UNIVERSIDAD DE GRANADA

E.T.S. DE INGENIERÍA INFORMÁTICA

Departamento de Ciencias de la Computación
e Inteligencia Artificial

DEPENDENCIAS FUNCIONALES DIFUSAS EN BASES DE DATOS RELACIONALES DIFUSAS

TESIS DOCTORAL

Juan Carlos Cubero Talavera

Granada, Mayo de 1994

DEPENDENCIAS FUNCIONALES DIFUSAS EN BASES DE DATOS RELACIONALES DIFUSAS

JUAN CARLOS CUBERO TALAVERA

A mis padres y a Belén

La memoria titulada **Dependencias Funcionales Difusas en Bases de Datos Relacionales Difusas**, que presenta Juan Carlos Cubero Talavera, para optar al grado de DOCTOR, ha sido realizada en el Departamento de Ciencias de la Computación e Inteligencia Artificial de la Universidad de Granada, bajo la dirección de Mª Amparo Vila Miranda, Catedrática del Departamento en el que se ha realizado la memoria.

Granada, Junio de 1994.

El Doctorando	El Director
Fdo. Juan Carlos Cubero Talavera	Fdo. Mª Amparo Vila Miranda

AGRADECIMIENTOS

A la profesora Mª Amparo Vila Miranda, sin cuya dirección, consejos y entusiasmo esta memoria jamás habría visto la luz.

A los profesores Francisco José Cortijo, Juan Huete, Juan Miguel Medina y Olga Pons cuyas discusiones y comentarios me fueron de gran ayuda en el desarrollo de distintos puntos de este trabajo, y por el contínuo apoyo, consejo y ánimo que siempre me han demostrado. También quiero hacer extensiva mi gratitud al resto de los miembros del departamento de Ciencias de la Computación e Inteligencia Artificial, en especial, a los profesores Javier Abad, Juan Luis Castro, Miguel Delgado y Francisco Herrera.

A la Junta de Andalucía por la concesión de una beca F.P.I que me permitió iniciar el trabajo que se presenta en esta memoria.

Finalmente, pero no en último lugar, quiero agradecer a mi familia el apoyo que siempre me ha brindado durante el período de confección de este trabajo.

Indice

Figuras

Notación Específica Utilizada en la Memoria

Término	Notación	Página
Esquema relacional de r	$REL(r)$	17
Existe un único	$\exists !$	207
Restricción de r a X	$r\|_X$	52
Reconstrucción (crisp) de una relación (crisp) r	$m_\rho(r)$	39
	$m(r)$	238
$(\alpha,\beta)-$d.f.d	$\xrightarrow{(\alpha,\beta)}$	88
$(\alpha,\beta)-$d.f.d.r	$\stackrel{(\alpha,\beta)}{\Longrightarrow}$	194
Semejanza débil	$\simeq^d$	128
Semejanza fuerte	$\simeq^f$	136
Redundancia entre difusos	$\mathcal{R}$	168
Redundancia entre tuplas	$\mathcal{R}$	168
Fusión de tuplas redundantes	$\mathcal{F}$	60 y 169
Fusión de tuplas $X-$redundantes	$\mathcal{F}^X$	177
$X-$Proyección difusa sobre W	$\widetilde{\Pi}^X_W$	178
Clase de una tupla	$[t]^X$	192
Operador extracción reglas	$RULE^X_{XY}(r)$	203
Reunión Dirigida	$\sqsupset\!\!\bowtie$	219
Reunión Dirigida Difusa	$\sqsupset\!\!\widetilde{\bowtie}$	224
Reconstrucción difusa de una relación r		
utilizando la proyección difusa	$\widetilde{m}\,(r)$	239
utilizando el operador $RULE$	$\widetilde{m}^*\,(r)$	239
Inclusión entre relaciones	$\preceq$	241
Extensión entre relaciones	$\sqsubseteq$	241

Prólogo

A. Introducción

La necesidad de almacenamiento y gestión eficiente de un gran número de datos, plantea la necesidad de definir modelos teóricos para su gestión. En la literatura existen distintas aproximaciones, entre las que destacamos las siguientes: ***bases de datos relacionales, bases de datos en red, bases de datos jerárquicas y bases de datos orientadas a objetos.*** El modelo relacional fue introducido por Codd, y ha sido hasta la fecha el de mayor desarrollo investigador y comercial, por lo que será éste el marco de trabajo de nuestra investigación.

La importancia del modelo relacional radica en el tratamiento de una relación como una asociación entre una clave y una propiedad. Para mantener la semántica de dicha interpretación se desarrolla una teoría de mantenimiento de restricciones y de diseño de relaciones que constituyen los pilares de este modelo. Una de las principales herramientas desarrolladas para llevar a cabo dicho cometido, la constituye el concepto de dependencia funcional, así como los de normalización y descomposición sin pérdidas.

Una de las principales desventajas del modelo relacional es que no ha sido desarrollado para tratar con informaciones imprecisas. Este tipo de informaciones pueden ser representadas a través de la teoría de conjuntos difusos desarrollada originalmente por Zadeh, por lo que resultaría de extrema importancia poder incorporar dicha teoría al modelo relacional clásico. Hay diversas aproximaciones a este problema, y en este trabajo se destacarán las principales. Estas extensiones pueden agruparse en distintas categorías, como son:

- Estudios que se limitan a incorporar valores difusos en una base de datos y a realizar consultas para extraer información, pero no a realizar una extensión difusa de los pilares del modelo relacional a los que aludíamos anteriormente.

– Aquellos que estudian la extensión difusa de definición de dependencia funcional. Sin embargo, o no realizan un estudio de descomposiciones sin pérdidas, o aquellos que sí abordan dicha extensión, definen operadores de álgebra relacional no difusos.

Analizando este panorama, podemos concluir que es necesario definir un modelo relacional de base de datos difusa y una teoría de diseño concebida desde una perspectiva difusa.

B. Esquema del trabajo

Pasamos ahora a describir los epígrafes que componen este trabajo:

En el primer capítulo revisaremos brevemente el modelo relacional clásico. En primer lugar haremos una breve revisión de los conceptos básicos utilizados para el almacenamiento de datos y su consulta, y fijaremos la nomenclatura y notación que seguiremos a lo largo de este trabajo. A continuación pasaremos a describir brevemente el proceso de diseño (verdadero pilar de la teoría *relacional*) de una base de datos relacional clásica, para lo cual se introducirá previamente el concepto de dependencia funcional. Los epígrafes de este capítulo, marcarán las directrices a seguir en el desarrollo de esta tesis.

En el capítulo segundo revisaremos las extensiones difusas propuestas para el almacenamiento de datos difusos (*base de datos relacional difusa*), así como de los operadores necesarios para su tratamiento. Posteriormente revisaremos las extensiones difusas propuestas para las dependencias funcionales, es decir, las *dependencias funcionales difusas*. En todos los resultados que se incluyen en este trabajo relativos al modelo clásico (capítulo 1), o a las extensiones difusas (primera parte del capítulo 2), omitiremos las desmostraciones, refiriendo al lector al artículo o libro dónde pueden encontrarse.
Finalmente, introduciremos una primera aproximación original, en base a la identificación de ciertas propiedades que serían deseables establecer en la definición de una dependencia funcional difusa.
En este segundo capítulo no entraremos en el estudio de la fase de diseño global de una base de datos relacional difusa, debido a que es un tema no abordado hasta la fecha por ningún autor. La aproximación a este problema propuesta en el trabajo, se presentará extensamente en el último capítulo.

En el capítulo número tres, analizaremos cómo han de compararse entre sí los valores de los atributos *antecedentes*, así como para los atributos *consecuentes* presentes en una dependencia funcional. Veremos que habrá de ser distinta, siendo una medida de semejanza *débil* para los del primer tipo, mientras que deberemos utilizar una medida *fuerte* para los atributos consecuentes. Así mismo, se introducirá un concepto importante, a saber, el *nivel de granularidad* permitido en una base de datos relacional difusa: éste concepto establecerá que los datos que pueden aparecer en atributos antecedentes y consecuentes no puedan ser demasiado difusos. Esta imposición puede verse como una relajación de la restricción presente en el modelo clásico, en el que no se permiten valores imprecisos como datos presentes en dichos atributos.
Introduciremos una primera aproximación heurística a estos problemas, y a continuación presentaremos una aproximación formal que, bajo ciertas restricciones, será una extensión de la primera. Incluiremos ciertas propiedades sobre las medidas de semejanza, necesarias para obtener buenas propiedades en capítulos posteriores.

En el cuarto capítulo trataremos el problema de la eliminación de redundancia de forma coherente, de tal manera que se obtenga un proceso cerrado. Introduciremos el operador de proyección difusa, y en base a estos puntos, y junto a los conceptos vistos en el capítulo dos, daremos una definición de dependencia funcional difusa más restrictiva que la dada en el segundo capítulo. Ello, junto con el concepto de redundancia, nos permitirá introducir la definición de llave difusa, de vital importancia en el proceso de diseño.

En el último capítulo introduciremos el operador del álgebra relacional de reunión difusa y demostraremos que las dependencias se conservan bajo este operador. Posteriormente definiremos qué es una descomposición sin pérdidas difusas, que nos permitirá demostrar el teorema fundamental de descomposiciones sin pérdidas, de vital importancia para el diseño de una base de datos. Aplicaremos casos particulares de este teorema para justificar la necesidad de llegar en el diseño, hasta una relación en tercera forma normal (difusa), pero no hasta la forma normal (difusa) de Boyce y Codd.

Capítulo 1

El Modelo de Base de Datos Relacional Clásico

El estudio del modelo clásico de base de datos relacional (**b.d.r**) abarca tres puntos fundamentales:

1. Definición de las herramientas necesarias para almacenar datos (concepto de *relación*, *llave candidata*, etc), así como de los operadores necesarios para extraer información (planteamiento de consultas) de la base de datos. Dichos operadores constituyen el *álgebra y cálculo relacional.*

2. Imposición de las restricciones de integridad necesarias para mantener una coherencia en y entre las relaciones que conforman la base de datos.

3. Definición de las herramientas necesarias para decidir cuales han de ser los campos de información que han de conformar una relación de la base de datos. Este estudio es conocido como la fase de *diseño* de la base, y se fundamenta principalmente en los conceptos de depedencia funcional y descomposición sin pérdidas.

En este primer capítulo, revisaremos el primer y segundo punto de una forma resumida, abordando después el tercero con más detalle, por constituir un punto clave dentro de esta memoria.

1.1 El Sistema de Almacenamiento de Datos y de Extracción de Información

1.1.1 Elementos Básicos del Modelo Relacional

El objetivo de esta sección es establecer las definiciones y notaciones del modelo relacional clásico, para que sirvan de marco de referencia en la extensión difusa que se aborda en este trabajo. Por ello, realizaremos una descripción muy breve; para más detalles sobre cualquiera de los puntos que trataremos en este apartado, se recomiendan los textos de Codd [27], Date [34] y Ullman [78]. Los conceptos básicos con los que trataremos son lo siguientes:

* **Relación**: Puede considerarse en principio como un fichero plano o *tabla* en la que se guardan los datos en cada *casilla*: posteriormente presentaremos una definición más formal. Se utilizarán minúsculas itálicas del final del alfabeto para su designación, como por ejemplo q, r, s. En ejemplos concretos, utilizaremos identificadores con el siguiente tipo de letra: `Persona`, `Comerciantes`, ...

* **Tupla**: Es cada una de las filas (*registro*) de la relación. Se utilizarán minúsculas itálicas del final del alfabeto posteriores a la letra s para su designación, como por ejemplo t, u, v, w.

* **Atributo**: Es cada una de las columnas (*campo*) de la relación. Utilizaremos letras mayúsculas itálicas del principio del alfabeto para nombrar atributos simples, como por ejemplo A, A_i, B. Cuando queramos designar conjuntos de atributos, utilizaremos letras mayúsculas itálicas al final del alfabeto: X, Y, Z, es decir:

  $$X = (A_i)_{i=1\ldots n}$$

  Cuando se quiera expresar explícitamente la relación a la que pertenece el atributo, utilizaremos el nombre de ésta como superíndice:

  $$A^r \; , \; X^r \; , \; Y^s$$

Puntualicemos que en los textos clásicos, suele especificarse lo anterior anteponiendo el nombre de la relación al atributo, separándolos con un punto. Los ejemplos anteriores quedarían como siguen:

$$r.A \ , \ r.X \ , \ s.Y$$

Hemos cambiado esta notación para hacerla más sintética, con vistas a emplearla en desarrollos matemáticos.

Normalmente utilizaremos simple yuxtaposición para indicar unión conjuntista de atributos, es decir:

$$ABC \equiv \{A, B, C\}$$

* **Esquema Relacional**: Es el conjunto de todos los atributos incluidos en una relación r. Lo designaremos con $REL(r)$ o simplemente REL si no hay confusión.

* **Dominio de un atributo**: Es el conjunto de todos los valores posibles que puede tomar un atributo. Lo notaremos con D o bien D_A para enfatizar que es el dominio correspondiente al atributo A. Es frecuente que varios atributos de una misma relación tengan el mismo dominio. Los elementos (ó valores) de un dominio, los escribiremos en el siguiente tipo de letra: `15, Mario, rojo, ...`

La definición formal de relación surge cuando se considera una tupla t de una relación r con esquema relacional $REL = (A_i)_{i=1...n}$, como un elemento del producto cartesiano de los correspondientes dominios

$$t \in D_1 \times \ldots \times D_n$$

y una relación, como un subconjunto de dicho producto:

$$r \subseteq D_1 \times \ldots \times D_n$$

Así pues, una relación puede considerarse como una **instancia** de un esquema relacional.

Al valor del atributo A para una tupla arbitraria t, lo designaremos por $A(t)$. Utilizaremos también, para un conjunto de atributos $X = (A_i)_{i\in I}$, la siguiente notación:

$$X(t) = (A_i(t))_{i\in I}$$

* **Valor nulo** denotado por `nulo`. Es un valor especial que puede aparecer como dato en cualquier atributo y que representa *valor desconocido, propiedad no aplicable*, etc. Ver sección 1.2.4, en la que se realiza un estudio más detenido.

* **Llave** o **clave candidata**: Es un conjunto de atributos que identifican unívocamente cada tupla de una relación. Deben verificar varias propiedades (denotamos por K al conjunto de dichos atributos):

  1. No pueden tomar el valor nulo en ninguno de los atributos que conforman K:

  $$A_i(t) \neq \texttt{nulo} \quad \forall\, t \quad \forall\, A_i \in K$$

  Esta imposición se conoce como como la **restricción de integridad de entidad**, y es vital para mantener la interpretación semántica de una relación como una asociación entre una entidad conformada por la llave, y una serie de propiedades (el resto de los atributos del esquema) que la describen.

  Relación $\equiv$ Asociación Clave – Propiedad

  2. *Unicidad*: Dos tuplas de la relación no pueden tener los mismos valores para todos los atributos que componen la llave.

  $$K(t) \neq K(t') \quad \forall\, t, t' \in r \tag{1.1}$$

  o lo que es lo mismo:

  $$\forall\, t, t' \in r \quad A_i(t) = A_i(t') \quad \forall\, A_i \in K \;\Leftrightarrow\; t = t'$$

  3. *Minimalidad*: Ningún subconjunto de atributos de las llaves candidatas verifica la unicidad.

  $$\forall\, X \subseteq REL \;\text{ tal que } X(t) \neq X(t') \quad \forall\, t, t' \in r \;\Rightarrow\; K \subseteq X \tag{1.2}$$

En una misma relación pueden existir varias llaves candidatas, pero obligatoriamente ha de elegirse una de ellas, que llamaremos **llave primaria** , quedando las restantes como **llaves alternativas.**

Cuando una llave candidata es compuesta, es decir, la conforman más de un atributo simple, entonces cualquier atributo incluido en la clave se dirá que es un **atributo primo**

* **Llave Externa**. Un conjunto de atributos X^r en una relación r, se dice que forman una llave externa a una relación s, si y solo si X^s es una llave candidata en s, y además, toda tupla t verifica lo siguiente:

  – O bien todos los valores son nulos:

$$A_h(t) = \texttt{nulo} \quad \forall\ A_h \in X$$

  – O bien existe una tupla en s con los mismos valores en X:

$$\exists\ u \in s \ \text{ tal que } A_h^r(t) = A_h^s(u) \quad \forall\ A_h \in X$$

La imposición de llaves externas se realiza en base a cuestiones semánticas relativas a la asociación entre dos relaciones de la misma base, y se conoce como la **regla de integridad referencial**. Esta imposición obliga a que las relaciones r y s de la definición anterior estén siempre ligadas a través de X, de forma que, por ejemplo, las actualización de r con una tupla t ha de ser tal que $X^r(t)$ verifique la definición de llave externa.

* Un concepto que es trivial de definir en la teoría clásica (no ocurrirá en la extensión difusa) es el de redundancia entre dos tuplas:

Definición 1.1.1 (Redundancia modelo clásico) *Dos tuplas t_1, t_2 de una relación r de una base de datos relacional clásica, se dice que son* **redundantes** *si y solo si* $A(t_1) = A(t_2) \quad \forall\ A \in REL(r)$

Una imposición del modelo relacional es que no puede haber dos tuplas redundantes (duplicadas) en la base de datos. Obsérvese que, derivada de esta imposición, toda relación contiene al menos una llave candidata (todo el esquema).

* Otros conceptos inmediatos son los de inclusión e igualdad entre dos relaciones:

  – r está **incluida** en s, y lo notaremos por $r \subseteq s$, si y solo si:

$$REL(r) = REL(s) \text{ y además } \ \forall\ t \in r\ \exists\ u \in s \ \text{ tal que } t = u \tag{1.3}$$

– r es **igual** a s y lo notaremos por $r = s$, si y solo si:

$$r \subseteq s \text{ y además } s \subseteq r \tag{1.4}$$

Nota. Hay que señalar que los conceptos anteriores y los que siguen a continuación son teóricos; de hecho, no todos los s.b.d.r soportan todos los aspectos del modelo relacional, ni los implementan de la misma manera.

Ejemplo 1.1.2 . Un ejemplo de una relación válida sería la relación `Persona` de la figura 1.1 dónde podemos identificar los elementos básicos descritos anteriormente:

– Relación $r =$ `Persona`.

– Atributos: $A_1 =$ `Nombre`, $A_2 =$ `Edad`, $A_3 =$ `Altura`, $A_4 =$ `Color-de-Pelo`

– Llave primaria: el atributo `Nombre`.

– Dominio de por ejemplo el atributo `Altura`: $D_{\texttt{Altura}} = \{160 + i;\ i = 1 \ldots 50\}$

– Un valor de dominio que aparece en la base de datos: `15` como valor del atributo `Edad` para la tupla t_4 (la correspondiente a `Mario`):

$$A_3^r(t_4) = \texttt{Mario}$$

– Una posible tupla sería: $t_2 =$ `(Jose , 17 , 190 , Negro)`

■

Propiedades importantes presentes en el modelo relacional, tácitamente asumidas en las definiciones anteriores, son las siguientes:

1. Los atributos no están ordenados. De ellos sólo tiene importancia su posición, no su nombre.

2. Las tuplas no están ordenadas. En este sentido hay que tener en cuenta que una tabla es una representación abstracta y que distintas tablas, con las mismas tuplas ordenadas de forma diferente, representan la misma construcción.

Persona =

	Nombre	Edad	Altura	Color-de-Pelo
t_1	Juan	16	180	Marrón
t_2	Jose	17	190	Negro
t_3	Pedro	15	186	Rubio
t_4	Mario	15	178	Rojo
t_5	Carlos	27	186	Marrón

Figura 1.1. Relación Clásica `Persona`

3. Todos los valores de los atributos son atómicos; es decir, todos los dominios subyacentes son simples. Esto se traduce en que los valores de dominio sólo pueden ser átomos de los dominios. Esta es una de las restricciones más fuertes del modelo relacional, ya que los dominios permitidos no aceptan la representación de información imprecisa. Una relación que cumpla esta propiedad se dice que verifica la *primera forma normal* (ver definición 1.2.6 en el apartado 1.2.3), o simplemente, que está *normalizada*. Entonces, una **base de datos relacional** se define como una base de datos que es percibida por el usuario como una colección de relaciones normalizadas que varían con el tiempo.

1.1.2 Algebra Relacional

En este apartado abordamos el problema de la manipulación de las estructuras que aparecen en un s.b.d.r, de especial importancia en el proceso de extracción (*consultas*) o recuperación (*diseño*) de información de la base de datos. Esta manipulación se hace por medio de algún lenguaje formal de manejo de datos a través de:

- *Algebra Relacional*
- *Cálculo Relacional*

La diferencia esencial entre ambos es que, mientras que el álgebra proporciona una colección de operadores y operaciones explícitas que se usan para construir nuevas relaciones a partir de otras ya existentes, el cálculo simplemente proporciona una notación para determinar o definir la relación resultante de una petición dada (no la calcula por medio de operadores). En este trabajo haremos sólo un breve repaso al álgebra relacional, ya que es la base en la teoría de diseño[1].

Cada operador del álgebra relacional, se aplicará en una ó dos relaciones y generará una nueva relación. Los operadores básicos pueden agruparse de la siguiente forma:

◇ *Operación de asignación*

Es una operación especial que asigna el resultado de otras operaciones sobre relaciones a una nueva relación. Se incluye como operador para permitir la conservación de resultados obtenidos con la aplicación de los distintos operadores.

◇ *Operaciones Tradicionales Sobre Conjuntos*

Son unión, intersección, diferencia y producto cartesiano. Para todas ellas excepto el producto cartesiano, es necesario que las dos relaciones operando sean *compatibles*, esto es, que tengan el mismo número n de atributos y que los j-ésimos atributos de las dos relaciones ($j = 1 \ldots n$) tengan el mismo dominio.

* **Unión**: La unión de dos relaciones r y s da como resultado otra relación cuyas tuplas pertenecen a r ó a s ó a ambas. Lo notaremos por

$$r \cup s \tag{1.5}$$

* **Intersección**: La intersección de dos relaciones r y s es el conjunto de tuplas que pertenecen a r y a s. Lo notaremos por

$$r \cap s \tag{1.6}$$

[1]De hecho, ha sido la aproximación adoptada en todas las extensiones difusas al modelo relacional, a excepción de Buckles y Petry [17] y Vila et al [85]

* **Diferencia**: La diferencia entre dos relaciones r y s es el conjunto de las tuplas que pertenecen a r y no a s. Lo notaremos por

$$r/s \tag{1.7}$$

* **Producto Cartesiano Extendido**: El producto cartesiano extendido de dos relaciones r y s es el conjunto de todas las tuplas tales que cada una de ellas es la concatenación de una tupla t de r y una tupla t' de s.

  La concatenación de una tupla $t = (t_1, t_2, \ldots, t_m)$ con otra $t' = (t'_{m+1}, \ldots, t'_{m+n})$ es la tupla $t = (t_1, t_2 \ldots t_m, t'_{m+1} \ldots t'_{m+n})$. Lo notaremos por

$$r \times s \;\; \text{dónde} \; REL(r \times s) = REL(r) \cup REL(s) \tag{1.8}$$

◇ *Operaciones Especiales*

Son selección, proyección y reunión:

* **Selección**: Este operador algebraico produce un subconjunto "horizontal" de una relación específica, es decir, el subconjunto de las tuplas de una relación para las cuales se cumple un predicado dado, expresado como combinación booleana de términos, y dónde cada término es una comparación. Estas comparaciones, generalmente, comprenden valores para los atributos de la relación. Lo notaremos por:

$$\sigma_{\Theta}(r) \tag{1.9}$$

  dónde Θ expresa la condición exigida. Por ejemplo, la selección de aquellas tuplas que verifican tener como campo `Edad = 15`, de la relación `Persona` (figura 1.1) devolvería las tuplas t_3 y t_4, es decir:

$$\sigma_{\texttt{Edad=15}}(\texttt{Persona}) = \{t_3, t_4\}$$

* **Proyección** : Produce un subconjunto "vertical" de una relación dada, es decir, el subconjunto obtenido al seleccionar los atributos especificados en un orden también especificado y eliminando luego tuplas duplicadas. Lo notaremos por $\Pi_X(r)$, o bien

por $r|_X$ entendida la segunda notación como la *restricción* de r a X. Formalmente, si r tiene esquema relacional $REL \supseteq X$, entonces $\Pi_X(r)$ tiene esquema relacional X y verifica ser una relación sin tuplas redundantes y tal que:

$$t \in \Pi_X(r) \;\Leftrightarrow\; t \in r|_X \;\Leftrightarrow\; \exists\, t' \in r \;\text{ tal que } X^r(t') = t \tag{1.10}$$

Por ejemplo, sobre la relación $r =$ `Persona`:

$$\Pi_{\texttt{Edad}}(r) \equiv r|_{\texttt{Edad}} =$$

Edad
16
17
15
27

* **Reunión** ó **Join**: La reunión de la relación r sobre el atributo X con la relación s sobre el atributo Y da como resultado todas las tuplas t tales que t es la concatenación de una tupla t' de r con una tupla t'' de s de forma que se verifica una condición determinada entre los valores que toma el atributo(s) X de r y el atributo(s) Y de s. Casos particulares de esta definición son los siguientes:

  - **Equi-Reunión**: la condición exigida es la igualdad $X = Y$, y lo notaremos por $r \bowtie s$ o bien $r \bowtie_X s$ cuando se quiera expresar explícitamente el atributo sobre el que se hace la reunión. Una vez realizada la reunión se proyecta sobre X^r o equivalentemente sobre X^s. En términos de los operadores del álgebra relacional anteriores, se ve como una proyección aplicada a una selección del producto cartesiano de ambas relaciones.
  Si utilizamos la notación $R' \equiv REL(r) - X^r$ y $S' \equiv REL(s) - X^s$ tendremos:

$$r \bowtie_X s = \Pi_{R' X^r S'}\left(\sigma_{X^r = X^s}(r \times s)\right) = \Pi_{R' X^s S'}\left(\sigma_{X^r = X^s}(r \times s)\right) \tag{1.11}$$

  - **Reunión natural** entre r y s: es un caso particular de equireunión igual a

$$r \bowtie_{REL(r) \cap REL(s)} s$$

Ejemplo 1.1.3 . Supongamos dos esquemas relaciones $REL1$ y $REL2$ en los que se detallan representantes y productos respectivamente:

```
R = (Codigo-Rep , Codigo-Prod , Comision )
S = (Codigo-Prod , PVP-Producto)
```


Supongamos dos instancias de dichos esquemas, r y s:

$r =$

Codigo-Rep	Codigo-Prod	Comision
A1	1356	20
B7	1356	21
B6	3500	20

$s =$

Codigo-Prod	PVP-Producto
1356	1500
3000	1700

Entonces, la reunión natural entre r y s quedaría en la forma siguiente:

$r \bowtie_{\texttt{Codigo-Prod}} s =$

Codigo-Rep	Codigo-Prod	Comision	PVP-Producto
A1	1356	20	1500
B7	1356	21	1500

■

Nota. Obsérvese que si $X \in r$ es llave externa a s, entonces todas las tuplas de r se reunen con alguna tupla de s al realizar $r \bowtie_X s$.

$*$ **Operadores adicionales.** Bajo este epígrafe englobamos toda una serie de operadores utilizados para extraer más información de la base. No vamos a entrar en detalle a su estudio y únicamente plantearemos las definiciones que tengan algun tipo de conexión con los operadores difusos que introduciremos en los últimos capítulos de esta memoria. Destacamos los siguientes:

- **Reunión exterior izquierda**, **reunión exterior derecha** y **reunión exterior simétrica**. Fueron introducidos por primera vez por Heath en [47], y posteriormente estudiados por Codd [21] y Lacroix y Pirotte en [54]. Por ejemplo, la reunión natural a la izquierda entre r y s, se forma tomando la reunión natural entre r y s y se añaden todas las tuplas de r que no han entrado en la reunión anterior completándolas en los atributos de $REL(s)$ con valores nulos.

- **Operadores cualificados con *maybe*.** Son introducidos por Codd [25, 26] (ver también [56]) y utilizados cuando hay presentes valores nulos como datos en las relaciones a reunir. El formulismo teórico se basa en cualificar con un grado de verdad adicional (*maybe*), las comparaciones efectuadas entre valores de dominio presentes en una relación y valores nulos presentes en la misma (para operadores unarios) u otra relación (para operadores 2-arios como la reunión). No vamos a entrar en un estudio detallado; únicamente comentaremos que se necesita una lógica $n+2$ valuada si se distinguen n tipos de nulos diferentes (para $n = 1$ ver [25] y para $n = 2$ ver [26]). Vamos a expresar en términos de los operadores clásicos del álgebra relacional el operador de reunión cualificado con *maybe* ($\bowtie^{mb}$), pues tendrá alguna conexión con la extensión difusa que propondremos en el último capítulo de esta memoria:

$$r \bowtie^{mb} s = \sigma_{X^r = \mathtt{nulo} \text{ ó } X^s = \mathtt{nulo}}(r \times s) \tag{1.12}$$

1.2 El Diseño de una Base de Datos Relacional Clásica

Existen numerosas aproximaciones al problema del diseño de una base de datos relacional. Destacan la metodología más o menos heurística, dada por el *modelo entidad relación*, y la *teoría de normalización*. Nosotros nos centraremos en la segunda, basada fundamentalmente en el concepto de *dependencia funcional* que se introducirá en el apartado 1.2.2. Analizaremos propiedades importantes como los axiomas de Armstrong, que son reglas de inferencia que permiten obtener todas las dependencias posibles que se puedan derivar de un conjunto dado F de dependencias funcionales. A continuación introduciremos el concepto de forma normal, que no es sino la imposición de ciertas restricciones en cuanto a los tipos de dependencias que pueden aparecer en un esquema relacional. Finalmente revisaremos uno de los pilares fundamentales en la teoría de diseño, que corresponde al concepto de *descomposición sin pérdidas*. Al igual que los puntos de la sección anterior, vamos a describir muy brevemente los conceptos básicos; para un estudio detallado, se recomienda el texto de Ullman [78].

1.2.1 Planteamiento del Problema

El Modelo Relacional introducido previamente se fundamenta en la interpretación de una relación como un conjunto de atributos que describen a alguna entidad o clave. Podemos distinguir básicamente, dos clases de relaciones:

- Relaciones en las que todos los atributos del esquema relacional forman parte de alguna llave candidata. En este caso, tenemos una relación que describe conexiones entre distintas entidades (cada llave es una entidad).

- Relaciones en las que existen atributos que no conforman una llave. Dichos atributos describen propiedades sobre las llaves candidatas. Retomando el ejemplo 1.1.2, la tupla:

(Pedro , 15 , 186 , Rubio) $\in$ Persona

describe la entidad Pedro (suponiendo que el Nombre es el atributo clave, por lo que no hay dos tuplas con valor de Nombre igual a Pedro) a través de la Edad, la Altura y el Color de Pelo.

En cualquiera de los casos anteriores, en el momento que se viola esta interpretación surgen problemas importantes. Vamos a analizarlos con el siguiente ejemplo:

Ejemplo 1.2.1 . Supongamos que utilizamos el siguiente esquema relacional para representar una base de datos de representantes comerciales, es decir, hemos **diseñado** la base de datos de acuerdo al siguiente esquema:

$$REL = (\texttt{Codigo-Rep , Codigo-Prod , PVP-Producto , Comisión})$$

con llave primaria Codigo-Rep , Codigo-Prod. Esta relación pretende describir una asociación entre la clave y ciertos atributos que la describen. Supongamos que un producto se vende siempre a un mismo precio. Entonces, esta restricción semántica, que podríamos representar por ahora en la forma

$$\texttt{Codigo-Prod} \longrightarrow \texttt{PVP-Producto} \tag{1.13}$$

nos dice que el atributo PVP-Producto describe una propiedad del atributo Codigo-Prod que no es la llave primaria. Este hecho está rompiendo ya la filosofía del modelo relacional lo que plantea serios problemas[1] como son los siguientes:

* *Problemas de Redundancia:* Se repite el precio de un producto, cada vez que un representante vende dicho artículo.

[1]Hemos de observar que este tipo de restricciones semánticas son también un problema a resolver en otros modelos de bases de datos no relacionales. Se recomienda consultar el texto de Ullman [78] para un estudio más detallado.

$*$ *Problemas de Consistencia:*

- *Problemas de Actualización:* Si se decide cambiar el precio de un producto, hay que hacerlo en todas las tuplas en las que aparezca dicho artículo: esto es necesario para mantener la restricción semántica dada por 1.13.
- *Problemas de Inserción:* No podemos introducir el precio de un producto hasta que no exista un representante que lo venda, ya que en caso contrario, no estaría definido el valor de la llave primaria.
- *Problemas de Borrado:* Supongamos que un determinado producto sólo ha sido vendido por un único representante. Si dicho vendedor se va de la empresa y se decide borrar las tuplas correspondientes, entonces perderemos la información del precio de dicho producto. ■

La forma de evitar los anteriores problemas es *descomponiendo* REL en dos esquemas distintos: uno que contenga la información relativa a los representantes, y otro que refiera a los productos, es decir,

$$REL_1 = (\texttt{Codigo-Rep , Codigo-Prod , Comision})$$
$$REL_2 = (\texttt{Codigo-Prod , PVP-Producto})$$

que son precisamente los que utilizábamos en el ejemplo 1.1.3. Esto se verá con detalle en el apartado 1.2.3. Por ahora debemos responder la siguiente pregunta: ¿cómo detectamos matemáticamente estos atributos que no concuerdan con la interpretación semántica de *relación*? La respuesta la damos en el siguiente apartado:

1.2.2 Dependencias Funcionales Clásicas

◇ *Definición*

Debemos expresar matemáticamente las restricciones semánticas del tipo de la expresada en la ecuación 1.13. El que cada producto tenga asociado un único precio, nos dice que, si en cualquier relación con esquema REL (ejemplo 1.2.1) aparecen dos tuplas con el mismo producto, entonces debe aparecer el mismo valor para el atributo `PVP-Producto`. Cuando esto sucede, decimos que existe una dependencia funcional del tipo:

$$\texttt{Codigo} - \texttt{Prod} \longrightarrow \texttt{PVP} - \texttt{Producto}$$

Estamos en condiciones de dar la definición de dependencia funcional[1] introducida por primera vez por Codd en [21].

Definición 1.2.2 *Sea un esquema relacional $REL = \{A_h\}_{h=1\ldots n}$. Diremos que REL satisface una* **dependencia funcional clásica** *(***d.f.c** *o simplemente* **d.f***) denotado por $X \to Y$, con X, Y cualquier subconjunto de atributos de REL, con $X = (A_i)_{i \in I}$, $Y = (A_j)_{j \in J}$, $I, J \subseteq \{1 \ldots n\}$ si se verifica la siguiente restricción para cualquier instancia r de REL.*

$$\forall\, t_1,\ t_2 \in r,\ si\ X(t_1) = X(t_2)\ entonces\ debe\ ser\ Y(t_1) = Y(t_2)$$

dónde $X, Y \subseteq R$ son los atributos **antecedente** y **consecuente** respectivamente.

Es importante destacar que es el experto quien debe elicitar la restricción semántica $X \to Y$ sobre un esquema REL, y será el procedimiento matemático expresado en la definición 1.2.2 el que compruebe la validez de dicha afirmación sobre una instancia particular r de REL. Por abuso del lenguaje, utilizaremos también la siguiente expresión:

$$r \text{ satisface una dependencia funcional } X \to Y$$

Vamos a introducir una formulación equivalente a la dada en la definición 1.2.2 con objeto de plantear en el capítulo cuarto una extensión difusa al concepto de dependencia funcional. Podemos interpretar dicha restricción suponiendo la existencia de una función univaluada:

$$X \to Y \;\Leftrightarrow\; \exists\, f : D_X(r) \longrightarrow D_Y(r) \quad \forall\, r \tag{1.14}$$

[1] Existen muchos otros tipos de dependencias como *multivaluadas, de reunión, de inclusión,* ... (una buena revisión se encuentra en el artículo de Fagin y Vardi [44]), pero no se verán en esta memoria ya que no entraremos en su extensión difusa

dónde $D_X(r)$ representa el conjunto de valores del dominio D_X que aparecen en la relación r, y análogamente para D_Y. Al ser univaluada, nos aseguramos que a cada valor que aparezca en $D_X(r)$, le corresponda un único valor del atributo consecuente Y.

◇ *Derivación de Dependencias*

Cuando una relación satisface un conjunto F de d.f, podemos afirmar que también satisface otra serie de d.f *implicadas* por F. Por ejemplo, si r satisface

$$F = \{A \rightarrow B\ ,\ B \rightarrow C\}$$

entonces, es inmediato demostrar que por *transitividad*, r también satisface la d.f $A \rightarrow C$. Esto nos lleva a introducir las siguientes definiciones:

Definición 1.2.3 *Sea F un conjunto de d.f para un esquema relacional REL, y sea $I = X \rightarrow Y$ una d.f. arbitraria. Diremos que I se* **deriva lógicamente** *de F, y lo denotaremos por $F \models I$, si y solo si, toda relación r con esquema REL que satisfazga F, también satisface I.*

Definición 1.2.4 *Sea F un conjunto de d.f para un esquema relacional REL. Definimos el* **cierre** *o* **clausura** *de F, denotado por F^+, como el conjunto de <u>todas</u> las d.f que se derivan lógicamente de F:*

$$F^+ = \{I \ \text{ tal que } F \models I\}$$

Es obvio que si de partida hubiese sido, por ejemplo, $F = \{X \rightarrow Y\}$, entonces la dependencia $X \rightarrow Z$ no podría haberse derivado usando la transitividad. Con esto queremos enfatizar el hecho de que estamos interesados en derivar, <u>a partir de un conjunto F dado de antemano</u> todas las dependencias que se deriven lógicamente de él (F^+).

Con el uso de estas definiciones podemos reformular la definición que dimos de llave candidata[1]

[1] Esta definición la utilizaremos en la extensión difusa del concepto de llave

Definición 1.2.5 *Sea un esquema $REL = (A_1, A_2, \ldots, A_n)$ verificando un conjunto F de d.f. Diremos que el subconjunto $K \subseteq REL$ forma una llave candidata de REL si y solo si se satisfacen las siguientes condiciones:*

- $K \rightarrow A_1 A_2 \ldots A_n \in F^+$

- *Si* $\exists\, Y \subseteq REL$ *tal que* $Y \rightarrow A_1 A_2 \ldots A_n \in F^+$, *entonces debe ser* $Y \supseteq K$.

Ahora bien, ¿cómo se construye el conjunto F^+? Para responder a esta pregunta, debemos plantearnos otra cuestión previa, a saber ¿podemos aplicar ciertas reglas de inferencia para deducir nuevas dependencias? Por ejemplo, supongamos que

$$F = \{X \rightarrow Y\}$$

Entonces es obvio que, en general

$$X \not\rightarrow YZ \quad \forall\, Z \;\Leftrightarrow\; X \rightarrow YZ \notin F^+$$

Así pues, la anterior no sería una regla de inferencia válida. Sin embargo, la regla de transitividad introducida al principio de este apartado sí es aplicable, de forma que:

$$\{X \rightarrow Y \;,\; Y \rightarrow Z\} \in F \;\Rightarrow\; X \rightarrow Z \in F^+$$

Ahora nos podemos plantear la siguiente pregunta ¿hay algún conjunto pequeño de reglas de inferencia que permitan deducir F^+ a partir de F? La respuesta es afirmativa (ecuación 1.15): dicho conjunto lo forman los denominados *Axiomas de Armstrong*, introducidos por primera vez por dicho autor en [4] y extendidas al caso de *dependencias multivaluadas* por Beeri, Fagin y Howard en [7]. Para describir formalmente dichos axiomas preferimos introducir una notación distinta a la utilizada por los propios autores y diferente también a la que aparece en el libro de Ullman [78], definiendo el siguiente conjunto:

$$F^{+A} = F \cup \{I \text{ tal que } I \text{ se obtiene aplicando los Ax. Armstrong al conjunto } F\}$$

<u>**Axiomas de Armstrong:**</u>

Reflexividad . $Y \subseteq X \subseteq REL \Rightarrow X \rightarrow Y \in F^{+A}$

Aumento . Si $X \rightarrow Y \in F^{+A}$ y $Z \subseteq REL \Rightarrow XZ \rightarrow YZ \in F^{+A}$

Transitividad . Si $X \rightarrow Y \in F^{+A}$ y $Y \rightarrow Z \in F^{+A} \Rightarrow X \rightarrow Z \in F^{+A}$

Puede demostrarse fácilmente que dichos axiomas son *consistentes*[1], es decir, que al utilizarlos, no puede deducirse a partir de F una dependencia falsa. En términos de nuestra notación:

$$F^{+A} \subseteq F^{+}$$

Además, dichos axiomas son *completos* en el sentido de que, tal y como habíamos anunciado, son suficientes para obtener todas las dependencias derivadas lógicamente de F. En términos de nuestra notación:

$$F^{+A} \supseteq F^{+}$$

por lo que deduciríamos finalmente que:

$$F^{+A} = F^{+} \tag{1.15}$$

En definitiva, la ecuación 1.15 nos dice que para deducir todas las dependencias derivadas lógicamente de un conjunto F, basta aplicar los axiomas de Armstrong.

Ahora bien, retomando la primera pregunta que nos planteábamos, ¿cómo se construye dicho conjunto F^{+} en la práctica? Hay un problema de eficiencia, pues el tiempo requerido para calcular F^{+} resulta excesivo. Por lo tanto, en vez de computar F^{+}, nos conformamos con calcular el conjunto de todos los posibles atributos consecuentes para un conjunto dado X de atributos antecedentes. A dicho conjunto lo denotaremos por X_F^{+} y es la **clausura de X respecto** F. Obviamente, según 1.15, para deducir X_F^{+} basta aplicar los axiomas de Armstrong, es decir:

$$X_F^{+} = \{C \in REL \text{ tal que } X \rightarrow C \in F^{+}\} = \{C \in REL \text{ tal que } X \rightarrow C \in F^{+A}\}$$

Es inmediato demostrar que

$$X \rightarrow Y \in F^{+} \Leftrightarrow Y \subseteq X_F^{+}$$

[1]El término en inglés es *sound*

por lo que el cómputo de X_F^+ devuelve, de hecho, todas las posibles dependencias derivadas lógicamente de F y con atributo antecedente dado por el conjunto X. Un algoritmo para calcular X_F^+ es el dado en la figura 1.2 (Bernstein [8]):

Algoritmo: Cómputo de X_F^+

Entrada:	REL, F, X
Salida:	X_F^+
Método:	Cálculo de la siguiente sucesión:
	1. $X^0 = X$
	2. $X^{i+1} = X^i \cup \{C \in REL$ tal que
	$\exists\, Y \rightarrow Z \in F$ con $C \in Z,\ Y \subseteq X^i\}$
	Ultima iteración: $X^{n+1} = X^n = X_F^+$

Figura 1.2. Algoritmo para calcular la clausura de X

Puede demostrarse que dicho algoritmo calcula, efectivamente la clausura de X respecto F (ver [78, 8]). Lo único que nos interesa de dicha demostración es enfatizar que la única herramienta necesaria es la consistencia y completitud de los axiomas de Armstrong.

1.2.3 Normalización de un Esquema Relacional

En este apartado vamos a plantear la necesidad de trabajar con relaciones que satisfazgan una serie de restricciones (englobadas bajo el término genérico de *forma normal*).

En el caso de que no exista físicamente la base de datos, sino que el estudio es previo al almacenamiento de datos, entonces la teoría de normalización nos dirá que el diseño de los esquemas relacionales ha de verificar las restricciones dadas por las formas normales.

En caso contrario, estamos ante una base de datos existente y si no verifica las formas normales porque el esquema no fué bien concebido en su día, entonces se presentan importantes problemas, tal y como vimos en el ejemplo 1.2.1. La forma de resolverlos será *descomponiendo* la base respecto a una partición del esquema original, pero este punto se abordará posteriormente.

Empecemos revisando el concepto de forma normal. Las restricciones dadas por las formas normales pueden englobarse en dos grandes apartados:

1. Restricciones sobre los datos que puedan aparecer en una tupla: dependen de la semántica de los elementos de los dominios.

2. Restricciones sobre las dependencias funcionales que puedan aparecer: dependen únicamente de la igualdad o no entre ciertos valores de dominio.

El primer punto corresponderá al concepto de primera forma normal, y el segundo al planteamiento de las formas normales segunda, tercera y Boyce Codd.

◇ *Primera Forma Normal*

La primera forma normal (introducida por Codd en [21]) establece la restricción de que los datos no pueden ser *subconjuntos*, sino únicamente valores atómicos (alfanuméricos o numéricos). Hemos de entender *subconjunto* o bien como una colección de valores que representan posibles asignaciones <u>no</u> excluyentes, como por ejemplo `Idioma Hablado = Inglés y Francés`, o bien como una colección de posibles asignaciones mutuamente excluyentes, como por ejemplo `Altura = 175 ó 176`. Esta imposición es obligada para mantener la interpretación semántica de una tupla como la descripción de una entidad.

Definición 1.2.6 *Una relación r se dice que verifica la* **primera forma normal** *(*1FN*) si y solo si:*

$$A^r(t) \in D_A \quad (A^r(t) \not\subseteq D_A) \quad \forall\, t \in r \quad \forall\, A \in REL(r)$$

dónde D_A es un dominio de valores alfanuméricos o numéricos.

<u>**Restricción**</u>. *Toda relación en una b.d.r. ha de estar en 1FN*

Por lo tanto, si una relación no verifica la anterior restricción, es porque habrá una tupla con algún valor dado por un subconjunto. Entonces, habrá que desdoblar dicha tupla en dos.

Ejemplo 1.2.7 . La relación

Nombre	Idioma	Altura
Pedro	{Inglés , Francés}	180
Juan	Castellano	182

no está en 1FN, por lo que debe transformarse en la siguiente:

Nombre	Idioma	Altura
Pedro	Inglés	180
Pedro	Francés	180
Juan	Castellano	182

■

◇ *Segunda, Tercera y Forma Normal de Boyce Codd*

Para definir el resto de las formas normales, nos apoyamos en el concepto de dependencia funcional y de llave candidata. Empezaremos revisando la forma normal más restrictiva, a saber, la forma normal de Boyce y Codd, introducida por primera vez por Heath en [48] (bajo el nombre de tercera forma normal) y por Codd en [23]. Posteriormente definiremos la tercera y segunda forma normal, introducidas por Codd en [21, 22].

Definición 1.2.8 *Un esquema relacional REL se dice que está en* **forma normal de Boyce Codd** *(*FNBC*) si siempre que REL satisfazga una d.f $X \to A$ con $A \notin X$, entonces X contiene a alguna llave candidata.*

Definición 1.2.9 *Un esquema relacional REL se dice que está en* **tercera forma normal** *(*3FN*) si siempre que REL satisfazga una d.f $X \to A$ con $A \notin X$, entonces o bien A es primo, o bien X contiene a alguna llave candidata.*

Definición 1.2.10 *Un esquema relacional REL se dice que está en* **segunda forma normal** *(*2FN*) si siempre que REL satisfazga una d.f. $X \to A$ con $A \notin X$, entonces o bien A es primo, o bien X no está incluido estrictamente en ninguna llave candidata.*

Los conceptos de 2FND, 3FND y FNBC van siendo más restrictivos, por lo que, por ejemplo, toda relación que esté en FNBC está también en 3FN y 2FN.

Ejemplo 1.2.11 . Supongamos una relación r que no se encuentra en segunda forma normal (2FN). Entonces, según la definición 1.2.10, debe existir alguna d.f. de la forma $X \to A$ con A atributo no primo y X contenido propiamente en alguna llave candidata C. Por ejemplo, el esquema de la figura 1.3 no está en 2FN.

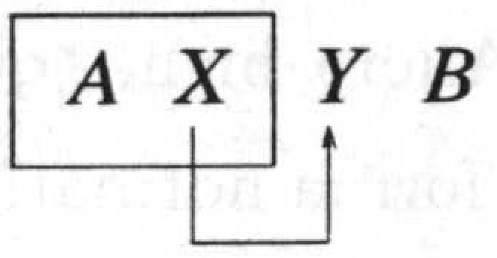

Figura 1.3. Violación de una 2FN, 3FN y FNBC

Supongamos ahora una relación r que no se encuentra en tercera forma normal (3FN). Entonces, según la definición 1.2.9, debe existir alguna d.f.d.r de la forma $X \to A$ con A atributo no primo y X no contenido en una llave candidata. Por ejemplo, el esquema de la figura 1.4 no está en 3FN.

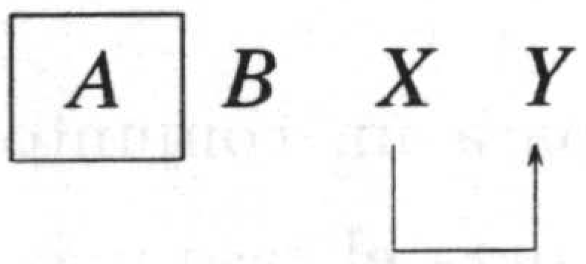

Figura 1.4. Violación de una 3FN, FNBC

Consideremos ahora una relación r que no se encuentra en FNBC. Entonces, según la definición 1.2.8, debe existir una d.f. del tipo $X \to A$ dónde X no contiene a ninguna llave de $REL(r)$. En el caso de que A no fuese primo, entonces la relación tampoco está en tercera forma normal. Así que supongamos que A es primo, es decir, que r ($REL(r)$) no satisface una FNBC *propia*. En general, una relación no está en FNBC cuando presenta una d.f. con antecedente que no es una llave candidata, y con consecuente un subconjunto de una llave candidata. Por ejemplo, los esquemas de la figura 1.5 (la llave primaria va en trazo continuo y la otra llave candidata en línea quebrada) no están en FNBC.

■

Figura 1.5. Violación de una FNBC

◇ *Descomposición sin Pérdidas de una Relación: Teorema de Heath*

Hemos introducido las formas normales a las que se debería llegar al diseñar un esquema relacional, antes de introducir datos. Ahora bien, ¿qué debemos hacer si ya existía una relación, y no está en una determinada forma normal? habrá que descomponerla en otras relaciones (utilizando el operador de proyección dado en la ecuación 1.10), de forma que:

1. Las nuevas relaciones así obtenidas satisfazgan las formas normales.
2. Podamos recuperar toda la información que había en la relación original, es decir, que la *descomposición no tenga pérdidas.* Esto se comprobará a través de la reunión natural de las relaciones proyectadas.

En general, el estudio se realiza para un conjunto arbitrario de dependencias. Sin embargo, nosotros vamos a detallarlo para el caso más simple de una única dependencia, ya que será el punto de partida en nuestra extensión difusa. Necesitamos introducir los siguientes conceptos:

Definición 1.2.12 *Sea REL un esquema relacional. Diremos que $\rho = \rho_1\rho_2$ es una* **descomposición** *de REL si y solo si:*

$$\rho_1 \cup \rho_2 = REL \quad , \quad \rho_1 \cap \rho_2 \neq \emptyset$$

Nos planteamos ahora cómo se lleva a cabo la descomposición de una relación dada. El operador natural a utilizar es el de proyección, de forma que r se descompondría en las relaciones $\Pi_{\rho_1}(r)$ y $\Pi_{\rho_2}(r)$. Una vez descompuesta la relación r, nos preguntamos cómo reconstruir la información original que había en r. El operador natural es el de reunión, por lo que deberíamos realizar:

$$\Pi_{\rho_1}(r) \bowtie_{\rho_1 \cap \rho_2} \Pi_{\rho_2}(r)$$

A la anterior relación la denotaremos por $m_\rho(r)$. Veamos qué relación hay entre r y $m_\rho(r)$. Un resultado inmediato de demostrar es que

$$r \subseteq m_\rho(r) \tag{1.16}$$

dónde la inclusión entre dichas relaciones viene dada por el operador definido en la ecuación 1.3. Esto nos dice que en $m_\rho(r)$ aparecen las mismas tuplas que había en r. El problema que se presenta es que pueden aparecer más tuplas (*tuplas espúreas*) que no estaban originalmente en r, por lo que realmente estamos perdiendo información (pueden verse ejemplos de este hecho en [78, 34]). En el caso de que no aparezcan dichas tuplas tenemos la siguiente definición:

Definición 1.2.13 *Sea REL un esquema relacional verificando un conjunto de dependencias F. Diremos que una descomposición $\rho = \rho_1\rho_2$ es una* **descomposición sin pérdidas respecto** *F si y solo si, para toda instancia r de REL (verificando F) se satisface lo siguiente:*

$$r = m_\rho(r) \tag{1.17}$$

La definición 1.2.13 se realiza respecto a un conjunto de dependencias F. Este hecho es necesario ya que se ha de definir la descomposición de un esquema relacional REL, y por tanto se ha de considerar cualquier instancia r de REL. Así pues, dicha instancia ha de ser legal en el sentido de que satisfazga las restricciones dadas por las dependencias funcionales presentes en dicho esquema.

Claro está, la pregunta que nos planteamos ahora es ¿cómo conseguir una descomposición sin pérdidas? El siguiente teorema[1] garantiza que, si se aislan en un esquema (llamémosle ρ_2) los atributos presentes en una d.f, entonces la descomposición es sin pérdidas. Este resultado fué generalizado por Aho, Beeri y Ullman en [1] en un algoritmo (llamado ABU como acrónimo de los autores) que trataba con un conjunto de más de dos dependencias. No incluimos el último resultado más general, ya que sólo podremos extender al caso difuso el teorema de Heath.

[1] La condición suficiente fue demostrada por Heath en [48] y posteriormente por Delobel [38], mientras que la condición necesaria la demostró Rissanen en [70]

Teorema 1.2.14 (Heath: fundamental de descomposición sin pérdidas) *Supongamos un esquema relacional REL, verificando un conjunto de dependencias F. Sea $\rho = \rho_1\rho_2$ una descomposición de REL. Entonces, ρ es una descomposición sin pérdidas para REL respecto F si y solo si se satisface alguna de las siguientes condiciones:*

$$\rho_1 \cap \rho_2 \rightarrow \rho_1 - \rho_2 \quad , \quad \rho_1 \cap \rho_2 \rightarrow \rho_2 - \rho_1$$

◇ *Descomposiciones sin Pérdida de Dependencias*

Hemos anotado que una descomposición ha de verificar que sea sin pérdidas; sin embargo, esta no es la única propiedad deseable. También ha de poderse recuperar la información de cuales eran las dependencias funcionales que estaban presentes en el esquema original. En pocas palabras, dada una descomposición $\rho = (\rho_i)_{i=1\ldots n}$, y llamando F_i al conjunto de d.f $X \rightarrow Y$ tal que $XY \subseteq \rho_i$, entonces ha de verificarse que:

$$\bigcup_i F_i^+ = F^+$$

Puede demostrarse (se recomienda seguir el texto de Ullman [78]) que siempre se puede encontrar una descomposición sin pérdidas de dependencias, de forma que las relaciones proyectadas $\Pi_{\rho_i}(r)$ estén en 3FN. Sin embargo, no puede extenderse este resultado al caso general de FNBC: un ejemplo de este hecho puede verse en la figura 1.5, dónde el esquema ZYX con llave ZY y d.f $X \rightarrow Y$ ($F = \{ZY \rightarrow X \ , \ X \rightarrow Y\}$), se descompone en $\rho_1 = ZX$ y $\rho_2 = XY$. Es inmediato comprobar que $F_1^+ = \emptyset$, $F_2^+ = X \rightarrow Y$ y por lo tanto $F_1^+ \cup F_2^+ = X \rightarrow Y \neq F^+$.

Bernstein da un algoritmo en [8] para obtener dicha descomposición en relaciones en 3FN y sin pérdidas de dependencias. Brevemente comentaremos que la idea es aislar las dependencias transitivas $X \rightarrow Y$ en esquemas independientes. Para fijar ideas, dado un esquema XYZ con $X \rightarrow Y$, entonces el algoritmo de Bernstein asegura que la descomposición XY, XZ es sin pérdidas en dependencias, mientras que el teorema de Heath (en general el algoritmo ABU [1]) asegura que la descomposición es sin pérdidas en cuanto que no aparecen tuplas espúreas.

Podemos concluir de este punto que siempre hemos de diseñar o descomponer esquemas hasta obtener relaciones en tercera forma normal. Sin embargo, no siempre será deseable llegar hasta una FNBC, debido a las posibles pérdidas de dependencias.

1.2.4 Valores Nulos

En este apartado vamos a ver las aproximaciones que diversos autores plantean para el tratamiento de los valores nulos. Revisaremos brevemente los estudios relativos a las posibles interpretaciones del valor `nulo` y posteriormente analizaremos los problemas que pueden plantear a la hora de definir una dependencia funcional, y por tanto sus repercusiones en el diseño.

◇ *Tratamiento de la Información Incompleta*

Es usual que la información de la que se dispone en un momento dado, no esté completamente especificada. En estas situaciones, no se pueden incluir valores de este tipo sin que violemos la primera forma normal, por lo que, en principio, un s.b.d.r no admitiría este tipo de información. Sin embargo, los sistemas clásicos utilizan un valor especial, a saber, el valor `nulo` para manejar algunos tipos especiales de informaciones no especificadas completamente. En general, la falta de especifidad puede venir originada por distintos motivos (puede consultarse el trabajo de Demolombe [39]):

- Información disyuntiva no exclusiva. Por ejemplo, la información podría venir dada por: $A \vee B$ es verdad. Este aserto implica que puede ser verdad A, o lo puede ser B, o ambos.

- Falta de especifidad debida a la validez o no de la información. Por ejemplo, si se almacena A, pero no se está seguro de si realmente es una información verdadera o falsa.

- Posible inconsistencia al almacenar distintos datos que, en un determinado momento, pueden producir una contradicción lógica.

- Vaguedad, entendiéndola como información disyuntiva exclusiva. Por ejemplo, la información podría venir dada por: $A \vee B$ es verdad. Este aserto implica, en este caso, que puede ser verdad A, o lo puede ser B, pero no ambos simultaneamente.

De todos estos, la teoría relacional clásica sólo ha abordado la resolución del último problema, y además en un caso muy particular, a saber cuando la vaguedad afecta a todo el dominio. Este tipo de información corresponde al valor `desconocido` y se utilizan los valores nulos para representar dicha información. Por ejemplo, Codd establece en [25] (ver también las aproximaciones de Biskup [9, 10] y Yue [92]) una lógica trivaluada para tratar con estos datos, considerando tres grados de verdad: *verdad, falso, posible.* Posteriormente ([26]) introdujo una lógica 4-valuada para distinguir otro tipo de valor nulo, a saber `valor o propiedad no aplicable`. Obsérvese que este último tipo de dato no corresponde a una falta de especifidad en la información.

Date, en [33, 31, 32, 34], realiza una crítica a la aproximación de Codd, argumentando que, de hecho, cabría considerar muchos *tipos* de nulos, como por ejemplo *valor no existe*, *valor indefinido*, etc. Propone por tanto, utilizar únicamente el tipo `nulo` hasta que se desarrolle más extensamente estudios teóricos al efecto. Este valor nulo es considerado como una valor destacado del propio dominio y *asignado por defecto.*

Nosotros consideramos que es necesario distinguir, tal y como realiza Codd, al menos entre los dos tipos anteriores de valores nulos. La razón que argumentamos es que no podemos utilizar un valor nulo genérico para representar cualquier tipo de falta de especifidad en la información. En lo que sigue, consideraremos propiamente como valor `nulo`, el valor `inaplicable`. El otro tipo de nulo, a saber, `desconocido`, será un caso particular de un *conjunto difuso*, herramienta que introduciremos en el siguiente capítulo y mediante la cual, podremos representar datos imprecisos de los tipos 1 y cuatro anteriores, es decir información disyuntiva exclusiva y no exclusiva.[1]

[1]Los otros tres tipos de informaciones incompletamente especificadas no son abordadas en esta memoria

◇ *Implicaciones en el Diseño*

Si bien podemos utilizar ciertos valores destacados (`nulo`) para representar en la base de datos algunos tipos de informaciones incompletas, nos planteamos si éstos pueden aparecer como valores de atributos antecedentes o consecuentes de una d.f. El problema estriba en interpretar qué significa que dos elementos nulos (o un valor de dominio y un nulo) sean iguales o distintos, para poder así extender la definición de d.f. Brevemente comentamos que los problemas se derivan porque, en el proceso de comprobación de existencia de una dependencia funcional, cuando hay que comparar dos valores de dos tuplas distintas, no es posible dar una interpretación unívoca de los siguientes condicionales:

- *si* $X(t) =$ `inaplicable`
- *si* $X(t) =$ `desconocido`
- *entonces debe ser* $Y(t) =$ `inaplicable`
- *entonces debe ser* $Y(t) =$ `desconocido`

Cuando la única interpretación considerada del valor nulo es la de `desconocido`, Vassiliou [83] introdujo sendas definiciones de dependencia funcional (*débil* y *fuerte*) considerando algunos casos especiales en los que se permite incluir valores nulos en determinadas tuplas de atributos antecedentes o consecuentes. En el caso de que se imponga la existencia de una d.f. usual, se plantea la posibilidad de sustituir los valores nulos por aquellos valores que no rompan la d.f. Por ejemplo, si se obliga a que sea $X \rightarrow Y$ con valores:

X	Y
x	`desconocido`
x	y

debería sustituirse el valor `desconocido` por y.

En cualquier caso, y en esto sí están de acuerdo todos los autores, es que, parafraseando a Elmasri y Navathe [43], *no existe una teoría de diseño relacional totalmente satisfactoria, en cuanto se incluyan valores nulos.*

Es muy importante destacar que, en la extensión difusa que nosotros plantearemos, este aserto seguirá siendo cierto. Es decir, nuestro trabajo no pretende resolver el problema del diseño de una base de datos relacional cuando hay valores nulos, sino resolver el problema del diseño de una base de datos relacional cuando hay presentes valores difusos. Pero dichos valores difusos no podrán ser cualesquiera: por ejemplo no se permitirán, al igual que el caso clásico, que sea el valor `desconocido`.

Capítulo 2

Modelos Relacionales Difusos

Vimos en el capítulo anterior, que el modelo de base de datos relacional clásico sólo ha sido desarrollado para incorporar casos muy particulares de informaciones no especificadas completamente, por lo que se hace necesario extender el tratamiento a este tipo de informaciones. Como ya comentamos al final del primer capítulo, nosotros nos centraremos en el caso de informaciones expresadas *vagamente.*

En el primer apartado de este capítulo, introduciremos una de las principales herramientas para el tratamiento de este tipo de información, a saber la teoría de conjuntos difusos y la teoría de la posibilidad, elaborada principalmente por Zadeh ([93, 98]) entre los años sesenta y setenta. Los conceptos básicos aquí introducidos, forman, junto con el modelo relacional clásico, los dos pilares sobre los que se desarrolla nuestro trabajo.

Posteriormente, revisaremos las extensiones difusas al modelo relacional clásico. Estas fueron planteadas como una simple extensión del sistema de almacenamiento de datos y de definición de operadores del álgebra para realizar preguntas imprecisas a la base de datos, pero no desde una perspectiva del diseño de la base. Nos centraremos en el que llamaremos *modelo posibilístico* por ser éste el de mayor generalidad. Este será el modelo que adoptaremos para la representación de los datos. Analizaremos los inconvenientes que conllevan las distintas aproximaciones difusas para definir los operadores básicos del álgebra relacional, por lo que se justificará la necesidad de adoptar otros operadores que permitan elaborar una teoría de diseño para una base de datos relacional difusa. La solución que proponemos para estos problemas se abordarán en los capítulos cuarto (para la proyección difusa) y quinto (para la reunión difusa). En este capítulo, nos limitaremos a plantearlos.

En la cuarta sección, introduciremos las extensiones difusas planteadas en la definición de dependencia funcional, y plantearemos algunas propiedades importantes que toda extensión debiese verificar. Veremos que, aunque las definiciones de dependencia son difusas, sólo se han desarrollado teorías de diseño clásicas, en el sentido de que los conceptos básicos utilizados como la elimininación de redundancia, o el operador de reunión, no han sido concebidos con un planteamiento difuso.

Finalmente, introduciremos una nueva definición de dependencia funcional difusa coherente con las propiedades genéricas establecidas en la sección cuarta. Veremos sus propiedades, como por ejemplo, la completitud de la versión difusa de los axiomas de Armstrong, y analizaremos la relación de nuestra definición con las aproximaciones existentes.

2.1 La Teoría de Conjuntos Difusos y la Teoría de la Posibilidad

2.1.1 Conjuntos Difusos

Como ya hemos comentado anteriormente, vamos a revisar una de las principales herramientas para el tratamiento de información imprecisa: la teoría de conjuntos difusos. En este punto, el término imprecisión hace referencia a una falta de especificidad en los datos, derivada de la imposibilidad de especificar una única posible asignación a una variable representando cierta propiedad. Por ejemplo, puede tenerse cierto conocimiento de la altura de cierta persona, sin que se pueda especificar un valor concreto. Así pues, podríamos afirmar que `Juan es alto` pero no que `Juan mide 185.37cm`. En este ejemplo, `alto` es un *conjunto difuso* (ver figura 2.1).

Hemos de enfatizar que no abordaremos el tratamiento de información cuya falta de especifidad venga dada por cualquier otra naturaleza que no sea la imprecisión. Por ejemplo, no trataremos con información *incierta*, ya provenga la incertidumbre de una distribución de probabilidad o de un conjunto de medidas difusas duales en general [41].

Pasamos a definir en primer lugar, el concepto de *conjunto* difuso (Zadeh [93]), así como los operadores habituales. Posteriormente introduciremos las *relaciones* y los *números* difusos. Unicamente pretendemos fijar la notación y la terminología que seguiremos a lo largo de esta memoria, por lo que para un estudio detallado se recomienda consultar el artículo original de Zadeh [93] o los textos de Dubois y Prade [40, 41].

◇ *El Concepto de Conjunto Difuso*

Definición 2.1.1 *Un* **conjunto difuso** F *es un conjunto de pares*

$$F \equiv \{x, \mu_F(x)\}_{x \in D}$$

dónde μ_F *es la* **función de pertenencia** *de* F*, con rango e imagen dados por:*

$$\mu_F \; : \; D \longrightarrow [0,1]$$

con D un conjunto clásico.

Obviamente, esta definición engloba como caso particular la de un conjunto clásico. Representaremos los conjuntos difusos a través de su función de pertenencia, como puede apreciarse en la figura 2.1.

