}_{\mathbf{A}}$ más el modificador $\boldsymbol{\Delta}_{\mathbf{H}}$:

$$\mathbf{R}'_{\mathbf{A}} = \mathbf{R}_{\mathbf{A}} + \boldsymbol{\Delta}_{\mathbf{H}}$$

Suponiendo que el modificador afecta al conjunto en posición k de la primera entrada, puede escribirse

$$\begin{aligned} \boldsymbol{\gamma}^T\mathbf{D} = \mathbf{D}^T\boldsymbol{\gamma} &= \sum_{l=1}^{N} \alpha_l (\mathbf{C}\mathbf{D})^T \mathbf{G'}_l \boldsymbol{\beta}_k \\ &= \mathbf{D}^T \sum_{l=1}^{N} \mathbf{C}^T \alpha_l \mathbf{G'}_l \boldsymbol{\beta}_k \\ &= \mathbf{D}^T\mathbf{C}^T \left(\sum_{l=1}^{N} \alpha_l \mathbf{G'}_l \right) \boldsymbol{\beta}_k \end{aligned}$$

donde $\mathbf{G'}_l$ se define

$$\begin{aligned} \mathbf{G'}_l &= \mathbf{R}'_{\mathbf{A}} \otimes_{\mathcal{K}} [\mathbf{R}_{\mathbf{B}}\mathbf{E}_l] \\ &= (\mathbf{R}_{\mathbf{A}} + \boldsymbol{\Delta}_{\mathbf{H}}) \otimes_{\mathcal{K}} [\mathbf{R}_{\mathbf{B}}\mathbf{E}_l] \\ &= \mathbf{R}_{\mathbf{A}} \otimes_{\mathcal{K}} [\mathbf{R}_{\mathbf{B}}\mathbf{E}_l] + \boldsymbol{\Delta}_{\mathbf{H}} \otimes_{\mathcal{K}} [\mathbf{R}_{\mathbf{B}}\mathbf{E}_l] \end{aligned}$$

De está expresión es fácil deducir que

$$\begin{aligned} \boldsymbol{\Omega'}_l &= \mathbf{C}^T \left[\mathbf{R_A} \otimes_{\mathcal{K}} \left[\mathbf{R_B E}_l\right]\right] + \mathbf{C}^T \left[\boldsymbol{\Delta_H} \otimes_{\mathcal{K}} \left[\mathbf{R_B E}_l\right]\right] \\ &= \boldsymbol{\Omega}_l + \boldsymbol{\Delta\Omega}_l \\ \boldsymbol{\Delta\Omega}_l &= \mathbf{C}^T \left[\boldsymbol{\Delta_H} \otimes_{\mathcal{K}} \left[\mathbf{R_B E}_l\right]\right] \end{aligned} \tag{A.1}$$

BIBLIOTECA VIRTUAL MIGUEL DE CERVANTES

B

EXTENSIÓN DE LA METODOLOGÍA FITM A OPERADORES NO LINEALES

BIBLIOTECA VIRTUAL

Los desarrollos siguientes van a ser válidos para t-normas y t-conormas genéricas (representadas aquí por los operadores $\bigcap$ y $\bigcup$) si éstas cumplen las propiedades conmutativa, asociativa y distributiva [66], y si se usan las mismas normas para el cálculo de las cuatro operaciones básicas relacionadas con la inferencia (altura, truncado, unión e intersección, expresiones (2.54) y (2.56)).

B.1 DEFINICIONES DE OPERACIONES VECTORIALES Y MATRICIALES

Es necesario definir unos nuevos conceptos que permitan manejar intersecciones y uniones de forma matricial y vectorial. Se asume que las cantidades manejadas son reales.

Definición 16 (Grado de consistencia)
A pesar de estar definido anteriormente, se redefine aquí usando los operadores $\cup$ e $\cap$ en lugar del máximo y el mínimo:

$$\begin{aligned} dc_j(A_k) &= A_k \circ A_j \\ &= h(A_k \cap A_j) \\ &= \bigcup_x \Big(A_k(x) \bigcap A_j(x) \Big) \end{aligned} \tag{B.1}$$

siendo $h()$ una medida de la altura del conjunto.

Definición 17 (Operación $\times_{\cup\cap}$ entre dos vectores)
Siguiendo las mismas convenciones que para el producto interno de dos vectores, se define

$$\mathbf{a}^T \times_{\cup\cap} \mathbf{b} = \left[\begin{array}{ccc} a_1 & \dots & a_P \end{array} \right] \times_{\cup\cap} \left[\begin{array}{c} b_1 \\ \vdots \\ b_P \end{array} \right] = \bigcup_j \Big(a_j \bigcap b_j \Big) \tag{B.2}$$

Definición 18 (Operación $\times_{\cup\cap}$ entre un vector y una matriz)
Siguiendo las mismas convenciones que para el producto interno entre una matriz y un vector, y considerando los vectores como vectores columna, se define:

$$\mathbf{a}^T \times_{\cup\cap} \mathbf{B} = \begin{bmatrix} a_1 & \dots & a_P \end{bmatrix} \times_{\cup\cap} \begin{bmatrix} b_{11} & \dots & b_{1M} \\ \vdots & \vdots & \vdots \\ b_{P1} & \dots & b_{PM} \end{bmatrix} = \mathbf{c}^T = \begin{bmatrix} c_1 & \dots & c_M \end{bmatrix} \tag{B.3}$$

con

$$c_p = \bigcup_j \left(a_j \bigcap b_{jp} \right) \tag{B.4}$$

El número de filas de la matriz debe ser el mismo que el número de elementos del vector.

La definición de la operación cambiando el orden de los operandos es inmediata, siguiendo la convención para el producto matriz-vector. En particular

$$\mathbf{D} \times_{\cup\cap} \mathbf{a} = \mathbf{e} = \begin{bmatrix} e_1 \\ \vdots \\ e_M \end{bmatrix} \tag{B.5}$$

con

$$e_p = \bigcup_j \left(a_j \bigcap d_{pj} \right) \tag{B.6}$$

Como consecuencia de estas definiciones es claro que $(\mathbf{a}^T \times_{\cup\cap} \mathbf{B})^T = \mathbf{B}^T \times_{\cup\cap} \mathbf{a} = \mathbf{c}$.

Definición 19 (Operación $\times_{\cup\cap}$ entre dos matrices)
(Se asume que el número de columnas de la primera coincide con el número de filas de la segunda). En este caso:

$$\mathbf{A} \times_{\cup\cap} \mathbf{B} = \mathbf{C} \tag{B.7}$$

con

$$c_{pq} = \bigcup_j \left(a_{pj} \bigcap b_{jq} \right) \tag{B.8}$$

Como se demostrará en la sección B.4 esta operación es asociativa.

Definición 20 (Intersección escalar-matriz)
La intersección se calcula para cada elemento de la matriz

$$\alpha \bigcap \mathbf{A} = \alpha \bigcap \begin{bmatrix} a_{11} & \dots & a_{1M} \\ \vdots & \vdots & \vdots \\ a_{P1} & \dots & a_{PM} \end{bmatrix} = \bigcap \begin{bmatrix} \alpha \cap a_{11} & \dots & \alpha \cap a_{1M} \\ \vdots & \vdots & \vdots \\ \alpha \cap a_{P1} & \dots & \alpha \cap a_{PM} \end{bmatrix} \tag{B.9}$$

Definición 21 (Unión de matriz)
La unión se calcula para cada elemento

$$\mathbf{A} = \bigcup_l \left(\mathbf{A}^l \right) \tag{B.10}$$

con

$$a_{pq} = \bigcup_l \left(a_{pq}^l \right) \tag{B.11}$$

Definición 22 (Intersección tensorial de Kronecker)
La intersección de Kronecker entre las matrices $\mathbf{P}$ y $\mathbf{Q}$ se calcula

$$\mathbf{W} = \mathbf{P} \otimes_{\cap} \mathbf{Q} = \begin{bmatrix} p_{11} \cap \mathbf{Q} & p_{12} \cap \mathbf{Q} & \cdots & p_{1M} \cap \mathbf{Q} \\ p_{21} \cap \mathbf{Q} & p_{22} \cap \mathbf{Q} & \cdots & p_{2M} \cap \mathbf{Q} \\ \cdots & \cdots & \cdots & \cdots \\ p_{P1} \cap \mathbf{Q} & p_{P2} \cap \mathbf{Q} & \cdots & p_{PM} \cap \mathbf{Q} \end{bmatrix} \tag{B.12}$$

A pesar de no usarse en este apéndice, se definirá también la unión tensorial de Kronecker, por ser ésta de utilidad en reglas disyuntivas (sección 4.5).

Definición 23 (Unión tensorial de Kronecker)
La unión de Kronecker entre las matrices $\mathbf{P}$ y $\mathbf{Q}$ se calcula

$$\mathbf{W} = \mathbf{P} \otimes_{\cup} \mathbf{Q} = \begin{bmatrix} p_{11} \cup \mathbf{Q} & p_{12} \cup \mathbf{Q} & \cdots & p_{1M} \cup \mathbf{Q} \\ p_{21} \cup \mathbf{Q} & p_{22} \cup \mathbf{Q} & \cdots & p_{2M} \cup \mathbf{Q} \\ \cdots & \cdots & \cdots & \cdots \\ p_{P1} \cup \mathbf{Q} & p_{P2} \cup \mathbf{Q} & \cdots & p_{PM} \cup \mathbf{Q} \end{bmatrix} \tag{B.13}$$

con $p_{ij} \cup \mathbf{Q}$ una operación similar a (B.9) intercambiando el operador $\bigcap$ por $\bigcup$.

B.2 SISTEMA BORROSO CON UNA ENTRADA Y UNA SALIDA

Supóngase un sistema borroso con una entrada X y una salida Z, siendo ambas conjuntos borrosos definidos sobre respectivas VLs. La VL de entrada consiste en M posibles conjuntos borrosos A_k definidos sobre el universo $U \subset R$ y la VL de salida consta de L conjuntos borrosos D_n definidos sobre el universo $W \subset R$. El sistema lleva a cabo una transformación $f : U \subset R \mapsto W \subset R$.

Asúmase que X puede ser escrita como una composición de los conjuntos borrosos A_k de la VL sobre la que está definida. Específicamente

$$X = \bigcup_{k=1}^{M} [\beta_k \bigcap A_k] = \begin{bmatrix} \beta_1 & \beta_2 & \cdots & \beta_M \end{bmatrix} \times_{\cup\cap} \begin{bmatrix} A_1 & A_2 & \cdots & A_M \end{bmatrix}^T = \boldsymbol{\beta}^T \times_{\cup\cap} \mathbf{A} \tag{B.14}$$

En este caso, el *dc* entre X y $A_{p(j)}$ (siendo $p(j)$ una función de indexado) puede escribirse, usando las propiedades asociativa, conmutativa y distributiva de los operadores $\bigcup$ e $\bigcap$

$$\begin{aligned} dc_j(X) = X \circ A_{p(j)} &= \bigcup_x \left[X(x) \bigcap A_{p(j)}(x) \right] \\ &= \bigcup_x \left[\left(\bigcup_{k=1}^{M} (\beta_k \bigcap A_k(x)) \right) \bigcap A_{p(j)}(x) \right] \end{aligned}$$

$$
\begin{aligned}
&= \bigcup_x \left[\bigcup_{k=1}^{M} \Big((\beta_k \bigcap A_k(x)) \bigcap A_{p(j)}(x) \Big) \right] \\
&= \bigcup_x \left[\bigcup_{k=1}^{M} \Big(\beta_k \bigcap (A_k(x) \bigcap A_{p(j)}(x)) \Big) \right] \\
&= \bigcup_{k=1}^{M} \left(\beta_k \bigcap \left[\bigcup_x (A_k(x) \bigcap A_{p(j)}(x)) \right] \right) \\
&= \bigcup_{k=1}^{M} \Big(\beta_k \bigcap \Big[A_k \circ A_{p(j)} \Big] \Big) = \bigcup_{k=1}^{M} \Big(\beta_k \bigcap \Big[A_{p(j)} \circ A_k \Big] \Big) \\
&= \left[\begin{array}{cccc} \beta_1 & \beta_2 & \cdots & \beta_M \end{array} \right] \times_{\cup\cap} \left[\begin{array}{l} (A_{p(j)} \circ A_1) \\ (A_{p(j)} \circ A_2) \\ \vdots \\ (A_{p(j)} \circ A_M) \end{array} \right]
\end{aligned} \tag{B.15}
$$

Por otro lado, la salida D_o se calcula aplicando la ecuación (2.56),

$$D_o(z) = \bigcup_j [dc_j(X) \bigcap D_{r(j)}(z)] \tag{B.16}$$

donde $r(j)$ es un operador de indexado apropiado, que selecciona el consecuente correcto para la regla j-ésima de la base de reglas. Si se denotan las reglas por el par (i, n), donde i es el número de regla, y n indexa el conjunto de salida, D_n. Sea $nr(n)$ el número de reglas para las que el consecuente es D_n; para la regla (i, n) el índice i toma valores en el intervalo $i = \{1, nr(n)\}$. Claramente $\sum_{n=1}^{L} nr(n) = M$. Se define $dc^{(i,n)}(X)$ como el grado de consistencia de la entrada X con el antecedente de la regla (i, n). Con esta terminología, la ecuación (B.16) puede reescribirse

$$D_o(z) = \bigcup_{n=1}^{L} \left(\bigcup_{i=1}^{nr(n)} \Big[dc^{(i,n)}(X) \bigcap D_n(z) \Big] \right)$$

que da lugar a

$$
\begin{aligned}
D_o(z) &= \bigcup_{n=1}^{L} \left(\bigcup_{i=1}^{nr(n)} \Big[dc^{(i,n)}(X) \Big] \bigcap D_n(z) \right) \\
&= \bigcup_{n=1}^{L} \Big(\gamma_n \bigcap D_n(z) \Big) \\
&= \left[\begin{array}{cccc} \gamma_1 & \gamma_2 & \cdots & \gamma_L \end{array} \right] \times_{\cup\cap} \left[\begin{array}{cccc} D_1 & D_2 & \cdots & D_L \end{array} \right]^T = \boldsymbol{\gamma}^T \times_{\cup\cap} \mathbf{D}
\end{aligned}
$$

donde $\gamma_n = \bigcup_{i=1}^{nr(n)} dc^{(i,n)}(X)$. Por lo tanto, la ecuación (B.16) puede escribirse (eliminando la dependencia explícita con z)

$$D_o = \left[\begin{array}{cccc} \gamma_1 & \gamma_2 & \cdots & \gamma_L \end{array} \right] \times_{\cup\cap} \left[\begin{array}{cccc} D_1 & D_2 & \cdots & D_L \end{array} \right]^T = \boldsymbol{\gamma}^T \times_{\cup\cap} \mathbf{D} \tag{B.17}$$

Por otro lado, si se reescribe (B.16) en forma matricial, se obtiene

$$D_o = \left[\begin{array}{cccc} dc_1(X) & dc_2(X) & \cdots & dc_M(X) \end{array}\right] \times_{\cup\cap} \left[\begin{array}{cccc} D_{r(1)} & D_{r(2)} & \cdots & D_{r(M)} \end{array}\right]^T \quad \text{(B.18)}$$

lo cual, haciendo uso de (B.15) puede expresarse

$$D_o = \left(\left[\begin{array}{cccc} \beta_1 & \cdots & \beta_M \end{array}\right] \times_{\cup\cap} \left[\begin{array}{ccc} (A_{p(1)} \circ A_1) & \cdots & (A_{p(M)} \circ A_1) \\ (A_{p(1)} \circ A_2) & \cdots & (A_{p(M)} \circ A_2) \\ \vdots & \vdots & \vdots \\ (A_{p(1)} \circ A_M) & \cdots & (A_{p(M)} \circ A_M) \end{array}\right] \right) \times_{\cup\cap} \left[\begin{array}{c} D_{r(1)} \\ D_{r(2)} \\ \vdots \\ D_{r(M)} \end{array}\right] \quad \text{(B.19)}$$

Sea $\mathbf{C}$ una matriz de selección de reglas (de dimensiones $M \times L$) definida del mismo modo que para el caso SAM. Entonces, puede escribirse

$$\left[\begin{array}{c} D_{r(1)} \\ D_{r(2)} \\ \vdots \\ D_{r(M)} \end{array}\right] = \mathbf{C} \times_{\cup\cap} \left[\begin{array}{c} D_1 \\ D_2 \\ \vdots \\ D_L \end{array}\right] \quad \text{(B.20)}$$

Uniendo las expresiones se llega a

$$D_o = \left(\left[\begin{array}{cccc} \beta_1 & \cdots & \beta_M \end{array}\right] \times_{\cup\cap} \left[\begin{array}{ccc} (A_{p(1)} \circ A_1) & \cdots & (A_{p(M)} \circ A_1) \\ (A_{p(1)} \circ A_2) & \cdots & (A_{p(M)} \circ A_2) \\ \vdots & \vdots & \vdots \\ (A_{p(1)} \circ A_M) & \cdots & (A_{p(M)} \circ A_M) \end{array}\right] \right) \times_{\cup\cap} \left(\mathbf{C} \times_{\cup\cap} \left[\begin{array}{c} D_1 \\ D_2 \\ \vdots \\ D_L \end{array}\right] \right) \quad \text{(B.21)}$$

que debido a la propiedad asociativa de las operaciones matriciales usadas (ver apéndice B.4) puede escribirse

$$D_o = \left[\begin{array}{cccc} \beta_1 & \cdots & \beta_M \end{array}\right] \times_{\cup\cap} \left(\left[\begin{array}{ccc} (A_{p(1)} \circ A_1) & \cdots & (A_{p(M)} \circ A_1) \\ (A_{p(1)} \circ A_2) & \cdots & (A_{p(M)} \circ A_2) \\ \vdots & \vdots & \vdots \\ (A_{p(1)} \circ A_M) & \cdots & (A_{p(M)} \circ A_M) \end{array}\right] \times_{\cup\cap} \mathbf{C} \times_{\cup\cap} \left[\begin{array}{c} D_1 \\ D_2 \\ \vdots \\ D_L \end{array}\right] \right) \quad \text{(B.22)}$$

Usando las expresiones (B.17) y (B.22) puede escribirse

$$\left[\begin{array}{cccc} \gamma_1 & \cdots & \gamma_L \end{array}\right] \times_{\cup\cap} \left[\begin{array}{c} D_1 \\ D_2 \\ \cdots \\ D_L \end{array}\right] = \left[\begin{array}{cccc} \beta_1 & \cdots & \beta_M \end{array}\right] \times_{\cup\cap} \left(\left[\begin{array}{ccc} (A_{p(1)} \circ A_1) & \cdots & (A_{p(M)} \circ A_1) \\ (A_{p(1)} \circ A_2) & \cdots & (A_{p(M)} \circ A_2) \\ \vdots & \vdots & \vdots \\ (A_{p(1)} \circ A_M) & \cdots & (A_{p(M)} \circ A_M) \end{array}\right] \times_{\cup\cap} \mathbf{C} \right) \times_{\cup\cap} \left[\begin{array}{c} D_1 \\ D_2 \\ \vdots \\ D_L \end{array}\right] \quad \text{(B.23)}$$

y consecuentemente

$$\begin{aligned}
\left[\begin{array}{ccc}\gamma_1 & \cdots & \gamma_L\end{array}\right] &= \left[\begin{array}{ccc}\beta_1 & \cdots & \beta_M\end{array}\right] \times_{\cup\cap} \left(\left[\begin{array}{ccc} (A_{p(1)} \circ A_1) & \cdots & (A_{p(M)} \circ A_1) \\ (A_{p(1)} \circ A_2) & \cdots & (A_{p(M)} \circ A_2) \\ \vdots & \vdots & \vdots \\ (A_{p(1)} \circ A_M) & \cdots & (A_{p(M)} \circ A_M) \end{array}\right] \times_{\cup\cap} \mathbf{C} \right) \\
\boldsymbol{\gamma}^T &= \boldsymbol{\beta}^T \times_{\cup\cap} \left(\mathbf{G}^T \times_{\cup\cap} \mathbf{C} \right) \\
\boldsymbol{\gamma} &= \left(\mathbf{G}^T \times_{\cup\cap} \mathbf{C} \right)^T \times_{\cup\cap} \boldsymbol{\beta} \\
\boldsymbol{\gamma} &= \boldsymbol{\Omega} \times_{\cup\cap} \boldsymbol{\beta} \qquad \text{(B.24)}
\end{aligned}$$

siendo $\boldsymbol{\Omega} = \mathbf{C}^T \times_{\cup\cap} \mathbf{G}$ una matriz de transición borrosa, tal y como se veía para el caso SAM.