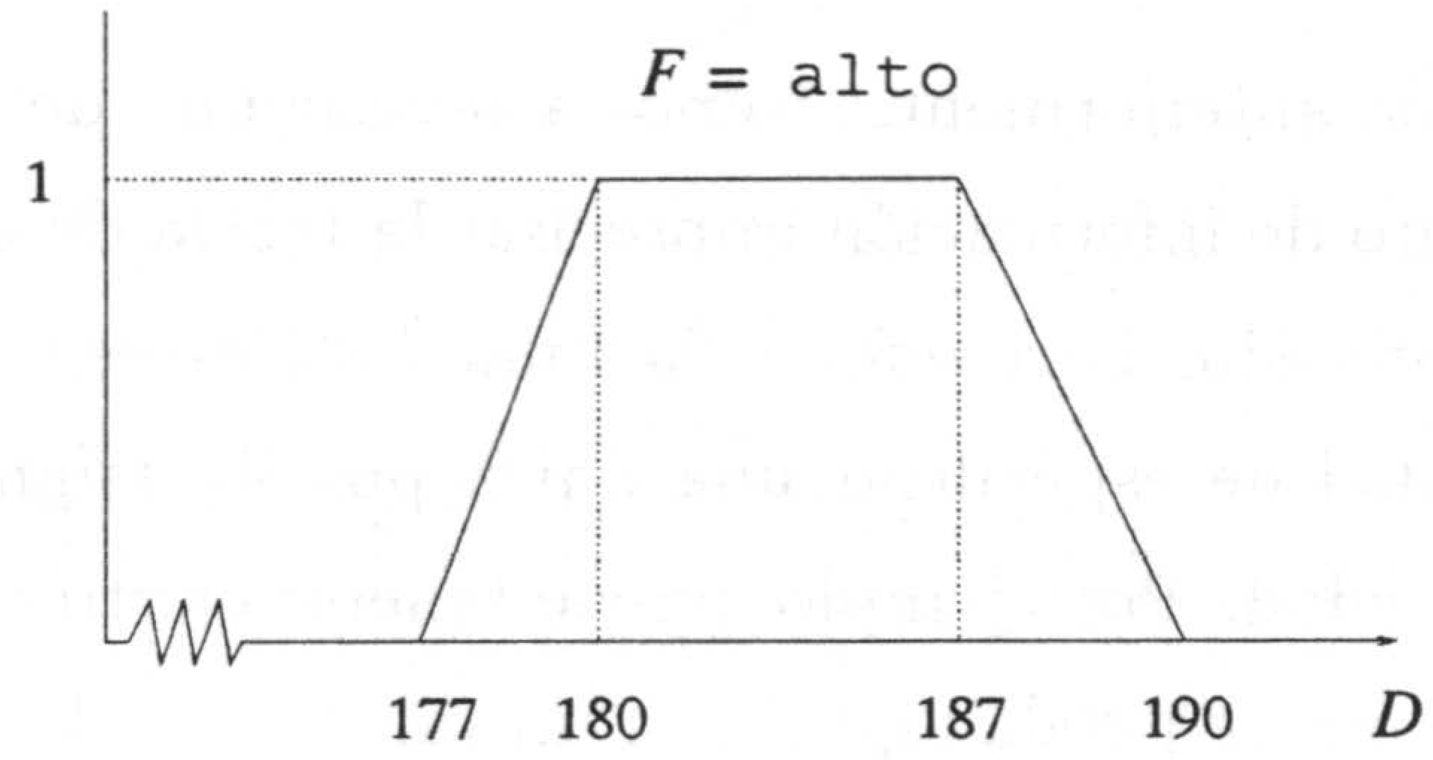

Figura 2.1. Conjunto difuso

Pasemos ahora a introducir brevemente otros conceptos relacionados con los conjuntos difusos. Para más detalle sobre cualquiera de estos puntos, se recomienda ver [93, 40].

* **Universo de discurso**. Es el conjunto clásico D sobre el que está definida la función de pertenencia.

* Un valor **crisp** es cualquier elemento del universo de discurso.

* Cuando D es finito (numerable en general, $D = \{x_1 \ldots x_n\}$) utilizaremos la siguiente notación:

$$F \equiv \{x_1/\mu_F(x_1) + x_2/\mu_F(x_2) + \ldots + x_n/\mu_F(x_n)\}$$

* Al conjunto (clásico) de todos los posibles conjuntos difusos definidos sobre un universo de discurso D, lo denotaremos por $\widetilde{\mathcal{P}}\ (D)$, por lo que tendrá sentido la expresión $F\ \in\widetilde{\mathcal{P}}\ (D)$. Como un conjunto clásico es un caso particular de un conjunto difuso, se verifica que:

$$\mathcal{P}(D) \subset \widetilde{\mathcal{P}}\ (D)$$

* El **soporte** de un conjunto difuso ($sop(F)$), es el conjunto de valores del universo con grado de pertenencia estrictamente positivo:

$$sop(F) = \{x \in D \ \text{ tal que } \mu_F(x) > 0\}$$

* El **núcleo** de un conjunto difuso ($ker(F)$), es el conjunto de valores del universo con grado de pertenencia igual a 1:

$$ker(F) = \{x \in D \text{ tal que } \mu_F(x) = 1\}$$

Obviamente se verifica que $sop(F) \subseteq ker(F)$.

* Diremos que un conjunto difuso está **normalizado** si y solo si

$$ker(F) \neq \emptyset \Leftrightarrow \exists\, x \in D \text{ tal que } \mu_F(x) = 1$$

Señalemos que el concepto de difuso normalizado no tiene nada que ver con la definición de relación normalizada, introducida en el primer capítulo.

* Se define el **alfa corte** de un conjunto difuso como sigue:

$$F_\alpha = \{x \in D \text{ tal que } \mu_F(x) \geq \alpha\}$$

* Dados dos conjuntos difusos F y G, definimos la **intersección** ($F \wedge G$) y la **unión** ($F \vee G$), dando sus funciones de pertenencia:

$$\mu_{F \wedge G}(x) = \min\{\mu_F(x), \mu_G(x)\}$$

$$\mu_{F \vee G}(x) = \max\{\mu_F(x), \mu_G(x)\}$$

Se utilizará indistintamente el símbolo $\vee$ ($\wedge$) para denotar al operador *máximo* (*mínimo*), como para referenciar la unión (intersección) difusa, por lo que las anteriores expresiones también las escribiremos como sigue:

$$\mu_{F \wedge G}(x) = \mu_F(x) \wedge \mu_G(x)$$

$$\mu_{F \vee G}(x) = \mu_F(x) \vee \mu_G(x)$$

* Una **etiqueta lingüística** es un conjunto difuso, al que se le asigna un determinado identificador. Por ejemplo, `alto` es una etiqueta lingüística, mientras que el intervalo [20, 28] (que puede interpretarse como un conjunto difuso) no lo sería. Este concepto fue desarrollado por Zadeh en [95, 96, 97].

◇ *Relaciones Difusas*

Definición 2.1.2 *R es una relación difusa sobre $D_1, \ldots D_n$ si y solo si:*

$$R \in \widetilde{\mathcal{P}} (D_1 \times \ldots \times D_n)$$

En definitiva, una **relación difusa** es un conjunto difuso definido sobre un producto cartesiano de universos de discurso. Obsérvese que, evidentemente:

$$\widetilde{\mathcal{P}} (D_1 \times \ldots \times D_n) \neq \widetilde{\mathcal{P}} (D_1) \times \ldots \times \widetilde{\mathcal{P}} (D_n)$$

Nos interesará trabajar con casos particulares de relaciones difusas, como los siguientes:

Definición 2.1.3 *(Zadeh [94], Rundensteiner [71]) Diremos que una relación difusa R, definida sobre $D \times D$, es una* **relación de semejanza** *si y solo si verifica las siguientes propiedades:*

- *Reflexiva:* $R(x, x) = 1 \quad \forall\, x \in D$

- *Simétrica:* $R(x, y) = R(y, x) \quad \forall\, x, y \in D$

Definición 2.1.4 *(Zadeh [94]). Se dice que una relación difusa sim definida sobre $D \times D$, es una* **relación de similitud** *si y solo si es una relación de semejanza y verifica además la propiedad Max–Min Transitiva, a saber:*

$$sim(x, z) \geq \max_{y \in D}\{\min(sim(x, y), sim(y, z))\} \tag{2.1}$$

Las relaciones de similitud son una extensión del concepto clásico de relación de equivalencia. De hecho, los α-cortes de una relación -difusa- de similitud, son relaciones -crisp- de equivalencia.

Potoczny demuestra en [64] que únicamente es necesario elicitar la similitud entre ciertos valores, viniendo determinados el resto, por la definición de max-min transitividad.

◇ *Números Difusos*

Cuando el universo de discurso es un subconjunto de la recta real $\mathbb{R}$, un conjunto difuso representa una *cantidad difuso*. Para evitar funciones de pertenencia anómalas, y para facilitar la operabilidad, se suelen añadir ciertas restricciones adicionales, obteniéndose entonces la definición de *número difuso*:

Definición 2.1.5 *Un* **número difuso** *F es una cantidad difusa $\left(F \in \tilde{\mathcal{P}}\ (\mathbb{R})\right)$ verificando las siguientes propiedades:*

i) La función de pertenencia es convexa, es decir:

$$\forall\ x, y \in \mathbb{R}, \quad \forall\ z \in [x, y] \;\; \mu_F(z) \geq \min\{\mu_F(x), \mu_F(y)\}$$

ii) μ_F es continua a la izquierda en todos los puntos. Por lo tanto, los α-cortes de F son intervalos cerrados de $\mathbb{R}$.

iii) El soporte de F está acotado:

$$\exists\ I, S \in \mathbb{R} \;\; \textit{tal que sop}(F) \subseteq [I, S]$$

Como consecuencia de esta propiedad y de ii)*, se deduce que todos los α-cortes de F son intervalos cerrados y acotados de $\mathbb{R}$, es decir, compactos.*

iv) F está normalizado:

$$\sup_x \mu_F(x) = 1$$

La forma más usual con la que se representan los números difusos, es a través de una función trapezoidal (ver Delgado et al [37]). En este caso, sólo necesitamos 4 parámetros para delimitar el conjunto difuso. Estos corresponden a los valores inferior y superior del núcleo y soporte (ver figura 2.2).

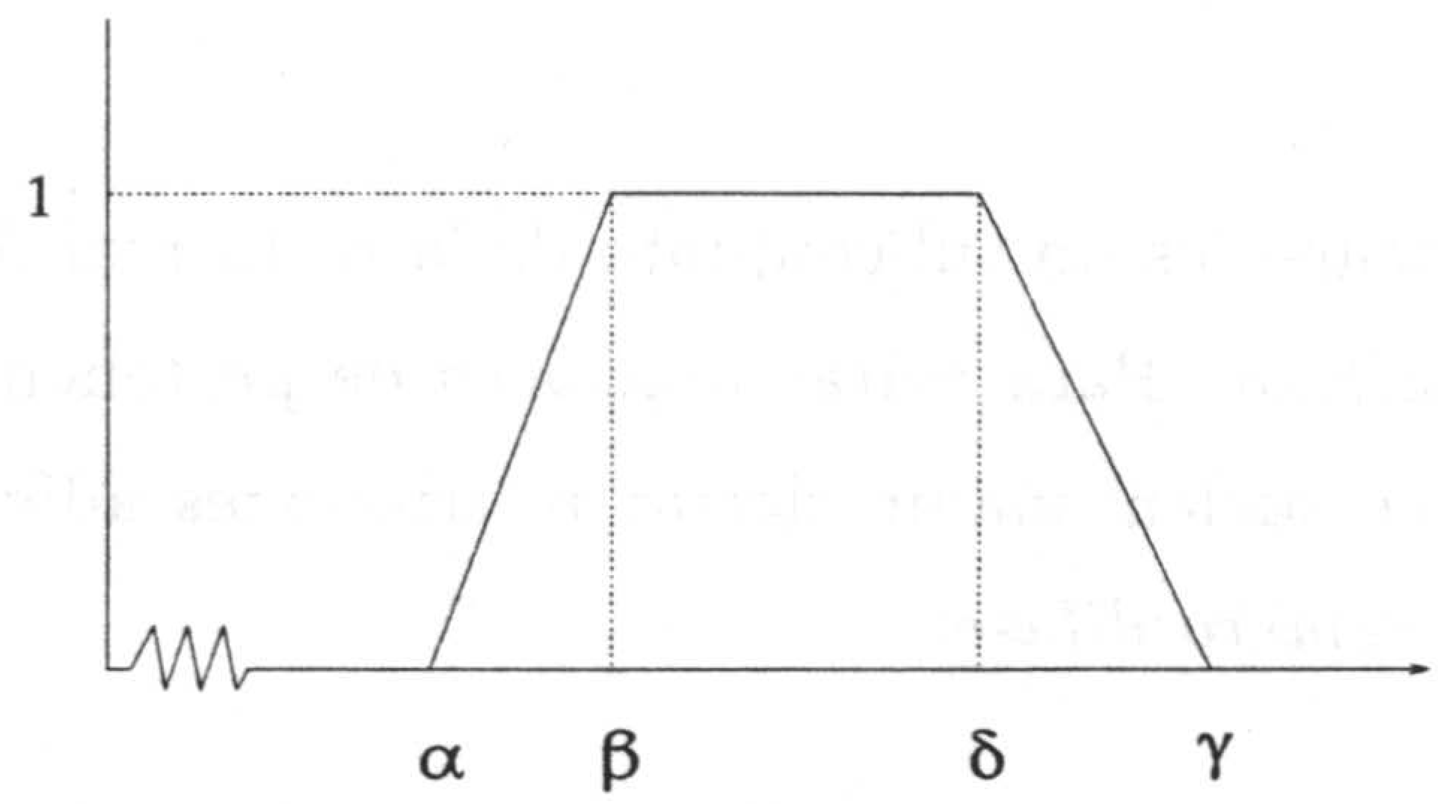

Figura 2.2. Conjunto difuso parametrizado con cuatro valores

2.1.2 Concepto de Distribución de Posibilidad

Una vez que hemos definido un conjunto difuso, pasamos a estudiar cómo se realiza formalmente el proceso de asignación de una propiedad dada por un conjunto difuso, a una variable. Para más detalle, se recomienda consultar el artículo original de Zadeh [98] y el texto de Dubois y Prade [41].

Consideremos una variable X que toma valores en el universo D y sea F un subconjunto difuso de D, $F \in \tilde{\mathcal{P}}(D)$. Entonces, la proposición:

$$"X \text{ es } F"$$

induce lo que se denomina una **distribución de posibilidad** sobre D, que es una función

$$\pi_X \; : \; D \longrightarrow [0,1]$$

de forma que el grado de concordancia de un elemento $d \in D$ con la propiedad dada por F viene dada por:

$$\pi_X(u) = \mu_F(u) \quad \forall\, d \in D \tag{2.2}$$

lo cual se nota usualmente por $\pi_X = F$. El difuso F actúa como una restricción difusa $(R(X))$ sobre los valores que pueden ser asociados a X.

Por ejemplo, suponer $X \equiv$ `altura de Juan`, $F \equiv$ `alto` (ver figura 2.1). Entonces, en la proposición `la altura de Juan es alto`, el difuso `alto` actúa como una restricción difusa sobre los valores que pueden ser asociados a la `altura de Juan`. Así pues, estaríamos diciendo que la altura de `Juan` (que es un único valor) podría ser perfectamente 180 cm, ya que éste es un valor con grado de pertenencia 1 al difuso alto, y que, por ejemplo, no mide 140 cm pues es un valor que no está en el soporte de `alto` (ver figura 2.3).

d	140	...	177	...	180	...	187	...	190
$\pi_{\texttt{altura de Juan}}(d)$	0	...	0	...	1	...	1	...	0

Figura 2.3. Distribución de posibilidad

Nota. El proceso anterior es reversible, en el sentido de que, partiendo de una definición de distribución de posibilidad de una variable X, se puede generar un conjunto difuso.

El concepto de distribución de posibilidad ha dado lugar a una fructífera teoría, muy útil en el manejo de la imprecisión (puede consultarse el texto básico de Dubois y Prade [41]) y que se basa en la cualificación de proposiciones a través de *medidas de posibilidad* (Π) y *necesidad* (N). Dichas medidas pueden definirse directamente a partir de una distribución de posibilidad, o pueden contemplarse como casos particulares de *medidas difusas* (introducidas por Sugeno en [76]).

2.1.3 Sistemas Relacionales Difusos de Almacenamiento de Datos y Extracción de Información

Una vez introducidos los conceptos básicos de la teoría de conjuntos difusos, tenemos que ver cómo han sido utilizado éstos, en la concepción de un sistema relacional que permita almacenar datos difusos y plantear preguntas imprecisas.

Históricamente, las primeras aproximaciones se limitaban a tratar preguntas planteadas de forma imprecisa, a una base de datos relacional clásica (sin datos difusos). En este sentido podemos destacar los trabajos de Kunii [53], Kacprzyk y Ziolkowski [50], Bosc et al [11, 13, 12], y recientemente Park et al [62].

No vamos a entrar en un análisis de estos trabajos pioneros, y nos centraremos en la revisión de modelos que incorporen la posibilidad de almacenar (y por supuesto tratar) datos difusos[1]. Englobaremos dichos modelos en dos grandes grupos, que llamaremos *modelo posibilístico*, y *modelo de unificación*, aunque nos centraremos en el primero pues constituirá, por su generalidad, el marco de trabajo en el que desarrollaremos nuestro estudio. No trataremos el llamado *modelo relacional simple* ó *modelo relacional puro* (Zadeh [99], Baldwin [5, 6]), ya que no permite representar datos imprecisos, por lo que resulta demasiado simple. Sólo comentaremos que la diferencia con el modelo relacional clásico estriba en que asocia a cada tupla un valor de pertenencia a la relación. En cualquier caso, Vila et al [85] definen un modelo lógico de base de datos relacional difusa, de forma que comprende como casos particulares todos los anteriores.

Puntualicemos finalmente que, en los últimos años, diversos autores han implementado sistemas de almacenamiento y consulta de datos difusos, utilizando los recursos y potencia de un sistema relacional clásico, como ORACLE, a través de su lenguaje de consultas SQL. En este sentido, caben destacar los trabajos de Bosc et al [13], Medina et al [58] y Nakajima et al [60].

Empezaremos revisando el modelo posibilístico y luego pasaremos a ver el modelo de unificación. Unicamente introduciremos los conceptos más generales sobre los que se asientan las principales aproximaciones, sin entrar a discutir los detalles que marcan las diferencias entre ellas. Seguiremos la misma notación que se estableció al principio del primer capítulo, en lo que concierne a la representación de atributos, relaciones, etc.

[1] Una revisión actual de gran parte de los trabajos sobre bases de datos relacionales difusas, puede encontrarse en [81, 82]

2.2 Modelo Posibilístico

Este modelo se caracteriza, en general, por la definición de un sistema de almacenamiento en tablas o relaciones con la posibilidad de incorporar datos difusos (distribuciones de posibilidad), así como la definición de los operadores necesarios para realizar preguntas imprecisas a la base de datos. Diversos autores han tratado este tipo de modelo, entre los que destacamos Umano [79, 81], Prade y Testemale [65, 41], Zemankova y Kandel [101, 101], Rundensteiner et al [71], Medina et al [58] y finalmente Vila et al [85] en el marco de bases de datos lógicas.

Veamos las características generales que definen al modelo posibilístico. Empezaremos viendo cómo se representa la vaguedad, y cómo se plantean preguntas a la base de datos. A continuación revisaremos los tratamientos que diversos autores han dado al problema de la redundancia y de la definición de los operadores del álgebra, analizando, en cada caso, los inconvenientes que cada aproximación conlleva.

2.2.1 Representación de la Vaguedad

Los **dominios básicos** permitidos son conjuntos discretos o continuos. Por ejemplo, un conjunto finito de valores alfanuméricos D_1={`rojo, marrón, negro, rubio`}, o bien un conjunto finito de números D_2={`15,16,17`}, o bien el intervalo unidad $D_3 = [0, 1]$, e.t.c. Además suele añadirse otro elemento adicional (por ejemplo Prade y Testemale [65] lo denominan e) para representar `propiedad no aplicable`

Los **valores de dominio** permitidos son, en general, distribuciones de posibilidad sobre los elementos de los dominios básicos. En particular, un valor de dominio permitido es un valor crisp, ya que éste es un caso degenerado de distribución de posibilidad. Utilizaremos la notación $\pi_{A(t)}$ para representar la distribución de posibilidad almacenada en la tupla t del atributo A, aunque por abuso de notación, también la llamaremos simplemente $A(t)$. Por ejemplo, utilizando los anteriores dominios, los siguientes valores estarían permitidos:

$$\{\texttt{rojo/1} + \texttt{marrón/0.8}\} \in \widetilde{\mathcal{P}}\ (D_1)$$

$$\{\texttt{aproximadamente15}\} \in \widetilde{\mathcal{P}}\ (D_2)$$

$$\{\texttt{Falso}\} \in \widetilde{\mathcal{P}}\ (D_3)$$

El caso particular de distribución de posibilidad $\{d/1 \quad \forall\, d \in D\}$, corresponde al valor **desconocido**, por lo que, tal y como adelantábamos en el primer capítulo, utilizando distribuciones de posibilidad no es necesario identificar un elemento adicional para representar dicho valor.

Prade y Testemale [65], incluyen en la distribución de posibilidad, el grado de pertenencia del valor e, para así poder incluir informaciones inciertas. Umano [79] y Rundesteiner et al [71] utilizan difusos de tipo 2 para representar dichas informaciones. Por comodidad en la representación, nosotros adoptaremos la primera aproximación.

Una **relación básica** corresponde a una relación usual ('crisp') sólo que se permiten valores de dominio difusos. Un posible ejemplo de una relación básica presente en una base de datos sería la dada en la figura 2.4:

Persona =

Nombre	Edad	Altura	Color-de-Pelo
Juan	16	180	Marrón
Jose	17	alto	Negro
Pedro	15	bajo	Rubio
Mario	15	185	Rojo/1 + Negro/0.8

Figura 2.4. Relación en un modelo posibilístico

Obsérvese que `alto` es una etiqueta lingüística que representa un difuso sobre el dominio de la `Altura`, es decir:

$$\texttt{alto} \in \widetilde{\mathcal{P}}\ (D_{\texttt{Altura}})$$

Sin embargo, el valor `alto`/1 + `muyalto`/.8 sería un difuso de tipo 2, y este no sería un valor de dominio válido.

2.2.2 Elementos Iniciales del Algebra Relacional Difusa. El Operador de Selección Generalizado

El operador de selección extendido o generalizado, es la base de partida para la realización de consultas en cualquier modelo difuso de base de datos ([79, 101, 65, 58]). En principio, no estamos interesados en el estudio del planteamiento de consultas. Sin embargo, hemos de introducir algunos conceptos fundamentales, para poder luego analizar operadores básicos del álgebra relacional como la proyección. Además, utilizaremos en la definición de dependencia funcional difusa (capítulo cuarto), una medida que es utilizada por Prade y Testemale en el contexto del planteamiento de consultas a la base, por lo que hemos creido conveniente incluir ahora un breve resumen. Seguimos el planteamiento genérico realizado por Prade y Testemale en [65].

Denotemos por $\sigma(r, \mathcal{C})$, al operador de selección generalizado, dónde r es la relación sobre la que se realiza la selección, y $\mathcal{C}$ es una condición, que, en general puede ser de la forma:

- $\mathcal{C} = A\Theta F$, con $A \in REL(r)$, Θ un comparador difuso del tipo `más o menos igual`, `mucho mayor que`, etc, y F es un difuso planteado en una pregunta. También puede extenderse para trabajar con varios atributos.

- $\mathcal{C} = A\Theta B$. En este caso $\mathcal{C}$, es una condición que han de satisfacer los valores de los atributos A y B.

Realmente, se utilizan dos medidas para hallar los grados de pertenencia a la condición $\mathcal{C}$. Por una parte se considera una *medida de posibilidad* ($\pi(\mathcal{C})$) y por otro lado, una *medida de necesidad* ($N(\mathcal{C})$). Por ejemplo, la medida de posibilidad de que el valor $A(t)$ del atributo A esté en relación Θ con el valor $B(t)$ del atributo B ($\mathcal{C}_t = A(t)\Theta B(t)$), viene dada por:

$$\pi(\mathcal{C}_t) \equiv \pi(A(t)\Theta B(t)) = \sup_{(d,d')\in D\times D} \min\{\mu_\Theta(d,d'), \pi_{(A(t),B(t))}(d,d')\}$$

dónde $\pi_{(A(t),B(t))}(d,d')$ es la distribución de posibilidad que restringe los valores que pueda tomar el par (A, B). En el caso de que los atributos sean no interactivos (es decir, que no haya ninguna restricción sobre $B(t)$ cuando A toma cualquier valor $A(t)$), entonces ([65]) la anterior expresión será igual a:

$$\pi(\mathcal{C}_t) \equiv \pi(A(t)\Theta B(t)) = \sup_{(d,d')\in D\times D} \min\{\mu_\Theta(d,d'), \pi_{A(t)}(d), \pi_{B(t)}(d')\} \tag{2.3}$$

que nos dice hasta qué punto algún valor del difuso $A(t)$ está en relación Θ con algún valor de $B(t)$. Para calcular la necesidad correspondiente a la misma condición $\mathcal{C}_t$, se computaría el valor siguiente (suponiendo atributos no interactivos):

$$N(\mathcal{C}_t) \equiv N(A(t)\Theta B(t)) = \inf_{(d,d')\in D\times D} \max\{\mu_\Theta(d,d'), 1-\pi_{A(t)}(d), 1-\pi_{B(t)}(d')\} \tag{2.4}$$

que nos dice hasta qué punto todos los valores del difuso $A(t)$ está en relación Θ con todos los valores de $B(t)$.

Una vez que se han calculado los valores $\pi(\mathcal{C}_t)$ y $N(\mathcal{C}_t)$ para todas las tuplas t, se forman dos relaciones difusas: una relación que llamaremos $\sigma\pi_{\mathcal{C}}(r)$ dónde cada tupla $t \in r$ aparece con el grado (de posibilidad) $\pi(\mathcal{C}_t)$ en el que satisface la condición $\mathcal{C}$, y otra relación $\sigma\, N_{\mathcal{C}}(r)$ en la que cada tupla $t \in r$ aparece con el grado (de necesidad) $N(\mathcal{C}_t)$ en el que satisface la misma condición.

2.2.3 Tratamiento de la Redundancia

Una vez que hemos visto los datos que son representables en el modelo posibilístico, vamos a analizar el concepto de redundancia, que es de crucial importancia para la concepción, tratamiento y diseño de la base de datos. En el caso clásico, en la definición de redundancia (recordar definición 1.1.1 del primer capítulo), están tácitamente asumidas las siguientes hipótesis:

1. Los valores $A_i(t)$ son elementos atómicos del dominio subyacente D_{A_i}.
2. Se tiene definida una relación de identidad δ tal que:

$$\forall\, x,y \in D_i \;\; \delta(x,y) = \begin{cases} 1 \text{ si } x = y \\ 0 \text{ si } x \neq y \end{cases}$$

Entonces, dos tuplas t_1 y t_2 de una relación r, se dice que son redundantes si

$$\delta(A_i(t_1), A_i(t_2)) = 1 \quad \forall\ A_i \in REL(r)$$

Por lo tanto, debemos eliminar una de las dos tuplas anteriores, para eliminar así redundancia en la base de datos.

Sin embargo, las anteriores suposiciones dejan de ser ciertas en el modelo relacional difuso, por lo que debe extenderse el concepto de redundancia.

◇ *Planteamiento General*

En nuestra opinión, el proceso de definición y eliminación de la redundancia en el caso difuso, debe realizarse de la siguiente forma:

* Elección del **criterio de redundancia** $\mathcal{R}$: éste nos dirá, cuándo dos datos x, y, que en general son distribuciones de posibilidad, son informaciones redundantes. Este criterio sería la correspondiente extensión de la medida δ. Dos tuplas t_1 y t_2 serían redundantes si y solo si todos los elementos de cada atributo son redundantes, es decir:

  $$t_1\ \mathcal{R}\ t_2 \ \Leftrightarrow\ A_i(t_1)\ \mathcal{R}\ A_i(t_2) \quad \forall\ A_i \in REL(r)$$

  Una consideración muy importante es la siguiente: ¿debe ser $\mathcal{R}$ una medida distinta $\mathcal{R}_{A_i}$ para cada atributo A_i? Nosotros pensamos que no, aunque todas las aproximaciones utilizan una medida de semejanza distinta en cada atributo para definir redundancia. Posteriormente analizaremos los problemas que puede plantear dicha perspectiva.

  Por otra parte, es de destacar que el criterio que proponemos para decidir cuándo dos tuplas son redundantes, es crisp. Es decir, diremos que dos tuplas son, o no son redundantes, pero no diremos que dos tuplas son redundantes con un cierto grado. Pero, obviamente, el criterio de redundancia $\mathcal{R}$ atenderá a una definición aplicable a valores difusos.

* La eliminación de tuplas redundantes en el caso difuso, no puede suponer (como en el caso clásico), eliminar una de ellas. Por lo tanto, es hace necesario *fundir* dos tuplas

redundantes en una tercera que recoja la información de ambas. Por lo tanto, habrá que elegir un **operador de fusión** $\mathcal{F}$ que fusione dos valores redundantes en una única conclusión $x\,\mathcal{F}\,y$.

Al igual que antes, proponemos la utilización de un único operador de fusión $\mathcal{F}$ para todos los atributos. Así pues, si $t_1\ \mathcal{R}\ t_2$, entonces fusionaríamos todos los valores de ambas tuplas obteniendo la nueva tupla

$$(A_i(t_1)\,\mathcal{F}\,A_i(t_2))_{A_i \in REL(r)}$$

A la relación así formada la llamaremos

$$\mathcal{F}\,(r) \tag{2.5}$$

o simplemente $\mathcal{F}\,r$. Por ejemplo, dada la siguiente relación

$$r = \begin{array}{|cc|} \hline B & A \\ \hline b & a \\ b' & a' \\ \hline \end{array}$$

si $a\ \mathcal{R}\ a'$ y $b\ \mathcal{R}\ b'$, entonces debe fundirse en la siguiente:

$$\mathcal{F}\,r = \begin{array}{|cc|} \hline B & A \\ \hline b\,\mathcal{F}\,b' & a\,\mathcal{F}\,a' \\ \hline \end{array}$$

◇ *Extensiones Existentes*

Una vez que hemos descrito el proceso general de cómo ha de plantearse la eliminación de redundancia en el caso difuso, revisemos algunas aproximaciones al respecto:

- Rundensteiner et al [71] definen redundancia en base a una medida de semejanza R (definición 2.1.3) establecida sobre los elementos de los dominios:

  Definición 2.2.1 (Redundancia Rundensteiner et al.) *Dos tuplas t_1, t_2 se dice que son redundantes, a nivel $\alpha = (\alpha_1, \ldots, \alpha_n)$, $\beta \in [0, 1]$ si y solo si se verifican las siguientes condiciones:*

1. $\min_i(R_{A_i}(x,y)) \geq \alpha_i \quad \forall\ x \in A_i(t_1) \quad \forall\ y \in A_i(t_2)$

2. $\min_{z \in D_i}(1- \mid \pi_{A_i(t_1)}(z) - \pi_{A_i(t_2)}(z) \mid) \geq \beta$

La eliminación de la redundancia se realiza a través de la unión difusa de cada valor de las tuplas, es decir, $\mathcal{F} = \vee$

Una seria objeción a esta definición estriba en que el punto *1.* considera únicamente relaciones entre elementos de dominios, sin tener en cuenta los grados de posibilidad asociados con dichos elementos, mientras que el criterio del segundo punto, considera sólo la correspondencia entre los valores de posibilidad, independientemente de los grados de semejanza entre los elementos de dominio. Esto puede derivar en resultados anómalos, como el siguiente: considerar la siguiente relación de semejanza

	muy bueno	bueno	normal	malo	muy malo
muy bueno	1	0.8	0.3	0.1	0
bueno		1	0.7	0.2	0.1
normal			1	0.7	0.3
malo				1	0.8
muy malo					1

Supongamos $\pi_{A_i(t_1)} = \{\texttt{muy bueno}/0.2\ ,\ \texttt{malo}/1\}$ y $\pi_{A_i(t_2)} = \pi_{A_i(t_1)}$. Entonces:

$$\min_{z \in t_1,t_2}(1- \mid \pi_{A_i(t_1)}(z) - \pi_{A_i(t_2)}(z) \mid) = 0.1$$

por lo que t_1 y t_2 no son redundantes a niveles mayores que 0.1, lo cual resulta contraintuitivo.

- Chen et al [19], proponen la siguiente definición:

Definición 2.2.2 (Redundancia-Chen) . *Se define la distribución de posibilidad de que dos tuplas t_1 y t_2 sean iguales como una distribución sobre $\{T, F\}$ (T= True, F= False) igual a $I(t_1,t_2) = \{T/g, F/g'\}$ con:*

$$g = \min\{E(A_1(t_1), A_1(t_2))(T), \ldots, E(A_n(t_1), A_n(t_2))(T)\}$$

$$g' = \sup_{e_1 \wedge \ldots \wedge e_n = F} \min\{E(A_1(t_1), A_1(t_2))(e_1), \ldots, E(A_n(t_1), A_n(t_2))(e_n)\}$$

dónde:

$$E(A_i(t_1), A_i(t_2))(T) = \sup_{x,y \in D_i \ \mathit{tal\ que}\ R_i(x,y) \geq \alpha_i} \min\{\pi_{A_i(t_1)}(x), \pi_{A_i(t_2)}(y)\}$$

$$E(A_i(t_1), A_i(t_2))(F) = \sup_{x,y \in D_i \ \mathit{tal\ que}\ R_i(x,y) < \alpha_i} \min\{\pi_{A_i(t_1)}(x), \pi_{A_i(t_2)}(y)\}$$

con R_i una medida de semejanza y α_i umbrales impuestos de antemano en D_i.

Dos tuplas t_1, t_2 de una relación r se dice que son λ-redundantes si y solo si

$$c(g, g') \geq \lambda \ \textit{con } c \textit{ una combinación convexa}$$

El operador $\mathcal{F}$ utilizado para eliminar la redundancia es la unión difusa.

- Prade y Testemale [66] definen redundancia entre dos valores difusos a través de la extensión de una distancia de Haussdorf entre las funciones de pertenencia.
- En el modelo de Raju y Majumdar [69, 68], cada tupla t tiene asociada un grado de pertenencia $\mu_r(t)$ a la relación r. Entonces, dos tuplas t_1 y t_2 son redundantes si $A_i(t_1)$ es exactamente igual (en el sentido clásico) a $A_i(t_2)$. En el caso de que $\mu_r(t_1) \neq \mu_r(t_2)$, entonces, al fusionar ambas tuplas, se toma el máximo de ambos grados de pertenencia.

◇ *Inconvenientes de las Aproximaciones Existentes*

Ninguna de las aproximaciones anteriores (excepto, claro está, la de Raju y Majumdar en la que el criterio de redundancia es clásico) analizan si el proceso de eliminación de redundancia es o no es cerrado, es decir si:

$$\boxed{\mathcal{F}\,r = \mathcal{F}\,(\mathcal{F}\,r)} \tag{2.6}$$

Ejemplo 2.2.3 . Denotemos por $\simeq$ una medida de semejanza arbitraria. Pueden existir elementos verificando las siguientes restricciones:

$$a \simeq a',\ a' \simeq a'',\ a \not\simeq a'',\quad b \simeq b',\ b' \simeq b'', b \not\simeq b'',\ a \vee a' \simeq a' \vee a'',\ b \vee b' \simeq b' \vee b''$$

Dada la relación r:

$$r = \begin{array}{|cc|} \hline A & B \\ \hline a & b \\ a' & b' \\ a'' & b'' \\ \hline \end{array}$$

Si fundimos dichas tuplas, tendríamos como resultado la relación $\mathcal{F}\, r$ dada por:

$$\mathcal{F}\, r = \begin{array}{|cc|} \hline A & B \\ \hline a \vee a' & b \vee b' \\ a' \vee a'' & b' \vee b'' \\ \hline \end{array}$$

y no se han fundido t_1, t_3 ya que $a \not\simeq a''$. Pero, si ahora partimos de $\mathcal{F}\, r$ resulta que $a \vee a' \simeq a' \vee a''$, (análogamente con B) por lo que dichas tuplas se fundirían en la siguiente:

$$\mathcal{F}\,(\mathcal{F}\, r) = \begin{array}{|cc|} \hline A & B \\ \hline a \vee a' \vee a'' & b \vee b' \vee b'' \\ \hline \end{array}$$

Y por tanto $\mathcal{F}\,(\mathcal{F}\, r) \neq \mathcal{F}\, r$, es decir, el proceso de eliminación de redundancia no sería cerrado.

■

El problema radica en la utilización de medidas de semejanza, aproximación común en todas las extensiones difusas que hemos revisado. Desde nuestro punto de vista, que dos elementos sean semejantes o próximos no implica que la información que conlleva uno cualquiera de ellos, sea redundante con la del otro. Por lo tanto, hemos de afirmar que una medida de redundancia no puede basarse en el concepto de semejanza o proximidad. Volveremos a este punto con más detenimiento, en el capítulo cuarto.

Otro punto importante a destacar relacionado con el concepto de redundancia en las aproximaciones existentes, es la imposición de que en toda relación debe aparecer siempre al menos un atributo(s) que sea del tipo 1 (admitiendo únicamente valores crisp), siendo éste es la llave primaria o atributo clave. Esta es una fuerte imposición que subyace en todos los modelos difusos de bases de datos relacionales. En nuestro trabajo la relajaremos coherentemente, es decir, de forma que atienda a un criterio de redundancia difuso consistente. Esto lo veremos en el capítulo cuarto, en el que definimos el importante concepto de *llave difusa*.

2.2.4 Algebra Relacional Difusa

Una vez que hemos visto el tratamiento de redundancia en el modelo posibilístico, pasamos a revisar las extensiones de los operadores del algebra relacional, comentando los inconvenientes que presentan las distintas aproximaciones. Antes, vamos a plantear como en el apartado anterior, cual habría de ser el esquema general a seguir para definir dichos operadores.

◇ *Planteamiento General*

De entre todos los operadores, vamos a centrarnos en los que intervienen en la fase de diseño de la base de datos. Estos son la proyección y la reunión. Existen versiones difusas de otros operadores como la división (Yager [91], Mouaddib [59] y recientemente Vila et al [84]), pero no vamos a entrar a detallarlos, ya que no intervienen en el proceso de normalización. Empezamos con el esquema general de la proyección y a continuación pasaremos al de la reunión.

Proyección Difusa . La definición de proyección clásica de una relación r sobre un conjunto de atributos X ($\Pi_X(r)$), consiste en tomar los valores $X(t)\ \ \forall\, t \in r$, y eliminar (fusionar) las tuplas duplicadas (redundantes). La extensión difusa ha de plantearse en los mismos términos, por lo que debemos fijar un criterio de redundancia $\mathcal{R}$ y de fusión $\mathcal{F}$, tal y como se planteó en el apartado 2.2.3. Si denotamos en general, por $\tilde{\Pi}_X\ (r)$ la proyección difusa de una relación, esta debe definirse como sigue:

$$\boxed{\tilde{\Pi}_X\ (r) \equiv \mathcal{F}\ (\Pi_X(r))} \tag{2.7}$$

Reunión Difusa . La extensión difusa del operador de reunión también es inmediata. En el caso clásico es una selección sobre el producto cartesiano. La correspondiente extensión únicamente cambiará en que la condición presente en la selección, pasará a ser una condición difusa. Si denotamos por $r \,\tilde{\bowtie}_{XY}\, s$ a una reunión difusa genérica, ésta será de la forma:

$$\boxed{r \,\tilde{\bowtie}_{XY}\, s \equiv \sigma_{X^r \Theta Y^s}(REL(r) \times REL(s))} \tag{2.8}$$

dónde Θ es un comparador difuso.

Reunión Difusa Natural . Veamos cómo puede definirse este operador. Llamemos X al conjunto de atributos comunes de dos relaciones r y s. Este operador se basa en utilizar como condición, la igualdad estricta entre los valores de X^r y X^s. Entonces, se realiza la reunión y a continuación se proyecta eliminando cualquiera de las columnas X^r ó X^s (recordar la ecuación 1.11). Por lo tanto, en primer lugar, debemos determinar qué condición hay que tomar en el caso difuso. Supongamos que dicha condición es Θ. Entonces, al igual que el caso clásico, para definir la reunión natural entre dos relaciones r y s, primero hay que construir la reunión usual con condición dada por $X^r \Theta X^s$. Para fijar ideas, supongamos las siguientes relaciones:

$r =$

B	A^r
b_1	a_1
b_2	a_2

$s =$

A^s	C
a'_1	c_1

dónde $a_1 \Theta a'_1$. La Θ-reunión de r y s a través del conjunto de atributos comunes vendría dada por:

$r \mathbin{\tilde{\bowtie}}_{A^r A^s} s =$

B	A^r	A^s	C
b_1	a_1	a'_1	c_1

Ahora bien, si no queremos duplicar atributos cada vez que realizamos una reunión natural, debemos realizar alguno de los siguientes procesos:

- Podemos proyectar (claro está, según una definición de proyección <u>difusa</u>) sobre alguno de los atributos comunes. Supongamos por ejemplo, que se decide proyectar sobre A^r. Entonces la relación $r \mathbin{\tilde{\bowtie}}_{A^r A^s} s$, debería transformarse en la siguiente:

B	A	C
b_1	a_1	c_1

Si denotamos por $r \mathbin{\tilde{\bowtie}} s$ a la reunión natural difusa entre r y s, y X al conjunto de atributos comunes, este ejemplo correspondería en general, al siguiente esquema:

$$\boxed{r \mathbin{\tilde{\bowtie}} s \equiv \tilde{\Pi}_{REL(r) \times REL(s) - X^s} \left(\sigma_{X^r \Theta X^s}(REL(r) \times REL(s))\right)} \qquad (2.9)$$

– Podemos fusionar los valores que satisfacen la condición $X^r \Theta X^s$ (la fusión en el caso clásico no plantea ningún problema pues la condición Θ exigida es la igualdad). Así pues la relación $r \stackrel{\sim}{\bowtie}_{A^r A^s} s$, debería transformarse en la siguiente:

B	A	C
b_1	$a_1 \mathcal{J} a'_1$	c_1

dónde $\mathcal{J}$ es un operador de fusión, que en principio, no tendría por qué ser el mismo que $\mathcal{F}$. A la relación así formada la llamaríamos $\mathcal{J}\ (r \stackrel{\sim}{\bowtie}_{A^r A^s} s)$. En general, este ejemplo correspondería al siguiente esquema:

$$\boxed{r \stackrel{\sim}{\bowtie} s \equiv \mathcal{J}\ (\sigma_{X^r \Theta X^s}(REL(r) \times REL(s)))} \tag{2.10}$$

Es muy importante destacar que cualquiera de las aproximaciones anteriores es válida para extraer información de la base de datos, pero no lo es si queremos definir operadores de reconstrucción sin pérdidas. La elección que nosotros propondremos para definir reunión natural corresponderá a la primera aproximación (ecuación 2.9), y lo justificaremos desde un punto de vista semántico en el capítulo quinto de esta memoria. Veremos que otros autores (Weiyi [89, 88]) adoptan la solución dada en la ecuación 2.10, mientras que el resto de las aproximaciones no definen ni siquiera el operador de reunión natural difusa, sino que se limitan al criterio crisp.

◇ *Extensiones Existentes*

• Prade y Testemale [65] definen el concepto de proyección difusa sobre una relación difusa derivada de plantear una consulta (ver sección 2.2.2). Para ello, se toma la definición usual de proyección clásica, pero tomando el máximo de los grados de pertenencia, para aquellas tuplas iguales, es decir:

$$\mu_{\Pi_X(r)}(t) = \max_{t_i \in r \text{ tal que } X(t_i)=X(t)} \mu_r(t_i)$$

dónde $\mu_r(t_i)$ es o bien $N_C(t_i)$ o bien $\pi_C(t_i)$.

La definición de Prade y Testemale de **Θ-reunión** (a través del conjunto de atributos X) entre dos relaciones r y s responde al formato general que vimos en la ecuación 2.8. De nuevo, la condición de la selección, puede cualificarse en forma posibilística o en términos de necesidad:

$$\sigma\pi_{X^r \Theta X^s}(REL(r) \times REL(s))$$

$$\sigma N_{X^r \Theta X^s}(REL(r) \times REL(s)) \tag{2.11}$$

Por ejemplo, si suponemos que $a_1 \Theta a_1'$ con grado de posibilidad 0.8, y $a_2 \Theta a_1'$ con grado 0, entonces la reunión de las relaciones:

$r =$

B	A
b_1	a_1
b_2	a_2

$s =$

A	C
a_1'	c_1

vendría dada por:

$\sigma\pi_{A^r \Theta A^s}(REL(r) \times REL(s)) =$

B	A	A	C	$\pi_{A^r \Theta A^s}$
b_1	a_1	a_1'	c_1	0.8
b_2	a_2	a_1'	c_1	0

Hemos de observar que la definición de Prade y Testemale de proyección y de reunión (realmente fue Umano [79] el primer autor que abordó este problema) es válida para la extracción de información de la base de datos, cuando se plantean preguntas que involucran alguna comparación entre atributos de distintas relaciones. De hecho, salvo algunos matices sin importancia, es la adoptada en la mayoría de los modelos de bases de datos difusas (Zemankova y Kandel [101], Medina et al [58], ...). Pero ninguno de dichos autores define un operador de proyección difusa que responda al formato general dado en la ecuación 2.7. Es decir, el criterio utilizado para eliminar redundancia de una proyección es el de igualdad estricta. Curiosamente, algunos de ellos, sí utilizan un criterio difuso cuando se aplica la proyección a todo el esquema relacional para eliminar redundancia (esto se vió en la sección 2.2.3).

Además, ninguno de ellos aplican la definición de proyección en un proceso de descomposición, y por tanto en un estudio del diseño de la base de datos.

- Raju y Majumdar [68, 69] definen proyección de la misma forma que Prade y Testemale (criterio clásico de igualdad), pero sin distinguir entre grados de posibilidad y necesidad.

  En cuanto a la reunión difusa, sí definen un operador de reunión natural para utilizarlo en el proceso de diseño de la base de datos relacional difusa. El criterio utilizado es el de igualdad crisp entre valores difusos, lo que les permite obtener descomposiciones sin pérdidas (ver el apartado 2.4 de este capítulo). Pero esta aproximación no es aceptable ya que no resulta coherente almacenar datos difusos y tratarlos como si fuesen valores alfanuméricos (al definir un criterio de reunión clásico), perdiendo entonces la información que conllevan dichos valores difusos.

- Weiyi [88] define proyección de una relación <u>respecto</u> un conjunto de dependencias. En primer lugar, parte de la definición de Raju y Majumdar de dependencia funcional difusa. Supongamos que es F el conjunto de las dependencias que se imponen en una relación r. En el caso de que la relación no satisfazga una dependencia I de F, entonces se <u>modifican</u> los datos difusos presentes en los consecuentes de I. Esta modificación viene dada por una serie de reglas que actúan achicando el soporte de los difusos, pero sin tener en cuenta grados de pertenencia. A dicho proceso, lo llama $P_F(r)$. Si se quiere proyectar la relación r sobre un conjunto de atributos $X \subseteq REL(r)$, entonces se tomaría $P_{F_X}(\Pi_X(r))$, dónde F_X es la proyección del conjunto de dependencias F sobre X.

  Por otra parte, Weiyi es el único autor que sí define un operador de reunión natural difuso: si llamamos X al conjunto de atributos comunes de r y s, dos tuplas $t \in r$ y $t' \in s$ se reunen si y solo si $sop(t) \cap sop(t') \neq \emptyset$, que sería la condición Θ si seguimos la terminología que introducimos en la ecuación 2.10. El operador de fusión $\mathcal{J}$ viene dado a través de las reglas mencionadas anteriormente. Sin embargo, el criterio de reunir dos tuplas porque los atributos de la reunión tengan intersección de soportes no vacía, es demasiado simple y no responde a ninguna interpretación semántica. Además, no se tienen en cuenta los grados de pertenencia, por lo que, de hecho, se está trabajando con valores intervalares, pero no difusos.

A la luz de las consideraciones anteriores, surge la necesidad de plantear operadores coherentes del álgebra relacional difusa que nos permitan desarrollar una teoría de diseño adecuada (demostrando teoremas fundamentales como el de descomposición sin pérdidas). Nosotros los introduciremos en los capítulos cuarto y quinto de esta memoria.

2.3 Modelo de Unificación

2.3.1 Representación de la Vaguedad

El modelo de unificación fue desarrollado principalmente por Buckles y Petry [14, 15, 16]. Otros autores como Shenoi y Melton [72, 73] y Anvari y Rose [3], contribuyeron también a su desarrollo.

Los **dominios básicos** permitidos en este modelo son:

1. Conjunto finito de escalares. `DE={negro, marrón, rojo, rubio}`
2. Conjunto finito de números. `D={15 , 16 , 17}`
3. Conjunto de números difusos. `D={muy alto, alto, medio, bajo, muy bajo}`

Buckles y Petry [15] estudiaron en primer lugar los tipos de dominios 1 y 2, y posteriormente extendieron su modelo al tipo 3 ([16]).

Los valores que se permiten como **valores de dominios** son los siguientes:

- Elementos de dominios del tipo 1, 2 y 3. Por ejemplo, siguiendo la figura 2.5:

$$A_4(t_1) = \{\texttt{marrón}\} \ ; \ A_3(t_2) = \{\texttt{bajo}\}$$

- Subconjuntos de dominios del tipo 1, 2 y 3. Por ejemplo, siguiendo la figura 2.5:

$$A_4(t_2) = \{\texttt{negro , marrón}\} \ ; A_4(t_1) = \{\texttt{muy alto , alto}\}$$

Es importante destacar que la interpretación de dichos subconjuntos es la de posibles asignaciones mutuamente excluyentes. Por lo tanto, podría haberse utilizado también una distribución de posibilidad. Por ejemplo, $A_4(t_2)$ podrían ponerse en la forma:

$$A_4(t_2) = \{\texttt{negro/1 + marrón/1}\}$$

o equivalentemente

$$A_4(t_2) = \{\texttt{negro/1} \vee \texttt{marrón/1}\}$$

que es una distribución de posibilidad sobre un dominio no difuso. Pero hay que tener cuidado cuando se trabaja con subconjuntos sobre dominios de tipo 3, es decir, conjuntos de difusos. Siguiendo el ejemplo anterior, ¿cómo se interpreta el conjunto $A_4(t_1) = \{\texttt{muy alto , alto}\}$? Si suponemos que es en la forma {`muy alto/1 + alto/1`}, estamos utilizando un difuso de tipo 2, y este caso lo habíamos excluido en el modelo posibilístico. Sin embargo, podemos utilizar el operador de unión difusa, obteniendo el dato

$$A_4(t_1) = \{\texttt{muy alto} \vee \texttt{alto}\}$$

que es una distribución de posibilidad sobre un dominio no difuso. En definitiva, desde el punto de vista de representación de los datos, el modelo de unificación es realmente un caso particular del modelo posibilístico[1]. Como consecuencia de este hecho, veremos que ocurre lo mismo con respecto a los operadores básicos.

En general, pues:

$$A_j(t_i) \subseteq D_j \quad \forall\, i \quad \forall\, j$$

Se define entonces una **relación** r como un subconjunto del producto cartesiano $2^{D_1} \times \ldots 2^{D_m}$, dónde 2^{D_i} es cualquier elemento (no subconjunto) de $\mathcal{P}\ (D_i)$. Un ejemplo de relación en este modelo sería el dado en la figura 2.5:

2.3.2 Tratamiento de la Redundancia

En el contexto del modelo de unificación de Buckles y Petry, existen varias aproximaciones al problema de la eliminación de la redundancia, basadas todas ellas en la imposición de particiones de los dominios de la base de datos. Una herramienta básica la constituye las

[1] Como ya hemos comentado en ocasiones anteriores, Vila et al [85] definen un modelo lógico de base de datos difusa que comprende como casos particulares, tanto en la representación como en el tratamiento, los modelos posibilísticos y de unificación

	Nombre	Edad	Altura	Color-de-Pelo
t_1	Juan	16	{muy alto, alto}	marrón
t_2	Jose	17	bajo	{negro,marrón}
t_3	Pedro	15	muy bajo	rubio
t_4	Mario	{15, 16}	178	rojo

Figura 2.5. Relación en el modelo de unificación

relaciones de similitud (definición 2.1.4). Estableciendo un nivel de similitud, se forman clases de equivalencia (subconjuntos) constituidas por los elementos similares a dicho nivel. Tenemos por ejemplo, las siguientes definiciones:

Definición 2.3.1 (Redundancia Buckles, Petry) *Dos tuplas t_1, t_2 se dice que son redundantes, a nivel $\alpha = (\alpha_1, \ldots, \alpha_n)$, dónde cada α_i se impone en el atributo A_i, si y solo si:*

$$\min_i(sim_i(x,y)) \geq \alpha_i \quad \forall\ x, y \in A_i(t_1) \cup A_i(t_2)$$

dónde sim_i son relaciones de similitud

Al utilizar medidas de similitud, se inducen particiones en D_i; es decir, $\forall\ \alpha_i$ los elementos del dominio D_i pueden agruparse en clases de equivalencia, de forma que todos los elementos dentro de una misma clase se consideran semejantes (a nivel α_i), mientras que elementos en clases distintas se consideran distintos. En definitiva, los *datos átomicos*, que en el modelo clásico son valores crisp, pasan a ser clases de equivalencia en el modelo de Buckles Petry.

La relación resultante despues de permitir unos umbrales máximos α_i para los diferentes dominios, se obtiene a través de un proceso de fusión de tuplas que los autores denominan *mezcla* (en inglés *merging*). Este proceso consiste en tomar la unión conjuntista (para cada atributo) de los valores de dominio que sean α_i-similares (sin que se viole ningún umbral). Obsérvese que este proceso de eliminación de redundancia responde al esquema general que vimo en el modelo posibilístico (sección 2.2.3). Basta tomar como medida de redundancia $\mathcal{R}$ la dada por la utilización de la medida de semejanza, siendo el operador de fusión la unión difusa, tal y como hemos planteado al principio de esta sección en la página 71.

Para números difusos (dominios de tipo 3), Buckles y Petry [16] extendieron su modelo original, definiendo un tipo particular de "cercanía" en base a los $\alpha-$cortes. En principio, se define:

Definición 2.3.2 *Dos números difusos F_1 y F_2 se dice que son $\alpha-$similares $F_1 S_\alpha F_2$ si dado $\beta \in [0,1]$, $x, y \in (F_1 \vee F_2)_\alpha$ y $z = \beta x + (1-\beta) y$, entonces $z \in (F_1 \vee F_2)_\alpha$.*

Obsérvese que esta definición obliga a que el α-corte de la unión difusa de ambos valores, ha de ser un intervalo de la recta real. Sin embargo, al no obtenerse una partición del dominio con esta medida de comparación, se fuerza a la utilización de una medida más restrictiva, que sí la induzca. En este sentido definen:

Definición 2.3.3 *Dos números difusos F_1 y F_2 se dice que son $\alpha-$próximos $F_1 S_\alpha^+ F_2$ si existe una secuencia de cero o más números difusos $F_{h_1} \dots F_{h_k}$ tal que:*

$$F_1 S_\alpha F_{h_1} S_\alpha \dots S_\alpha F_{h_k} S_\alpha F_2$$

Shenoi y Melton [72, 73] extendieron el trabajo suprimiendo la transitividad en la relación de similitud, pero obligando a la existencia de una partición en cada dominio:

Definición 2.3.4 (Redundancia Shenoi, Melton) *Dos tuplas t_1, t_2 se dice que son redundantes, a nivel $\alpha = (\alpha_1, \dots, \alpha_n)$, si y solo si:*

$$x P_{\alpha_i} y \quad \forall\, x, y \in A_i(t_1) \cup A_i(t_2)$$

dónde $x P_{\alpha_i} y$ si y solo si existe una secuencia finita de valores $y_1, \dots, y_k \in D_i$ tal que:

$$p_i(x, y_1) \geq \alpha_i,\ p_i(y_1, y_2) \geq \alpha_i, \dots,\ p_i(y_r, y) \geq \alpha_i$$

con p_i relaciones de semejanza (definición 2.1.3).

Es de destacar que, aunque p_i no es una medida de similitud, en la definición de P_{α_i} se está imponiendo una propiedad de transitividad, al imponer proximidad entre x y z, siempre que $p_i(x,y) \geq \alpha$, $p_i(y,z) \geq \alpha$. Con esta fuerte restricción, se obtiene de nuevo, como en el modelo anterior, una partición del dominio D_i.

2.3.3 Algebra Relacional Difusa

En cuanto a los operadores del álgebra relacional de proyección y reunión, tanto Buckles y Petry, como Shenoi y Melton los definen según el planteamiento general que introducíamos en las ecuaciones 2.7 y 2.10, dónde los operadores de fusión $\mathcal{F}$ y $\mathcal{J}$ pasan a ser la unión conjuntista, o equivalentemente, la unión difusa.

2.3.4 Inconvenientes de las Aproximaciones Existentes

El tratamiento de redundancia de Buckles y Petry permite generalizar buenas propiedades de las b.d.r como que el proceso de eliminación de redundancia sea cerrado, y que tras él, no haya dos tuplas en las que las clases de equivalencia que aparecen puedan solaparse. Sin embargo, la imposición de transitividad en la relación de similitud utilizada por Buckles y Petry resulta demasiado restrictiva en la práctica; algunos ejemplos pueden encontrarse en [19, 71, 101]. Es por ello que, si bien se obtienen buenas propiedades en cuanto al comportamiento de la base de datos, no es aplicable en sistemas reales, en los que las relaciones que modelan la semejanza no son transitivas.

La definición de Shenoi y Melton presenta la misma objeción ya que la definición de redundancia se basa en una partición <u>no difusa</u> del dominio de los conjuntos difusos, por lo que presenta la misma desventaja que Buckles y Petry de inaplicabilidad en sistemas reales.

Es muy importante destacar que la crítica que realizamos al modelo de unificación es que intenta incorporar y tratar informaciones imprecisas, utilizando clases de equivalencia, y esto no es permisible desde la perspectiva de un modelo difuso. Claro está, como modelo matemático no tiene ningún inconveniente, pero su desarrollo no plantea ninguna dificultad ya que la filosofía sigue siendo la misma que la del modelo relacional clásico; de hecho, si consideramos las clases de equivalencia como valores atómicos, el modelo de unificación se convierte en el modelo relacional clásico.

2.4 Dependencias Funcionales Difusas

Una vez que hemos revisado los modelos de bases de datos relacionales difusas más importantes, pasamos a ver las extensiones difusas al concepto de dependencia funcional. Así mismo, veremos los casos en los que se llega a obtener descomposiciones sin pérdidas, así como los inconvenientes que éstos plantean. Posteriormente, analizaremos una serie de propiedades que toda definición de dependencia difusa debiese cumplir y finalmente introduciremos una nueva definición de dependencia que sí satisfazga dichas propiedades. Esta definición será la base de la que se desarrolle en el capítulo cuarto, y que nos permitirá obtener descomposiciones sin pérdidas.

2.4.1 Extensiones Difusas a las Dependencias Funcionales

◇ *Aproximación de Kiss y de Tripathy y Saxena*

A. Kiss [51, 52] trata el problema de la definición de una dependencia funcional, bajo el modelo relacional simple, dónde las tuplas son clásicas y únicamente se añade un grado de pertenencia de cada tupla a la relación. Tripathy y Saxena [77] realizan un estudio similar con dependencias multivaluadas.

Ambos autores obtienen descomposiciones sin pérdidas (Kiss en el caso de dependencias funcionales y Tripathy y Saxena en el caso de multivaluadas), pero no vamos a entrar en su estudio por lo restringido que resulta el modelo relacional puro.

◇ *Aproximación de Shenoi y Melton*

En el contexto del modelo de unificación, Shenoi y Melton [74], definen dependencia funcional como una extensión de la definición 1.2.2 clásica de dependencia, utilizando la definición 2.3.4 de tuplas redundantes a unos niveles dados.

Definición 2.4.1 (Dependencia Shenoi y Melton) *Una relación r satisface una* **dependencia funcional difusa** *a niveles* (α, β), *denotada por* $X \overset{(\alpha,\beta)}{\rightsquigarrow} Y$ *con* $X = (A_i)_{i \in I}$, $Y = (A_j)_{j \in J}$, $\alpha = (\alpha_i)_{i \in I}$, $\beta = (\beta_j)_{j \in J}$, $I, J \subseteq \{1 \ldots n\}$, *si y solo si se verifica lo siguiente para cualquier par de tuplas* t_1, t_2*:*

$$Si\ X_i(t_1)\, P_{\alpha_i}\, X_i(t_2) \quad \forall\ X_i \in X \quad entonces\ debe\ ser\ Y_j(t_1)\, P_{\beta_j}\, Y_j(t_2) \quad \forall\ Y_j \in Y$$

dónde P_{α_i} *es el operador de similitud dado en la definición 2.3.4*

Como ya vimos, las relaciones P_{α_i} inducían clases de equivalencia, por lo que tomando éstas como los átomos de información en un modelo relacional clásico, es fácil extender todos los resultados sobre descomposiciones sin pérdidas del caso no difuso. Pero, al igual que analizábamos en la sección anterior, la imprecisión o vaguedad de la información no puede modelarse estableciendo particiones no difusas de los dominios. Por lo tanto, podemos concluir que la definición de dependencia funcional difusa, y por tanto el modelo de Shenoi y Melton no es válido como modelo de diseño difuso.

◇ *Aproximación de Dubois, Prade y Testemale*

En un principio, Prade y Testemale [65] definieron una dependencia funcional difusa $A \rightarrow B$ entre atributos simples, a través de la siguiente condición:

$$\text{Si } A(t_1) = A(t_2), \text{ entonces debe ser } \pi(B(t_1)\Theta B(t_2)) \geq \beta$$

dónde el término $\pi(B(t_1)\Theta B(t_2))$ es el dado por la ecuación 2.3, con Θ una relación de semejanza.

Sin embargo, Prade y Testemale [67] y posteriormente Dubois y Prade [42] estudiaron el tema de la restricción de integridad derivada de una dependencia difusa, a través de una distribución de posibilidad condicional $\pi_{A_j|A_i}$, afirmando que una dependencia funcional entre dos atributos puede venir modelada a través de dicha distribución.

De todas formas, dichos autores no estudian el comportamiento de dicha definición frente a los Axiomas de Armstrong, ni las posibles descomposiciones sin pérdidas, etc.

Unicamente sugieren en [42], la posibilidad de tomar la intersección de los valores difusos para obligar al cumplimiento de la dependencia funcional difusa, tal y como realiza Vasiliou [83] en el caso clásico y Weiyi [89, 88] en un caso difuso restringido.

◇ *Aproximación de Raju y Majumdar*

Raju y Majumdar [68, 69] definieron dependencia funcional difusa, en el contexto del modelo posibilístico, pero añadiendo a cada tupla un grado de pertenencia a la relación, como sigue:

Definición 2.4.2 (Raju Majumdar) *Una instancia r de un esquema relacional dado por $R(A_1, \ldots, A_n)$ satisface una dependencia funcional difusa (que en nuestro trabajo denotaremos por* **RM-dfd** *si y solo si:*

$$c(Y(t_1), Y(t_2)) \geq c(X(t_1), X(t_2)) \quad \forall\ t_1,\ t_2$$

dónde

$$c(X(t_1), X(t_2)) = \min_{i \in I} c_i(A_i(t_1), A_i(t_2))$$

con c_i una medida de igualdad entre los conjuntos difusos en el atributo A_i, y de forma análoga con $c(Y(t_1), Y(t_2))$

$$c_i(A_i(t_1), A_i(t_2)) = \min_{d_i} \tau(\mu_{A_i(t_1)}(d_i), \mu_{A_i(t_2)}(d_i))$$

y τ una relación de semejanza (reflexiva y simétrica) sobre $[0, 1]$.

La definición de dependencia resulta restrictiva, ya que para dos tuplas t_1 y t_2 con $c(X(t_1), X(t_2)) = 0.2$ y $c(Y(t_1), Y(t_2)) = 0.1$ se violaría la dependencia, y sin embargo los antecedentes son prácticamente diferentes. Volveremos a este punto posteriormente.

Por otra parte, vimos en la sección 2.2 que el criterio de reunión utilizado es el clásico de igualdad estricta entre valores. Esto les permita obtener (junto a la restricción adicional $c(a, b) < 1 \quad \forall\ a \neq b$) descomposiciones sin pérdidas. Sin embargo, volvemos a enfatizar la necesidad en la utilización de operadores difusos para tratar con datos difusos.

◇ *Aproximación de Chen et al*

Chen et al [20, 18] extendieron la definición de Raju y Majumdar, usando un operador de implicación difusa (Zadeh [100]) en vez de $\geq$. Por lo tanto, su definición de d.f.d queda como sigue:

Definición 2.4.3 (Chen) *Una instancia r de un esquema relacional $R(A_1, \ldots, A_n)$ satisface una ϕ-dependencia funcional difusa, que en nuestro trabajo denotaremos por* **Ch-dfd** *si y solo si:*

$$\min_{t_1,t_2 \in R} I(c(X(t_1), X(t_2)), c(Y(t_1), Y(t_2))) \geq \phi$$

dónde I es un operador de implicación difuso, y

$$c(X(t_1), X(t_2)) = \min_{i \in I} c_i(A_i(t_1), A_i(t_2))$$

con c_i una medida de igualdad entre los difusos definidos en el atributo A_i:

$$c_i(A_i(t_1), A_i(t_2)) = \sup_{\substack{d_i, d'_i \\ \psi_i(d_i, d'_i) \geq \lambda_i}} \min(\mu_{A_i(t_1)}(d_i), \mu_{A_i(t_2)}(d'_i))$$

con ψ_i una relación de semejanza sobre el dominio de A_i, $\lambda_i \in [0,1]$, y de forma análoga con $c(Y(t_1), Y(t_2))$.

Los autores, imponen la utilización de:

$$I(a,b) = \begin{cases} 1 & \text{si } a \leq b \\ f(a) \text{ o } f(b) & \text{en otro caso} \end{cases}$$

dónde f es una función de $[0,1] \times [0,1]$ en $[0,1]$, y más especificamente:

$$I^*(a,b) = \begin{cases} 1 & \text{si } a \leq b \\ b & \text{en otro caso} \end{cases}$$

para así poder obtener ciertas propiedades, como el cumplimiento de los axiomas de Armstrong[1]. Por lo tanto, a partir de ahora, cuando hablemos de Ch-dfd supondremos que estamos trabajando con I^*. Tenemos el siguiente resultado, si suponemos que se está utilizando la misma medida de compatibilidad c_i en ambos casos:

[1] Observar que si tomamos $f \equiv 0$ obtenemos la definición de RM-dfd

Proposición 2.4.4 *Si una relación r satisface una RM-dfd entonces r satisface también una Ch-dfd.*

Demostración.

Partiendo de las hipótesis, tenemos:

$$c(Y(t_1), Y(t_2)) \geq c(X(t_1), X(t_2)) \quad \forall\, t_1, t_2 \Rightarrow$$

$$\Rightarrow \quad I(c(X(t_1), X(t_2)), c(Y(t_1), Y(t_2))) = 1 \geq \phi \quad \forall\, \phi \quad \forall\, t_1, t_2$$

por lo que podemos concluir que existe Ch-dfd para cualquier valor de ϕ. □

Por ahora, dichos autores no han establecido una teoría de diseño (aunque sí obtienen la completitud de los axiomas de Armstrong), por lo que ni siquiera definen el concepto de reunión natural. En cualquier caso, pensamos que el concepto de dependencia funcional difusa ha de ser un criterio crisp, de forma que al experto se le pueda informar de la existencia o no de una dependencia (eso sí, respecto a unos umbrales de semejanza). Es por ello, que no nos parece adecuado el término de ϕ-dependencia. Posteriormente, en la sección 2.4.2. analizaremos algunos inconvenientes adicionales que presenta esta definición de dependencia funcional.

◇ *Aproximación de Weiyi*

Weiyi utiliza en [89, 88] la misma definición de dependencia funcional difusa que Raju y Majumdar. Cambia en la definición de una descomposición en base a la imposición de la dependencia funcional sobre la relación original. Veámos cómo: por una parte, partiendo de una relación r verificando un conjunto de d.f.d F, se van modificando los datos hasta obtener una relación r^* que verifica la d.f.d. Este proceso ya fué descrito en la página 68, dónde utilizabámos la notación $r^* \equiv P_F(r)$. Por otra parte, se toman las proyecciones clásicas de r (r_i) sobre los esquemas de la descomposición y sobre estas relaciones r_i se aplica el mismo proceso de compresión de datos, obteniendo las relaciones $r_i^* \equiv P_{F_i}(r)$. Ahora, la descomposición es sin pérdidas si toda tupla de r^* está *representada* en la relación $r_1^* \bowtie \ldots \bowtie r_n^*$, dónde el operador de reunión es el descrito también en la página 68.

Esta forma de trabajar ya la sugirió Vasiliou en [83] restringiéndose al caso del valor `desconocido`. Sin embargo, nosotros pensamos que los datos originales no deben modificarse, a no ser que así lo requiera el experto.

Además, aunque no utiliza una definición de reunión natural clásica como ya vimos en el apartado 2.2, presentaba el inconveniente de que no podía considerarse un operador difuso, en el sentido de que no tenía en cuenta los grados de pertenencia sino únicamente atendía al soporte de los datos. Además, reunir dos tuplas t_1 y t_2 a través de un conjunto de atributos X, simplemente porque verifiquen que $sop(X(t_1)) \cap sop(X(t_2)) \neq \emptyset$, no tiene ninguna justificación desde el punto de vista semántico (desde la perspectiva del diseño).