B.3 SISTEMA BORROSO DE 2 ENTRADAS Y UNA SALIDA

Supóngase que al sistema descrito en la sección anterior se le añade una segunda entrada Y, consistente en una VL con N posibles conjuntos borrosos B_l definidos sobre el universo $V \subset R$. El sistema realiza la transformación $f : U \times V \subset R^2 \mapsto W \subset R$. Las entradas de forma vectorial serán

$$\begin{aligned}
X &= \bigcup_{k=1}^{M} [\beta_k \bigcap A_k] = \boldsymbol{\beta}^T \times_{\cup\cap} \mathbf{A} \\
Y &= \bigcup_{l=1}^{N} [\alpha_l \bigcap B_l] = \left[\begin{array}{cccc}\alpha_1 & \alpha_2 & \cdots & \alpha_N\end{array}\right] \times_{\cup\cap} \left[\begin{array}{cccc} B_1 & B_2 & \cdots & B_N \end{array}\right]^T = \boldsymbol{\alpha}^T \times_{\cup\cap} \mathbf{B}
\end{aligned}$$

En este caso la salida se obtiene usando la ecuación (2.56) con el *dc* definido en la ecuación (2.58); puede escribirse

$$\begin{aligned}
D_o(z) &= \bigcup_j [dc_j(X,Y) \bigcap D_j(z)] \\
&= \bigcup_j [(dc_j(X) \bigcap dc_j(Y)) \bigcap D_j(z)] \qquad \text{(B.25)}
\end{aligned}$$

La estructura de la ecuación (B.15) se mantiene para la segunda entrada, por lo que

$$\begin{aligned}
dc_j(Y) = Y \circ B_{q(j)} &= \bigcup_{l=1}^{N} \left(\alpha_l \bigcap \left[B_{q(j)} \circ B_l \right] \right) \\
&= \left[\begin{array}{cccc}\alpha_1 & \alpha_2 & \cdots & \alpha_N\end{array}\right] \times_{\cup\cap} \left[\begin{array}{c} (B_{q(j)} \circ B_1) \\ (B_{q(j)} \circ B_2) \\ \vdots \\ (B_{q(j)} \circ B_N) \end{array}\right] \qquad \text{(B.26)}
\end{aligned}$$

con $q(j)$ una función de indexado. Denotando por simplicidad $\Lambda_{lj} = \left(\alpha_l \bigcap \left[B_{q(j)} \circ B_l\right]\right)$, la ecuación (B.25) puede reescribirse

$$\begin{aligned} D_o(z) &= \bigcup_j \left[dc_j(X) \bigcap \left(\bigcup_{l=1}^{N} \Lambda_{lj} \right) \bigcap D_j(z) \right] \\ &= \bigcup_j \left[\left(\bigcup_{l=1}^{N} \Lambda_{lj} \right) \bigcap (dc_j(X) \bigcap D_j(z)) \right] \\ &= \bigcup_j \bigcup_{l=1}^{N} \left[\Lambda_{lj} \bigcap dc_j(X) \bigcap D_j(z) \right] \\ &= \bigcup_{l=1}^{N} \bigcup_j \left[\left(\alpha_l \bigcap \left[B_{q(j)} \circ B_l \right] \right) \bigcap (dc_j(X) \bigcap D_j(z)) \right] \\ &= \bigcup_{l=1}^{N} \alpha_l \bigcap \left(\bigcup_j \left[\left[(B_{q(j)} \circ B_l) \bigcap dc_j(X) \right] \bigcap D_j(z) \right] \right) \end{aligned} \tag{B.27}$$

y

$$\begin{aligned} (B_{q(j)} \circ B_l) \bigcap dc_j(X) &= (B_{q(j)} \circ B_l) \bigcap \left[\bigcup_{k=1}^{M} \left(\beta_k \bigcap (A_{p(j)} \circ A_k) \right) \right] \\ &= \bigcup_{k=1}^{M} \left[\beta_k \bigcap \left((A_{p(j)} \circ A_k) \bigcap (B_{q(j)} \circ B_l) \right) \right] \\ &= \begin{bmatrix} \beta_1 & \beta_2 & \cdots & \beta_M \end{bmatrix} \times_{\cup\cap} \begin{bmatrix} (A_{p(j)} \circ A_1) \bigcap (B_{q(j)} \circ B_l) \\ (A_{p(j)} \circ A_2) \bigcap (B_{q(j)} \circ B_l) \\ \vdots \\ (A_{p(j)} \circ A_M) \bigcap (B_{q(j)} \circ B_l) \end{bmatrix} \\ &= \boldsymbol{\beta}^T \times_{\cup\cap} \mathbf{g}_{p(j),q(j),l} \end{aligned} \tag{B.28}$$

La expresión afectada por el operador $\bigcup_j$ en la ecuación (B.27) puede ser reescrita, con la matriz $\mathbf{C}$ definida de forma usual,

$$\bigcup_j \left[\left[(B_{q(j)} \circ B_l) \bigcap dc_j(X) \right] \bigcap D_j(z) \right] =$$

$$= \begin{bmatrix} D_{r(1)} & D_{r(2)} & \cdots & D_{r(MN)} \end{bmatrix} \times_{\cup\cap} \begin{bmatrix} \boldsymbol{\beta}^T \times_{\cup\cap} \mathbf{g}_{p(1),q(1),l} \\ \boldsymbol{\beta}^T \times_{\cup\cap} \mathbf{g}_{p(2),q(2),l} \\ \cdots \\ \boldsymbol{\beta}^T \times_{\cup\cap} \mathbf{g}_{p(MN),q(MN),l} \end{bmatrix}$$

$$= \begin{bmatrix} D_{r(1)} & D_{r(2)} & \cdots & D_{r(MN)} \end{bmatrix} \times_{\cup\cap} \left(\begin{bmatrix} \mathbf{g}^T_{p(1),q(1),l} \\ \mathbf{g}^T_{p(2),q(2),l} \\ \cdots \\ \mathbf{g}^T_{p(MN),q(MN),l} \end{bmatrix} \times_{\cup\cap} \boldsymbol{\beta} \right)$$

$$= \begin{bmatrix} D_{r(1)} & D_{r(2)} & \cdots & D_{r(MN)} \end{bmatrix} \times_{\cup\cap} (\mathbf{G}_l \times_{\cup\cap} \boldsymbol{\beta})$$

$$= \left(\left[\begin{array}{cccc} D_1 & D_2 & \cdots & D_L \end{array} \right] \times_{\cup\cap} \mathbf{C}^T \right) \times_{\cup\cap} (\mathbf{G}_l \times_{\cup\cap} \boldsymbol{\beta})$$
$$= \left(\mathbf{D}^T \times_{\cup\cap} \mathbf{C}^T \times_{\cup\cap} \mathbf{G}_l \right) \times_{\cup\cap} \boldsymbol{\beta} \tag{B.29}$$

La matriz $\mathbf{G}_l$ que aparece en la expresión se define

$$\mathbf{G}_l = \left[\begin{array}{ccc} (A_{p(1)} \circ A_1) \bigcap (B_{q(1)} \circ B_l) & \cdots & (A_{p(1)} \circ A_M) \bigcap (B_{q(1)} \circ B_l) \\ (A_{p(2)} \circ A_1) \bigcap (B_{q(2)} \circ B_l) & \cdots & (A_{p(2)} \circ A_M) \bigcap (B_{q(2)} \circ B_l) \\ \vdots & \vdots & \vdots \\ (A_{p(MN)} \circ A_1) \bigcap (B_{q(MN)} \circ B_l) & \cdots & (A_{p(MN)} \circ A_M) \bigcap (B_{q(MN)} \circ B_l) \end{array} \right] \tag{B.30}$$

Con este resultado previo, y usando la ecuación (B.17), puede reescribirse la ecuación (B.27) como

$$D_o = \mathbf{D}^T \times_{\cup\cap} \boldsymbol{\gamma} = \bigcup_{l=1}^{N} \alpha_l \bigcap \left(\left(\mathbf{D}^T \times_{\cup\cap} \mathbf{C}^T \times_{\cup\cap} \mathbf{G}_l \right) \times_{\cup\cap} \boldsymbol{\beta} \right) \tag{B.31}$$

que tal y como se indica en la sección B.5 se puede reescribir

$$D_o = \mathbf{D}^T \times_{\cup\cap} \left[\left(\bigcup_{l=1}^{N} \alpha_l \bigcap (\mathbf{C}^T \times_{\cup\cap} \mathbf{G}_l) \right) \times_{\cup\cap} \boldsymbol{\beta} \right] \tag{B.32}$$

La relación entrada-salida queda

$$\begin{aligned} \boldsymbol{\gamma} &= \left(\bigcup_{l=1}^{N} \alpha_l \bigcap (\mathbf{C}^T \times_{\cup\cap} \mathbf{G}_l) \right) \times_{\cup\cap} \boldsymbol{\beta} \\ &= \left(\bigcup_{l=1}^{N} \alpha_l \bigcap \boldsymbol{\Omega}_l \right) \times_{\cup\cap} \boldsymbol{\beta} \end{aligned} \tag{B.33}$$

siendo $\boldsymbol{\Omega}_l$ la matriz de transición del sistema cuando $Y = B_l$. La matriz $\mathbf{G}_l$ por su parte, puede definirse de forma más compacta usando intersecciones tensoriales de Kronecker. De este modo la matriz $\mathbf{G}_l$ puede escribirse como

$$\mathbf{G}_l = \mathbf{R}_\mathbf{A} \otimes_\cap [\mathbf{R}_\mathbf{B} \mathbf{E}_l] \tag{B.34}$$

siendo $\mathbf{E}_l$ un vector de selección de columna, similar al definido para el caso SAM.

B.4 SOBRE LA ASOCIATIVIDAD DEL OPERADOR MATRICIAL $\times_{\cup\cap}$

Sea a_{is} el elemento de la fila i y la columna s de la matriz $\mathbf{A}$. b_{sr} y c_{ij} denotan elementos de las matrices $\mathbf{B}$ y $\mathbf{C}$ respectivamente, de acuerdo con sus subíndices. El propósito de esta sección es mostrar que la siguiente igualdad se cumple

$$\mathbf{D} = (\mathbf{A} \times_{\cup\cap} \mathbf{B}) \times_{\cup\cap} \mathbf{C} = \mathbf{A} \times_{\cup\cap} (\mathbf{B} \times_{\cup\cap} \mathbf{C}) \tag{B.35}$$

Para probarlo se operará elemento a elemento. Partiendo de que las matrices utilizadas tienen sus dimensiones coherentes con las operaciones, se escribe el elemento d_{ij} usando las operaciones del lado derecho de la igualdad.

$$d_{ij} = \bigcup_r \Big(p_{ir} \bigcap c_{rj} \Big) \tag{B.36}$$

con

$$p_{ir} = \bigcup_s \Big(a_{is} \bigcap b_{sr} \Big) \tag{B.37}$$

Consecuentemente

$$\begin{aligned} d_{ij} &= \bigcup_r \left(\left[\bigcup_s \Big(a_{is} \bigcap b_{sr} \Big) \right] \bigcap c_{rj} \right) \\ &= \bigcup_r \left(\bigcup_s \Big[a_{is} \bigcap b_{sr} \bigcap c_{rj} \Big] \right) \\ &= \bigcup_r \left(\bigcup_s \Big[a_{is} \bigcap (b_{sr} \bigcap c_{rj}) \Big] \right) \\ &= \bigcup_s \left[a_{is} \bigcap \left(\bigcup_r (b_{sr} \bigcap c_{rj}) \right) \right] \end{aligned} \tag{B.38}$$

que resulta ser igual a las operaciones del lado izquierdo, con lo que la igualdad se cumple, q.e.d.

B.5 SOBRE LA ECUACIÓN (B.32)

En esta sección se verá cómo deducir la ecuación (B.32) a partir de la ecuación (B.31). Por simplicidad en la notación, se define

$$\mathbf{C}^T \times_{\cup\cap} \mathbf{G}_l = [\ \boldsymbol{\lambda}_1^l \quad \dots \quad \boldsymbol{\lambda}_M^l\ ] \tag{B.39}$$

con $\boldsymbol{\lambda}_j^l$ un vector columna igual a la j-ésima columna ($j = \{1, \dots, M\}$) de la matriz $\mathbf{C}^T \times_{\cup\cap} \mathbf{G}_l$. Se define λ_{nj}^l como la n-ésima entrada ($n = \{1, \dots, L\}$) del vector columna $\boldsymbol{\lambda}_j^l$. De manera similar se define

$$\mathbf{D}^T \times_{\cup\cap} \mathbf{C}^T \times_{\cup\cap} \mathbf{G}_l = \mathbf{D}^T \times_{\cup\cap} [\ \boldsymbol{\lambda}_1^l \quad \dots \quad \boldsymbol{\lambda}_M^l\ ] = [\ \varphi_1^l \quad \dots \quad \varphi_M^l\ ] \tag{B.40}$$

De acuerdo con esta definición claramente

$$\varphi_j^l = \bigcup_n \Big(D_n \bigcap \lambda_{nj}^l \Big) \tag{B.41}$$

Con esta terminología puede escribirse

$$\bigcup_l \alpha_l \bigcap \Big(\Big(\mathbf{D}^T \times_{\cup\cap} \mathbf{C}^T \times_{\cup\cap} \mathbf{G}_l \Big) \times_{\cup\cap} \boldsymbol{\beta} \Big) \;=\; \bigcup_l \left(\alpha_l \bigcap \left[\bigcup_j \Big(\varphi_j^l \bigcap \beta_j \Big) \right] \right)$$

$$
\begin{aligned}
&= \bigcup_l \bigcup_j \left(\alpha_l \bigcap \varphi_j^l \bigcap \beta_j\right) = \bigcup_j \bigcup_l \left(\alpha_l \bigcap \varphi_j^l \bigcap \beta_j\right) \\
&= \bigcup_j \bigcup_l \left(\alpha_l \bigcap \left[\bigcup_n \left(D_n \bigcap \lambda_{nj}^l\right)\right] \bigcap \beta_j\right) \\
&= \bigcup_j \bigcup_l \bigcup_n \left(\left[D_n \bigcap \lambda_{nj}^l\right] \bigcap \alpha_l \bigcap \beta_j\right) \\
&= \bigcup_j \bigcup_l \bigcup_n \left(D_n \bigcap \lambda_{nj}^l \bigcap \alpha_l \bigcap \beta_j\right) \\
&= \bigcup_n D_n \bigcap \left(\bigcup_j \bigcup_l \left(\lambda_{nj}^l \bigcap \alpha_l \bigcap \beta_j\right)\right) \\
&= \bigcup_n D_n \bigcap \left(\bigcup_j \left[\beta_j \bigcap \left(\bigcup_l (\lambda_{nj}^l \bigcap \alpha_l)\right)\right]\right) \\
&= \bigcup_n D_n \bigcap \left(\bigcup_j \left[\beta_j \bigcap \tau_{nj}\right]\right) \\
&= \mathbf{D}^T \times_{\cup\cap} \begin{bmatrix} \psi_1 \\ \vdots \\ \psi_L \end{bmatrix} \qquad \text{(B.42)}
\end{aligned}
$$

con

$$
\psi_n = \begin{bmatrix} \tau_{n1} & \dots & \tau_{nM} \end{bmatrix} \times_{\cup\cap} \begin{bmatrix} \beta_1 \\ \vdots \\ \beta_M \end{bmatrix}. \qquad \text{(B.43)}
$$

Consecuentemente

$$
\begin{aligned}
\begin{bmatrix} \psi_1 \\ \vdots \\ \psi_L \end{bmatrix} &= \begin{bmatrix} \tau_{11} & \dots & \tau_{1M} \\ \vdots & \vdots & \vdots \\ \tau_{L1} & \dots & \tau_{LM} \end{bmatrix} \times_{\cup\cap} \begin{bmatrix} \beta_1 \\ \vdots \\ \beta_M \end{bmatrix} \\
&= \left(\bigcup_l \alpha_l \begin{bmatrix} \lambda_{11}^l & \dots & \lambda_{1M}^l \\ \vdots & \vdots & \vdots \\ \lambda_{L1}^l & \dots & \lambda_{LM}^l \end{bmatrix}\right) \times_{\cup\cap} \begin{bmatrix} \beta_1 \\ \vdots \\ \beta_M \end{bmatrix} \qquad \text{(B.44)}
\end{aligned}
$$

Insertando este resultado en la ecuación (B.42) se obtiene

$$
\bigcup_l \alpha_l \bigcap \left(\left(\mathbf{D}^T \times_{\cup\cap} \mathbf{C}^T \times_{\cup\cap} \mathbf{G}_l\right) \times_{\cup\cap} \boldsymbol{\beta}\right) = \mathbf{D}^T \times_{\cup\cap} \left[\left(\bigcup_{l=1}^N \alpha_l \bigcap (\mathbf{C}^T \times_{\cup\cap} \mathbf{G}_l)\right) \times_{\cup\cap} \boldsymbol{\beta}\right] \qquad \text{(B.45)}
$$

que coincide con la ecuación (B.32).

C

APLICACIÓN DE LA METODOLOGÍA FITM AL CÁLCULO DE ÁRBOLES DE DECISIÓN BORROSOS

C.1 INTRODUCCIÓN A LOS ÁRBOLES DE DECISIÓN

En *aprendizaje supervisado*, una muestra se representa por un conjunto de características expresadas con algún lenguaje descriptivo. Las muestras de las que se conoce la clasificación, las cuales se usan para el entrenamiento, se denominan *ejemplos*. El objetivo es inducir procedimientos de decisión con un sesgo *discriminativo* (en la mayoría de los casos), *descriptivo* o *taxonómico*, para clasificar otras muestras. Siguiendo el *principio de comprensión* (que impone al proceso de decisión el uso de lenguajes y mecanismos adecuados para la comprensión y la interpretación humanas), los sistemas simbólicos son de una importancia extrema en muchas aplicaciones y entornos. Entre ellos, los árboles de decisión son uno de los más populares.

ID3 y CART [57] son los dos algoritmos de aprendizaje más importantes entre los que usan particionado recursivo. Sus ideas básicas son las mismas: partir el espacio de muestras de un modo dependiente de los datos, y representar la partición como un árbol. Un ejemplo de los árboles obtenidos por estos dos métodos se encuentra representado en la figura C.1. Una propiedad importante de esta clase de árboles es que tratan de minimizar el tamaño del árbol al mismo tiempo que optimizan alguna medida de calidad.

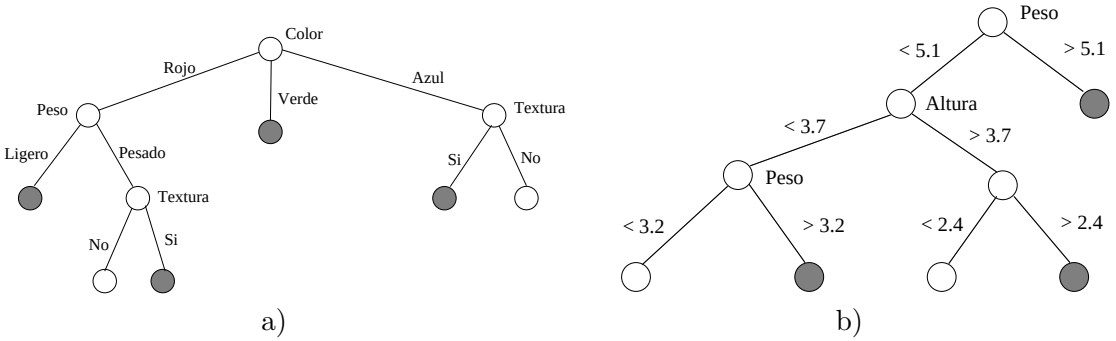


Figura C.1 Ejemplos de árboles de decisión generados por a) ID3 b) CART.

Existen ciertas diferencias entre ambos tipos de árboles. ID3 [9, 134] asume dominios discretos de baja cardinalidad. Esto supone una gran ventaja a la hora de aumentar la

comprensibilidad del conocimiento inducido, pero puede requerir particionado *a priori*. Existen numerosos estudios acerca de cómo se debe realizar esta partición, así como numerosas técnicas de particionado *a priori*. Debido a un posible ruido, a ausencia de alguna característica en las descripciones de los ejemplos, a un número insuficiente de ejemplos de entrenamiento o a una mala partición, podría tenerse un árbol con inconsistencias o incompleto. A pesar de ello, hay que resaltar que estos árboles escogen las mejores muestras de todo el conjunto de entrenamiento. Además, dado que cada atributo produce un nodo, aumenta la comprensibilidad del resultado. En el ejemplo de la Fig. C.1-a, puede verse un árbol de decisión generado usando un algoritmo ID3. La bifurcación en ramas se hace en función de valores predefinidos de los distintos atributos.

El algoritmo CART no requiere partición *a priori*. Las condiciones en el árbol están basadas en umbrales (para dominios continuos), que se calculan dinámicamente. A causa de ello, a lo largo del árbol se puede usar una característica varias veces con diferentes umbrales. Generalmente el número de umbrales es igual al de ejemplos usados. De este modo, se aumenta la calidad del árbol (ya que no existe una partición *a priori*; ésta se realiza dinámicamente) pero se reduce su comprensibilidad. CART, permite, además introducir nuevas características, restringidas a combinaciones lineales de las existentes. En el ejemplo de la Fig. C.1-b, puede verse un árbol de decisión generado usando un algoritmo CART. Como puede observarse, el criterio de decisión entre ramas se basa en umbrales numéricos.

Después de la construcción, ambos emplean el mismo mecanismo de inferencia. Las características de una nueva muestra son contrastadas con las condiciones de árbol. En caso de existir un nodo *hoja* que cumpla exactamente esas características, la muestra es clasificada como de ese tipo.

C.1.1 El algoritmo ID3

Para la presente sección va a considerarse a modo de ejemplo el caso particular de la clasificación de patrones pertenecientes a dos clases distintas, a las que se denominará $+$ y $-$. La extensión a un número mayor de clases es inmediata.

A partir de un conjunto de muestras clasificadas $\mathbf{D}$, ID3 genera un árbol de decisión que clasifica correctamente todos los objetos de $\mathbf{D}$ y que puede ser utilizado para la clasificación de otros objetos. Si los ejemplos de $\mathbf{D}$ son significativos, esta última clasificación tendrá éxito.

Todos los objetos considerados vienen caracterizados por una serie de atributos A, B, C,... con dominios finitos y reducidos, de valores $\{A_1, A_2, \ldots, A_m\}$, $\{B_1, B_2, \ldots, B_n\}$, $\{C_1, C_2, \ldots, C_l\}$,... Si el dominio de algún atributo es continuo, debe ser discretizado. Si es muy grande, puede resultar conveniente transformarlo en otro más pequeño en el que cada valor corresponda a un subconjunto de valores del dominio original. El tamaño de estos dominios es un factor muy importante en la complejidad del algoritmo, de ahí el interés en reducirlos razonablemente.

Cada objeto viene descrito por los valores que toman en él los distintos atributos y por la clase a la que pertenece. Por lo tanto el conjunto **D** se define del siguiente modo:

$$\mathbf{D} = \{A_i B_j C_k \ldots +, A_r B_s C_t \ldots -, \ldots\}$$

Las hojas del árbol de decisión generado corresponden a las clases de salida (+ ó −). Los nodos *no hoja* corresponden a los atributos. De cada uno de los nodos saldrán tantas ramas como valores tenga el atributo asociado. La clasificación de un nuevo objeto se realiza recorriendo el árbol desde la raíz hasta alcanzar alguna hoja. En cada nodo se toma el camino correspondiente al valor del atributo asociado que tenga el objeto a clasificar. La clase que se encuentra en la hoja alcanzada es la que se asigna al objeto en cuestión.