2.4.2 Propiedades de una Dependencia Funcional Difusa

Una vez que hemos revisado las aproximaciones existentes al problema de la extensión difusa del concepto de dependencia funcional, pasamos a analizar qué tipo de propiedades mínimas habría que exigirle a cualquier extensión de dicho concepto, analizando el comportamiento de dichas aproximaciones respecto a estas propiedades. Estas, deberían responder a consideraciones semánticas mínimamente exigibles:

i) Las dependencias funcionales clásicas deben ser un caso particular en la definición de una d.f.d. Por lo tanto, si una relación satisface una d.f.c. entonces debe satisfacer un caso particular de la d.f.d, pero no una d.f.d. en general.

ii.1) Valores de X diferentes no deben considerarse en la dependencia funcional, ya que debemos comparar hasta qué punto "son iguales" dos valores y **no** hasta qué punto "no son distintos". Por ejemplo, si tenemos dos tuplas t_1, t_2 con grados de igualdad dados por $c(X(t_1), X(t_2)) = 0.2$ y $c(Y(t_1), Y(t_2)) = 0.1$, no deberían violar la d.f.d.

Practicamente Diferentes	X	Y	Ninguna Restricción
	$X(t_1)$	$Y(t_1)$	
	$X(t_2)$	$Y(t_2)$	

ii.2) Los valores de Y deben ser suficientemente similares, cuando los valores de X son suficientemente similares. Por ejemplo, las tuplas t_1, t_2 con $c(X(t_1), X(t_2)) = 1$ y $c(Y(t_1), Y(t_2)) = 0.9$ no deberían violar la d.f.d.

Suficientemente Similares	X	Y	Suficientemente Similares
	$X(t_1)$	$Y(t_1)$	
	$X(t_2)$	$Y(t_2)$	

iii) Cada atributo debe tener su propia e inherente tipo de vaguedad, y por ejemplo, el grado de indiferencia considerado en cada atributo puede ser distinto al impuesto en otro atributo. Así pues hablaremos de valores X-similares e Y-similares y podremos imponer restricciones del tipo:

Si los valores de X son *muy similares*, entonces
los valores de Y deben ser *bastante similares*

o por ejemplo

Si los valores de X son *similares*, entonces
los valores de Y deben ser *muy similares*

Pasemos a estudiar el comportamiento de las aproximaciones existentes en el modelo posibilístico con respecto a estas propiedades:

Proposición 2.4.5 *Las RM-dfd satisfacen i, pero no ii.1, ii.2, iii.*

Demostración.

Para ver *i)* tomar c_i de forma que $c_i(a,b) = \delta_{ab} = 1$ si y solo si $a = b$ (en otro caso $\delta_{ab} = 0$).

Para *iii)* debemos destacar que la medida de igualdad entre dos datos $X(t_1), X(t_2)$ se calcula a través del mínimo entre c_i, de forma que no se puede imponer, por ejemplo, un nivel permitido de similaridad mayor en Y que el permitido en X.

Para *ii.1* y *ii.2* ver ejemplo 2.5.9, al final del capítulo (página 96). □

Proposición 2.4.6 *Las Ch-dfd satisfacen i., ii.2, pero no ii.1, iii.*

Demostración.

Para demostrar *i)* basta tomar $c_i(a,b) = \delta_{ab}$ y $\phi > 0$.

La violación de *iii)* se sigue de la definición y aplicando el mismo razonamiento que en la proposición anterior.

Para las otras propiedades, sea $c_1 = c(X(t_1), X(t_2))$, $c_2 = c(Y(t_1), Y(t_2))$ para dos tuplas cualesquiera t_1, t_2. Si $c_1 > c_2$ entonces $I(c_1, c_2) = c_2$. Por lo tanto, para obtener dependencia, debe ser $c_2 \geq \phi$. Así, para valores de c_2 pequeños, no habría dependencia para valores usuales de ϕ, por lo que se violaría la propiedad *ii.1)*. Si c_2 es alto, entonces las tuplas no violan la dependencia para valores razonables de ϕ (no demasiado pequeños): por lo tanto, podemos concluir que *ii.2)* sí se verifica.

Por ejemplo, consideremos una relación r en la que se espera que B depende de A. Supongamos dos tuplas en la base de datos con valores:

A	B
$a_1 = 1/20 + 0.2/30$	b_1
$a_2 = 0.2/20 + 1/30$	b_2

Supongamos que $c(b_1, b_2) = 0$ y $\psi_A(20, 30) = 0.3$. Entonces, $c(a_1, a_2) = 0.2$ $\forall \lambda_A > 0.3$ y $I^*(c(a_1, a_2), c(b_1, b_2)) = 0$. Por lo tanto, no hay Ch-dfd $\forall \phi > 0$ pero es obvio que, en términos difusos, las tuplas anteriores no violan una dependencia difusa $A \longrightarrow B$ porque tenemos valores de A practicamente diferentes.

En el caso de que, por ejemplo, fuese $c(a_1, a_2) = 0.9$ y $c(b_1, b_2) = 0.8$, entonces

$$I^*(c(a_1, a_2), c(b_1, b_2)) = 0.8$$

y las tuplas no romperían la dependencia $\forall \phi \geq 0.8$. Si se considera que ϕ está fijo, entonces se pierde la dependencia $\forall\ \phi < 0.8$, pero la anterior restricción para ϕ podría ser razonable; si no se considera fijo, entonces obtendríamos una "($\phi <$ 0.8)$-$dependencia". □

Este mal comportamiento de las extensiones difusas a las dependencias funcionales, nos lleva a proponer otra definición sin que viole las propiedades citadas. Puntualicemos que la definición de Shenoi y Melton de d.f.d sí satisface las anteriores propiedades, pues, de hecho, también trabajan con niveles de semejanza en antecedentes, distintos a los de los consecuentes. Sin embargo, nos hemos centrado en las aproximaciones que tienen interés desde un punto de vista difuso, es decir, aquellas definidas para el modelo posibilístico.

2.5 (α, β) Dependencias Funcionales Difusas

En esta sección procedemos a introducir un nuevo concepto de dependencia funcional difusa (este trabajo se presentó en [30]). Analizaremos el comportamiento de esta definición respecto a las propiedades básicas descritas en la sección anterior, y demostraremos la completitud de los axiomas de Armstrong. Finalizaremos este capítulo analizando cuales son los siguientes pasos que hemos de dar para proseguir el estudio del diseño de una base de datos relacional difusa.

2.5.1 Preliminares

El marco de base de datos difusa en el que vamos a trabajar es el modelo posibilístico, en el que los valores permitidos en una tupla son, en general, distribuciones de posibilidad. Impondremos que no exista un grado de pertenencia de la tupla a la relación. Así que consideraremos una relación r como un subconjunto de $\times_{i=1}^{n} \widetilde{\mathcal{P}}(D_i)$ con D_i el correspondiente dominio crisp.

Para no violar la propiedad *iii)* de la sección 2.4.2, debemos considerar junto con cada atributo A_i lo siguiente:

- Una relación de compatibilidad denotada por $comp_i$ dada directamente, o construida a través de los valores del universo de discurso.
- Umbrales α_{A_i} dados por un experto, de forma que grados de compatibilidad mayores que dichos umbrales implican indistinguibilidad.

En este contexto, planteamos la siguiente definición:

Definición 2.5.1 *En una base de datos relacional difusa, un* **esquema relacional difuso** *REL es una terna:*

$$REL = (R, \alpha, comp) \tag{2.12}$$

dónde R es un conjunto de atributos $(A_i)_{i=1\dots n}$, α es un conjunto de umbrales de semejanza $\alpha = (\alpha_{A_i})_{i=1\dots n}$, y comp es un conjunto de medidas de compatibilidad $comp = (comp_{A_i})_{i=1\dots n}$ asociadas a cada A_i.

Con esta definición se pretende enfatizar el hecho de que, al especificar un esquema relacional en una base de datos relacional difusa, debemos especificar, no sólo los atributos que lo componen, sino también las relaciones de compatibilidad[1] y umbrales de semejanza asociados a cada atributo. En cualquier caso, a lo largo de este trabajo, y para facilitar la notación, cuando nos refiramos a un esquema relacional, se omitirán los dos últimos términos de la terna. Utilizaremos la misma nomenclatura del caso clásico para denotar el esquema relacional de una relación r, a saber $REL(r)$ o simplemente R si no hay confusión.

Cada α_{A_i} debe ser fijado previamente, pero no hay ninguna restricción en la selección, excepto $\alpha_{A_i} > 0$. Denotaremos por α_Z al vector de umbrales para el conjunto de atributos $Z = (A_h)_{h\in\aleph}$ dónde $\aleph \subseteq \{1\dots n\}$, es decir:

$$\alpha_Z = (\alpha_{A_h})_{h\in\aleph}$$

Utilizaremos también la siguiente notación

$$\overrightarrow{comp}\,(Z(t_1), Z(t_2)) \equiv (comp_{A_h}(A_h(t_1), A_h(t_2)))_{h\in\aleph}$$

para tuplas t_1, t_2 arbitrarias.
La desigualdad

$$\overrightarrow{comp}\,(Z(t_1), Z(t_2)) \geq \alpha_Z$$

se entiende que es componente a componente, por lo que la anterior aserción es equivalente a la siguiente:

$$comp_{A_h}(A_h(t_1), A_h(t_2)) \geq \alpha_{A_h} \quad \forall\, h \in \aleph$$

[1]Por ahora, no se impondrá ninguna restricción en la elección de *comp*. Sin embargo, para obtener descomposiciones sin pérdidas, deberemos restringir el estudio a medidas de compatibilidad concretas. Este problema se abordará en el tercer capítulo

Cuando queramos designar el conjunto de umbrales de los atributos antecedentes (X) presentes en una dependencia funcional, utilizaremos únicamente la notación α; para los consecuentes será β. Es decir:

$$\alpha_X \equiv \alpha \ , \ \alpha_Y \equiv \beta$$

En este caso, también usaremos α_i para denotar a α_{A_i} (con $A_i \in X$) y β_j para β_{A_j} (con $A_j \in Y$). En cuanto a las relaciones de compatibilidad, y para facilitar la nomenclatura, suprimiremos el subíndice que hace referencia al atributo A_i, utilizando la notación $comp_i$ para designar a $comp_{A_i}$.

Una vez fijada la notación pasamos a extender el concepto de dependencia funcional. Este se basaba en la comparación por igualdad de pares de tuplas. Debemos definir por tanto, la extensión difusa del comparador utilizado:

Definición 2.5.2 *Diremos que dos valores difusos F y F' son* **semejantes a nivel** α*, respecto la medida de compatibilidad comp, y lo notaremos por*

$$F \simeq_\alpha F'$$

si y solo si

$$comp(F, F') \geq \alpha$$

Nota. En realidad, deberíamos utilizar la notación $\simeq_\alpha^{comp}$, para enfatizar que la medida de semejanza depende de la medida de compatibilidad. Sin embargo, por comodidad, omitiremos el superíndice *comp*, tal y como aparece en la definición anterior.

Definición 2.5.3 *Diremos que dos tuplas t_1 y t_2 de una relación r, son* **semejantes a nivel** α*, si y solo si:*

$$A_i(t_1) \simeq_\alpha A_i(t_2) \quad \forall\ A_i \in REL(r)$$

Es de destacar que al medir grado de semejanza entre dos tuplas, no conectamos los grados de semejanza correspondientes a cada atributo a través de una t-norma cualquiera, como ocurre por ejemplo en la aproximación de Chen [20, 18]. Para mantener la independencia en el trato de cada atributo, obligamos a que deba cumplirse, separadamente, la semejanza entre valores de tuplas. Estamos en condiciones de dar la siguiente definición de dependencia funcional difusa ([30]):

Definición 2.5.4 ((α, β)-d.f.d) *Un esquema relacional difuso $R(A_1 \ldots A_n)$ satisface una $(\alpha, \beta)-$***dependencia funcional difusa** *denotado por $X \xrightarrow{(\alpha,\beta)} Y$ con $X = (A_i)_{i \in I}$, $Y = (A_j)_{j \in J}$, $\alpha = (\alpha_i)_{i \in I}$, $\beta = (\beta_j)_{j \in J}$, $I, J \subseteq \{1 \ldots n\}$, si y solo si toda relación r con esquema R satisface lo siguiente para cualquier par de tuplas t_1 y t_2:*

$$\text{Si } \overrightarrow{comp}\,(X(t_1), X(t_2)) \geq \alpha \text{ entonces debe ser } \overrightarrow{comp}\,(Y(t_1), Y(t_2)) \geq \beta$$

o lo que es lo mismo:

$$\text{Si } X(t_1) \simeq_\alpha X(t_2) \text{ entonces debe ser } Y(t_1) \simeq_\beta Y(t_2) \tag{2.13}$$

dónde:

$$X(t_1) \simeq_\alpha X(t_2) \;\Leftrightarrow\; A_i(t_1) \simeq_{\alpha_i} A_i(t_2) \quad \forall\, A_i \in X$$

Recordemos que para definir redundancia entre dos valores difusos, abogábamos en la sección 2.2.3, por la utilización de medidas que no fuesen de semejanza. Sin embargo, para definir dependencia funcional, sí debemos imponer proximidad (pero no redundancia) en los valores consecuentes, cuando los valores antecendentes son también próximos (no únicamente redundantes). Veremos en los capítulos tercero, cuarto y quinto que, gracias a esta aproximación, podremos extender al caso difuso los resultados más importantes del caso clásico.

Por otra parte, hemos de destacar que el término (α, β)-dependencia no es utilizado en el mismo sentido que Chen et al [20, 18], sino en la forma utilizada por Shenoi y Melton [74]. Los primeros definen el criterio de dependencia con un grado ϕ, de forma que una relación puede satisfacer una d.f.d, por ejemplo, con grado 0.3. En nuestra aproximación, el criterio para establecer si una relación satisface o no una d.f.d, es un criterio crisp, por lo que no diremos que una relación satisface la d.f.d a nivel $(0.2, 0.4)$, sino que la satisface o no, teniendo en cuenta niveles $(0.2, 0.4)$ para comparar los atributos antecedente y consecuente respectivamente, y establecidos a priori. Este planteamiento no presenta ninguna ventaja de cara al experto, ya que ambas aproximaciones le permitirían una interpretación del grado de conexión entre los datos que aparecen en una relación. Sin embargo, sí presenta otras ventajas:

1. En cuanto al formulismo de la definición, Chen et al computan el valor de ϕ, a partir de una relación concreta r, por lo que la definición se aplica sobre una instancia r de un esquema R. Sin embargo, debería aplicarse directamente sobre R, de forma que los niveles deben ser impuestos de antemano pero no *calculados* a partir de los datos. Es por ello que al definir un esquema relacional difuso, introdujimos en la terna los umbrales de semejanza.

2. Nos permite generalizar de forma inmediata, todos los resultados presentes en la teoría clásica que se hayan demostrado utilizando <u>únicamente</u> el concepto de dependencia funcional.

2.5.2 Propiedades

Ahora, procedemos a estudiar las propiedades que cumple esta definición. Empezaremos analizando el comportamiento frente a las propiedades básicas descritas en la sección 2.4.2, y a continuación demostraremos la completitud de la correspondiente versión difusa de los axiomas de Armstrong.

Proposición 2.5.5 *(α, β)-d.f.d satisfacen i, ii.1, ii.2, iii.*

<u>Demostración</u>.

i. se verifica tomando $comp_i(a, b) = \delta_{ab}$ y $\alpha_i > 0$. Las demás propiedades están tácitamente asumidas en la definición de d.f.d y de semejanza entre dos valores cualesquiera, ya que el experto es libre en fijar los umbrales en cada atributo. Por ejemplo, respecto *ii.1*, si se establece un umbral de semejanza α y es $X(t_1) \not\simeq_\alpha X(t_2)$, entonces $comp(X(t_1), X(t_2)) < \alpha$, y por tanto no hay ninguna restricción sobre $comp(Y(t_1), Y(t_2))$.

ii.2 también se satisface ya que, con nuestra aproximación, $X(t_1)$ y $X(t_2)$ son suficientemente semejantes cuando $comp(X(t_1), X(t_2)) \geq \alpha$ y análogamente con Y. Pero no imponemos ninguna restricción sobre $comp(Y(t_1), Y(t_2))$ en relación al grado específico en el que $X(t_1)$ y $X(t_2)$ son semejantes. Esto permite que tuplas t_1 y t_2 verificando por ejemplo

$$comp(X(t_1), X(t_2)) = 1 \quad , \quad comp(Y(t_1), Y(t_2)) = 0.9$$

no violen la d.f.d (claro está con niveles $\alpha < 1$ y $\beta < 0.9$) □

En cuanto a las relaciones entre las distintas aproximaciones (en el supuesto de que usásemos las mismas relaciones de compatibilidad $comp_i$ en todas ellas), tenemos los siguientes resultados:

Teorema 2.5.6 *Si una relación satisface una RM-d.f.d entonces satisface una* (α, β)*-d.f.d con* $\alpha_i = \beta_j$ *constantes arbitrarias* $\forall i, j$

Demostración.

Sea λ una constante arbitraria en $[0, 1]$. Supongamos que

$$\overrightarrow{comp}\ (X(t_1), X(t_2)) = \{comp_i(A_i(t_1), A_i(t_2))\}_i \geq \alpha = \{\lambda\}_i$$

para arbitrarias t_1, t_2. Entonces,

$$comp_i(A_i(t_1), A_i(t_2)) \geq \alpha_i = \lambda \quad \forall\ i$$

de lo que obtenemos

$$\min_i comp_i(A_i(t_1), A_i(t_2)) \geq \lambda$$

Aplicando la definición de RM-d.f.d tenemos:

$$\min_j comp_j(A_j(t_1), A_j(t_2)) \geq \lambda$$

por lo que

$$\overrightarrow{comp}\ (Y) \geq \beta = \{\lambda\}_i$$

Por lo tanto, existe (α, β)–ffd con $\alpha_i = \beta_j = \lambda$ □

Teorema 2.5.7 *Si una relación satisface una Ch-d.f.d con grado* (ϕ) *entonces satisface una* $(\alpha, \beta)-$*d.f.d con* $\alpha_i = \beta_j = \phi \quad \forall\, i, j$

Demostración.

Sean α, β tales que $\alpha_i = \beta_j = \phi$. Supongamos que $\overrightarrow{comp}\ (X(t_1), X(t_2)) \geq \vec{\phi}$. Designemos por $comp(X)$ el valor real:

$$comp(X) \equiv \min_i comp_i(A_i(t_1), A_i(t_2))$$

Entonces, $comp(X) \geq \phi$. Consideremos los siguientes casos:

- Es posible que sea $comp(Y) \geq comp(X)$ ya que en este caso,

$$I(comp(X), comp(Y)) = 1 \geq \phi \quad \forall\, \phi$$

y no se rompe la Ch-d.f.d. Entonces

$$comp(Y) \geq comp(X) \geq \phi$$

por lo que

$$\min_j comp_j(A_j(t_1), A_j(t_2)) \geq \phi \ \Rightarrow \overrightarrow{comp}\ (Y(t_1), Y(t_2)) \geq \vec{\phi}$$

de lo que se deduce que t_1, t_2 no violan una $(\vec{\phi}, \vec{\phi})-$d.f.d

- Si $comp(Y) < comp(X)$ entonces $I(comp(X), comp(Y)) = comp(Y)$. Para que se satisfazga una Ch-d.f.d debe ser

$$comp(Y) \geq \phi$$

o lo que es lo mismo

$$\overrightarrow{comp}\ (Y(t_1), Y(t_2)) \geq \vec{\phi}$$

Por lo tanto, t_1, t_2 no rompen la $(\vec{\phi}, \vec{\phi})-$d.f.d

□

Podemos resumir las distintas proposiciones sobre las relaciones entre unas dependencias y otras la figura 2.6. Esto nos lleva a concluir que:

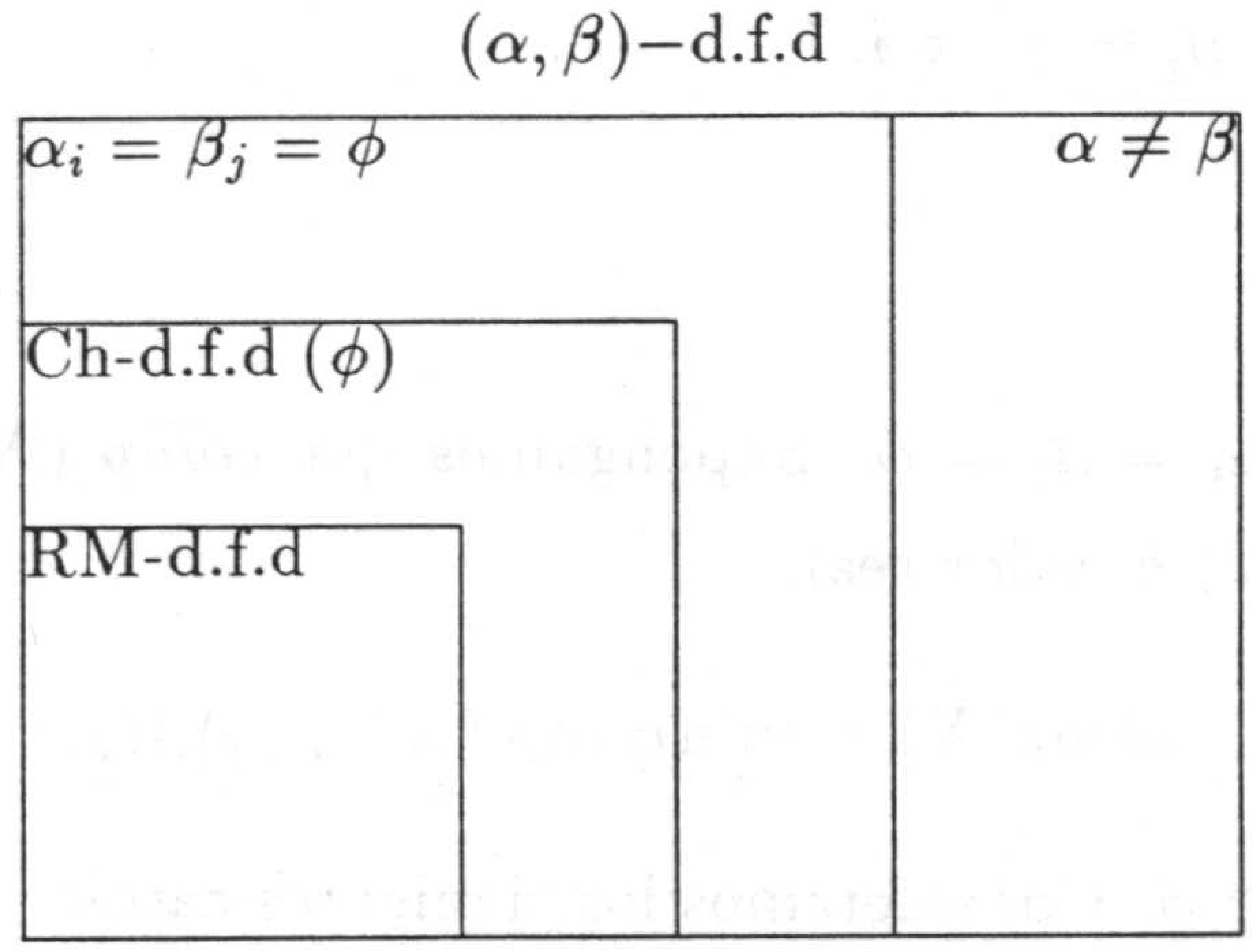

Figura 2.6. Relación entre las d.f.d existentes

- Detectamos dependencias difusas que no se detectaban con las aproximaciones existentes.

- Si no queremos perder todas las dependencias que se detectaban con las Ch-d.f.d y RM-d.f.d, entonces debemos imponer que sean $\forall i$ α_i = constantes arbitrarias y aún estamos detectando más dependencias.

Hay que hacer notar también, que otra ventaja de nuestra aproximación es que conlleva un esfuerzo computacional menor ya que sólo evaluamos $\overrightarrow{comp}\ (Y(t_1), Y(t_2))$ cuando es $\overrightarrow{comp}\ (X(t_1), X(t_2)) \geq \alpha$. Esto es posible, ya que hemos considerado unos umbrales fijos, por debajo de los cuales consideramos que no hay semejanza.

Una vez que hemos obtenido una definición coherente de dependencia difusa, procedemos a estudiar sus propiedades, tales como la derivación de nuevas d.f.d. Vimos que en el caso clásico y en las aproximaciones difusas dadas en [20] y [69], los axiomas de Armstrong ([4, 7]) son suficientes para obtener todas las dependencias derivadas a partir de un conjunto de dependencias F. Este es también el caso de las $(\alpha, \beta)-$d.f.d. Vamos a ver ahora las correspondientes versiones difusas de los axiomas de Armstrong utilizando la definición de $(\alpha, \beta)-$d.f.d. En la sección 1.2.2, revisamos el caso clásico definiendo la clausura F^+ de un conjunto de dependencias F. Esta definición se daba para realizar un estudio de las dependencias impuestas sobre un esquema relacional. No hay ningún problema en seguir esta misma filosofía, pero por comodidad ahora vamos a considerar una única relación. Así pues, por ejemplo, el axioma de transitividad lo pondremos en la forma:

$$\text{Si } X \overset{(\alpha_X,\alpha_Y)}{\longrightarrow} Y \text{ y } Y \overset{(\alpha_Y,\alpha_Z)}{\longrightarrow} Z \text{ entonces } X \overset{(\alpha_X,\alpha_Z)}{\longrightarrow} Z$$

que es una abreviatura de:

Si una relación r satisface la dependencia $X \overset{(\alpha_X,\alpha_Y)}{\longrightarrow} Y$ y $Y \overset{(\alpha_Y,\alpha_Z)}{\longrightarrow} Z$, entonces dicha relación también satisface la dependencia $X \overset{(\alpha_X,\alpha_Z)}{\longrightarrow} Z$

Axioma 2.1 (Reflexiva) *Si* $Y \subseteq X$ *entonces* $X \overset{(\alpha_X,\alpha_Y)}{\longrightarrow} Y$

Demostración.

Dados $\overrightarrow{comp}\ (X(t_1), X(t_2)) \geq \alpha_X$, entonces tomando los correspondientes atributos para el subconjunto Y de X, tenemos: $\overrightarrow{comp}\ (Y(t_1), Y(t_2)) \geq \alpha_Y$ □

Axioma 2.2 (Aumento) *Si* $X \overset{(\alpha_X,\alpha_Y)}{\longrightarrow} Y$ *entonces* $XZ \overset{(\alpha_{XZ},\alpha_{YZ})}{\longrightarrow} YZ$

Demostración.

$$(\overrightarrow{comp}\ (X(t_1), X(t_2)), \overrightarrow{comp}\ (Z(t_1), Z(t_2))) \geq (\alpha_X, \alpha_Z) \Rightarrow$$

$$\Rightarrow \overrightarrow{comp}\ (X(t_1), X(t_2)) \geq \alpha_X\ ,\ \overrightarrow{comp}\ (Z(t_1), Z(t_2)) \geq \alpha_Z$$

A partir de la primera dependencia obtenemos $\overrightarrow{comp}\ (Y(t_1), Y(t_2)) \geq \alpha_Y$ y por lo tanto

$$\overrightarrow{comp}\ (YZ(t_1), YZ(t_2)) \geq \alpha_{YZ}$$

tal y como queríamos demostrar. □

Axioma 2.3 (Transitiva) *Si* $X \overset{(\alpha_X,\alpha_Y)}{\longrightarrow} Y$ *y* $Y \overset{(\alpha_Y,\alpha_Z)}{\longrightarrow} Z$ *entonces* $X \overset{(\alpha_X,\alpha_Z)}{\longrightarrow} Z$

Demostración.

Si $\overrightarrow{comp}\ (X(t_1), X(t_2)) \geq \alpha_X$ entonces $\overrightarrow{comp}\ (Y(t_1), Y(t_2)) \geq \alpha_Y$. Por lo tanto, aplicando la definición de d.f.d tenemos $\overrightarrow{comp}\ (Y(t_1), Y(t_2)) \geq \alpha_Y$, y aplicando de nuevo la definición, deducimos finalmente que $\overrightarrow{comp}\ (Z(t_1), Z(t_2)) \geq \alpha_Z$. □

Se pueden demostrar también otras reglas de inferencia asociadas a los axiomas de Armstrong, como son:

Regla 4 (Descomposición) *Si* $X \overset{(\alpha_X,\alpha_{YZ})}{\longrightarrow} YZ$ *entonces* $X \overset{(\alpha_X,\alpha_Y)}{\longrightarrow} Y$ *y* $X \overset{(\alpha_X,\alpha_Z)}{\longrightarrow} Z$.

Demostración.

Si $\overrightarrow{comp}\ (X(t_1), X(t_2)) \geq \alpha_X$ entonces $\overrightarrow{comp}\ (YZ(t_1), YZ(t_2)) \geq \alpha_{YZ}$ y por lo tanto $\overrightarrow{comp}\ (Y(t_1), Y(t_2)) \geq \alpha_Y$ y $\overrightarrow{comp}\ (Z(t_1), Z(t_2)) \geq \alpha_Z$ □

Regla 5 (Unión) *Si* $X \overset{(\alpha_X,\alpha_Y)}{\longrightarrow} Y$ *y* $X \overset{(\alpha_X,\alpha_Z)}{\longrightarrow} Z$ *entonces* $X \overset{(\alpha_X,\alpha_{YZ})}{\longrightarrow} YZ$.

Demostración.

Si $\overrightarrow{comp}\ (X(t_1), X(t_2)) \geq \alpha_X$ entonces

$$\overrightarrow{comp}\ (Y(t_1), Y(t_2)) \geq \alpha_Y \text{ y } \overrightarrow{comp}\ (Z(t_1), Z(t_2)) \geq \alpha_Z$$

Por lo tanto, concluimos que $\overrightarrow{comp}\ (YZ(t_1), YZ(t_2)) \geq \alpha_{YZ}$ □

Regla 6 (Pseudotransitiva) *Si* $X \overset{(\alpha_X,\alpha_Y)}{\longrightarrow} Y$ *y* $YW \overset{(\alpha_{YW},\alpha_Z)}{\longrightarrow} Z$ *entonces* $XW \overset{(\alpha_{XW},\alpha_Z)}{\longrightarrow} Z$.

Demostración.

Si $\overrightarrow{comp}\ (XW(t_1), XW(t_2)) \geq \alpha_{XW}$ entonces tenemos

$$\overrightarrow{comp}\ (X(t_1), X(t_2)) \geq \alpha_X \text{ y } \overrightarrow{comp}\ (W(t_1), W(t_2)) \geq \alpha_W$$

Aplicando la definición de d.f.d obtenemos

$$\overrightarrow{comp}\ (Y(t_1), Y(t_2)) \geq \alpha_Y$$

por lo que

$$\overrightarrow{comp}\,(YW(t_1), YW(t_2)) \geq \alpha_{YW}$$

Aplicando de nuevo la definición, se tiene $\overrightarrow{comp}\,(Z(t_1), Z(t_2)) \geq \alpha_Z$ □

La pregunta que nos debemos hacer es ¿cuantos axiomas son necesarios para obtener, a partir de un conjunto de d.f.d F, todas las dependencias que se deriven de él? Al igual que ocurre en el caso clásico tenemos el siguiente importante resultado.

Teorema 2.5.8 *Consideremos un esquema relacional difuso con umbrales de semejanza estrictamente positivos $\alpha_i > 0$, y con medidas de compatibilidad verificando*

$$\forall\, A_i \in REL\,, \quad comp_i(a,a) = 1 \quad \forall\, a \in A_i$$

$$\forall\, A_i \in REL\,, \quad comp_i(a_{i1}, a_{i2}) = 0 \quad \text{para algún } a_{i1}, a_{i2} \in A_i$$

Entonces, A.1, A.2, A.3 constituyen un conjunto completo de axiomas de inferencia para la definición de $(\alpha,\beta)-$d.f.d.

La demostración de este teorema es exactamente la misma que la del caso clásico dada en [78] (que es también la base para Raju y Majumdar [69] y para Chen et al [20]). La idea de la demostración consiste en la construcción de una relación en la forma siguiente:

$$\begin{array}{cc} \overbrace{a_{11}\ldots a_{n1}}^{REL_1} & \overbrace{a_{m1}\ldots a_{k1}}^{REL_2} \\ a_{11}\ldots a_{n1} & a_{m2}\ldots a_{k2} \end{array}$$

dónde $REL_2 = REL - REL_1$, y REL_1 es el conjunto de atributos A_i tales que $X \overset{(\alpha_X,\beta_{A_i})}{\longrightarrow} A_i$ es una dependencia que puede derivarse de F por simple aplicación de los axiomas de Armstrong.

La restricción $comp_i(a_{i1}, a_{i2}) = 0$, obliga a que se viole la condición ($si\ X(t_1) = X(t_2)$) que constituye la premisa en la definición de la dependencia, al menos para las dos tuplas anteriores. Este hecho es la clave de la demostración.

Podemos concluir por tanto, que cualquier $(\alpha,\beta)-$d.f.d que se deduzca de F, puede ser derivada utilizando únicamente estos axiomas.

Nota. Hemos asumido que los umbrales se imponen por el experto y no se construyen a partir de los datos de las tuplas. Si estuviésemos interesados en estudiar (α, β)−d.f.d sin fijar α, β, entonces, para obtener un conjunto completo de reglas de inferencia, debemos considerar un axioma adicional:

Axioma 2.4 *Si* $X \stackrel{(\alpha,\beta)}{\longrightarrow} Y$ *entonces* $X \stackrel{(\alpha',\beta')}{\longrightarrow} Y \quad \forall\, \alpha', \beta'$ *con* $\alpha' \geq \alpha$ *y* $\beta' \leq \beta$.

La consistencia de A.4. se sigue de forma inmediata aplicando la definición de (α, β)−d.f.d

Finalmente, destaquemos que el algoritmo de clausura visto en la figura 1.2 del primer capítulo, sigue siendo aplicable con nuestra definición de dependencia funcional difusa (manteniendo fijos los umbrales). Esto se debe a que, en su demostración, únicamente se necesita la consistencia y completitud de los axiomas de Armstrong.

Ejemplo 2.5.9 . Consideremos un esquema simple REL=(`Edad, Salario`) con las siguientes tuplas:

	Edad	Salario
t_1	41	Alto
t_2	Adulto	2800
t_3	26	1200

con grados de compatibilidad para Edad:

$$comp(\texttt{Adulto,41})=1;\ comp(\texttt{Adulto,26})=0.3;\ comp(\texttt{41,26})=0.3$$

y para Salario:

$$comp(\texttt{Alto,2800})=0.8;\ comp(\texttt{Alto,1200})=0.3;\ comp(\texttt{2800,1200})=0.2$$

Es obvio que las anteriores tuplas no rompen una dependencia intuitiva del tipo

Personas con edad similar tienen salarios similares pero no es el caso para RM-d.f.d (t_1 y t_2 violan la dependencia) y para Ch-d.f.d (el fallo ocasionado por t_2, t_3 para cualquier $\phi \geq 0.3$). Sin embargo, estas tuplas no violan nuestra definición de d.f.d para umbrales razonables ($\alpha = \alpha_{\text{Edad}}$ y $\beta = \alpha_{\text{Salario}}$ no demasiado pequeños).

■

Como conclusión de este capítulo podemos afirmar que a la hora de establecer una definición de d.f.d, debemos realizarla atendiendo a las propiedades básicas descritas en la sección 2.4.2, lo que nos lleva a la definición 2.5.4 de las (α, β)−d.f.d. Hemos visto que, por ejemplo, seguimos conservando los axiomas de Armstrong. Sin embargo, no hemos abordado por ahora el problema de la eliminación de redundancia, o la definición de los operadores relacionales difusos, para establecer una teoría de diseño difusa. Estos temas no podrán ser resueltos con la definición 2.5.4, por lo que tendremos que restringirla. En primer lugar, vamos a ver que no pueden utilizarse medidas arbitrarias de comparación *comp*, sino que deben restringirse adecuadamente. Esto lo vemos en el capítulo tercero. Posteriormente, en el capítulo cuarto, veremos que para comprobar la existencia o no de una dependencia, no bastará con comparar tuplas dos a dos. También será necesario comparar los valores de aquellas tuplas cuyos antecedentes sean difusos con núcleos *solapados*.

Capítulo 3

Definición de las Medidas de Compatibilidad. El Nivel de Granularidad

En el apartado 2.5 del capítulo anterior, hemos definido una dependencia funcional difusa, en base a la comparación (a través de una medida de semejanza *comp*) de los valores de atributos antecedentes y consecuentes. Con este tratamiento, hemos visto la completitud de los axiomas de Armstrong. Sin embargo, dicha definición no es suficiente para elaborar una teoría completa de diseño difusa que nos permita demostrar teoremas fundamentales como el de descomposiciones sin pérdidas. Necesitaremos restringirla con la utilización de medidas de compatibilidad concretas, que serán las que se introducirán en este capítulo.

En primer lugar, plantearemos una aproximación heurística trabajando sobre los α-cortes de los valores difusos. Justificaremos intuitivamente la necesidad de comparar los valores de los atributos antecedentes a través de una medida de compatibilidad *débil*, mientras que deberemos utilizar una semejanza *fuerte* para los consecuentes. Así mismo, introduciremos el importante concepto de *granularidad*, que no es sino una restricción (no difusa) de integridad de identidad difusa sobre los atributos antecedentes y consecuentes presentes en una dependencia. Este desarrollo fué presentado en [28].

En el siguiente apartado (sección 3.2), propondremos una aproximación formal al problema planteado en la sección primera, introduciendo medidas de compatibilidad definidas en términos de posibilidad y necesidad. Estas serán justificadas a través de una *función difusa* que explique la dependencia.

Finalizaremos el capítulo analizando las analogías existentes entre las aproximaciones introducidas en las dos secciones anteriores. Veremos que la primera puede considerarse, bajo ciertas restricciones, un caso particular de la segunda. Los contenidos básicos de las dos últimas secciones fueron presentados en [29].

3.1 Planteamiento con Niveles de Precisión

En este apartado planteamos la aproximación heurística que mencionábamos en la introducción al capítulo. Empezaremos justificando semánticamente la necesidad de utilizar medidas de semejanza débil en los atributos antecedentes y fuerte para los consecuentes. Para definir dichas medidas, utilizaremos distintos niveles de precisión γ_i. Estos niveles nos permitirán representar un conjunto difuso a través de sus γ_i-cortes, de forma que compararemos los valores de un corte dado, a través de un nivel de semejanza dado.

Finalizaremos esta sección introduciendo el concepto del nivel de granularidad permitido en la base de datos. Utilizando esta definición, justificaremos la necesidad de imponer una restricción que nos garantice que los datos presentes en los atributos antecedentes y consecuentes de una d.f.d. no puedan ser *demasiado* difusos.

3.1.1 Introducción

Supongamos un esquema relacional sobre el que se impone una dependencia funcional difusa entre dos atributos A y B (para fijar ideas vamos a trabajar con atributos simples), que denotamos en general por:

$$A \rightsquigarrow B$$

Supongamos una relación con dicho esquema, y satisfaciendo la dependencia. Entonces, podemos concluir que a cada valor $A(t)$ le *corresponde* el valor $B(t)$, es decir:

$$A(t) \rightsquigarrow B(t) \quad \forall\, t \in r$$

Vamos a estudiar con detenimiento esta correspondencia, analizando la semántica que conlleva dicha afirmación cuando $A(t)$ y $B(t)$ son valores difusos. Para ello, vamos a seguir un ejemplo, considerando la siguiente tupla:

`(..., alto , pesado ,...)`

dónde $A(t)$=`alto` y $B(t)$=`pesado`. Supongamos que existe una dependencia difusa entre la `altura` y el `peso`. Por lo tanto, al valor concreto `alto` $\in D_{\texttt{Altura}}$ le corresponde el valor `pesado` $\in D_{\texttt{Peso}}$, es decir:

$$\texttt{alto} \rightsquigarrow \texttt{pesado} \tag{3.1}$$

Tal y como se interpretó en la ecuación 2.2, las anteriores etiquetas representan una distribución de posibilidad que restringe los posibles valores que pueden tomar $A(t)$ y $B(t)$ respectivamente. Por lo tanto, si suponemos que 185 y 186 son asignaciones con grado 1 a la etiqueta `alto`, y que 80 y 81 son asignaciones en el núcleo de `pesado` entonces, las siguientes conclusiones son lícitas:

$$185 \rightsquigarrow \texttt{pesado}\ ,\ 186 \rightsquigarrow \texttt{pesado}$$

mientras que no lo sería:

$$\texttt{alto} \rightsquigarrow \texttt{80} \bigwedge \texttt{alto} \not\rightsquigarrow \texttt{81}$$

ni tampoco la siguiente interpretación:

$$\texttt{186} \rightsquigarrow \texttt{80} \bigwedge \texttt{186} \not\rightsquigarrow \texttt{81}$$

ya que éstas últimas están extrayendo más información de la que se tenía realmente con los datos iniciales. En términos naturales, podemos concluir que:

$B(t)$ está determinado por $A(t)$ significa que: a **cada** *posible interpretación 'a' de $A(t)$ ($a \subseteq A(t)$) le corresponde* **cualquier** *posible interpretación de $B(t)$.*

lo cual puede representarse gráficamente como aparece en la figura 3.1.

3.1.2 Elección de las Medidas de Compatibilidad

En el apartado anterior, hemos dado una interpretación semántica del siguiente aserto:

El valor $B(t)$ está determinado por $A(t)$

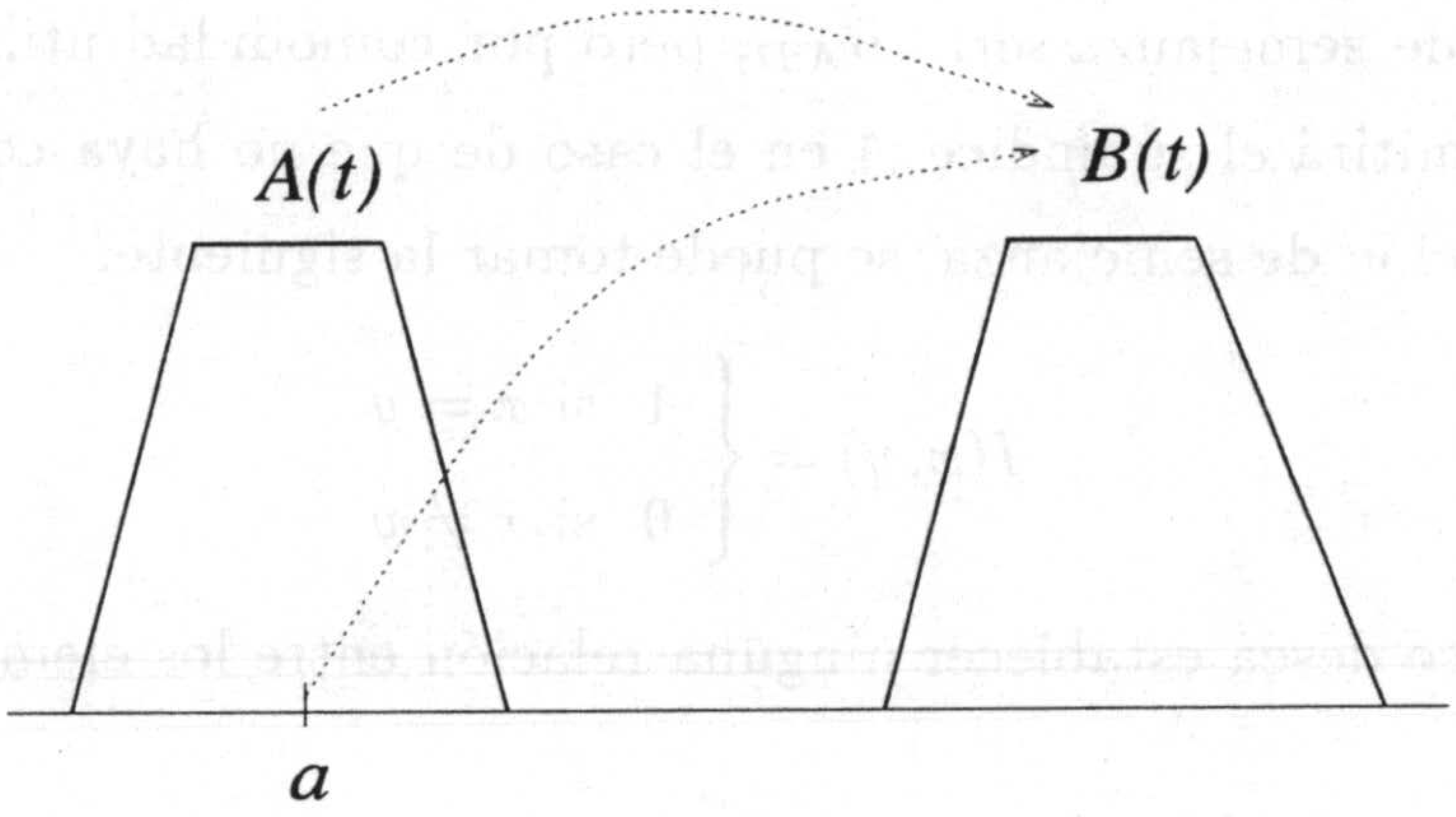

Figura 3.1. Correspondencia entre valores difusos

Pero para poder definir dependencia funcional entre dos conjuntos de atributos, debemos *comparar* dos pares de elementos $A(t_1)$, $A(t_2)$ y $B(t_1)$, $B(t_2)$. Trataremos primero el caso de atributos simples, y luego pasaremos al caso general de atributos compuestos.

◇ *Preliminares*

La idea básica que vamos a seguir para comparar dos datos difusos cualesquiera, es comparar los α-cortes de dichos conjuntos difusos. Por lo tanto, junto con cada atributo, debemos utilizar un conjunto de cortes, que formarán un vector de **niveles de precisión:**

$$\vec{\gamma}_A = \{\gamma_{j,A}\}_{j=1\ldots h'} \quad \text{con } 0 \leq \gamma_{j,A} \leq 1$$

Denotaremos $\vec{\gamma}_A$ por γ_A. Estos son los niveles que vamos a distinguir en A y podrían ser diferentes de aquellos asignados a otro atributo A'. Dependiendo del criterio del experto, podría decidir no incluir niveles por debajo de cierto umbral; en este caso, está perdiendo toda la información que conlleva el dato difuso, pero podría ser un precio que esté dispuesto a pagar. Por ejemplo, podría decidir que aquellos valores con grado de pertenencia menor que 0.4 no son significativos para sus propósitos. En cualquier caso, esta es una elección para el experto y no afecta al funcionamiento de nuestro modelo.

Por otra parte, recordemos que para aquellos atributos que admiten datos difusos como valores de tupla, existe una **relación de semejanza** asociada (definición 2.1.3), definida sobre los elementos del correspondiente dominio. Si el dominio del atributo A

es D, la relación de semejanza sería $R_{A,D}$, pero por comodidad utilizaremos la notación R_A. Incluso se omitirá el subíndice A en el caso de que no haya confusión. Como caso particular de relación de semejanza, se puede tomar la siguiente:

$$I(x,y) = \begin{cases} 1 & \text{si } x = y \\ 0 & \text{si } x \neq y \end{cases}$$

En este caso, no se desea establecer ninguna relación entre los elementos presentes en el dominio D.

Cuando introdujimos en la definición 2.5.1 un esquema relacional difuso, asociábamos a cada atributo A un único nivel de semejanza α_A. Sin embargo, ahora vamos a asociarle un vector de **niveles de semejanza**:

$$\vec{\alpha}_A = \{\alpha_{i,A}\}_{i=1\ldots h}$$

que denotaremos α_A en caso de que no haya confusión. La razón para ello, es que, tal y como comentábamos al inicio de este apartado, para comparar dos valores difusos vamos a comparar los γ_i-cortes de dichos conjuntos; por lo tanto, es lógico establecer distintos niveles de semejanza para los elementos de distintos cortes. A partir de ahora, y para simplificar notación, asumiremos que $h = h'$, es decir, que hay el mismo número de niveles de precisión y semejanza para un atributo dado. Así mismo, supondremos que hay el mismo número de niveles de precisión en cada atributo del esquema relacional.

Cuando estemos trabajando con atributos X e Y presentes en una dependencia, y para facilitar la nomenclatura en el caso de que no haya confusión, utilizaremos la siguiente notación:

Notación en Antecedentes : $\alpha \equiv \alpha_{i,X},\ \alpha_i \equiv \alpha_{i,X},\ \gamma \equiv \gamma_X,\ \gamma_i \equiv \gamma_{i,X}$

Notación en Consecuentes : $\beta \equiv \alpha_Y,\ \beta_i \equiv \alpha_{i,Y},\ \delta \equiv \gamma_Y,\ \delta_i \equiv \gamma_{i,Y}$

Al trabajar con un vector de niveles, tanto en atributos antecedentes, como en atributos consecuentes habrá que cambiar la definición 2.5.4 de dependencia funcional difusa, ya que esta se planteó con un único nivel de semejanza para cada atributo. Una primera extensión cuando trabajamos con niveles de precisión, y que posteriormente habrá que especificar completamente, podría ser la siguiente:

$$\textit{Para aquellos } i \textit{ tal que los } \gamma_i\textit{-cortes de } X(t_1)\ X(t_2) \textit{ sean semejantes a nivel } \alpha_i,$$
$$\textit{deben ser los } \delta_i\textit{-cortes de } Y(t_1) \text{ y } Y(t_2) \textit{ semejantes a nivel } \beta_i \qquad (3.2)$$

El término *semejante*, deberá especificarse posteriormente. Por otra parte, impondremos una restricción lógica en cuanto a la conexión entre los niveles de semejanza y precisión: con cada nivel de precisión debe haber asociado un nivel de semejanza en la forma:

$$\gamma_{j,A} \text{ está relacionado con } \alpha_{\gamma_{j,A}} \text{ de tal forma que } \gamma_{j,A} > \gamma_{j',A} \Rightarrow \alpha_{\gamma_{j,A}} \geq \alpha_{\gamma_{j',A}} \qquad (3.3)$$

Esta restricción nos asegura que a niveles de precisión menores, debemos considerar niveles de semejanza también menores. Por ejemplo, $\gamma_A = (1, 0.9, 0.7)$, $\alpha_A = (0.8, 0.7, 0.7)$. Obsérvese que, según esta restricción, estamos considerando en la ecuación 3.2 una dependencia del tipo:

Cuanto más semejantes sean los valores de los antecedentes,
más semejantes han de ser los valores de los consecuentes

Pasemos ahora a especificar completamente la ecuación 3.2. Empezaremos restringiendo los valores al caso crisp, y a partir de esta interpretación, extenderemos el estudio al caso de valores difusos cualesquiera. Trabajaremos primero con atributos simples y luego extenderemos los desarrollos al caso de atributos compuestos.

◇ *Caso: Valores Crisp*

En el caso de valores crisp, el criterio de semejanza utilizado en la ecuación 3.2, pasa a ser el siguiente:

Definición 3.1.1 *Diremos que un valor crisp x es* **semejante a nivel** α_i *con otro valor crisp y, ambos pertenecientes a un dominio D_A de un atributo A, si y solo si:*

$$R_A(x, y) \geq \alpha_i \qquad (3.4)$$

Ejemplo 3.1.2 . Supongamos una relación con las siguientes tuplas:

	A	B
t_1	24	61
t_2	23	61.3

dónde los niveles de semejanza del atributo vienen dados por

$$\alpha_1 = 1, \ \alpha_2 = 0.9 \ , \ \beta_1 = 0.9, \ \beta_2 = 0.85$$

Supongamos que

$$R_A(23, 24) = 0.9 \not\geq \alpha_1$$

Al no ser semejantes a nivel α_1, y siguiendo la ecuación 3.2, no se ha de imponer ninguna restricción sobre los valores de B. Es decir, no hay que imponer que sea $R_B(61, 61.3)$ mayor que β_1. Pero si consideramos el segundo nivel, entonces

$$R_A(23, 24) = 0.9 \geq \alpha_2$$

de forma que podemos concluir que 23 y 24 son dos valores semejantes a nivel α_2; entonces hay que imponer semejanza, al menos a nivel β_2, para los valores de B, es decir:

$$\text{Debe ser } R_B(61, 61.3) \geq \beta_2$$

■

Siguiendo este ejemplo podemos concluir que, en general, trabajando con valores crisp, la ecuación 3.2 quedaría en la forma:

$$\begin{aligned} &\textit{Para aquellos } i \textit{ tal que } R_A(A(t_1), A(t_2)) \geq \alpha_i \\ &\textit{entonces, debe ser } R_B(B(t_1), B(t_2)) \geq \beta_i \end{aligned} \tag{3.5}$$

◇ *Caso: Valor Crisp versus Valor Difuso en Antecedentes*

Utilizando la definición 3.1.1 de semejanza entre valores crisp, vamos a ver cómo hay que definir la semejanza entre dos valores difusos de atributos antecedentes, cuando tenemos un vector de niveles. Para ello, vamos a considerar el siguiente ejemplo:

Ejemplo 3.1.3 . Consideremos una relación con dos tuplas en la forma siguiente:

	A	B
t_1	entre 24 y 25	61
t_2	23	61.3

dónde "entre 24 y 25" es el difuso que aparece en la figura 3.2, con los mismos niveles de semejanza del ejemplo 3.1.2 y con los siguientes niveles de precisión:

$$\text{Atributo } A: \ \gamma_1 = 1,\ \gamma_2 = 0.8$$

$$\text{Atributo } B: \ \delta_1 = 1,\ \delta_2 = 0.8$$

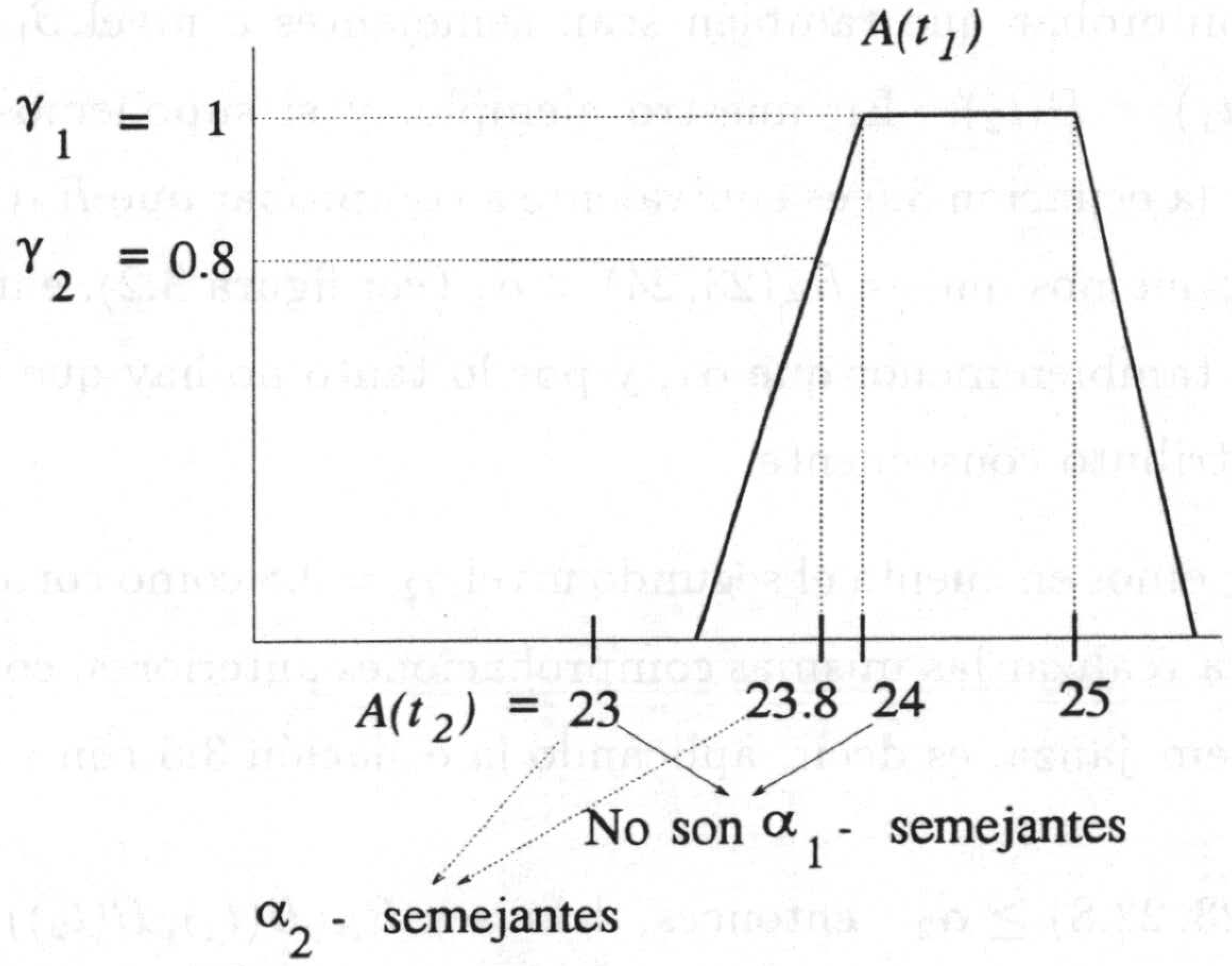

Figura 3.2. Semejanza entre crisp y fuzzy en antecedentes

Vamos a considerar el γ_1-corte de $A(t_1)$ y de $A(t_2)$, y vamos a comparar los elementos de dichos cortes. Posibles interpretaciones en el núcleo ($\gamma_1 = 1$) de $A(t_1)$ son $24, 24.5, \ldots, 25$. Con respecto a $A(t_2)$ la única interpretación es 23, ya que es un valor crisp. Consideremos estas asignaciones y a cada una de ellas le aplicamos la interpretación del caso crisp del apartado anterior. Aplicando la ecuación 3.5 con $i = 1$, obtendremos:

$$\begin{cases} \text{Si } R_A(23,24) \geq \alpha_1 & \text{entonces, debe ser } R_B(B(t_1),B(t_2)) \geq \beta_1 \\ & \vdots \\ \text{Si } R_A(23,24.5) \geq \alpha_1 & \text{entonces, debe ser } R_B(B(t_1),B(t_2)) \geq \beta_1 \\ & \vdots \\ \text{Si } R_A(23,25) \geq \alpha_1 & \text{entonces, debe ser } R_B(B(t_1),B(t_2)) \geq \beta_1 \end{cases}$$

Por lo tanto, en el momento que exista alguna posible asignación en el núcleo de $A(t_1)$ que sea semejante a nivel α_1 con $A(t_2)$, es decir

$$\sup\{R_A(23,24),\ldots,R_A(23,24.5),\ldots,R_A(23,25)\} \geq \alpha_1 \tag{3.6}$$

entonces debemos comprobar que también sean semejantes a nivel β_1 los valores crisp correspondientes $B(t_1)$ y $B(t_2)$. En nuestro ejemplo, y si suponemos que R_A es una distancia, comprobar la ecuación 3.6 es equivalente a comprobar que $R_A(23,24)$ sea mayor o igual que α_1. Si suponemos que es $R_A(23,24) < \alpha_1$ (ver figura 3.2), entonces el superior de la ecuación 3.6 es también menor que α_1, y por lo tanto no hay que imponer ninguna restricción sobre el atributo consecuente.

Sin embargo, si tenemos en cuenta el segundo nivel $\gamma_2 = 0.8$ como corte del difuso $A(t_1)$, entonces pasaríamos a realizar las mismas comprobaciones anteriores, considerando ahora el segundo nivel de semejanza, es decir, aplicando la ecuación 3.5 con $i = 2$:

$$\begin{cases} \text{Si } R_A(23,23.8) \geq \alpha_2 & \text{entonces, debe ser } R_B(B(t_1),B(t_2)) \geq \beta_2 \\ & \vdots \\ \text{Si } R_A(23,24) \geq \alpha_2 & \text{entonces, debe ser } R_B(B(t_1),B(t_2)) \geq \beta_2 \\ & \vdots \\ \text{Si } R_A(23,24.5) \geq \alpha_2 & \text{entonces, debe ser } R_B(B(t_1),B(t_2)) \geq \beta_2 \\ & \vdots \\ \text{Si } R_A(23,25) \geq \alpha_2 & \text{entonces, debe ser } R_B(B(t_1),B(t_2)) \geq \beta_2 \end{cases}$$

Si suponemos que $R_A(23,23.8) \geq 0.9(=\alpha_2)$, entonces, para no violar la dependencia, deberemos imponer que sea $R_B(61,61.3) \geq 0.85(=\beta_2)$ ■

En definitiva, del ejemplo anterior deducimos que hay que tomar como medida de compatibilidad para los antecedentes (a nivel de precisión γ_i), el máximo valor de semejanza entre $A(t_2)$ y los elementos del γ_i-corte de $A(t_1)$. Esto nos lleva a introducir la siguiente definición:

Definición 3.1.4 *Diremos que un valor crisp y es* **semejante débil a nivel de precisión γ_i y a nivel de semejanza** α_i *con un difuso F, ambos definidos sobre un atributo A, y lo representaremos por*

$$y \simeq^{d\ \gamma_i}_{\alpha_i} F$$

si y solo si:

$$comp^d_{\gamma_i}(y, F) \geq \alpha_i$$

dónde:

$$comp^d{}_{\gamma_i}(y, F) = \sup_{x \in F_{\gamma_i}} R_A(x, y)$$

Es inmediato comprobar que la definición 3.1.1 de semejanza entre valores crisp, es un caso particular de la definición anterior, ya que todos los cortes de un valor crisp coinciden con dicho valor. Por lo tanto, podemos afinar ahora las ecuaciones 3.2 y 3.5 (en el caso de que $A(t_1)$ sea un valor difuso y el resto sean crisp), en la forma siguiente:

$$\begin{gathered}\textit{Para aquellos } i \textit{ tal que } A(t_1) \simeq^{d\ \gamma_i}_{\alpha_i} A(t_2) \textit{ entonces,} \\ \text{debe ser: } R_B(B(t_1), B(t_2)) \geq \beta_i\end{gathered} \tag{3.7}$$

Reescribiendo el ejemplo 3.1.3 con esta terminología, obtendríamos:

$$A(t_1) \not\simeq^{d\ 1}_{1} A(t_2) \;\Rightarrow\; \text{no hay restricción sobre } B(t_1),\ B(t_2)$$

$$A(t_1) \simeq^{d\ 0.8}_{0.9} A(t_2) \;\Rightarrow\; \text{debe ser } R_B(B(t_1), B(t_2)) \geq 0.85$$

En el caso de que fuese $R_B(B(t_1, B(t_2)) \geq 0.85$, entonces t_1 y t_2 no violarían la d.f.d, mientras que en caso contrario, sí la violarían.

◇ *Caso: Valor Crisp versus Valor Difuso en Consecuentes*

Utilizando de nuevo la definición 3.1.1 de semejanza entre valores crisp, vamos a ver cómo hay que definir la semejanza entre dos valores difusos de atributos consecuentes. Para ello, vamos a considerar el siguiente ejemplo:

Ejemplo 3.1.5 . Consideremos ahora una relación con dos tuplas en la forma:

	A	B
t_1	20	entre 61 y 62
t_2	20	63

con los mismo niveles de precisión y semejanza del ejemplo 3.1.3:

$$\text{Atributo } A: \quad \alpha_1 = 1, \ \alpha_2 = 0.9, \ \gamma_1 = 1, \ \gamma_2 = 0.8$$

$$\text{Atributo } B: \quad \beta_1 = 0.9, \ \beta_2 = 0.85, \ \delta_1 = 1, \ \delta_2 = 0.8$$

La distribución de posibilidad "entre 61 y 62" (ver figura 3.3), restringe los posibles valores de $B(t_1)$.

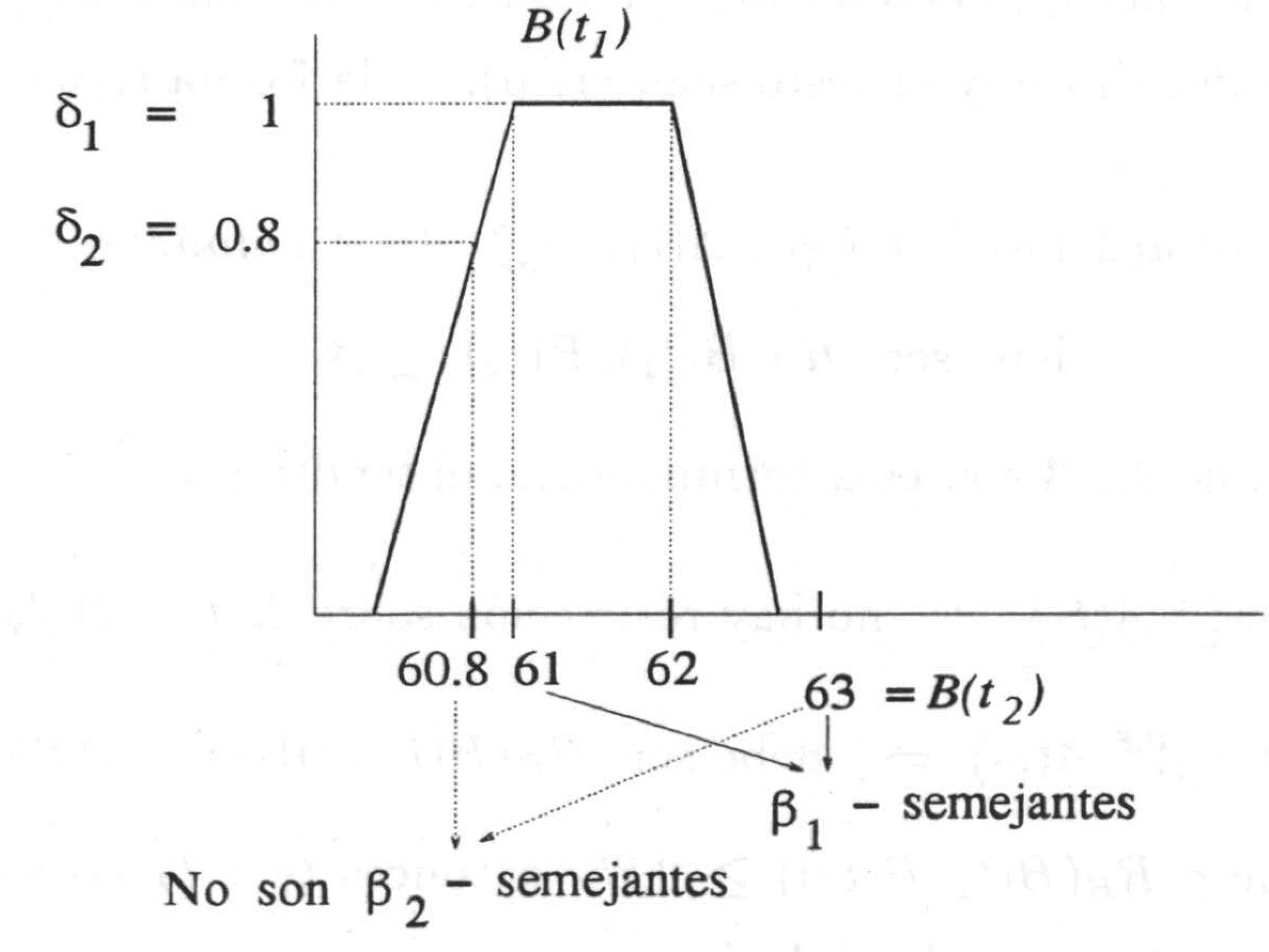

Figura 3.3. Semejanza entre crisp y fuzzy en consecuentes

Consideremos el primer nivel de precisión ($\gamma_1 = 1$). Entonces, posibles interpretaciones en el núcleo ($\gamma_1 = 1$) son $61, 61.5, \ldots, 62$. Recordando la interpretación semántica introducida en el apartado 3.1.1, al valor $A(t_1) = 20$ le corresponde cualquier posible asignación del valor $B(t_1) =$ "entre 61 y 62", es decir:

$$20 \rightsquigarrow 61$$

$$20 \rightsquigarrow 61.5$$

$$20 \rightsquigarrow 62$$

Por otra parte al valor $A(t_2) = 20$ le corresponde el elemento 63. Por lo tanto, al conjunto de asignaciones anteriores debemos añadir la siguiente:

$$20 \rightsquigarrow 63$$

Como se verifica que:

$$R_A(A(t_1), A(t_2)) = R_A(20, 20) = 1 \geq \alpha_1$$

entonces, retomando el caso crisp, para que exista dependencia según la ecuación 3.2, debemos imponer que todas las posibles asignaciones de "entre 61 y 62" sean semejantes al valor 63, o lo que es lo mismo:

$$\inf\{R_B(63, 61), \ldots, R_B(63, 61.5), \ldots, R_B(63, 62)\} \geq \beta_1 \tag{3.8}$$

En nuestro ejemplo, y si suponemos que R_B es una distancia, comprobar la ecuación 3.8 es equivalente a comprobar que $R_B(63, 61)$ sea mayor o igual que β_1. Si suponemos que es $R_B(63, 61) \geq \beta_1$ (ver figura 3.3), entonces el inferior de la ecuación 3.8 es también mayor o igual que β_1, y por lo tanto las tuplas t_1 y t_2 no violarían la dependencia.