C.1.2 Descripción del algoritmo ID3

El algoritmo ID3 parte de la raíz del árbol en la que se encuentran todos los objetos de **D**. En los nodos se pueden presentar tres casos:

1. Todos los objetos de **D** pertenecen a la misma clase. Esto significa que al llegar a este nodo ya se conoce la clase sin necesidad de observar ningún atributo más. Por lo tanto, éste será un nodo hoja etiquetado con la clase a la que pertenecen todos sus objetos.
2. El conjunto **D** asociado está vacío. Esto se produce cuando no se dispone de ningún ejemplo con los valores de los atributos que conducen hasta este nodo. Dicho nodo será una hoja y se etiqueta aleatoriamente con + o −.
3. **D** contiene objetos de ambas clases. la decisión no se puede realizar todavía, por lo que se elige un atributo T y se crean tantos nodos descendientes como valores pueda tomar T. El conjunto **D** se divide en D_1, D_2..., de tal modo que en cada D_I estén todos los objetos de **D** que toman el valor T_i para el atributo T Cada rama corresponde a un T_i y el nodo al que conduce está asociado al correspondiente D_I.

El algoritmo funciona correctamente a menos que haya objetos que teniendo idénticos valores en todos sus atributos pertenezcan a distintas clases.

Selección de atributos

La estrategia utilizada por ID3 para la selección del atributo en cada nodo se basa en la Teoría de la Información, tal como se describe a continuación.

Consideremos un nodo en el que se tiene un conjunto de objetos **D** de los que m pertenecen a la clase + y n pertenecen a la clase −. La probabilidad de que un objeto

de **D** pertenezca a la clase + se puede aproximar mediante la frecuencia relativa:

$$P(+) = \frac{m}{m+n} \tag{C.1}$$

y de igual modo

$$P(-) = \frac{n}{m+n} \tag{C.2}$$

De forma general [57], podemos definir el cardinal de **D** (lo representamos por $|\mathbf{D}|$) como el número de elementos presentes en el conjunto. Del mismo modo, podemos definir $|\mathbf{D}^{C_k}|$ como la suma de todos los elementos de **D** que pertenecen a la clase C_k. La probabilidad (aproximada por la frecuencia relativa) de que un objeto pertenezca a la clase C_k viene dada por:

$$P_k = \frac{|\mathbf{D}^{C_k}|}{|\mathbf{D}|} \tag{C.3}$$

La entropía del árbol de decisión que clasifica a los objetos de **D** será

$$H(\mathbf{D}) = -\sum_k P_k \log_2 P_k \tag{C.4}$$

con $H(0) = 0$. Si P_k es cero, forzamos a que el producto $P_k \log_2 P_k$ sea cero. En algunos textos, a la entropía H, se le llama también "medida de información", y se denota por $I(\mathbf{D})$ [134].

Si se selecciona el atributo T, las entropías separadas en cada uno de los nodos hijo serán las $H(D_i)$

$$H(D_i) = \sum_j P_{j|i} \log_2 P_{j|i} \tag{C.5}$$

y la correspondiente a todo el árbol será

$$H(\mathbf{D}/T) = \sum_i P(T = T_i) H(D_i) \tag{C.6}$$

$P(T = T_i)$ es la probabilidad de que un objeto de **D** tome el valor T_i para el atributo T. Al aproximarla mediante la frecuencia relativa queda

$$P(T = T_i) = \frac{|D_i|}{\sum |D_i|} = \frac{|D_i|}{|\mathbf{D}|} \tag{C.7}$$

Nótese que $|D_i|$ es la suma de todos los ejemplos de **D** que en la propiedad T tienen el valor T_i. Para evitar confusiones, se suele representar $P(T = T_i)$ como P_{ij}, y el atributo en uso como A_i. De este modo, la probabilidad de que objetos del subconjunto D_i pertenezcan a la clase j será

$$P_{j|i} = P(A_i = A_{ij}) = \frac{|D_{ji}|}{\sum_j |D_{ji}|} = \frac{|D_{ji}|}{|\mathbf{D}|}$$

con $D_{ji} \subset D_i$ consistente en el subconjunto de elementos que, teniendo atributo T_i pertenecen a C_j. El criterio que sigue ID3 para seleccionar un atributo se basa en que el

subárbol de decisión más sencillo será aquél que ofrezca una menor entropía. La entropía de un sistema, en Teoría de la Información, mide el grado de desinformación que de él se tiene. Cuanto menor sea este grado, más información se tendrá y más sencillo (en número de nodos) resultará el resto del árbol.

El método a seguir consiste en calcular $H(\mathbf{D}/T)$ para todos los atributos que no hayan sido tratados en nodos ascendientes respecto del actual ($T = A$, $T = B$, ...) y seleccionar el atributo que minimice dicho valor. (Hay que maximizar $H(\mathbf{D}) - H(\mathbf{D}/T)$. Como $H(\mathbf{D})$ no depende del atributo elegido, esto es equivalente a minimizar $H(\mathbf{D}/T)$).

A la resta $H(\mathbf{D}) - H(\mathbf{D}/T)$ se le llama *ganancia de información*, G.

BIBLIOTECA VIRTUAL

C.2 ALGORITMO ID3 BORROSO

El algoritmo ID3 se aplica a un conjunto de datos para generar un árbol de decisión para clasificación. ID3 borroso, introducido por Umano *et al* [134], es una extensión para ser aplicado sobre un conjunto borroso de datos (varios datos con un grado de pertenencia) y genera un árbol de decisión borroso, usando los conjuntos borrosos definidos *a priori* por el usuario para todos los atributos.

Estudios posteriores (como [173]) van encaminados a que sea el algoritmo, basándose en la forma de los datos, el que defina la forma y el número de los conjuntos borrosos. Sin embargo, debido a las características propias del problema que nos atañe se ha considerado más oportuno seguir el método original de Umano, para poder tener un control mejor sobre la forma y el solapamiento de los conjuntos borrosos que representan las características.

Un árbol de decisión borroso consiste en una serie de nodos que comprueban atributos; *nodos*, en los que se ramifica el árbol al comprobar un determinado valor, y *hojas* en las que se decide a qué clase de salida corresponde el patrón analizado.

Este algoritmo es muy similar a ID3, pero en este caso la *ganancia de información* se calcula en función de la pertenencia, en lugar de usar la probabilidad de los datos. Para el algoritmo se asumirá que:

- Se dispone de un conjunto de datos de entrenamiento $\mathbf{D}$.
- El número de clases de salida es n. Representamos por $\mathbf{C}$ el conjunto de clases de salida: $\mathbf{C} = \{C_1, C_2, \cdots, C_n\}$
- Cada dato tiene l atributos, que representamos por $A_1, A_2, \cdots, A_l$
- Cada atributo A_i está definido por una variable lingüística, formada por $m(A_i)$ conjuntos borrosos (el valor de $m(A_i)$ varía para cada atributo): $F_{i1}, F_{i2}, \cdots, F_{im(A_i)}$.

- Se denota por $\mathbf{D}^{C_k}$ al subconjunto borroso de $\mathbf{D}$ cuya clase es C_k.
- Sea $|\mathbf{D}|$ la suma de los valores de pertenencia de los datos de $\mathbf{D}$.

El algoritmo para generar el árbol de decisión es el siguiente:

1. Se genera el nodo raíz con todos los datos (conjunto $\mathbf{D}$), esto es, se crea un conjunto borroso de todos los datos con valor de pertenencia 1.
2. Si un nodo t con un conjunto de datos $\mathbf{D}$ es una hoja si satisface las siguientes condiciones:
   (a) La proporción de un conjunto de datos de la clase C_k es mayor o igual que un umbral θ_r, esto es
   $$\frac{|\mathbf{D}^{C_k}|}{|\mathbf{D}|} \geq \theta_r$$
   (b) El número de datos en el conjunto es menor que un umbral θ_n, esto es
   $$|\mathbf{D}| \leq \theta_n$$
   (c) No hay más atributos para seguir clasificando
3. Si no satisface las condiciones anteriores, no es una *hoja*, por lo que hay que generar un nodo de test de la siguiente forma:
   (a) Para cada atributo A_i $(i = 1, 2, \cdots, l)$, se calcula la *ganancia de información* $\mathbf{G}(A_i, \mathbf{D})$ y se selecciona el atributo A_{max} que maximice dicha ganancia:
   $$A_{max} = \arg\max\{\mathbf{G}(A_i, \mathbf{D})\}$$
   (b) Se divide $\mathbf{D}$ en $m(A_{max})$ subconjuntos borrosos $D_1, D_2, \cdots, D_{m(A_{max})}$, de acuerdo con A_{max}. El valor de pertenencia de los datos en D_j es el producto del valor de pertenencia en $\mathbf{D}$ y el valor de pertenencia del conjunto $F_{max,j}$ de A_{max} en $\mathbf{D}$.
   (c) Se generan nuevos nodos $t_1, t_2, \cdots, t_{m(A_{max})}$ para los subconjuntos borrosos $D_1, D_2, \cdots, D_{m(A_{max})}$, y se etiquetan con $F_{max,j}$ las líneas que unen los nodos t_j con el nodo t, tal como se muestra en la figura C.2.
   (d) Se reemplaza $\mathbf{D}$ por D_j $(j = 1, 2, \cdots, m)$ y se repite recursivamente desde el punto 2.

La ganancia de información $\mathbf{G}(A_i, \mathbf{D})$ para el atributo A_i y para un conjunto de datos $\mathbf{D}$ se define

$$\mathbf{G}(A_i, \mathbf{D}) = \mathbf{I}(\mathbf{D}) - \mathbf{E}(A_i, \mathbf{D}) \tag{C.8}$$

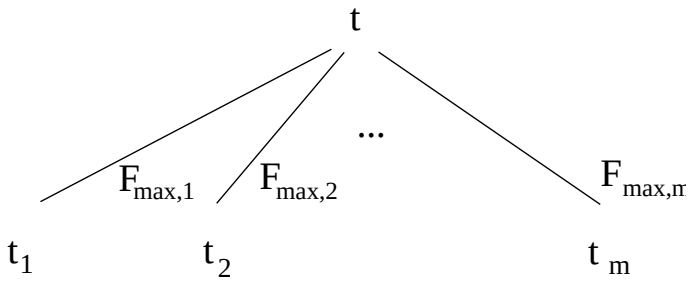


Figura C.2 Etiquetado de uniones

donde

$$\mathbf{I}(\mathbf{D}) = -\sum_{k=1}^{n}(p_k \cdot \log_2 p_k) \tag{C.9}$$

$$\mathbf{E}(A_i, \mathbf{D}) = \sum_{j=1}^{m}(p_{ij} \cdot \mathbf{I}(D_{F_{ij}})) \tag{C.10}$$

$$p_k = \frac{|\mathbf{D}^{C_k}|}{|\mathbf{D}|} \tag{C.11}$$

$$p_{ij} = \frac{|D_{F_{ij}}|}{\sum_{j=1}^{m}|D_{F_{ij}}|} \tag{C.12}$$

Para asignar el nombre de la clase a un nodo *hoja*, se proponen tres métodos:

1. El nodo se asigna a la clase que tenga el mayor valor de pertenencia.
2. Si no se cumple la propiedad de ser mayor que un umbral, el nodo queda vacío.
3. Se asignan todas las clases con sus correspondientes valores de pertenencia, esto es, todos los datos son tomados en cuenta.

C.3 APLICACIÓN DEL MÉTODO FITM AL CÁLCULO DE ÁRBOLES DE DECISIÓN

Los árboles borrosos ID3 que han sido explicados anteriormente pueden ser implementados de manera sencilla usando matrices de transición. De hecho, el uso de características no numéricas en muchas aplicaciones de este tipo las hace idóneas para ser implementadas usando los conceptos de granulo borroso e inferencia matricial.

Como punto de partida hay que tener en cuenta lo siguiente:

- Las características de entrada al árbol de decisión tienen como atributos términos que pueden ser expresados en lenguaje natural. Se definen las características por

A_j, y los valores que puede tomar cada característica por F_{jl}. Cada valor llevará asociado un conjunto borroso, que por facilidad denotaremos también como F_{jl}.

Podemos definir, por ejemplo, la característica *color del pelo*, cuyos valores son *rubio*, *rojo*, *negro*. Por lo tanto A_1=*color del pelo*, F_{11}=*rubio*,F_{12}=*rojo*, F_{13}=*negro*.

- Se denota por **C** el conjunto de salida, que se corresponde con las clases C_i en las que se han de clasificar los patrones.

A modo de ejemplo, (que por facilidad de entendimiento se entrelazará con la parte más teórica) se considerará un árbol ya construido con las siguientes propiedades (tal como se muestra en la figura C.3):

1. Características: A_1=(Altura)={Baja, Media, Alta}, A_2=(Peso)={Ligero, Medio, Pesado} y A_3=(Pelo)={Claro,Oscuro}.
2. Dos clases de salida, C_1 y C_2.

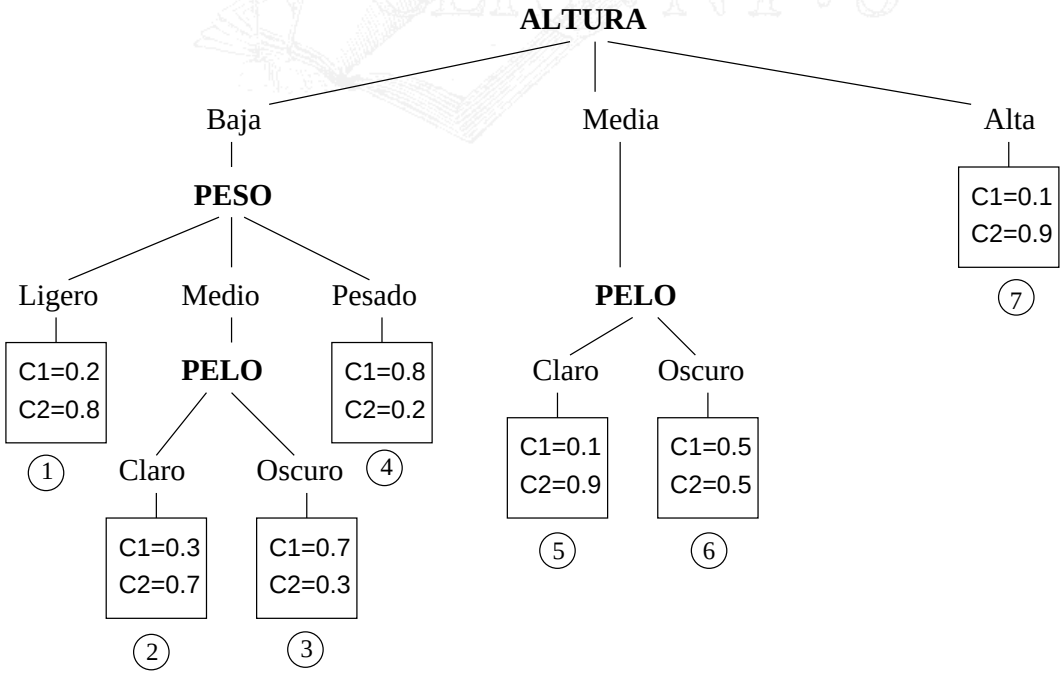


Figura C.3 Árbol borroso de ejemplo

Basándose en las matrices de transición, se plantean dos posibles algoritmos de inferencia.

C.3.1 Procesado en árbol

En este caso, se hace uso exclusivo de las matrices de activación (ecuación (4.2)). De este modo se mantiene el razonamiento *en árbol*, si perder la parte más "visual" de esta clase de algoritmos, y por otro lado, nos se permite una futura integración de valores no borrosos de entrada. El proceso es el siguiente:

1. Se genera la matriz de intersecciones $\mathbf{R_A}$ tal como se muestra en la ecuación (4.2). Esto va a dar lugar a tres matrices:

$$\mathbf{R_A}_{altura} = \begin{bmatrix} 1 & h_1 & 0 \\ h_1 & 1 & h_2 \\ 0 & h_2 & 1 \end{bmatrix} \quad \mathbf{R_A}_{peso} = \begin{bmatrix} 1 & w_1 & 0 \\ w_1 & 1 & w_2 \\ 0 & w_2 & 1 \end{bmatrix} \quad \mathbf{R_A}_{pelo} = \begin{bmatrix} 1 & p_1 \\ p_1 & 1 \end{bmatrix}$$

   siendo h_i, w_i y p_i los grados de solapamiento entre conjuntos entre conjuntos.

2. Se definen los vectores de características de entrada, siguiendo lo propuesto en el capítulo 4. En este caso, se tienen 3 vectores, uno por característica:

$$\mathbf{h} = \begin{bmatrix} \beta_1 \\ \beta_2 \\ \beta_3 \end{bmatrix} \quad \mathbf{w} = \begin{bmatrix} \gamma_1 \\ \gamma_2 \\ \gamma_3 \end{bmatrix} \quad \mathbf{p} = \begin{bmatrix} \alpha_1 \\ \alpha_2 \end{bmatrix}$$

3. Se calcula la activación de cada característica multiplicando las matrices $\mathbf{R_A}_i$ por los correspondientes vectores:

$$\begin{bmatrix} \beta'_1 \\ \beta'_2 \\ \beta'_3 \end{bmatrix} = \mathbf{A}_{altura} \times_{\cup\cap} \mathbf{h} \quad \begin{bmatrix} \gamma'_1 \\ \gamma'_2 \\ \gamma'_3 \end{bmatrix} = \mathbf{A}_{peso} \times_{\cup\cap} \mathbf{w} \quad \begin{bmatrix} \alpha'_1 \\ \alpha'_2 \end{bmatrix} = \mathbf{A}_{pelo} \times_{\cup\cap} \mathbf{p}$$

   donde $\times_{\cup\cap}$ puede ser sustituido por la multiplicación estándar en el caso de que se use operativa SAM. Los valores de salida deben ser normalizados por el máximo de ellos, ya que si no hay posibilidades de que a la salida del árbol no exista una coherencia entre los valores obtenidos para cada una de las características.

4. Se sustituyen los valores en el árbol de decisión, tal como aparece en la figura C.4. Para hallar los valores en cada hoja, sólo hay que realizar la intersección de los valores de cada rama en dirección al nodo raíz. En la figura se muestran los resultados (usando operativa SAM) para los nodos 2, 5 y 7.

La ventaja de este método es que sustituye toda la operativa borrosa por una simple multiplicación matricial. Cuando un árbol es especialmente denso, esto disminuye considerablemente el tiempo de ejecución.

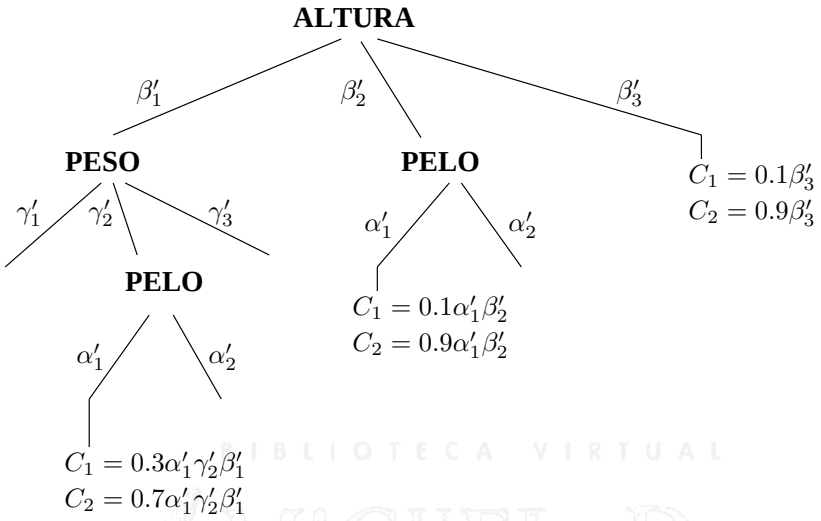


Figura C.4 Múltiples intersecciones en la definición de conjuntos borrosos.

C.3.2 Procesado matricial

En este caso se va a hacer uso de toda la teoría desarrollada en el capítulo 4. Esta clase de procesado va a hacer que se pierda la estructura de árbol a la hora de realizar la inferencia, por lo que se pierde en *comprensión* del funcionamiento, pero se gana en eficiencia y compacidad. Por lo tanto, el procesado matricial debe emplearse como método de inferencia una vez que el árbol ya está diseñado.

El algoritmo es el siguiente:

1. Se generan las matrices $\mathbf{R}_{\mathbf{A}i}$. En este ejemplo, al igual que en el caso anterior, surgen tres matrices, $\mathbf{R}_{\mathbf{A}altura}$, $\mathbf{R}_{\mathbf{A}peso}$ y $\mathbf{R}_{\mathbf{A}pelo}$.

2. Cálculo de las matrices de *composición*. En este caso, al tener 3 características, se definen las matrices $\mathbf{G}_{ij}$. Los subíndices i y j tendrá las dimensiones de las características elegidas como segunda y tercera. Si se opta para ello por *peso* y *pelo*, entonces $i = 1, 2, 3$, $j = 1, 2$. Las matrices $\mathbf{G}_{ij}$ pueden definirse de la siguiente manera para el caso SAM:

$$\mathbf{G}_{ij} = \mathbf{R}_{\mathbf{A}altura} \otimes_{\mathcal{K}} \left[\left(\mathbf{R}_{\mathbf{A}peso} \otimes_{\mathcal{K}} \left(\mathbf{R}_{\mathbf{A}pelo} \mathbf{E}_j \right) \right) \mathbf{E}_i \right] \quad i = 1, 2, 3 \;\; j = 1, 2 \tag{C.13}$$

   con lo que habrá 6 matrices de composición.

3. Construcción de las matrices $\mathbf{\Omega}_{ij}$.Para ello hay que comprender el sentido de las matrices $\mathbf{G}_{ij}$. Cada línea de esas matrices se corresponde con una combinación de las distintas características. En este caso tendrán 18 filas, ya que se cuenta con 3 características de tamaño 3, 3 y 2 respectivamente ($3 \times 3 \times 2 = 18$). Es equivalente a formular todos los posibles antecedentes de una base de reglas. Al seleccionar como característica principal *altura*, como secundaria *peso* y en tercer lugar *pelo*, el orden de las filas es equivalente a ordenar los antecedentes del siguiente modo:

Fila	Altura	Peso	Pelo
1	Baja	Ligero	Claro
2	Baja	Ligero	Oscuro
3	Baja	Medio	Claro
4	Baja	Medio	Oscuro
5	Baja	Pesado	Claro
6	Baja	Pesado	Oscuro
7	Media	Ligero	Claro
8	Media	Ligero	Oscuro
9	Media	Medio	Claro
10	Media	Medio	Oscuro
11	Media	Pesado	Claro
12	Media	Pesado	Oscuro
13	Alta	Ligero	Claro
14	Alta	Ligero	Oscuro
15	Alta	Medio	Claro
16	Alta	Medio	Oscuro
17	Alta	Pesado	Claro
18	Alta	Pesado	Oscuro

Sin embargo, si se escribe el árbol obtenido (Fig. C.3) en forma de reglas borrosas en conjunto de reglas que aparece es un conjunto no completo, en el que hay valores implícitos:

Regla	Altura	Peso	Pelo	C_1	C_2
1	Baja	Ligero		0.2	0.8
2	Baja	Medio	Claro	0.3	0.7
3	Baja	Medio	Oscuro	0.7	0.3
4	Baja	Pesado		0.8	0.2
5	Media		Claro	0.1	0.9
6	Media		Oscuro	0.5	0.5
7	Alta			0.1	0.9

En el apéndice C.4 se ha introducido un pequeño estudio de equivalencia entre reglas implícitas y completado de reglas, que justifica el desarrollo posterior.

Existen dos modos de generar las matrices $\mathbf{\Omega}_{ij}$, y ambos deben llevar a los mismos resultados:

(a) *Completado de reglas.* Con este método se van a mantener intactas las matrices $\mathbf{G}_{ij}$ y se genera una matriz $\mathbf{C}$ a la medida. Para ello, se va a generar un nuevo conjunto de reglas, incluyendo los valores ausentes. A las entradas *duplicadas* se les aplicará el mismo consecuente que a la entrada original. El nuevo conjunto de reglas ahora quedaría:

Regla	Altura	Peso	Pelo	C_1	C_2
1	Baja	Ligero	Claro	0.2	0.8
2	Baja	Ligero	Oscuro	0.2	0.8
3	Baja	Medio	Claro	0.3	0.7
4	Baja	Medio	Oscuro	0.7	0.3
5	Baja	Pesado	Claro	0.8	0.2
6	Baja	Pesado	Oscuro	0.8	0.2
7	Media	Ligero	Claro	0.1	0.9
8	Media	Ligero	Oscuro	0.5	0.5
9	Media	Medio	Claro	0.1	0.9
10	Media	Medio	Oscuro	0.5	0.5
11	Media	Pesado	Claro	0.1	0.9
12	Media	Pesado	Oscuro	0.5	0.5
13	Alta	Ligero	Claro	0.1	0.9
14	Alta	Ligero	Oscuro	0.1	0.9
15	Alta	Medio	Claro	0.1	0.9
16	Alta	Medio	Oscuro	0.1	0.9
17	Alta	Pesado	Claro	0.1	0.9
18	Alta	Pesado	Oscuro	0.1	0.9

Existe pues un desdoblamiento de valores. Se crea ahora la matriz $\mathbf{C}$ tal como se indica en el capítulo 4, a partir de los valores C_1 y C_2:

$$\mathbf{C} = \left[\begin{array}{cccccccccccccccccc} 0.2 & 0.2 & 0.3 & 0.7 & 0.8 & 0.8 & 0.1 & 0.5 & 0.1 & 0.5 & 0.1 & 0.5 & 0.1 & 0.1 & 0.1 & 0.1 & 0.1 & 0.1 \\ 0.8 & 0.8 & 0.7 & 0.3 & 0.2 & 0.2 & 0.9 & 0.5 & 0.9 & 0.5 & 0.9 & 0.5 & 0.9 & 0.9 & 0.9 & 0.9 & 0.9 & 0.9 \end{array} \right]^T$$

De aquí es inmediato la construcción de las matrices $\mathbf{\Omega}_{ij}$:

$$\mathbf{\Omega}_{ij} = \mathbf{C}^T \mathbf{G}_{ij} \tag{C.14}$$

(b) *Compresión de las matrices de composición.* En este caso, se crea la matriz $\mathbf{C}$ a partir de las reglas existentes y se modifican las matrices $\mathbf{G}_{ij}$ para adecuarse al tamaño real de la nueva base de reglas. La matriz $\mathbf{C}$ en este caso será:

$$\mathbf{C} = \left[\begin{array}{ccccccc} 0.2 & 0.3 & 0.7 & 0.8 & 0.1 & 0.5 & 0.1 \\ 0.8 & 0.7 & 0.3 & 0.2 & 0.9 & 0.5 & 0.9 \end{array} \right]^T$$

Existe por lo tanto una discrepancia entre las matrices $\mathbf{G}_{ij}$ (que tienen 18 filas) y la matriz $\mathbf{C}$ (con sólo 7 filas). En este caso, en vez de "desdoblar" las filas de $\mathbf{C}$, se van a combinar las filas de $\mathbf{G}_{ij}$. Un hueco en la base de reglas indica que da lo mismo ese valor, que no se tiene en cuenta. Por lo tanto,en el caso SAM para anular de esa forma un valor en las matrices de composición, se suman las filas correspondientes a la multiplicidad de ese valor.

Por ejemplo, la primera regla es de la forma

Regla	Altura	Peso	Pelo	C_1	C_2
1	Baja	Ligero		0.2	0.8

lo que quiere decir que no se tiene en cuenta la característica *pelo*. Para anularla, deben sumarse las filas correspondientes a los valores {Baja, Ligero, Claro} y {Baja, Ligero, Oscuro}, consiguiendo una línea que se podría leer como {Baja, Ligero, Claro+Oscuro}[1]. Esto supone sumar la primera y la segunda fila de la matriz $\mathbf{G}_{ij}$ original. Razonando del mismo modo, se obtiene una nueva matriz $\mathbf{G}'_{ij}$ que se define

$$\mathbf{G}'_{ij} = \begin{bmatrix} \mathbf{G}_{ij}(1) + \mathbf{G}_{ij}(2) \\ \mathbf{G}_{ij}(3) \\ \mathbf{G}_{ij}(4) \\ \mathbf{G}_{ij}(5) + \mathbf{G}_{ij}(6) \\ \mathbf{G}_{ij}(7) + \mathbf{G}_{ij}(9) + \mathbf{G}_{ij}(11) \\ \mathbf{G}_{ij}(8) + \mathbf{G}_{ij}(10) + \mathbf{G}_{ij}(12) \\ \mathbf{G}_{ij}(13) + \mathbf{G}_{ij}(14) + \mathbf{G}_{ij}(15) + \mathbf{G}_{ij}(16) + \mathbf{G}_{ij}(17) + \mathbf{G}_{ij}(18) \end{bmatrix}$$

siendo $\mathbf{G}_{ij}(k)$ la fila k-ésima de la matriz $\mathbf{G}_{ij}$. De aquí es inmediato la construcción de las matrices $\mathbf{\Omega}_{ij}$ (que tendrán las mismas dimensiones que las calculadas con el método anterior):

$$\mathbf{\Omega}_{ij} = \mathbf{C}^T \mathbf{G}'_{ij} \tag{C.15}$$

4. Para el cálculo de los valores de salida basta aplicar el método FITM sobre las matrices $\mathbf{\Omega}_{ij}$ ya calculadas. Para este ejemplo sería:

$$\begin{bmatrix} C_1 \\ C_2 \end{bmatrix} = \sum_{i=1}^{3} \sum_{j=1}^{2} \gamma_i \alpha_j \mathbf{\Omega}_{ij} \begin{bmatrix} \beta_1 \\ \beta_2 \\ \beta_3 \end{bmatrix} \tag{C.16}$$

C.4 COMPLETADO DE REGLAS

Al pasar un árbol de decisión a una base de reglas, tal como se ha hecho en este capítulo, aparecen reglas implícitas, es decir, reglas que no usan todas las características existentes en el sistema. Aunque en los árboles de decisión esto se hace más patente, el problema es fácilmente extrapolable a otros entornos borrosos.

Podría pensarse que el trabajar sin una característica es similar a completar la base de reglas con todos los valores de dicha característica. Veamos un ejemplo. Supóngase las características A_i y B_j (con $j = 1, 2, 3$). Supóngase que se tiene la regla

$$\text{Si } X \text{ es } A_i, \text{ entonces } Z \text{ es } C_k$$

Si se expande esta regla con todos los posibles valores de B_j se tiene

[1] Si se emplea una operativa basada en normas, esta línea debe entenderse como {Baja, Ligero, Claro∪Oscuro}

Si X es A_i e Y es B_1 entonces Z es C_k
Si X es A_i e Y es B_2 entonces Z es C_k
Si X es A_i e Y es B_3 entonces Z es C_k

La salida de cualquiera de los dos conjuntos de reglas es la misma si se usa una operativa basada en *max-min*. Sin embargo, no va a ser así en caso de usar el método SAM, tal y como se va a ver en el siguiente ejemplo. Se tiene la siguiente base incompleta de reglas:

A_i	B_j	C_k
A_1	B_1	C_1
A_1	B_2	C_2
A_2		C_3

BIBLIOTECA VIRTUAL MIGUEL DE CERVANTES

Esta base completada da lugar a la siguiente tabla:

A_i	B_j	C_k
A_1	B_1	C_1
A_1	B_2	C_2
A_2	B_1	C_3
A_2	B_2	C_3

Supóngase que la entrada al sistema es $\{A_1,B_1\}$. La inferencia usando SAM se define como

$$F(A,B) = \frac{\sum \omega_j A_j(A_1) B_{r(j)}(B_1) C_{p(j)}}{\sum \omega_j A_j(A_1) B_{r(j)}(B_1)} \tag{C.17}$$

Si se aplica sobre los dos casos (base incompleta y base completa), y teniendo en cuenta que

$$\begin{cases} A_1 \circ A_1 = 1 \\ A_1 \circ A_2 = a \\ B_1 \circ B_1 = 1 \\ B_1 \circ B_2 = b \end{cases}$$

los conjuntos de salida serán

$$F_1(A_1,B_1) = \frac{C_1 + bC_2 + aC_3}{1+b+a} \tag{C.18}$$

$$F_2(A_1,B_1) = \frac{C_1 + bC_2 + a(1+b)C_3}{1+b+a+ab} \tag{C.19}$$

Se puede ver claramente que en este caso la salida no es equivalente. Dado que no se está usando *max-min*, al combinar todos los conjuntos de una característica no

se obtendrá un valor unidad, sino mayor. Para paliar este efecto, que hace que en el primer caso (reglas implícitas) indirectamente tenga menos peso la regla que toma menos características, se debería ponderar dicha regla por la suma-producto de todos sus conjuntos. En este caso habría que ponderar la regla por $\omega = (1+b)$. Haciendo esto, se tendría:

$$F_1(A_1, B_1) = \frac{C_1 + bC_2 + a(1+b)C_3}{1+b+a(1+b)} \tag{C.20}$$

que coincide exactamente con el valor de F_2, es decir del completado de reglas.

Un trabajo interesante respecto a la influencia de los pesos en un sistema de clasificación se puede encontrar en [52].

BIBLIOTECA VIRTUAL MIGUEL DE CERVANTES

BIBLIOTECA VIRTUAL
MIGUEL DE CERVANTES

D

MHT BORROSO

D.1 REVISIÓN DE LOS SISTEMAS BORROSOS REALIMENTADOS

Una clase particular de los sistemas borrosos son aquéllos en los que una salida es realimentada al sistema convirtiéndose en entrada. Estos sistemas se conocen como Sistemas Borrosos Realimentados (se usará la abreviatura FFS del inglés *fuzzy feedback systems*).

Los FFSs han sido usados tradicionalmente como controladores, donde han demostrado su efectividad en una multitud de aplicaciones. La estabilidad de dichos controladores se estudia generalmente por medio de técnicas de análisis no lineal, como los métodos de Lyapunov [44, 72, 98], que dan un valor de confianza de la viabilidad del sistema, pero no caracterizan su comportamiento recursivo. De este modo, el diseño de estos controladores ha sido realizado tradicionalmente *ad-hoc*, debido a la dificultad para desarrollar una base teórica que caracterice matemáticamente el comportamiento del sistema.

Han existido diversos intentos de formalizar rigurosamente estos FFSs. El primero se debe a Tong [129] en 1980. El FFS propuesto es el que se muestra en la figura D.1.

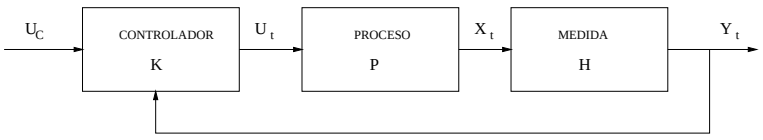


Figura D.1 Sistema en bucle cerrado (Tong)

El autor obtiene una descripción teórica de la respuesta en bucle cerrado en función del estado inicial, específicamente, $X_{t+1} = X_0 \circ S_t$, donde $\circ$ es una transformación definida por una composición *max-min*. La función de transferencia S_t depende de la entrada constante U_c y de los componentes del sistema (H, P y K), que son definidos como relaciones borrosas. Debido a la dificultad para resolver algunas ecuaciones relacionales

que aparecen para encontrar S_t, el problema tiene sólo solución bajo ciertas condiciones del controlador K.

A pesar de que el resultado no es directamente aplicable a otros sistemas en uso, es claramente una primera aproximación a una caracterización rigurosa de un FFS.

Un paso más avanzado en esta dirección es el dado por Chen y Tsao en [18]. Los autores afirman que *la principal causa del fallo de los anteriores intentos de caracterizar un FFS se debe al uso del operador max-min.* Este operador tiene el efecto colateral de achatar las funciones de pertenencia de los conjuntos borrosos. Cuando un FFS se describe de manera recursiva, la acumulación de este efecto hace que la monitorización de la evolución del sistema sea una tarea imposible. Para evitar este problema, los autores proponen el uso de conceptos de análisis de sistemas no lineales, específicamente, del *cell-to-cell mapping* desarrollado por Hsu [50], para analizar el comportamiento global de sistemas dinámicos no lineales.

El FFS utilizado por los autores está representado en la figura D.2. Este sistema es el que ha sido usado como paradigma de control borroso por otros muchos autores [39, 98]. La entrada es básicamente la diferencia (error) entre la salida de la planta y una señal de control. La relación entrada-salida se expresa recursivamente por $Y_{t+1} = f_1(Y_t)$, con f_1 una relación derivada del conjunto de reglas.

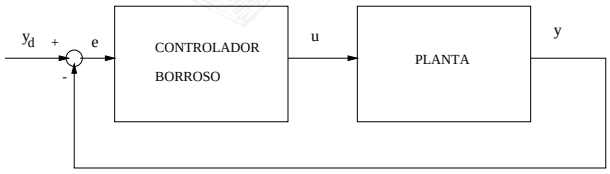


Figura D.2 Sistema Borroso de Control (Chen-Tsao)

A grandes rasgos, la clave en [18] consiste en mapear el sistema borroso en un sistema no borroso, de tal modo que su comportamiento pueda ser fácilmente entendido. El comportamiento global del sistema se extrae empleando el método de Hsu. El problema de la acumulación de *borrosidad* en cada iteración queda esquivado, ya que la parte borrosa se reduce a la transformación de un dominio al otro. Sin embargo, este método sólo puede dar una predicción aproximada del comportamiento del sistema; más aún, el método *cell-to-cell* no se puede aplicar a todos los sistemas dinámicos, de forma que en [18] sólo se dan criterios intuitivos para averiguar qué sistemas se ajustan a este marco de trabajo.

Kang [62] propone un método de diseño sistemático de controladores lingüísticos borrosos. Su sistema de control es el mostrado en la figura D.3.

Cuando la entrada r_k se suprime, el autor puede expresar la salida del sistema por medio de la ecuación recursiva $X_{k+1} = A^k \circ X_0$ (donde A^k no es necesariamente una potencia de una matriz, sino una composición de la forma $A^k = A \circ A \circ \ldots \circ A$, de forma similar a (4.71)), con A una matriz de transición que depende del conjunto de

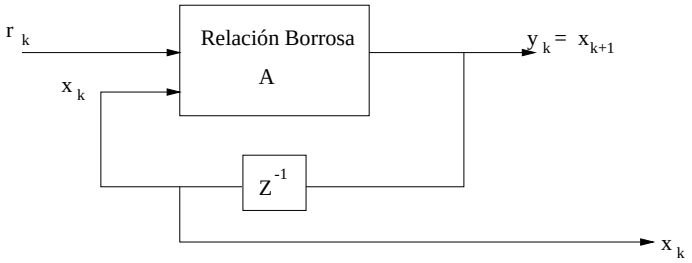


Figura D.3 Sistema borroso dinámico (Kang)

reglas y de las funciones de pertenencia de los conjuntos borrosos. La matriz tiene un tamaño $M \times M$, siendo M la dimensión del universo discreto sobre el que se definen los conjuntos. Se usa siempre el operador *max-min*. El autor establece que si existe un entero positivo n a partir del cual $A^{n+1} = A^n$ entonces X_k alcanza un estado estacionario que es $X_{n+1} = X_n$. El autor fuerza una relación K (no aparece dibujada en la figura D.3) entre las dos entradas r_k y X_k, para garantizar la estabilidad del sistema.

En [3] Aja, Alberola y Cybenko realizan un estudio de la convergencia de un FFS similar al usado por Kang.

D.2 DISCRIMINACIÓN DE HIPÓTESIS DEL ALGORITMO MHT BORROSO

Un sistema MHT borroso debe ser capaz de asociar una etiqueta de probabilidad mayor para la hipótesis real en un determinado instante de tiempo. En esta sección se demostrará que si la base de reglas está correctamente diseñada, el sistema propuesto en [7] y generalizado en la sección 5.1, discriminará correctamente la hipótesis más probable de las otras. El encargado de hacer esta discriminación es el tercer bloque de la figura 4.9.

P_m^k es la etiqueta de verosimilitud asociada a la hipótesis m-ésima que hace uso de los datos recibidos en el instante k, y H_m^{k-1} es la verosimilitud acumulada hasta el instante anterior de tiempo, asociada a esta hipótesis. H_m^k es la historia de probabilidades actualizada.

La variable *verosimilitud* se modela como una variable lingüística llamada *Probabilidad*, definida por varios conjuntos borrosos (*probable, muy probable, improbable*, etc.) [66] con funciones de pertenencia PTS, tal como se muestra en la figura D.4.

P_m^k es por lo tanto un conjunto borroso con función de pertenencia $\mu_{P_m^k}(x)$ y H_m^{k-1} es un segundo conjunto borroso con función de pertenencia $\mu_{H_m^{k-1}}(x)$. Por simplicidad, nos referiremos al conjunto y a la función de pertenencia de la misma manera, usando la notación del conjunto borroso, al igual que se hizo en el capítulo 4.

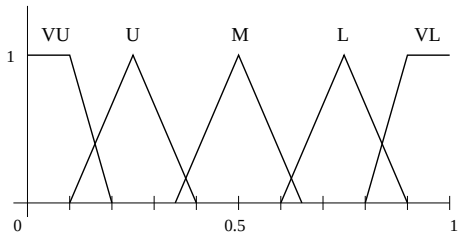


Figura D.4 Componentes de la Variable Lingüística *Probabilidad*

Como se ha visto en la sección 2.1.3, si A_i son conjuntos borroso normales y consistentes en $U \subset R$ con funciones de pertenencia PTS $A_i(x) = A_i(x; a_i, b_i, c_i, d_i)$ (i=1,2,... N), entonces existe un orden $\{i_1, i_2, ..., i_N\}$ en $\{1, 2, ..., N\}$ tal que

$$A_{i_1} < A_{i_2} < ... < A_{i_N} \tag{D.1}$$

En nuestro caso:

$$A_0 = VU < A_1 < ... < A_M = VL$$

El algoritmo MHT borroso debe garantizar que las probabilidades acumuladas de dos hipótesis deben poder ordenarse $H_1^k > H_2^k$ dado que en un instante de tiempo k la hipótesis Θ_1^k puede ser más probable que la hipótesis Θ_2^k. Para demostrar que esto es realmente así, se considerará un escenario en el peor caso posible (ver figura D.5), en el que dos hipótesis sólo difieren en la probabilidad de una asignación:

1. En un instante de tiempo t el sistema tiene una hipótesis en memoria, con alguna etiqueta de probabilidad. Se sume, sin pérdida de generalidad, que la etiqueta de probabilidad es máxima, esto es $H^{t-1} = A_M$.
2. En ese instante, una nueva observación llega. Se asume que es posible asociar la observación a dos pistas diferentes, dando lugar a dos posibles hipótesis:
   (a) Hipótesis Θ_1^t, de donde se calcula la etiqueta de probabilidad $P_1^t = A_M$ (a partir de los dos primeros bloques de la figura 4.9), con A_M el máximo conjunto borroso de la variable lingüística (en nuestro caso *Muy Probable*).
   (b) Hipótesis Θ_2^t, de la que se calcula la etiqueta de probabilidad $P_2^t = A_K$, con $A_k \neq A_M$.
3. Las observaciones subsiguientes darán lugar en ambas hipótesis al máximo valor de probabilidad, $P_1^{t+i} = P_2^{t+i} = A_M$ con $i = \{1, 2, \ldots\}$.