Consideremos ahora el segundo nivel de precisión. El corte a nivel $\gamma_2 = 0.8$ de $A(t_1)$ y $A(t_2)$ es el mismo valor 20 y sigue siendo cierto que $R_A(A(t_1), A(t_2) \geq \alpha_2$. Ahora bien, si consideramos el segundo nivel de precisión, el corte a nivel $\delta_2 = 0.8$ de $B(t_1)$ es el intervalo $[60.8, 62.2]$, y el δ_2-corte de $B(t_2)$ es él mismo. Realizando el mismo razonamiento anterior, debemos imponer semejanza para todos estos valores, es decir:

$$\inf\{R_B(63, 60.8), \ldots, R_B(63, 61), \ldots, R_B(63, 61.5), \ldots, R_B(63, 62), \ldots, R_B(63, 62.8)\} \geq \beta_2$$

Si suponemos que $R_B(63, 60.8) < \beta_2$ (ver figura 3.3), entonces el inferior anterior es también menor que β_2, y por lo tanto las tuplas t_1 y t_2 sí violarían la dependencia.

■

En definitiva, del ejemplo anterior deducimos que tomamos como medida de compatibilidad para los consecuentes (a nivel de precisión δ_i), el mínimo valor de semejanza entre $B(t_2)$ y los elementos del δ_i-corte de $B(t_1)$. Esto nos lleva a introducir la siguiente definición:

Definición 3.1.6 *Diremos que un valor crisp y es* **semejante fuerte a nivel de precisión** δ_i **y a nivel de semejanza** β_i *con un difuso G, ambos definidos sobre un atributo B, y lo representaremos por*

$$y \simeq^{f\,\delta_i}_{\beta_i} G$$

si y solo si:

$$comp^f_{\delta_i}(y, G) \geq \beta_i$$

dónde

$$comp^f{}_{\delta_i}(y, G) = \inf_{x \in G_{\delta_i}} R_B(x, y)$$

Es inmediato comprobar que la definición 3.1.1 de semejanza entre valores crisp, es un caso particular de la definición anterior, ya que todos los cortes de un valor crisp coinciden con dicho valor. Se ha de observar que el criterio débil o fuerte atiende al carácter difuso de los datos, por lo que en la definición 3.1.1 de semejanza entre valores crisp, no se distingue entre débil y fuerte. Podemos afinar ahora las ecuaciones 3.2, 3.5 y 3.9, que quedarían en la forma siguiente:

$$\begin{array}{c} \textit{Para aquellos } i \textit{ tal que } A(t_1) \simeq^{d\,\gamma_i}_{\alpha_i} A(t_2) \textit{ entonces,} \\ \textit{debe ser } B(t_1) \simeq^{f\,\delta_i}_{\beta_i} B(t_2) \end{array} \tag{3.9}$$

Reescribiendo el ejemplo 3.1.5 con esta terminología, obtendríamos:

$$A(t_1) \simeq^{d\,1}_{1} A(t_2) \;\Rightarrow\; \text{debe ser} \;\; B(t_1) \simeq^{f\,1}_{0.9} B(t_2)$$

$$A(t_1) \simeq^{d\,0.8}_{0.9} A(t_2) \;\Rightarrow\; \text{debe ser} \;\; B(t_1) \simeq^{f\,0.8}_{0.85} B(t_2)$$

Como resulta que $B(t_1) \not\simeq^{f\,0.8}_{0.85} B(t_2)$, entonces las tuplas t_1 y t_2 violarían la dependencia funcional.

◇ *Caso: Valores Difusos en General*

Si aplicamos el mismo razonamiento que seguimos en los casos anteriores, podemos definir ahora:

Definición 3.1.7 *Diremos que un valor difuso F es* **semejante débil a nivel de precisión γ_i y de semejanza α_i** *con otro valor difuso F', ambos definidos sobre un atributo A, y lo representaremos por*

$$F \simeq_{\alpha_i}^{d\,\gamma_i} F'$$

si y solo si:

$$comp^d_{\gamma_i}(F, F') \geq \alpha_i$$

dónde:

$$comp^d{}_{\gamma_i}(F, F') = \sup_{x \in F_{\gamma_i},\, y \in F'_{\gamma_i}} R_A(x, y) \tag{3.10}$$

Ejemplo 3.1.8 . Supongamos los niveles $\alpha_1 = 0.8$, $\alpha_2 = 0.7$, $\gamma_1 = 1$, $\gamma_2 = 0.8$, y consideremos los difusos trapezoidales de la figura 3.4.

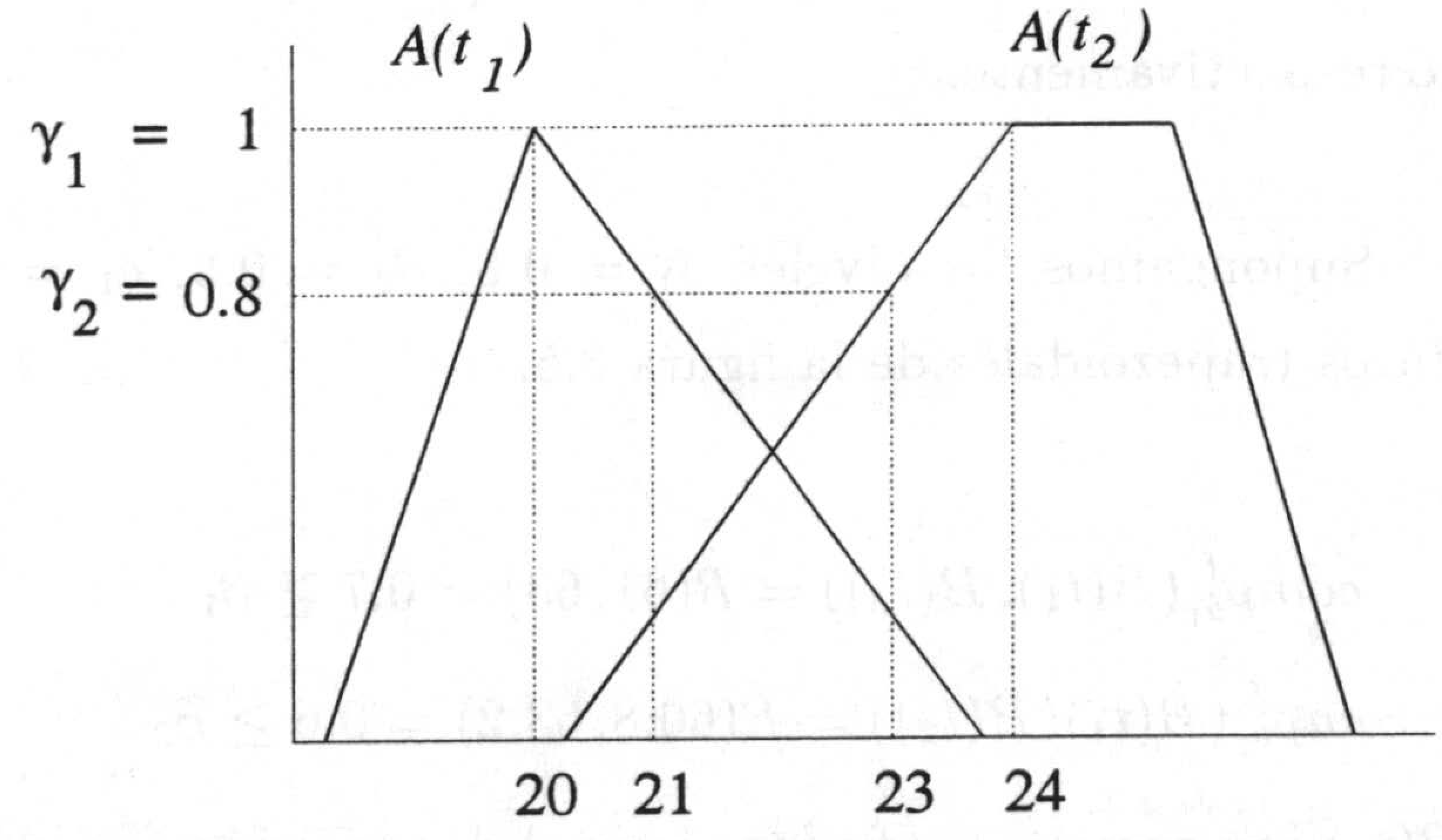

Figura 3.4. Semejanza entre difusos en antecedentes

Supongamos que:

$$comp^d_{\gamma_1}(A(t_1), A(t_2)) = R(20, 24) = 0.6 \not\geq \alpha_1$$

$$comp^d_{\gamma_2}(A(t_1), A(t_2)) = R(21, 23) = 0.8 \geq \alpha_2$$

Entonces, $A(t_1)$ y $A(t_2)$ son semejantes débiles a nivel de precisión γ_2 y de semejanza α_2, pero no a nivel de precisión γ_1 y semejanza α_1.

■

Definición 3.1.9 *Diremos que un valor difuso G es* **semejante fuerte a nivel de precisión** δ_i **y de semejanza** β_i *con otro valor difuso G', ambos definidos sobre un atributo B, y lo representaremos por*

$$G \simeq^{f\,\delta_i}_{\beta_i} G'$$

si y solo si:

$$comp^f_{\delta_i}(G, G') \geq \delta_i$$

dónde

$$comp^f{}_{\delta_i}(G, G') = \inf_{x \in G_{\delta_i}, y \in G'_{\delta_i}} R_B(x, y) \tag{3.11}$$

Nota. Las definiciones 3.1.7 y 3.1.9 son también válidas cuando el universo de discurso es discreto (finito). Unicamente cambian en que el superior y el ínfimo pasan a ser el máximo y el mínimo respectivamente.

Ejemplo 3.1.10 . Supongamos los niveles $\beta_1 = 0.8$, $\beta_2 = 0.6$, $\delta_1 = 1$, $\delta_2 = 0.7$, y consideremos los difusos trapezoidales de la figura 3.5.

Supongamos que:

$$comp^f_{\delta_1}(B(t_1), B(t_2)) = R(61, 63) = 0.7 \not\geq \beta_1$$

$$comp^f_{\delta_2}(B(t_1), B(t_2)) = R(60.8, 63.2) = 0.6 \geq \beta_2$$

Entonces, $B(t_1)$ y $B(t_2)$ son semejantes fuertes a nivel de precisión δ_2 y de semejanza β_2, pero no a nivel de precisión δ_1 y semejanza β_1.

■

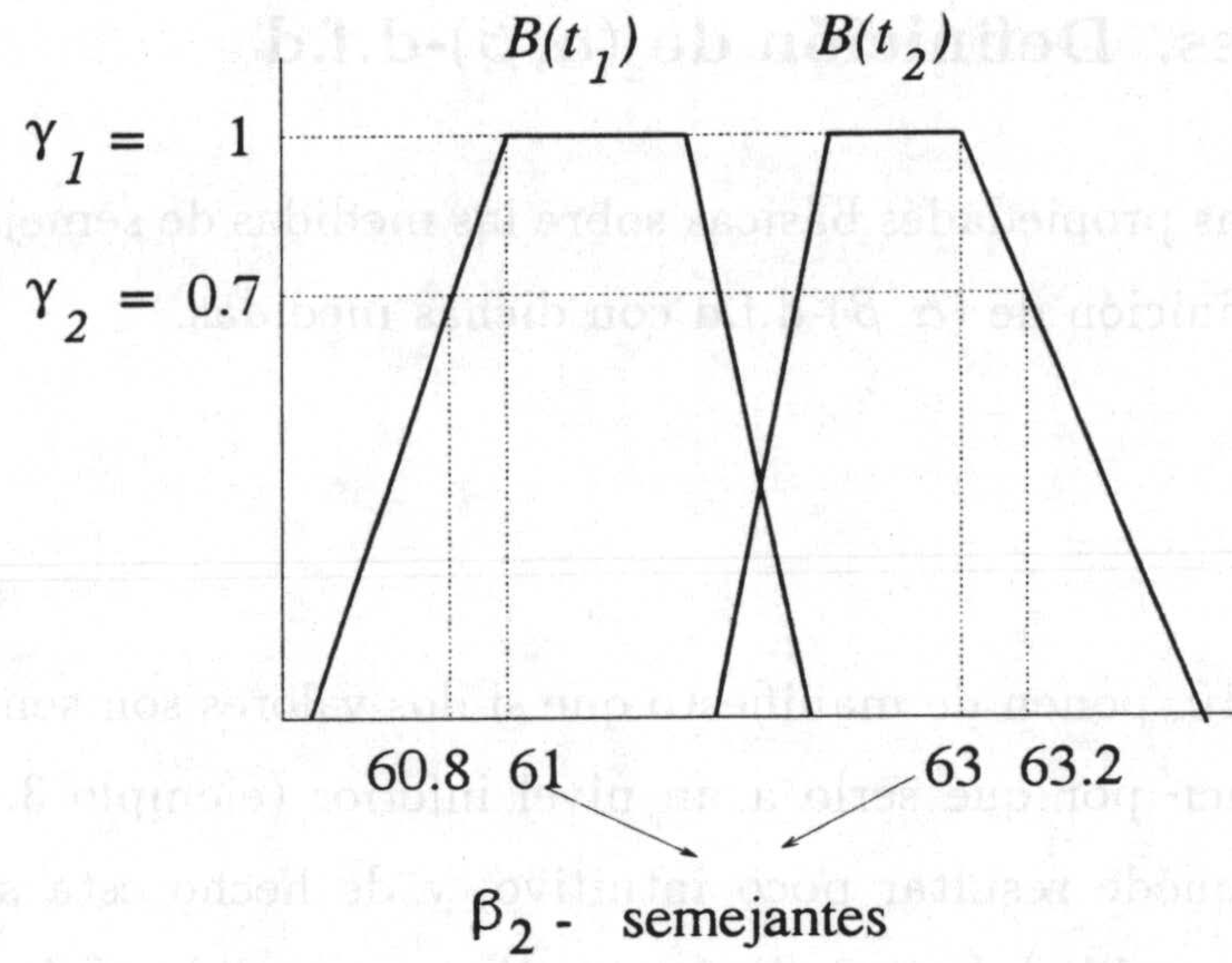

Figura 3.5. Semejanza entre difusos en consecuentes

◇ *Caso: Conjunto de Atributos en general*

Procedemos ahora a tratar el caso de atributos compuestos. Retomaremos la definición de semejanza fuerte y débil, y las impondremos en todos y cada uno de los atributos simples que los conforman.

Definición 3.1.11 *Sea X un conjunto de atributos $X = (A_k)_{k\in K}$, diremos que $X(t_1)$ y $X(t_2)$ son semejantes débiles a nivel de precisión γ_i y de semejanza α_i, dónde $\gamma_i = (\gamma_{i,A_k})_{A_k\in X}$ y $\alpha_i = (\alpha_{i,A_k})_{A_k\in X}$ y lo notaremos por*

$$X(t_1) \simeq_{\alpha_i}^{d\,\gamma_i} X(t_2)$$

si y solo si:

$$A_k(t_1) \simeq_{\alpha_{i,A_k}}^{d\,\gamma_{i,A_k}} A_k(t_2) \quad \forall\, A_k \in X \tag{3.12}$$

Definición 3.1.12 *Sea Y un conjunto de atributos $Y = (B_h)_{h\in H}$, diremos que $Y(t_1)$ y $Y(t_2)$ son semejantes fuertes a nivel de precisión δ_i y de semejanza β_i, dónde $\delta_i = (\delta_{i,B_h})_{B_h\in Y}$ y $\beta_i = (\beta_{i,B_h})_{B_h\in Y}$ y lo notaremos por*

$$Y(t_1) \simeq_{\beta_i}^{f\,\delta_i} Y(t_2)$$

si y solo si:

$$B_h(t_1) \simeq_{\beta_{i,B_h}}^{f\,\delta_{i,B_h}} B_h(t_2) \quad \forall\, B_h \in Y$$

3.1.3 Propiedades. Definición de (α, β)-d.f.d

Vamos a ver ahora algunas propiedades básicas sobre las medidas de semejanza introducidas, y cómo queda la definición de (α, β)-d.f.d con dichas medidas.

◇ *Propiedades*

Los ejemplos 3.1.5 y 3.1.10, ponen de manifiesto que si dos valores son semejantes fuertes a un nivel dado, no tienen por qué serlo a un nivel inferior (ejemplo 3.1.5) o superior (ejemplo 3.1.10). Esto puede resultar poco intuitivo, y de hecho esta será una de las causas que justifican la necesidad de introducir otro tipo de medidas más coherentes. En relación a la semejanza débil, los ejemplos 3.1.3 y 3.1.8, nos dicen que dos datos difusos semejantes débiles a un nivel, también lo son a un nivel inferior. Esto lo demostramos en la siguiente proposición:

Proposición 3.1.13 *Si* $F \simeq_{\alpha_i}^{d\,\gamma_i} F'$ *entonces* $F \simeq_{\alpha_j}^{d\,\gamma_j} F' \quad \forall\, \gamma_j \leq \gamma_i \quad \forall\, \alpha_j \leq \alpha_i$

<u>Demostración</u>.

Al ser $\gamma_j \leq \gamma_i$ se tiene

$$F_{\gamma_i} \subseteq F_{\gamma_j} \ , \ F'_{\gamma_i} \subseteq F'_{\gamma_j}$$

y por lo tanto:

$$\sup_{x \in F_{\gamma_j}, y \in F'_{\gamma_j}} R(x,y) \ \geq \ \sup_{x \in F_{\gamma_i}, y \in F'_{\gamma_i}} R(x,y)$$

pero el segundo término es mayor o igual que α_i por hipótesis; por otra parte, al ser $\gamma_j \leq \gamma_i$, y aplicando la restricción dada en la ecuación 3.3, se tiene que $\alpha_j \leq \alpha_i$. Encadenando desigualdades obtenemos:

$$\sup_{x \in F_{\gamma_j}, y \in F'_{\gamma_j}} R(x,y) \ \geq \ \alpha_j$$

□

Por otra parte, el siguiente resultado nos dice que, como cabía de esperar, el criterio de semejanza fuerte es más restrictivo que el de semejanza débil:

Proposición 3.1.14 *Si* $F \simeq^{f\,\gamma_i}_{\alpha_i} F'$ *entonces* $F \simeq^{d\,\gamma_i}_{\alpha_i} F'$

Demostración.

Inmediata, sin más que aplicar la propia definición de semejanza débil y fuerte. □

◇ *Definición de* (α, β)*-d.f.d*

En los apartados anteriores hemos ido detallando (ecuaciones 3.2, 3.7 y 3.9) cómo quedaría la definición 2.5.4 de (α, β)-dependencia funcional difusa, complementando la filosofía de la definición 2.5.4, con la utilización de cortes de precisión sobre los datos difusos. Formalmente, las ecuaciones 3.2, 3.7 y 3.9 son un caso particular de la siguiente definición:

Definición 3.1.15 *En las condiciones de la definición 2.5.4, una relación r satisface una* (α, β)*-d.f.d si y solo si todo par de tuplas* t_1*,* t_2 *satisface lo siguiente:*

$$\begin{aligned} &\text{Para aquellos } i \in \{1 \ldots h\} \text{ tales que } X(t_1) \simeq^{d\,\gamma_i}_{\alpha_i} X(t_2) \\ &\text{debe ser } \; Y(t_1) \simeq^{f\,\delta_i}_{\beta_i} Y(t_2) \end{aligned} \tag{3.13}$$

Ejemplo 3.1.16 . Si consideramos los valores $A(t_1)$ y $A(t_2)$ del ejemplo 3.1.8, y los valores $B(t_1)$ y $B(t_2)$ del ejemplo 3.1.10, entonces las tuplas t_1 y t_2 no violarían la dependencia funcional, ya que al primer nivel no hay que imponer ninguna restricción sobre los consecuentes ya que los antecedentes no son semejantes, y al segundo nivel tanto los antecedentes como los consecuentes sí son semejantes.

■

Nota. Si en la comprobación de la condición de la ecuación 3.13, resulta que:

$$X(t_1) \simeq^{d\,\gamma_i}_{\alpha_i} X(t_2)$$

entonces, aplicando la proposición 3.1.13 se tiene también que:

$$X(t_1) \simeq^{d\,\gamma_j}_{\alpha_j} X(t_2) \quad \forall\, \gamma_j \leq \gamma_i$$

Sin embargo:

$$Y(t_1) \simeq^{f\,\delta_i}_{\beta_i} Y(t_2) \not\Rightarrow Y(t_1) \simeq^{f\,\delta_j}_{\beta_j} Y(t_2)$$

por lo que se ha de comprobar que $Y(t_1)$ es semejante fuerte a nivel δ_j con $Y(t_2)$ para aquellos j tales que $\delta_j \leq \delta_i$.

3.1.4 El Nivel de Granularidad

Planteadas las medidas de semejanza asociadas a los atributos de una base de datos relacional difusa, y cómo pueden definirse dependiendo de la semejanza entre α-cortes, nos preguntamos ahora sobre cómo esta semejanza puede influir en las restricciones que se imponen a los valores difusos de los atributos. Es decir, nos planteamos si debemos imponer alguna restricción sobre el tipo de datos difusos que pueden aparecer en los atributos antecedentes y consecuentes de una dependencia funcional difusa. En el caso clásico sabemos que no puede aparecer el valor `desconocido` (ver sección 1.2.4). Ahora, veremos que los datos no pueden ser *demasiado difusos*, para lo cual definiremos la *granularidad* de un valor difuso. Posteriormente justificaremos este concepto, distinguiendo entre atributos antecedentes y consecuentes.

◇ *Definición de Granularidad*

Definición 3.1.17 *Diremos que un valor $A(t)$ satisface el* **nivel de granularidad por niveles** *impuesto en el atributo A, si se verifica:*

$$\inf_{d,d' \in A(t)_{\gamma_{j,A}}} R_A(d,d') \geq \alpha_{j,A} \quad \forall\, j$$

Al anterior término lo llamaremos

$$U(A(t)_{\gamma_{j,A}})$$

Restricción. *Sea A un atributo con niveles de precisión y semejanza dados por* $\gamma_{j,A}$ *y* $\alpha_{j,A}$ *respectivamente. Entonces* $A(t)$ *es un valor de tupla válido para un atributo antecedente o consecuente de una d.f.d, si y solo si* $A(t)$ *satisface el nivel de granularidad por niveles impuesto en A, es decir:*

$$U(A(t)_{\gamma_{j,A}}) \geq \alpha_{j,A} \quad \forall j$$

Así pues, la restricción de granularidad sobre un valor difuso F, nos dice que todos los elementos a, b en F con nivel de precisión igual a $\gamma_{1,A} = 1$, deben ser $\alpha_{1,A}$ semejantes ($R_A(a,b) \geq \alpha_{1,A}$); todos los elementos a, b con nivel de precisión $\gamma_{2,A}$ deben ser $\alpha_{2,A}$ semejantes ($R_A(a,b) \geq \alpha_{2,A}$), etc. En definitiva, esta restricción nos dice que los valores de tuplas de los atributos no pueden ser demasiado difusos, tal y como se ve en el siguiente ejemplo:

Ejemplo 3.1.18 . Consideremos un conjunto de niveles de precisión dado por 1, 0.9, 0.7 y de semejanza dado por 0.8, 0.7, 0.7. La figura 3.6 ilustra la restricción de granularidad.

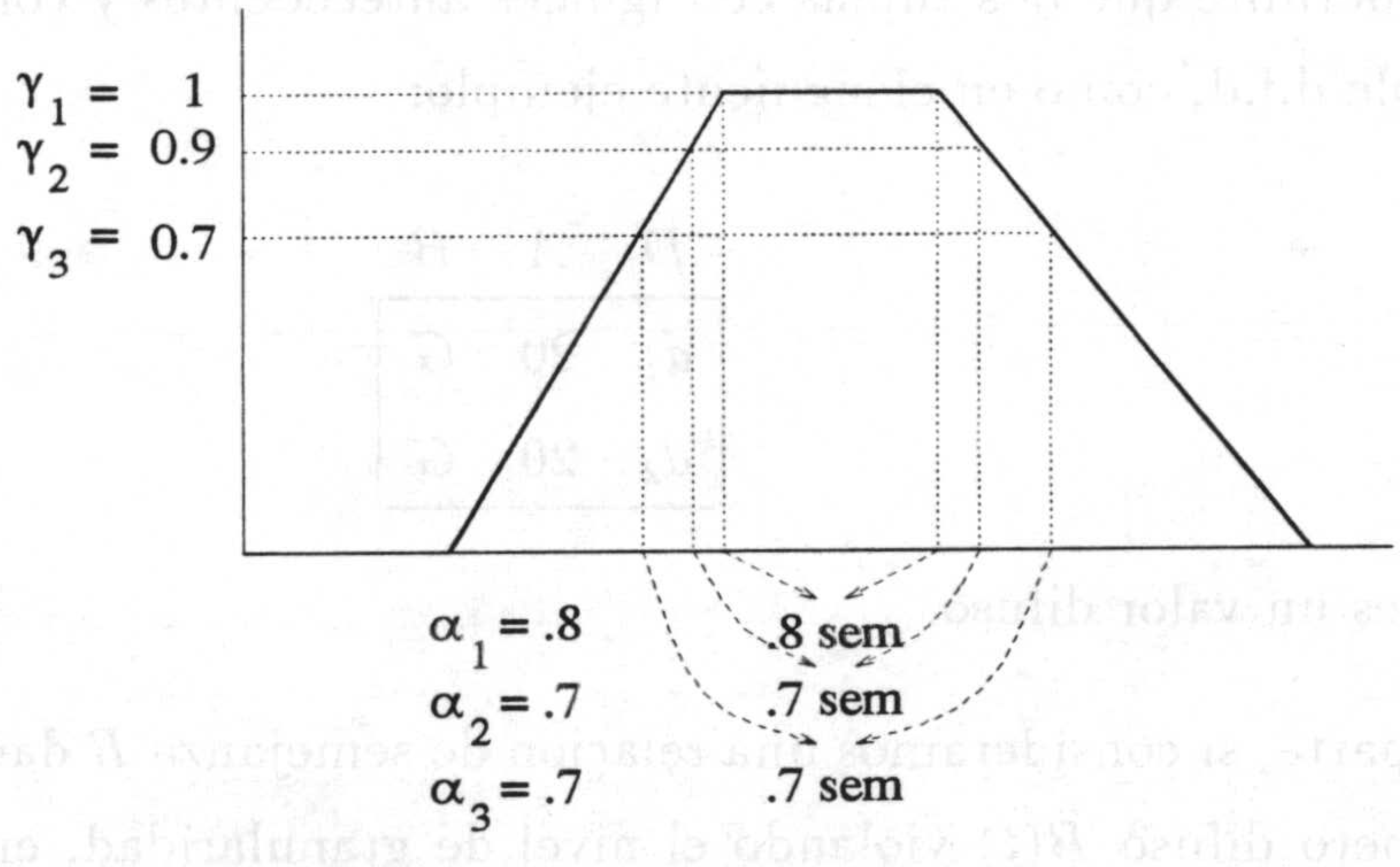

Figura 3.6. Restricción de Granularidad

■

.

Pasemos a justificar la restricción de granularidad; para ello vamos a considerar números difusos y una relación de semejanza dada por una distancia.

◇ *Justificación de la Granularidad en Consecuentes*

Vamos a justificar el nivel de granularidad utilizando los siguientes ejemplos:

1. Por una parte, la medida de semejanza débil definida en 3.1.7 es reflexiva, es decir:

   $$B(t) \simeq_{\beta_i}^{d\,\delta_i} B(t) \quad \forall\, t,\ \delta_i,\ \beta_i$$

   Sin embargo, no ocurre lo mismo para la semejanza fuerte, lo cual es inmediato comprobar tomando un difuso F y una relación de semejanza R de forma que:

   $$\exists\ x, y \in ker(F) \ \text{ tal que } R(x,y) < \beta_i$$

   es decir, tomando como F un valor *demasiado* difuso. Por lo tanto, si queremos obtener una medida de semejanza reflexiva, hemos de imponer la restricción de granularidad:

   $$B(t) \simeq_{\beta_i}^{f\,\delta_i} B(t) \quad \forall t,\ \delta_i,\ \beta_i$$

   Esto nos permite que dos tuplas con iguales antecedentes y consecuentes, no violen una posible d.f.d, como en el siguiente ejemplo:

D	A	B
d_1	20	G
d_2	20	G

   dónde G es un valor difuso.

2. Por otra parte, si consideramos una relación de semejanza R dada por una distancia, y un número difuso $B(t)$ violando el nivel de granularidad, entonces es inmediato comprobar que para cualquier otro número difuso $B(t') \not\subseteq B(t)$ se tiene:

   $$B(t) \not\simeq_{\beta_i}^{f\,\delta_i} B(t') \quad \forall\, t' \quad \forall\, \delta_i,\ \beta_i$$

   Es decir, toda tupla t' que tenga como valor de antecedente $A(t')$ un difuso semejante a $A(t)$, violaría la dependencia, al no ser semejantes (fuertes) los correspondientes valores del atributo consecuente ($B(t)$ y $B(t')$).

Es de destacar (para los dos puntos anteriores) que cuando no existe t' tal que $A(t) \simeq_{\alpha_i}^{d\,\gamma_i} A(t')$ entonces, la restricción de granularidad no es necesaria. En el modelo clásico, se sigue el mismo planteamiento. De hecho, el valor `desconocido` sólo puede aparecer como valor de un atributo consecuente si no hay en la base de datos otra tupla t' tal que $A(t) = A(t')$. Pero, si existe dicha tupla t', entonces `desconocido` no es una asignación válida para $B(t)$ si queremos mantener la dependencia. Esto se debe al hecho de que una dependencia funcional clásica no nos dice que "*cada $a \in D_A$ tiene un valor $b \in D_B$ asociado*" sino que "**cuando** *existen dos datos $A(t)$, $A(t')$ iguales, entonces $B(t)$ y $B(t')$ deben ser también iguales, y por lo tanto, cada $A(t)$ tendrá asociado un valor $B(t)$*". En el caso difuso, tenemos la misma interpretación, pero cambiando *iguales* por *semejantes.*

◇ *Justificación de la Granularidad en Antecedentes*

La justificación de la granularidad para los valores de un atributo antecedente no resulta tan imprescindible como para los consecuentes, pero sí es muy aconsejable por lo siguiente. Supongamos un difuso F violando la restricción de granularidad, y una tupla t con $A(t) = F$:

A	B
F	B_0
$A(t_1)$	B_1
$A(t_2)$	B_2
$\vdots$	$\vdots$

Entonces, el número de tuplas t_j con $A(t_j) \simeq_{\alpha_i}^{d\,\gamma_i} F$ aumenta considerablemente, por lo que habría que imponer semejanza fuerte entre todos los correspondientes consecuentes ($B_0 \simeq_{\beta_i}^{f\,\delta_i} B_1$, $B_0 \simeq_{\beta_i}^{f\,\delta_i} B_2$, e.t.c). El caso extremo se presenta cuando se permite como valor del atributo antecedente el valor `desconocido`. En este caso, tenemos:

$$A(t_j) \subseteq \texttt{desconocido} \Rightarrow A(t_j) \simeq_{\alpha_i}^{d\,\gamma_i} \texttt{desconocido} \quad \forall\, \gamma_i \quad \forall\, \alpha_i$$

Por lo tanto, para que se satisfazga la d.f.d, todos los valores del atributo(s) consecuente(s) deberían ser semejantes fuertes.

3.2 Planteamiento General

En el apartado anterior hemos abordado el problema de determinar cuales debieran ser las medidas de semejanza a considerar en la definición de una dependencia funcional difusa, a través de varios cortes efectuados sobre los distintos valores difusos. Dichas definiciones estaban justificadas desde un punto de vista semántico, tal y como se vió en el apartado 3.1.1; sin embargo no deja de ser una primera aproximación *heurística* al problema.

En este apartado vamos a plantear una aproximación formal al problema de la comparación de los valores de atributos presentes en una dependencia, de forma que, en la sección 3.3 podremos relacionar, bajo ciertas restricciones, ambas aproximaciones.

Empezaremos introduciendo el concepto fundamental de función difusa que explicará la semántica de una dependencia. Esto nos nos permitirá justificar la definición de las medidas de semejanza débil y fuerte que se proponen en las secciones 3.2.2 y 3.2.3. Posteriormente demostraremos una serie de propiedades que necesitaremos en capítulos posteriores. Veremos también que trabajar con un vector de niveles resulta más restrictivo que la utilización de un único umbral. Nosotros adoptaremos el segundo caso como marco de trabajo, de forma que demostraremos los teoremas fundamentales de los capítulos cuarto y quinto, bajo las condiciones más generales posibles.

3.2.1 El Concepto de Función Difusa

Vimos en el primer capítulo (ecuación 1.14) que una dependencia funcional clásica $X \rightarrow Y$ podía interpretarse como una función univaluada en la forma:

$$f: \; D_X(r) \rightarrow D_Y(r) \tag{3.14}$$

dónde $D_X(r)$ y $D_Y(r)$ eran los conjuntos de valores de los dominios D_X y D_Y que aparecían en la relación r.

Vamos a extender al caso difuso este concepto. Para fijar ideas trabajaremos con atributos simples y seguiremos el mismo ejemplo del apartado 3.1.1 en el que se establecía una relación entre la altura y el peso. Por ejemplo, la ecuación 3.1 establecía la siguiente relación entre dos valores concretos:

$$\texttt{alto} \rightsquigarrow \texttt{pesado}$$

Cabría esperar que pudiésemos modelar dicha correspondencia a través de una función. Claro está, el concepto de *función* a utilizar, debe concebirse con un planteamiento difuso. Llamemos f a la función difusa cuya forma y definición concreta habrá que determinar. Entonces, 3.1 es equivalente a:

$$f(\texttt{alto}) = \texttt{pesado}$$

Siguiendo el mismo razonamiento del apartado 3.1.1, las etiquetas `alto` y `pesado`, representan una distribución de posibilidad que restringe los posibles valores que pueden tomar $A(t)$ y $B(t)$ respectivamente. Por lo tanto, si suponemos que 185 y 186 son asignaciones con grado 1 a la etiqueta `alto`, y que 80 y 81 son asignaciones en el núcleo de `pesado` entonces, las siguientes conclusiones son lícitas:

$$f(185) = \texttt{pesado}\ ,\ f(186) = \texttt{pesado}$$

mientras que no lo sería:

$$f(\texttt{alto}) = \texttt{80} \bigwedge f(\texttt{alto}) \neq \texttt{81}$$

ni tampoco la siguiente interpretación:

$$f(\texttt{186}) = \texttt{80} \bigwedge f(\texttt{186}) \neq \texttt{81}$$

Este razonamiento nos lleva a introducir la siguiente definición de función difusa:

Definición 3.2.1 *Una* **función difusa** f *es una correspondencia:*

$$f : \widetilde{\mathcal{P}}(D) \longrightarrow \widetilde{\mathcal{P}}(E)$$

tal que, dados dos valores arbitrarios $A \in \widetilde{\mathcal{P}}(D)$, $B \in \widetilde{\mathcal{P}}(E)$:

$$f(A) = B \Leftrightarrow f(A') = B' \quad \forall\, A' \subseteq A \quad \forall\, B' \subseteq B \text{ con } A' \in \widetilde{\mathcal{P}}(D),\ B' \in \widetilde{\mathcal{P}}(E)$$

Esta definición se extiende de forma inmediata al caso de dominios compuestos en la forma:

$$f : \widetilde{\mathcal{P}}(D_1) \times \ldots \times \widetilde{\mathcal{P}}(D_n) \longrightarrow \widetilde{\mathcal{P}}(E_1) \times \ldots \times \widetilde{\mathcal{P}}(E_m)$$

$$f(A_1, \ldots, A_n) = (B_1, \ldots, B_m) \Leftrightarrow$$

$$\Leftrightarrow f(A'_1, \ldots, A'_n) = (B'_1, \ldots, B'_m) \quad \forall\, A'_i \subseteq A_i \quad \forall\, B'_j \subseteq B_j$$

Nota. En la literatura, existen un gran número de clases de funciones difusas verificando la definición 3.2.1. Por ejemplo, la clase de las *funciones difusas extendidas*, que se construyen a partir de una relación S en la forma:

$$f(A) = A \circ S$$

dónde $\circ$ representa composición sup-min. Estas funciones fueron introducidas por primera vez por Delgado en [35] y [36].

Si aplicamos la definición anterior en el marco relacional, podemos entonces extender la ecuación 3.14 mencionada al inicio de este apartado, en la forma siguiente: (por comodidad utilizamos en la notación atributos simples):

$$f : \widetilde{\mathcal{P}}(D_A)(r) \longrightarrow \widetilde{\mathcal{P}}(D_B)(r) \tag{3.15}$$

dónde $\widetilde{\mathcal{P}}(D_A)(r)$ representa el conjunto de valores difusos que aparecen en el atributo A de la relación r, y análogamente para B. Sería de esperar que dicha función fuese suficiente para establecer una dependencia entre dos atributos. Es decir, llegado este punto, nos preguntamos ¿es posible introducir el concepto de dependencia funcional difusa utilizando <u>únicamente</u> el concepto de función difusa? La respuesta es que no, y vamos a justificarla. Supongamos que establecemos la definición de dependencia funcional difusa entre dos atributos A y B, atendiendo únicamente a la ecuación 3.15. Para ser coherentes con la definición 3.2.1 tendríamos:

$$\forall\, t_1, t_2 \text{ Si } X(t_1) \subseteq X(t_2) \bigvee X(t_1) \supseteq X(t_2) \Rightarrow f(X(t_1)) = f(X(t_2)) \Leftrightarrow$$
$$\Leftrightarrow Y(t_1) \subseteq Y(t_2) \bigvee Y(t_1) \supseteq Y(t_2) \tag{3.16}$$

Sin embargo, en el caso de utilizar dicha definición, las tuplas de la relación siguiente

X	Y
10	20
10	20.001

violarían la dependencia. Esto no es un resultado aceptable, pues estamos violando la propiedad básica *ii.2.* establecida en la sección 2.4.2, que nos decía que tuplas con valores de antecedentes y consecuentes semejantes, no deberían romper la dependencia. En definitiva, la ecuación 3.16 no es válida como extensión difusa del concepto de dependencia, y debemos seguir la filosofía de las (α, β)-d.f.d y comprobar que valores *cercanos* en antecedentes les corresponden valores *cercanos* en consecuentes. Ahora bien, complementando el concepto de función difusa con dicha filosofía, podremos dar otra definición de d.f.d (en el capítulo cuarto), bajo la cual podremos demostrar teoremas fundamentales como el de descomposición sin pérdidas. Pero por ahora, no podemos introducirla ya que no disponemos de las herramientas necesarias, como por ejemplo, la definición de semejanza débil y fuerte, que pasamos a describir en el siguiente apartado.

3.2.2 Semejanza Débil

En este apartado vamos a proponer la definición de una medida de semejanza débil[1], en base al concepto de función difusa, y sin tener que recurrir a los γ_i-cortes (que sí eran necesarios en la aproximación heurística de la primera sección de este capítulo). Para su justificación, seguiremos un ejemplo tratando primero el caso crisp y luego el caso difuso en general. Finalmente, procederemos a la definición formal de la semejanza débil, y veremos algunas propiedades.

[1]Esta medida se utilizará para comparar los valores de los atributos antecedentes

$\diamond$ *Caso: Valores Crisp*

Definición 3.2.2 *Diremos que un valor crisp y es* **semejante a nivel** α *con otro valor crisp x, ambos definidos sobre un atributo A si y solo si:*

$$R_A(y, x) \geq \alpha \tag{3.17}$$

Consideremos una relación r en la que se espera una dependencia difusa en la forma

$$A \rightsquigarrow B$$

y supongamos dos tuplas t_1 y t_2 con valores crisp dados por $t_1|_{AB} = (a_1, b_1)$ y $t_2|_{AB} = (a_2, b_2)$. Siguiendo la filosofía de las (α, β)-d.f.d, en el caso de que a_1 y a_2 sean semejantes a nivel α, entonces también deben ser semejantes los valores b_1 y b_2. Si consideramos la siguiente interpretación:

$$b_1 = f(a_1)\ ,\ b_2 = f(a_2)$$

entonces, la restricción impuesta por la (α, β)-d.f.d se traduce en la forma siguiente:

$$\textit{Si } R_A(a_1, a_2) \geq \alpha \textit{ entonces debe ser } R_B(f(a_1), f(a_2)) \geq \beta \tag{3.18}$$

$\diamond$ *Caso: Valores Difusos en Antecedentes*

Supongamos ahora dos tuplas t_1 y t_2 con valores dados por $t_1|_{AB} = (a_1, b_1)$ y $t_2|_{AB} = (a_2, b_2)$, dónde los a_i son difusos y los b_i son crisp. De nuevo, consideramos que hay una función difusa f que explica la dependencia, por lo que tendremos:

$$f(a_1) = b_1\ ,\ f(a_2) = b_2$$

Pero ahora, los a_i son difusos, por lo que también tendremos las siguientes interpretaciones:

$$f(a'_1) = b_1 \quad \forall\ a'_1 \subseteq a_1\ ,\ f(a'_2) = b_2 \quad \forall\ a'_2 \subseteq a_2$$

Supongamos que existen dos valores crisp $a'_1/1$ y $a'_2/1$ que son semejantes a nivel α, es decir:

$$R_A(a'_1, a'_2) \geq \alpha$$

Entonces, estamos en condiciones de aplicar la ecuación 3.18 que se había establecido para valores crisp, por lo que tendremos que imponer que sea:

$$R_B(f(a'_1), f(a'_2)) \geq \beta \;\Leftrightarrow\; R_B(b_1, b_2) \geq \beta$$

Así pues, debemos comprobar hasta qué punto algún valor del núcleo de a_1 es semejante a algún valor del núcleo de a_2, en cuyo caso habrá que imponer semejanza en los valores consecuentes.

Este razonamiento lo hemos realizado con dos valores crisp a'_1 y a'_2 en el núcleo de los datos a_1 y a_2. De la misma forma debemos considerar elementos a'_i que estén en general en el soporte de a_i. Para dichos valores a'_i debemos tener en cuenta lo siguiente:

- Considerar hasta qué punto son semejantes a'_1 y a'_2. Este grado vendrá dado por $R_A(a'_1, a'_2)$.
- Considerar hasta qué punto cada elemento a'_i pertenece al difuso a_i. Estos grados vendrán dados por $\mu_{a_i}(a'_i)$.

Una medida que nos diga hasta qué punto algún valor en el soporte de a_1 es semejante a algún valor en el soporte de a_2, fue introducida por Prade y Testemale en [65, 41] y viene dada por la siguiente expresión:

$$\sup_{a'_1, a'_2} \{\mu_{a_1}(a'_1) \wedge \mu_{a_2}(a'_2) \wedge R_A(a'_1, a'_2)\} \tag{3.19}$$

En el caso de que dicha medida supere el umbral α, entonces habrá que imponer semejanza en los consecuentes, es decir:

$$R_B(f(a_1), f(a_2)) \geq \beta \;\Leftrightarrow\; R_B(b_1, b_2) \geq \beta$$

Definimos formalmente los conceptos introducidos en el siguiente apartado.

◇ *Definición*

Estamos ya en condiciones de definir semejanza débil. Lo haremos primero para el caso de semejanza entre un valor crisp y un valor difuso, y utilizaremos esta definición para plantear el caso general de dos valores difusos cualesquiera. Es por ello que daremos la definición de semejanza a través de una ecuación con distinta forma a la dada en la ecuación 3.19, pero veremos en la proposición 3.2.5 que ambos planteamientos son equivalentes.

Definición 3.2.3 *Diremos que un valor crisp y es* **semejante débil a nivel** α *con un difuso F, ambos definidos sobre un atributo A, y lo representaremos por*

$$y \simeq_{\alpha}^{d} F$$

si y solo si:

$$\sup_{x}(\mu_F(x) \wedge R_A(x,y)) \geq \alpha$$

Cuando no haya peligro de confusión omitiremos el subíndice que hace referencia al atributo sobre el que se está trabajando. En lo que sigue, utilizaremos la siguiente notación:

$$R_A^d(F,y) = \sup_{x}(\mu_F(x) \wedge R_A(x,y)) \tag{3.20}$$

por lo que la expresión de la definición anterior queda en la forma:

$$y \simeq_{\alpha}^{d} F \;\Leftrightarrow\; R_A^d(F,y) \geq \alpha \tag{3.21}$$

Análogamente, se extiende al caso de dos valores difusos en general, variando ahora la variable y en la ecuación 3.21, sobre el segundo conjunto difuso.

Definición 3.2.4 *Diremos que dos difusos F y F' definidos sobre un atributo A, son* **semejantes débiles a nivel** α *y lo representaremos por*

$$F \simeq_{\alpha}^{d} F'$$

si y solo si:

$$R_A^d(F,F') \geq \alpha \tag{3.22}$$

dónde:

$$R_A^d(F,F') = \sup_{y}(\mu_{F'}(y) \wedge R_A^d(F,y)) \tag{3.23}$$

Vamos a dar una forma más cómoda de trabajar con la ecuación 3.23

Proposición 3.2.5 $\boxed{R^d_A(F,F') = \sup_{x,y}(R_A(x,y) \wedge \mu_F(x) \wedge \mu_{F'}(y))}$

Demostración.

Al ser $\mu_{F'}(y)$ una función independiente de x, tenemos:

$$\begin{aligned}\mu_{F'}(y) \wedge R^d_A(F,y) &= \mu_{F'}(y) \wedge \{\sup_x(\mu_F(x) \wedge R_A(x,y))\} = \\ &= \sup_x(\mu_F(x) \wedge R_A(x,y) \wedge \mu_{F'}(y))\end{aligned}$$

por lo que la ecuación 3.23 queda como sigue:

$$\sup_y\left(\sup_x(\mu_F(x) \wedge R_A(x,y) \wedge \mu_{F'}(y))\right) = \sup_{x,y}(R_A(x,y) \wedge \mu_F(x) \wedge \mu_{F'}(y))$$

□

La medida de semejanza débil nos diría **hasta qué punto algún valor de F' está relacionado con algún elemento de F**. Vimos en la ecuación 2.3 del primer capítulo, que esta medida fué utilizada por Prade y Testemale [65, 41] para hallar el grado de compatibilidad en términos de *posibilidad*, entre dos valores de la base de datos $A(t)$ y $B(t)$ (misma tupla, atributos diferentes) en la evaluación de una pregunta o consulta a la base.

◇ *Propiedades*

Pasamos ahora a introducir una serie de propiedades sobre la semejanza débil que serán necesarias en la demostración de varios teoremas fundamentales que se verán en los últimos capítulos. Por comodidad en la notación, en el enunciado y demostración de dichas propiedades, se omitirá en R_A el subíndice que hace referencia al atributo correspondiente. En primer lugar, al no estar ordenadas las tuplas de una relación en la base de datos, debemos demostrar que la relación de compatibilidad débil es simétrica; efectivamente:

Proposición 3.2.6 *Para cualquier umbral α se verifica:*

i) $R^d(F, F') = R^d(F', F)$

ii) La medida de semejanza débil es simétrica, es decir, $F \simeq^d_\alpha F' \Leftrightarrow F' \simeq^d_\alpha F$

Demostración.

$$R^d(F, F') = \sup_{x,y}(\mu_F(x) \wedge \mu_{F'}(y) \wedge R(x, y))$$

Al ser R simétrica $R(x, y) = R(y, x)$ por lo que el anterior término es igual a:

$$\sup_{y,x}(\mu_{F'}(y) \wedge \mu_F(x) \wedge R(y, x)) = R^d(F', F)$$

Por lo tanto:

$$R^d(F, F') \geq \alpha \Leftrightarrow R^d(F', F) \geq \alpha$$

□

Además, la relación de semejanza débil es reflexiva, siempre y cuando existan elementos con un grado de pertenencia al difuso mayor que el nivel de semejanza establecido:

Proposición 3.2.7 *Si existe* $x_0 \in sop(F)$ *tal que* $\mu_F(x_0) \geq \alpha$*, entonces la medida de semejanza débil es reflexiva, es decir,* $F \simeq^d_\alpha F$

Demostración.

Por hipótesis, la medida de semejanza R utilizada en los elementos del dominio verifica que $R(x_0, x_0) = 1$. Por lo tanto, si aplicamos la hipótesis, obtenemos:

$$\mu_F(x_0) \wedge \mu_F(x_0) \wedge R(x_0, x_0) \geq \alpha$$

Como el anterior término es menor o igual que el superior en todos los elementos del dominio, deducimos que:

$$\sup_{x,\, y} \min\{R(x, y) \wedge \mu_F(x) \wedge \mu_F(y)\} \geq \alpha \Leftrightarrow F \simeq^d_\alpha F$$

□

Así pues, tenemos una semejanza reflexiva y simétrica. Es importante destacar que no es transitiva, por lo que no podemos establecer una partición del dominio a través de clases de equivalencia clásicas. Recordemos (sección 2.3) que esta fué una de las principales objeciones que hicimos del modelo de unificación de Buckles y Petry y de Shenoi y Melton. Veamos ahora otras propiedades de la medida de semejanza introducida:

Proposición 3.2.8 $F \simeq^d_\alpha F' \Rightarrow F \simeq^d_\phi F' \quad \forall\, \phi \leq \alpha$

Demostración.

Inmediata, aplicando la misma definición 3.2.4 □

Los siguientes resultados estudian la relación entre la semejanza débil y la inclusión difusa. Por ejemplo, la siguiente proposición nos dice que si un difuso H es semejante a otro valor difuso F, entonces también lo será a cualquier F' que contenga a F. Es de destacar que H **no** tiene por qué ser semejante débil a cualquier F' contenido en F.

Proposición 3.2.9

$$\left.\begin{array}{l} F \subseteq F' \\ F \simeq^d_\alpha H \end{array}\right\} \Rightarrow F' \simeq^d_\alpha H$$

Demostración.

$$F \subseteq F' \Rightarrow \mu_F(x) \leq \mu_{F'}(x) \quad \forall\, x$$

$$R^d(F', H) = \sup_{x,y}(\mu_{F'}(x) \wedge \mu_H(y) \wedge R(x,y)) \geq$$

$$\geq \sup_{x,y}(\mu_F(x) \wedge \mu_H(y) \wedge R(x,y)) = R^d(F, H) \geq \alpha$$

Así pues:

$$R^d(F', H) \geq \alpha$$

o lo que es lo mismo:

$$F' \simeq^d_\alpha H$$

□

Una consecuencia inmediata de este resultado es la siguiente proposición:

Proposición 3.2.10 *Si $F \subseteq F'$, $H \subseteq H'$ y $F \simeq^d_\alpha H$, entonces $F' \simeq^d_\alpha H'$*

La siguiente proposición es una relajación de la reflexividad, y nos dice que en las mismas condiciones de la proposición 3.2.7, cualquier difuso F incluido en otro valor F', es semejante a él.

Proposición 3.2.11 *Si existe $x_0 \in sop(F)$ tal que $\mu_F(x_0) \geq \alpha$, entonces,*

i) $\forall\ F,\ F'$ *tales que* $F \subseteq F' \Rightarrow F \simeq^d_\alpha F' \quad \forall\, \alpha$

ii) Si $F \subseteq F_1$ *y* $F \subseteq F_2$ *entonces* $F_1 \simeq^d_\alpha F_2$

Demostración.

Sólo vamos a probar *i)* ya que *ii)* se deduce de forma análoga, siguiendo el mismo esquema de demostración. Por hipótesis, tenemos:

$$\left.\begin{array}{l} \exists\ x_0 \ \text{ tal que } \mu_F(x_0) \geq \alpha \\ F \subseteq F' \ \Rightarrow\ \mu_F(x) \leq \mu_{F'}(x) \quad \forall\, x \end{array}\right\} \Rightarrow\ \mu_{F'}(x_0) \geq \alpha$$

Al ser R reflexiva $R(x_0, x_0) = 1$, por lo que:

$$\mu_{F'}(x_0) \wedge \mu_F(x_0) \wedge R(x_0, x_0) \geq \alpha \ \Rightarrow\ \sup_x(\mu_F(x) \wedge \mu_{F'}(y) \wedge R(x,y)) \geq \alpha \ \Leftrightarrow$$

$$\Leftrightarrow\ R^d(F, F') \geq \alpha \ \Leftrightarrow\ F \simeq^d_\alpha F'$$

$\square$

Proposición 3.2.12 *Sean tres difusos F, F' y H, y supongamos que $F \vee F' \simeq^d_\alpha H$. Entonces se tiene que o bien $F \simeq^d_\alpha H$, o bien $F' \simeq^d_\alpha H$*

Demostración.

Vamos a demostrarlo por reducción al absurdo. Supongamos pues que ninguno es semejante débil:

$$\sup_{x,y} \mu_F(x) \wedge \mu_H(y) \wedge R(x,y) < \alpha \ , \ \sup_{x,y} \mu_{F'}(y) \wedge \mu_H(y) \wedge R(x,y) < \alpha$$

Aplicando ahora la hipótesis, tenemos:

$$\sup_{x,y} \mu_{F \vee F'}(x) \wedge \mu_H(y) \wedge R(x,y) \geq \alpha$$

Ahora bien, el anterior término es igual a:

$$\sup_{x,y}\{(\mu_F(x) \vee \mu_{F'}(x)) \wedge \mu_H(y) \wedge R(x,y)\} =$$

$$= \sup_{x,y} \max\{\mu_F(x) \wedge \mu_H(y) \wedge R(x,y) \ , \ \mu_{F'}(x) \wedge \mu_H(y) \wedge R(x,y)\} \leq$$

$$\leq \max\left\{\sup_{x,y} \mu_F(x) \wedge \mu_H(y) \wedge R(x,y) \ , \ \sup_{x,y} \mu_{F'}(x) \wedge \mu_H(y) \wedge R(x,y)\right\} < \alpha$$

lo que contradice la hipótesis. □

Estas propiedades las vamos a aplicar sobre los difusos presentes en las tuplas de los atributos antecedentes. Por lo tanto, para poder aplicar las proposiciones 3.2.11 y 3.2.7, deberemos exigir que tengan al menos un elemento x_0 con grado de pertenencia mayor que el umbral de semejanza impuesto. Esta sería una restricción de integridad sobre los difusos que pueden aparecer como valores en los atributos antecedentes de una dependencia. Esta restricción es coherente, ya que, de hecho, cabría esperar que los valores difusos de cualquier atributo presente en una dependencia, estuviesen normalizados, es decir, con $\mu_{A(t)}(x_0) = 1$. Esto es lógico, ya que la correspondencia $A(t) \rightsquigarrow B(t) \quad \forall\, t \in r$, puede interpretarse como una *regla* en la forma *si* $A(t)$ *entonces* $B(t)$, y la premisa de dicha regla debería tener al menos medida de posibilidad igual a 1 ($\sup_x \mu_{A(t)}(x) = 1$)[1]. Imponemos por tanto la siguiente restricción:

> **Restricción de Integridad.** *Todo valor $A(t)$ de un atributo A antecedente de una dependencia funcional difusa, debe estar normalizado, es decir:*
>
> $$\sup_x \mu_{A^r(t)}(x) = 1 \quad \forall\, t \in r$$

[1] En el apartado 4.2.3, trataremos más extensamente esta interpretación relativa a las reglas

Nota. Posteriormente se justificará la necesidad de normalización, también para los atributos consecuentes (ver página 137)

3.2.3 Semejanza Fuerte

En este apartado vamos a proponer la definición de una medida de semejanza fuerte para comparar los valores de los atributos consecuentes. Nos basaremos en el concepto de función difusa, de forma que no necesitaremos recurrir a los γ_i-cortes (que sí eran necesarios en la aproximación heurística de la primera sección de este capítulo). Para su justificación, empezaremos tratando el caso crisp y luego lo extenderemos al caso difuso en general.

◇ *Caso: Valores Crisp*

Recordemos (definición 3.2.2) que al tratar con valores crisp se definía semejanza sin distinguir entre débil y fuerte. La razón, como ya se vió en el apartado 3.1 era que el criterio fuerte o débil está relacionado con los niveles de pertenencia de los valores difusos, por lo que no tiene sentido aplicar un criterio débil o fuerte a valores crisp. Este hecho sigue siendo válido en la aproximación que vamos a abordar en este punto. En cualquier caso, lo que sí tendrá sentido será graduar el grado de semejanza con varios niveles, pero esto se verá posteriormente.

Como ya vimos en la ecuación 3.18 del apartado 3.2.2, al trabajar con valores crisp, la definición de dependencia difusa quedaba en la forma:

$$\textit{Si } R_A(a_1, a_2) \geq \alpha \;\; \textit{entonces debe ser} \;\; R_B(f(a_1), f(a_2)) \geq \beta \tag{3.24}$$

dónde $a_i = A(t_i)$ y $f(a_i) = B(t_i)$, con A y B los atributos antecedente y consecuente respectivamente de la d.f.d.

◇ *Caso: Valores Difusos en Consecuentes*

Supongamos ahora dos tuplas t_1 y t_2 con valores dados por $t_1|_{AB} = (a_1, b_1)$ y $t_2|_{AB} = (a_2, b_2)$, dónde los a_i son valores crisp, y los b_i son difusos. De nuevo, consideramos que hay una función f que explica la dependencia, por lo que tendremos:

$$f(a_1) = b_1 \ , \ f(a_2) = b_2$$

Pero ahora, los b_i son difusos, por lo que también tendremos las siguientes interpretaciones:

$$f(a_1) = b'_1 \quad \forall\, b'_1 \subseteq b_1 \ , \ f(a_2) = b'_2 \quad \forall\, b'_2 \subseteq b_2$$

Supongamos que $R_A(a_1, a_2) \geq \alpha$. Entonces, si consideramos sólo valores crisp $b'_i \equiv b'_i/1 = f(a_i)$, se satisface la precondición de la ecuación 3.24, por lo que debemos imponer:

$$\begin{aligned} & R_B(f(a_1), f(a_2)) \geq \beta \ \Leftrightarrow \\ \Leftrightarrow \ & R_B(b'_1, b'_2) \geq \beta \quad \forall\, b'_1/1 \subseteq b_1 \quad \forall\, b'_2/1 \subseteq b_2 \end{aligned} \tag{3.25}$$

Este razonamiento lo hemos realizado con valores crisp b'_1 y b'_2 en el núcleo de los datos b_1 y b_2 (obviamente, si b'_i es un valor crisp que verifica $b'_i \subseteq b_i$, entonces b'_i está en el núcleo de b_i). De la misma forma debemos considerar elementos b'_i en el soporte. Para dichos valores debemos tener en cuenta lo siguiente:

- Considerar hasta qué punto son semejantes b'_1 y b'_2. Este grado vendrá dado por $R_B(b'_1, b'_2)$.
- Considerar hasta qué punto cada elemento b'_i pertenece al difuso b_i. Estos grados vendrán dados por $\mu_{b_i}(b'_i)$.

Según la ecuación 3.25, debemos imponer semejanza en <u>todos</u> los posibles valores en el núcleo. Si lo extendemos al soporte, y teniendo en cuenta los grados de pertenencia, una medida que nos dé hasta qué punto todos los elementos en el soporte de b_1 son semejantes a todos los elementos en el soporte de b_2, fue introducida por Prade y Testemale en [65, 41] y vendría dada por:

$$\inf_{b'_1, b'_2} \{(1 - \mu_{b_1}(b'_1)) \vee (1 - \mu_{b_2}(b'_1)) \vee R_A(b'_1, b'_2)\} \tag{3.26}$$

◇ *Definición*

Estamos ya en condiciones de definir semejanza fuerte. Lo haremos primero para el caso de semejanza entre un valor crisp y un valor difuso, y utilizaremos esta definición para plantear el caso general de dos valores difusos cualesquiera. Es por ello que daremos la definición de semejanza a través de una ecuación con distinta forma a la dada en la ecuación 3.26, pero veremos en la proposición 3.2.15 que ambos planteamientos son equivalentes.

Definición 3.2.13 *Diremos que un valor crisp y es* **semejante fuerte a nivel** β *con un difuso G, ambos definidos sobre un atributo B, y lo representaremos por*

$$y \simeq^f_\beta G$$

si y solo si:

$$\inf_x((1-\mu_G(x)) \vee R_B(x,y)) \geq \beta$$

En lo que sigue, vamos a utilizar la siguiente notación:

$$R^f_B(G,y) = \inf_x((1-\mu_G(x)) \vee R_B(x,y)) \tag{3.27}$$

por lo que la expresión de la definición anterior queda en la forma:

$$y \simeq^f_\beta G \;\Leftrightarrow\; R^f_B(G,y) \geq \beta \tag{3.28}$$

Análogamente, se extiende al caso de dos valores difusos en general, variando ahora la variable y en la ecuación 3.28, sobre el segundo conjunto difuso.

Definición 3.2.14 *Diremos que dos difusos G y G' definidos sobre un atributo B, son* **semejantes fuertes a nivel** β *y lo representaremos por*

$$G \simeq^f_\beta G'$$

si y solo si:

$$R^f_B(G,G') \geq \beta \tag{3.29}$$

dónde

$$R^f_B(G,G') = \inf_y((1-\mu_G(y)) \vee R^f_B(G,y)) \tag{3.30}$$

Vamos a dar una forma más cómoda de trabajar con la ecuación 3.30

Proposición 3.2.15 $\boxed{R^f_B(G,G') = \inf_{x,y}\{R_B(x,y) \vee (1-\mu_G(x)) \vee (1-\mu_{G'}(y))\}}$

Demostración.

Inmediata, siguiendo los mismos pasos que la demostración de la proposición 3.2.5

□

Esta medida de compatibilidad fuerte, nos daría **hasta qué punto todos los elementos de G' están relacionados con todos los elementos G.** Recordemos (ecuación 2.4 del primer capítulo) que Prade y Testemale [65, 41] utilizaron dicha medida para hallar el grado de compatibilidad en términos de *necesidad*, entre dos valores $A(t)$ y $B(t)$ (misma tupla, diferentes atributos) en el contexto de la evaluación de una pregunta a la base de datos.

Hay que destacar que esta medida tiene únicamente sentido cuando se aplica a difusos normalizados. Esto se ve claramente con el siguiente ejemplo: supongamos que $G = a/0.3$, $G' = b/0.2$, con $R(a,b) = 0$. En este caso tendríamos:

$$R^f(G,G') = \{0.7 \vee 1 \vee 0\} \wedge \{1 \vee 0.8 \vee 0\} = 1$$

y obviamente esto no es una medida de hasta qué punto todas las asignaciones de G están relacionadas con las de G'. Por lo tanto, si vamos a utilizar la medida fuerte para comparar los valores de los atributos consecuentes de una dependencia difusa, debemos imponer la siguiente restricción de granularidad que complementa a la que vimos en la página 133 para los atributos antecedentes:

Restricción de Integridad. *Todo valor $B(t)$ de un atributo B consecuente de una dependencia funcional difusa, debe estar normalizado, es decir:*

$$\sup_x \mu_{B^r(t)}(x) = 1 \quad \forall\, t \in r$$

◇ *El Nivel de Granularidad*

Vamos a revisar el concepto de granularidad que vimos en el apartado 3.1.4, utilizando ahora la definición de semejanza fuerte que acabamos de introducir. La restricción de granularidad se justificaba plenamente para los valores de los atributos consecuentes, ya que, de no hacerlo, la medida de semejanza fuerte no sería reflexiva y esto provocaría resultados anómalos en ciertas situaciones (ver página 120). Este hecho sigue siendo cierto con la actual definición de semejanza fuerte, así como la justificación para los atributos antecedentes. Definimos por tanto:

Definición 3.2.16 *Diremos que un valor de tupla $A(t)$ satisface el* **nivel de granularidad** *impuesto en un atributo A (con nivel de semejanza asociado α), si:*

$$A(t) \simeq^f_\alpha A(t) \tag{3.31}$$

Restricción de Integridad. *Sea A un atributo con nivel de semejanza asociado α. Entonces $A(t)$ es un valor de tupla válido para un atributo antecedente o consecuente, si y solo si $A(t)$ satisface el nivel de granularidad impuesto en A.*

Nota 1. Es de destacar que cuando la relación de semejanza impuesta en A es la identidad, entonces no se permite como dato ningún valor difuso. Esta imposición es coherente con la definición de d.f.d, ya que si definimos dependencia atendiendo al grado de semejanza entre los valores del dominio, y se supone que sólo son semejantes aquellos elementos iguales, entonces no podemos permitir que como valor de una tupla aparezca un difuso abarcando varios valores distintos.

Nota 2. En la última sección de este capítulo, relacionaremos esta definición de granularidad con la que introdujimos en la sección anterior, trabajando con cortes de precisión sobre los difusos. Intuitivamente, ambas restricciones nos vienen a decir lo mismo: que en los atributos antecedentes y consecuentes de una dependencia funcional difusa, no pueden existir valores *demasiado difusos*.

Las implicaciones en el diseño del nivel de granularidad son muy importantes. Vimos en el apartado 1.2.4, que en la teoría relacional clásica, "*no hay una teoría de diseño relacional* *totalmente satisfactoria en cuanto se incluyan valores nulos*". Además, esta afirmación debía leerse más correctamente (cuando se interpreta el valor nulo como `desconocido`) como: "*no hay una teoría de diseño relacional (clásica) en cuanto se incluyan valores difusos*". Nuestra extensión difusa será que "*no existe una teoría de diseño relacional difusa totalmente satisfactoria en cuanto se incluyan valores que violen el nivel de granularidad*". Desde luego, debemos desarrollar una metodología de diseño para demostrar que existe una teoría de diseño relacional difusa totalmente satisfactoria, pero este punto no se abordará hasta los capítulos cuarto y quinto de esta memoria.

◇ *Propiedades*

Pasamos ahora a demostrar algunos resultados que necesitaremos en los últimos capítulos de esta memoria. En lo que sigue, y si no hay confusión, omitiremos el subíndice que hace referencia al atributo en R_B. Es importante destacar que, aunque hemos justificado en el apartado anterior la necesidad de imponer restricciones de integridad de normalización y granularidad sobre los datos de los atributos antecedentes y consecuentes, no las vamos a asumir en los los resultados de esta sección. En su lugar, indicaremos explícitamente las condiciones necesarias míinimas para poder demostrar cada uno de los resultados. Eso sí, todas las condiciones se derivan de las restricciones de integridad.