En la figura D.5 se describe gráficamente el planteamiento de este problema para el caso de seguimiento de vehículos. En la figura D.5-a, se asocia una pista a un vehículo

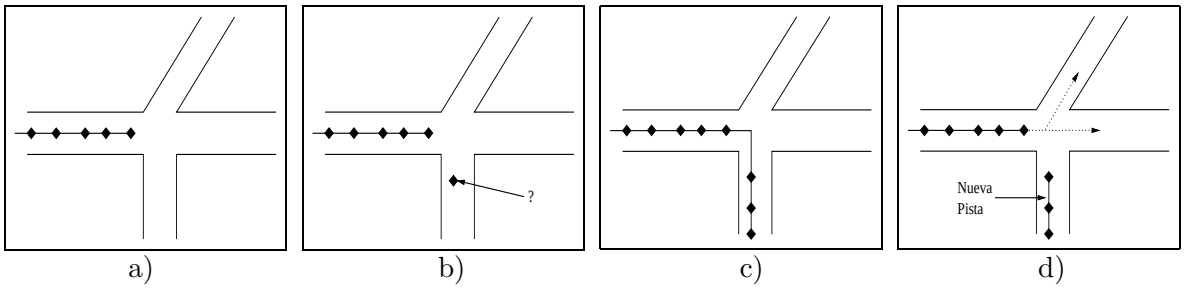


Figura D.5 Ilustración de una situación ambigua. a) Existe una pista asociada a un vehículo. b) Llega una nueva observación del mismo vehículo c) Hipótesis 1 (correcta): La pista se alarga con la nueva observación. d) Hipótesis 2 (errónea): Se crea una nueva pista. Se asume que hay presentes dos vehículos en el escenario.

BIBLIOTECA VIRTUAL

existente (H^{t-1}); por ejemplo, observaciones temporales de un camión. Suponemos que llega una nueva observación del mismo vehículo (figura D.5-b); sin embargo, debido a la ambigüedad en su descripción y al giro a la derecha del vehículo se pueden considerar dos posibles hipótesis:

- Hipótesis 1 (Θ_1^t): se considera que la nueva observación viene del mismo vehículo que ha girado a la derecha (figura D.5-c). La pista se alarga con esta nueva observación.
- Hipótesis 2 (Θ_2^t): se considera que la nueva observación es la primera de un segundo vehículo que acaba de aparecer en el escenario. Se asume que en este instante de tiempo no se da ninguna observación sobre el camión (figura D.5-d).

Teorema 1
El algoritmo MHT es capaz de discriminar que

$$H_1^k \geq H_2^k, \forall k \geq t$$

Demostración:
La demostración tendrá dos pasos. El primer paso muestra que se puede manifestar que $H_1^t > H_2^t$. El segundo paso es un método inductivo derivado del primero, que permite ordenar las hipótesis en un instante arbitrario de tiempo futuro.

Primer paso: análisis en el instante de tiempo t

Se considera un razonamiento con un motor de inferencias SAM y el método de activación de reglas *sum-product*. Se eligen estos métodos por ser ambos lineales, lo que facilitará el estudio formal; por otro lado, se siguen las conclusiones presentadas en [7], donde los resultados usando operadores de Kosko eran similares a los del caso Bayesiano, en oposición a la aproximación max-min, el cual obtenía los resultados más pobres.

Para reglas de la siguiente forma:

$$R_j: \quad \text{Si } H_{l(m)}^{k-1} \text{ es } a_j \quad \text{y } P_m^k \text{ es } b_j \quad \text{entonces } \mathcal{Y} \text{ es } C_j$$

puede escribirse para el instante de tiempo k

$$H_m^k = \sum_j C_j a_j(H_{l(m)}^{k-1}) b_j(P_m^k) \tag{D.2}$$

que puede ser expresado como

$$H_m^k = \frac{1}{K_1} \sum_j C_j \left(\sum a_j H_{l(m)}^{k-1}\right) \left(\sum b_j P_m^k\right) \tag{D.3}$$

BIBLIOTECA VIRTUAL

De acuerdo con el planteamiento del problema, la probabilidad en el instante de tiempo $t-1$, $H_1^{t-1} = A_M$, de acuerdo con (D.3) se puede escribir

$$H_1^t = \frac{1}{K_1} \sum_j C_j \left(\sum a_j A_M\right) \left(\sum b_j A_M\right) \tag{D.4}$$

y

$$H_2^t = \frac{1}{K_1'} \sum_j C_j \left(\sum a_j A_M\right) \left(\sum b_j A_K\right) \tag{D.5}$$

Para el siguiente paso se necesita lo siguiente:

Proposición 2
Si A_i son conjuntos borrosos normales y consistentes que dan lugar a una partición completa de $U \subset R$, con funciones de pertenencia PTS $A_i(x) = A_i(x; a_i, b_i, c_i, d_i)(i = 1, 2, ..., N)$, cada función de pertenencia, excepto la mayor y la menor, intersecará a un solo conjunto en cada extremo.

Observación 1
Cada conjunto borroso interseca con otros dos conjuntos, uno a cada lado del máximo del conjunto, excepto los conjuntos A_1 y A_N, que sólo intersecan con un conjunto borroso.

Demostración
Dado que se manejan conjuntos borrosos normales y consistentes, cada conjunto intersecará sólo a un conjunto en cada extremo, ya que de otro modo, el conjunto tendría un valor no nulo en el subconjunto normal del otro conjunto, por lo que no se mantendría la propiedad de consistencia. Dado que la partición es completa, dos conjuntos consecutivos deben intersecarse.

Para la demostración asumimos sin pérdida de generalidad que $A_K = A_{M-1}$. Si la propiedad se cumple para $A_K = A_{M-1}$ necesariamente se cumplirá para $A_K < A_{M-1}$.

Para Θ_1^t, dado que se cumple la proposición 2, se aplica la siguiente tabla de reglas activadas:

$a_j(x)$	$b_j(x)$	C_j	$R_j = a_j(A_M)b_j(A_M)c_j$
A_M	A_M	C_I	c_I
A_M	A_{M-1}	C_{II}	$\delta_1 c_{II}$
A_{M-1}	A_M	C_{III}	$\delta_1 c_{III}$
A_{M-1}	A_{M-1}	C_{IV}	$\delta_1^2 c_{IV}$

con $c_I, c_{II}, ...$ los centroides de los conjuntos borrosos C_j. La proposición 2 nos permite escribir respecto a la composición:

$$A_i \circ A_j = \begin{cases} = 1 & if \quad i = j \\ = \delta_i & if \quad |i-j| = 1 \\ = 0 & if \quad |i-j| > 1 \end{cases} \tag{D.6}$$

En este caso, $A_M \circ A_{M-1} = \delta_1$ y $A_{M-1} \circ A_{M-2} = \delta_2$, donde ambas δ_i dependen del método real de activación de reglas (max-min, sum-product, etc.).

De manera similar para Θ_2^t

$a_j(x)$	$b_j(x)$	C_j	$a_j(A_M)b_j(A_{M-1})c_j$
A_M	A_M	C_I	$\delta_1 c_I$
A_M	A_{M-1}	C_{II}	c_{II}
A_{M-1}	A_M	C_{III}	$\delta_1^2 c_{III}$
A_{M-1}	A_{M-1}	C_{IV}	$\delta_1 c_{IV}$
A_M	A_{M-2}	C_V	$\delta_2 c_V$
A_{M-1}	A_{M-2}	C_{VI}	$\delta_1 \delta_2 c_{VI}$

El valor de la probabilidad lo obtenemos por el método de los centroides:

$$c(A', B') = \frac{\sum_j a_j(A')b_j(B')c_j}{\sum_j a_j(A')b_j(B')} \tag{D.7}$$

lo que en el presente caso resulta

$$c_1 = \frac{c_I + \delta_1 c_{II} + \delta_1 c_{III} + \delta_1^2 c_{IV}}{(1+\delta_1)^2} \tag{D.8}$$

para H_1^t y

$$c_2 = \frac{c_{II} + \delta_1 c_I + \delta_1 c_{IV} + \delta_2 c_V + \delta_1 \delta_2 c_{VI} + \delta_1^2 c_{III}}{(1+\delta_1)(1+\delta_1+\delta_2)} \tag{D.9}$$

para H_2^t.

Es necesario establecer unas restricciones suaves sobre la base de reglas para garantizar una correcta discriminación. Dado que el sistema es un comparador de probabilidades, entradas mayores darán lugar a salidas mayores. Específicamente, la salida cuando las entradas son A_M y A_M debería ser mayor que la salida para las entradas A_{M-1} y A_{M-1}. Cuando las entradas son A_M y A_{M-1} sólo se puede afirmar que la

salida debe ser menor o igual que la salida cuando las entradas son A_M y A_M. Estas restricciones dan lugar a las siguientes relaciones: $c_I \geq c_{II}$, $c_I \geq c_{III}$, $c_I > c_{IV}$, $c_I \geq c_V$, $c_I > c_{VI}$, $c_{II} \geq c_{IV}$, $c_{II} \geq c_V$, $c_{II} \geq c_{VI}$, $c_{III} \geq c_{IV}$, $c_{III} \geq c_V$, $c_{III} \geq c_{VI}$, $c_{IV} \geq c_{VI}$, $c_V \geq c_{VI}$.

Se pueden comparar los centroides escribiendo

$$c_1 = \frac{\left(c_I + \delta_1 c_{II} + \delta_1 c_{III} + \delta_1^2 c_{IV}\right)(1 + \delta_1 + \delta_2)}{(1 + \delta_1)^2(1 + \delta_1 + \delta_2)} = \frac{N_1}{(1 + \delta_1)^2(1 + \delta_1 + \delta_2)} \tag{D.10}$$

$$c_2 = \frac{\left(c_{II} + \delta_1 c_I + \delta_1 c_{IV} + \delta_2 c_V + \delta_1 \delta_2 c_{VI} + \delta_1^2 c_{III}\right)(1 + \delta_1)}{(1 + \delta_1)^2(1 + \delta_1 + \delta_2)} = \frac{N_2}{(1 + \delta_1)^2(1 + \delta_1 + \delta_2)} \tag{D.11}$$

Si se calcula la diferencia de numeradores:

$$\begin{aligned} N_1 - N_2 \;=\; & (c_I - c_{II}) + \delta_1(c_{III} - c_{IV}) + \delta_2(c_I - c_V) + \delta_1^2(c_{II} - c_I) + \\ & + \delta_1\delta_2(c_{II} - c_V + c_{III} - c_{VI}) + \delta_1^3(c_{IV} - c_{III}) + \delta_1^2\delta_2(c_{IV} - c_{VI}) \end{aligned}$$

y haciendo uso de las relaciones mencionadas

$$\begin{aligned} (c_I - c_{II}) \geq 0 \quad ; \quad & \delta_1(c_{III} - c_{IV}) \geq 0 \\ \delta_2(c_I - c_V) \geq 0 \quad ; \quad & \delta_1^2(c_{II} - c_I) \leq 0 \\ \delta_1\delta_2(c_{II} - c_V) \geq 0 \quad ; \quad & \delta_1\delta_2(c_{III} - c_{VI}) \geq 0 \\ \delta_1^3(c_{IV} - c_{III}) \leq 0 \quad ; \quad & \delta_1^2\delta_2(c_{IV} - c_{VI}) \geq 0 \end{aligned} \tag{D.12}$$

se observa que todos los elementos son positivos excepto $\delta_1^2(c_{II} - c_I)$ y $\delta_1^3(c_{IV} - c_{III})$. Sin embargo la siguiente desigualdad se cumple:

$$\begin{aligned} (c_I - c_{II}) \;&\geq\; \delta_1^2(c_I - c_{II}) \\ \delta_1(c_{III} - c_{IV}) \;&\geq\; \delta_1^3(c_{III} - c_{IV}) \end{aligned} \tag{D.13}$$

dado que $\delta_1 \leq 1$ y $\delta_2 \leq 1$. Por lo tanto, se puede escribir:

$$\begin{aligned} (c_I - c_{II}) - \delta_1^2(c_I - c_{II}) \;&\geq\; 0 \\ \delta_1(c_{III} - c_{IV}) - \delta_1^3(c_{III} - c_{IV}) \;&\geq\; 0 \end{aligned} \tag{D.14}$$

y consecuentemente $N_1 - N_2 \geq 0$ o, de manera equivalente, $c_1 \geq c_2$.

La igualdad se obtiene cuando $c_I = c_{II} = c_V$ y $c_{III} = c_{IV} = c_{VI}$, lo que es evitable mediante una buena elección de la base de reglas. Si se asume un buen diseño de dicha base de reglas se puede afirmar que

$$H_2^{t+1} \leq H_1^{t+1} \tag{D.15}$$

q.e.d.

Segundo paso: Probabilidades en un instante arbitrario de tiempo futuro

El problema se plantea ahora de la siguiente forma:

- Las dos hipótesis Θ_1^t y Θ_2^t, tienen probabilidades acumuladas H_1^t y H_2^t respectivamente, y se ha probado que $H_2^t < H_1^t$.
- Suponemos ahora que $P_1^{t+i} = P_2^{t+i} = A_M, \forall i > 0$, esto es, las observaciones siguientes dan lugar a la máxima salida en el segundo bloque de la figura 4.9 para ambas hipótesis.

BIBLIOTECA VIRTUAL

Se pretende demostrar que $H_2^k < H_1^k$ para $\forall k = t + t_1 > t$, lo que resulta sencillo aplicando inducción sobre el resultado del paso anterior.

Se definen los siguientes parámetros:

$$\begin{aligned} P_{a_j} &= \sum a_j A_M \\ P_{b_j} &= \sum b_j A_M \\ M_{a_j} &= \sum a_j A_{M-1} \end{aligned} \tag{D.16}$$

las historias acumuladas en el instante de tiempo $t + t_1$ son

$$H_1^{t+t_1} = \frac{1}{K_{t_1}} \sum_j C_j P_{b_j} \left(... \sum_j C_j P_{b_j} \left(\sum a_j \left(\sum_j C_j P_{b_j} \left(\sum a_j \left(\sum_j C_j P_{b_j} P_{a_j} \right) \right) \right) \right) ... \right) \tag{D.17}$$

$$H_2^{t+t_1} = \frac{1}{K_{t_1}} \sum_j C_j P_{b_j} \left(... \sum_j C_j P_{b_j} \left(\sum a_j \left(\sum_j C_j P_{b_j} \left(\sum a_j \left(\sum_j C_j P_{b_j} M_{a_j} \right) \right) \right) \right) ... \right) \tag{D.18}$$

lo que puede ser escrito recursivamente como

$$\begin{aligned} H_1^{t+t_1} &= \frac{1}{K_{t_1}} \sum_j C_j P_{b_j} \left(\sum a_j H_1^{t+t_1-1} \right) \\ H_2^{t+t_1} &= \frac{1}{K_{t_1}} \sum_j C_j P_{b_j} \left(\sum a_j H_2^{t+t_1-1} \right) \end{aligned} \tag{D.19}$$

De estas dos ecuaciones es obvio que si se cumple $H_2^t \leq H_1^t$, entonces necesariamente se cumple que $H_2^{t+t_1} \leq H_1^{t+t_1}$, q.e.d.

D.3 MATRICES DEL SISTEMA MHT BORROSO

D.3.1 Matrices del bloque de compatibilidad lógica

$$\mathbf{R}_{\mathcal{V}} = \begin{pmatrix} 1 & 0.05 & 0 & 0 & 0 & 0 \\ 0.05 & 1 & 0.2 & 0.1 & 0 & 0 \\ 0 & 0.2 & 1 & 0.1 & 0.1 & 0 \\ 0 & 0.1 & 0.1 & 1 & 0.1 & 0 \\ 0 & 0 & 0 & 0.1 & 1 & 0.2 \\ 0 & 0 & 0 & 0.1 & 0.2 & 1 \end{pmatrix} \tag{D.20}$$

BIBLIOTECA VIRTUAL MIGUEL DE CERVANTES

$$\mathbf{C}_{\mathcal{V}} = \begin{pmatrix} 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix} \tag{D.21}$$

$$\mathbf{\Omega}_1^{\mathcal{V}} = \begin{pmatrix} 0 & 0 & 0.1000 & 1.1050 & 1.3600 & 1.2600 \\ 0 & 0.2000 & 1.0050 & 0.1500 & 0.0050 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0.1000 & 1.0125 & 0.2500 & 0.0050 & 0 & 0 \\ 1.0025 & 0.1000 & 0.0100 & 0 & 0 & 0 \end{pmatrix} \tag{D.22}$$

$$\mathbf{\Omega}_2^{\mathcal{V}} = \begin{pmatrix} 0 & 0 & 0.0050 & 0.1550 & 1.2650 & 1.2600 \\ 0.2000 & 0.0200 & 0.1500 & 1.0050 & 0.1400 & 0.2000 \\ 0 & 0 & 0 & 0.0200 & 0.2000 & 0.0400 \\ 1.0125 & 0.5000 & 1.0700 & 0.3000 & 0.0200 & 0 \\ 0.1000 & 1.0425 & 0.4000 & 0.0200 & 0 & 0 \end{pmatrix} \tag{D.23}$$

$$\mathbf{\Omega}_3^{\mathcal{V}} = \begin{pmatrix} 0.1000 & 0.0050 & 0 & 0.0200 & 0.2400 & 0.2400 \\ 1.0050 & 0.1500 & 0.0400 & 0.2000 & 0.2200 & 1.0000 \\ 0 & 0 & 0 & 0.1000 & 1.0200 & 0.3000 \\ 0.2500 & 1.0700 & 0.6000 & 1.0400 & 0.2000 & 0.0200 \\ 0.0100 & 0.4000 & 1.0500 & 0.2000 & 0.0100 & 0 \end{pmatrix} \tag{D.24}$$

$$\mathbf{\Omega}_4^{\mathcal{V}} = \begin{pmatrix} 1.1050 & 0.1550 & 0.0200 & 0 & 0 & 0 \\ 0.1500 & 1.0050 & 0.2000 & 0 & 0.0200 & 0.1000 \\ 0 & 0.0200 & 0.1000 & 0.0200 & 0.3000 & 1.0200 \\ 0.0050 & 0.3000 & 1.0400 & 0.4000 & 1.0400 & 0.3000 \\ 0 & 0.0200 & 0.2000 & 1.0200 & 0.2000 & 0.0200 \end{pmatrix} \tag{D.25}$$

$$\mathbf{\Omega}_5^{\mathcal{V}} = \begin{pmatrix} 1.3600 & 1.2650 & 0.2400 & 0 & 0 & 0 \\ 0.0050 & 0.1400 & 0.2200 & 0.0200 & 0 & 0 \\ 0 & 0.2000 & 1.0200 & 0.3000 & 0.0400 & 0.1000 \\ 0 & 0.0200 & 0.2000 & 1.0400 & 0.6000 & 1.0600 \\ 0 & 0 & 0.0100 & 0.2000 & 1.0500 & 0.4000 \end{pmatrix} \tag{D.26}$$

$$\mathbf{\Omega}_6^{\mathcal{V}} = \begin{pmatrix} 1.2600 & 1.2600 & 0.2400 & 0 & 0 & 0 \\ 0 & 0.2000 & 1.0000 & 0.1000 & 0 & 0 \\ 0 & 0.0400 & 0.3000 & 1.0200 & 0.1000 & 0 \\ 0 & 0 & 0.0200 & 0.3000 & 1.0600 & 0.4000 \\ 0 & 0 & 0 & 0.0200 & 0.4000 & 1.0400 \end{pmatrix} \tag{D.27}$$

D.3.2 Matrices del bloque de compatibilidad física y lógica

$$\mathbf{R}_{\mathcal{E}} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \tag{D.28}$$

$$\mathbf{C}_{\mathcal{E}} = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \end{pmatrix} \tag{D.29}$$

$$\mathbf{\Omega}_1^{\mathcal{E}} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 1.0000 & 0.1000 & 0 & 0 & 0 \\ 0.1000 & 1.0000 & 0.0100 & 0 & 0 \\ 0 & 0.0100 & 1.0000 & 0.0100 & 0 \\ 0 & 0 & 0.0100 & 1.1000 & 1.1000 \end{pmatrix} \tag{D.30}$$

$$\mathbf{\Omega}_2^{\mathcal{E}} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 1.0000 & 0.1000 & 0 & 0 & 0 \\ 0.1000 & 1.0100 & 1.0100 & 0.0100 & 0 \\ 0 & 0 & 0.0100 & 1.0000 & 0.1000 \\ 0 & 0 & 0 & 0.1000 & 1.0000 \end{pmatrix} \tag{D.31}$$

$$\mathbf{\Omega}_3^{\mathcal{E}} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 1.1000 & 1.1000 & 0.0100 & 0 & 0 \\ 0 & 0.0100 & 1.0000 & 0.0100 & 0 \\ 0 & 0 & 0.0100 & 1.1000 & 1.1000 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix} \tag{D.32}$$

$$\mathbf{\Omega}_4^{\mathcal{E}} = \begin{pmatrix} 1.1000 & 1.1000 & 0.0100 & 0 & 0 \\ 0 & 0.0100 & 1.0000 & 0.0100 & 0 \\ 0 & 0 & 0.0100 & 1.1000 & 1.1000 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix} \tag{D.33}$$

$$\mathbf{\Omega}_5^{\mathcal{E}} = \begin{pmatrix} 1.1000 & 1.1000 & 0.0100 & 0 & 0 \\ 0 & 0.0100 & 1.0100 & 1.1100 & 1.1000 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix} \tag{D.34}$$

$$\mathbf{\Omega}_6^{\mathcal{E}} = \begin{pmatrix} 1.1000 & 1.1100 & 1.0200 & 1.1100 & 1.1000 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix} \tag{D.35}$$

D.3.3 Matrices del bloque de actualización de probabilidades

$$\mathbf{R}_{\mathcal{P}} = \begin{pmatrix} 1 & 0.1 & 0 & 0 & 0 \\ 0.1 & 1 & 0.01 & 0 & 0 \\ 0 & 0.01 & 1 & 0.01 & 0 \\ 0 & 0 & 0.01 & 1 & 0.1 \\ 0 & 0 & 0 & 0.1 & 1 \end{pmatrix} \tag{D.36}$$

$$\mathbf{C}_{\mathcal{P}} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix} \tag{D.37}$$

$$\mathbf{\Omega}_1^{\mathcal{P}} = \begin{pmatrix} 1.2100 & 1.2100 & 0.0110 & 0 & 0 \\ 0 & 0.0110 & 1.1110 & 1.2210 & 1.2100 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix} \tag{D.38}$$

$$\mathbf{\Omega}_2^{\mathcal{P}} = \begin{pmatrix} 1.2100 & 1.2100 & 0.0110 & 0 & 0 \\ 0.0110 & 0.0220 & 1.1111 & 1.2210 & 1.2100 \\ 0 & 0.0001 & 0.0101 & 0.0101 & 0.0010 \\ 0 & 0 & 0 & 0.0010 & 0.0100 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix} \tag{D.39}$$

$$\mathbf{\Omega}_3^{\mathcal{P}} = \begin{pmatrix} 0.0110 & 0.0110 & 0.0001 & 0 & 0 \\ 1.1110 & 1.1111 & 0.0202 & 0.0111 & 0.0110 \\ 0 & 0.0101 & 1.0200 & 1.0101 & 0.1000 \\ 0 & 0 & 0.0001 & 0.1100 & 1.0010 \\ 0 & 0 & 0 & 0.0010 & 0.0100 \end{pmatrix} \tag{D.40}$$

$$\boldsymbol{\Omega}_4^{\mathcal{P}} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 1.2210 & 1.2210 & 0.0111 & 0 & 0 \\ 0 & 0.0101 & 1.0101 & 0.0201 & 0.0010 \\ 0 & 0.0010 & 0.1100 & 1.0020 & 0.1100 \\ 0 & 0 & 0.0010 & 0.2100 & 1.1100 \end{pmatrix} \tag{D.41}$$

$$\boldsymbol{\Omega}_5^{\mathcal{P}} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 1.2100 & 1.2100 & 0.0110 & 0 & 0 \\ 0 & 0.0010 & 0.1000 & 0.0010 & 0 \\ 0 & 0.0100 & 1.0010 & 0.1100 & 0.0100 \\ 0 & 0 & 0.0100 & 1.1100 & 1.2000 \end{pmatrix} \tag{D.42}$$

BIBLIOTECA VIRTUAL MIGUEL DE CERVANTES

E

FILTRO BORROSO DE KALMAN EN ENTORNOS DE SEGUIMIENTO ESTÉREO

BIBLIOTECA VIRTUAL MIGUEL DE CERVANTES

E.1 INTRODUCCIÓN

El seguimiento de blancos en un espacio tridimensional es una tarea poco sencilla, pero a la vez muy interesante, de la que se puede recoger información relevante acerca de la dinámica de objetos complejos. Ésta es la causa de que el problema haya sido tratado profusamente dentro del campo de visión artificial.

Uno de estos trabajos es [112], en é los autores realizan un estudio detallado de diversos esquemas de seguimiento basados en el filtro de Kalman, todos ellos aplicados al seguimiento de móviles en el espacio 3D mediante un sistema de visión estereoscópica. Los autores derivan expresiones cerradas para la matriz de covarianza del ruido de reconstrucción de medida en el espacio 3D asumiendo que las medidas se ven afectadas por ruido gaussiano. A partir de dicho conocimiento, es posible predecir el funcionamiento de diferentes esquemas de seguimiento en un escenario estéreo. En dicho trabajo se cita como posibilidad y línea de trabajo el desarrollo de filtros independientes del modelo, siguiendo la línea desarrollada en [16]. Se propone la creación de un sistema de reglas borrosas que fueran las encargadas de controlar dicho filtro, de modo que no resultase necesaria la estimación de ciertos parámetros *a priori*.

En [5] se plantea esta ampliación aplicada al filtro de Kalman en coordenadas principales, por ser el que ofrece una perfecta separación de variables en las tres coordenadas espaciales. Los resultados obtenidos pueden ser fácilmente extrapolables a otros esquemas de seguimiento, como los filtros de Kalman en coordenadas acopladas y desacopladas. En [5] se siguen dos líneas de trabajo para mejorar el filtro de Kalman espacial: primero, siguiendo la línea de otros autores, se crea una base de reglas que permita una variación borrosa de la matriz de covarianza de ruido de planta (maniobra), $\mathbf{Q}$, para que se adapte a las necesidades del filtro en cada instante. En segundo lugar, teniendo como referencia los esquemas de seguimiento de maniobras propuestos en [12], se desarrolla un control borroso que reinicie el filtro en caso de que exista un desacople grande.

La filosofía de diseño en que se basa este segundo controlador es la que aparece en la sección 4.4.2 y en [3]. La posibilidad de que el filtro de Kalman se desacople debido

a un cambio de trayectoria se modela como un proceso borroso que da lugar a un FFS con entradas borrosas, del que es posible conocer su comportamiento a priori.

E.2 ESQUEMA DE SEGUIMIENTO ADOPTADO

Se ha considerado un sistema de visión estereoscópica consistente en dos cámaras de vídeo CCD convencionales e idénticas, capaces de operar de forma síncrona. Se asume que los planos retinales (los planos sobre los que se proyectarán las imágenes) son paralelos y localizados ambos en el plano $z = f$, siendo f la distancia focal (Fig. E.1); se asume que ambas cámaras tienen la misma distancia focal y la misma orientación. El eje X contiene la *línea base* (la línea que une los focos c_1 y c_2 de las cámaras 1 y 2 respectivamente) y el eje Y es ortogonal al plano XZ con orientación hacia arriba. Los ejes coordenados de las cámaras (u_1, v_1) y (u_2, v_2) son tal y como se muestran en la Fig. E.1.

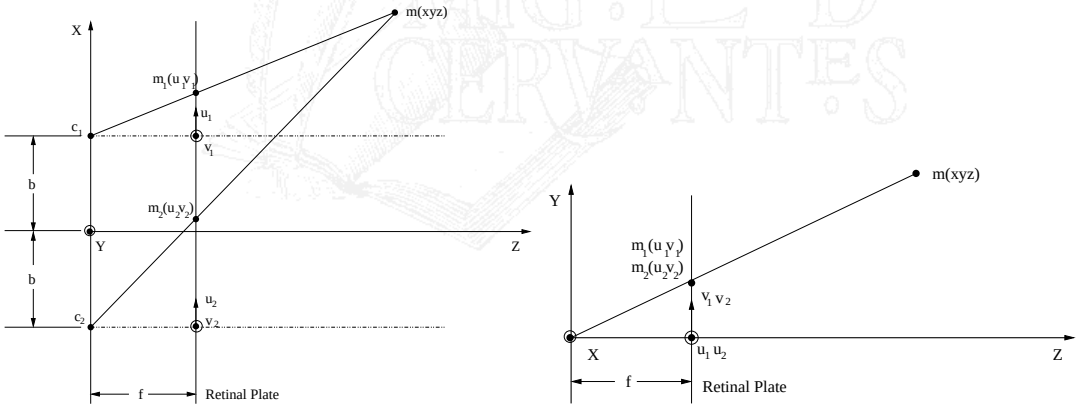


Figura E.1 Sistema de visión estereoscópica. a) Proyección sobre el plano xz. b) Proyección sobre el plano yz.

Las medidas se realizan sobre las dos cámaras y se asume que ambas se ven afectadas por ruido blanco gaussiano, de media cero y con varianza σ^2 en cada una de las coordenadas espaciales. Dado que el seguimiento se va a realizar en un espacio 3D, es necesario hacer una conversión de las medidas 2D al espacio 3D. Al no ser lineal esta transformación, el ruido en el espacio 3D no puede considerarse gaussiano. Sin embargo, en [112] se hace una aproximación lineal, describiendo las condiciones de validez de dicha aproximación. Bajo estas condiciones se obtiene una expresión cerrada para la matriz de covarianza de ruido de observación $\mathbf{R}(x, y, z)$ en función de la geometría del sistema, la posición del blanco y la varianza del ruido de observación en cada cámara.

El modelo de movimiento considerado es rectilíneo uniforme con velocidad constante. La incertidumbre propia del desconocimiento del verdadero movimiento del blanco se contempla en las ecuaciones de estado mediante la introducción de un ruido de planta que se identifica con una aceleración aleatoria del blanco, la cual se considerará estacionaria, gaussiana, de media nula, blanca e independiente para cada componente.

Particularizando las ecuaciones, el modelo discreto de estados resulta:

$$\mathbf{x}[n+1] = \phi\mathbf{x}[n] + \mathbf{\Gamma}[n]\mathbf{w}[n] \tag{E.1}$$

donde $\mathbf{x}[n] = (x[n]\ v_x[n]\ y[n]\ v_y[n]\ z[n]\ v_z[n])^t$ es el vector de estado compuesto por la posición y velocidad de cada coordenada Cartesiana y $\mathbf{w}[n] = (w_x[n]\ w_y[n]\ w_z[n])^t$ es el vector de aceleración (también llamado ruido de planta, del modelo, o del sistema), con componentes gaussianas independientes e idénticamente distribuidas en cada eje. Su matriz de covarianza es:

$$\mathbf{Q}[n] = E\{\mathbf{w}[n]\mathbf{w}^t[n]\} = \sigma_w^2\mathbf{I} \tag{E.2}$$

BIBLIOTECA VIRTUAL

Las matrices $\boldsymbol{\phi}[n]$ y $\mathbf{\Gamma}[n]$ recogen las relaciones cinemáticas necesarias entre posiciones, velocidades y aceleraciones. En este modelo van a ser constantes en el tiempo:

$$\boldsymbol{\phi}[n] = \begin{bmatrix} 1 & T & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & T & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & T \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{E.3}$$

$$\mathbf{\Gamma}[n] = \begin{bmatrix} \frac{1}{2}T^2 & 0 & 0 \\ T & 0 & 0 \\ 0 & \frac{1}{2}T^2 & 0 \\ 0 & T & 0 \\ 0 & 0 & \frac{1}{2}T^2 \\ 0 & 0 & T \end{bmatrix} \tag{E.4}$$

Estas ecuaciones muestran claramente la formulación independiente del movimiento en cada una de las coordenadas Cartesianas.

A pesar del modelo usado, se supondrá la posibilidad de que un móvil pueda realizar una maniobra repentina. Se analizará cómo se comporta el filtro ante estas situaciones que no están consideradas de forma específica en el modelo.

Existe una extensa bibliografía sobre los fundamentos del filtro de Kalman [19, 64], por lo que no se describirán aquí. El filtro de *Kalman Generalizado* actualiza su estado de acuerdo con las siguientes ecuaciones:

$$\begin{aligned} \hat{\mathbf{x}}_f[n|n] &= \hat{\mathbf{x}}_p[n|n-1] + (\mathbf{P2C}[n])\,\mathbf{K}[n]\,(\mathbf{C2P}[n]) \\ &\quad [\mathbf{r}(\mathbf{z}[n]) - \mathbf{h}(\hat{\mathbf{x}}_p[n|n-1], n)] \\ &= \hat{\mathbf{x}}_p[n|n-1] + (\mathbf{P2C}[n])\,\mathbf{K}[n]\,(\mathbf{C2P}[n])\,\mathbf{i}[n] \qquad \text{(E.5)} \\ \hat{\mathbf{x}}_p[n+1|n] &= \boldsymbol{\phi}[n]\hat{\mathbf{x}}_f[n|n] \qquad \text{(E.6)} \end{aligned}$$

donde $\mathbf{z}[n]$ es el vector con las cuatro coordenadas medidas en los planos de imagen, $r(\bullet)$ es la ecuación de conversión de las medidas en las cámaras a las medidas en 3D, $\hat{\mathbf{x}}_f[n|n]$

es el vector de estados actualizado (filtrado), $\hat{\mathbf{x}}_p[n|n-1]$ es el vector de estados predichos; $\mathbf{h}(\bullet, n)$ es la ecuación de medidas que relaciona el vector de estados con las medidas, y $\mathbf{K}[n]$ es la ganancia Kalman. Respecto a la ecuación de extrapolación (ecuación E.6), $\boldsymbol{\phi}[n]$ es la matriz de transición del modelo de estados. Finalmente $\mathbf{C2P}[n]$ y $\mathbf{P2C}[n]$ son matrices de cambio de base usadas en caso de que el proceso de innovación $\mathbf{i}[n] = \mathbf{r}(\mathbf{z}[n]) - \mathbf{h}(\hat{\mathbf{x}}_p[n|n-1], n)$ y la ganancia Kalman no estén definidos en el mismo marco de referencia. Se construyen con los autovectores de la matriz de covarianza de ruido de observación $\mathbf{R}$.

La covarianza actualiza las ecuaciones del siguiente modo:

$$\mathbf{P}_p[n+1|n] = \boldsymbol{\phi}[n]\mathbf{P}_f[n|n]\boldsymbol{\phi}^t[n] + \boldsymbol{\Gamma}[n]\mathbf{Q}[n]\boldsymbol{\Gamma}^t[n] \tag{E.7}$$

$$\mathbf{K}[n] = \mathbf{P}_p[n|n-1]\mathbf{H}^t[n](\mathbf{H}[n]\mathbf{P}_p[n|n-1]\mathbf{H}^t[n] + \mathbf{R}[n])^{-1} \tag{E.8}$$

$$\mathbf{P}_f[n|n] = (\mathbf{I} - \mathbf{K}[n]\mathbf{H}[n])\,\mathbf{P}_p[n|n-1] \tag{E.9}$$

donde $\mathbf{P}_f$, $\mathbf{P}_p$ hacen referencia a la matriz de covarianza del vector de estados filtrados y predichos respectivamente, e $\mathbf{I}$ es la matriz identidad. $\mathbf{H}[n]$ es la matriz Jacobiana de la transformación $\mathbf{h}(\bullet, n)$.

Existen diversos esquemas de seguimiento basados en el filtro de Kalman, algunos de ellos estudiados en [112], con el propósito de lograr un bajo coste computacional a la vez que se mantiene una alta precisión en el seguimiento. Básicamente, los autores comparan el funcionamiento del *filtro de Kalman acoplado* (FKA), el *filtro de Kalman desacoplado* (FKD) y el *filtro de Kalman en coordenadas principales* (FKCP) con el *filtro extendido de Kalman* (FEK). Este último está considerado en la literatura como óptimo en términos de precisión de seguimiento, aunque es el que requiere una mayor carga computacional. El FKA opera como un filtro de Kalman convencional sobre las coordenadas Cartesianas 3D. El FKD opera sobre cada una de las 3 coordenadas espaciales independientemente, ignorando el hecho de que la matriz de covarianza no es diagonal por bloques. El FKCP está inspirado en la literatura de RADAR; la idea en que se basa es realizar los cálculos para la ganancia Kalman y la actualización de la matriz de covarianza en el marco de referencia de las *direcciones principales* de la covarianza de las posiciones 3D medidas, en donde las coordenadas reconstruidas están incorreladas. La ganancia de Kalman se calcula en *coordenadas principales* y por lo tanto, la innovación debe transformarse a este nuevo marco de referencia. Una vez multiplicada, se devuelve a coordenadas cartesianas 3D y es usada para extrapolar el estado. Estos cambios de base se realizan mediante las matrices **C2P** y **P2C**.

E.3 TRABAJOS PREVIOS

En esta sección se revisarán algunos esquemas de seguimiento basados en lógica borrosa, así como una parte de la teoría de detección de maniobras necesaria para la derivación del filtro final.

E.3.1 Esquemas de seguimiento basados en lógica borrosa

Chan y otros en [16], proponen un control borroso de los parámetros de un filtro $\alpha-\beta$. Como variables de entradas al control usan el error y el incremento de error, ambos normalizados (tal como se explica en la sección E.4.1). En [26] los autores hacen una extensión (casi literal) de las ideas de [16] aplicadas a un filtro de Kalman bidimensional. La covarianza del ruido de planta, $\mathbf{Q}$, se controla mediante una inferencia borrosa, y toma valores entre 0 y 1.

Estas dos aproximaciones tienen su punto débil en la forma que utilizan para normalizar el error y el incremento del error (sec. E.4.1). En caso de que las trayectorias sean constantes (salvo ruido de observación) en una determinada dirección del espacio, el error normalizado oscilará constantemente entre 0 y 1, haciendo muy difícil un control de éste. Así mismo, se ha comprobado que para aplicaciones en estéreo-visión, el uso de sólo 4 conjuntos uniformemente distribuidos no es suficiente para una correcta conversión de la variable de salida.

Roberts *et al.* en [111] utilizan estimadores neuro-difusos para inicializar un filtro de Kalman. Otro trabajo interesante es el de Kwok y Chambers [74]. Los autores plantean un control borroso de un filtro de Kalman 2D mediante la sintonización de las varianzas de ruido de observación, usando como variables de entrada el error cuadrático medio de la velocidad estimada y su incremento, siguiendo un planteamiento muy similar a [26], pero a nuestro juicio bastante más eficiente.

En otra línea se encuentra el trabajo de Chen *et al.* [17]. Partiendo del *Filtro de Kalman por Intervalos*, los autores sustituyen la operativa de dicho filtro por una operativa borrosa, con el fin de hacer borrosos cada uno de los intervalos de entrada. Su trabajo es equivalente a la creación de 3 modelos simultáneos y la posterior unión de éstos mediante el uso de lógica borrosa.

Otros trabajos que merece la pena citar son los desarrollados por McGinnity [88, 89] encaminados a permitir que un filtro de Kalman lineal pueda ser usado en sistemas con un alto nivel de no-linealidad. Siguiendo una línea totalmente distinta de razonamiento, en la que se sustituye toda la operativa por una serie de inferencias borrosas (en vez de controlar algún parámetro) nos encontramos con el trabajo de Alberola y Cybenko [7].

E.3.2 Detección de Maniobras y Filtrado Adaptativo

El filtro de Kalman está determinado por el modelo cinemático que asumimos. La elección de los parámetros para cualquier filtro aproximado o simplificado debe depender inherentemente de un modelo de maniobra asumido. Cuando la cinemática real del blanco difiere de la del modelo usado por el filtro ocurren graves errores de seguimiento. El filtro de Kalman modela la dinámica de los blancos usando variables aleatorias continuas, descritas estadísticamente por parámetros conocidos. El filtro sólo dará es-

timaciones apropiadas de la posición y velocidad del blanco si el modelo subyacente es correcto.

Sin embargo, si el blanco realiza y mantiene una maniobra repentina (como la maniobra evasiva de un avión de combate), entonces, el modelo subyacente falla. En ese momento hay una discontinuidad en la aceleración. Estos *saltos* en la aceleración no pueden ser manejados con eficiencia por el modelo. El resultado puede ser la pérdida de la pista.

A continuación se presentarán diversas técnicas representativas de las tres aproximaciones generales al filtrado adaptativo de maniobras [12]:

1. *Detección de Maniobra y Reajuste de Parámetros:* La primera, y probablemente la más simple, consiste en mantener el mismo filtro que antes de la detección de la maniobra, pero variando sus parámetros. Lo más sencillo es usar dos o más regiones de correlación. Los parámetros del filtro se ajustan en función de la región en que caiga la observación. De este modo, pares *observación-predicción* que indiquen maniobra muy pequeña, o no maniobra (NM) hacen que no haga falta el reajuste. Maniobras pequeñas (SM) darán lugar a un cierto incremento en la matriz de covarianza Kalman, pero no hay modificación de la estimación de la variable de estado. Por último, una maniobra grande (LM), da lugar a una reinicialización del filtro usando los dos últimos puntos.

   Existen otros métodos de detección de maniobras basados en el estudio de la historia del residuo [12].

2. *Detección de Maniobra y Aumento de Estado:*

   En este caso, tras la detección de una maniobra se usa en el filtro un modelo de estado diferente. Antes de ésta, suponemos un modelo de blanco con velocidad constante. Tras la detección de la maniobra, el seguimiento se realiza aumentando un estado el filtro, de tal manera que el modelo haga uso de la aceleración. Mantenemos este nuevo modelo hasta que la aceleración estimada sea insignificante, momento en el que podemos volver al modelo lineal.

3. *Uso de Varios Filtros en Paralelo*

   Se basa en el uso simultáneo de varios modelos, y por lo tanto de varios filtros trabajando a la vez. Las salidas de dichos filtros pueden ser ponderadas por diversos pesos o bien se puede escoger *la mejor* en función de un criterio estadístico. Esta aproximación por lo tanto requiere de varios filtros de Kalman operando en paralelo y de una estimación permanente *a posteriori* para validar los resultados.

E.4 FILTRO DE KALMAN CON CONTROL BORROSO

Los dos principales puntos de una aplicación que use el filtro de Kalman son la estimación de la matriz de covarianza de ruido de planta y la robustez del filtro en

presencia de discrepancias respecto al modelo, como en el caso de que el objeto realice una maniobra [12, 19]. La elección de un valor demasiado alto para la varianza del ruido de planta hace que el filtro sea más impulsivo: responde mejor a las observaciones a costa de perder capacidad de suavizado. Por otro lado, un valor demasiado bajo hace que el filtro pierda la pista tan pronto como el móvil realice un movimiento inesperado.

Respecto a la estimación de la varianza de ruido de planta se ha optado por la filosofía de [26, 74], haciendo que ésta (y por consiguiente su matriz de covarianza, $\mathbf{Q}$) sea variable y adaptativa al escenario, en función del error de predicción en el instante anterior y del incremento de dicho error. Esto permite al filtro tomar el valor de ruido más adecuado para cada caso.

Respecto a la robustez frente a maniobras, se ha optado por seguir la filosofía de ajuste de parámetros comentada anteriormente. Se monitorizará la historia del error de predicción del filtro, y, si los errores llegan a mantenerse durante intervalos largos, se reiniciará el filtro. Para ello se empleará un FFS similar al usado en la aplicación de seguimiento de información basada en texto ya descrita (sección 5.1).

Se ha tomado como filtro de partida el filtro de Kalman en Coordenadas Principales [112] (FKCP), debido a que opera de manera independiente sobre cada una de las coordenadas espaciales. Esto da lugar a que tanto las matrices $\mathbf{P}_f$, $\mathbf{P}_p$, como la ganancia Kalman sean matrices diagonales por bloques. Sin embargo el paso a otros esquemas Kalman es casi inmediato.

El primer paso del diseño será encontrar un criterio correcto para la normalización del error de predicción, de modo que éste caiga entre 0 y 1, para hacer posible un futuro control borroso.