El primer resultado nos dice que el criterio de semejanza fuerte es más restrictivo que el débil:

Proposición 3.2.17 *Sean G y G' dos difusos normalizados; entonces se verifica lo siguiente:*

i) $R^f(G, G') \geq R^d(G, G')$

ii) $G \simeq^f_\beta G' \;\Rightarrow\; G \simeq^d_\beta G'$

Demostración.

ii) es inmediato a partir de *i)*, así que pasamos a demostrar únicamente la parte primera. Si por hipótesis los difusos están normalizados, tenemos:

$$\exists\ x_0,\ y_0 \ \text{ tal que } \mu_G(x_0) = 1,\ \ \mu_{G'}(y_0) = 1 \ \Rightarrow$$

$$\Rightarrow\ \min\{\mu_G(x_0), \mu_{G'}(y_0), R(x_0, y_0)\} = R(x_0, y_0) \ \Rightarrow$$

$$\Rightarrow\ \sup_{x,y} \min\{\mu_G(x), \mu_{G'}(y), R(x,y)\} \geq R(x_0, y_0) \ \Leftrightarrow\ R^d(G, G') \geq R(x_0, y_0)$$

Pero, por otra parte:

$$1 - \mu_G(x_0) = 0\ ,\ 1 - \mu_{G'}(y_0) = 0 \Rightarrow$$

$$\Rightarrow\ \max\{1 - \mu_G(x_0), 1 - \mu_{G'}(y_0), R(x_0, y_0)\} = R(x_0, y_0) \ \Rightarrow$$

$$\Rightarrow\ \inf_{x,y} \max\{1 - \mu_G(x), 1 - \mu_{G'}(y), R(x,y)\} \leq R(x_0, y_0) \ \Leftrightarrow\ R^f(G, G') \leq R(x_0, y_0)$$

Encadenando desigualdades obtenemos la tesis pedida. □

Nota. En el caso de que los difusos no estuviesen normalizados, sólo podríamos obtener la siguiente relación:

$$Si\ \alpha > 0.5 \ \ y\ \ \exists\ x_0,\ y_0 \ \textit{tal que}\ \mu_G(x_0) \geq \alpha,\ \ \mu_{G'}(y_0) \geq \alpha \ \textit{entonces}$$

$$R^f(G, G') \geq \alpha \ \Rightarrow\ R^d(G, G') \geq \alpha$$

Por otra parte, dos difusos semejantes fuertes a un nivel, también lo son respecto a cualquier otro umbral menor. Recordemos que esta deseable propiedad, no se obtenía con la aproximación heurística por niveles introducida en la primera sección de este capítulo.

Proposición 3.2.18 $G \simeq^f_\beta G' \Rightarrow G \simeq^f_\phi G' \ \ \forall\ \phi \leq \beta$

Demostración.

Inmediata, aplicando la definición 3.2.14 □

El siguiente resultado nos dice que la medida fuerte es simétrica, lo cual es esencial ya que en la definición de dependencia funcional, no debe intervenir el orden en el que se consideran las tuplas. Por otra parte, recordemos que la semejanza fuerte no es en general reflexiva, pero la restricción de la granularidad obliga a que sí lo sea.

Proposición 3.2.19 *Para cualquier umbral β se verifica:*

i) $R^f(G,G') = R^f(G',G)$

ii) La medida de semejanza fuerte es simétrica, es decir, $G \simeq^f_\beta G' \;\Leftrightarrow\; G' \simeq^f_\beta G$

Demostración.

Inmediata, siguiendo los mismos pasos de la demostración de la proposición 3.2.6 □

Los siguientes resultados estudian el comportamiento de la semejanza fuerte frente a los operadores de inclusión y unión difusos. La proposición siguiente nos dice que dos subconjuntos cualesquiera de un conjunto difuso G' satisfaciendo el nivel de granularidad, son semejantes fuertes. Este resultado es intuitivo, ya que si G' satisface el nivel de granularidad, entonces todos los elementos de G' son semejantes a todos los elementos de G': como caso particular, se debe satisfacer también para cualquier par de subconjuntos de G'.

Proposición 3.2.20

$$\left.\begin{array}{l} G' \textit{ satisface el nivel de granularidad} \\ G, H \subseteq G' \end{array}\right\} \Rightarrow G \simeq^f_\beta H$$

Demostración.

Aplicando las hipótesis, tenemos:

$$G \subseteq G' \Rightarrow 1 - \mu_G(x) \geq 1 - \mu_{G'}(x) \quad \forall\, x$$

$$H \subseteq G' \Rightarrow 1 - \mu_H(x) \geq 1 - \mu_{G'}(x) \quad \forall\, x$$

Por lo tanto:

$$R^f(G,H) = \inf_{x,y}(R(x,y) \vee (1 - \mu_G(x)) \vee (1 - \mu_H(y))) \geq$$
$$\geq \inf_{x,y}(R(x,y) \vee (1 - \mu_{G'}(x)) \vee (1 - \mu_{G'}(x))) = R^f(G',G')$$

Como G' satisface el nivel de granularidad (β), se tiene que:

$$R^f(G', G') \geq \beta$$

por lo que:

$$R^f(G, H) \geq \beta$$

c.q.d. □

Una consecuencia inmediata de esta propiedad es la siguiente:

Proposición 3.2.21 *Sea $G \subseteq G'$ y G' satisface el nivel de granularidad. Entonces:*

i) G satisface el nivel de granularidad.

ii) $G \simeq^f_\beta G'$

Demostración.

Basta aplicar la proposición 3.2.20 tomando como H, el difuso G. □

En relación a la unión difusa, tenemos que si un conjunto difuso es semejante fuerte con otros dos, entonces también lo es con la unión difusa de dichos valores:

Proposición 3.2.22 *Si $G_1 \simeq^f_\beta G_2$ y $G_1 \simeq^f_\beta G_3$, entonces $G_1 \simeq^f_\beta G_2 \vee G_3$. En general, se tiene lo siguiente:*

Si $G_1 \simeq^f_\beta G_2$, $G_1 \simeq^f_\beta G_3$, $G_4 \simeq^f_\beta G_2$ y $G_4 \simeq^f_\beta G_3$, entonces $G_1 \vee G_4 \simeq^f_\beta G_2 \vee G_3$

Demostración.

Empezamos demostrando la primera parte:

$$R^f(G_1, G_2 \vee G_3) = \inf_{x,y}((1 - \mu_{G_1}(x)) \vee (1 - \mu_{G_2 \vee G_3}(y)) \vee R(x, y))$$

Como:

$$1 - \mu_{G_2 \vee G_3}(y) = \min(1 - \mu_{G_2}(y), 1 - \mu_{G_3}(y))$$

se tiene que:

$$R^f(G_1, G_2 \vee G_3) = \inf_{x,y}\{(1 - \mu_{G_1}(x)) \vee [(1 - \mu_{G_2}(y)) \wedge (1 - \mu_{G_3}(y))] \vee R(x,y)\} =$$

$$= \inf_{x,y}\{[(1-\mu_{G_1}(x)) \vee ((1-\mu_{G_2}(y)) \vee R(x,y)] \wedge [(1-\mu_{G_1}(x)) \vee ((1-\mu_{G_3}(y)) \vee R(x,y)]\} =$$

$$= R^f(G_1, G_2) \wedge R^f(G_1, G_3)$$

Por lo tanto:

$$R^f(G_1, G_2 \vee G_3) \geq \beta \;\Leftrightarrow\; \min\{R^f(G_1, G_2), R^f(G_1, G_3)\} \geq \beta$$

lo cual se verifica por hipótesis.

Para demostrar la segunda parte, basta considerar que acabamos de demostrar lo siguiente:

$$R^f(G_1, G_2 \vee G_3) = R^f(G_1, G_2) \wedge R^f(G_1, G_3)$$

En particular podemos cambiar G_1 por $G_1 \vee G_4$, por lo que tendremos:

$$R^f(G_1 \vee G_4, G_2 \vee G_3) = R^f(G_1 \vee G_4, G_2) \wedge R^f(G_1 \vee G_4, G_3) =$$

$$= R^f(G_1, G_2) \wedge R^f(G_4, G_2) \wedge R^f(G_1, G_3) \wedge R^f(G_4, G_3)$$

Como cada uno de dichos términos es mayor o igual que β, tendremos que

$$R^f(G_1 \vee G_4, G_2 \vee G_3) \geq \beta$$

tal y como queríamos demostrar. □

Como aplicación de esta proposición, tenemos la siguiente:

Proposición 3.2.23 *Sean G y G' satisfaciendo el nivel de granularidad. Supongamos que $G \simeq^f_\beta G'$. Entonces:*

i) $(G \vee G') \simeq^f_\beta G$

ii) $(G \vee G') \simeq^f_\beta G'$

iii) $(G \vee G') \simeq^f_\beta (G \vee G')$, *es decir,* $G \vee G'$ *satisface el nivel de granularidad.*

Demostración.

Al ser $G \simeq^f_\beta G'$ y $G \simeq^f_\beta G$, aplicando la proposición 3.2.22 se tiene:

$$G \simeq^f_\beta (G \vee G')$$

Aplicando ahora que $G \simeq^f_\beta G'$ y que $G' \simeq^f_\beta G'$ y de nuevo la proposición 3.2.22, se tiene:

$$G' \simeq^f_\beta (G \vee G')$$

Aplicando de nuevo el resultado 3.2.22 con $G_1 = G \vee G'$ tenemos finalmente que:

$$(G \vee G') \simeq^f_\beta (G \vee G')$$

□

Demostramos en la proposición 3.2.10 que cualquier par de conjuntos que contuviesen a dos valores difusos semejantes débiles, tambien eran semejantes. Esta relación no es obviamente cierta con la semejanza fuerte. Sin embargo sí podemos demostrar que son semejantes fuertes cualquier par de subconjuntos de dos valores difusos semejantes fuertes:

Proposición 3.2.24 *Si* $G \simeq^f_\beta G'$ *y* $H \subseteq G$, $H' \subseteq G'$ *entonces* $H \simeq^f_\beta H'$

Demostración.

$$R^f(H,H') = \inf_{x,y}(R(x,y) \vee (1 - \mu_H(y)) \vee (1 - \mu_{H'}(x)))$$

como $H \subseteq G$ y $H' \subseteq G' \Rightarrow \quad \forall\, y \;\; \mu_H(y) \leq \mu_G(y), \quad \forall\, x \;\; \mu_{H'}(x) \leq \mu_{G'}(x)$ por lo que:

$$R^f(H,H') \geq \inf_{x,y}(R(x,y) \vee (1 - \mu_G(y)) \vee (1 - \mu_{G'}(x))) \;=\; R^f(G,G') \geq \beta$$

c.q.d □

Con ayuda de este resultado, vamos a demostrar ahora el recíproco de la proposición 3.2.22:

Proposición 3.2.25 *Si* $G_1 \simeq^f_\beta G_2 \vee G_3$ *entonces* $G_1 \simeq^f_\beta G_2$ *y* $G_1 \simeq^f_\beta G_3$.

Demostración.

Al estar $G_2 \subseteq G_2 \vee G_3$, aplicando la proposición 3.2.24 se tiene que

$$G_1 \simeq^f_\beta G_2$$

y análogamente se demuestra para G_3 □

3.2.4 Definición de Dependencia Funcional Difusa

En el anterior apartado se han definido las medidas de semejanza a utilizar en la comparación de los valores de atributos antecedentes y consecuentes. Ahora vamos a extender la definición 2.5.4 de (α, β)-dependencia funcional difusa, utilizando estas medidas. En primer lugar nos planteamos una definición utilizando un único nivel de semejanza y, posteriormente extendemos el estudio con un vector de niveles. En cualquier caso, hay que reseñar que no vamos a ver ninguna propiedad sobre estas definiciones de dependencia, ya que en el próximo capítulo justificaremos la necesidad de introducir otra definición más restrictiva. De todas formas, hemos creido conveniente realizar en esta sección, un primer y breve apunte de cómo se utilizan las medidas de semejanza débil y fuerte en la comprobación de una dependencia.

◇ *Planteamiento con un único Nivel*

La extensión de la definición 2.5.4 de (α, β)-d.f.d es inmediata cuando únicamente consideramos un nivel de semejanza, ya que sólo debemos especificar las medidas de comparación a utilizar. Por lo tanto, nos quedaría:

Definición 3.2.26 *En las condiciones de la definición 2.5.4, una relación r satisface una (α,β)-d.f.d si y solo si todo par de tuplas t_1, t_2 satisface lo siguiente:*

$$\textit{Si } X(t_1) \simeq^d_\alpha X(t_2) \;\; \textit{entonces debe ser } \; Y(t_1) \simeq^f_\beta Y(t_2) \tag{3.32}$$

Podemos representar gráficamente el proceso de definición de una (α,β)-d.f.d como aparece en la figura 3.7.

$$\boxed{\text{relación } r} \longrightarrow \boxed{\begin{array}{c}\text{Comprobación: } \exists \text{ función difusa?} \\ f : \tilde{\mathcal{P}}\,(D_A(r)) \rightarrow \tilde{\mathcal{P}}\,(D_B(r))\end{array}} \xrightarrow{SI}$$

$$\xrightarrow{SI} \boxed{\begin{array}{l}\text{Comprobación: ¿Las tuplas con antecedentes semejantes} \\ \text{débiles, tienen consecuentes semejantes fuertes?}\end{array}} \xrightarrow{SI} \boxed{\begin{array}{l}\text{Conclusión:} \\ r \text{ satisface } \;\; A \xrightarrow{(\alpha,\beta)} B\end{array}}$$

Figura 3.7. Razonamiento seguido para establecer la definición 3.2.26 de (α,β)-d.f.d

◇ *Planteamiento con un Vector de Niveles*

Ahora, vamos a plantear en qué términos se interpreta la ecuación 2.13 de la definición 2.5.4 de (α,β)-dependencia funcional difusa, cuando utilizamos un vector de niveles de semejanza, que lo notaremos por:

$$\alpha_{i,X} = (\alpha_{i,X_k})_{X_k \in X} \;,\;\; \alpha_{i,X_k} \in [0,1] \;\;\; i = 1 \dots h$$
$$\beta_{i,Y} = (\beta_{i,Y_h})_{Y_h \in Y} \;,\;\; \beta_{i,Y_h} \in [0,1] \;\;\; i = 1 \dots h$$

Como medidas de compatibilidad utilizamos las semejanzas débil y fuerte. Una primera solución al problema, la vimos cuando estudiamos la aproximación heurística: la definición 3.1.15 nos decía lo siguiente:

$$\textit{Para aquellos } i \in \{1 \dots h\} \textit{ tales que } X(t_1) \simeq^{d\,\gamma_i}_{\alpha_i} X(t_2)$$
$$\textit{debe ser } \;\; Y(t_1) \simeq^{f\,\delta_i}_{\beta_i} Y(t_2)$$

Pero ahora, no trabajamos con niveles de precisión, por lo que el anterior planteamiento debería cambiarse por el siguiente:

$$\textit{Para aquellos } i \in \{1 \ldots h\} \textit{ tales que } X(t_1) \simeq^d_{\alpha_{i,X}} X(t_2)$$

$$\textit{debe ser } Y(t_1) \simeq^f_{\beta_{i,Y}} Y(t_2)$$

dónde

$$X(t_1) \simeq^d_{\alpha_{i,X}} X(t_2) \Leftrightarrow X_k(t_1) \simeq^d_{\alpha_{i,X_k}} X_k(t_2) \quad \forall\, X_k \in X$$

$$Y(t_1) \simeq^f_{\beta_{i,Y}} Y(t_2) \Leftrightarrow Y_h(t_1) \simeq^f_{\beta_{i,Y_h}} Y_h(t_2) \quad \forall\, Y_h \in Y$$

En este caso, el criterio de semejanza débil o fuerte está graduado a través de los ditintos niveles de semejanza de cada atributo. Ahora, dos valores $X(t_1)$ y $X(t_2)$ pueden ser semejantes débilmente a grado α_2 pero no a grado α_1 ($\alpha_1 > \alpha_2$). En este caso, y al igual que se razonó en el apartado 3.1, los correspondientes valores del atributo consecuente, únicamente deben restringirse a que sean β_2 semejantes fuertemente. Por otra parte, es de destacar que si fuese

$$X(t_1) \simeq^d_{\alpha_i} X(t_2) \text{ para cierto } i$$

entonces aplicando la proposición 3.2.8, tendríamos que

$$X(t_1) \simeq^d_{\alpha_j} X(t_2) \quad \forall\, \alpha_j < \alpha_i$$

pero bastaría comprobar que

$$Y(t_1) \simeq^f_{\beta_i} Y(t_2)$$

ya que aplicando la proposición 3.2.18 se tendría que:

$$Y(t_1) \simeq^f_{\beta_j} Y(t_2) \quad \forall\, \beta_j < \beta_i$$

Hay que destacar que al permitir un vector de niveles por atributo, no se está suavizando la definición de dependencia funcional difusa, sino todo lo contrario: esto se debe, como acabamos de ver, a que al añadir niveles de semejanza, valores de tuplas que no eran semejantes a un nivel, sí lo son respecto a otro nivel menor, y por lo tanto, deben restringirse los valores de tupla de los correspondientes atributos consecuentes. Obsérvese que si tomamos el siguiente conjunto de umbrales:

$$\forall\, X \quad \alpha_X = \min_i\{\alpha_{i,X}\}$$

entonces, cualesquiera par de valores $X(t_1)$ y $X(t_2)$ verifican:

$$\exists\ i \ \text{ tal que } X(t_1) \simeq^d_{\alpha_{i,X}} X(t_2) \quad \Leftrightarrow \quad X(t_1) \simeq^d_{\alpha_X} X(t_2)$$

Ahora bien, para los correspondientes valores de atributos consecuentes sólo se tiene la implicación en un sentido:

$$Y(t_1) \simeq^f_{\alpha_{j,Y}} Y(t_2) \ \Rightarrow \ Y(t_1) \simeq^f_{\alpha_Y} Y(t_2)$$

por lo que, efectivamente, la aproximación con un vector de niveles resulta más restrictiva, y veremos en el capítulo quinto que no es necesaria dicha restricción para poder desarrollar una teoría de diseño difusa. Por lo tanto, a partir de ahora, trabajaremos con un único umbral de semejanza para cada atributo.

3.3 Relación entre las dos Aproximaciones

En este apartado vamos a dar una caracterización de las semejanzas fuerte y débil definidas a través de las ecuaciones 3.22 y 3.29 vistas en la aproximación formal (sección 3.2), que nos permitirá además relacionarlas con las ecuaciones 3.10 y 3.11 presentadas en la aproximación por niveles (sección 3.1). Veremos que, considerando un nivel de precisión adecuado, ambas son equivalentes, y que el tratamiento con un único nivel de semejanza es menos restrictivo que con un vector de niveles, por lo que nos centraremos en el primero.

Una primera restricción que hemos de imponer es que, en el caso continuo, las funciones de pertenencia y la relación de semejanza sean continuas en sus variables y tomen valores en un conjunto compacto (cerrado y acotado), lo cual no representa ninguna restricción real en la práctica. En el caso discreto no es necesario añadir ninguna restricción adicional. Las restricciones de normalización en antecedentes y consecuentes vistas en la sección anterior, no serán necesarias en las demostraciones que seguiremos, aunque sí necesitaremos la existencia de valores con grado de pertenencia mayor que el umbral de semejanza impuesto (obviamente, la restricción de normalización sería una condición suficiente). Hemos preferido no incluir dicha restricción, ya que la finalidad de esta sección es demostrar una equivalencia entre distintas definiciones de medidas de semejanza, y por tanto creemos que resulta más conveniente acometer este objetivo en un marco lo más general posible, independizándolo del estudio de las dependencias funcionales difusas.

En lo que sigue, omitiremos por comodidad el subíndice que hace referencia al atributo en R_A, R_B.

3.3.1 Caracterización de la Semejanza Débil

Vamos a ver que que el cómputo del superior en 3.20 (para la comprobación de la ecuación 3.21) puede realizarse tomando un único corte de precisión al difuso, que coincide, precisamente, con el nivel de semejanza impuesto. Vemos primero el caso correspondiente a la semejanza entre un valor crisp y uno difuso, y luego pasaremos al caso general entre dos datos difusos.

Teorema 3.3.1 *Sea D un conjunto compacto, R una relación de semejanza continua en ambas variables definida sobre $D \times D$, y sea un conjunto difuso F con función de pertenencia continua sobre D, verificando $F_\alpha \neq \emptyset$. Entonces, denotando por $H(x)$ a la expresión $\mu_F(x) \wedge R(x,y)$, se tiene que:*

$$y \simeq_\alpha^d F \;\Leftrightarrow\; \max_x H(x) \geq \alpha \;\Leftrightarrow\; \max_{x \in F_\alpha} H(x) \geq \alpha$$

<u>Demostración</u>.

En primer lugar, hay que destacar que F_α es la imagen inversa por una función continua del intervalo cerrado $[\alpha, 1]$, y por lo tanto F_α es cerrado. Al estar incluido en D, también estará acotado, y por tanto deducimos que F_α es un compacto. Como H es el mínimo de dos funciones continuas, también será una función continua, y por lo tanto alcanzará el superior en F_α en algún punto, es decir:

$$\sup_{x \in F_\alpha} H(x) = \max_{x \in F_\alpha} H(x)$$

Análogamente, obtenemos:

$$\sup_x H(x) = \sup_{x \in D} H(x) = \max_x H(x)$$

Como la definición de semejanza débil nos dice que

$$y \simeq_\alpha^d F \;\Leftrightarrow\; \sup_x H(x) \geq \alpha$$

entonces, aplicando lo anterior obtenemos:

$$y \simeq_\alpha^d F \;\Leftrightarrow\; \max_x H(x) \geq \alpha$$

por lo que el teorema se reduce a demostrar que:

$$\max_x H(x) \geq \alpha \;\Leftrightarrow\; \max_{x \in F_\alpha} H(x) \geq \alpha$$

$\Rightarrow$)

$$\max_x H(x) \geq \alpha \;\Rightarrow\; \exists\, x_0 \in D \text{ tal que } H(x_0) \geq \alpha$$

Ahora bien, $H(x_0) = \mu_F(x_0) \wedge R(x_0, y)$, por lo que si $H(x_0) \geq \alpha$, entonces es:

$$\mu_F(x_0) \geq \alpha \;\Leftrightarrow\; x_0 \in F_\alpha \;,\; R(x_0, y) \geq \alpha$$

$$x_0 \in F_\alpha \;\Rightarrow\; R(x_0, y) \leq \max_{x \in F_\alpha} R(x, y)$$

Encadenando desigualdades obtenemos:

$$\max_{x \in F_\alpha} R(x, y) \geq \alpha$$

tal y como queríamos demostrar.

$\Leftarrow$) Debemos demostrar ahora que:

$$\max_{x \in F_\alpha} H(x) \geq \alpha \;\Rightarrow\; \max_x H(x) \geq \alpha$$

De la hipótesis deducimos que

$$\left.\begin{array}{l} \exists\, x_0 \in F_\alpha \text{ tal que } R(x_0, y) = \max_{x \in F_\alpha} H(x) \geq \alpha \\ x_0 \in F_\alpha \;\Rightarrow\; \mu_F(x_0) \geq \alpha \end{array}\right\} \Rightarrow$$

$$\Rightarrow\; \exists\, x_0 \in F_\alpha \text{ tal que } \mu_F(x_0) \wedge R(x_0, y) \geq \alpha \;\Rightarrow\; \exists\, x_0 \in F_\alpha \text{ tal que } H(x_0) \geq \alpha$$

Ahora bien, como $x_0 \in F_\alpha$ se tiene que:

$$\max_{x \in F_\alpha} H(x) \geq H(x_0)$$

Encadenando desigualdades obtenemos:

$$\max_{x \in F_\alpha} H(x) \geq \alpha$$

Finalmente tenemos:

$$F_\alpha \subseteq D \;\Rightarrow\; \max_{x \in F_\alpha} H(x) \leq \max_x H(x)$$

por lo que encadenando de nuevo desigualdades deducimos

$$\max_x H(x) \geq \alpha$$

tal y como queríamos demostrar.

□

Nota 1. En el caso discreto, siempre se alcanza el supremo en algún punto, por lo que el razonamiento seguido en la anterior demostración sigue siendo válido.

Nota 2. Para no tener que calcular el término

$$\max_{x \in F_\alpha} R(x, y)$$

podemos limitarnos a los extremos del corte, cuando trabajamos con números difusos trapezoidales y con R una distancia. En este caso, la comprobación del teorema 3.3.1 pasaría a ser la siguiente:

$$R(x^-_{F_\alpha}, y) \vee R(x^+_{F_\alpha}, y) \geq \alpha \tag{3.33}$$

dónde:

$$x^-_{F_\alpha} = \min\{x : \; x \in F_\alpha\} \; , \; x^+_{F_\alpha} = \max\{x : \; x \in F_\alpha\}$$

Trabajando con pares de difusos en general, tenemos la siguiente caracterización:

Teorema 3.3.2 (Caracterización de la semejanza débil) *Sean dos difusos F y F' definidos sobre un universo de discurso D, y $\alpha \in [0,1]$ tal que F_α, $F'_\alpha \neq \emptyset$. Entonces, si o bien D es discreto, o bien D es continuo pero se satisfacen las condiciones del teorema 3.3.1, se tiene que:*

$$F \simeq^d_\alpha F' \;\Leftrightarrow\; \max_{y \in F_\alpha, x \in F'_\alpha} R(x, y) \geq \alpha$$

Demostración.

$\Rightarrow$) Aplicando la hipótesis, tenemos:

$$\max_y(\mu_F(y) \wedge R^d(F', y)) \geq \alpha$$

La función $R^d(F', y) = \max_{x \in F'_\alpha} \mu_F(x) \wedge R(x, y)$ es continua respecto y, por lo que podemos aplicar los mismos pasos que se dieron en la demostración del teorema 3.3.1, por lo que deducimos lo siguiente:

$$\max_y(\mu_F(y) \wedge R^d(F', y)) \geq \alpha \;\Leftrightarrow\; \max_{y \in F_\alpha} R^d(F', y) \geq \alpha$$

por lo que:

$$\exists\ y_0 \in F_\alpha \quad \text{tal que } R^d(F', y_0) \geq \alpha$$

Aplicando ahora el teorema 3.3.1, obtenemos:

$$R^d(F', y_0) \geq \alpha \ \Leftrightarrow\ \max_{x \in F'_\alpha} R(x, y_0) \geq \alpha$$

Así pues, hemos demostrado que:

$$\exists\ y_0 \in F_\alpha \quad \text{tal que } \max_{x \in F'_\alpha} R(x, y_0) \geq \alpha$$

por lo que también se verificará:

$$\max_{y \in F_\alpha} \max_{x \in F'_\alpha} R(x, y) \geq \alpha$$

c.q.d

$\Leftarrow$) Se demuestra siguiendo los mismos pasos que en la demostración del teorema 3.3.1.

$\square$

Nota. Para no tener que calcular el término

$$\max_{y \in F_\alpha, x \in F'_\alpha} R(x, y)$$

podemos limitarnos a los extremos del corte, cuando trabajamos con números difusos trapezoidales y con R una distancia. En este caso, la comprobación del teorema 3.3.2 pasaría a ser la siguiente:

$$R(x^-_{F_\alpha}, x^+_{F'_\alpha}) \vee R(x^+_{F_\alpha}, x^-_{F'_\alpha}) \geq \alpha \tag{3.34}$$

3.3.2 Caracterización de la Semejanza Fuerte

Lema 3.3.3 *Sea un conjunto difuso G definido sobre un universo de discurso continuo D. Supongamos que D es un conjunto compacto, y que la función de pertenencia μ_G es continua sobre D, verificando $G_{1-\beta} \neq \emptyset$. Sea R una relación de semejanza continua en* *ambas variables y definida sobre $D \times D$. Entonces, denotando por $H(x)$ a la expresión $(1 - \mu_F(x)) \vee R(x,y)$, se tiene que:*

$$\inf_x H(x) \geq \beta \Leftrightarrow \min_x H(x) \geq \beta \Leftrightarrow \min_{x \in G_{1-\beta}} H(x) \geq \beta$$

Demostración.

En primer lugar, hay que destacar que $G_{1-\beta}$ es la imagen inversa por una función continua del intervalo cerrado $[1-\beta, 1]$, y por lo tanto $G_{1-\beta}$ es cerrado. Al estar incluido en D, también estará acotado, y por tanto deducimos que $G_{1-\beta}$ es un compacto. Como H es el máximo de dos funciones continuas, también será una función continua, y por lo tanto alcanzará el inferior en $G_{1-\beta}$ en algún punto, es decir:

$$\inf_{x \in G_{1-\beta}} H(x) = \min_{x \in G_{1-\beta}} H(x)$$

Análogamente, obtenemos:

$$\inf_x H(x) = \inf_{x \in D} H(x) = \min_x H(x)$$

Así pues, debemos demostrar la siguiente equivalencia:

$$\min_x H(x) \geq \beta \Leftrightarrow \min_{x \in G_{1-\beta}} H(x) \geq \beta$$

Descompongamos ahora la primera expresión en la siguiente forma:

$$\min_x H(x) = \min \left\{ \min_{x \in G_{1-\beta}} H(x) \ , \ \inf_{x \notin G_{1-\beta}} H(x) \right\} \tag{3.35}$$

En el caso de que sea:

$$\min_x H(x) \geq \beta$$

entonces, aplicando la ecuación 3.35, también será

$$\min_{x \in G_{1-\beta}} H(x) \geq \beta$$

por lo que quedaría demostrada la implicación suficiente del lema.

Por lo tanto, debemos demostrar ahora la condición necesaria, es decir:

$$\min_{x \in G_{1-\beta}} H(x) \geq \beta \Rightarrow \min_x H(x) \geq \beta$$

Para ello, vamos a ver que el segundo término de la ecuación 3.35 es siempre mayor o igual que β:

$$\forall\, x \notin G_{1-\beta} \;\Rightarrow\; \mu_G(x) < 1-\beta \;\Rightarrow\; 1-\mu_G(x) > \beta \;\Rightarrow$$

$$(1-\mu_G(x)) \vee R(x,y) > \beta \;\Leftrightarrow$$

$$\Leftrightarrow\; H(x) > \beta \quad \forall\, x \notin G_{1-\beta} \;\Rightarrow\; \inf_{x \notin G_{1-\beta}} H(x) \geq \beta$$

Pero si por hipótesis es $\min_{x \in G_{1-\beta}} H(x) \geq \beta$, entonces deducimos que:

$$\min \left\{ \min_{x \in G_{1-\beta}} H(x) \;,\; \inf_{x \notin G_{1-\beta}} H(x) \right\} \geq \beta$$

lo que implica, aplicando la ecuación 3.35, que

$$\min_x H(x) \geq \beta$$

tal y como queríamos demostrar. □

Teorema 3.3.4 *En las condiciones del lema 3.3.3, se tiene que:*

$$y \simeq^f_\beta G \;\Leftrightarrow\; \min_{x \in G_{1-\beta}} R(x,y) \;\geq \beta$$

Demostración.

Por definición de semejanza débil, y denotando $(1-\mu_G(x)) \vee R(x,y)$ por $H(x)$, tenemos:

$$y \simeq^f_\beta G \;\Leftrightarrow\; \inf_x H(x) \geq \beta$$

Teniendo en cuenta las hipótesis, $H(x)$ es una función continua sobre un compacto, por lo que alcanzará el mínimo, es decir:

$$y \simeq^f_\beta G \;\Leftrightarrow\; \min_x H(x) \geq \beta$$

Si aplicamos ahora el lema 3.3.3, entonces habrá que demostrar lo siguiente:

$$\min_x H(x) \geq \beta \;\Leftrightarrow\; \min_{x \in G_{1-\beta}} R(x,y) \;\geq \beta$$

$\Leftarrow$) Aplicando la hipótesis tenemos:

$$\min_{x\in G_{1-\beta}} R(x,y) \geq \beta \Rightarrow R(x,y) \geq \beta \quad \forall\, x \in G_{1-\beta} \Rightarrow$$
$$\Rightarrow (1-\mu_G(x)) \vee R(x,y) \geq \beta \quad \forall\, x \in G_{1-\beta} \Rightarrow \min_{x\in G_{1-\beta}} H(x) \geq \beta$$

por lo que quedaría demostrada la implicación necesaria

$\Rightarrow$) Vamos a demostrarlo por reducción al absurdo. Así pues, supongamos que tenemos simultáneamente las siguientes condiciones:

$$\min_{x\in G_{1-\beta}} R(x,y) < \beta\,, \quad \min_{x\in G_{1-\beta}} H(x) \geq \beta$$

Sea $\beta' = \min_{x\in G_{1-\beta}} R(x,y)$. Entonces tenemos:

$$\forall\, x \in G_{1-\beta} \quad R(x,y) \leq \beta' \tag{3.36}$$

Por otra parte, al ser $\beta' < \beta \Rightarrow 1-\beta' > 1-\beta$, y como μ_G es una función continua verificará:

$$\exists\, x_0 \in D \ \text{ tal que } 1-\beta' > \mu_G(x_0) > 1-\beta \tag{3.37}$$

Si unimos lo anterior a 3.36, tenemos:

$$\left.\begin{array}{l} x_0 \in G_{1-\beta} \Rightarrow R(x_0,y) \leq \beta' < \beta \\ 1-\mu_G(x_0) < \beta \end{array}\right\} \Rightarrow$$
$$\Rightarrow \quad \exists\, x_0 \in G_{1-\beta} \ \text{ tal que } (1-\mu_G(x_0)) \vee R(x_0,y) < \beta$$

Pero por hipótesis se tiene que:

$$\min_{x\in G_{1-\beta}} H(x) \geq \beta \Rightarrow \quad \forall\, x \in G_{1-\beta} \ (1-\mu_G(x)) \vee R(x,y) \geq \beta$$

lo que lleva a una contradicción.

$\square$

Nota. Para no tener que calcular el término

$$\min_{x\in G_{1-\beta}} R(x,y))$$

podemos limitarnos a los extremos del corte, cuando trabajamos con números difusos trapezoidales y con R una distancia. En este caso, la comprobación del teorema 3.3.4 pasaría a ser:

$$R(x^-_{G_{1-\beta}}, y) \wedge R(x^+_{G_{1-\beta}}, y) \geq \beta \tag{3.38}$$

Observemos que en el caso discreto, no se cumple el anterior teorema ya que puede no haber valores en el universo con grado de pertenencia verificando 3.37. Por ejemplo, considerar el difuso $G = \{a/0.2 + b/1\}$ con $R(y, b) = 0.9$, $R(y, 9) = 0$ y $\beta = 0.8$. Entonces

$$R^d(y, F) = (0.8 \vee 0) \wedge (0 \vee 0.9) = 0.8$$

Así pues $R^d(y, F) \geq \beta$ pero sin embargo $\min_{x \in G_{0.2}} R(x, y) = 0$. En este ejemplo, podemos comprobar que el problema del caso discreto es que los valores x con grado de pertenencia igual a $1 - \beta$ verifican que $(1 - \mu_G(x)) \vee R(x, y) = \beta \quad \forall\ R(x, y)$, por lo que no *aportan información* al mínimo. Así pues, debemos quedarnos con aquellos valores con grado de pertenencia estrictamente mayor que $1 - \beta$. Esto lo demostramos en el siguiente teorema:

Teorema 3.3.5 *Sea un conjunto difuso G definido sobre un universo de discurso discreto D, verificando $G_{1-\beta} \neq \emptyset$. Sea R una relación de semejanza definida sobre $D \times D$, y $\gamma \in [0, 1]$ tal que:*

$$1 - \gamma = \min\{\mu_G(x) \ \ \textit{tal que}\ \mu_G(x) > 1 - \beta\}$$

Entonces, se tiene lo siguiente:

$$y \simeq^f_\beta G \ \Leftrightarrow \ \min_{x \in G_{1-\gamma}} R(x, y) \ \geq \beta$$

Demostración.

En principio, hemos de observar que

$$G_{1-\beta} \neq \emptyset \ \Rightarrow \quad \exists\ \gamma \ \text{ tal que } 1 - \gamma = \{\mu_G(x) \ \text{ tal que } \mu_G(x) > 1 - \beta\}$$

Denotemos por $H(x)$ a la expresión $(1 - \mu_G(x)) \vee R(x, y)$. Aplicando la definición de semejanza fuerte, tenemos:

$$y \simeq^f_\beta G \Leftrightarrow \min_x H(x) \geq \beta$$

Consideremos los siguientes conjuntos:

$$G^{<}_{1-\beta} = \{x \in D \text{ tal que } \mu_G(x) < 1-\beta\}$$

$$G^{=}_{1-\beta} = \{x \in D \text{ tal que } \mu_G(x) = 1-\beta\}$$

$$G^{>}_{1-\beta} = \{x \in D \text{ tal que } \mu_G(x) > 1-\beta\}$$

Como $G^{<}_{1-\beta} \cup G^{=}_{1-\beta} \cup G^{>}_{1-\beta} = G$, entonces podemos expresar $\min_x H(x)$ en la forma siguiente:

$$\min_x H(x) = \min\left\{\min_{x\in G^{<}_{1-\beta}} H(x),\ \min_{x\in G^{=}_{1-\beta}} H(x),\ \min_{x\in G^{>}_{1-\beta}} H(x)\right\} \tag{3.39}$$

Por lo tanto $\min_x H(x) \geq \beta$ si y solo si cada uno de los anteriores términos es mayor o igual que β. Por hipótesis es $G^{>}_{1-\beta} = G_{1-\gamma} \neq \emptyset$. Si los otros conjuntos fuesen vacíos, tendríamos demostrada la tesis. Así que suponemos el caso general $G^{=}_{1-\beta} \neq \emptyset$ y $G^{>}_{1-\beta} \neq \emptyset$. Vamos a ver que, en este caso, siempre es el mínimo de $H(x)$ mayor o igual que β.

– Sea $x \in G^{<}_{1-\beta} \Rightarrow \mu_G(x) < 1-\beta \Rightarrow 1-\mu_G(x) > 1-\beta \Rightarrow$

$$\Rightarrow (1-\mu_G(x)) \vee R(x,y) > \beta \Rightarrow min_{x\in G^{<}_{1-\beta}} H(x) > \beta$$

– Sea $x \in G^{=}_{1-\beta} \Rightarrow \mu_G(x) < 1-\beta \Rightarrow 1-\mu_G(x) = 1-\beta \Rightarrow$

$$\Rightarrow (1-\mu_G(x)) \vee R(x,y) \geq \beta \Rightarrow min_{x\in G^{<}_{1-\beta}} H(x) \geq \beta$$

Así pues, aplicando 3.39, tenemos:

$$\min_x H(x) \geq \beta \Leftrightarrow \min_{x\in x\in G^{>}_{1-\beta}} H(x) \geq \beta \Leftrightarrow \min_{x\in x\in G_{1-\gamma}} H(x) \geq \beta$$

□

Nota. Obsérvese que si imponemos la restricción de normalización de los datos difusos, entonces se satisfacen las condiciones del teorema anterior.

Trabajando con pares de difusos en general, tenemos la siguiente caracterización:

Teorema 3.3.6 (Caracterización de la semejanza fuerte)

- Caso Discreto. *Sean dos difusos G y G' en las condiciones del teorema 3.3.5, y γ_1, $\gamma_2 \in [0,1]$ tales que:*

$$1 - \gamma_1 = \min\{\mu_G(x) \ \ \textit{tal que} \ \ \mu_G(x) > 1 - \beta\}$$

$$1 - \gamma_2 = \min\{\mu_{G'}(x) \ \ \textit{tal que} \ \ \mu_{G'}(x) > 1 - \beta\}$$

Sea $\gamma = \max\{\gamma_1, \gamma_2\} \ \Leftrightarrow \ 1 - \gamma = \min\{1 - \gamma_1, 1 - \gamma_2\}$. Entonces, se tiene lo siguiente:

$$G \simeq^f_\beta G' \ \Leftrightarrow \min_{x \in G_{1-\gamma}, y \in G'_{1-\gamma}} R(x,y) \geq \beta$$

- Caso Continuo. *Sean dos difusos G y G' en las condiciones del lema 3.3.3. Entonces, se tiene lo siguiente:*

$$G \simeq^f_\beta G' \ \Leftrightarrow \min_{x \in G_{1-\beta}, y \in G'_{1-\beta}} R(x,y) \geq \beta$$

Demostración.

Consideremos primero el caso continuo y luego veamos el discreto. Empezamos demostrando la condición suficiente.

$\Rightarrow$) Aplicando la hipótesis, obtenemos:

$$\min_y((1 - \mu_G(y)) \vee R^f(G', y)) \geq \beta$$

Como $G_{1-\beta}$ es un compacto y R^f es una función continua en y, podemos aplicar los mismos pasos que se dieron en la demostración del teorema 3.3.4, por lo que obtenemos lo siguiente:

$$\min_y((1 - \mu_G(y)) \vee R^f(G', y)) \geq \beta \ \Leftrightarrow \min_{y \in G_{1-\beta}} R^f(G', y) \geq \beta$$

El anterior mínimo se alcanza en un punto de $G_{1-\beta}$, por lo que:

$$\exists \ y_0 \in G_{1-\beta} \ \text{ tal que } R^f(G', y_0) \geq \beta$$

Aplicando ahora el teorema 3.3.4, obtenemos:

$$R^f(G', y_0) \geq \beta \Leftrightarrow \min_{x \in G'_{1-\beta}} R(x, y_0) \geq \beta$$

Hemos demostrado que:

$$\exists\ y_0 \in G_{1-\beta} \text{ tal que } \min_{x \in G'_{1-\beta}} R(x, y_0) \geq \beta$$

por lo que también se verificará:

$$\min_{y \in G_{1-\beta}} \min_{x \in G'_{1-\beta}} R(x, y) \geq \beta$$

c.q.d

$\Leftarrow$) Se demuestra siguiendo los mismos pasos que la demostración del teorema 3.3.4

Para el caso discreto, basta seguir la misma demostración que la del teorema 3.3.5, pero descomponiendo ahora el mínimo en la siguiente forma:

$$\begin{aligned}&\min_x H(x) = \min\\ &\big\{\min_{x \in G^{<}_{1-\beta}, y \in G'^{<}_{1-\beta}} H(x),\ \min_{x \in G^{<}_{1-\beta}, y \in G'^{=}_{1-\beta}} H(x),\ \min_{x \in G^{<}_{1-\beta}, y \in G'^{>}_{1-\beta}} H(x),\\ &\min_{x \in G^{=}_{1-\beta}, y \in G'^{<}_{1-\beta}} H(x),\ \min_{x \in G^{=}_{1-\beta}, y \in G'^{=}_{1-\beta}} H(x),\ \min_{x \in G^{=}_{1-\beta}, y \in G'^{>}_{1-\beta}} H(x),\\ &\min_{x \in G^{>}_{1-\beta}, y \in G'^{<}_{1-\beta}} H(x),\ \min_{x \in G^{>}_{1-\beta}, y \in G'^{=}_{1-\beta}} H(x),\ \min_{x \in G^{>}_{1-\beta}, y \in G'^{>}_{1-\beta}} H(x)\big\}\end{aligned}$$

y razonando de forma análoga se concluye que el único término que no es siempre mayor o igual que β es el último, por lo que:

$$\min_x H(x) \geq \beta \Leftrightarrow \min_{x \in G^{>}_{1-\beta}, y \in G'^{>}_{1-\beta}} H(x) \geq \beta$$

$$\Leftrightarrow \min_{x \in G_{1-\gamma_1}, y \in G'_{1-\gamma_2}} H(x) \geq \beta \Leftrightarrow \min_{x \in G_{1-\gamma}, y \in G'_{1-\gamma}} \geq \beta$$

dónde $1 - \gamma = \min\{1 - \gamma_1, 1 - \gamma_2\}$, por lo que queda demostrado el teorema. □

Nota. En el caso continuo, para no tener que calcular el término

$$\min_{y \in G_{1-\beta}, x \in G'_{\beta}} R(x, y)$$

podemos limitarnos a los extremos del corte, cuando trabajamos con números difusos trapezoidales y con R una distancia. En este caso, la comprobación del teorema 3.3.6 pasaría a ser:

$$R(x^-_{G_{1-\beta}}, x^+_{G'_{1-\beta}}) \wedge R(x^+_{G_{1-\beta}}, x^-_{G'_{1-\beta}}) \geq \beta \tag{3.40}$$

3.3.3 Equivalencia

Una vez que hemos caracterizado las semejanzas débiles y fuertes a través de los teoremas 3.3.2 y 3.3.6, es inmediato relacionar las definiciones de semejanza débil 3.1.7, 3.2.4 y las de semejanza fuerte 3.1.9, 3.2.14. Vamos a dar las relaciones entre las definiciones de semejanza entre dos difusos en general.

Teorema 3.3.7 *En las condiciones del teorema 3.3.2, F es semejante débil con F' a nivel α según la definición 3.2.4 si y solo si F es semejante débil a nivel de precisión α y nivel de semejanza α con F' según la definición 3.1.7:*

$$F \simeq_{\alpha}^{d\,\alpha} F' \Leftrightarrow F \simeq_{\alpha}^{d} F'$$

Demostración.

Inmediata aplicando la definición 3.1.7, la ecuación 3.10 y el teorema de caracterización 3.3.2 □

Teorema 3.3.8

- Caso Discreto. *Sean dos difusos G y G' en las condiciones del teorema 3.3.5. Entonces, G es semejante fuerte con G' a nivel β según la definición 3.2.14 si y solo si G es semejante fuerte a nivel de precisión $1-\gamma$ y nivel de semejanza β con G' según la definición 3.1.9:*

$$G \simeq_{\beta}^{f\,1-\gamma} G' \Leftrightarrow G \simeq_{\beta}^{f} G'$$

El valor de γ es el dado por el enunciado del teorema 3.3.6.

- Caso Continuo. *Sean dos difusos G y G' en las condiciones del lema 3.3.3. Entonces, G es semejante fuerte con G' a nivel β según la definición 3.2.14 si y solo si G es semejante fuerte a nivel de precisión $1-\beta$ y nivel de semejanza β con G' según la definición 3.1.9:*

$$G \simeq_{\beta}^{f\,1-\beta} G' \Leftrightarrow G \simeq_{\beta}^{f} G'$$

Demostración.

Inmediata aplicando la definición 3.1.9, la ecuación 3.11 y el teorema de caracterización 3.3.6 □

Nota. Tomando como G' el mismo difuso G, se tiene la relación entre la definición 3.1.17 de granularidad por niveles (en este caso sólo habría un único nivel de precisión) y la definición 3.2.16 general de granularidad.

En definitiva, podemos concluir que podemos prescindir de los niveles de precisión a la hora de definir la semejanza débil y fuerte entre dos difusos, siempre que estemos dispuestos a tomar el mismo nivel de precisión que de semejanza en el criterio débil, y tomando uno menos el nivel de semejanza como nivel de precisión en el criterio fuerte. Por lo tanto, a partir de ahora, asumiremos las definiciones 3.2.4 y 3.2.14 para establecer la semejanza entre atributos antecedentes y consecuentes respectivamente. Por otra parte, no olvidemos que ya habíamos visto en el apartado 3.2.4 (página 145) que resultaba más general trabajar con un único nivel de semejanza que con todo un vector de umbrales. Además, en el apartado 3.2.2 (página 133) y en el apartado 3.2.3 (página 137), justificábamos la conveniencia de imponer antecedentes y consecuentes normalizados, en una dependencia funcional difusa. En resumen, a partir de ahora, asumiremos el siguiente marco de trabajo:

- *Comparación de antecedentes: definición 3.2.4 de semejanza débil.*
- *Comparación de consecuentes: definición 3.2.14 de semejanza fuerte.*
- *Restricciones de Integridad: Normalización en antecedentes y consecuentes, y verificando el nivel de granularidad.*
- *Si quieren aplicarse los teoremas de equivalencia 3.3.7 y 3.3.8, entonces deben imponerse las siguientes restricciones:*
  - *Caso Discreto: La normalización de los datos difusos es suficiente*

– *Caso Continuo: Son suficientes la normalización de los datos, que éstos estén definidos sobre un compacto y que las funciones de pertenencia y de semejanza sean continuas en sus variables.*

Capítulo 4

Dependencias Funcionales Difusas Basadas en Reglas

El objetivo de este capítulo es doble. Por una parte, vamos a definir el concepto de redundancia en una base de datos relacional difusa, así como el operador de proyección para proceder a su eliminación. Por otra parte, vamos a introducir la definición de dependencia funcional difusa que nos servirá para establecer una teoría completa de diseño en el capítulo quinto. El desarrollo de estos puntos se realizará en la siguiente forma:

En primer lugar, abordaremos la extensión difusa del concepto de redundancia y de proyección. Recordemos que en la sección 2.2.3 del capítulo segundo (página 58), vimos cómo se habría de plantear genéricamente la definición de dicho concepto, a través de un criterio arbitrario $\mathcal{R}$. Para proceder a su eliminación se definió en la ecuación 2.7 (página 64) un operador genérico de proyección difusa. Ahora, y siguiendo dicho planteamiento general, vamos a introducir el concepto de redundancia y el operador de proyección difusa que nosotros proponemos aplicar en el marco de una base de datos relacional difusa[1]. Para ello, introduciremos los conceptos de tuplas redundantes *respecto* todo el esquema relacional y el de tuplas redundantes *respecto* a un conjunto de atributos.

A continuación presentaremos una nueva definición de dependencia (*dependencia funcional difusa basada en reglas*), más restrictiva que las dadas en las definiciones 2.5.4 y 3.2.26, pero necesaria para establecer una teoría completa de diseño. Para ello, nos basaremos en el concepto de redundancia introducido en la primera sección de este

[1] El operador de reunión difusa no se abordará hasta el capítulo quinto

capítulo, en el de función difusa introducido en el apartado 3.2.1, y en las medidas de semejanza débil y fuerte introducidas en el capítulo tercero. Finalizaremos el capítulo introduciendo el concepto de *llave difusa*, de vital importancia para establecer una teoría de diseño difusa.

4.1 La Redundancia en la Base de Datos

En esta sección vamos a introducir un nuevo concepto de redundancia en el marco de las bases de datos relacionales difusas. Recordemos del primer capítulo, que en la teoría relacional clásica, el criterio de redundancia utilizado, es el de igualdad crisp entre valores, y tiene especial significación en los siguientes puntos:

1. La existencia de redundancia plantea un problema obvio de almacenamiento, por lo que no se han de permitir tuplas redundantes en la base de datos.

2. También hay un problema de redundancia, derivado de la existencia en una relación, de dependencias funcionales que no corresponden a la llave (recordar ejemplo 1.1.3, página 25).

3. Definición de un atributo llave (recordar ecuación 1.1, página 18).

Este mismo esquema será el que nosotros seguiremos en nuestra extensión difusa. El primer punto nos llevará a definir el concepto de *redundancia* entre dos tuplas. Posteriormente definiremos el operador de proyección difusa, que se aplica a una relación con tuplas redundantes. El segundo punto nos llevará a definir en el apartado 4.1.2, el concepto de *redundancia* entre dos tuplas *respecto de un conjunto de atributos* X y a la definición de proyección difusa respecto X. El tercer punto (definición de *llave difusa*) no lo abordaremos hasta la sección 4.2.2.

Pasemos por tanto al estudio del primer punto, en el que analizamos el concepto de redundancia entre tuplas de una base de datos, y vemos cómo ha de eliminarse a través de un operador de proyección difusa.

4.1.1 Identificación del Concepto de Redundancia Difusa

◇ *Preliminares*

En los apartados 2.2.3 y 2.3.2 realizamos una revisión de las principales aproximaciones difusas al importante problema de la eliminación de la redundancia, y analizamos las desventajas que surgían en su tratamiento. Recordemos que el modelo de unificación [15, 72, 73] conlleva un tratamiento coherente y robusto, pero en la definición de redundancia impone restricciones de tipo crisp, como es la existencia de particiones en cada dominio. Por otra parte, en las aproximaciones existentes en el modelo posibilístico, el tratamiento de la redundancia se restringía al planteamiento de consultas a la base de datos.

Por lo tanto, surge la necesidad de introducir un criterio distinto de redundancia en una base de datos difusa. Es importante destacar que dicho criterio no puede venir dado por una relación de semejanza o similitud $\simeq$, que en definitiva, es una generalización de una distancia. Este tipo de medidas se utilizan para comprobar hasta qué punto están cercanos dos valores difusos. Por ejemplo, nosotros la hemos utilizado en la definición 2.5.4 de las (α, β)-d.f.d. Pero si dos datos están *próximos*, no implica que representen el mismo concepto, es decir, que la información que conllevan sea redundante.

Para definir redundancia entre dos valores difusos vamos a considerar el concepto de información difusa englobada bajo otra información difusa:

Definición 4.1.1 *Dos* **valores difusos** *de la base de datos, F y F' se dirá que son* **redundantes**, *y lo notaremos por $F \; \mathcal{R} \; F'$, si y solo si cualquiera de ellos está incluido en el otro, es decir:*

$$\{F \subseteq F' \Leftrightarrow \pi_F(x) \leq \pi_{F'}(x) \quad \forall\, x\} \bigvee \{F' \subseteq F \Leftrightarrow \pi_F(x) \geq \pi_{F'}(x) \quad \forall\, x\}$$

Nota. En la definición 4.1.1, debe tenerse en cuenta que un valor crisp, x, es considerado como un valor difuso con núcleo y soporte igual a x: por lo tanto, F y x son redundantes si y solo si $\pi_F(x) = 1$.

Definición 4.1.2 *Sea una relación r con esquema REL. Diremos que dos* **tuplas** *t_1 y t_2 son* **redundantes**, *y lo notaremos por $t_1 \; \mathcal{R} \; t_2$, si y solo si todos sus valores son* *redundantes, es decir:*

$$t_1 \ \mathcal{R} \ t_2 \Leftrightarrow A_k(t_1) \ \mathcal{R} \ A_k(t_2) \quad \forall \ A_k \in REL$$

◇ *Eliminación de Redundancia. Proyección Difusa*

Una vez que hemos definido el concepto de redundancia entre dos tuplas de una relación, debemos imponer la siguiente restricción:

Restricción. *No pueden haber tuplas redundantes en una relación de una base de datos relacional difusa.*

Ahora bien, podemos encontrarnos con los siguientes casos:

1. Estamos ante una relación en la que ya existían tuplas redundantes. Entonces, debemos fundirlas en una única tupla a través de un operador $\mathcal{F}$, de forma que eliminemos dicha redundancia. En general, este proceso lo introdujimos en el apartado 2.2.3 (ecuación 2.5), y consistía básicamente en aplicar $\mathcal{F}$ a la proyección clásica de la relación, es decir:
$$\mathcal{F}\left(\Pi_W(r)\right) \tag{4.1}$$
dónde W es un conjunto de atributos. Ahora nos ocupa el caso $W = REL(r)$, por lo que el proceso general de eliminación de redundancia en la relación r sería de la forma:
$$\mathcal{F}\left(\Pi_{REL(r)}(r)\right) \equiv \mathcal{F}\left(\Pi(r)\right)$$

2. Estamos ante una relación r sin tuplas redundantes y queremos proyectar sobre un conjunto de atributos W. Entonces, debemos aplicar la ecuación 4.1, es decir, proyectar en la forma clásica sobre W y a continuación eliminar la redundancia.

¿Qué operador $\mathcal{F}$ hemos de utilizar? El resultado obtenido al fusionar con $\mathcal{F}$ dos valores difusos, ha de ser tal que represente una generalización de ambas fuentes de información; la intersección difusa no puede utilizarse ya que, de aplicarlo, estaríamos obteniendo más información de la presente realmente en la base de datos. Por lo tanto, proponemos la utilización de la unión difusa.

Axioma 4.1 *El operador de fusión elegido para eliminar cualquier tipo de redundancia en una relación, sin perder la información aportada por los datos difusos, ha de ser la unión difusa.*

Claro está, cuando aplicamos la unión a dos difusos redundantes F y F', como resultado de la fusión se obtiene el valor más difuso de ellos, es decir:

$$\text{Si} \quad F \mathcal{R} F' \Rightarrow \begin{cases} F \subseteq F' \Rightarrow F \vee F' = F' \\ F \supseteq F' \Rightarrow F \vee F' = F \end{cases}$$

Así pues, estamos considerando que todas las particularizaciones de un conjunto difuso, deben ser englobadas en él, eliminando así redundancia.

Estamos ya en condiciones de introducir el operador de proyección difusa. Este vendrá definido por la ecuación 4.1, sustituyendo $\mathcal{F}$ por la unión difusa, es decir, proyectaremos de forma clásica sobre un conjunto de atributos W, y sobre la relación así obtenida, eliminaremos tuplas redundantes fundiéndolas a través de la unión difusa de cada uno de los valores.

Definición 4.1.3 *Sea una relación r con esquema relacional $REL \supseteq W$. Se define la* **proyección difusa de** r **sobre** W*, y lo notaremos por $\widetilde{\Pi}_W\ (r)$, como una relación con esquema W construida como sigue:*

$$\widetilde{\Pi}_W\ (r) \equiv \mathcal{F}\left(\Pi_W(r)\right) = \left\{ \bigvee_{t_i \in t^{\mathcal{R}}} W(t_i) \right\}_{t \in r} \tag{4.2}$$

con

$$t^{\mathcal{R}} = \{t_i \in r \ \ \textit{tal que}\ A_h(t_i)\ \mathcal{R}\ A_h(t) \quad \forall\ A_h \in W\}$$

y dónde se entiende que la unión es componente a componente, es decir:

$$\bigvee_{t_i \in t^{\mathcal{R}}} W(t_i) = \left(\bigvee_{t_i \in t^{\mathcal{R}}} A_h(t_i) \right)_{A_h \in W}$$

Cuando $W = REL$, omitiremos el subíndice en el operador de proyección, es decir:

$$\widetilde{\Pi}\ (r) \equiv \widetilde{\Pi}_{REL}\ (r)$$

Nota 1. Hemos de destacar que la eliminación de redundancia sólo tiene sentido para aquellas tuplas que verifican el nivel de granularidad en todos los atributos, ya que, en caso contrario, podríamos tener una tupla con todos los valores igual a **desconocido** y por tanto se fundiría con el resto de las tuplas de la base de datos.

Nota 2. Posteriormente, en la definición 4.1.7, introduciremos otro operador de proyección más general, que englobará al anterior como caso particular.

Ejemplo 4.1.4 . Supongamos que $a^F \supseteq a$ y $b^F \supseteq b$. Entonces las tuplas siguientes

$$r = \begin{array}{|cc|} \hline A & B \\ \hline a^F & b \\ a & b^F \\ \hline \end{array}$$

son redundantes y deben fundirse en la dada por:

$$\widetilde{\Pi}\,(r) \equiv \widetilde{\Pi}_{AB}\,(r) = \begin{array}{|cc|} \hline A & B \\ \hline a^F & b^F \\ \hline \end{array}$$

■

4.1.2 Redundancia Derivada de las Dependencias

Una vez que hemos definido el concepto de redundancia entre dos tuplas y el de proyección difusa, pasamos a analizar el segundo punto al que aludíamos en la introducción de esta sección, a saber, la redundancia originada por la presencia de dependencias difusas en una relación, y que habrá que eliminar a través de un operador de proyección difusa. En este apartado vamos a introducir un planteamiento general para abordar dicho problema, y posteriormente, en el apartado 4.1.3, daremos la definición formal del operador de proyección difusa que proponemos en nuestro trabajo. Empezamos recordando el caso clásico, de forma que nos servirá como marco de referencia en la extensión difusa.

◇ *Esquema General: Caso Clásico*

En una base de datos relacional crisp, una dependencia funcional clásica $X \rightarrow Y$ no proveniente de la llave primaria, produce un problema de redundancia ya que siempre que

$$X(t_1) = X(t_2) \tag{4.3}$$

se verifica también que $Y(t_1) = Y(t_2)$, por lo que, llamando x a los valores comunes $X(t_1)$, $X(t_2)$, $X(t_3), \ldots$ y análogamente con Y, se tiene que la información:

$$x \longrightarrow y$$

se encuentra en numerosas tuplas de la base de datos. Vimos en el primer capítulo (ecuación 1.14) que podemos interpretar esta restricción suponiendo la existencia de una función univaluada en la forma

$$\begin{aligned} f:\ D_X(r) &\longrightarrow D_Y(r) \\ f(x) &= y \end{aligned} \tag{4.4}$$

dónde $D_X(r)$ representa el conjunto de valores del dominio D_X que aparecen en la relación r, y análogamente para D_Y. Al proyectar sobre XY (en general sobre el conjunto de atributos antecedentes y consecuentes de la dependencia) según el operador clásico (ecuación 1.10), se elimina dicha redundancia, de forma que en la nueva relación sólo aparece una vez el valor x como valor de atributo antecendente; es decir, todos los valores del atributo antecedente que sean iguales, se fusionan en la proyección en un único valor. Por lo tanto se pasa a considerar la misma función anterior pero sobre la relación proyectada:

$$\begin{aligned} f:\ D_X(\Pi_{XY}(r)) &\longrightarrow D_Y(\Pi_{XY}(r)) \\ f(x) &= y \end{aligned} \tag{4.5}$$

En definitiva, podemos concluir que:

> *La redundancia originada por una dependencia funcional $X \rightarrow Y$, es debida a la aparición de valores de antecedentes iguales, y se elimina proyectando sobre XY la relación original*

(4.6)

◇ *Esquema General: Caso Difuso*

Vamos a plantear el problema de la redundancia originada por la presencia de dependencias difusas, de una forma análoga al caso crisp. Para ello, vamos a extender al caso difuso las ecuaciones 4.3 y 4.4. Por una parte, la extensión difusa de 4.4, la realizamos utilizando la definición 3.2.1 de función difusa introducida en el capítulo tercero. Así pues, suponemos la existencia de una función difusa en la forma siguiente:

$$f : \widetilde{\mathcal{P}}\,(D_X)(r) \longrightarrow \widetilde{\mathcal{P}}\,(D_Y)(r)$$

dónde recordamos que $\widetilde{\mathcal{P}}\,(D_X)(r)$ era el conjunto de datos difusos que aparecía en el atributo(s) X de la relación r.

Por otra parte, la extensión difusa de 4.3 debe realizarse teniendo en cuenta que 4.3 debe leerse como:

$$X(t_1)\ \textit{es redundante con}\ X(t_2) \tag{4.7}$$

Formalmente, podemos introducir la siguiente definición:

Definición 4.1.5 *Sea una relación r con esquema relacional REL, y sea un conjunto de atributos $X \subseteq REL$. Diremos que las* **tuplas** *t_1, $t_2 \in r$ son* **redundantes respecto** *X si y solo si:*

$$X_k(t_1)\ \mathcal{R}\ X_k(t_2) \quad \forall\ X_k \in X$$

Nota. Según la definición 4.1.2, dos tuplas son redundantes si y solo si son redundantes **respecto** todo el esquema relacional.

Con esta definición, la ecuación 4.7 se leería como sigue:

Las tuplas t_1 y t_2 son redundantes respecto X

Por lo tanto, la extensión de 4.6 quedaría en la forma siguiente:

> *La redundancia originada por una dependencia funcional difusa es debida a la aparición de valores redundantes en los atributos antecedentes, o lo que es lo mismo, debida a la aparición de tuplas redundantes respecto el conjunto de atributos antecedentes*

Para eliminar dicha redundancia, debemos definir un operador de proyección difusa que podemos denotar por $Proy$, de forma que siga verificándose la existencia de la función difusa en la relación proyectada. En definitiva, la extensión difusa de la ecuación 4.5 quedaría como sigue:

$$f : \widetilde{\mathcal{P}}\,(D_X)(Proy_{XY}(r)) \longrightarrow \widetilde{\mathcal{P}}\,(D_Y)(Proy_{XY}(r)) \tag{4.8}$$

Al final de este proceso de proyección, se obtendrían un conjunto de pares $\widetilde{x}, \widetilde{y}$, con $f(\widetilde{x}) = \widetilde{y}$, constituyendo un compendio de la información presente en la base de datos original.

La pregunta que nos debemos plantear es ¿qué operador de proyección $Proy$ hemos de considerar en el anterior proceso? En la definición 4.1.3 introducíamos un operador de proyección difusa sobre un conjunto de atributos W que utilizaba la definición de redundancia entre valores difusos, por lo que cabría esperar que pudiésemos utilizarlo en la ecuación 4.8 como sustituto de $Proy$. Sin embargo, esto no es posible. La razón es que, al contrario del caso clásico, dos antecedentes redundantes no tienen por qué tener consecuentes redundantes, ya que la imposición de una (α, β)-d.f.d (en la versión de la definición 3.2.26 -página 146- del capítulo tercero) sólo obliga a que los consecuentes sean semejantes fuertes. Por lo tanto, al realizar la proyección $\widetilde{r} = \widetilde{\Pi}_{XY}\ (r)$, no tienen por qué fundirse las tuplas de r con antecedentes redundantes, y por tanto podrían aparecer tuplas redundantes respecto X en $\widetilde{r}$. Esto no es deseable, ya que no estaríamos eliminando la redundancia originada por la dependencia funcional difusa, o lo que es lo mismo, la redundancia originada por la presencia de tuplas redundantes respecto X.

Concluimos que necesitamos introducir otro operador de proyección difusa, de forma que en la proyección sobre XY no aparezcan valores de X redundantes. Esto lo hacemos en el siguiente apartado.

4.1.3 Proyección Difusa

◇ *Introducción*

Vamos a seguir un ejemplo para plantear la definición de proyección difusa que proponemos para eliminar la redundancia originada por la presencia de dependencias difusas. Supongamos pues una d.f.d entre dos atributos simples, que en general denotaremos por

$$A \rightsquigarrow B$$

y consideremos dos tuplas con los siguientes valores:

A	B	C
a	b	c_1
a	b'	c_2

$$a \rightsquigarrow b \quad (f(a) = b)$$
$$a \rightsquigarrow b' \quad (f(a) = b')$$

Podemos concluir que ambas tuplas son *redundantes respecto* A, por lo que, en un proceso de proyección difusa sobre AB, deberíamos fundirlas en una única tupla. Teniendo en cuenta el axioma 4.1 debemos utilizar la unión difusa para fundir los valores redundantes del atributo antecedente. Una vez que se han fundido en la proyección dichos valores, nos planteamos qué información se ha de deducir de los correspondientes valores del atributo consecuente, o equivalentemente, qué operador de fusión $\mathcal{F}$ hemos de escoger para obtener una única conclusión en la forma siguiente:

$$a \rightsquigarrow b\,\mathcal{F}\,b' \quad (f(a) = b\,\mathcal{F}\,b') \tag{4.9}$$

Siguiendo la filosofía del axioma 4.1, proponemos de nuevo la utilización de la unión difusa[1]. Siguiendo el ejemplo anterior, la ecuación 4.9 se reescribiría como sigue:

$$a \rightsquigarrow b \vee b' \quad (f(a) = b \vee b')$$

En general, dos tuplas dadas por

$r =$

A	B	C
a	b	c_1
a^F	b'	c_2

$$a \rightsquigarrow b \quad (f(a) = b)$$
$$a^F \rightsquigarrow b' \quad (f(a^F) = b')$$

[1]Es claro que la intersección difusa no puede tomarse, ya que b y b' no tienen por qué tener intersección común

con $a^F \supseteq a$ se fundirían en un proceso de proyección sobre AB, en una única tupla[2]. Llamando $\widetilde{r}$ a dicha proyección, tendríamos:

$$\widetilde{r} = \begin{array}{|cc|} \hline A & B \\ \hline a^F & b \vee b' \\ \hline \end{array} \qquad a^F \rightsquigarrow b \vee b' \qquad (f(a^F) = b \vee b')$$

Este sería, intuitivamente, el proceso de eliminación de redundancia, y por tanto la extensión difusa del proceso de proyección clásico. Pero tenemos que ver que, tal y como planteábamos en la ecuación 4.8, la función difusa f presente en la relación r, sigue manteniéndose en la proyección $\widetilde{r}$, es decir:

$$Si \;\; f : \widetilde{\mathcal{P}}\,(D_A)(r) \longrightarrow \widetilde{\mathcal{P}}\,(D_B)(r) \;\; es\ una\ función\ difusa$$

$$entonces \;\; f : \widetilde{\mathcal{P}}\,(D_A)(\widetilde{r}) \longrightarrow \widetilde{\mathcal{P}}\,(D_B)(\widetilde{r}) \;\; es\ una\ función\ difusa$$

Para ello, consideremos la tupla $(a^F, b \vee b')$ de $\widetilde{r}$:

$$f(a^F) = b \vee b'$$

Si aplicamos la definición 3.2.1 de función difusa tenemos que

$$f(a^F) = b \vee b' \Leftrightarrow f(\widetilde{a}) = \widetilde{b} \quad \forall\ \widetilde{a} \subseteq a^F \quad \forall\ \widetilde{b} \subseteq b \vee b'$$

Aplicándolo a los valores de las tuplas del ejemplo, tomando como $\widetilde{a}$ cada uno de los valores $a \subseteq a^F$ y $a^F \subseteq a^F$ y como $\widetilde{b}$ los valores $b \subseteq b \vee b'$ y $b' \subseteq b \vee b'$, tenemos efectivamente que:

$$f(a) = b \;\; , \;\; f(a^F) = b'$$

En definitiva, podemos considerar los datos originales (a, b), (a^F, b') presentes en la relación r, como particularizaciones de $(a^F, b \vee b') \in \widetilde{r}$. Dicho de otra forma, el par $(a^F, b \vee b')$ puede considerarse como una *regla* que comprende como casos particulares a (a, b) y (a^F, b').