E.4.1 Normalización del Error

Para realizar el control borroso, se han tomado como variables de entrada el error de predicción ($\boldsymbol{E}_p[n]$) y el incremento del error de observación ($\boldsymbol{\Delta E}_p[n]$), siguiendo lo propuesto por diversos autores [71, 74, 16, 26]; estos se definen de la siguiente manera:

- Error de predicción:
$$\mathbf{E}_p[n] = \mathbf{r}(\mathbf{z}[n]) - \hat{\mathbf{x}}_p[n|n-1] \tag{E.10}$$
Es un vector de tres elementos, cuyas componentes son $\mathbf{E}_p[n] = [E_{px}[n]\ E_{py}[n]\ E_{pz}[n]]^t$. Para futura referencia $\mathbf{r}(\mathbf{z}[n]) = [x_0[n]\ y_0[n]\ z_0[n]]^t$ y $\hat{\mathbf{x}}_p[n|n-1] = [x_p[n]\ y_p[n]\ z_p[n]]^t$.
- Cambio del error, definido
$$\boldsymbol{\Delta E}_p[n] = \mathbf{E}_p[n] - \mathbf{E}_p[n-1] \tag{E.11}$$
En este caso $\boldsymbol{\Delta E}_p[n] = [E_{px}[n] - E_{px}[n-1]\ E_{py}[n] - E_{py}[n-1]\ E_{pz}[n] - E_{pz}[n-1]]^t$.

Es importante (aunque no siempre reciba la atención que merece) definir un método que permita normalizar estas dos medidas, de modo que sean apropiadas para ser entradas de un controlador borroso. En este caso, la normalización no es algo trivial, ya que la matriz de covarianza del ruido de observación es dependiente de la posición dentro del sistema estéreo.

Existen distintos métodos de normalización del valor del error:

- Método propuesto por Chan y otros [16, 26]. Dado que el máximo valor del error depende de la velocidad del móvil, los autores normalizan las componentes x e y del siguiente modo

$$E_{px}[n] = \begin{cases} \frac{x_0[n]-x_p[n]}{x_0[n]-x_0[n-1]} & si \ \ |x_0[n]-x_p[n]| < |x_0[n]-x_0[n-1]| \\ \frac{x_0[n]-x_p[n]}{|x_0[n]-x_p[n]|} & si \ \ |x_0[n]-x_p[n]| > |x_0[n]-x_0[n-1]| \\ 0.0 & si \ \ x_0[n]-x_p[n] = x_0[n]-x_0[n-1] = 0 \end{cases} \tag{E.12}$$

$$\Delta E_{px}[n] = \begin{cases} \frac{E_{px}[n]-E_{px}[n-1]}{E_{px}[n-1]} & si \ \ |E_{px}[n]-E_{px}[n-1]| < |E_{px}[n-1]| \\ \frac{E_{px}[n]-E_{px}[n-1]}{|E_{px}[n]-E_{px}[n-1]|} & si \ \ |E_{px}[n]-E_{px}[n-1]| > |E_{px}[n-1]| \\ 0.0 & si \ \ E_{px}[n]-E_{px}[n-1] = E_{px}[n-1] = 0 \end{cases} \tag{E.13}$$

  El problema aparece cuando una trayectoria es constante (excepto ruido) en una dirección (el objeto no se mueve en esa dirección). Cualquier ruido hará que el error se vaya a 1, por lo que el error de salida oscilará entre 0 y 1. No es lo idóneo para la aplicación. (En [5] pueden encontrarse una serie de ejemplos de este problema).

- Normalizar por la *desviación típica* de ruido de observación. Sin embargo, en nuestra aplicación, al tratarse de un sistema estereoscópico, debemos acudir a la matriz de covarianzas, lo que nos da un valor de normalización distinto en cada instante de tiempo, para cada coordenada espacial. De este modo, las varianzas serían:

$$\begin{aligned} \sigma_x^2 &= \frac{(b^2+x^2)z^2}{2b^2f^2}\sigma^2 \\ \sigma_y^2 &= \frac{(b^2+y^2)z^2}{2b^2f^2}\sigma^2 \\ \sigma_z^2 &= \frac{z^4}{2b^2f^2}\sigma^2 \end{aligned} \tag{E.14}$$

  que se corresponden con la diagonal principal de la matriz de covarianza $\mathbf{R}(x, y, z)$. Resultados experimentales [5] muestran que este tipo de normalización tampoco es adecuada para la aplicación en estudio.

- Otra forma de calcular el nivel de error para una normalización, es usar la matriz de covarianza del residuo $\mathbf{S}$ tal y como se muestra en [12]. La matriz se define de la siguiente forma:

$$\mathbf{S} = \mathbf{S}[n|n-1] = \mathbf{H}[n]\mathbf{P}_p[n|n-1]\mathbf{H}^t[n] + \mathbf{R}[n] \tag{E.15}$$

Si se asume una distribución gaussiana del residuo, puede definirse un valor de normalización de la siguiente forma:

$$\mathbf{S}_M = (2\pi)^{M/2}\sqrt{|\mathbf{S}[n|n-1]|} \tag{E.16}$$

siendo M la dimensión del filtro, en este caso 3. Resultados experimentales muestran que para el FKCP el valor adecuado es 3.[1]. Para obtener un mayor rango dinámico se dobla este valor de normalización:

$$\mathbf{S}_M = 2(2\pi)^{M/2}\sqrt{|\mathbf{S}[n|n-1]|} \tag{E.17}$$

Este último método es el empleado. Conviene tener en cuenta que el valor de normalización varía en el tiempo y con la posición. La normalización se realizará de manera separada para cada una de las componentes espaciales, por lo que se minimizarán los efectos de absorción de valores [5] cuando el error presenta un nivel diferente en cada dirección. El coeficiente de normalización para cada dirección será

$$\begin{aligned}
\mathbf{S}_\mathbf{x}[n] &= 2(2\pi)^{M/2}\sqrt{\mathbf{R}(1,1)[n|n-1] + \mathbf{P}_p(1,1)[n|n-1]} \\
\mathbf{S}_\mathbf{y}[n] &= 2(2\pi)^{M/2}\sqrt{\mathbf{R}(2,2)[n|n-1] + \mathbf{P}_p(3,3)[n|n-1]} \\
\mathbf{S}_\mathbf{z}[n] &= 2(2\pi)^{M/2}\sqrt{\mathbf{R}(3,3)[n|n-1] + \mathbf{P}_p(5,5)[n|n-1]}
\end{aligned} \tag{E.18}$$

con $\mathbf{A}(i,j)[n|n-1]$ la componente (i,j) de la matriz $\mathbf{A}[n|n-1]$. Nótese que la matriz $\mathbf{H}$ (ecuación (E.8)) es una matriz de unos y ceros que simplemente extrae las posiciones predichas del vector de estados si el seguimiento se realiza en un espacio 3D. Por lo tanto la operación $\mathbf{H}[n]\mathbf{P}_p[n|n-1]\mathbf{H}^t[n]$ resulta ser una simple extracción de la componente apropiada de la matriz $\mathbf{P}_p$.

E.4.2 Motor de Inferencias Borroso. Adaptación de la matriz de covarianza de ruido de planta

Como ya se ha comentado anteriormente, el control borroso del filtro de Kalman consta de dos partes bien diferenciadas:

1. Adaptación de la matriz de covarianza de ruido de planta ($\mathbf{Q}[n]$) en función del error de predicción.
2. Control del reinicio del filtro en caso de desacople grande.

Para la adaptación de la matriz de covarianza, se toman como entradas al sistema dos variables, el error de predicción y el incremento del error. El error de predicción se

[1] Para el *FKCA* y el *FKCD* la práctica ha demostrado que la normalización debe realizarse con $M = 1$, $\mathbf{S}_M = 2(2\pi)^{1/2}\sqrt{|\mathbf{S}[n|n-1]|}$

halla normalizado, de tal manera que sus valores oscilan en el intervalo $[0, 1]$. Los valores del incremento estarán por lo tanto en el intervalo $[-1, 1]$. Para el control de la matriz $\mathbf{Q}$ existe la posibilidad de considerar de forma separada los errores en las tres direcciones espaciales, de forma que cada componente se maneje de forma independiente, o bien de forma conjunta, considerando la media cuadrática de los errores. La total separación de las coordenadas espaciales en el *FKCP* hace que sea mejor un control independiente de cada una de ellas. Experimentalmente se ha comprobado que existe una clara mejora del control, trabajando de forma independiente, cuando en una de las dimensiones el error es muy diferente a las otras. Esto se debe, a que de forma conjunta hay un cierto grado de *absorción* de valores[2], por lo que el error total tiende a ir a un valor medio.

Se definen dos variables lingüísticas de entrada, *Error* ($e[n]$) e *Incremento del Error* ($\dot{e}[n]$), que se modelan como una serie de conjuntos borrosos, tal como se muestra en las figuras E.2 y E.3.

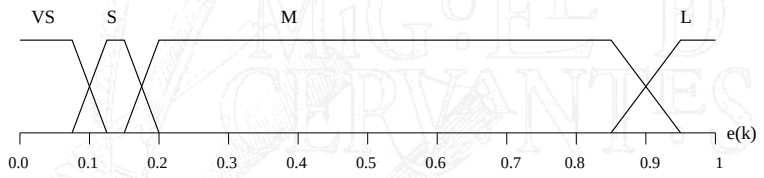


Figura E.2 Funciones de pertenencia para la entrada borrosa *Error*.

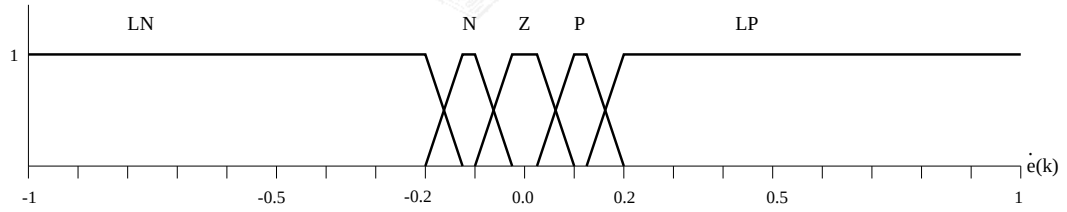


Figura E.3 Funciones de pertenencia para la entrada borrosa *Incremento del Error*.

Para la variable *Error* se han creado 4 conjuntos borrosos PTS, llamados *Muy pequeño*, *Pequeño*, *Medio* y *Grande*. En la figura aparecen como VS, S, M y L, siguiendo la inicial de la notación inglesa (tal y como aparecen en [5]). De forma análoga, para el incremento se han creado 5 conjuntos: *Negativo Grande, Negativo, Cero, Positivo* y *Positivo Grande* (LN, N, Z, P y LP).

Como variable de salida, se considera la desviación típica del ruido de planta para cada coordenada espacial. Los centroides de esta variable son

$$C[n] = [0.005, 0.025, 0.05, 0.1, 1.5] \tag{E.19}$$

Para el razonamiento borroso se han empleado operadores max-min y obtención de valores por centroides [71, 90]. Se ha usado una base de reglas que se puede expresar como

[2]Entendemos por tal el hecho de que si una componente presenta errores grandes, las variaciones del error en las otras componentes quedan enmascaradas por el primero, no pudiendo realizarse una adaptación acorde a cada componente [5].

Si $E_p[n]$ es $e[n]$ y $\Delta E_p[n]$ es $\dot{e}[n]$ entonces $\sigma_w[n]$ es $C(i)[n]$

La base de reglas completa expresada en forma de tabla es la siguiente:

		$\Delta E_p[n]$				
		LN	N	Z	P	LP
$E_p[n]$	VS	C1	C2	C2	C2	C5
	S	C1	C2	C3	C3	C5
	M	C2	C3	C3	C3	C3
	L	C4	C4	C4	C5	C5

BIBLIOTECA VIRTUAL

Esta base se ha obtenido mediante entrenamiento del sistema borroso [116, 136] con simulaciones sobre el escenario.

El valor de salida corresponde con la desviación típica del ruido de planta, σ_w^2. La matriz de covarianza $\mathbf{Q}[n]$ será entonces

$$\mathbf{Q}[n] = \begin{bmatrix} \sigma^2_{w(x)}[n] & 0 & 0 \\ 0 & \sigma^2_{w(y)}[n] & 0 \\ 0 & 0 & \sigma^2_{w(z)}[n] \end{bmatrix} \tag{E.20}$$

E.5 EJEMPLOS PRÁCTICOS Y RESULTADOS

En la presente sección se pretende dar una visión real del funcionamiento del filtro, y probar cómo en ciertos casos su uso mejora notablemente el seguimiento. Los experimentos se han hecho utilizando como base el FKCP, que es con el que se ha trabajado. La comparación se realiza entre el filtro convencional y el filtro con control borroso. Una descripción más exhaustiva de estos experimentos puede encontrarse en [5].

Para todos los experimentos se ha considerado un ruido de observación con desviación típica de ruido $\sigma_v = \sigma_u = \sigma = 10^{-4}m$, y unas distancias $b = 0.1m$ y $f = 0.0505m$ (ver Fig. E.1) según las restricciones planteadas en [112]. En todos los experimentos se proporcionan estimaciones del error de filtrado a la salida del filtro obtenidas a partir de 1000 iteraciones de Monte Carlo. Los filtros *convencionales* utilizan por defecto una desviación típica de la maniobra $\sigma_w = 0.05m/s^2$. El movimiento se registra con una frecuencia de muestreo de 50 *Hz*. La eficiencia del filtro en cada instante de tiempo se mide por medio de la raíz del error cuadrático medio total (TRMSE), que se calcula como la raíz cuadrada de la suma del error cuadrático medio de cada iteración de Monte Carlo para cada instante de tiempo.

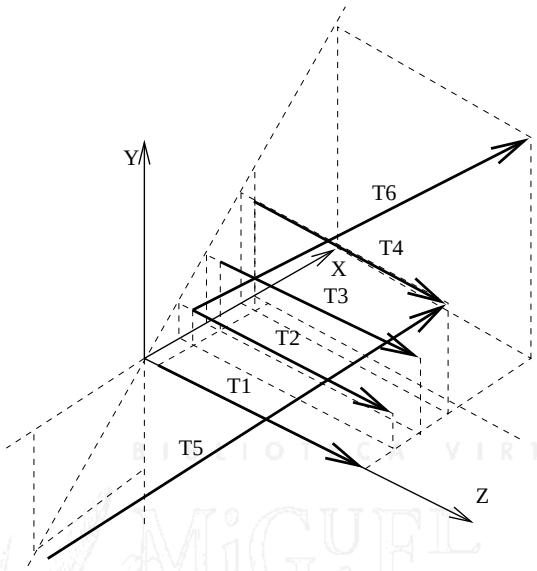


Figura E.4 Trayectorias del primer experimento

E.5.1 Experimento 1: Trayectorias Rectilíneas Uniformes

Como punto de partida, se compararán los filtros para las 6 trayectorias utilizadas en [112], ya que los resultados de estas trayectorias para los filtros son conocidos. Además, dichas trayectorias recogen un buen número de casos significativos para el seguimiento en un entorno estereoscópico (Fig. E.4):

- *Trayectoria T1:* El punto inicial es $\mathbf{r_0} = (0\ 0\ 0.1)$ y el punto final es $\mathbf{r_f} = (0\ 0\ 1.1)$. Toda la trayectoria discurre a lo largo del eje Z. Los coeficientes de correlación cruzadas teóricos son nulos.
- *Trayectoria T2:* El punto inicial es $\mathbf{r_0} = (0.1\ 0.1\ 0.1)$ y el punto final es $\mathbf{r_f} = (0.1\ 0.1\ 1.1)$. Aparecen ya coeficientes de correlación cruzada.
- *Trayectoria T3:* El punto inicial es $\mathbf{r_0} = (0.3\ 0.3\ 0.1)$ y el punto final es $\mathbf{r_f} = (0.3\ 0.3\ 1.1)$. Hay coeficientes de correlación cruzada.
- *Trayectoria T4:* El punto inicial es $\mathbf{r_0} = (0.5\ 0.5\ 0.1)$ y el punto final es $\mathbf{r_f} = (0.5\ 0.5\ 1.1)$. Los coeficientes de correlación cruzada son muy altos.
- *Trayectoria T5:* El punto inicial es $\mathbf{r_0} = (-0.5\ -0.5\ 0.1)$ y el punto final es $\mathbf{r_f} = (0.5\ 0.5\ 1.1)$. Existe movimiento uniforme en las tres direcciones del espacio.
- *Trayectoria T6:* El punto inicial es $\mathbf{r_0} = (0.1\ 0.1\ 0.1)$ y el punto final es $\mathbf{r_f} = (1.1\ 1.1\ 1.1)$. Existe movimiento uniforme en las tres direcciones del espacio. Trayectoria radial.

Los resultados de dichos experimentos se encuentran reflejados en la figura E.5. Se puede apreciar que sí existe una cierta disminución del error en todos los casos.

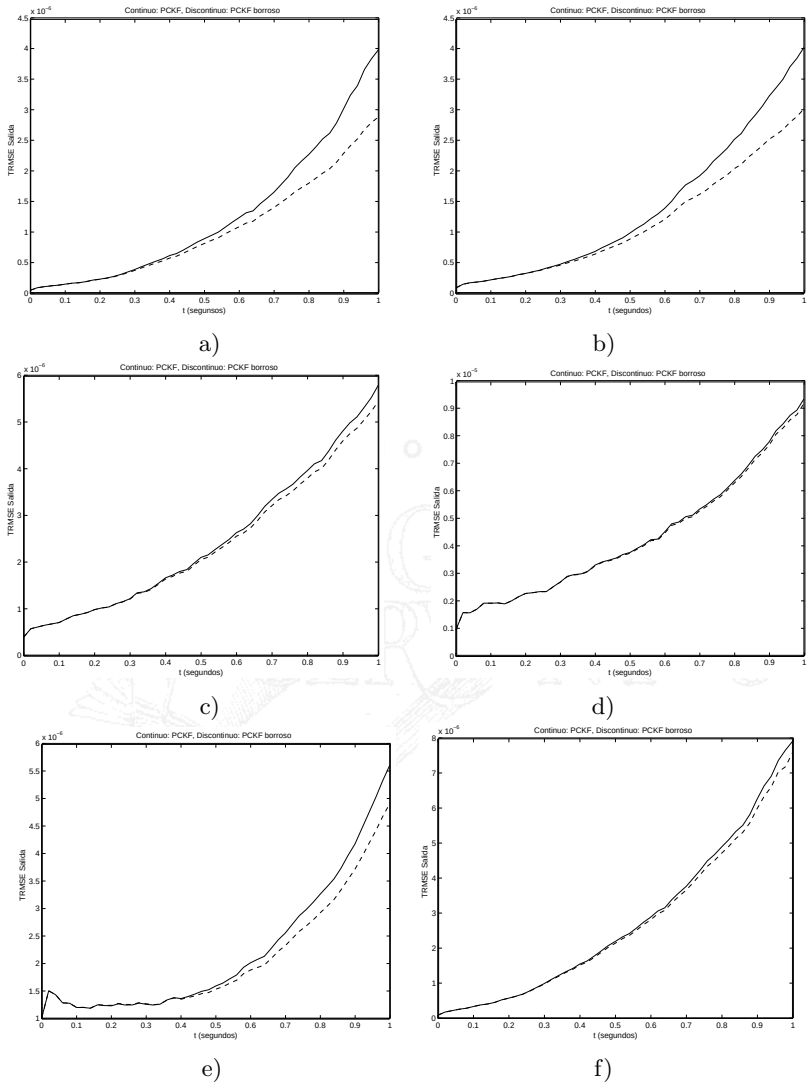


Figura E.5 Comparación del FKCP clásico con el borroso. a) Trayectoria T1 b) Trayectoria T2. c) Trayectoria T3. d) Trayectoria T4. e) Trayectoria T5. f) Trayectoria T6.

Esta disminución por sí sola, no es lo suficientemente importante como para justificar la operativa adicional que supone un control borroso. Sin embargo, el control va a funcionar mejor en otros casos, en los que el filtro de Kalman presenta grandes errores de salida, o bien en casos en los que el movimiento es lento. En tales casos, la adaptación de la matriz $\mathbf{Q}[n]$ es mucho más que significativa.

E.5.2 Experimento 2: Trayectorias Rectilíneas Uniformes de Baja Velocidad

En los modelos de baja velocidad, el seguimiento del filtro es esperable que sea más exacto que en los ejemplos antes mencionados. Sin embargo, una σ_w grande en este caso hará que las predicciones oscilen a un lado y a otro del valor real con un margen demasiado amplio. En esos casos, el uso de un control que disminuya el valor de la varianza de ruido de planta puede ser significativo.

Se han planteado dos trayectorias distintas (Fig. E.6):

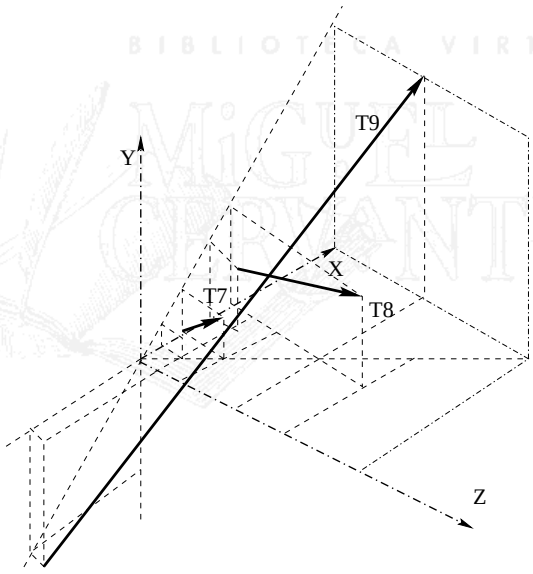


Figura E.6 Trayectorias usadas en el segundo experimento.

- *Trayectoria T7:* El punto inicial es $\mathbf{r_0}$ = (0.1 0.1 0.1) y el punto final es $\mathbf{r_f}$ = (0.2 0.2 0.2). El móvil se mueve con una velocidad de $0.1m/s$ en cada una de las direcciones espaciales.
- *Trayectoria T8:* El punto inicial es $\mathbf{r_0}$ = (0.3 0.3 0.1) y el punto final es $\mathbf{r_f}$ = (0.4 0.4 0.6). El vector de velocidad es $\mathbf{v}$ = (0.1 0.1 0.5).

Los resultados se encuentran reflejados en la figura E.7. La mejora del filtro es tanto mayor cuanto menor es la velocidad de éste.

Podría pensarse que el filtro no funciona correctamente a altas velocidades. Si se define la trayectoria T9), entre los puntos $\mathbf{r_0}$ = (-0.5 -0.5 0.1) y $\mathbf{r_f}$ = (1.5 1.5 0.4) (vector de velocidad $\mathbf{v}$ = (2 2 0.5)) se observa (Fig. E.8) que el filtro borroso sigue funcionando mejor.

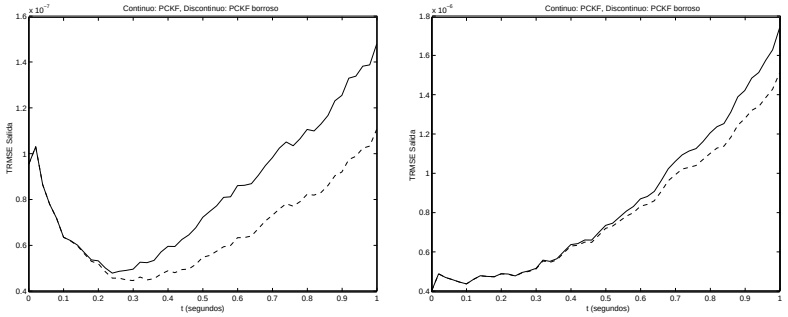


Figura E.7 Comparación del FKCP convencional con el filtro con control borroso. a) Trayectoria T7 b) Trayectoria T8

BIBLIOTECA VIRTUAL

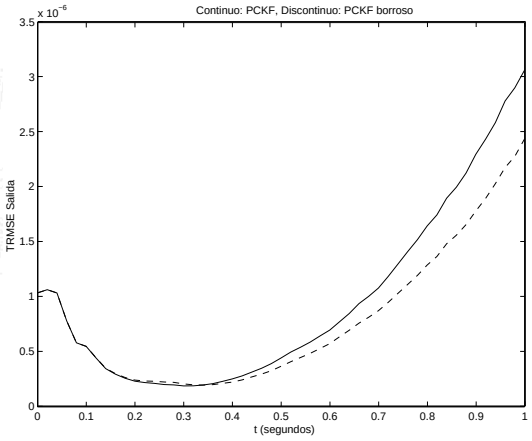


Figura E.8 Comparación entre el FKCP clásico y borroso para una trayectoria de alta velocidad (T9)

En la sección 5.2.3 se describe un experimento sobre trayectorias rectilíneas con presencia de maniobra.

E.6 CONCLUSIONES

Los resultados prácticos muestran que el control borroso hecho sobre el filtro de Kalman (concretamente sobre el Filtro de Kalman en Coordenadas Principales, *FKCP*) mejora en la mayoría de los casos los resultados de dicho filtro. Aunque no en todas las aplicaciones va a merecer la pena ampliar la carga computacional para obtener una mejora del filtro (ya que para eso hay otros esquemas basados en Kalman que con menos operativa obtienen mejores resultados), sí será útil en todos esos entornos en que se pueda producir un cambio en la dirección del móvil a observar.

Se ha demostrado que el control borroso (tanto de la matriz de covarianza de ruido de planta, $\mathbf{Q}[n]$, como del reinicio en caso de desacoplo fuerte), proporciona unos resultados

muy buenos en casos extremos, tales como móviles de baja velocidad o cambios de trayectoria.

Aunque la filosofía aquí desarrollada se basa en el *FKCP*, la extrapolación a otros esquemas Kalman es casi inmediata, requiriendo tan sólo un re-entrenamiento para afinar la base de reglas (que seguramente será muy parecida a la actual) o ajustar los conjuntos de las variables lingüísticas al nuevo nivel de normalización. Pero sin duda, la filosofía que subyace será la misma.

BIBLIOTECA VIRTUAL MIGUEL DE CERVANTES

F

APÉNDICE AL MÉTODO TW2

F.1 CARACTERÍSTICAS DE LOS 13 HUESOS RUS

F.1.1 Radio

Estadio	Características
A	· Epífisis no presente
B	· Centro apenas visible, borde mal definido.
C	· Centro con forma oval, borde liso y definido
D	· $d \geq D/2$ · Epífisis ensanchada en cara lateral (forma picuda) · Tercio medio de la superficie proximal se ha aplanado. Separación 1mm.
E	· Línea blanca densa en borde distal de epífisis
F	· Zona palmar y dorsal (2 líneas) · Borde proximal de la epífisis se adapta a metáfisis.
G	· Bordes articulares de escafoides y semilunar (2 semicurvas). · Superficie dorsal y palmar (borde interno) · Borde proximal cóncavo
H	· Epífisis corona metáfisis en un punto
I	· Comienza la fusión

F.1.2 Cúbito

Estadio	Características
A	· Epífisis no presente
B	· Centro apenas visible, borde mal definido.
C	· Centro con forma oval, borde liso y definido
D	· $d \geq D/2$ · Alargada (diámetro horizontal $\gg$ diámetro vertical) · Borde distal y proximal aplanados. Forma picuda.
E	· Apófisis estiloides visible
F	· Cabeza de cúbito definida y diferenciada. · Borde lateral aplanado
G	· $d = D$ · Solapamiento en el tercio medio
H	· Comienzo de fusión

F.1.3 Metacarpiano I

Estadio	Características
A	· Epífisis no presente
B	· Centro apenas visible, borde mal definido.
C	· Centro con forma oval, borde liso y definido
D	· $d \geq D/2$
E	· $d = D/2$ · Concavidad en borde proximal
F	· Diferenciación de la superficie proximal (palmar y dorsal) Articulación en forma de silla de montar se adapta al trapecio
G	· Epífisis corona metáfisis
H	· Comienza fusión entre epífisis y metáfisis
I	· Fusión casi completa

F.1.4 Metacarpianos III y V

Estadio	Características
A	· Epífisis no presente
B	· Centro apenas visible, borde mal definido.
C	· Centro con forma oval, borde liso y definido
D	· $d \geq D/2$
E	· Epífisis forma de uña
F	· Superficie palmar y dorsal (Líneas blancas densas)
G	· $d = D$ · Epífisis corona metáfisis
H	· Comienza fusión entre epífisis y metáfisis · Cartílago menos de 3/4 de hueso
I	· Fusión completa

F.1.5 Falange Proximal I

Estadio	Características
A	· Epífisis no presente
B	· Centro apenas visible, borde mal definido.
C	· Centro con forma oval, borde liso y definido
D	· $d \geq D/2$
E	· Borde proximal cóncavo y ensanchado · Cara interna mayor que externa
F	· $d = D$ · Forma similar epífisis y metáfisis
G	· Epífisis corona metáfisis
H	· Comienza fusión entre epífisis y metáfisis
I	· Fusión completa

F.1.6 Falanges Proximales III y V

Estadio	Características
A	· Epífisis no presente
B	· Centro apenas visible, borde mal definido.
C	· Centro con forma oval, borde liso y definido
D	· $d \geq D/2$
E	· Borde proximal cóncavo y ensanchado
F	· $d = D$ · Forma similar epífisis y metáfisis
G	· Epífisis corona metáfisis
H	· Comienza fusión entre epífisis y metáfisis
I	· Fusión completa

F.1.7 Falanges Medias III y V

Estadio	Características
A	· Epífisis no presente
B	· Centro apenas visible, borde mal definido.
C	· Centro con forma oval, borde liso y definido
D	· $d \geq D/2$
E	· Centro de superficie proximal ha crecido
F	· $d = D$
G	· Epífisis corona metáfisis
H	· Comienza fusión entre epífisis y metáfisis
I	· Fusión completa

F.1.8 Falange Distal I

Estadio	Características
A	· Epífisis no presente
B	· Centro apenas visible, borde mal definido.
C	· Centro con forma oval, borde liso y definido
D	· $d \geq D/2$
E	· $d = D/2$ · Forma distinta
F	· Borde proximal cóncavo. Se adapta a la otra falange · $d > D$ · Borde distal: Superficie interna y externa. (Forma de silla de montar)
G	· Epífisis corona metáfisis
H	· Comienza fusión entre epífisis y metáfisis
I	· Fusión casi completa

F.1.9 Falanges Distales III y V

Estadio	Características
A	· Epífisis no presente
B	· Centro apenas visible, borde mal definido.
C	· Centro con forma oval, borde liso y definido
D	· $d \geq D/2$
E	· $d = D$ · Forma distinta (pero no superficies palmar y dorsal)
F	· Superficies palmar y dorsal. Adaptadas a articulaciones
G	· Epífisis corona metáfisis
H	· Comienza fusión entre epífisis y metáfisis
I	· Fusión completa

F.2 CARACTERÍSTICAS DE LOS HUESOS PARA EL MÉTODO RUS

A continuación se exponen las características que se han seleccionado para clasificar cada hueso. Cada valor de las característica va acompañada de un número entre paréntesis, que es el número que representará dicho valor en tablas posteriores.

Radio 1. *Epífisis:* No presente (1), Presente y apenas visible (2), presente y separada (3), epífisis corona metáfisis (4), comienza la fusión (5).

2. *Diámetros:* Relación entre diámetro de metáfisis (D) y diámetro de la epífisis (d). Valor de la división D/d.
3. *Forma:* Forma oval (1), forma picuda con base plana (2), forma picuda con borde proximal adaptado a metáfisis (3), bordes articulares de escafoides y semilunar y borde proximal cóncavo (4).
4. *Superficies:* Ausentes (1), en forma de una línea blanca densa (2), en forma de 2 líneas densas (3), superficies palmar y dorsal visibles (4).

Cúbito 1. *Epífisis:* No presente (1), Presente y apenas visible (2), presente y separada (3), epífisis corona metáfisis (4), comienza la fusión (5).

2. *Diámetros:* Relación entre diámetro de metáfisis (D) y diámetro de la epífisis (d). Valor de la división D/d.
3. *Forma:* Forma oval (1), forma picuda con bordes aplanados (2), apófisis estiloides visible (3), cabeza de cúbito definida (4).

Metacarpiano I 1. *Epífisis:* No presente (1), Presente y apenas visible (2), presente y separada (3), epífisis corona metáfisis (4), comienza la fusión (5), fusión casi completa (6).

2. *Diámetros:* Relación entre diámetro de metáfisis (D) y diámetro de la epífisis (d). Valor de la división D/d.
3. *Forma:* Forma oval (1), concavidad en borde lateral (2), articulación en silla de montar (3).
4. *Superficies:* Ausentes (1), superficie proximal (2).

Metacarpiano III y V 1. *Epífisis:* No presente (1), Presente y apenas visible (2), presente y separada (3), epífisis corona metáfisis (4), comienza la fusión (5), fusión casi completa (6).

2. *Diámetros:* Relación entre diámetro de metáfisis (D) y diámetro de la epífisis (d). Valor de la división D/d.
3. *Forma:* Forma oval (1), forma de uña (2).
4. *Superficies:* Ausentes (1), superficie palmar y dorsal (2).

Falange proximal I 1. *Epífisis:* No presente (1), Presente y apenas visible (2), presente y separada (3), epífisis corona metáfisis (4), comienza la fusión (5), fusión casi completa (6).

2. *Diámetros:* Relación entre diámetro de metáfisis (D) y diámetro de la epífisis (d). Valor de la división D/d.
3. *Forma:* Forma oval (1), borde proximal cóncavo (2), forma de epífisis y metáfisis parecida (3).

Falanges proximal III y V 1. *Epífisis:* No presente (1), Presente y apenas visible (2), presente y separada (3), epífisis corona metáfisis (4), comienza la fusión (5), fusión casi completa (6).
2. *Diámetros:* Relación entre diámetro de metáfisis (D) y diámetro de la epífisis (d). Valor de la división D/d.
3. *Forma:* Forma oval (1), borde proximal cóncavo (2), forma de epífisis y metáfisis parecida (3).

Falanges Medias III y V 1. *Epífisis:* No presente (1), Presente y apenas visible (2), presente y separada (3), epífisis corona metáfisis (4), comienza la fusión (5), fusión casi completa (6).
2. *Diámetros:* Relación entre diámetro de metáfisis (D) y diámetro de la epífisis (d). Valor de la división D/d.
3. *Forma:* Forma oval (1), forma triangular (2).
4. *Superficies:* Ausentes (1), superficie proximal (2).

Falange distal I 1. *Epífisis:* No presente (1), Presente y apenas visible (2), presente y separada (3), epífisis corona metáfisis (4), comienza la fusión (5), fusión casi completa (6).
2. *Diámetros:* Relación entre diámetro de metáfisis (D) y diámetro de la epífisis (d). Valor de la división D/d.
3. *Forma:* Forma oval (1), forma triangular (2), silla de montar (3).

Falanges Distales III y V 1. *Epífisis:* No presente (1), Presente y apenas visible (2), presente y separada (3), epífisis corona metáfisis (4), comienza la fusión (5), fusión casi completa (6).
2. *Diámetros:* Relación entre diámetro de metáfisis (D) y diámetro de la epífisis (d). Valor de la división D/d.
3. *Forma:* Forma oval (1), forma triangular (2).
4. *Superficies:* Ausentes (1), superficies palmar y dorsal (2).

F.3 DATOS DE ENTRENAMIENTO PARA LOS DISTINTOS HUESOS

F.3.1 Radio

Estadio	Pertenencia	Epífisis	Forma	Superficies	Diámetros
A	1	1	1	1	10
B	1	2	1	1	5
B	1	2	1	1	3
C	1	3	1	1	2.5
C	0.8	3	1	1	2
D	0.2	3	1	2	2.5
C	0.2	3	1	2	1.9
C	0.5	3	2	1	2.5
C	0.5	3	2	1	2.2
D	0.9	3	2	1	2
D	1	3	2	1	1.9
E	0.9	3	2	2	2
E	1	3	2	2	1.5
E	0.5	3	2	3	2
F	0.9	3	2	3	1.5
F	0.5	3	3	2	1.8
F	0.8	3	3	2	1.10
F	0.5	3	3	3	2
F	1	3	3	3	1.11
F	0.8	3	3	3	1
F	0.8	3	3	4	1
G	0.7	3	4	3	1
G	1	3	4	4	1
H	0.2	4	3	3	1
H	0.8	4	3	4	1
G	0.6	4	4	3	0.95
H	1	4	4	4	1
I	0.7	5	3	4	0.88
I	0.3	5	4	3	0.90
I	0.2	5	3	3	0.91
I	1	5	4	4	0.8

F.3.2 Cúbito

Estadio	Pertenencia	Epífisis	Forma	Diámetros
A	1	1	1	10
B	1	2	1	7
C	0.7	2	1	3
C	1	3	1	3
C	0.5	3	1	2.2
D	0.8	3	1	2
D	1	3	2	1.9
D	0.8	3	2	1.5
E	0.8	3	2	1.3
D	0.2	3	2	2
E	0.5	3	3	1.6
E	1	3	3	1.3
F	0.9	3	4	1.3
F	1	3	4	1.1
F	0.5	4	3	1
G	0.5	4	3	1
G	1	4	4	1
H	1	5	4	1

F.3.3 Metacarpiano I

Estadio	Pertenencia	Epífisis	Forma	Superficies	Diámetros
A	1	1	1	1	10
B	1	2	1	1	5
C	0.8	2	1	1	2.5
C	1	2	1	1	4
C	0.9	2	1	1	2.8
C	0.8	3	1	1	2.8
D	1	3	1	1	2
D	0.8	3	1	1	1.3
D	0.5	3	2	1	1.5
E	0.5	3	1	1	1
E	1	3	2	1	1
E	0.5	3	3	1	1
F	0.8	3	2	2	1
F	1	3	3	2	1
G	0.5	4	2	2	0.9
G	1	4	3	2	0.9
G	0.9	4	3	2	1
H	1	5	3	2	0.9
H	0.9	5	3	2	1
G	0.1	5	3	2	1
I	1	6	3	2	0.9

F.3.4 Metacarpianos III y V

Estadio	Pertenencia	Epífisis	Forma	Superficies	Diámetros
A	1	1	1	1	10
B	1	2	1	1	4
C	0.5	2	1	1	2.5
C	1	3	1	1	2.5
C	0.5	3	1	1	2
D	1	3	1	1	1.9
D	1	3	1	1	1.5
D	0.7	3	1	1	1.4
E	0.2	3	1	2	1
E	1	3	2	1	1.1
E	0.8	3	2	1	1.3
F	1	3	2	2	1.1
F	0.8	3	2	2	1.5
G	0.6	4	2	1	1
G	1	4	2	2	1
H	0.8	5	2	1	1
H	1	5	2	2	0.75
H	1	5	2	2	1
I	1	6	2	2	1

F.3.5 Falange proximal I

Estadio	Pertenencia	Epífisis	Forma	Diámetros
A	1	1	1	10
B	1	2	1	5
C	0.8	2	1	2.5
C	1	3	1	2.5
C	0.5	3	1	2.1
D	0.5	3	1	2
D	1	3	1	1.8
D	1	3	1	1.3
E	1	3	2	1.3
E	1	3	2	1.8
D	0.5	3	2	2
E	0.5	3	3	1.2
F	1	3	3	1
E	0.8	3	1	1
G	1	4	3	1
G	0.5	4	2	0.9
H	1	5	3	1
I	1	6	3	1

F.3.6 Falange proximal III y V

Estadio	Pertenencia	Epífisis	Forma	Diámetros
A	1	1	1	10
B	1	2	1	5
C	0.8	2	1	2.2
C	1	3	1	2.5
C	0.5	3	1	2
D	1	3	1	1.9
D	0.8	3	1	1.5
D	0.8	3	1	1.4
E	0.5	3	2	2
E	1	3	2	1.8
E	1	3	2	1.5
E	1	3	2	1.2
E	0.6	3	3	1.5
F	1	3	3	1
F	0.9	3	3	0.9
E	0.8	3	1	0.9
G	1	4	3	0.9
G	0.5	4	2	0.9
H	1	5	3	0.9
I	1	6	3	0.9

F.3.7 Falange media III y V

Estadio	Pertenencia	Epífisis	Forma	Superficies	Diámetros
A	1	1	1	1	10
B	1	2	1	1	5
C	0.8	2	1	1	2.5
C	1	3	1	1	2.5
C	0.5	3	1	1	2
D	1	3	1	1	1.8
D	1	3	1	1	1.4
F	0.8	3	1	2	1
E	0.8	3	2	1	1.2
E	0.8	3	2	2	1.5
E	0.8	3	2	2	1.4
E	1	3	2	2	1.2
F	1	3	2	2	1
F	0.8	3	2	1	1
F	0.8	3	2	2	0.9
G	1	4	2	2	1
G	0.6	4	2	1	1
H	1	5	2	2	1
H	0.8	5	2	1	1
I	1	6	2	2	1

BIBLIOTECA VIRTUAL MIGUEL DE CERVANTES

F.3.8 Falange distal I

Estadio	Pertenencia	Epífisis	Forma	Diámetros
A	1	1	1	10
B	1	2	1	5
C	0.2	2	1	2.5
C	1	3	1	2.5
C	0.5	3	1	2
D	0.5	3	1	2
D	1	3	1	1.8
D	1	3	1	1.5
D	1	3	1	1.1
D	0.5	3	2	1.2
E	0.5	3	1	1
E	1	3	2	1
E	0.6	3	3	1.1
F	1	3	3	0.9
E	0.8	3	1	1
F	0.5	3	3	1.1
G	1	4	3	0.9
G	0.5	4	2	0.9
F	0.8	3	2	0.8
H	1	5	3	0.9
I	1	6	3	0.9

F.3.9 Falange distal III y V

Estadio	Pertenencia	Epífisis	Forma	Superficies	Diámetros
A	1	1	1	1	10
B	1	2	1	1	5
C	0.5	2	1	1	2.3
C	1	3	1	1	2.5
C	0.5	3	1	1	2
D	1	3	1	1	1.8
D	1	3	1	1	1.5
D	0.8	3	1	1	1.3
E	0.5	3	1	1	1
F	0.8	3	1	2	1
E	1	3	2	1	1
F	1	3	2	2	1
F	0.8	3	2	2	1.2
F	0.8	3	2	2	0.9
G	0.7	4	2	1	1
G	1	4	2	2	0.8
H	0.8	5	2	1	0.8
H	1	5	2	2	0.8
I	0.8	6	2	1	0.8
I	1	6	2	2	0.8

RESUMEN

BIBLIOTECA VIRTUAL
MIGUEL DE CERVANTES

Resumen

En este trabajo se plantea un nuevo marco matricial para realizar inferencias en sistemas borrosos de manera eficiente. Para este tipo de sistemas, si las entradas cumplen unas restricciones suaves, es posible precalcular varios pasos intermedios, lo cual se traduce en que el proceso de inferencia requiera de menos operativa, logrando así un método rápido y compacto. El método es eficiente en el caso en que las entradas al sistema sean conjuntos borrosos o bien gránulos borrosos. Este último concepto ha sido redefinido en la tesis como una representación de conceptos lingüísticos basada en grados de solapamiento. Se ha realizado un estudio de los sistemas borrosos realimentados que hace uso de su representación matricial, obteniendo una caracterización de su comportamiento asintótico ante una entrada constante. Este nuevo marco operativo se ha aplicado sobre varios problemas reales. Por un lado sobre esquemas de seguimiento, como el algoritmo MHT borroso y un detector borroso de maniobra en un entorno de seguimiento específico. Por otro lado, se ha aplicado a la construcción de un clasificador automático para el cálculo del estado de maduración esquelética en la infancia, el cual replica el método médico actualmente en uso.

Abstract

In this thesis we propose a new matrix framework to carry out inferences in fuzzy systems efficiently. For this sort of systems, provided that inputs satisfy a number of mild conditions, it is possible to precalculate several intermediate steps, which draws an inference process with minimal actual operative, giving rise to a fast and compact processing method; this procedure is efficient in the case that inputs are fuzzy sets or fuzzy granules. This latter concept has been redefined in the thesis as a new representation of linguistic concepts based on the overlap degree between them. A specific study of fuzzy feedback systems—based on the above mentioned matrix framework— has been carried out, and a characterization of their asymptotic behavior with an steady input has been derived. This new operative framework has been applied to several real-world problems. On one hand, for two tracking systems, the first of which is the fuzzy MHT algorithm and the second a fuzzy maneuver detector applied to an specific tracking scenario. On the other hand, the methodology has been applied to create an automatic classifier for bone age assessment in the infancy, which mimics the medical method currently in use.