Un punto importante a tener en cuenta es que el valor $b \vee b'$ ha de verificar el nivel de granularidad para que la relación obtenida esté de acorde con las restricciones impuestas en

[2]Obsérvese que si no hay datos difusos, entonces no se fusiona ningún par de valores de un atributo antecedente, y por lo tanto tampoco de los consecuentes

nuestra base de datos relacional. Una condición suficiente viene dada por la existencia de una (α, β) dependencia funcional difusa según la definición 3.2.26. Para ello, basta aplicar la proposición 3.2.23 (esto se demostrará en el teorema 4.2.11). En cualquier caso no se impondrá la existencia de una (α, β) d.f.d sino otra condición más fuerte, que motivará la definición de dependencias basadas en reglas, que se verá posteriormente. Pasemos ya a formalizar en el siguiente apartado, el proceso de proyección descrito anteriormente.

◇ *Definición de Proyección Difusa*

Ahora, procedemos a definir el operador base para definir la proyección difusa.

Definición 4.1.6 *Dada una relación r con esquema relacional $REL \supseteq X$, se define el operador $\mathcal{F}^X$ para eliminar tuplas redundantes respecto X. $\mathcal{F}^X(r)$ es una relación con esquema relacional REL, construida como sigue: para cada $t \in r$ se considera*

$$t^{\subset} = \{t' \in r \ \ \textit{tal que} \ X^r(t) \subset X^r(t')\}$$

entonces se añade a $\mathcal{F}^X(r)$ la tupla $\tilde{t}$ construida de la forma siguiente:

$$\textit{Si } t^{\subset} \neq \emptyset \ \Rightarrow \ \ \tilde{t} = \emptyset$$

$$\textit{Si } t^{\subset} = \emptyset \ \Rightarrow \ \ \textit{Se considera } t^{\supseteq} = \{t' \in r \ \ \textit{tal que} \ X^r(t) \supseteq X^r(t')\}$$

$$\tilde{t} = \vee_{t' \in t^{\supseteq}} REL^r(t')$$

Nota 1. En la anterior definición, la inclusión entre conjuntos de atributos se entiende que debe ser componente a componente. Obsérvese también que al tomar la unión difusa de los antecedentes, estamos realmente tomando el difuso mayor de entre ellos, tal y como se comentó en el apartado anterior.

Nota 2. La expresión constructiva que hemos dado del operador $\mathcal{F}^X$ puede parecer poco intuitiva, pero simplemente funde (a través de la unión difusa) las tuplas redundantes respecto X.

Nota 3. Por otra parte, como notación, asumiremos que cuando omitamos el superíndice en $\mathcal{F}^X$ supondremos que X es el esquema relacional. Obsérvese entonces que el operador

$\mathcal{F}^X$ quedaría como $\mathcal{F}$, y esta notación ya se utilizó en el operador introducido en la ecuación 4.2, cuando vimos la definición 4.1.3 de proyección difusa. Sin embargo, no hay ningún inconveniente en lo anterior ya que demostraremos en la proposición 4.1.9 que bajo estas condiciones ambos operadores son, efectivamente, los mismos.

Estamos ya en condiciones de introducir la definición del operador de proyección difuso, que será una extensión del operador clásico Π definido en la ecuación 1.10, y del operador difuso visto en la definición 4.1.3.

Definición 4.1.7 *Dada una relación r con esquema relacional $REL \supseteq W \supseteq X$, se define la* **$X$-proyección difusa de r sobre W**, *como una relación con esquema W construida de la forma siguiente:*

$$\overset{\sim X}{\Pi_W}(r) = \mathcal{F}^X(\Pi_W(r)) \tag{4.10}$$

Nota. En lo que sigue, por abuso de notación, y sólo en aquellos casos en los que no exista ninguna confusión, llamaremos simplemente proyección difusa a la X-proyección difusa. Además, para simplificar la notación, cuando omitamos el subíndice W se entiende que $W = REL$, es decir, el esquema relacional de la relación sobre la que se aplica el operador:

$$\overset{\sim X}{\Pi}(r) \equiv \overset{\sim X}{\Pi_{REL}}(r)$$

La elección de X y W no puede ser arbitraria y debe responder a alguno de los siguientes casos:

1. Si no se tiene información sobre la existencia de dependencias funcionales, sólo tiene sentido eliminar tuplas redundantes en el proceso de proyección, por lo que debe tomarse $X = W = REL$. En este caso, tal y como demostraremos en la proposición 4.1.9, obtenemos el operador de proyección difusa básico, introducido en la definición 4.1.3 (página 171).

2. Si existe una dependencia funcional

$$X \leadsto Y$$

entonces la $X-$proyección sobre $W = XY$ eliminará redundancia originada por la presencia de valores redundantes respecto a los atributos antecedentes.

Ejemplo 4.1.8 . Supongamos una relación con tres tuplas:

$r =$

A	B	C
practicamente 1.70	mas o menos 71	c0
1.85	80	c1
alto	pesado	c2
aproximadamente 1.70	71	c3
alto	muy pesado	c4

dónde

$$\mu_{\texttt{alto}}(1.85) = \mu_{\texttt{pesado}}(80) = 1 \;\;,\;\; \mu_{\texttt{mas o menos 71}}(71) = 1$$

$$\texttt{practicamente 1.70} \subseteq \texttt{aproximadamente 1.70} \Leftrightarrow$$

$$\Leftrightarrow \mu_{\texttt{practicamente 1.70}}(x) \leq \mu_{\texttt{aproximadamente 1.70}}(x)$$

y suponemos que todos los valores que aparecen en C son distintos. Entonces:

$$\widetilde{\Pi}\,(r) \equiv \widetilde{\Pi}^{ABC}_{ABC}\,(r) = r$$

$\widetilde{\Pi}_{AB}\,(r) \equiv \widetilde{\Pi}^{AB}_{AB}\,(r) = \mathcal{F}^{AB}\,(\Pi_{AB}(r)) =$

A	B
alto	pesado
alto	muy pesado
aproximadamente 1.70	mas o menos 71

$\widetilde{\Pi}^{A}_{AB}\,(r) = \mathcal{F}^{A}\,(\Pi_{AB}(r)) =$

A	B
alto	pesado $\vee$ muy pesado
aproximadamente 1.70	mas o menos 71

■

◇ *Propiedades de la Proyección Difusa*

Pasemos ahora a ver algunas propiedades sobre la proyección difusa que necesitaremos en capítulos posteriores. En primer lugar vamos a incluir explícitamente lo que ya hemos comentado anteriormente, y es que el operador de X-proyección difusa es una extensión del operador de proyección difusa y del clásico.

Proposición 4.1.9 *Sea una relación r con esquema relacional $REL \supseteq W \supseteq X$. Entonces se verifica lo siguiente:*

1. *En la relación $\overset{\sim X}{\Pi}_W (r)$ no existen tuplas redundantes respecto X.*
2. $\tilde{\Pi}_W (r) = \overset{\sim W}{\Pi}_W (r)$
3. *El operador de proyección clásico es un caso particular del operador de proyección difuso, cuando todos los datos con crisp, es decir:*

$$\Pi_X(r) = \overset{\sim X}{\Pi}_X (r)$$

<u>Demostración</u>.

El primer punto se obtiene por la propia definición del operador $\mathcal{F}^X$ ya que funde aquellas tuplas con valores redundantes en X. El segundo es también inmediato ya que, según la ecuación 4.2 se tiene:

$$\tilde{\Pi}_W (r) = \mathcal{F}(\Pi_W(r))$$

Pero el operador $\mathcal{F}$ definido en la ecuación 4.2 coincide por definición con $\mathcal{F}^W$, y por tanto:

$$\tilde{\Pi}_W (r) = \mathcal{F}^W(\Pi_W(r)) = \overset{\sim W}{\Pi}_W (r)$$

En cuanto al tercer punto, según la definición del operador $\mathcal{F}^X$ y la definición de proyección clásica dada en la ecuación 1.10, se tiene que, en el caso de que todos los valores de la base de datos sean crisp:

$$\mathcal{F}^X (\Pi_X(r)) = \Pi_X(r)$$

por lo que, efectivamente, la definición de proyección difusa contempla como caso particular la definición de proyección clásica. □

Nota. Si aplicamos el segundo punto de esta proposición al caso particular $W = REL(r)$, y siguiendo la notación introducida en la definición 4.1.3, tenemos:

$$\widetilde{\Pi}\ (r) \equiv \widetilde{\Pi}_{REL(r)} = \overset{\sim REL(r)}{\Pi}_{REL(r)}$$

Un resultado importante, que no se cumplía en general, con las aproximaciones existentes en el modelo posibilístico, es el siguiente:

Teorema 4.1.10 *El proceso de eliminación de redundancia es cerrado, es decir, dada una relación r con esquema relacional REL*

$$\overset{\sim X}{\Pi}_W \left(\overset{\sim X}{\Pi}_W\ (r)\right) = \overset{\sim X}{\Pi}_W\ (r) \quad \forall\ X, W \subseteq REL$$

Demostración.

Supongamos r con esquema $REL \supseteq W \supseteq X$ y que se realiza la proyección

$$\overset{\sim X}{\Pi}_W\ (r)$$

Aplicando la definición 4.1.7, dos tuplas cualesquiera t y t' se funden cuando tienen todos los valores de X redundantes:

$$\{A_h(t) \subseteq A_h(t')\}\ \vee\ \{A_h(t') \subseteq A_h(t)\} \quad \forall\ A_h \in X$$

Y una vez que se han fusionado t y t', no se verifica la premisa anterior para ningún A_h, por lo que queda demostrada la tesis. □

Vamos a demostrar ahora otros resultados que utilizaremos en el último capítulo, y que son consecuencias inmediatas de la definición 4.1.7 de proyección difusa. Pero antes, necesitamos demostrar otros resultados sobre el operador $\mathcal{F}^X$. La parte *i)* de la siguiente proposición, nos dice que podemos intercambiar los operadores de proyección clásica y de fusión, en la ecuación 4.10 de la definición 4.1.7. Este punto será muy útil para la demostración de otras propiedades de la proyección difusa. El apartado *ii)* nos dice que la fusión de tuplas redundantes respecto X, da el mismo resultado si previamente fundimos las tuplas redundantes respecto cualquier conjunto W verificando $X \subseteq W$. En el caso particular $W = X$, nos dice que el proceso de fusión es cerrado.

Proposición 4.1.11 *Sea una relación r con esquema $REL \supseteq W \supseteq X$. Entonces, se verifica lo siguiente:*

i) $\mathcal{F}^X(\Pi_W(r)) = \Pi_W\left(\mathcal{F}^X(r)\right)$

ii) $\mathcal{F}^X(r) = \mathcal{F}^X\left(\mathcal{F}^W(r)\right)$

Demostración.

Para ver la parte *i)*, veamos cómo se construyen dichas relaciones. Las tuplas de $\mathcal{F}^X(\Pi_W(r))$ se forman fundiendo las tuplas redundantes respecto X de la relación $\Pi_W(r)$. Pero las tuplas de $\Pi_W(r)$ se forman fundiendo las tuplas de r con valores de W iguales. Pero si los valores de W son iguales, entonces, al estar X incluido en W, también seran los valores de X iguales, y por lo tanto redundantes. En definitiva, las anteriores tuplas de r son redundantes respecto X. Por lo tanto, podemos decir que las tuplas de $\mathcal{F}^X(\Pi_W(r))$ se forman fundiendo las tuplas de r redundantes respecto X, y quedándonos con la parte correspondiente a W, es decir, restringiéndonos a W:

$$\mathcal{F}^X(\Pi_W(r)) = \mathcal{F}^X(r)\Big|_W = \Pi_W\left(\mathcal{F}^X(r)\right)$$

La parte *ii)* se demuestra siguiendo un razonamiento análogo al anterior. □

En la siguiente proposición hemos incluido una serie de propiedades que se derivan de la definición de proyección difusa, y que serán necesarias en la demostración de los principales teoremas del capítulo quinto. En los dos primeros apartados se analiza la conexión entre la X-proyección difusa sobre W, y la X'-proyección difusa sobre W'. En particular, el punto *i)* establece que la restricción al conjunto de atributos X de una X-proyección difusa sobre W es igual a la proyección difusa sobre X. Los otros dos apartados estudian la conexión que existe entre la relación original y la relación proyectada. En concreto, el punto *iii)* nos dice que todo valor de atributo que aparece en la relación proyectada es igual a algún valor de atributo en la relación original, mientras que en el punto *iv)* se demuestra que toda tupla de r está incluida (componente a componente, y en el sentido de inclusión difusa) en alguna tupla de la proyección.

Proposición 4.1.12 *Sea una relación r con esquema $REL \supseteq W \supseteq X$. Denotemos por $\tilde{r}$ a la relación $\tilde{\Pi}_W^X(r)$. Entonces se verifica lo siguiente:*

i) $\tilde{r}\big|_X = \tilde{\Pi}_X(r)$

ii) Si $X \subseteq X' \subseteq W \subseteq W'$ *entonces* $\tilde{\Pi}_W^X(r) = \tilde{\Pi}_W^X\left(\tilde{\Pi}_{W'}^{X'}(r)\right)$

iii) $\forall\, \tilde{t} \in \tilde{r} \quad \forall\, X_h \ \ \exists\, t_h \in r$ *tal que* $X_h^r(t_h) = X_h^{\tilde{r}}(\tilde{t})$

iv) $\forall\, t \in r \ \ \exists\, \tilde{t} \in \tilde{r}$ *tal que* $W_h^r(t_h) \subseteq W_h^{\tilde{r}}(\tilde{t}) \ \ \forall\, h$

<u>Demostración</u>.

i) Por una parte, tenemos:

$$\tilde{r}\big|_X = \left(\tilde{\Pi}_W^X(r)\right)\Big|_X = \Pi_X\left(\tilde{\Pi}_W^X(r)\right) = \Pi_X\left(\mathcal{F}^X(\Pi_W(r))\right)$$

Ahora, aplicando la parte *i)* de la proposición 4.1.11, deducimos que:

$$\Pi_X\left(\mathcal{F}^X(\Pi_W(r))\right) = \mathcal{F}^X(\Pi_X(\Pi_W(r)))$$

Por definición de proyección clásica, al estar X incluido en W, se tiene que $\Pi_X(\Pi_W(r)) = \Pi_X(r)$, por lo que:

$$\mathcal{F}^X(\Pi_X(\Pi_W(r))) = \mathcal{F}^X(\Pi_X(r))$$

Por otra parte, por definición de proyección difusa, tenemos:

$$\mathcal{F}^X(\Pi_X(r)) = \tilde{\Pi}_X(r)$$

por lo que encadenando igualdades, obtenemos la tesis pedida.

ii) Aplicando la definición de proyección difusa, tenemos:

$$\widetilde{\Pi}_W^{X}\left(\widetilde{\Pi}_{W'}^{X'}(r)\right) = \mathcal{F}^X\left(\Pi_W\left(\widetilde{\Pi}_{W'}^{X'}(r)\right)\right)$$

Aplicando la parte *i)* de la proposición 4.1.11, deducimos:

$$\mathcal{F}^X\left(\Pi_W\left(\widetilde{\Pi}_{W'}^{X'}(r)\right)\right) = \Pi_W\left(\mathcal{F}^X\left(\widetilde{\Pi}_{W'}^{X'}(r)\right)\right)$$

Aplicando de nuevo la definición de proyección difusa, obtenemos que $\widetilde{\Pi}_{W'}^{X'}(r) = \mathcal{F}^{X'}(\Pi_{W'}(r))$, por lo que:

$$\Pi_W\left(\mathcal{F}^X\left(\widetilde{\Pi}_{W'}^{X'}(r)\right)\right) = \Pi_W\left(\mathcal{F}^X\left(\mathcal{F}^{X'}(\Pi_{W'}(r))\right)\right)$$

Aplicando ahora la parte *ii)* de la proposición 4.1.11 concluimos que:

$$\Pi_W\left(\mathcal{F}^X\left(\mathcal{F}^{X'}(\Pi_{W'}(r))\right)\right) = \Pi_W\left(\mathcal{F}^X(\Pi_{W'}(r))\right)$$

Aplicando de nuevo la parte *i)* de la proposición 4.1.11, tenemos:

$$\Pi_W\left(\mathcal{F}^X(\Pi_{W'}(r))\right) = \Pi_W\left(\Pi_{W'}\left(\mathcal{F}^X(r)\right)\right)$$

Pero por defnción de proyección clásica, al estar W incluido en W', se tiene que $\Pi_W(\Pi_{W'}(s)) = \Pi_W(s)$. Tomando como s la relación $\mathcal{F}^X(r)$, deducimos que:

$$\Pi_W\left(\Pi_{W'}\left(\mathcal{F}^X(r)\right)\right) = \Pi_W\left(\mathcal{F}^X(r)\right)$$

Aplicando de nuevo la parte *i)* de la proposición 4.1.11 deducimos que:

$$\Pi_W\left(\mathcal{F}^X(r)\right) = \mathcal{F}^X(\Pi_W(r))$$

y el anterior término es igual a $\widetilde{\Pi}_W^{X}(r)$ por la propia definición de proyección difusa. Encadenando las igualdades, queda demostrada la tesis.

Los apartados *iii), iv)* son inmediatos y se siguen de la misma definición del operador $\mathcal{F}^X$ utilizado en la definición de proyección difusa. □

◇ *Proyección Difusa y (α, β)-d.f.d*

Pasemos ahora a ver la relación entre la definición de proyección difusa y la existencia de (α, β)-dependencias funcionales difusas. Hemos visto anteriormente (página 178) que cuando una relación r satisface una dependencia en la forma $X \rightsquigarrow Y$, hemos de utilizar el operador de proyección difusa $\overset{\sim X}{\Pi}_{XY}(r)$ para aislar dicha dependencia en una relación aparte, de forma que no existan valores redundantes respecto X. Sin embargo, el siguiente resultado nos dice que, al contrario de lo que ocurre en la teoría relacional clásica, el operador $\tilde{\Pi}$ no conserva las (α, β) dependencias funcionales. Este mal comportamiento justifica la necesidad de introducir otra definición de d.f.d más restrictiva.

Proposición 4.1.13 *Sea una relación r que satisface una (α, β)-d.f.d $I = X \xrightarrow{(\alpha,\beta)} Y$, según la definición 3.2.26. Entonces, $\overset{\sim X}{\Pi}_{XY}(r)$ no tiene por qué satisfacer I.*

<u>Demostración</u>.

Veamos un contraejemplo. Considerar una relación r con las siguientes tuplas:

$r =$

A	B	C
1.75	60	c1
alto	70	c2
1.9	pesado	c3

dónde se supone que:

$$\mu_{\texttt{alto}}(1.9) = \mu_{\texttt{pesado}}(70) = 1$$

$$R(1.75, 1.9) < \alpha \quad 1.9 \simeq^d_\alpha \texttt{alto} \quad R(60, 70) \geq \beta \quad 60 \not\simeq^f_\beta \texttt{pesado}$$

Entonces, aplicando la definición de (α, β) d.f.d se tiene que:

$$A \xrightarrow{(\alpha,\beta)} B$$

Sin embargo:

$$\overset{\sim A}{\Pi}_{AB}(r) = \begin{array}{|cc|} \hline A & B \\ \hline \texttt{alto} & \texttt{pesado} \\ \texttt{1.75} & \texttt{60} \\ \hline \end{array}$$

y es inmediato comprobar que $\overset{\sim A}{\Pi}_{AB}(r)$ no satisface la d.f.d □

Nota. Cambiando los criterios débil y fuerte por un criterio cualquiera de semejanza $\simeq$, también se tiene que las (α, β)-d.f.d según la definición 2.5.4, presentan el mal comportamiento descrito en la proposición anterior.

Este mal comportamiento es debido a que, como ya argumentamos al principio de este capítulo, el proceso de proyección debe obtener, a partir de los datos existentes, una serie de *reglas* como resumen de la dependencia funcional. En el anterior ejemplo, la tupla `(alto,pesado)` es un resumen (o compendio) de la información `(alto,70)` y `(1.9,pesado)`. Ante una nueva entrada `(1.75,60)` debemos comparar con la *regla* deducida hasta el momento, y no aisladamente, dos a dos, con los valores presentes en la base. Es por ello, que proponemos una definición de dependencia funcional (más restrictiva) en base a este criterio, y que pasamos a introducir en la siguiente sección.

4.2 Dependencia Funcional Basada en Reglas

En la sección anterior hemos visto que para comprobar la existencia en una relación r, de una dependencia funcional difusa $X \xrightarrow{(\alpha,\beta)} Y$, no deberíamos comparar dos a dos los valores de X presentes en r. Esto era debido a los inconvenientes planteados en la proposición 4.1.13. Por lo tanto, si queremos evitar dichos problemas, debemos introducir otra definición de dependencia funcional difusa. Este es el objetivo de esta sección.

Empezaremos planteando con un ejemplo, cómo habría de ser el proceso de definición de dicha dependencia. Para ello, retomaremos el concepto de función difusa introducido en el apartado 3.2.1. Una vez descrito intuitivamente este proceso, pasaremos a definir formalmente todos los conceptos introducidos, de forma que estaremos en condiciones de introducir las que llamaremos *dependencias funcionales difusas basadas en reglas.* Veremos algunas propiedades importantes y finalmente introduciremos un operador ($RULE$) de características semejantes a la proyección difusa.

4.2.1 Planteamiento del Problema

Recordemos que en el capítulo tercero, se utilizó el concepto de función difusa para comparar débilmente los atributos antecedentes, y fuertemente para los consecuentes. Sin embargo, en la proposición 4.1.13 hemos visto que el operador de proyección difusa $\overset{\sim X}{\Pi}_{XY}(r)$ no mantiene la (α, β)-d.f.d. Esto se debe a que no basta comparar aquellas tuplas con antecedentes semejantes débiles, sino que va a ser necesario tener en cuenta, hasta qué punto están *separadas* las etiquetas o valores difusos que aparezcan como antecedentes en una relación[1]. Específicamente, habrá que analizar especialmente aquellos valores cuyos núcleos se solapen. Esquemáticamente, vamos a seguir los siguientes pasos (ver también figura 4.3 en la página 195):

[1]Se suele utilizar el término *nivel de granularidad* para referirse al grado de separación del conjunto de todas las etiquetas linguísticas sobre un dominio. Este es el concepto que estamos utilizando ahora, pero no lo definimos bajo el mismo epígrafe ya que en esta memoria, estamos utilizando *nivel de granularidad* para *medir* cómo es de difuso un dato.

1. En primer lugar, retomaremos la definición 3.2.1 de función difusa que debe explicar cualquier dependencia funcional difusa $A \rightsquigarrow B$. Un análisis de las restricciones necesarias que conlleva la imposición de tal función, nos llevará a deducir que cualquier comprobación que tengamos que realizar sobre la relación r original, deberá efectuarse sobre otra relación que llamaremos $RULE^A_{AB}(r)$

2. En segundo lugar, aplicaremos la filosofía de las (α, β)-d.f.d, comparando valores semejantes débiles a nivel α en antecedentes y valores semejantes fuertes a nivel β en consecuentes. Esto se realizará sobre la relación $RULE^A_{AB}(r)$

Para ilustrar estos puntos, vamos a considerar una relación cualquiera con esquema $REL \supseteq AB$ dónde hay que comprobar si existe una dependencia $A \rightsquigarrow B$. Supongamos tres tuplas con los siguientes valores en los atributos A y B:

$$t_1 = (a, b) \ , \ t_2 = (a', b') \ , \ t_3 = (a'', b'')$$

dispuestas como muestra la figura 4.1, dónde se supone que $\text{Ker}(a) \cap \text{Ker}(a') \neq \emptyset$, $\text{Ker}(a') \cap \text{Ker}(a'') \neq \emptyset$.

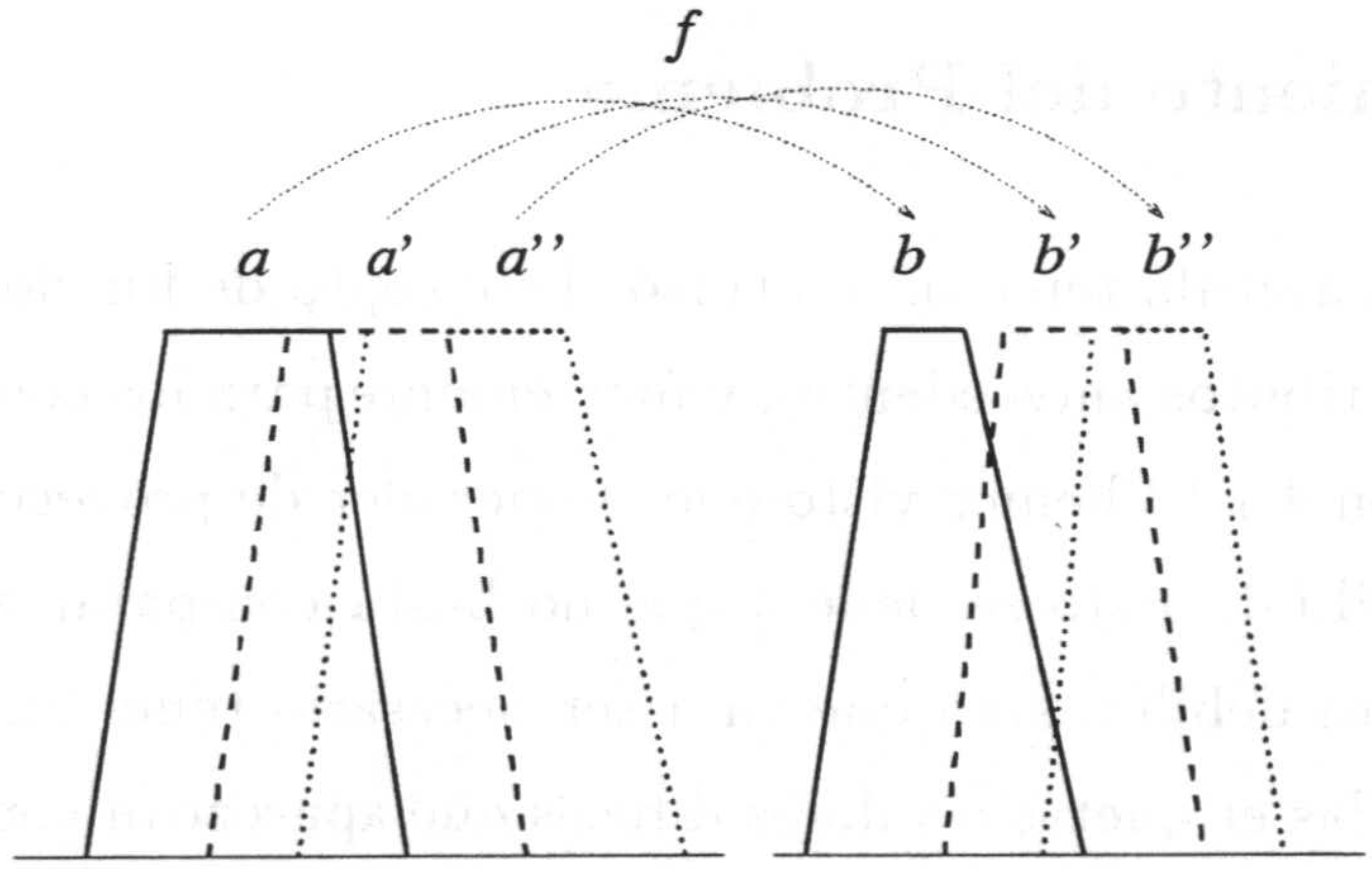

Figura 4.1. Correspondencias entre varios difusos solapados

Argumentábamos en el apartado 3.2.1, que cualquier dependencia funcional difusa $A \rightsquigarrow B$, debería venir modelada por una función difusa según la definición 3.2.1. Retomando esta consideración en nuestro ejemplo, podemos deducir que existe una función f tal que:

$$\begin{aligned} a &\rightsquigarrow b \Leftrightarrow f(a) = b \\ a' &\rightsquigarrow b' \Leftrightarrow f(a') = b' \\ a'' &\rightsquigarrow b'' \Leftrightarrow f(a'') = b'' \end{aligned}$$

Aplicando la definición 3.2.1 de función difusa, obtenemos lo siguiente:

$$f(a) = b \Leftrightarrow f(c) = d \quad \forall\, c \subseteq a \quad \forall\, d \subseteq b$$

$$f(a') = b' \Leftrightarrow f(c') = d' \quad \forall\, c' \subseteq a' \quad \forall\, d' \subseteq b'$$

En particular podemos concluir, tomando como c y c' la intersección de los núcleos de a y a', como d el valor b y como d' el valor b', lo siguiente:

$$Ker(a) \cap Ker(a') \subseteq a \Rightarrow f(Ker(a) \cap Ker(a')) = b$$

$$Ker(a) \cap Ker(a') \subseteq a' \Rightarrow f(Ker(a) \cap Ker(a')) = b'$$

Si queremos obtener un único consecuente, y recordando que el axioma 4.1 nos decía que debía utilizarse la unión difusa para no perder ninguna información (la aportada por b y por b'), debemos concluir lo siguiente:

$$f(\mathrm{Ker}(a) \cap \mathrm{Ker}(a')) = b \vee b'$$

lo que obliga a que sea

$$f(a) = f(a') = b \vee b'$$

y por tanto también:

$$f(a \vee a') = b \vee b'$$

para que f sea coherente con la definición 3.2.1 de función difusa.

En definitiva, de las informaciones $f(a) = b$, $f(a') = b'$, concluimos que ha de ser $f(a \vee a') = b \vee b'$. Si ahora realizamos el mismo razonamiento con las reglas así construidas y con la tercera tupla debemos concluir finalmente lo siguiente:

$$a \vee a' \vee a'' \rightsquigarrow b \vee b' \vee b''$$

tal y como se representa en la figura 4.2

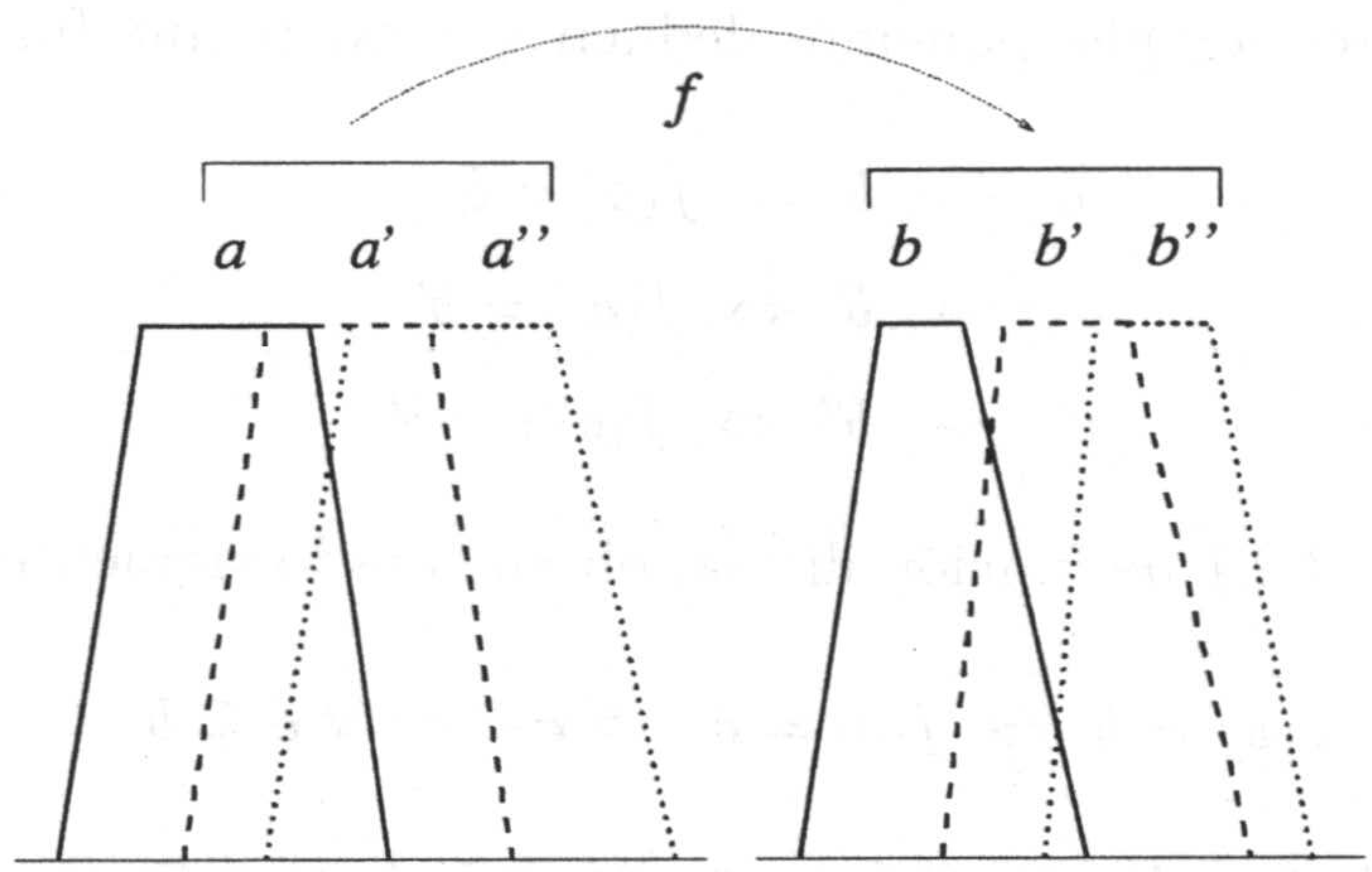

Figura 4.2. Extracción de una única conclusión

Hay que señalar que no se realiza el mismo razonamiento sobre dos difusos cualesquiera a, a' verificando simplemente tener intersección de soportes no vacía, ya que, en este caso, $a \cap a'$ no es un difuso normalizado, y por lo tanto la interpretación $a \cap a'$ ni siquiera puede estar presente en la base de datos (recordar la restricción de normalización en antecedentes y en consecuentes vistas en las páginas 133 y 137).

En definitiva, no sólamente habrá que tener en cuenta que los valores consecuentes han de ser semejantes fuertes para aquellos valores antecedentes semejantes débiles, sino también para aquellos valores antecedentes con núcleos solapados. Por lo tanto, si ahora hubiesen otros valores dados por $(a_0, b_0), (a'_0, b'_0), (a''_0, b''_0)$ con $\mathrm{Ker}(a_0) \cap \mathrm{Ker}(a'_0) \neq \emptyset$, $\mathrm{Ker}(a'_0) \cap \mathrm{Ker}(a''_0) \neq \emptyset$ y con $\mathrm{Ker}(b_0) \cap \mathrm{Ker}(b'_0) \neq \emptyset$, $\mathrm{Ker}(b'_0) \cap \mathrm{Ker}(b''_0) \neq \emptyset$ como los que aparecen en la siguiente tabla:

$r =$

A	B
a	b
a'	b'
a''	b''
a_0	b_0
a'_0	b'_0
a''_0	b''_0

entonces deberíamos concluir las reglas dadas por:

$$a \vee a' \vee a'' \rightsquigarrow b \vee b' \vee b''$$
$$a_0 \vee a'_0 \vee a''_0 \rightsquigarrow b_0 \vee b'_0 \vee b''_0$$

que podrían formar una relación que llamaremos (ver definición 4.2.8) $RULE^A_{AB}(r)$ en la forma siguiente:

$RULE^A_{AB}(r) =$

A	B
$a \vee a' \vee a''$	$b \vee b' \vee b''$
$a_0 \vee a'_0 \vee a''_0$	$b_0 \vee b'_0 \vee b''_0$

Ahora, sobre esta relación, debemos comprobar que a antecedentes semejantes débiles les corresponden consecuentes semejantes fuertes. Si interpretamos las tuplas de $RULE^A_{AB}(r)$ como un conjunto de reglas, entonces la comprobación anterior podría interpretarse como una comprobación de la *consistencia* de dichas reglas (volveremos a este punto en la página 211). Por lo tanto, si suponemos que eran por ejemplo $a' \simeq^d_\alpha a''_0$, entonces, aplicando la proposición 3.2.11 también se verifica que $(a \vee a' \vee a'') \simeq^d_\alpha (a_0 \vee a'_0 \vee a''_0)$, y por tanto debe ser $(b \vee b' \vee b'') \simeq^f_\beta (b_0 \vee b'_0 \vee b''_0)$ para que sean consistentes las reglas extraidas. Ahora bien, esto último equivale, según la proposición 3.2.23 a que cada uno de los valores de B sea semejante fuerte al resto. En definitiva, trabajando sobre la relación de partida r, habría que imponer semejanza fuerte en todos los valores $Y(t_i)$ tales que los $X(t_i)$ estén solapados en sus núcleos.

Pasemos a definir formalmente los conceptos introducidos con el anterior ejemplo.

4.2.2 Definición de Dependencia Funcional Basada en Reglas. Propiedades

◇ *Definición*

Definición 4.2.1 *La* **clase de una tupla** *t respecto un atributo simple X en una relación r, que lo notaremos por $[t]^{X^r}$ es el conjunto:*

$$[t]^{X^r} = \Big\{ t_{i_j} \in r \; (j = 1 \ldots k) \quad \textit{tal que} \; \mathrm{Ker}(X^r(t)) \cap \mathrm{Ker}(X^r(t_{i_1})) \neq \emptyset \; \ldots$$
$$\ldots \mathrm{Ker}(X^r(t_{i_{k-1}})) \cap \mathrm{Ker}(X^r(t_{i_k})) \neq \emptyset \Big\}$$

En el caso de que X sea compuesto, se define:

$$[t]^{X^r} = \bigcap_{X_h^r \in X^r} [t]^{X_h^r}$$

Nota. Obsérvese que $[t]^{X^r}$ forma una clase de equivalencia. En el caso de atributos simples, es la formada por aquellas tuplas que forman parte de una sucesión (de tuplas) con valores en X con intersección de núcleos no vacía. Para el caso de atributos compuestos, se toma la intersección de todas las clases respecto a cada atributo de X. Puede verse un ejemplo de construcción de estas clases de tuplas en la página 203.

Una consecuencia inmediata de la definición es la siguiente proposición, que nos dice que toda clase $[t]$ es no vacía pues al menos $t \in [t]$, y que dos tuplas redundantes respecto X según la definición 4.1.2, pertenecen a la misma clase respecto X.

Proposición 4.2.2

i) $t \in [t]^{X^r} \quad \forall \, t \in r \quad \forall \, X \subseteq REL(r)$

ii) Si $X^r(t_1) \; \mathcal{R} \; X^r(t_2)$ *entonces* $[t_1]^{X^r} = [t_2]^{X^r}$

Demostración.

Inmediata, aplicando la misma definición de clases de tuplas y de tuplas redundantes.
□

Definición 4.2.3 *Dos* **clases de tuplas** $[t_1]^{X^r}$ *y* $[t_2]^{X^r}$ *son* **semejantes** *a grado* α *y lo notaremos por:*

$$[t_1]^{X^r} \simeq^d_\alpha [t_2]^{X^r}$$

si y solo si:

$$\forall\, X^r_h \in X^r \;\; \exists\, t^h_1 \in [t_1]^{X^r} \;\; \exists\, t^h_2 \in [t_2]^{X^r} \;\; \textit{tal que } X^r_h(t^h_1) \simeq^d_{\alpha_h} X^r_h(t^h_2)$$

Nota. En lo que sigue, en aquellos casos que no haya confusión, omitiremos el superíndice de la relación en $[t]^{X^r}$.

Proposición 4.2.4 *Para cualquier par de tuplas t_1 y t_2, se verifica lo siguiente:*

i) $X(t_1) \simeq^d_\alpha X(t_2) \;\Rightarrow\; [t_1]^X \simeq^d_\alpha [t_2]^X$

ii) $[t_1]^X \simeq^d_{\alpha_X} [t_2]^X \;\Rightarrow\; [t_1]^V \simeq^d_{\alpha_V} [t_2]^V \quad \forall\, V \subseteq X$

Demostración.

i) Aplicando la parte *i)* de la proposición 4.2.2, se tiene que la clase de una tupla es siempre distinta de vacío. Aplicando ahora la propia definición de clases de tuplas semejantes, tenemos la tesis pedida.

ii) Por definición de clases de tuplas semejantes, tenemos:

$$[t_1]^X \simeq^d_{\alpha_X} [t_2]^X \;\Leftrightarrow$$

$$\Leftrightarrow \quad \forall\, X_h \in X \;\; \exists\, t^h_1 \in [t_1]^X \;\; \exists\, t^h_2 \in [t_2]^X \;\; \text{tal que } X_h(t^h_1) \simeq^d_{\alpha_h} X_h(t^h_2)$$

Como $V \subseteq X$, se tiene que, en particular, se satisface lo anterior para los atributos de V, es decir:

$$\forall\, X_h \in V \;\; \exists\, t^h_1 \in [t_1]^X \;\; \exists\, t^h_2 \in [t_2]^X \;\; \text{tal que } X_h(t^h_1) \simeq^d_{\alpha_h} X_h(t^h_2)$$

Además, tenemos:

$$V \subseteq X \;\Rightarrow\; \bigcap_{X_h \in X} [t]^{X_h} \subseteq \bigcap_{X_h \in V} [t]^{X_h} \;\Leftrightarrow\; [t]^X \subseteq [t]^V$$

por lo que deducimos finalmente que:

$$\forall\ X_h \in V\ \ \exists\ t_1^h \in [t_1]^V\ \ \exists\ t_2^h \in [t_2]^V\ \ \text{tal que}\ X_h(t_1^h) \simeq^d_{\alpha_h} X_h(t_2^h)$$

que es precisamente la definición de clases de tuplas semejantes respecto V.

□

Definición 4.2.5 ((α,β)-d.f.d.r) *Sean un esquema relacional $REL(A_1 \ldots A_n)$, y sendos subconjuntos de atributos $X = (A_i)_{i \in I} \neq \emptyset$, $Y = (A_j)_{j \in J}$ no necesariamente no vacío, con los umbrales correspondientes $\alpha = (\alpha_i \in [0,1])_{i \in I}$, $\beta = (\beta_j \in [0,1])_{j \in J}$, $I, J \subseteq \{1 \ldots n\}$. Sea una relación r satisfaciendo las restricciones de integridad de normalización y granularidad en antecedentes y consecuentes. En estas condiciones, diremos que r satisface una* **dependencia funcional difusa basada en reglas (d.f.d.r)** *denotado por $X \overset{(\alpha,\beta)}{\Longrightarrow} Y$, si y solo si se satisface alguno de los siguientes puntos:*

- *Por definición, $X \overset{(\alpha_{X'},\gamma)}{\Longrightarrow} \emptyset \quad \forall\ X \subseteq REL \quad \forall\ \gamma \in [0,1]$*

- *Si $X \cap Y = \emptyset$, entonces, cualquier par de tuplas t_1 y t_2, debe satisfacer lo siguiente:*

$$\textit{Si } t_1, t_2 \textit{ verifican } [t_1]^{X^r} \simeq^d_{\alpha} [t_2]^{X^r} \textit{ entonces,}$$
$$\textit{debe ser } Y_k(t_1) \simeq^f_{\beta} Y_k(t_2) \quad \forall\ Y_k \in Y. \tag{4.11}$$

- *En el caso de que $X \cap Y \neq \emptyset$, tomamos $Y' = Y - (X \cap Y) = Y - X$, y definimos:*

$$X \overset{(\alpha,\beta)}{\Longrightarrow} Y \ \Leftrightarrow\ X \overset{(\alpha,\beta_{Y'})}{\Longrightarrow} Y'$$

Nota 1. Obsérvese que la ecuación 4.11 equivale a:

$$Y_k(t_1^i) \simeq^f_{\beta} Y_k(t_2^j) \quad \forall\ Y_k \in Y \quad \forall\ t_1^i \in [t_1]^{Y_k} \quad \forall\ t_2^j \in [t_2]^{Y_k}$$

Nota 2. En la definición de d.f.d.r hemos aislado los casos en los que hay atributos comunes en los conjuntos X e Y que intervienen en la dependencia. Esto es necesario, ya que, de no hacerlo, estaríamos violando axiomas como el de reflexividad o aumento (posteriormente se verá este hecho con más detalle). Esto se debe a que dos valores semejantes débiles no tienen por qué ser semejantes fuertes. Así que estos casos triviales de dependencia los incluimos en la definición. Así pues, la ecuación 4.11 sólo se aplica cuando no hay atributos comunes en antecedentes y consecuentes.

Es muy importante destacar que no se está estableciendo una partición del dominio difuso en base a una relación de semejanza que induzca una relación de equivalencia, tal y como ocurría en las definiciones de Shenoi y Melton, y Buckles y Petry (definiciones 2.3.4 y 2.3.1). Esto es inmediato ya que la definición 3.2.4 de semejanza débil o la definición 3.2.14 de semejanza fuerte no impone ninguna transitividad. Ahora bien, sí se debe imponer una restricción sobre los consecuentes de aquellos valores antecedentes que se superponen en sus núcleos. En cualquier caso, es una restricción adicional a exigir para obtener reglas consistentes (no es un criterio de redundancia), y que obedece a una interpretación semántica, tal y como hemos justificado al inicio de esta sección.

Podemos representar gráficamente el proceso de definición de una (α, β)-d.f.d.r tal y como aparece en la figura 4.3, el cual puede compararse con los esquemas vistos en la página 146 para las (α, β)-d.f.d y las dependencias clásicas.

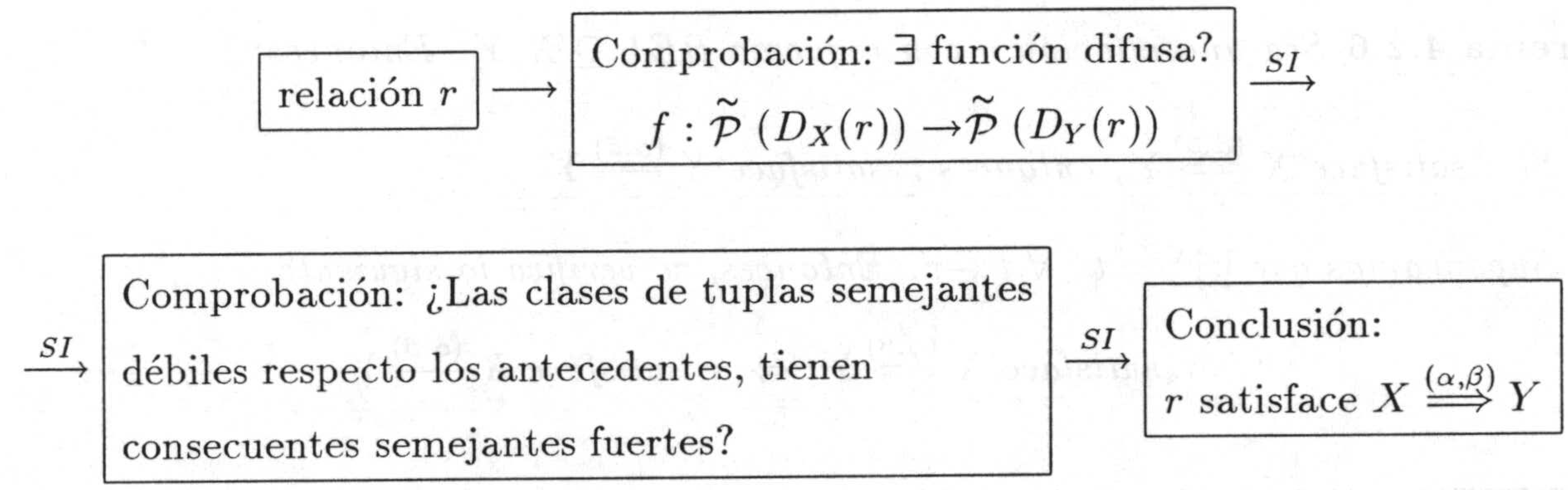

Figura 4.3. Proceso seguido en la definición de una (α, β)-d.f.d.r

Pasemos ahora a analizar algunas propiedades de la definición de dependencia funcional difusa basada en reglas.

◇ *Propiedades de las (α, β)-d.f.d.r*

En primer lugar, es de destacar que la definición de d.f.d.r satisface las propiedades básicas *i), ii), iii) ,iv)* que vimos en la sección 2.4.2. Esto es debido a que el factor clave para no violarlas estaba en comparar consecuentes sólo cuando la compatibilidad entre los antecedentes superase cierto umbral, y en considerar distintos niveles en cada atributo. Obviamente, esta sigue siendo la misma filosofía en la definición de d.f.d.r, por lo que se heredan dichas propiedades. Por contra, los teoremas 2.5.6 y 2.5.7 que analizaban la relación entre las RM-d.f.d, Ch-d.f.d y nuestra definición de (α, β)-d.f.d (ver figura 2.6), ya no se cumplen, puesto que estamos utilizando unas medidas de compatibilidad concretas para comparar los valores de los distintos atributos.

Veamos ahora casos extremos en la definición 4.2.5 de d.f.d.r. Por ejemplo, vamos a ver que si las tuplas están suficientemente separadas, entonces se obtiene la definición 3.2.26 de (α, β)-d.f.d introducida en el capítulo tercero. Además, siempre se tiene que la (α, β)-d.f.d.r implica la (α, β)-d.f.d, o lo que es lo mismo, que el concepto de (α, β)-d.f.d.r. es más restrictivo que el de (α, β)-d.f.d.

Teorema 4.2.6 *Sea una relación r con esquema $REL \supseteq X, Y$. Entonces:*

i) Si r satisface $X \overset{(\alpha,\beta)}{\Longrightarrow} Y$, entonces r satisface $X \overset{(\alpha,\beta)}{\longrightarrow} Y$.

ii) Supongamos que $[t]^X = t \quad \forall\, t \in r$. Entonces, se verifica lo siguiente:

$$r \textit{ satisface } X \overset{(\alpha,\beta)}{\Longrightarrow} Y \;\Leftrightarrow\; r \textit{ satisface } X \overset{(\alpha,\beta)}{\longrightarrow} Y$$

<u>Demostración</u>.

i) Tenemos que demostrar la ecuación 3.32 de la definición 3.2.26 de (α, β)-d.f.d. Así pues, consideremos dos tuplas arbitrarias t_1 y t_2 verificando ser semejantes débiles:

$$X(t_1) \simeq^d_\alpha X(t_2)$$

Aplicando la proposición 4.2.4, tenemos

$$[t_1]^X \simeq^d_\alpha [t_2]^X$$

por lo que aplicando la definición de d.f.d.r a las tuplas t_1 y t_2 deducimos que debe ser:

$$Y(t_1) \simeq^f_\beta Y(t_2)$$

tal y como queríamos demostrar.

ii) La implicación de la derecha es la demostrada en el punto *i)* anterior. Para la otra implicación, basta considerar que, en el caso de que que no haya tuplas con antecedentes X con núcleos solapados, entonces se verifica que:

$$[t]^X = t \quad \forall\, t \in r$$

por lo que la ecuación 4.11 de la definición 4.2.5 pasa a ser:

$$\text{Si } t_1, t_2 \text{ verifican } X(t_1) \simeq^d_\alpha X(t_2) \text{ entonces,}$$

$$\text{debe ser } Y_k(t_1) \simeq^f_\beta Y_k(t_2) \quad \forall\, Y_k \in Y.$$

que es justamente la ecuación 3.32 de la definición 3.2.26 de (α, β)-d.f.d.

□

Nota 1. El caso particular en el que no haya antecedentes solapados, aunque sí redundantes, constituía la definición que dimos en [28]. Obviamente, es un caso particular, pues vimos en la la proposición 4.2.2 que la clase de una tupla contiene al conjunto de tuplas con valores antecedentes redundantes entre sí.

Nota 2. En resumen, podemos concluir que cuanto menos *separados* estén los valores de los antecedentes, más difícil será obtener una dependencia, ya que se incrementa notablemente los valores de los consecuentes que han de ser semejantes. En el caso extremo de que todos los antecedentes estén *solapados*, entonces todas las tuplas de la relación pertenecen a una única clase por lo que habría que imponer semejanza fuerte en todos los consecuentes. De todas formas, preferimos permitir la posibilidad de la existencia de datos solapados en antecedentes, y no restringir las demostraciones de los teoremas fundamentales que veremos en el último capítulo al caso de un conjunto de valores antecedentes suficientemente separados.

Debemos tener cuidado con la interpretación del anterior teorema puesto que lo único que nos está diciendo es que la definición de (α, β)-d.f.d.r. es más restrictiva que la de (α, β)-d.f.d. pero no podemos concluir en general que las propiedades sobre d.f.d se extienden aplicando este teorema. Por ejemplo, para obtener el axioma de transitividad partiríamos de:

$$X \overset{(\alpha,\beta)}{\Longrightarrow} Y \ , \ Y \overset{(\beta,\gamma)}{\Longrightarrow} Z$$

por lo que concluiríamos

$$X \overset{(\alpha,\beta)}{\longrightarrow} Y \ , \ Y \overset{(\beta,\gamma)}{\longrightarrow} Z$$

y aplicando el axioma de transitividad demostrado en la sección 2.5 deduciríamos:

$$X \overset{(\alpha,\gamma)}{\longrightarrow} Z$$

pero esto no implica que sea

$$X \overset{(\alpha,\gamma)}{\Longrightarrow} Z$$

En definitiva, las propiedades descritas para (α, β)-d.f.d. no se heredan para (α, β)-d.f.d.r, por lo que debemos demostrar los resultados que necesitemos sobre estas últimas. Veamos por ejemplo los axiomas de Armstrong. De ellos, los axiomas de reflexividad y aumento se verifican por la propia definición de d.f.d.r, y el de transitividad se obtiene al ser la semejanza fuerte más restrictiva que la débil. En la demostración de estos teoremas omitiremos por comodidad los umbrales de semejanza en la notación de la d.f.d.r. Por lo tanto, la expresión $X \overset{(\alpha,\gamma)}{\Longrightarrow} Z$ la pondremos en la forma $X \Longrightarrow Z$.

Axioma 4.2 (Reflexiva) *Si* $Y \subseteq X$ *entonces* $X \overset{(\alpha,\beta)}{\Longrightarrow} Y$

Demostración.

Si $Y = \emptyset$, entonces se satisface por el punto *i)* de la definición de d.f.d.r. Consideremos entonces que $Y \neq \emptyset$. Si $Y \subseteq X$ entonces $X \cap Y = Y \neq \emptyset$, por lo que aplicando el tercer punto de la definición 4.2.5 de d.f.d.r, tenemos:

$$\{X \Longrightarrow Y\} \Leftrightarrow \{X \Longrightarrow Y - (X \cap Y)\}$$

Como $Y - (X \cap Y) = Y - Y = \emptyset$, entonces, aplicando el primer punto de la definición 4.2.5, se tiene que, efectivamente, $X - (X \cap Y) \Longrightarrow Y - (X \cap Y)$. $\square$

Axioma 4.3 (Transitiva) *Si* $X \stackrel{(\alpha_X,\alpha_Y)}{\Longrightarrow} Y$ *y* $Y \stackrel{(\alpha_Y,\alpha_Z)}{\Longrightarrow} Z$ *entonces* $X \stackrel{(\alpha_X,\alpha_Z)}{\Longrightarrow} Z$

Demostración.

Supongamos que, en general, las parejas de atributos $X\,Y$, y $Y\,Z$, tienen atributos comunes. En caso contrario, la demostración que realizaremos no varía. Así pues, aplicando la definición de d.f.d.r (punto *iii)*) tenemos como hipótesis la siguiente:

$$X \Longrightarrow Y - X \quad , \quad Y \Longrightarrow Z - Y$$

y tenemos que demostrar que $X \Longrightarrow Z - X$. Es inmediato comprobar que $Z - X \subseteq (Z - Y) \cup (Y - X)$, por lo que vamos a considerar los atributos de $Z - X$ que están en cada parte. Así pues, sea:

$$(Z - X) = Z_1 Z_2 \quad \text{tal que } Z_1 \subseteq Z - Y \ , \ Z_2 \subseteq Y - X$$

por lo que tendremos que demostrar que $X \Longrightarrow Z_1 Z_2$, lo cual equivale a demostrarlo componente a componente, es decir:

$$\{X \Longrightarrow Z_1 Z_2\} \ \Leftrightarrow \ \{X \Longrightarrow Z_1 \ \wedge \ X \Longrightarrow Z_2\}$$

En el caso de que alguno de los Z_i fuese vacío, se tendría la dependencia sin más que aplicar el punto primero de la definición de d.f.d.r. Así pues, suponemos que son distintos del vacío. Pasemos a demostrar ambas dependencias:

$X \Longrightarrow Z_2$) Esta parte es inmediata ya que por hipótesis:

$$\{X \Longrightarrow Y - X\} \ \Rightarrow \ \{X \Longrightarrow H \quad \forall H \subseteq (Y - X)\}$$

En particular $Z_2 \subseteq Y - X$, por lo que

$$X \Longrightarrow Z_2$$

$X \Longrightarrow Z_1$) Sean dos clases de tuplas semejantes débiles respecto X.

$$[t_1]^X \simeq^d_\alpha [t_2]^X$$

Sea $X \cap Y = W$. Entonces, podemos poner $Y = (Y - W) \cup W = (Y - X) \cup W$. Utilizando el hecho de que la clase de t_1 respecto X es semejante débil a la clase de t_2 respecto X, vamos a ver que de forma inmediata se obtiene que la clase de t_1 respecto W es semejante débil a la clase de t_2 respecto W, y que, aplicando la hipótesis de d.f.d.r, también se tendrá lo anterior respecto $(Y - W)$. Empecemos aplicando la hipótesis $X \Longrightarrow (Y - X)$ a las tuplas t_1 y t_2:

$$(Y - X)(t_1) \simeq^f (Y - X)(t_2)$$

Aplicando la proposición 3.2.17 obtenemos:

$$(Y - X)(t_1) \simeq^d (Y - X)(t_2)$$

Expresando el anterior término componente componente, tenemos:

$$(Y - X)_h(t_1) \simeq^d (Y - X)_h(t_2) \quad \forall\, (Y - X)_h \in (Y - X)$$

Si aplicamos la parte *i)* de la proposición 4.2.2, tenemos que $t_1 \in [t_1]^W$, $t_2 \in [t_2]^W$, por lo que podemos deducir que:

$$\exists\ t_1 \in [t_1]^W \ \ \exists\ t_2 \in [t_2]^W \ \text{ tal que}$$

$$(Y - X)_h(t_1) \simeq^d (Y - X)_h(t_2) \quad \forall\, (Y - X)_h \in (Y - X) \tag{4.12}$$

Por otra parte, considerando que $W \subseteq X$ y $[t_1]^X \simeq^d [t_2]^X$, entonces podemos aplicar la proposición 4.2.4, que nos dice lo siguiente:

$$[t_1]^W \simeq^d [t_2]^W$$

o equivalentemente, por definición de clases de tuplas semejantes:

$$\forall\ X_h \in W \ \ \exists\ t_h^1 \in [t_1]^W \ \ \exists\ t_h^2 \in [t_2]^W \ \text{ tal que } X_h(t_h^1) \simeq^d X_h(t_h^2)$$

de lo que se deduce, uniéndolo a 4.12, lo siguiente:

$$[t_1]^{(Y-X)\cup W} \simeq^d [t_2]^{(Y-X)\cup W}$$

Aplicando la hipótesis de dependencia, deducimos que debe ser $(Z - Y)(t_1) \simeq^f (Z - Y)(t_2)$. En particular, al ser $Z_1 \subseteq (Z - Y)$, se tiene que $Z_1(t_1) \simeq^f Z_1(t_2)$ $\square$

Axioma 4.4 (Aumento) *Si* $X \overset{(\alpha,\beta)}{\Longrightarrow} Y$ *entonces* $XZ \overset{(\alpha_{XZ},\alpha_{YZ})}{\Longrightarrow} YZ$

Demostración.

Vamos a distinguir los siguientes casos:

- Si $Y = \emptyset$, entonces $YZ = Z$, y por tanto $XZ \supseteq Z$, por lo que $XZ \Longrightarrow Z$ sin más que aplicar el axioma de reflexividad.
- Supongamos que $Y \neq \emptyset$. Aplicando el tercer punto de la definición 4.2.5 de d.f.d.r, tenemos:

$$(XZ \Longrightarrow YZ) \Leftrightarrow (XZ \Longrightarrow YZ - Z - X) \Leftrightarrow (XZ \Longrightarrow Y - X)$$

Para demostrarlo, basta considerar los siguientes puntos: por una parte, aplicando el axioma de reflexividad tenemos que $XZ \Longrightarrow X$. Ahora, aplicando la hipótesis y el punto tercero de la definición de d.f.d.r, obtenemos:

$$(X \Longrightarrow Y) \Leftrightarrow (X \Longrightarrow Y - X)$$

Aplicando ahora el axioma de transitividad, deducimos finalmente que:

$$XZ \Longrightarrow Y - X$$

tal y como queríamos demostrar □

Nota. Obsérvese que el axioma de aumento es una consecuencia inmediata de la definición de d.f.d.r y del axioma de transitividad.

Se pueden demostrar otras reglas de inferencia para así poder derivar nuevas d.f.d.r, como son las correspondientes versiones de los axiomas de descomposición, unión, pseudotransitiva, e.t.c. Sin embargo, para no extendernos demasiado, no vamos a incluir dichas reglas. En cuanto al teorema de completitud de los axiomas de Armstrong, se sigue satisfaciendo en nuestro caso. La demostración, al igual que ocurría con las (α,β)-d.f.d, es una simple extensión del caso clásico. Como siempre, debe imponerse la posibilidad de poder violar la precondición de la definición de la dependencia. En nuestro caso, dicha violación viene dada por la existencia de al menos dos clases de tuplas no semejantes débilmente. Por otra parte, tal y como hemos visto, el axioma de aumento puede deducirse de los axiomas de reflexividad y transitividad, por lo que no se incluye en el conjunto mínimo de axiomas necesarios para poder deducir cualquier d.f.d.r presente en una relación. En resumen, tenemos el siguiente resultado:

Teorema 4.2.7 *Consideremos un esquema relacional difuso, con umbrales de semejanza estrictamente positivos $\alpha_i > 0$, y verificando lo siguiente:*

$$\forall\ A_i \quad \exists\ a_{i1}, a_{i2} \in A_i \quad tal\ que\ [a_{i1}]^{A_i} \neq [a_{i2}]^{A_i}$$

Entonces, los axiomas de reflexividad y transitividad, constituyen un conjunto completo de axiomas de inferencia para la definición de d.f.d.r.

Para finalizar este apartado sobre las propiedades de las d.f.d.r, deberíamos estudiar su comportamiento frente a la proyección difusa. Esto lo veremos en un teorema global más adelante (ver página 206). Por otra parte, el resultado principal que podemos obtener con las d.f.d.r, a saber, el teorema de descomposición difusa sin pérdidas difusas, lo veremos en el capítulo quinto.

4.2.3 El Operador *RULE*

En este apartado vamos a introducir un operador de similares características al de proyección difusa, pero que presenta de una forma más compacta las reglas extraidas de los datos presentes en una relación con una d.f.d.r. Dicho operador lo introdujimos informalmente en el apartado 4.2.1 al inicio de esta sección. Empezamos viendo la definición del operador y luego demostraremos algunas propiedades de interés.

◇ *Definición*

Definición 4.2.8 *Supongamos una relación r con esquema relacional $REL \supseteq X, Y$ y supongamos que $X \overset{(\alpha,\beta)}{\Longrightarrow} Y$. Entonces se define el operador $RULE^X_{XY}(r)$, como una relación con esquema XY, formada por el conjunto de tuplas:*

$$RULE^X_{XY}(r) = \left\{ \bigvee_{t_i \in [t]^{X^r}} XY^r(t_i) \right\}_{t \in r}$$

dónde $[t]^{X^r}$ es la clase de la tupla t respecto X en la relación r tal y como se introdujo en la definición 4.2.1.

En pocas palabras, el operador $RULE$ fusiona a través de la unión difusa, aquellas tuplas que están en la misma clase respecto al conjunto de atributos antecedentes (recordemos que la proyección difusa fundía aquellas tuplas redundantes respecto al conjunto de atributos antecedentes).

Ejemplo 4.2.9 . Supongamos una relación r con esquema $REL \supseteq ACB$, y que $AC \overset{(\alpha,\beta)}{\Longrightarrow} B$:

$$r = \begin{array}{c|ccc|} & A & C & B \\ \hline t_1 & a & c & b \\ t_2 & a' & c'' & b' \\ t_3 & a'' & c' & b'' \\ t_4 & a & c_0 & b_0 \\ t_5 & a_0 & c' & b_0 \\ \hline \end{array}$$

dónde los valores a, a', a'' y b, b', b'' son los representados en la figura 4.1 y suponiendo que a_0 (b_0) no tiene (el núcleo) intersección con ninguno de los valores anteriores (respectivamente de A y de B). Suponemos que

$$\text{Ker}(c) \cap \text{Ker}(c') \neq \emptyset \ , \ \text{Ker}(c') \cap \text{Ker}(c'') \neq \emptyset$$

y que c_0 no tiene (el núcleo) intersección con ninguno de estos valores de C. Entonces, tenemos:

$$[t_1]^A = [t_2]^A = [t_3]^A = [t_4]^A = \{t_1, t_2, t_3, t_4\}$$

$$[t_1]^C = [t_2]^C = [t_3]^C = [t_5]^C = \{t_1, t_2, t_3, t_5\}$$

$$[t_1]^{AC} = [t_2]^{AC} = [t_3]^{AC} = [t_1]^A \cap [t_1]^C = \{t_1, t_2, t_3\}$$

por lo que la relación $RULE^{AC}_{ACB}(r)$ quedaría en la forma:

$RULE^{AC}_{ACB}(r) =$

A	C	B
$a \vee a' \vee a''$	$c \vee c' \vee c''$	$b \vee b' \vee b''$
a	c_0	b_0
a_0	c'	b_0

Obsérvese que, en este caso, $\overset{\sim AC}{\Pi}_{ABC}(r) = r$, ya que en r no había tuplas redundantes respecto AC.

■

◇ *Propiedades del operador RULE*

La siguiente proposición estudia la relación entre el operador *RULE* y la proyección difusa.

Proposición 4.2.10 *Sea una relación r con esquema $REL \supseteq W \supseteq X$. Entonces se verifican las siguientes propiedades:*

i) $RULE^X_W(r) = RULE^X_W\left(\overset{\sim X}{\Pi}_W(r)\right)$

ii) Sean X', W' tales que $X \subseteq X' \subseteq W \subseteq W'$. Entonces $RULE^X_W\left(\overset{\sim X'}{\Pi}_{W'}(r)\right) = RULE^X_W(r)$

iii) Si $\forall\ t$ se verifica que $[t]^{X^r} = \{t\}$, entonces $RULE^X_W(r) = \overset{\sim X}{\Pi}_W(r)$

Demostración.

i) Vamos a seguir un razonamiento análogo al que realizamos en la demostración de la proposición 4.1.11. La relación $RULE_W^X(r)$ se forma fundiendo las tuplas que están en la misma clase respecto X. Si ahora aplicamos la proposición 4.2.2, entonces cualesquiera dos tuplas redundantes respecto X están en la misma clase respecto X, por lo que podemos fundir primero aquellas tuplas de r redundantes respecto X y luego fusionar las tuplas que estén en la misma clase. En definitiva:

$$RULE_W^X(r) = RULE_W^X\left(\overset{\sim X}{\Pi}_W(r)\right)$$

ii) Si aplicamos la parte *i)* de esta proposición, y la parte *ii)* de la proposición 4.1.12, con $X' = X$ y $W' = W$, deducimos lo siguiente:

$$RULE_W^X(r) = RULE_W^X\left(\overset{\sim X}{\Pi}_W(r)\right) = RULE_W^X\left(\overset{\sim X}{\Pi}_W\left(\overset{\sim X'}{\Pi}_{W'}(r)\right)\right)$$

Aplicando ahora la parte *i)* de esta proposición obtenemos:

$$RULE_W^X\left(\overset{\sim X}{\Pi}_W\left(\overset{\sim X'}{\Pi}_{W'}(r)\right)\right) = RULE_W^X\left(\overset{\sim X'}{\Pi}_{W'}(r)\right)$$

Encadenando igualdades, obtenemos la tesis pedida.

iii) Inmediata aplicando la misma definición del operador $RULE$.

□

Veamos ahora un resultado importante, que nos dice que, al igual que el caso clásico, la comprobación de la dependencia puede hacerse directamente en la proyección difusa de la relación, lo cual supone un ahorro computacional puesto que hay menos tuplas en la proyección. Pero si además, trabajamos con la relación obtenida de aplicar el operador $RULE$, entonces sólo debemos comprobar la existencia de una (α, β) d.f.d. lo cual supone aún más ventajas computacionales ya que las comparaciones se hacen sólo con pares de datos y no con las clases de tuplas. Por lo tanto, podemos comprobar la dependencia, ante una nueva entrada, con la relación derivada de aplicar el operador $RULE$, y no con toda la base de datos (al igual que ocurría en el caso clásico).

Teorema 4.2.11 *Sea una relación r, con esquema $REL \supseteq XY$, satisfaciendo las restricciones de integridad. Denotemos por $\tilde{r}$ a la proyección difusa $\overset{\sim X}{\Pi}_{XY}(r)$ y por r^* a la relación $RULE^X_{XY}(r)$. Entonces, las siguientes afirmaciones son equivalentes:*

i) r *satisface* $X \overset{(\alpha,\beta)}{\Longrightarrow} Y$

ii) $\tilde{r}$ *satisface* $X \overset{(\alpha,\beta)}{\Longrightarrow} Y$ *y además* Y *satisface en* $\tilde{r}$, *el nivel de granularidad exigido en* r.

iii) r^* *satisface* $X \overset{(\alpha,\beta)}{\Longrightarrow} Y$ *y además* Y *satisface en* r^*, *el nivel de granularidad exigido en* r.

iv) r^* *satisface* $X \overset{(\alpha,\beta)}{\longrightarrow} Y$ *utilizando semejanza débil en antecedentes y semejanza fuerte en consecuentes, y además* Y *satisface en* r^*, *el nivel de granularidad exigido en* r.

Demostración.

Vamos a demostrarlo en el siguiente orden: $i) \Leftrightarrow iv)$, $iii) \Leftrightarrow iv)$, y finalmente $i) \Leftrightarrow ii)$.

$i) \Rightarrow iv)$ Consideremos dos tuplas semejantes débiles respecto X, y debemos demostrar que los valores de Y son semejantes fuertes:

$$t_1, t_2 \in r^* \text{ tal que } X^{r^*}(t_1) \simeq^d_\alpha X^{r^*}(t_2) \tag{4.13}$$

Los valores $X^{r^*}(t_i)$ se construyen como la unión difusa de las tuplas de r que están en la misma clase respecto X. Consideremos un representantes cualquiera u_1, u_2 para cada una de las clases:

$$XY^{r^*}(t_1) = \bigvee_{u_i^1 \in [u_1]^{X^r}} XY^r(u_i^1) \Rightarrow X_h^{r^*}(t_1) = \bigvee_{u_i^1 \in [u_1]^{X^r}} X_h^r(u_i^1) \quad \forall X_h \in X \tag{4.14}$$

$$XY^{r^*}(t_2) = \bigvee_{u_j^2 \in [u_2]^{X^r}} XY^r(u_j^2) \Rightarrow X_h^{r^*}(t_2) = \bigvee_{u_j^2 \in [u_2]^{X^r}} X_h^r(u_j^2) \quad \forall X_h \in X \tag{4.15}$$

Ahora bien, podemos expresar 4.13 en la siguiente forma:

$$X^{r^*}(t_1) \simeq^d_\alpha X^{r^*}(t_2) \Leftrightarrow X_h^{r^*}(t_1) \simeq^d_\alpha X_h^{r^*}(t_2) \quad \forall X_h \in X$$

Aplicando la proposición 3.2.12, deducimos que:

$$\forall\, X_h \in X \;\; \exists\, u_h^1 \in [u_1]^{X^r} \;\; \exists\, u_h^2 \in [u_2]^{X^r} \;\; \text{tal que } X_h^r(u_h^1) \simeq_{\alpha_h}^d X_h^r(u_h^2)$$

lo que equivale, por definición de clase de tuplas, a que:

$$[u_h^1]^{X^r} \simeq_{\alpha}^d [u_h^2]^{X^r}$$

Aplicando la hipótesis de existencia de d.f.d.r, a estas clases de tuplas, deducimos que debe ser:

$$Y^r(u_i^1) \simeq_{\beta}^f Y^r(u_j^2) \quad \forall\, u_i^1 \in [u_h^1]^{X^r} \quad \forall\, u_j^2 \in [u_h^2]^{X^r}$$

Como resulta que $[u_h^1]^{X^r} = [u_1]^{X^r}$ y $[u_h^2]^{X^r} = [u_2]^{X^r}$, deducimos finalmente que debe ser:

$$Y^r(u_i^1) \simeq_{\beta}^f Y^r(u_j^2) \quad \forall\, u_i^1 \in [u_1]^{X^r} \quad \forall\, u_j^2 \in [u_2]^{X^r}$$

Aplicando ahora la proposición 3.2.22, obtenemos:

$$\bigvee_{u_i^1 \in [u_1]^{X^r}} Y^r(u_i^1) \simeq_{\beta}^f \bigvee_{u_j^2 \in [u_2]^{X^r}} Y^r(u_j^2)$$

pero, teniendo en cuenta las ecuaciones 4.14 y 4.15, lo anterior es equivalente a que:

$$Y^{r^*}(t_1) \simeq_{\beta}^f Y^{r^*}(t_2)$$

tal y como queríamos demostrar. Además, si Y satisface el nivel de granularidad en r, entonces todos los valores $Y^r(u_i^1)$, $Y^r(u_j^2)$ satisfacen el nivel de granularidad. Por lo tanto, aplicando la proposición 3.2.23 concluimos que $Y^{r^*}(t_1)$ y $Y^{r^*}(t_2)$ también satisfacen el nivel de granularidad.

iv) $\Rightarrow$ *i)* Consideremos dos clases de tuplas semejantes débiles:

$$[t_1]^{X^r} \simeq_{\alpha}^d [t_2]^{X^r} \;\Leftrightarrow$$

$$\Leftrightarrow \quad \forall\, X_h \in X \;\; \exists\, t_h^1 \in [t_1]^{X^r} \;\; \exists\, t_h^2 \in [t_2]^{X^r} \;\; \text{tal que } X_h^r(t_h^1) \simeq_{\alpha_h}^d X_h^r(t_h^2)$$

Como todas las tuplas de una clase se fusionan por unión en una única tupla (u_1) de r^*, tenemos que:

$$\exists\,!\, u_1 \in r^* \;\; \text{tal que } XY^r(t_i^1) \subseteq XY^{r^*}(u_1) \quad \forall\, t_i^1 \in [t_1]^{X^r}$$

$$\exists\, !\, u_2 \in r^* \text{ tal que } XY^r(t_j^2) \subseteq XY^{r^*}(u_2) \quad \forall\, t_j^2 \in [t_2]^{X^r}$$

Si aplicamos lo anterior a las tuplas $t_h^1 \in [t_1]^{X^r}$ y $t_h^2 \in [t_2]^{X^r}$ tendremos que:

$$\forall\, X_h \in X \;\; \exists\, t_h^1 \in [t_1]^{X^r} \;\; \exists\, t_h^2 \in [t_2]^{X^r} \text{ tal que}$$

$$X_h^r(t_h^1) \subseteq X_h^r(u_1)\ ,\ X_h^r(t_h^2) \subseteq X_h^r(u_2)$$

Aplicando ahora la proposición 3.2.10 deducimos que:

$$X^{r^*}(u_1) \simeq_\alpha^d X^{r^*}(u_2)$$

Podemos entonces, aplicar la hipótesis de d.f.d.r, por lo que los valores de Y para u_1 y u_2 serán semejantes fuertes, es decir:

$$Y^{r^*}(u_1) \simeq_\beta^f Y^{r^*}(u_2)$$

Ahora bien, si unimos este resultado a los siguientes hechos:

$$Y^{r^*}(u_1) \text{ y } Y^{r^*}(u_2) \text{ satisfacen el nivel de granularidad}$$

$$Y_k^r(t_i^1) \subseteq Y^{r^*}(u_1) \quad \forall\, Y_k \in Y \quad \forall\, t_i^1 \in [t_1]^{X^r}$$

$$Y_k^r(t_j^2) \subseteq Y^{r^*}(u_2) \quad \forall\, Y_k \in Y \quad \forall\, t_j^2 \in [t_2]^{X^r}$$

entonces, estamos en condiciones de aplicar la proposición 3.2.24, por lo que deducimos:

$$Y_k^r(t_i^1) \simeq_{\beta_k}^f Y_k^r(t_j^2) \quad \forall\, t_i^1 \in [t_1]^{X^r} \quad \forall\, t_j^2 \in [t_2]^{X^r} \quad \forall\, Y_k \in Y$$

Pero por la parte *i)* de la proposición 4.2.2, tenemos que $t_1 \in [t_1]^{X^r}$ y $t_2 \in [t_2]^{X^r}$, por lo que podemos aplicar la expresión anterior a estas tuplas y concluimos que:

$$Y_k^r(t_1) \simeq_{\beta_k}^f Y_k^r(t_2) \quad \forall\, Y_k \in Y$$

tal y como queríamos demostrar.

Así pues, hemos demostrado la equivalencia *iv)* $\Leftrightarrow$ *i)*. Veamos ahora el resto:

iii) $\Leftrightarrow$ *iv)* Como en la relación $r^* = RULE^X_{XY}(r)$ no hay tuplas redundantes respecto X, tenemos que:

$$[t]^{Xr^*} = t \quad \forall\, t \in r^*$$

Así pues, estamos en condiciones de aplicar el punto *ii)* del teorema 4.2.6, que nos dice lo siguiente:

$$r^* \text{ satisface } X \overset{(\alpha,\beta)}{\Longrightarrow} Y \Leftrightarrow r^* \text{ satisface } X \overset{(\alpha,\beta)}{\longrightarrow} Y$$

que es la equivalencia buscada.

i) $\Leftrightarrow$ *ii)* Si aplicamos la equivalencia *i)* $\Leftrightarrow$ *iv)* obtenemos:

$$\tilde{r} \text{ satisface } X \overset{(\alpha,\beta)}{\Longrightarrow} Y \Leftrightarrow RULE^X_{XY}(\tilde{r}) \text{ satisface } X \overset{(\alpha,\beta)}{\Longrightarrow} Y$$

Si aplicamos el punto *i)* de la proposición 4.2.10, tenemos que:

$$RULE^X_{XY}(\tilde{r}) = RULE^X_{XY}(r)$$

por lo que tendremos:

$$RULE^X_{XY}(\tilde{r}) \text{ satisface } X \overset{(\alpha,\beta)}{\Longrightarrow} Y \Leftrightarrow RULE^X_{XY}(r) \text{ satisface } X \overset{(\alpha,\beta)}{\Longrightarrow} Y$$

Aplicando de nuevo la equivalencia *i)* $\Leftrightarrow$ *iv)*, concluimos:

$$RULE^X_{XY}(r) \text{ satisface } X \overset{(\alpha,\beta)}{\Longrightarrow} Y \Leftrightarrow r \text{ satisface } X \overset{(\alpha,\beta)}{\Longrightarrow} Y$$

Encadenando equivalencia deducimos finalmente que

$$\tilde{r} \text{ satisface } X \overset{(\alpha,\beta)}{\Longrightarrow} Y \Leftrightarrow r \text{ satisface } X \overset{(\alpha,\beta)}{\Longrightarrow} Y$$

tal y como queríamos demostrar.

□

Utilizando este teorema, podemos representar gráficamente el proceso de comprobación de una (α,β)-d.f.d.r. como aparece en la figura 4.4

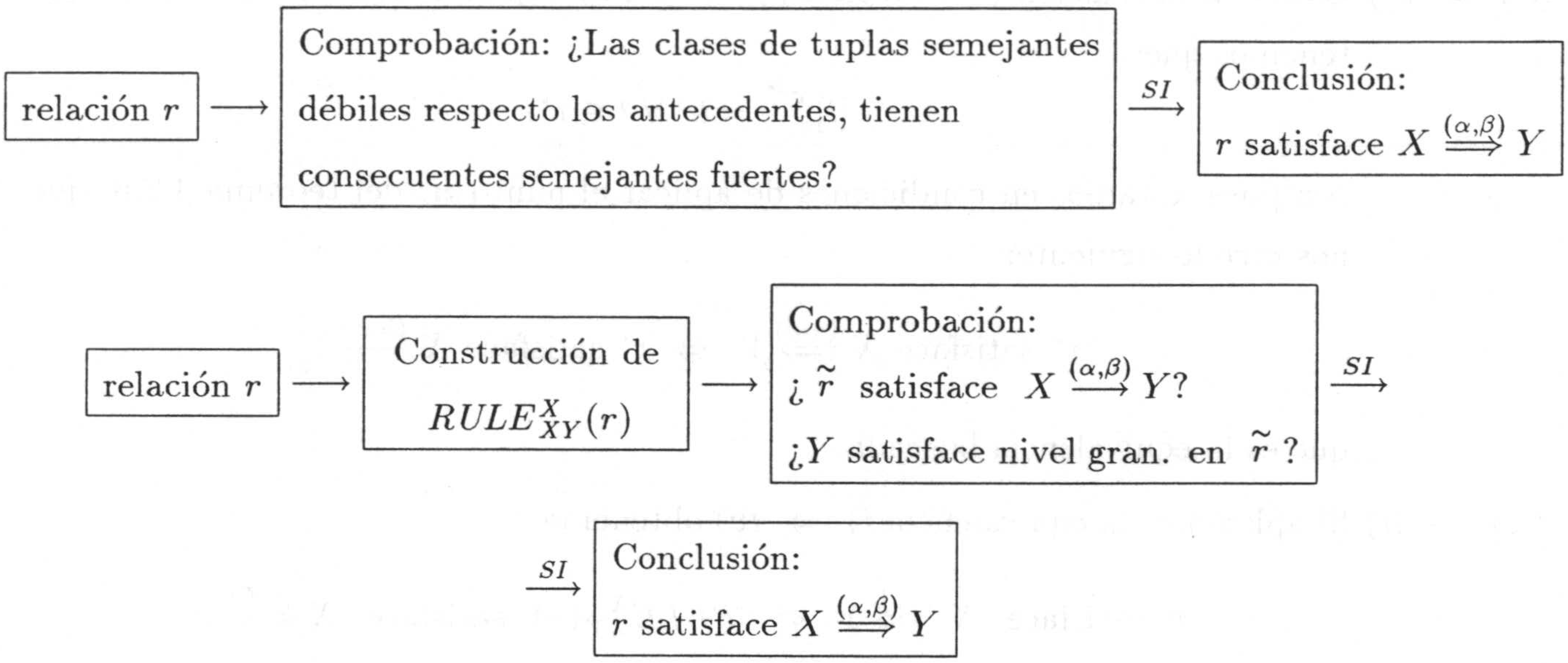

Figura 4.4. Formas de comprobar la existencia de una (α, β)-d.f.d.r $X \overset{(\alpha,\beta)}{\Longrightarrow} Y$

◇ *Elección entre el Operador RULE y la Proyección Difusa*

Hemos introducido dos operadores, a saber, el de proyección difusa y $RULE$. ¿Cuándo habremos de utilizar cada uno? Veremos que es indiferente la elección de uno u otro, para la obtención de las buenas propiedades como descomposiciones sin pérdidas, que se abordarán en el próximo capítulo. En cualquier caso, hay que enfatizar que sólo tiene sentido aplicar el operador $RULE$, para obtener las reglas que explican una d.f.d.r, es decir, dada r con esquema $REL = XYZ$ dónde $X \overset{(\alpha,\beta)}{\Longrightarrow} Y$, entonces la relación

$$RULE^{X}_{XY}(r)$$

explica la dependencia de una forma más compacta que la dada por la relación:

$$\tilde{\Pi}^{X}_{XY}(r)$$

Así pues, ambos son de utilidad para eliminar redundancia proveniente de la existencia de una dependencia difusa, pero el operador de proyección difusa puede aplicarse también sobre relaciones que no presentan dependencias, como por ejemplo podría ser el caso si quisiésemos eliminar redundancia en todo el esquema relacional de una relación.

◇ *Interpretación del Operador RULE*

Las tuplas resultantes de la aplicación del operador $RULE$ pueden interpretarse como un conjunto de reglas en la forma:

$$\textbf{Si } X(t_i) \textbf{ entonces } Y(t_i) \quad \text{con } t_i \in RULE^{X}_{XY}(r)$$

dónde $X(t_i)$ e $Y(t_i)$ son valores difusos, pero la implicación es crisp, y constituyen la extensión difusa de las reglas obtenidas en un proceso de proyección clásico (ecuación 4.4). Este conjunto de reglas no es cualquiera sino que verifican una propiedad de *consistencia*, la dada por la dependencia funcional difusa (teorema 4.2.11). Esquemáticamente el proceso seguido puede verse en la figura 4.5.

Relación r $REL(r) \supseteq X, Y$	$\longrightarrow$	Comprobación: ¿r satisface $X \overset{(\alpha,\beta)}{\Longrightarrow} Y$?	$\overset{SI}{\longrightarrow}$	Almacenar sistema de reglas consistente: $RULE^{X}_{XY}(r)$

Figura 4.5. Construcción de $RULE^{X}_{XY}(r)$

Cabría preguntarse si el diagrama de la figura 4.5 puede realizarse a la inversa; es decir, si un experto puede dar un sistema de reglas $\mathbb{R} = \{X_i \Rightarrow Y_i\}_{i=1\ldots n}$ *consistente* (verificando una d.f.d.r), de forma que se pueda trabajar con dicho conjunto de reglas. Salvo que haya información adicional sobre la estructura de las reglas, la respuesta es que no. Este tema se tratará en el último capítulo, pero intuitivamente podemos justificarlo ahora de la siguiente forma: Cuando se comprueba la existencia de una d.f.d.r, se realizará un proceso de descomposición de la relación original. Este proceso tratará de recuperar los datos presentes en la relación original con la información proveniente de los datos que había en $REL(r)-Y$ y con los datos que había en XY. Si guardamos como segunda información, únicamente un conjunto de reglas ($\mathbb{R}$), entonces no podremos recobrar obviamente los datos originales. Claro está, si tuviésemos alguna información adicional sobre la forma de las reglas, como por ejemplo, que *el crecimiento en los consecuentes es directamente proporcional al crecimiento de los antecedentes*, entonces sí sería posible realizar un proceso de descomposición considerando únicamente el conjunto de datos presentes en $\mathbb{R}$; pero este es un tema que no abordaremos en esta memoria.

En definitiva, si el experto elicita un sistema de reglas $\mathbb{R} = \{X_i \Rightarrow Y_i\}_{i=1\ldots n}$, entonces, la forma de comprobar si la base de datos se adecúa a $\mathbb{R}$, es añadiendo el conjunto $\mathbb{R}$ al conjunto de todas las tuplas, obteniendo así el siguiente:

$$\aleph = \mathbb{R} \cup \{X(t), Y(t)\}_{t \in r} = \{X_i, Y_i\}_{i=1\ldots n} \cup \{X(t), Y(t)\}_{t \in r}$$

Y pasaríamos a comprobar si $\aleph$ es un sistema de reglas *consistente*, es decir, si se verifica una d.f.d.r entre antecedentes y consecuentes.

4.3 Definición de Llave Difusa

Una vez que se ha planteado el concepto de redundancia en una base de datos difusa, hemos de definir el concepto de llave difusa. En la teoría relacional clásica, una llave puede definirse directamente, a través del concepto de dependencia funcional ([78]) tal y como vimos en la definición 1.2.5. Vamos a extender esta definición al caso difuso:

Definición 4.3.1 *Sea una relación r con esquema relacional difuso $REL = (A_i)_{i=1...n}$, con los correspondientes niveles de precisión y semejanza.*
Un conjunto de atributos $K = (A_k)_{k\in\aleph}$ se dice que forman una **llave difusa** *para r si y solo si se verifican las siguientes condiciones:*

i) $\forall\ k\ \ \forall\ t$, *$A_k(t)$ es un difuso normalizado que satisface el nivel de granularidad impuesto en A_k.*

ii) $\forall\ t_1, t_2\ \ \exists\ A_k \in K$ *tal que $A_k(t_1)$ no es redundante con $A_k(t_2)$*

iii) Si Y representa los atributos no llave, entonces se debe satisfacer lo siguiente:

$$K \stackrel{(\alpha,\beta)}{\Longrightarrow} Y$$

Nota 1. Es importante destacar de esta definición que no estamos restringiendo que la llave esté compuesta por datos crisp, lo cual es una imposición en todas las aproximaciones existentes en las bases de datos relacionales difusas. Por contra, debemos imponer que, para valores de la llave *próximos*, deben haber valores también próximos para el resto de los atributos, es decir, que se satisfazga una d.f.d.r. No basta, como en el caso clásico, que no existan tuplas redundantes respecto al conjunto de atributos antecedentes.

Nota 2. Por otra parte, observemos que debido a la restricción que vimos en la página 169, de que no puede haber tuplas redundantes en una relación, se tiene que toda relación verifica al menos la parte *ii)* de la definición de llave difusa. Además, según la definición 4.2.5 de d.f.d.r, se tiene que $K \stackrel{(\alpha,\beta)}{\Longrightarrow} \emptyset\ \ \forall\ K$: en particular $REL(r) \stackrel{(\alpha,\beta)}{\Longrightarrow} \emptyset$. Por lo tanto, una condición necesaria y suficiente para afirmar que toda relación difusa tiene al menos a todo el esquema como llave difusa, es que se cumpla el punto *i)* de la definición de llave difusa para cualquier atributo, es decir:

$$\forall\ A_h \in REL(r), \quad \forall\ t \in r,\ A_h(t) \text{ debe ser un difuso normalizado}$$
$$\text{y satisfacer el nivel de granularidad}$$

Nota 3. Por último, señalemos que la imposición de *i)*, se interpreta como la extensión difusa de la restricción de integridad de identidad del caso clásico, (ver página 18).

Estamos en condiciones de demostrar el siguiente importante resultado, que nos dice que en un proceso de proyección difusa para eliminar la redundancia originada por la presencia de una dependencia en la forma $X \stackrel{(\alpha,\beta)}{\Longrightarrow} Y$, X pasa a ser llave difusa de la relación proyectada. Incluimos explícitamente la imposición de que la relación de partida debe satisfacer las restricciones de integridad (los datos difusos presentes en los atributos de una d.f.d.r deben estar normalizados y satisfacer el nivel de granularidad), aunque no es necesario ya que era una restricción impuesta para toda relación.

Teorema 4.3.2 *Sea r una relación con esquema $REL \supseteq X, Y$ sin tuplas redundantes y satisfaciendo las restricciones de integridad. Supongamos que $X \stackrel{(\alpha,\beta)}{\Longrightarrow} Y$. Entonces X es una llave difusa para la relación $\overset{\sim X}{\Pi}_{XY}(r)$.*

<ins>Demostración</ins>.

La parte *i)* de la definición de llave difusa, se sigue sin más que aplicar el apartado *iii)* de la proposición 4.1.12, que nos dice que todo valor difuso de X_h en la relación $\overset{\sim X}{\Pi}_{XY}(r)$, es igual a algún valor de X_h en la relación r. Si r satisface el nivel de granularidad, entonces también se satisface en la relación proyectada.

ii) se obtiene aplicando el primer punto de la proposición 4.1.9.

Finalmente, *iii)* nos lo da el punto *iii)* del teorema 4.2.11

□

Nota. Es importante destacar que el teorema 4.2.11 nos dice que $RULE^X_{XY}(r)$ satisface la dependencia $X \stackrel{(\alpha,\beta)}{\Longrightarrow} Y$. Sin embargo, X <ins>no</ins> es llave difusa en esta relación. El único punto que se viola de la definición es *i)*, ya que los valores de X, pueden violar el nivel de granularidad (ver ejemplo 4.2.9 en la página 203).

Capítulo 5

Normalización en una Base de Datos Relacional Difusa

En el capítulo anterior vimos cómo eliminar redundancia en una base de datos relacional difusa, definiendo el operador de proyección difusa así como las dependencias basadas en reglas y se introdujo el operador relacional difuso *RULE*, y el concepto de llave difusa.

En este capítulo introduciremos la extensión difusa que proponemos para el operador de reunión natural, de forma que se justifique a través de una interpretación semántica adecuada. Es importante destacar que se definirá en términos difusos, y no en base a un criterio de igualdad estricta. A continuación demostraremos el teorema fundamental de conservación de dependencias y el de descomposiciones sin pérdidas difusas, que nos llevará también a concluir cuáles son las formas normales a las que se ha de llegar en un diseño de una base de datos relacional difusa.

5.1 El Operador de Reunión Difusa

5.1.1 Planteamiento del Problema

Vimos en el primer capítulo que la normalización en el modelo relacional clásico, pretende ser un mecanismo por el que el diseño de la B.D responda a la filosofía del modelo relacional, en cuanto que cada tupla represente una

Asociación: *Clave* $\rightarrow$ *Propiedad*

La clave vendría dada o identificada por la llave, y la propiedad corresponde a los atributos que no forman parte de la llave. Esta asociación se modela matemáticamente a través del concepto de dependencia funcional, y en el proceso de normalización se pretende aislar todas las dependencias funcionales no completas o transitivas presentes en una relación r. Esto se realiza a través de la descomposición (utilizando el operador de proyección) de r, de forma que podamos recuperar la información originalmente presente en r a través de la reunión natural de las relaciones proyectadas.

En el modelo de base de datos relacional difuso, debemos seguir los mismos pasos del modelo clásico. En el capítulo anterior introdujimos la definición que adoptaremos para la dependencia funcional difusa, así como el operador de proyección difusa. Debemos ahora ver en qué términos se realiza la descomposición y la reunión natural difusa. En general, la reunión se utiliza para extraer información de la base de datos, fundamentalmente en los procesos siguientes:

1. En el planteamiento de consultas, especificadas a través de una condición que involucra una comparación Θ.

2. En el proceso de diseño: cuando una relación presenta una dependencia no deseable, se descompone en dos relaciones, de forma que se puedan recuperar los datos que había en la relación original, a través de un proceso de reunión de las relaciones descompuestas.

En el caso clásico, se utiliza el concepto de Θ-reunión para abordar el primer punto. El segundo se soluciona con la utilización de la reunión natural, que es una equi-reunión (Θ es la igualdad) sobre los atributos comunes de las relaciones[1]. Esto ya lo vimos en la ecuación 1.11 del primer capítulo. Gráficamente:

$$\begin{array}{lcl} \Theta\text{-reunión} & \longrightarrow & \text{Consultas con condición } \Theta \\ \text{Reunión natural} & \longrightarrow & \left\{ \begin{array}{l} \text{Consultas con condición } = \\ \text{Recuperación datos originales} \end{array} \right. \end{array}$$

En el caso difuso, diversos autores han planteado operadores de reunión para resolver el primer punto (dimos una revisión en el capítulo segundo), pero sólo Weiyi [88] define un operador de reunión natural difuso, susceptible de aplicarse para resolver el segundo punto. Sin embargo, ya analizamos en la página 68 las desventajas que presentaba esta aproximación.

En definitiva, surge la necesidad de introducir un operador de reunión natural difusa, o lo que es lo mismo, de $\tilde{\Theta}$-reunión dónde $\tilde{\Theta}$ es una extensión difusa de la igualdad, de forma que podamos aplicarlo en el contexto del diseño de una base de datos relacional difusa. Obviamente, al igual que en el caso clásico, dicho operador también podrá aplicarse para responder consultas que involucren una comparación de igualdad difusa:

$$\begin{array}{lcl} \tilde{\Theta}\text{-reunión} & \longrightarrow & \text{Consultas difusas con condición } \tilde{\Theta} \\ \text{Reunión difusa natural} & \longrightarrow & \left\{ \begin{array}{l} \text{Consultas con condición igualdad difusa} \\ \text{Recuperación datos originales} \end{array} \right. \end{array}$$

En nuestra aproximación, al operador de reunión difusa natural, lo denotaremos por $\sqsupset\!\tilde{\bowtie}$, y en su definición impondremos ciertas condiciones específicas para su aplicabilidad, fundamentalmente dos:

1. Es un operador *dirigido* en el sentido de que $r \sqsupset\!\tilde{\bowtie}\, s \neq s \sqsupset\!\tilde{\bowtie}\, r$.

2. El conjunto de atributos sobre los que se realiza la reunión ha de ser llave difusa de s.

[1] Por supuesto, también puede aplicarse la reunión natural en un proceso de planteamiento de consultas, pero no al revés, es decir, un operador arbitrario de Θ-reunión no puede aplicarse en el proceso de diseño

Como podemos comprobar, la segunda es una condición que surge de forma natural en el proceso de descomposición clásico. Sin embargo, la primera imposición no tiene sentido en el caso clásico, excepto en la definición del operador de reunión exterior (en inglés *outer join*). Esto se debe a que en el modelo crisp, la reunión se define en base a la igualdad estricta entre valores; en el caso difuso, lo haremos en base al concepto de inclusión difusa, por lo que al no ser simétrica la inclusión, tampoco será simétrico el operador.

5.1.2 Reunión Dirigida

◇ *Definición de Reunión Dirigida*

Vamos a definir un primer operador de reunión, y en el siguiente apartado, introduciremos la definición definitiva de reunión difusa. Para justificar este primer operador, vamos a seguir el siguiente ejemplo:

Ejemplo 5.1.1 . Supongamos una relación r en la que se guarda información sobre el nombre (clave) y la altura (propiedad), y con una tupla igual a (`Juan` , `190`). Supongamos otra relación s dónde se guarda información sobre el peso (propiedad) según la altura (clave), con una tupla igual a (`alto` , `pesado`). Supongamos también que `190` $\subseteq$ `alto`, es decir, `190` es una de las posibles asignaciones de la etiqueta `alto`. Entonces podremos concluir que `Juan` es `pesado`. En definitiva, podemos deducir la siguiente tupla:

(`Juan` , `190` , `pesado`)

En general, podríamos deducir que es pesada cualquier persona que tenga una altura que sea una particularización del concepto `alto`. Obsérvese que es fundamental el conocimiento de que en la relación s se guarda información sobre el peso según la altura, y en la relación r información sobre la altura de ciertas personas, para poder deducir el peso de ciertas personas según su altura. En el ejemplo, deducimos que `Juan` es `pesado`, sin perder la información de que `Juan` mide `190 cm`. Si no tuviésemos dicho conocimiento, no podríamos aplicar la interpretación clave - propiedad y por tanto deberíamos deducir la tupla

(Juan , 190 , alto , pesado)

dónde se ha duplicado el atributo Altura por cada una de las relaciones que intervienen en la reunión. Esta última, es la aproximación adoptada en [65], en el contexto de la definición de una $\widetilde{\Theta}$-reunión difusa.

■

Proponemos por tanto la siguiente definición:

Definición 5.1.2 *Sean dos relaciones r, s con esquema relacional R y S respectivamente, tales que tienen asociado el mismo esquema para los atributos comunes. Supongamos que $R \cap S = X$ es una llave difusa en s según la definición 4.3.1. Entonces se define la reunión -dirigida- entre r y s a través de X, denotado por $r \sqsupset\!\!\bowtie_X s$, o simplemente la* **reunión natural -dirigida-** *entre r y s, denotado por*

$$r \sqsupset\!\!\bowtie s$$

como una relación con esquema relacional $RS - X^s$ construida de la forma siguiente: Dadas dos tuplas t_i y t_j tales que $X^r(t_i) \subseteq X^s(t_j)$, entonces se añade la nueva tupla:

$$\{(R-X)^r(t_i), X^r(t_i), (S-X)^s(t_j)\}$$

En términos de los operadores del álgebra relacional:

$$\boxed{r \sqsupset\!\!\bowtie_X s = \Pi_{RS-X^s}(\sigma_{X^r \subseteq X^s}(R \times S))} \qquad (5.1)$$

Nota 1. La inclusión en el término $X^r(t_i) \subseteq X^s(t_j)$ de la definición 5.1.2, se entiende que es componente a componente para el caso general en el que X es un conjunto de atributos. Así mismo, al definir la reunión, se supone que se eliminan aquellas tuplas que resulten iguales (sentido clásico) en todas las componentes: esto queda expresado en la ecuación 5.1, dónde se utiliza el operador de proyección clásico (igualdad estricta entre los diferentes valores). En definitiva, el operador de reunión dirigido concatena las tuplas $t \in r$ y $u \in s$, cuando $X^r(t) \subseteq X^s(u)$ (se entiende que el operador $\subseteq$ representa la inclusión entre conjuntos difusos).

Nota 2. Obviamente este operador de reunión es difuso, en el sentido que estamos utilizando el criterio de inclusión difusa. De todas formas no lo hemos definido bajo el epígrafe de *reunión* *difusa* *natural*, ya que posteriormente introduciremos otro operador más general, bajo esta segunda denominación. Por otra parte, por comodidad en la notación, a veces omitiremos al nombrar la reunión, el término *dirigida.*

Nota 3. Si queremos aplicar las $Y-$propiedades que sobre el atributo de X en r ($REL(r) = KX$) están descritas es una relación s en la que se verifica $X \overset{(\alpha,\beta)}{\Longrightarrow} Y$, pero X no es llave difusa en s, entonces también podemos aplicar la definición de reunión dirigida, construyendo la relación auxiliar:

$$\tilde{s}=\widetilde{\Pi}^{X}_{XY}(r)$$

y ya estamos en las condiciones de la definición pues X es llave difusa de $\tilde{s}$.

Ejemplo 5.1.3 . Supongamos una relación r con $REL(r) = AB$ y llave difusa A^r y otra relación s con $REL(s) = BC$ y llave difusa B^s. Entonces, según la interpretación semántica dada al principio de este apartado, r especifica una relación entre el (los) atributo(s) B y la llave A, y s especifica una relación entre el (los) atributo(s) C y la llave B. Por lo tanto, estamos en las condiciones de la definición 5.1.2, y podemos aplicar el operador de reunión dirigida a ambas relaciones:

$$rs = r \sqsupset\!\!\!\bowtie s$$

rs debe ser una relación que describa la relación entre A y B de la misma forma que antes, y además que describa la relación entre A y C obtenida a través de B. Efectivamente, para una tupla arbitraria $t \in r$ consideremos el conjunto de tuplas

$$\{t_{i_1}, \ldots t_{i_k} \in s \ \text{ tal que } B^r(t) \subseteq B^s(t_{i_j})\}$$

Realizando lo anterior para todas las tuplas $t \in r$ obtenemos la reunión dirigida, como puede verse en la figura 5.1

Efectivamente, $r \sqsupset\!\!\!\bowtie s$ es una relación que describe la relación entre A y B de la misma forma que lo hacía r (no se ha fundido el valor de $B(t)$ con $B(t_{i_j})$), y además describe la relación entre A y C obtenida a través de B.

■

$$r = \begin{array}{|cc|}\hline A & B \\ \hline A(t) & B(t) \\ \vdots & \vdots \\ \hline \end{array} \qquad s = \begin{array}{|cc|}\hline B & C \\ \hline B(t_{i_1}) & C(t_{i_1}) \\ \vdots & \vdots \\ B(t_{i_k}) & C(t_{i_k}) \\ \vdots & \vdots \\ \hline \end{array}$$

$$r \sqsupset\!\!\!\bowtie s = \begin{array}{|ccc|}\hline A & B & C \\ \hline A(t) & B(t) & C(t_{i_1}) \\ \vdots & \vdots & \vdots \\ A(t) & B(t) & C(t_{i_k}) \\ \vdots & \vdots & \vdots \\ \hline \end{array}$$

Figura 5.1. Proceso de reunión dirigida

◇ *Relación con el Concepto de Función Difusa*

Veamos que la definición de reunión dirigida es coherente con el concepto de función difusa. Suponemos atributos simples para fijar ideas y poder seguir el esquema de la figura 5.1. Partimos de la relación s, que satisface una d.f.d.r en la forma $B \overset{(\alpha,\beta)}{\Longrightarrow} C$ dónde $REL(s) = BC$. Esta dependencia se basaba en considerar la existencia de una función difusa f en la forma:

$$f : \tilde{\mathcal{P}}\,(D_B)(s) \longrightarrow \tilde{\mathcal{P}}\,(D_C)(s)$$

dónde $\tilde{\mathcal{P}}\,(D_B)(s)$ representa el conjunto de valores difusos que aparecen en el atributo B de la relación s, y análogamente para C. Aplicando la definición 3.2.1 de función difusa, tenemos:

$$f(B(t_i)) = C(t_i) \;\Leftrightarrow\; f(x) = y \quad \forall\, x \subseteq B(t_i) \quad \forall\, y \subseteq C(t_i)$$

Ahora tenemos los consecuentes presentes en el atributo B fijos, por lo que tenemos:

$$f(B(t_i)) = C(t_i) \;\Leftrightarrow\; f(x) = C(t_i) \quad \forall\, x \subseteq B(t_i)$$

y podemos aplicar esta definición a un valor de antecedente $B(t)$ presente en la relación r. Así pues, podemos deducir lo siguiente:

$$f(B(t_i)) = C(t_i) \Leftrightarrow f(B(t)) = C(t_i) \quad \forall\, B(t) \in r \text{ tal que } B(t) \subseteq B(t_i) \tag{5.2}$$

En definitiva, la ecuación 5.2, nos dice que, para ser coherentes con el concepto de función difusa, el criterio de reunión de una tupla $t \in r$ con otra tupla $t_i \in s$ debe ser, efectivamente, el de inclusión difusa.

◇ *Restricción sobre la Interpretación de los Datos*

En la definición de reunión, hemos asumido tácitamente la interpretación de un valor difuso como un conjunto de posibles asignaciones mutuamente excluyentes (ver ejemplo 5.1.1). Sin embargo el criterio debería ser distinto si la interpretación del difuso fuese la de un conjunto de posibles asignaciones mutuamente **no** excluyentes. Esto lo vemos con el siguiente ejemplo. Supongamos que hay una dependencia entre el idioma hablado y el sueldo:

`Idioma` $\longrightarrow$ `Sueldo`

Si tenemos únicamente la información

`inglés` $\longrightarrow$ `alto`

entonces, ante una entrada de una persona que hable los idiomas `inglés y francés`, también podremos concluir que tiene un salario alto, ya que en particular, habla inglés. Entonces el criterio utilizado no es el de inclusión difusa, sino que tengan únicamente intersección no vacía. En cualquier caso, este es un tema que necesita una investigación más profunda, por lo que no lo vamos a tratar en esta memoria.

◇ *Relación con el Operador de Reunión Cualificada de Codd*

Recordemos que en el caso clásico, el operador de reunión cualificada de Codd (en inglés *Maybe Natural Join*) visto en la ecuación 1.12 de la página 26, reunía dos tuplas $t \in r$ y $u \in$

s, cuando alguno de los valores de $X = REL(r) \cap REL(s)$ era `nulo` (es importante recordar que la reunión se dirige en ambos sentidos). Esto se hacía bajo la interpretación del nulo como valor `desconocido`. En este caso, si $X^s(u)$=`desconocido`, entonces, en términos de posibilidad, cualquier valor $X^r(t)$ es una posible interpretación de $X^s(u)$, y por ello se reunen las tuplas t y s. En la definición de la reunión dirigida, seguimos una interpretación análoga, reuniendo las tuplas t y s porque $X^r(t)$ sea una posible interpretación de $X^s(u)$. En definitiva, la reunión dirigida puede considerarse una extensión del operador *Maybe Natural Join*, cuando trabajamos con valores difusos en general y dirigiendo la reunión únicamente en un sentido.

5.1.3 Reunión Difusa Dirigida

◇ *Definición*

En el ejemplo 5.1.3 puede verse claramente (ver figura 5.1) que al realizar la reunión dirigida entre dos relaciones, aparecen valores $A(t)$, $B(t)$ iguales, de forma que si por ejemplo la llave difusa de r era todo el esquema (o parte de él), ya no lo será en la nueva relación $r \sqsupset\!\!\bowtie s$. Esto se debe al hecho de que una tupla $t \in r$ puede reunirse con más de una tupla de s, ya que el criterio seguido al definir la reunión es la inclusión difusa. Esto no sucedía en el caso clásico, siempre claro está, bajo las mismas hipótesis de que el conjunto de atributos sobre los que se realiza la reunión sea llave de la relación s, puesto que en dicho caso, cada tupla de r se reune con una única tupla de s.

Por lo tanto, si queremos mantener la llave difusa que hubiese originalmente en r, debemos eliminar redundancia de la relación resultado de la reunión dirigida, por lo que debemos proyectar sobre el conjunto de atributos AB. En general, deberemos realizar:

$$\tilde{\Pi}^{REL(r)} (r \sqsupset\!\!\bowtie s)$$

Si la relación r no tenía tuplas redundantes, este proceso de proyección únicamente eliminará la redundancia proveniente estrictamente de la reunión dirigida.

Ejemplo 5.1.4 . Siguiendo el ejemplo 5.1.3, tenemos:

$$\widetilde{\Pi}^{AB}(r \mathrel{\widetilde{\sqsupset\!\bowtie}} s) = \begin{array}{|c|c|c|}\hline A & B & C \\ \hline A(t) & B(t) & C(t_{i_1}) \vee \ldots \vee C(t_{i_k}) \\ \vdots & \vdots & \vdots \\ \hline\end{array}$$

Figura 5.2. Proceso de reunión difusa dirigida

Observar que, intuitivamente, el proceso de fusión de los valores de C está bien definido ya que C depende funcionalmente de B y por tanto los valores $C(t_{i_j})$ eran semejantes fuertes y la unión de ellos verificará el nivel de granularidad. Posteriormente (teorema 5.1.6) se demostrará formalmente este hecho.

■

Este razonamiento nos lleva a introducir la siguiente definición de reunión difusa dirigida, bajo las mismas hipótesis que se impusieron en la definición 5.1.2 de reunión dirigida:

Definición 5.1.5 *Sean dos relaciones r, s con esquema relacional R y S respectivamente, tales que tienen asociado el mismo esquema para los atributos comunes. Supongamos que $R \cap S = X$ es una llave difusa según la definición 4.3.1 en s. Entonces se se define la reunión difusa -dirigida- entre r y s a través de X, denotado por $r \mathrel{\widetilde{\sqsupset\!\bowtie}}_X s$, o simplemente la* **reunión natural difusa -dirigida-** *entre r y s, y lo notaremos por*

$$r \mathrel{\widetilde{\sqsupset\!\bowtie}} s$$

como una relación con esquema relacional $RS - X^s$ construida de la forma siguiente:

$$r \mathrel{\widetilde{\sqsupset\!\bowtie}} s = \widetilde{\Pi}^{R}(r \sqsupset\!\bowtie s) = \widetilde{\Pi}^{R}_{R\,S-X^s}(\sigma_{X^r \subseteq X^s}(R \times S)) \tag{5.3}$$

Nota 1. A partir de ahora, utilizaremos únicamente el término *reunión difusa*, para designar el operador de reunión natural difusa.

Nota 2. Hemos de observar en esta definición, que al suprimir el subíndice en la proyección $\widetilde{\Pi}^{R}$, se entiende que se proyecta sobre todo el esquema RS de la nueva relación.

Finalmente, obsérvese que el formato de este operador, corresponde al formato genérico que debería verificar la definición de un operador de reunión natural difusa (ecuación 2.9 de la página 65). Este consistía en aplicar un operador de proyección difusa sobre $R\ S-X^s$ a la relación resultante de una Θ-selección sobre el producto cartesiano $R \times S$. Así pues, basta con utilizar el operador de proyección difusa $\overset{\sim R}{\Pi}_{R\ S-X^s}$ y tomar como Θ la inclusión difusa.

◇ *Interpretación de la Reunión Difusa como Sistema de Inferencia*

Vamos a ver la interpretación semántica de la reunión difusa desde el punto de vista de un sistema de inferencia difuso. Supongamos una relación s en las condiciones de la definición 5.1.2. Para fijar ideas, sigamos el ejemplo 5.1.3 con $REL(s) = BC$ y llave difusa B. Podemos interpretar la relación s como un conjunto de *reglas con antecedentes y consecuentes difusos*:

$$\textbf{Si } B(t_i) \textbf{ entonces } C(t_i)$$

dónde t_i recorre todas las tuplas de s, y la *implicación de la regla* __*no*__ *es difusa.* Ante un dato de entrada $B(t)$, se escogerían aquellas reglas aplicables, y deduciríamos los consecuentes respectivos de las reglas aplicadas. Un criterio de selección de dichas reglas podría ser el de la inclusión difusa de la entrada y los antecedentes de las reglas, es decir, se seleccionarían aquellas reglas cuyos antecedentes verificasen $B(t_i) \supseteq B(t)$. Este proceso quedaría gráficamente como sigue:

$$\begin{array}{l} \text{Si } B(t_{i_1}) \quad \text{entonces } C(t_{i_1}) \\ \qquad B(t) \\ \hline \qquad\qquad\qquad\qquad C(t_{i_1}) \end{array}$$

$$\vdots$$

$$\begin{array}{l} \text{Si } B(t_{i_k}) \quad \text{entonces } C(t_{i_k}) \\ \qquad B(t) \\ \hline \qquad\qquad\qquad\qquad C(t_{i_k}) \end{array}$$

Criterio para establecer la d.f.d.r (*Consistencia de las reglas*) : **Semejanzas débil y fuerte**	Criterio para establecer la reunión (*Aplicación de las reglas*) : **Inclusión difusa**

Figura 5.3. Equivalencia entre criterios

Como resultado de la inferencia sobre cada regla, se obtiene como deducción el valor $C(t_i)$. Obviamente, el dato $B(t)$ no varía por el hecho de aplicar un conjunto de reglas, por lo que debe permanecer tal y como estaba: este esquema correspondería al operador de reunión dirigida (ver figura 5.1). Si queremos deducir un único consecuente, entonces podemos fusionar los valores $C(t_i)$ a través de la unión difusa: este esquema correspondería al operador de reunión difusa dirigida (ver figura 5.2).

Recordemos que esta interpretación de reglas difusas, ya se estableció al aplicar el operador $RULE$ a una relación con una d.f.d.r. De hecho, en un proceso de descomposición, la relación s será la relación $RULE^X_{XY}(rel - orig)$ o si se prefiere $\overset{\sim X}{\Pi}_{XY} (rel - orig)$.

Obsérvese que no *construimos* un consecuente en base al antecedente, sino que directamente se *dispara* la regla, deduciéndose el mismo consecuente que aparece en dicha regla. Es decir, estaríamos en el marco de un *sistema de inferencia no difuso*. Esto corresponde a un sistema de inferencia restrictivo, pero es necesario para poder recuperar la información original a través de los operadores de proyección y reunión, como se verá en el apartado 5.2. Además, el sistema de reglas no puede ser cualquiera, sino que ha de verificarse la condición sobre la dependencia funcional difusa basada en reglas, y por tanto antecedentes semejantes débiles, han de tener consecuentes semejantes fuertes. Diríamos que dicho sistema de reglas es *consistente*. Podemos resumir todos estos criterios en la figura 5.3.

En la actualidad, el tema de la deducción de información de la base de datos (sin aplicaciones al diseño), es de gran interés: como marco de referencia se recomienda consultar los trabajos de Pons [63] y [61].

5.1.4 Propiedades

◇ *Asociatividad*

Es inmediato comprobar que el operador de reunión (difusa o no) dirigida es asociativo cuando los atributos sobre los que se realiza la reunión no se solapan; es decir, en general, para tres relaciones cualesquiera r, s, w se tiene que:

$$X \cap X' = \emptyset \;\Rightarrow\; (r \sqsupset\!\!\bowtie_X s) \sqsupset\!\!\bowtie_{X'} w = r \sqsupset\!\!\bowtie_X (s \sqsupset\!\!\bowtie_{X'} w)$$

y por lo tanto

$$X \cap X' = \emptyset \;\Rightarrow\; (r \,\widetilde{\sqsupset\!\!\bowtie}_X\, s) \,\widetilde{\sqsupset\!\!\bowtie}_{X'}\, w = r \,\widetilde{\sqsupset\!\!\bowtie}_X\, (s \,\widetilde{\sqsupset\!\!\bowtie}_{X'}\, w)$$

Sin embargo, esto no ocurre si se solapan los atributos sobre los que se realiza la reunión, es decir:

$$X \cap X' \neq \emptyset \;\Rightarrow\; (r \sqsupset\!\!\bowtie_X s) \sqsupset\!\!\bowtie_{X'} w \neq r \sqsupset\!\!\bowtie_X (s \sqsupset\!\!\bowtie_{X'} w)$$

y por lo tanto

$$X \cap X' \neq \emptyset \;\Rightarrow\; (r \,\widetilde{\sqsupset\!\!\bowtie}_X\, s) \,\widetilde{\sqsupset\!\!\bowtie}_{X'}\, w \neq r \,\widetilde{\sqsupset\!\!\bowtie}_X\, (s \,\widetilde{\sqsupset\!\!\bowtie}_{X'}\, w)$$

Para verlo basta considerar las relaciones siguientes

$r =$

B	A
b	a

$s =$

A	D
$\tilde{a}$	d

$w =$

A	E
$\tilde{a}'$	e

Si consideramos que $a \,\widetilde{\subseteq}\, \tilde{a}$, $\tilde{a}' \,\widetilde{\subseteq}\, \tilde{a}$ entonces:

$(r \sqsupset\!\!\bowtie_A s) \sqsupset\!\!\bowtie_A w =$

B	A	D	E
b	a	d	e

mientras que:

$$r \sqsupset\!\!\bowtie_A (s \sqsupset\!\!\bowtie_A w) = \emptyset$$

Este comportamiento obedece a que, semánticamente, se ha establecido que el proceso de reunión es un proceso de inferencia que tiene como finalidad preservar la interpretación de cada relación como una asociación entre una clave y la correspondiente propiedad. Por lo tanto, el resultado de realizar

$$s \sqsupset\!\!\bowtie_A w$$

es una relación que describe las propiedades que sobre X estaban descritas en w, y además, las propiedades entre los atributos originales presentes en r y la llave de r. En el ejemplo, no había ninguna propiedad que pudiese inferirse para $\widetilde{a}$, ya que sólo se tenía información sobre el valor $\widetilde{a}'$ (incluido en $\widetilde{a}$). Sin embargo, al realizar la reunión

$$r \sqsupset\!\!\bowtie_A s$$

sí tenemos información sobre a, ya que $a \subseteq \widetilde{a}$ y por tanto podemos deducir la tupla (b, a, d). Realizando el mismo razonamiento con $(r \sqsupset\!\!\bowtie_A s) \sqsupset\!\!\bowtie_A w$ obtendríamos la tupla (b, a, d, e). En definitiva, el hecho de que el operador de reunión difusa no sea asociativo resulta coherente con el proceso natural de aplicación de propiedades.

◇ *Conservación de Dependencias*

Un teorema fundamental que debemos demostrar es que el proceso de reunión difusa conserva las dependencias funcionales difusas basadas en reglas. Esto lo vemos en la parte *i)* del siguiente teorema. La parte *ii)* nos dice que el proceso de reunión difusa dirigida, es coherente con la imposición del nivel de granularidad y que sigue conservándose también la d.f.d.r, o lo que es lo mismo, la interpretación clave propiedad para la relación resultante de la reunión. Esto se ve en los puntos *ii.a)* y *ii.b)*. Los apartados *c)* y *d)* nos dicen que en la reunión difusa, se obtienen los mismos valores que había en r y que por lo tanto, se conservan las llaves difusas presentes en la relación r (aunque no en general las de s).

Teorema 5.1.6 (fundamental de conservación de dependencias) *Sean r y s dos relaciones con $REL(r) \cap REL(s) \neq \emptyset$. Denotemos por rs a la relación $r \sqsupset\!\!\bowtie s$, y por $\widetilde{rs}$ a la relación $r \,\widetilde{\sqsupset\!\!\bowtie}\, s$. Supongamos que s satisface $X \overset{(\alpha,\beta)}{\Longrightarrow} Y$. Entonces, se verifica lo siguiente:*

i) rs satisface $X \overset{(\alpha,\beta)}{\Longrightarrow} Y$

ii) Si r no tiene tuplas redundantes, entonces:

a) $\widetilde{rs}$ satisface $X \overset{(\alpha,\beta)}{\Longrightarrow} Y$.

b) Si r o s satisface el nivel de granularidad en un atributo, entonces las relaciones $\widetilde{rs}$ y rs también lo satisface.

c) $\widetilde{rs}\,|_{REL(r)} = r$, es decir, $\Pi_{REL(r)}(\widetilde{rs}) = r$.

d) Si K es llave difusa de r, entonces K es llave difusa de $\widetilde{rs}$

Demostración.

Denotemos por X a la intersección de los esquemas relacionales, sobre los que se realiza la reunión, es decir:

$$X \equiv REL(r) \cap REL(s)$$

En el caso de que X e Y tengan intersección común, entonces, aplicando la definición 4.2.5 de d.f.d.r (página 194), tenemos:

$$s \text{ satisface } X \overset{(\alpha,\beta)}{\Longrightarrow} Y \;\Leftrightarrow\; s \text{ satisface } X \overset{(\alpha,\beta_{Y-X})}{\Longrightarrow} Y - X$$

$$\widetilde{rs} \text{ satisface } X \overset{(\alpha,\beta)}{\Longrightarrow} Y \;\Leftrightarrow\; \widetilde{rs} \text{ satisface } X \overset{(\alpha,\beta_{Y-X})}{\Longrightarrow} Y - X$$

por lo que, en definitiva, podemos restringir la demostración del apartado *i)* al caso de $X \overset{(\alpha,\beta)}{\Longrightarrow} Y$ con $X \cap Y = \emptyset$.

i)

Para demostrar que $X \overset{(\alpha,\beta)}{\Longrightarrow} Y$, aplicamos la definición 4.2.5 de d.f.d.r, por lo que consideramos dos clases de tuplas arbitrarias t_1 y t_2 en rs semejantes débiles (recordar definición 4.2.1):

$$[t_1]^{X^{rs}} \simeq^d_\alpha [t_2]^{X^{rs}} \tag{5.4}$$

Debemos demostrar que

$$Y^{rs}(t_1) \simeq^f_\beta Y^{rs}(t_2)$$

Para ello, vamos a ver cómo se han construido los valores $Y^{rs}(t_i)$: por definición de reunión dirigida sobre X, las tuplas t_1, $t_2 \in rs$ verifican:

$$\exists\ p_1 \in r \ \text{ tal que } X^r(p_1) = X^{rs}(t_1)$$

$$\exists\ p_2 \in r \ \text{ tal que } X^r(p_2) = X^{rs}(t_2)$$

y si consideramos el conjunto de tuplas de s con las que se han reunido:

$$H^1 = \{w^1_j \in s \ \text{ tal que } X^r(p_1) \subseteq X^s(w^1_j)\}$$

$$H^2 = \{w^2_j \in s \ \text{ tal que } X^r(p_2) \subseteq X^s(w^2_j)\}$$

entonces:

$$Y^{rs}(t_1) = Y^s(w^1_{j0}) \text{ con } w^1_{j0} \in H^1$$

$$Y^{rs}(t_2) = Y^s(w^2_{j0}) \text{ con } w^2_{j0} \in H^2$$

Así pues, tenemos que demostrar que $Y^s(w^1_{j0}) \simeq^f_\beta Y^s(w^2_{j0})$. Para ello, vamos a ver que

$$[w^1_{j0}]^{X^s} \simeq^d_\alpha [w^2_{j0}]^{X^s}$$

ya que, en dicho caso, aplicando la hipótesis de d.f.d.r en la relación s, se tendría que $Y^s(w^1_{j0}) \simeq^f_\beta Y^s(w^2_{j0})$. Así pues, habrá que demostrar (según la definición de clases de tuplas semejantes) que:

$$\forall\ h \ \ \exists\ w^{h,1,j_0} \in [w^1_{j0}]^{X^s_h} \ \ \exists\ w^{h,2,j_0} \in [w^2_{j0}]^{X^s_h} \ \text{ tal que}$$

$$w^{h,1,j_0} \simeq^d_\alpha w^{h,2,j_0} \tag{5.5}$$

En primer lugar, tenemos que:

$$w^1_{j0} \in H^1 \ \Rightarrow X^r(p_1) \subseteq X^s(w^1_{j0})$$

y también se verifica que $X^r(p_1) \subseteq X^s(w^1_j) \ \ \forall\ j$, por lo que:

$$X^s_h(w^1_{j0}) \cap X^s_h(w^1_j) \neq \emptyset \ \forall\, h \ \forall\, j \ \Rightarrow \ [w^1_{j0}]^{X^s_h} = [w^1_j]^{X^s_h} \ \forall\, h \ \forall\, j \ \Leftrightarrow \ [w^1_{j0}]^{X^s} = [w^1_j]^{X^s} \ \forall\, j$$

y realizando el mismo razonamiento es obvio que:

$$[w_j^1]^{X_h^s} = [w_{j'}^1]^{X_h^s} \quad \forall\ j, j'$$

Teniendo en cuenta esto, la ecuación 5.5 que debemos demostrar se reescribe como sigue:

$$\forall\ h \quad \exists\ w^{h,1} \in [w_j^1]^{X_h^s} \quad \exists\ w^{h,2} \in [w_j^2]^{X_h^s} \text{ (j arbitrario)} \quad \text{tal que}$$

$$X_h^s(w^{h,1}) \simeq_\alpha^d X_h^s(w^{h,2}) \tag{5.6}$$

Para demostrarlo, partimos de la ecuación 5.4, que por definición de clases de tuplas semejantes nos dice lo siguiente:

$$\forall\ h \quad \exists\ t^{1,h} \in [t_1]^{X_h^{rs}} \quad \exists\ t^{2,h} \in [t_2]^{X_h^{rs}} \quad \text{tal que}$$

$$X_h^{rs}(t^{1,h}) \simeq_\alpha^d X_h^{rs}(t^{2,h}) \tag{5.7}$$

Por otra parte, si aplicamos la definición de reunión dirigida a las tuplas $t^{1,h}, t^{2,h} \in rs$ obtenemos:

$$\exists\ p^{1,h} \in r \quad \text{tal que } X_h^r(p^{1,h}) = X_h^{rs}(t^{1,h})$$

$$\exists\ p^{2,h} \in r \quad \text{tal que } X_h^r(p^{2,h}) = X_h^{rs}(t^{2,h})$$

por lo que también serán:

$$X_h^r(p^{1,h}) \simeq_\alpha^d X_h^r(p^{2,h})$$

Además, $t^{1,h} \in [t_1]^{X_h^{rs}}$ nos dice que existe una secuencia de tuplas $\{t_k^{1,h} \in rs\}_k$ tales que sus núcleos (para X_h^{rs}) se solapan, empezando en t_1 y terminando en $t^{1,h}$, o lo que es lo mismo, que existe una secuencia de tuplas $\{p_k^{1,h} \in r\}_k$ tales que sus núcleos (para X_h^r) se solapan, empezando en p_1 y terminando en $p^{1,h}$. Aplicando de nuevo la definición de reunión dirigida, tenemos:

$$X_h^r(p_1) \subseteq X_h^s(w_j^1) \quad \forall\ j$$

$$X_h^r(p^{1,h}) \subseteq X_h^s(w_j^{1,h}) \quad \forall\ j$$

$$X_h^r(p_k^{1,h}) \subseteq X_h^s(w_j^{1,k,h}) \quad \forall\ j$$

por lo que también se verificará que existe una secuencia de tuplas $\{w_j^{1,k,h} \in s\}_k$ tales que sus núcleos (para X_h^s) se solapan, empezando en w_j^1 y terminando en $w_j^{1,h}$, y por tanto:

$$\forall\ h \quad w_j^{1,h} \in [w_j^1]^{X_h^s}$$

Si razonamos análogamente con t_2 y reunimos los anteriores resultados obtenemos:

$$X_h^r(p^{1,h}) \subseteq X_h^s(w_j^{1,h})$$
$$X_h^r(p^{2,h}) \subseteq X_h^s(w_j^{2,h})$$

Pero $X_h^r(p^{1,h}) \simeq_\alpha^d X_h^r(p^{2,h})$ y por tanto, aplicando la proposición 3.2.10 tenemos que

$$X_h^s(w_j^{1,h}) \simeq_\alpha^d X_h^s(w_j^{2,h})$$

por lo que hemos demostrado que:

$$\forall\ h \ \ \exists\ w_j^{1,h} \in [w_j^1]^{X_h^s} \text{ y } \exists\ w_j^{2,h} \in [w_j^2]^{X_h^s} \text{ tal que } X_h^s(w_j^{1,h}) \simeq_\alpha^d X_h^s(w_j^{2,h})$$

que es lo que se pretendía en 5.6

ii)

ii.a) Debemos demostrar que $\widetilde{rs}$ satisface $X \overset{(\alpha,\beta)}{\Longrightarrow} Y$.

Para ello, consideremos dos clases de tuplas arbitrarias en $\widetilde{rs}$ semejantes débiles:

$$[u_1]^{X^{\widetilde{rs}}} \simeq_\alpha^d [u_2]^{X^{\widetilde{rs}}}$$

Debemos demostrar que

$$Y^{\widetilde{rs}}(u_1) \simeq_\beta^f Y^{\widetilde{rs}}(u_2)$$

Para ello, vamos a ver cómo se han construido los valores $Y^{\widetilde{rs}}(u_i)$. Para facilitar la notación, consideremos:

$$Z \equiv REL(r) - X \ , \ R \equiv REL(r) = XZ$$

Una tupla $p \in r$ se puede fundir con varias tuplas $w_i^p \in s$, por lo que puede aparecer repetido en varias tuplas $t_i \in rs$ el conjunto de valores $R^r(p)$ (por ejemplo, ver figura 5.1 con $Z = A$, $X = B$, $Y = C$) es decir:

$$t_i \in rs \ \Leftrightarrow \ t_i = (R^{rs}(t_i), Y^{rs}(t_i)) = (R^r(p), Y^s(w_i^p)) \tag{5.8}$$

Si realizamos ahora la R-proyección difusa sobre rs, dichas tuplas (t_i) se fundirán en una única tupla $u \in \widetilde{rs}$, ya que $R^{rs}(t_i) = R^{rs}(t_{i'}) = R^r(p)$:

$$u = (R^{\widetilde{rs}}(u), Y^{\widetilde{rs}}(u)) = \left(R^r(p), \bigvee_i Y^{rs}(t_i) = \bigvee_i Y^s(w_i^p) \right) \in \widetilde{rs} \qquad (5.9)$$

y como no hay tuplas redundantes en r por hipótesis, entonces se verificará que:

$$\nexists\ p, p' \in r \quad \text{tal que } R^r(p)\ \mathcal{R}\ R^r(p')$$

por lo que al realizar la R-proyección estaremos fundiendo únicamente las tuplas de rs con valores iguales de XZ, es decir, aquellos valores iguales originados por la reunión dirigida. En definitiva, todas las tuplas de $\widetilde{rs}$ son de la forma dada en la ecuación 5.9 (con t_i y p dadas por la ecuación 5.8):

$$u \in \widetilde{rs} \Leftrightarrow \ u = \left(R^r(p), \bigvee_i Y^s(w_i^p) \right) \ \text{con}$$

$$R^{\widetilde{rs}}(u) = R^r(p) = R^{rs}(t_i) \ , \quad Y^{\widetilde{rs}}(u) = Y^s(w_i^p) = Y^{rs}(t_i) \qquad (5.10)$$

Recuperando las tuplas u_1, u_2 que habíamos tomado al principio, tendremos que:

$$\exists\ t^1,\ t^2 \in rs \text{ tales que } R^{\widetilde{rs}}(u_1) = R^{rs}(t^1)\ ,\ R^{\widetilde{rs}}(u_2) = R^{rs}(t^2)$$

por lo que:

$$[u_1]^{X^{\widetilde{rs}}} \simeq_\alpha^d [u_2]^{X^{\widetilde{rs}}} \ \Rightarrow\ [t^1]^{X^{rs}} \simeq_\alpha^d [t^2]^{X^{rs}}$$

Aplicando la parte *i)* del teorema, tenemos que

$$Y^{rs}(t_k^1) \simeq_\beta^f Y^{rs}(t_{k'}^2) \quad \forall\ t_k^1 \in [t^1]^{X^{rs}}\ , \quad \forall\ t_{k'}^2 \in [t^2]^{X^{rs}} \qquad (5.11)$$

Ahora bien, el valor $Y^{\widetilde{rs}}(u_1)$ es de la forma:

$$Y^{\widetilde{rs}}(u_1) = \bigvee_i Y^{rs}(t_i^1)$$

y por otra parte:

$$Y^{\widetilde{rs}}(u_2) = \bigvee_j Y^{rs}(t_j^2)$$

Como $R^{rs}(t_i^1) = R^{rs}(t_{i'}^1) = R^{\widetilde{rs}}(u_1) = R^{rs}(t^1) \quad \forall\ i, i'$ entonces $X^{rs}(t_i^1) = X^{rs}(t_{i'}^1) = X^{\widetilde{rs}}(u_1) = X^{rs}(t^1) \quad \forall\ i, i'$ por lo que podemos concluir que

$$[t_i^1]^{X^{rs}} = [t_{i'}^1]^{X^{rs}} = [t^1]^{X^{rs}} \quad \forall\ i, i' \Leftrightarrow\ t_i^1 \in [t^1]^{X^{rs}} \quad \forall\ i$$

por lo que podemos aplicar 5.11 a las tuplas t_i^1:

$$Y^{rs}(t_i^1) \simeq_\beta^f Y^{rs}(t_j^2) \quad \forall\, i,\, j$$

Aplicando la proposición 3.2.23 obtenemos finalmente que:

$$\bigvee_i Y^{rs}(t_i^1) \simeq_\beta^f \bigvee_j Y^{rs}(t_j^2)$$

o lo que es lo mismo:

$$Y^{\widetilde{rs}}(u_1) \simeq_\beta^f Y^{\widetilde{rs}}(u_2)$$

como queríamos demostrar.

ii.b) Demostremos ahora que si r o s satisface el nivel de granularidad en un atributo, entonces $\widetilde{rs}$ también lo satisface (la demostración para la relación rs es inmediata a partir de ésta).

Por definición de reunión difusa dirigida, es obvio que:

$$\forall\, u \in \widetilde{rs} \quad \forall\, A_h \in XZ \;\; \exists\, t_h \in r \;\text{ tal que } A_h^{\widetilde{rs}}(u) = A_h^r(t_h)$$

por lo que si r satisface el nivel de granularidad en el atributo $A_h \in XZ$, también lo satisface la relación $\widetilde{rs}$. Veamos pues que también se satisface el nivel de granularidad en Y; para ello consideremos una tupla arbitraria $u \in \widetilde{rs}$, que según 5.10 verificará:

$$Y^{\widetilde{rs}}(u) = \bigvee_i Y^{rs}(t_i) = \bigvee_i Y^s(w_i^p)$$

Aplicando la parte *i)* del teorema, rs satisface $X \overset{(\alpha,\beta)}{\Longrightarrow} Y$ por lo que:

$$Y^s(w_i^p) \simeq_\beta^f Y^s(w_{i'}^p) \quad \forall\, i,\, i'$$

en particular se tiene que $\vee_i Y^s(w_i^p)$ satisface el nivel de granularidad sin más que aplicar la parte *iii)* de la proposición 3.2.23.

ii.c)

Para demostrar que $\widetilde{rs}\ |_{REL(r)} = r$, basta tener en cuenta la ecuación 5.10 del apartado *ii.a)*. En particular, nos dice que:

$$u \in \widetilde{rs} \Leftrightarrow R^{\widetilde{rs}}(u) = R^r(p) \text{ con } p \in r$$

ii.d) Demostremos ahora que si K es llave difusa de r, entonces K es llave difusa de $\widetilde{rs}$ Por comodidad, omitimos los superíndices en la notación de la d.f.d.r. Si K es llave difusa de r, entonces $K^r \Longrightarrow R^r$. Aplicando *ii.c)* obtenemos que $K^{\widetilde{rs}} \Longrightarrow R^{\widetilde{rs}}$. En particular $K^{\widetilde{rs}} \Longrightarrow X^{\widetilde{rs}}$. Aplicando *ii.a)* obtenemos $X^{\widetilde{rs}} \Longrightarrow Y^{\widetilde{rs}}$ por lo que aplicando el axioma de transitividad tendremos que $X^{\widetilde{rs}} \Longrightarrow Y^{\widetilde{rs}}$. En definitiva $K^{\widetilde{rs}} \Longrightarrow REL(\widetilde{rs})$.

Además, según *ii.c)* $\widetilde{rs}\ |_{REL(r)} = r$. Como $K \subseteq R$ entonces $\widetilde{rs}\ |_K = r|_K$ por lo que si no había dos tuplas con valores de K redundantes en r tampoco las habrá en $\widetilde{rs}$.

Si, finalmente, tenemos en cuenta que aplicando la parte *ii.b)* de este teorema, la relación $\widetilde{rs}$ satisface el nivel de granularidad en los mismos atributos que r, tenemos que, en definitiva, K es llave difusa de $\widetilde{rs}$. □

5.1.5 Llaves Externas Difusas. Restricciones de Integridad

El concepto de llave externa es de vital importancia en el modelo clásico, desde el punto de vista del diseño, para establecer una restricción de integridad referencial. Por ejemplo, surge de forma natural, cuando una relación r con un esquema relacional XYZ y una dependencia funcional $X \to Y$, se descompone en $r_1 = \Pi_{XZ}(r)$ y en $r_2 = \Pi_{XY}(r)$; entonces el atributo X en r_1 es una llave externa a la relación r_2.

En el caso difuso tendremos el mismo resultado mencionado anteriormente (ver nota después del lema 5.2.8), pero en términos difusos. Introducimos por tanto el concepto de llave externa difusa como sigue:

Definición 5.1.7 *Un conjunto de atributos X se dice que forman en una relación r, una* **llave externa difusa** *a una relación s si y solo si X es una llave difusa para s y además se verifica que toda tupla $t \in r$ debe satisfacer lo siguiente:*

- *O bien* $X^r_h(t) = \texttt{nulo} \quad \forall\, X^r_h \in X$

- *O bien* $\exists\, t' \in s$ *tal que* $X^r(t) \subseteq X^s(t')$

Cuando el gestor de la base de datos impone que un conjunto de atributos $X \in r$ han de ser una llave externa difusa a una relación s, diremos que está estableciendo una **restricción de integridad referencial difusa.**

Esta es la última restricción de integridad que vamos a ver en esta memoria, por lo que resulta conveniente realizar un breve repaso de todas la restricciones que han ido apareciendo a lo largo de este trabajo:

1. *Restricción sobre la no existencia de tuplas redundantes en una relación de una base de datos relacional difusa. Página 169*

2. *Restricción de integridad de identidad sobre la normalización de los datos difusos presentes en los antecedentes y consecuentes de una dependencia funcional difusa (páginas 133 y 137) y sobre los atributos de una llave difusa (página 213).*

3. *Restricción de integridad de identidad sobre el nivel de granularidad permitido en los datos difusos presentes en los antecedentes y consecuentes de una dependencia funcional difusa (página 137) y sobre los atributos de una llave difusa (página 213).*

4. *Restricción de integridad referencial: vista en este apartado.*

5.2 Descomposición de una Relación

En esta sección vamos a introducir los términos en los que se debe plantear una descomposición de una relación. El concepto de descomposición es análogo al caso clásico y consiste en proyectar (ahora en forma difusa) una relación atendiendo a una partición del esquema relacional. A continuación extenderemos los conceptos clásicos de inclusión e igualdad entre relaciones, y los aplicaremos en el siguiente apartado para definir qué es una descomposición sin pérdidas en términos difusos. Finalmente demostraremos el teorema fundamental que nos asegure la existencia de dicha descomposición bajo la hipótesis de existencia de una dependencia funcional difusa basada en reglas. Este teorema nos asegura que podemos recuperar la *misma* información que había en la relación original sobre la que se aplicó la descomposición. Estos resultados se aplicarán en la última sección de este capítulo, para establecer una teoría de diseño para una base de datos relacional difusa.

5.2.1 El Operador de Descomposición

Vamos a centrar el estudio en el caso de una descomposición en dos relaciones. No vamos a entrar en el caso general de la descomposición de una relación cuando hay presentes más de una dependencia funcional difusa, proponiendo este tema como una vía futura de investigación.

Recordemos brevemente el caso clásico: se parte de una relación r con esquema relacional $REL(r) = \rho_1\rho_2$, verificando una dependencia del tipo $\rho_1 \cap \rho_2 \overset{(\alpha,\beta)}{\Longrightarrow} \rho_2 - \rho_1$ o bien $\rho_1 \cap \rho_2 \overset{(\alpha,\beta)}{\Longrightarrow} \rho_1 - \rho_2$. Entonces, el teorema de Heath (teorema 1.2.14) aseguraba que la descomposición de r en $\Pi_{\rho_1}(r)$ y en $\Pi_{\rho_2}(r)$ era tal que podíamos recuperar exactamente las mismas tuplas que había en r, es decir:

$$r = m_\rho(r) = \Pi_{\rho_1}(r) \bowtie \Pi_{\rho_2}(r)$$

Podemos suponer que $REL(r) = XYZ$ y $\rho_1 = XZ$, $\rho_2 = XY$, y utilizamos el término

$m(r)$ para denotar a $m_\rho(r)$. Así pues, el anterior término quedaría en la forma siguiente:

$$m(r) = \Pi_{XZ}(r) \bowtie \Pi_{XY}(r) \tag{5.12}$$

Ahora bien, el operador de proyección difusa introducido en la definición 4.1.7 necesita un segundo parámetro en las proyecciones que aparecen en 5.12. Vamos a recuperar las interpretaciones semánticas introducidas al principio de este capítulo, para determinar dichos parámetros.

* Vimos en el capítulo cuarto, que los atributos presentes en una d.f.d.r, debían aislarse en la relación siguiente:

  $$\widetilde{\Pi}^{X}_{XY}(r)$$

  Por lo tanto, deberíamos tomar como último término de 5.12 dicho operador.

  En cualquier caso, también vimos en dicho capítulo que podía utilizarse el operador $RULE$ como un operador de extracción de las reglas difusas que expliquen la dependencia, en una forma más compacta:

  $$RULE^{X}_{XY}(r)$$

  Por ahora, trabajaremos con la proyección difusa, pero sin pérdida de generalidad, ya que los teoremas que demostraremos en este capítulo, utilizando la proyección difusa, serán inmediatamente extendidos si se utiliza el operador $RULE$.

* Por otra parte, los valores de los atributos de r distintos a XY, se almacenan en la relación $\Pi_{XZ}(r)$ que aparece en la ecuación 5.12. Ahora, debemos considerar la proyección difusa sobre XZ, es decir:

  $$\widetilde{\Pi}_{XZ}(r)$$

  o lo que es lo mismo, $\widetilde{\Pi}^{XZ}_{XZ}(r)$.

* Por último, hemos de observar que en el proceso de reunión difusa, aplicamos las propiedades obtenidas en $\widetilde{\Pi}^{X}_{XY}(r)$, a los datos presentes en la relación $\widetilde{\Pi}_{XZ}(r)$, por lo que ésta última deberá aparecer a la izquierda de la reunión.

Finalmente, la reunión quedaría en alguna de las formas siguientes:

$$\widetilde{\Pi}_{XZ}(r) \mathrel{\widetilde{\sqsupset\!\!\bowtie}} \widetilde{\Pi}^{X}_{XY}(r) \;\; , \;\; \widetilde{\Pi}_{XZ}(r) \mathrel{\widetilde{\sqsupset\!\!\bowtie}} RULE^{X}_{XY}(r)$$

Definimos por tanto:

Definición 5.2.1 *Sea una relación r con esquema $REL(r) = XYZ$, dónde X, Y, Z son conjuntos de atributos con $X \neq \emptyset$. Supongamos que $X \overset{(\alpha,\beta)}{\Longrightarrow} Y$. Entonces, bajo estas condiciones, definimos el operador de* **reconstrucción difusa** *de r, y lo denotaremos por $\widetilde{m}(r)$, como sigue:*

$$\widetilde{m}(r) \equiv \widetilde{\Pi}_{XZ}(r) \mathrel{\widetilde{\sqsupset\!\!\bowtie}} \widetilde{\Pi}^{X}_{XY}(r) \tag{5.13}$$

Bajo las mismas condiciones, se define también el siguiente operador de reconstrucción:

$$\widetilde{m}^{*}(r) \equiv \widetilde{\Pi}_{XZ}(r) \mathrel{\widetilde{\sqsupset\!\!\bowtie}} RULE^{X}_{XY}(r) \tag{5.14}$$

Nota. En lo que sigue, utilizaremos la siguiente notación:

$$r_1 \equiv \widetilde{\Pi}_{XZ}(r) \;\; , \;\; r_2 \equiv \widetilde{\Pi}^{X}_{XY}(r)$$

por lo que $\widetilde{m}(r)$ se expresaría en la forma:

$$\widetilde{m}(r) \equiv r_1 \mathrel{\widetilde{\sqsupset\!\!\bowtie}} r_2 = \widetilde{\Pi}^{XZ}(r_1 \sqsupset\!\!\bowtie r_2)$$

Un primer resultado que es consecuencia inmediata del teorema 5.1.6, es el siguiente:

Teorema 5.2.2 *Sea una relación r en las condiciones de la definición 5.2.1. Entonces se verifica que:*

$$\boxed{\widetilde{m}(r) \text{ y } \widetilde{m}^{*}(r) \text{ satisfacen } X \overset{(\alpha,\beta)}{\Longrightarrow} Y} \tag{5.15}$$

Además, las relaciones anteriores verifican el nivel de granularidad en los mismos atributos que r.

Demostración.

Por una parte, por hipótesis tenemos que r satisface una d.f.d.r $X \overset{(\alpha,\beta)}{\Longrightarrow} Y$. Además, se ha supuesto implícitamente que r satisface las restricciones de integridad, por lo que estamos en condiciones de aplicar el teorema 4.2.11, que nos dice que las relaciones $\widetilde{\Pi}^{X}_{XY}(r)$ y $RULE^{X}_{XY}(r)$ satisfacen también la dependencia $X \overset{(\alpha,\beta)}{\Longrightarrow} Y$.

Por otra parte, aplicando la proposición 4.1.9, tenemos que la relación $\widetilde{\Pi}_{XZ}(r)$ no tiene tuplas redundantes respecto XZ.

Así pues, estamos en condiciones de aplicar el teorema 5.1.6 a las relaciones anteriores. Este teorema nos dice que las siguientes reuniones dirigidas satisfacen la d.f.d.r $X \overset{(\alpha,\beta)}{\Longrightarrow} Y$:

$$\widetilde{\Pi}_{XZ}(r) \mathrel{\widetilde{\sqsupset\!\bowtie}} \widetilde{\Pi}^{X}_{XY}(r) = \widetilde{m}(r)$$

$$\widetilde{\Pi}_{XZ}(r) \mathrel{\widetilde{\sqsupset\!\bowtie}} RULE^{X}_{XY}(r) = \widetilde{m}^{*}(r)$$

Además, el punto *ii.b)* de dicho teorema, nos asegura que también satisfacen el nivel de granularidad en los mismos atributos de r. □

Nota. También pueden aplicarse los apartados *ii.c), ii.d)* del teorema 5.1.6, pero no los hemos visto en el anterior resultado ya que los incluiremos en el teorema 5.3.5 que veremos en la página 268.

5.2.2 Inclusión y Extensión entre Relaciones

Hemos definido el operador de reconstrucción difusa para una relación en la que hubiesen presentes d.f.d.r. Ahora, debemos determinar qué es, en términos difusos, una descomposición sin pérdidas. En el caso clásico una descomposición es sin pérdidas cuando se obtiene igualdad estricta en la ecuación $r = m(r)$. El concepto de igualdad entre dos relaciones es inmediato, pero realmente la ecuación $r = m(r)$ debe leerse como:

La información obtenida al construir $m(r)$
es la misma que había en la relación r original

Extendemos esta afirmación al caso difuso en la siguiente forma:

$$\text{La información obtenida al construir } \widetilde{m}\ (r) \text{ engloba a la que había en la relación } r \text{ original} \tag{5.16}$$

Necesitamos formalizar qué entendemos por el término *engloba*. Para ello, vamos a considerar de nuevo el concepto de inclusión difusa, que surgió también de forma natural cuando planteábamos el concepto de reunión.

Definición 5.2.3 *Sean dos relaciones r, s con el mismo esquema relacional REL. Para una tupla arbitraria $t \in r$ denotaremos por $t^{\subseteq,s}$ al conjunto $\{u \in s \ \text{ tal que } A_h^r(t) \subseteq A_h^s(u) \ \ \forall\, A_h \in REL\}$. Entonces diremos:*

- *r está* **incluida** *en s, y lo notaremos por $r \preceq s$, si y solo si* $\forall\, t \in r\,,\ \ t^{\subseteq,s} \neq \emptyset$.
- *s es una* **extensión** *de r, y lo notaremos por $r \sqsubseteq s$, si y solo si:*

  i) $r \preceq s$

  ii) $\bigcup_{t \in r} t^{\subseteq,s} = s$

El operador $\bigcup$ utilizado en *ii)* es el operador usual de unión conjuntista de tuplas introducido en el primer capítulo (ecuación 1.5). Mientras que la inclusión garantiza que toda tupla de r queda englobada en otra tupla más difusa de s, la extensión nos garantiza que todas las tuplas en s deben contener a alguna tupla de r. Por lo tanto, la inclusión $\preceq$ es la extensión difusa del operador clásico de inclusión entre relaciones (ecuación 1.3), mientras que el concepto de extensión $\sqsubseteq$ lo utilizamos para comprobar que la información que conlleva una relación es una particularización de otra. Teniendo esto en cuenta, (5.16) debe leerse como:

$$\text{La relación } \widetilde{m}\ (r) \text{ es una extensión de la relación } r$$

Nota. La anterior afirmación deberá ser complementada con una condición sobre el nivel de granularidad (esto se verá en la página 246)

Ejemplo 5.2.4 . Consideremos las siguientes relaciones:

$$r = \begin{array}{|cc|}\hline A & B \\ \hline a_1 & b_1 \\ a_2 & b_2 \\ \hline\end{array} \quad s = \begin{array}{|cc|}\hline A & B \\ \hline a & b \\ c & d \\ \hline\end{array} \quad s' = \begin{array}{|cc|}\hline A & B \\ \hline a & b \\ \hline\end{array}$$

dónde suponemos que $a_1 \subseteq a$, $a_2 \subseteq a$, $b_1 \subseteq b$, $b_2 \subseteq b$. Entonces, es inmediato comprobar que $r \preceq s$, $r \preceq s'$, $r \not\sqsubseteq s$, $r \sqsubseteq s'$, $s \preceq s'$, $s \not\sqsubseteq s'$.

■

Una propiedad básica que debe cumplir cualquier concepto de inclusión, es la reflexividad y la transitividad; efectivamente:

Proposición 5.2.5 *Sean r, s, q tres relaciones con esquema REL. Se verifica lo siguiente:*

i) Si $r \preceq s$ y $s \preceq q$, entonces $r \preceq q$.

ii) Si $r \sqsubseteq s$ y $s \sqsubseteq q$, entonces $r \sqsubseteq q$.

iii) $r \preceq r$, $r \sqsubseteq r$ $\forall\, r$

Demostración.

La demostración de *iii)* es inmediata, por lo que vamos a probar *ii)*, y en la demostración se verá también *i)*. Tenemos que demostrar los siguientes puntos:

* $\forall\, t \in r \;\; t^{\subseteq, q} \neq \emptyset$

Dada una tupla arbitraria $t \in r$, aplicando la hipótesis de que $r \sqsubseteq s$ se tiene:

$$\exists\, u \in s \text{ tal que } A_h^r(t) \subseteq A_h^s(u) \quad \forall\, h$$

Por otra parte, para este u, aplicando la hipótesis $s \sqsubseteq q$ se tendrá:

$$\exists\, w \in q \text{ tal que } A_h^s(u) \subseteq A_h^q(w) \quad \forall\, h$$

por lo tanto:

$$\exists\, w \in q \ \text{ tal que } A_h^r(t) \subseteq A_h^q(w) \quad \forall\, h$$

por lo que finalmente obtenemos:

$$t^{\subseteq,q} \neq \emptyset$$

Hemos demostrado pues la parte *i)*, a saber:

$$\text{si } r \preceq s \text{ y } s \preceq q \text{ entonces } r \preceq q$$

* $\bigcup_{t\in r} t^{\subseteq,q} = q$

Sea una tupla arbitraria $w \in q$. Entonces, al ser $\bigcup_{u\in s} u^{\subseteq,q} = q$ se tiene que:

$$\exists\, u_0 \in s \ \text{ tal que } w \in u_0^{\subseteq,q}$$

o lo que es lo mismo:

$$\exists\, u_0 \in s \ \text{ tal que } A_h^s(u_0) \subseteq A_h^q(w) \quad \forall\, h \tag{5.17}$$

Para esta tupla u_0, al ser $\bigcup_{t\in r} t^{\subseteq,s} = s$, se tendrá que:

$$\exists\, t_0 \in r \ \text{ tal que } u_0 \in t_0^{\subseteq,s}$$

o lo que es el mismo:

$$\exists\, t_0 \in r \ \text{ tal que } A_h^r(t_0) \subseteq A_h^s(u_0) \quad \forall\, h \tag{5.18}$$

Uniendo 5.17 y 5.18 obtenemos:

$$\exists\, t_0 \in r \ \text{ tal que } A_h^r(t_0) \subseteq A_h^q(w) \quad \forall\, h$$

o lo que es lo mismo:

$$\exists\, t_0 \in r \ \text{ tal que } w \in t_0^{\subseteq,q}$$

Cómo w era una tupla arbitraria, tendremos que:

$$q \subseteq \bigcup_{t\in r} t^{\subseteq,q}$$

dónde la inclusión se entiende en el sentido clásico, considerando una relación q como un conjunto de tuplas. Por otra parte, es inmediato por la propia definición que:

$$\bigcup_{t\in r} t^{\subseteq, q} \subseteq q$$

por lo que hemos demostrado que:

$$\bigcup_{t\in r} t^{\subseteq, q} = q$$

Si unimos esto a la parte *i)*, concluimos que $r \sqsubseteq q$. □

Otras propiedades importantes en conexión con la proyección difusa y el operador $RULE$, son las siguientes:

Proposición 5.2.6 *Sea s una relación con esquema $REL \supseteq W \supseteq V$. Entonces se verifica que:*

i) $\Pi_W(s) \sqsubseteq \overset{\sim V}{\Pi_W}(s)$, *o lo que es lo mismo,* $s|_w \sqsubseteq \overset{\sim V}{\Pi_W}(s)$. *En particular,* $s \sqsubseteq \overset{\sim V}{\Pi}(s)$

ii) Si $\exists\ V'$ *tal que* $V \subseteq V' \subseteq W$, *entonces*

$$\overset{\sim V}{\Pi_W}(s) \sqsubseteq \overset{\sim V'}{\Pi_W}(s)$$

iii) $\overset{\sim V}{\Pi_W}(s) \sqsubseteq RULE_W^V(s)$

La demostración es inmediata sin más que aplicar las propias definiciones de proyección difusa y del operador $RULE$. Ahora, en relación al concepto de llave externa tenemos el siguiente resultado:

Proposición 5.2.7 *Sean r y s dos relaciones con $REL(r) \cap REL(s) \supseteq X$. Supongamos que X^r es llave externa difusa a s. Si $s \sqsubseteq w$, entonces X^r también es llave externa difusa a w.*

<u>Demostración</u>.

Observar que, según la definición 5.1.7 de llave externa difusa y de inclusión entre relaciones, se verifica lo siguiente:

$$X^r \text{ es llave externa difusa a } s \Leftrightarrow r|_X \preceq s|_X$$

Por otra parte:

$$s \sqsubseteq w \Rightarrow s|_X \sqsubseteq w|_X$$

De ambas deducimos que:

$$r|_X \preceq w|_X \Leftrightarrow X^r \text{ es llave externa difusa a } w$$

□

5.2.3 Descomposiciones sin Pérdidas Difusas

Una vez que se han introducido los conceptos de inclusión y extensión entre relaciones, estamos en condiciones de *cuantificar* la información perdida en un proceso de descomposición y reconstrucción. En el caso difuso, diremos que el proceso de reconstrucción es sin pérdidas, si se obtiene la misma información que había originalmente, aunque tal vez en términos más difusos (esto lo formalizaremos con el concepto de <u>extensión</u> introducido en la definición 5.2.3).

Sobre el concepto de *pérdida* hay que realizar la misma puntualización que en el caso clásico: no significa que se pierden tuplas en el proceso de reconstrucción, sino que puedan aparecer tuplas nuevas (*espúreas*) que no estaban presentes en la relación original; es por ello que *pérdida* hace referencia a pérdida de información originada por la presencia de tuplas espúreas.

Además tendremos que imponer que la relación obtenida tras dicho proceso verifique el nivel de granularidad en los mismos atributos que r. Esta imposición garantiza que la información que obtengamos no sea demasiado difusa y se mantenga siempre en los mismos niveles (β) que se impusieron para definir semejanza:

La precisión con la que debemos recuperar la información original, debe verificar el nivel de granularidad al mismo nivel que se aceptó para definir la semejanza entre valores difusos.

Ahora, debemos demostrar que podemos conseguir, bajo las condiciones de la definición 5.2.1 dicha descomposición. Esto lo veremos en el teorema fundamental de descomposiciones sin pérdidas (página 250), pero antes, necesitamos demostrar dos lemas previos:

Lema 5.2.8 *Sea una relación r con esquema XYZ, verificando las hipótesis de la definición 5.2.1 (página 239). Denotemos por $\bar{r}$ a la reunión dirigida $\widetilde{\Pi}_{XZ}(r) \sqsupset\!\!\bowtie \widetilde{\Pi}^{X}_{XY}(r)$. Entonces, se verifica lo siguiente:*

i) $r \preceq \bar{r}$

ii) $r \preceq \widetilde{\Pi}^{V}(\bar{r}) \quad \forall\, V$. *En particular, para $V = XZ$ tenemos:* $\boxed{r \preceq \widetilde{m}\,(r)}$

Demostración.

Vamos a seguir la notación introducida en la definición 5.2.1, es decir, $r_1 \equiv \widetilde{\Pi}_{XZ}(r)$ y $r_2 \equiv \widetilde{\Pi}^{X}_{XY}(r)$.

i) Tenemos que demostrar que

$$\forall\, t \in r \;\; \exists\, \bar{t} \in \bar{r} \;\; \text{tal que } A_h^r(t) \subseteq A_h^{\bar{r}}(\bar{t}) \quad \forall\, h$$

Dada $t \in r$ será de la forma:

$$t = (X^r(t), Y^r(t), Z^r(t))$$

Consideremos el conjunto de tuplas $(t_i) \in r$ tales que los valores de XZ se funden con t al proyectar sobre $r_1 = \widetilde{\Pi}_{XZ}(r)$ obteniéndose una tupla $t^1 \in r_1$. Obviamente $t \in (t_i)$:

$$\exists\, t^1 \in r_1 \;\; \text{tal que } XZ^{r_1}(t^1) = \bigvee_{t_i} XZ^r(t_i) \supseteq XZ^r(t)$$

por lo tanto:

$$X^{r_1}(t^1) = \bigvee_{t_i} X^r(t_i) \supseteq X^r(t)$$

lo que implica que:

$$\forall\, i \;\; X^{r_1}(t^1)\; \mathcal{R}\; X^r(t_i) \;\Rightarrow\; \forall\, i \;\; \forall\, h \;\; X_h^{r_1}(t^1)\; \mathcal{R}\; X_h^r(t_i)$$

Ahora bien, aplicando la proposición 4.1.12 tenemos que:

$$\forall\, h \;\; \exists\; t^{1,h} \in r \;\; \text{tal que } X_h^r(t^{1,h}) = X_h^{r_1}(t^1)$$

Por lo tanto, todos los $X_h^r(t_i)$ se fundirán con $X_h^r(t^1) = X_h^{r_1}(t^1)$ al proyectar sobre r_2, por lo que:

$$\exists\; t^2 \in r_2 \;\; \text{tal que } X_h^{r_2}(t^2) \supseteq \bigvee_i X_h^r(t_i) \;\Rightarrow$$

$$\forall\; t^1 \in r_1 \;\; \exists\; t^2 \in r_2 \;\; \text{tal que } X^{r_2}(t^2) \supseteq X^{r_1}(t^1) = \bigvee_{t_i} X^r(t_i) \qquad (5.19)$$

y por definición de proyección difusa, también se verificará la inclusión para los valores de Y, es decir:

$$\exists\; t^2 \in r_2 \;\; \text{tal que } XY^{r_2}(t^2) \supseteq \bigvee_{t_i} XY^r(t_i)$$

pero $t \in (t_i)$; por lo tanto:

$$XY^{r_2}(t^2) \supseteq XY^r(t) \;\Rightarrow\; Y^{r_2}(t^2) \supseteq Y^r(t)$$

Tenemos pues dos tuplas t^1 y t^2 verificando:

$$X^{r_1}(t^1) = \bigvee_i X^r(t_i) \;\; y \;\; X^{r_2}(t^2) \supseteq \bigvee_{t_i} X^r(t_i)$$

por lo que $t^1 \in r_1$ y $t^2 \in r_2$ se reunen, y el resultado de la reunión será:

$$t^1 \sqsupset\!\!\bowtie t^2 = \left(X^{r_1}(t^1), Z^{r_1}(t^1), Y^{r_2}(t^2)\right) \supseteq (X^r(t), Z^r(t), Y^r(t))$$

tomando como $\bar{t}$ la tupla $(X^{r_1}(t^1), Z^{r_1}(t^1), Y^{r_2}(t^2))$, tenemos demostrado el resultado.

ii) Aplicando la parte *i)* de la proposición 5.2.6, tomando como s la relación $\bar{r} = \tilde{\Pi}_{XZ}(r) \sqsupset\!\!\bowtie \tilde{\Pi}^{X}_{XY}(r)$, tenemos:

$$\bar{r} \sqsubseteq \tilde{\Pi}^{V}(\bar{r})$$

Si tenemos en cuenta la parte *i)* del teorema, entonces obtenemos:

$$r \preceq \bar{r}$$

y por transitividad (aplicando la proposición 5.2.5) tendremos:

$$r \preceq \tilde{\Pi}^{V}(\bar{r})$$

□

Nota. Obsérvese que la ecuación 5.19 nos dice que X^{r_1} es llave externa difusa a r_2.

Lema 5.2.9 *Sea r en las condiciones de la definición 5.2.1, y sean:*

$$r' = \tilde{\Pi}^{XZ}(r)\ ,\ \tilde{r} = \tilde{m}(r) = \tilde{\Pi}_{XZ}(r) \sqsupset\!\!\tilde{\bowtie} \tilde{\Pi}^{X}_{XY}(r)$$

Entonces se verifica que

$$\forall\, t' \in r' \quad \exists\, !\ \tilde{t} \in \tilde{r} \quad \textit{tal que } XZ^{r'}(t') = XZ^{\tilde{r}}(\tilde{t}),\ Y^{r'}(t') \subseteq Y^{\tilde{r}}(\tilde{t})$$

Demostración.

Vamos a seguir la notación de la definición 5.2.1, por lo que tomamos

$$r_1 = \tilde{\Pi}_{XZ}(r)$$

Aplicando el punto *i)* de la proposición 4.1.12 se tiene que $r'|_{XZ} = r_1$. Por lo tanto:

$$\forall\, t' \in r' \ \ \exists\, t_1 \in r_1 \ \ \text{tal que } XZ^{r'}(t') = t_1$$

y en particular:

$$X^{r'}(t') = X^{r_1}(t_1)$$

Consideremos para cada t', el conjunto de tuplas de r de las que proviene por fusión:

$$\{t_j, t_{j'} \in r \text{ tal que } XZ^r(t_j)\ \mathcal{R}\ XZ^r(t_{j'})\} = \{t_j, \in r \text{ tal que } XZ^r(t_j) \subseteq XZ^{r'}(t')\}$$

Por definición de proyección difusa tenemos:

$$X^{r'}(t') = \bigvee_{t_j} X^r(t_j)\,,\ Y^{r'}(t') = \bigvee_{t_j} Y^r(t_j)$$

dónde la unión se entiende componente a componente. Ahora bien,

$$XZ^r(t_j)\ \mathcal{R}\ XZ^r(t_{j'}) \ \Rightarrow\ X^r(t_j)\ \mathcal{R}\ X^r(t_{j'}) \quad \forall\, j, j'$$

por lo tanto, estos valores se fusionarán en r_2 en una única tupla (y pueden fundirse otros más), es decir:

$$\exists\ t_2 \in r_2 \text{ tal que } X^{r_2}(t_2) \supseteq \bigvee_{t_j} X^r(t_j) = X^{r'}(t')\ ,\ \ Y^{r_2}(t_2) \supseteq \bigvee_{t_j} Y^r(t_j) = Y^{r'}(t') \tag{5.20}$$

Pero, tal y como se estableció al principio,

$$X^{r'}(t') = X^{r_1}(t_1)\ \ ,\ \ Z^{r'}(t') = Z^{r_1}(t_1)$$

por lo que $X^{r_1}(t_1) \subseteq X^{r_2}(t_2) \ \Rightarrow$ las tuplas t_1 y t_2 se reunen. El resultado será:

$$t = (Z^{r_1}(t_1), X^{r_1}(t_1), Y^{r_2}(t_2)) \in r_1 \sqsupset\!\!\bowtie r_2$$

o lo que es lo mismo:

$$t = (Z^{r'}(t'), X^{r'}(t'), Y^{r_2}(t_2)) \in r_1 \sqsupset\!\!\bowtie r_2$$

En definitiva, uniendo lo anterior a 5.20, hemos demostrado que:

$$\forall\ t' \in r'\ \exists\ t \in r_1 \sqsupset\!\!\bowtie r_2 \text{ tal que } XZ^{r'}(t') = XZ(t),\ Y^{r'}(t') \subseteq Y(t) \tag{5.21}$$

Ahora bien, según se demostró en la parte *ii.a* del teorema 5.1.6 (ecuación 5.10), toda tupla de $r_1 \sqsupset\!\!\bowtie r_2$ verifica lo siguiente:

$$\forall\ t_i \in r_1 \sqsupset\!\!\bowtie r_2\ \ \exists\ \widetilde{t} \in \widetilde{r} \text{ tal que } XZ^{\widetilde{r}}(\widetilde{t}) = XZ(t_i)\ ,\ Y^{\widetilde{r}}(\widetilde{t}) \supseteq Y(t_i) \tag{5.22}$$

Uniendo 5.21 y 5.22 deducimos la tesis. El hecho de que $\widetilde{t}$ sea única proviene del hecho de que en $\widetilde{r}$ no hay tuplas con valores de XZ redundantes, y por lo tanto tampoco iguales. □

Estamos en condiciones de demostrar el teorema fundamental que garantiza la descomposición de una relación sin pérdidas difusas, bajo las hipótesis de la definición 5.2.1. Obviamente, también se necesita que la relación de partida satisfazga las restricciones de integridad, pero esto es algo que hemos venido asumiendo implícitamente.

Teorema 5.2.10 (fundamental de descomposición sin pérdidas)
Sea una relación r con esquema $REL = XYZ$, dónde X, Y, Z son conjuntos de atributos con $X \neq \emptyset$. Supongamos que $X \cap Y = \emptyset$, y que r satisface $X \overset{(\alpha,\beta)}{\Longrightarrow} Y$. Entonces, se satisface lo siguiente:

i) $\boxed{r \sqsubseteq \widetilde{m}\,(r)}$
o expresándolo explícitamente:

$$r \sqsubseteq \widetilde{\Pi}^{XZ} \left(\widetilde{\Pi}_{XZ}\,(r) \sqsupset\!\!\bowtie \widetilde{\Pi}^{X}_{XY}\,(r) \right)$$

ii) Además, $\widetilde{m}\,(r)$ satisface el nivel de granularidad en los mismos atributos que r.

<u>Demostración</u>.

Antes de pasar a la demostración del teorema, observemos que la hipótesis $X \cap Y = \emptyset$ no supone ninguna restricción. En el caso de que fuese $X \cap Y \neq \emptyset$, entonces aplicando la definición 4.2.5 de d.f.d.r, tendríamos:

$$X \overset{(\alpha,\beta)}{\Longrightarrow} Y \;\Leftrightarrow\; X \overset{(\alpha,\beta_{Y-X})}{\Longrightarrow} Y - X$$

Y ahora $X \cap (Y - X) = \emptyset$, por lo que ya estaríamos bajo las condiciones requeridas en el teorema.

Vamos a utilizar la notación siguiente:

Notación: $r_1 \equiv \widetilde{\Pi}_{XZ}\,(r)$, $r_2 \equiv \widetilde{\Pi}^{X}_{XY}\,(r)$, $r^* \equiv RULE^{X}_{XY}(r)$, $r' \equiv \widetilde{\Pi}^{XZ}\,(r)$, $\widetilde{r} \equiv \widetilde{m}\,(r)$

Pasamos a probar *i)* estructurando la demostración en varios pasos:

Paso 1. Construcción de una relación auxiliar r'^*.

Aplicando el lema 5.2.9 tenemos:

$$\forall\, t' \in r' \;\; \exists\,!\, \tilde{t} \in \tilde{r} \;\; \text{tal que } XZ^{r'}(t') = XZ^{\tilde{r}}(\tilde{t}) \;,\; Y^{r'}(t') \subseteq Y^{\tilde{r}}(\tilde{t}) \tag{5.23}$$

Por definición del operador *RULE* (al no existir tuplas con valores de X solapados), cada valor de $X^{\tilde{r}}$ estará incluido en una única tupla de r^*:

$$\forall \;\; \tilde{t} \in \tilde{r} \;\;\; \exists\,!\, t^* \in r^* \;\; \text{tal que } X^{\tilde{r}}(\tilde{t}) \subseteq X^{r^*}(t^*)$$

$$\text{y además es } \; Y^{\tilde{r}}(\tilde{t}) \subseteq Y^{r^*}(t^*)$$

y por lo tanto, según 5.23, también se verificará lo siguiente:

$$\forall\, t' \in r' \;\; \exists\,!\, t^* \in r^* \;\; \text{tal que } X^{r'}(t') \subseteq X^{r^*}(t^*)$$

$$\text{y además es } \; Y^{r'}(t') \subseteq Y^{r^*}(t^*) \tag{5.24}$$

Partiendo de la relación r', vamos a construir otra relación que la llamaremos r'^*, que coincidirá con los valores de XZ en r', y para los valores de Y cambiamos cada $Y^{r'}(t')$ por el correspondiente valor $Y^{r^*}(t^*)$ dado por 5.24, a saber aquella única tupla $t^* \in r^*$ tal que $XY^{r^*}(t^*) \supseteq XY^{r'}(t')$. Así pues, 5.24 se reescribiría como sigue:

$$\forall\, t'^* \in r'^* \;\; \exists\,!\, t^* \in r^* \;\; \text{tal que } X^{r'^*}(t'^*) \subseteq X^{r^*}(t^*)$$

$$\text{y además es } \; Y^{r'^*}(t'^*) = Y^{r^*}(t^*) \tag{5.25}$$

La relación r'^* así obtenida verifica las siguientes propiedades:

i) $r'^*|_{XZ} = r_1$.
Por una parte, por la propia construcción de la relación r'^* tenemos que $r'^*|_{XZ} = r'|_{XZ}$. Además, si aplicamos la parte *i)* de la la proposición 4.1.12 tendremos que

$$\left.\left(\tilde{\Pi}^{XZ}_{XYZ}\,(r)\right)\right|_{XZ} = \tilde{\Pi}_{XZ}\,(r) \;\Leftrightarrow\; r_1 = r'|_{XZ}$$

por lo que, efectivamente, concluimos que:

$$r'^*|_{XZ} = r_1$$

ii) $\boxed{r \sqsubseteq r'^{*}}$.

Esto es inmediato, ya que aplicando i) de la proposición 5.2.6 con $s = r'$, $V = XZ$, $W = XYZ$ obtenemos $r \sqsubseteq r'$. Por otra parte es obvio que $r' \sqsubseteq r'^{*}$ ya que, según 5.25, cada $Y^{r'^{*}}(t'^{*}) = Y^{r^{*}}(t^{*}) \supseteq Y^{r'}(t')$, y en XZ son iguales por el punto i) anterior. Por transitividad (proposición 5.2.5) concluimos efectivamente que $r \sqsubseteq r'^{*}$.

iii) r'^{*} satisface una dependencia funcional clásica $X \to Y$, considerando los valores difusos como elementos atómicos.

En efecto, consideremos dos tuplas arbitrarias :

$$t_1'^{*},\ t_2'^{*} \ \text{ tal que } X^{r'^{*}}(t_1'^{*}) = X^{r'^{*}}(t_2'^{*})$$

Aplicando 5.25 deducimos que:

$$\exists\, !\, t_1^{*} \ \text{ tal que } X^{r'^{*}}(t_1'^{*}) \subseteq X^{r^{*}}(t_1^{*})$$

$$\text{y además es } \ Y^{r'^{*}}(t_1'^{*}) = Y^{r^{*}}(t_1^{*})$$

$$\exists\, !\, t_2^{*} \ \text{ tal que } X^{r'^{*}}(t_2'^{*}) \subseteq X^{r^{*}}(t_2^{*})$$

$$\text{y además es } \ Y^{r'^{*}}(t_2'^{*}) = Y^{r^{*}}(t_2^{*})$$

pero si $X^{r'^{*}}(t_1'^{*}) = X^{r'^{*}}(t_2'^{*})$, entonces, debido a la unicidad de t_1^{*} y de t_2^{*} obtenemos que $t_1^{*} = t_2^{*}$, es decir:

$$t_1^{*} = t_2^{*} \ \Rightarrow \ Y^{r^{*}}(t_1^{*}) = Y^{r^{*}}(t_2^{*}) \ \Rightarrow \ Y^{r'^{*}}(t_1'^{*}) = Y^{r'^{*}}(t_2'^{*})$$

por lo que, efectivamente hemos demostrado que r'^{*} satisface una dependencia funcional clásica.

iv) Teniendo en cuenta que r'^{*} no tiene dos tuplas con valores en XZ redundantes, obviamente tampoco tendrá dos tuplas con valores en XZ iguales. Así pues, denotando por $r_1'^{*}$ a la proyección <u>clásica</u> de r'^{*} sobre XZ (considerando de nuevo los elementos difusos como átomos) y análogamente para la proyección sobre XY:

Notación: $r_1'^{*} \equiv \Pi_{XZ}(r'^{*})\ ,\ r_2'^{*} \equiv \Pi_{XY}(r'^{*})$

se tendrá que $r_1'^* = r'^*|_{XZ}$. Aplicando i) tenemos finalmente que:

$$\boxed{r_1'^* = r_1} \tag{5.26}$$

Paso 2. Vamos a ver ahora la relación que hay entre las relaciones $r_1'^* \bowtie r_2'^*$ y $r_1 \,\tilde{\sqsupset\!\bowtie}\, r_2$ ($\bowtie$ es el operador de reunión clásico)

Sea una tupla arbitraria $v \in r_1'^*$ o equivalentemente (según 5.26) $v \in r_1$ (utilizamos por simplicidad la misma notación v para ambas tuplas $v \in r_1$, $v \in r_1'^*$), es decir:

$$XZ^{r_1}(v) = XZ^{r_1'^*}(v)$$

Paso 2.1. Veamos qué forma tiene la relación $m(r'^*) = r_1'^* \bowtie r_2'^*$

Consideremos el conjunto de tuplas de $r_2'^*$ con las que se reune $v \in r_1'^*$ (en sentido clásico). Este conjunto es no vacío y además sólo tendrá una única tupla, que la llamaremos u^v, ya que según iii) la relación r'^* satisface una dependencia funcional clásica $X \to Y$ y por lo tanto en la proyección sobre XY sólo aparece una vez cada valor de X. Esta tupla u^v tendrá la forma siguiente:

$$u^v = (X^{r_2'^*}(u^v) = X^{r_1'^*}(v)\,,\, Y^{r_2'^*}(u^v) \equiv y^v)$$

dónde hemos llamado y^v al valor de Y que tiene en la relación r'^* cualquier tupla con valor de X igual a $X^{r_1'^*}(v)$. Recordemos que, por construcción de r'^*, y^v es igual a aquel valor $Y^{r^*}(v^*)$ en la relación r^* que verificaba 5.24. ($X^{r^*}(v^*) \supseteq X^{r_1}(v)$)
El resultado de la reunión de v con $r_2'^*$ será $v \bowtie u^v = (Z^{r_1'^*}(v) = Z^{r_1}(v)\,,\, X^{r_1'^*}(v) = X^{r_1}(v)\,,\, y^v)$, y por tanto, el resultado de la reunión de $r_1'^*$ con $r_2'^*$ será:

$$m(r'^*) = r_1'^* \bowtie r_2'^* = \left(XZ^{r_1'^*}(v), Y^{r^*}(v^*)\right)_{v\in r_1'^*} \equiv (XZ^{r_1}(v), y^v)_{v\in r_1} \tag{5.27}$$

Paso 2.2. Veamos ahora cómo es la relación $r_1 \,\tilde{\sqsupset\!\bowtie}\, r_2$.

Consideremos el conjunto de tuplas de r_2 con las que se reune $v \in r_1$ (aplicando ahora la definición 5.1.2 de reunión dirigida):

$$\{s_k \in r_2 \text{ tal que } X^{r_1}(v) \subseteq X^{r_2}(s_k)\}_k$$

Como resultado de realizar $v \sqsupset\!\bowtie r_2$ se tendrán las siguientes tuplas:

$$(XZ^{r_1}(v)\,,\, Y^{r_2}(s_k))_k$$

Todas estas tuplas son redundantes respecto XZ, por lo que al tomar la XZ-proyección difusa $\widetilde{\Pi}^{XZ}(r_1 \sqsupset\!\bowtie r_2)$, quedará en una única tupla de la forma:

$$(XZ^{r_1}(v)\,,\,y'^v) \tag{5.28}$$

dónde

$$y'^v = \bigvee_{s_k} Y^{r_2}(s_k)$$

Además, como no existen dos tuplas v_1, v_2 con $XZ^{r_1}(v_1)\ \mathcal{R}\ XZ^{r_1}(v_2)$ se tendrá que al hacer $\widetilde{\Pi}^{XZ}(r_1 \sqsupset\!\bowtie r_2)$, es decir, al hacer $r_1 \sqsupset\!\widetilde{\bowtie}\, r_2$, no se fundirá la tupla dada por 5.28 con ninguna otra tupla $\{XZ^{r_1}(v')\ ,\ Y^{r_2}(s'_k)\}$

En definitiva, hemos demostrado que:

$$\widetilde{m}\,(r) = r_1 \sqsupset\!\widetilde{\bowtie}\, r_2 = (XZ^{r_1}(v)\,,\,y'^v)_{v \in r_1} \tag{5.29}$$

Paso 2.3

Aplicando 5.27 y 5.29 obtenemos que $m(r'^*)|_{XZ} = \widetilde{m}\,(r)|_{XZ}$, o si se prefiere, utilizando la siguiente notación

$$\textbf{Notación: } \bar{r'^*} \equiv m(r'^*) = r_1'^* \bowtie r_2'^*$$

se reescribiría como:

$$\bar{r'^*}|_{XZ} = \widetilde{r}\,|_{XZ} \tag{5.30}$$

Ahora bien, por otra parte, en el apartado *iii)* del paso 1 de la demostración, hemos visto que r'^* satisface la dependencia funcional clásica $X \rightarrow Y$; por lo tanto, aplicando el teorema de descomposiciones sin pérdidas de Heath del caso clásico (teorema 1.2.14, página 40) tendremos:

$$r'^* = \bar{r'^*}$$

es decir, al construir $\bar{r'^*} = m(r'^*)$, obtenemos las mismas tuplas, con los mismos elementos o átomos, que había en r'^*. En particular, obtenemos que:

$$r'^*|_{XZ} = \bar{r'^*}|_{XZ} \tag{5.31}$$

Paso 3. Vamos a demostrar finalmente que $r \sqsubseteq mpt(r)$, o lo que es lo mismo, $r \sqsubseteq \widetilde{r}$.

Según se demostró en el lema 5.2.8, tenemos que $r \preceq \widetilde{r}$, por lo que falta ver ahora (para obtener $r \sqsubseteq \widetilde{r}$) que toda tupla de $\widetilde{r}$ contenga a alguna tupla de r. Sea una tupla arbitraria $\widetilde{t} \in \widetilde{r}$. Aplicando 5.30 tendremos:

$$\forall \ \widetilde{t} \in \widetilde{r} \quad \exists \ t^{\prime *} \in r^{\prime *} \ \text{ tal que } XZ^{\widetilde{r}}(\widetilde{t}) = XZ^{r^{\prime *}}(t^{\prime *})$$

De lo que se deriva, aplicando 5.31 que:

$$\forall \ \widetilde{t} \in \widetilde{r} \quad \exists \ t'^{*} \in r'^{*} \ \text{ tal que } XZ^{\widetilde{r}}(\widetilde{t}) = XZ^{r'^{*}}(t'^{*}) \tag{5.32}$$

Aplicando el hecho de que $r \sqsubseteq r'^{*}$ (paso *1.ii)* de la demostración), tendremos que t'^{*} contendrá a alguna tupla de r. Sea pues el conjunto de dichas tuplas:

$$\{t_i \in r \ \text{ tal que } t_i \subseteq t'^{*}\} \neq \emptyset \ \Rightarrow$$

$$\Rightarrow \ \left\{t_i \in r \ \text{ tal que } XZ^{r}(t_i) \subseteq XZ^{r'^{*}}(t'^{*})\right\} \neq \emptyset$$

Uniendo lo anterior a 5.32, resulta:

$$\forall \ \widetilde{t} \in \widetilde{r} \quad \exists \ \{t_i\} \in r \ \text{ tal que } \bigvee_i XZ^{r}(t_i) = XZ^{\widetilde{r}}(\widetilde{t}) \tag{5.33}$$

Tenemos que demostrar que el valor de Y para dichas tuplas $t_i \in r$, también está contenido en el valor de Y para la tupla $\widetilde{t} \in \widetilde{r}$.

Veamos qué pasa con el valor correspondiente de $Y^{\widetilde{r}}(\widetilde{t})$: aplicando 5.32 a dicha tupla $\widetilde{t} \in \widetilde{r}$, tendremos lo siguiente:

$$\forall \ \widetilde{t} \in \widetilde{r} \quad \exists \ t'^{*} \in r'^{*} \ \text{ tal que } X^{\widetilde{r}}(\widetilde{t}) = X^{r'^{*}}(t'^{*})$$

Aplicando que $r'^{*}|_{XZ} = r_1$ (parte *i)* de la demostración) obtenemos

$$\forall \ t'^{*} \in r'^{*} \ \ \exists \ t^1 \in r_1 \ \text{ tal que } X^{r'^{*}}(t'^{*}) = X^{r_1}(t^1)$$

Aplicando ahora la definición de reunión difusa dirigida (ecuación 5.19), tendremos que:

$$\forall \ t^1 \in r_1 \ \exists \ t^2 \in r_2 \ \text{ tal que } X^{r_1}(t^1) \subseteq X^{r_2}(t^2)$$

por lo que podemos concluir lo siguiente:

$$\forall \ \widetilde{t} \in \widetilde{r} \quad \exists \ t^1 \in r_1 \ \exists \ t^2 \in r_2 \ \text{ tal que } X^{\widetilde{r}}(\widetilde{t}) = X^{r_1}(t^1) \subseteq X^{r_2}(t^2) \tag{5.34}$$

Ahora bien, por definición de reunión difusa, la tupla t_1 de r_1 se reune con la tupla t_2 de r_2 y posiblemente con más tuplas. Al tomar la XZ-proyección difusa de dichas tuplas, obtendremos el valor $Y^{\tilde{r}}(\tilde{t})$ que obviamente verificará que:

$$Y^{\tilde{r}}(\tilde{t}) \supseteq Y^{r_2}(t^2) \tag{5.35}$$

Por otra parte, la ecuación 5.33 nos decía que $X^r(t_i) \subseteq X^{\tilde{r}}(\tilde{t})$, por lo que aplicando 5.34 tendremos que $X^r(t_i) \subseteq X^{r_2}(t^2)$. Ahora bien, r_2 es la X-proyección difusa de r, por lo que:

$$\begin{aligned} X^r(t_i) \subseteq X^{r_2}(t^2) \;&\Rightarrow\; Y^r(t_i) \subseteq Y^{r_2}(t^2) \\ &\Rightarrow\; \bigvee_i Y^r(t_i) \subseteq Y^{r_2}(t^2) \end{aligned}$$

Encadenando 5.35 y lo anterior, deducimos:

$$\bigvee_i Y^r(t_i) \subseteq Y^{\tilde{r}}(\tilde{t})$$

Si unimos lo anterior a 5.33, obtenemos:

$$\forall\ \tilde{t} \in \tilde{r}\quad \exists\ \{t_i\} \in r \ \text{ tal que } \bigvee_i XZ^r(t_i) = XZ^{\tilde{r}}(\tilde{t})\ ,\ \bigvee_i Y^r(t_i) \subseteq Y^{\tilde{r}}(\tilde{t}) \;\Rightarrow$$

$$\Rightarrow\quad \forall\ \tilde{t} \in \tilde{r}\quad \exists\ \{t_i\} \in r \ \text{ tal que } XZ^r(t_i) = XZ^{\tilde{r}}(\tilde{t})\ ,\ Y^r(t_i) \subseteq Y^{\tilde{r}}(\tilde{t})$$

por lo que, efectivamente, cada tupla de $\tilde{r}$ contiene a alguna tupla de r, lo que uniéndolo al lema 5.2.8 se obtiene:

$$r \sqsubseteq \tilde{m}\ (r)$$

como queríamos demostrar.

Paso 4. Nos falta ver que la relación $\tilde{m}\ (r)$ satisface el nivel de granularidad en todos los atributos, pero esto ya lo vimos en el teorema 5.2.2.

□

Nota 1. También se verifica el recíproco de este teorema, pero no es de interés práctico. La demostración puede realizarse siguiendo exactamente los mismos pasos que la que aparece en el caso clásico en el artículo de Rissanen ([70]), imponiendo como hipótesis la existencia de dos valores al menos, en cada dominio, que no pertenezcan a la misma clase. En definitiva, que la parte condicional de la hipótesis de dependencia:

$$\text{Si } X(t_1) \text{ es semejante a } X(t_2)$$

se viole para al menos dos tuplas.

Nota 2. Aplicando la parte *ii)* de la proposición 5.2.6 tendremos que:

$$r \sqsubseteq \tilde{\Pi}^{V} \left(\tilde{\Pi}_{XZ}(r) \sqsupset\!\!\bowtie \tilde{\Pi}^{X}_{XY}(r) \right) \quad \forall\, V \subseteq XZ$$

pero no lo incluimos dentro del teorema ya que no es un resultado de aplicación práctica.

Nota 3. De la demostración del teorema se deduce también que $r \preceq r_1 \sqsupset\!\!\bowtie r_2$ (inmediato) y además que:

$$\forall\, s \in r_1 \sqsupset\!\!\bowtie r_2 \;\; \exists\, t \in r \;\; \text{tal que } XZ(t) = XZ(s)\;,\; Y(t) \simeq^{f}_{\beta} Y(s)$$

Este podría considerarse también como definición del concepto de reunión dirigida (no difusa) sin pérdidas difusas. Obtendríamos las mismas tuplas que originalmente, excepto en que para los atributos consecuentes se tendría semejanza fuerte en vez de inclusión.

Nota 4. Como ya hemos comentado en ocasiones anteriores, hemos demostrado el teorema fundamental bajo las condiciones lo más generales posibles. En concreto, no hemos impuesto que las etiquetas de los antecedentes o consecuentes, no puedan solaparse en los núcleos, restricción que, por otra parte, hubiese sido lógica.

Pasemos ahora a ver la expresión del teorema fundamental en términos del operador $RULE$. En primer lugar, vamos a ver un lema que nos dice que podemos limitarnos a trabajar con $\tilde{m}\,(r)$, ya que siempre que sea $r \sqsubseteq \tilde{m}\,(r)$ también será $r \sqsubseteq \tilde{m}^{*}\,(r)$. Recordemos que habíamos utilizado la notación $\tilde{m}^{*}\,(r)$ para designar a la relación $\tilde{\Pi}^{XZ} \left(\tilde{\Pi}_{XZ}(r) \sqsupset\!\!\bowtie_X RULE^{X}_{XY}(r) \right)$.

Lema 5.2.11 *Sea una relación r con esquema $REL = XYZ$, dónde X, Y, Z son conjuntos de atributos con $X \neq \emptyset$. Entonces, se verifica que $\tilde{m}\,(r) \sqsubseteq \tilde{m}^{*}\,(r)$.*

<u>Demostración</u>.

Seguimos la misma notación que introdujimos en el teorema fundamental (teorema 5.2.10), es decir:

$$r_1 \equiv \tilde{\Pi}_{XZ}(r) \;\;,\;\; r_2 \equiv \tilde{\Pi}^{X}_{XY}(r) \;\;,\;\; r^{*} \equiv RULE^{X}_{XY}(r)$$

por lo que la condición $r \sqsubseteq \tilde{m}^* (r)$ que tenemos que demostrar, se reescribe en la forma siguiente:

$$r \sqsubseteq \tilde{\Pi}^{XZ} (r_1 \sqsupset\!\bowtie r^*)$$

Para ello, vamos a ver la conexión que hay entre la relación $r_1 \sqsupset\!\bowtie r^*$ y la relación $r_1 \sqsupset\!\bowtie r_2$. Aplicando la definición de reunión dirigida, tenemos:

$$\forall\ u \in r_1 \sqsupset\!\bowtie r_2\ \ \exists\ v \in r_1\ \ y\ \ \ \exists\ v_2 \in r_2\ \text{ tal que } u = (XZ^{r_1}(v), Y^{r_2}(v_2))$$

dónde $X^{r_1}(v) \subseteq X^{r_2}(v_2)$. Aplicando ahora la parte *iii)* de la proposición 5.2.6, tenemos que:

$$r_2 \sqsubseteq r^*$$

Teniendo en cuenta lo anterior, deducimos que:

$$\forall\ v_2 \in r_2\ \ \exists\,!\ v_2^* \in r^*\ \text{ tal que } XY^{r_2}(v_2) \subseteq XY^{r^*}(v_2^*) \qquad (5.36)$$

dónde la unicidad de v_2^* proviene por la propia definición del operador $RULE$ ya que no hay dos tuplas con valores de X^{r^*} con intersección no vacía.

Ahora bien:

$$X^{r_1}(v) \subseteq X^{r_2}(v_2)\ \Rightarrow\ X^{r_1}(v) \subseteq X^{r^*}(v_2^*)$$

por lo que las tuplas $v \in r_1$ y $v_2^* \in r^*$ se reunen, obteniéndose la tupla:

$$u' = (XZ^{r_1}(v)\ ,\ Y^{r^*}(v_2^*)) \in r_1 \sqsupset\!\bowtie r^*$$

Hemos demostrado pues que:

$$\forall\ u \in r_1 \sqsupset\!\bowtie r_2\ \ \exists\ u' \in r_1 \sqsupset\!\bowtie r^*\ \text{ tal que } XZ(u) = XZ(u')\ ,\ Y(u) \subseteq Y(u') \quad (5.37)$$

Por contra, consideremos ahora una tupla arbitraria $u' \in r_1 \sqsupset\!\bowtie r^*$. Aplicando de nuevo la definición de reunión, tenemos:

$$u' = (XZ^{r_1}(v)\ ,\ Y^{r^*}(v^*)) \qquad (5.38)$$

dónde $X^{r_1}(v) \subseteq X^{r^*}(v^*)$ para cierta tupla $v^* \in r^*$.

Aplicando la ecuación 5.19 que nos dice (ver nota al final de dicho teorema) que X^{r_1} es llave externa difusa a r_2, tenemos:

$$\forall\ v \in r_1\ \ \exists\ v_2 \in r_2\ \ \text{tal que } X^{r_1}(v) \subseteq X^{r_2}(v_2)\ \Rightarrow$$

$$\Rightarrow\quad \exists\ u = (XZ^{r_1}(v)\ ,\ Y^{r_2}(v_2)) \in r_1 \sqsupset\!\bowtie r_2 \tag{5.39}$$

Aplicando 5.36 obtenemos:

$$XY^{r_2}(v_2) \subseteq XY^{r^*}(v_2^*)$$

Así pues, v_2^* es una tupla que verifica $X^{r^*}(v_2^*) \supseteq X^{r_1}(v)$, pero por la propia definición del operador $RULE$, sólo hay una tupla v_2^* verificando lo anterior, por lo que podemos concluir que la tupla v^* de 5.38 es la misma, es decir, $v^* = v_2^*$. Así pues,

$$Y^{r_2}(v_2) \subseteq Y^{r_2}(v_2^*) = Y^{r^*}(v^*) \tag{5.40}$$

Uniendo 5.38, 5.39 y 5.40 resulta:

$$\forall\ u' \in r_1 \sqsupset\!\bowtie r^*\ \ \exists\ u \in r_1 \sqsupset\!\bowtie r_2\ \ \text{tal que } XZ(u) = XZ(u')\ ,\ Y(u) \subseteq Y(u') \tag{5.41}$$

Uniendo 5.37 y 5.41 resulta que:

$$(r_1 \sqsupset\!\bowtie r_2)|_{XZ} = (r_1 \sqsupset\!\bowtie r^*)|_{XZ}$$

$$\forall\ u \in r_1 \sqsupset\!\bowtie r_2,\quad \forall\ u' \in r_1 \sqsupset\!\bowtie r^*\ \text{con } u|_{XZ} = u'|_{XZ}$$
$$\text{se verifica que } Y(u') \supseteq Y(u) \tag{5.42}$$

Una vez obtenida la conexión que hay entre las relaciones $r_1 \sqsupset\!\bowtie r_2$ y $r_1 \sqsupset\!\bowtie r^*$, va a ser inmediato deducir la tesis pedida. En primer lugar, tenemos la siguiente implicación:

$$(r_1 \sqsupset\!\bowtie r_2)|_{XZ} = (r_1 \sqsupset\!\bowtie r^*)|_{XZ}\ \Rightarrow \widetilde{\Pi}_{XZ}\,(r_1 \sqsupset\!\bowtie r_2) = \widetilde{\Pi}_{XZ}\,(r_1 \sqsupset\!\bowtie r^*) \tag{5.43}$$

Por otra parte, podemos aplicar la parte *i)* de la proposición 4.1.12 y obtenemos que:

$$\overset{\sim XZ}{\Pi}_{XZ}\left(\overset{\sim XZ}{\Pi}_{REL(s)}(s)\right) = \overset{\sim XZ}{\Pi}_{REL(s)}(s)\Big|_{XZ}$$

o lo que es lo mismo, según la notación que estamos siguiendo para la proyección difusa:

$$\tilde{\Pi}_{XZ}\left(\tilde{\Pi}^{XZ}(s)\right) = \tilde{\Pi}^{XZ}(s)\Big|_{XZ}$$

Aplicando lo anterior, tomando como s las relaciones $r_1 \sqsupset\!\!\bowtie r_2$ y $r_1 \sqsupset\!\!\bowtie r^*$, y teniendo en cuenta 5.43, deducimos que:

$$\tilde{\Pi}^{XZ}(r_1 \sqsupset\!\!\bowtie r_2)\Big|_{XZ} = \tilde{\Pi}^{XZ}(r_1 \sqsupset\!\!\bowtie r^*)\Big|_{XZ}$$

o lo que es lo mismo:

$$\tilde{m}(r)|_{XZ} = \tilde{m}^*(r)|_{XZ}$$

Por lo tanto, en la XZ-proyección difusa, se funden las mismas tuplas en $\tilde{m}(r)$ que las de $\tilde{m}^*(r)$. Así pues, si ahora aplicamos 5.42 tendremos que:

$$\forall\ \tilde{u} \in \tilde{m}(r), \quad \forall\ \tilde{u}' \in \tilde{m}^*(r) \text{ con } \tilde{u}\,|_{XZ} = \tilde{u}'\,|_{XZ}$$

$$\text{se verifica que } Y(\tilde{u}') \supseteq Y(\tilde{u})$$

En definitiva, hemos demostrado que $\tilde{m}(r) \sqsubseteq \tilde{m}^*(r)$. □

Estamos ya en condiciones de dar la varsión del teorema fundamental de descomposiciones sin pérdidas utilizando el operador $RULE$:

Teorema 5.2.12 *Sea una relación r en las condiciones del teorema 5.2.10, es decir, con esquema $REL = XYZ$, dónde X, Y, Z son conjuntos de atributos con $X \neq \emptyset$, $X \cap Y = \emptyset$, y $X \overset{(\alpha,\beta)}{\Longrightarrow} Y$. Entonces, se verifica lo siguiente:*

i) $r \sqsubseteq \tilde{m}^*(r)$

ii) $\tilde{m}^*(r)$ *satisface el nivel de granularidad en los mismos atributos de r.*

Demostración.

Veamos la demostración de *i)*. Aplicando el teorema 5.2.10, obtenemos que $r \sqsubseteq \tilde{m}(r)$. Si aplicamos ahora el lema 5.2.11 deducimos por transitividad que $r \sqsubseteq \tilde{m}^*(r)$. La parte *ii)* nos la da el teorema 5.2.2. □

Ejemplo 5.2.13 . Supongamos una relación r en la que existe una d.f.d.r $B \overset{(\beta_B,\beta_C)}{\Longrightarrow} C$:

$$r = \begin{array}{c|ccc|} & A & B & C \\ \hline t_1 & a_1 & b_1 & c \\ t_2 & \widetilde{a} & b_2 & \widetilde{c} \\ t_3 & a_3 & \widetilde{b} & c_3 \\ t_4 & a_4 & b_4 & c_4 \\ t_5 & a_5 & b_5 & c_5 \\ \hline \end{array}$$

dónde los datos son los representados en la figura 5.4.

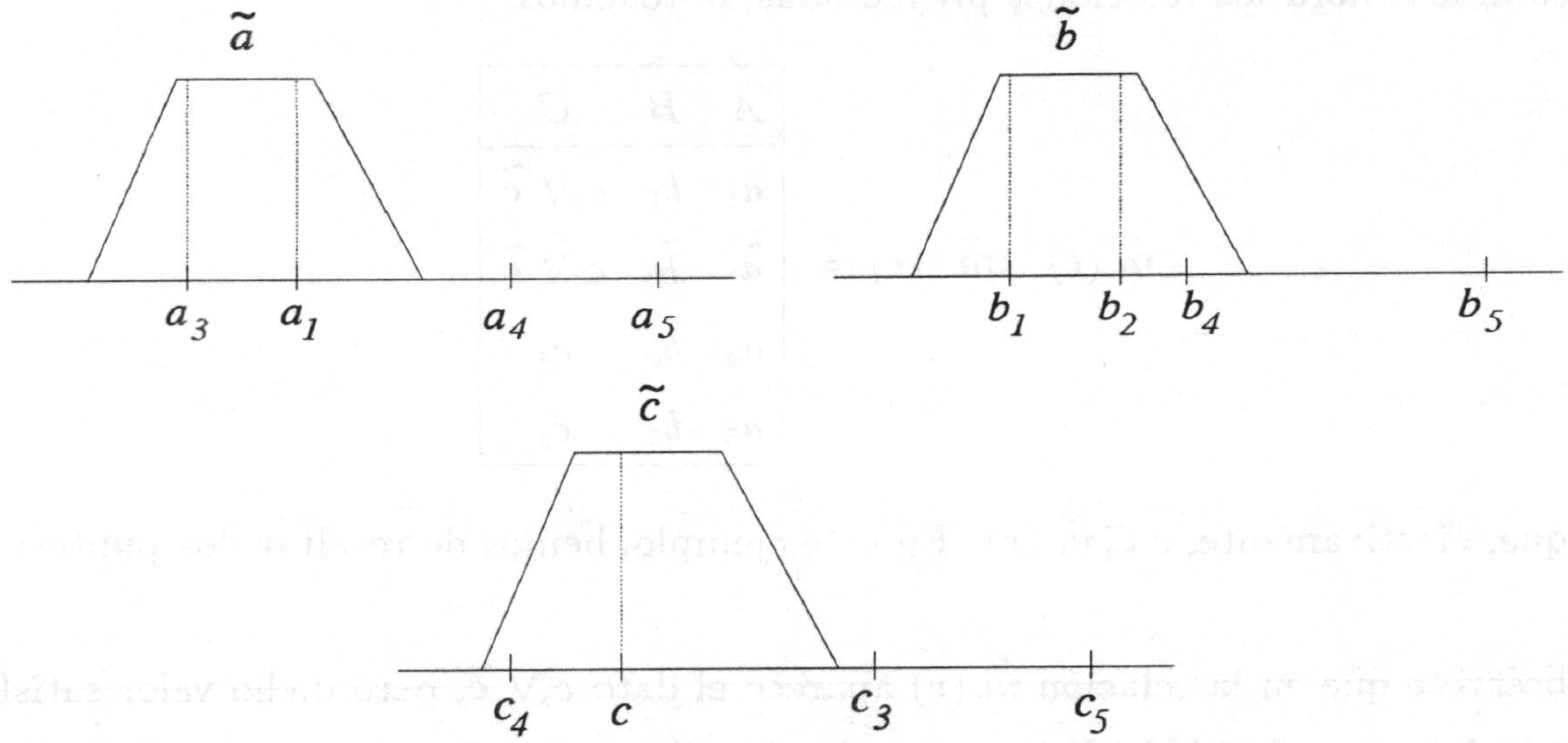

Figura 5.4. Ejemplo de Aplicación del Teorema Fundamental de Descomposiciones sin Pérdidas

$$[t_1]^B = \{t_1, t_2, t_3\} = [t_2]^B = [t_3]^B$$

$$[t_4]^B = t_4 \ , \ [t_5]^B = t_5$$

Supongamos que $B(t_4) \simeq^d_{\beta_B} B(t_3)$, es decir, que $b_4 \simeq^d_{\beta_B} \widetilde{b}$. Entonces, para no violar la d.f.d.r, debe ser $c_4 \simeq^f_{\beta_C} c \vee \widetilde{c} \vee c_3$. Sobre b_5 supongamos que no es semejante débil a ningún valor. Descompongamos la relación r:

$$\widetilde{\Pi}_{AB}(r) = \begin{array}{|cc|} \hline A & B \\ \hline a_1 & b_1 \\ \tilde{a} & \tilde{b} \\ a_4 & b_4 \\ a_5 & b_5 \\ \hline \end{array} \qquad \widetilde{\Pi}^{B}_{BC}(r) = RULE^{B}_{BC}(r) = \begin{array}{|cc|} \hline B & C \\ \hline \tilde{b} & c_3 \vee \tilde{c} \\ b_4 & c_4 \\ b_5 & c_5 \\ \hline \end{array}$$

Obsérvese que $c_4 \simeq^{f}_{\beta_C} c \vee \tilde{c} \vee c_3 \Rightarrow c_4 \simeq^{f}_{\beta_C} c_3 \vee \tilde{c}$, por lo que, tal y como nos decía el teorema 5.1.6, la relación $\widetilde{\Pi}^{B}_{BC}(r)$ satisface la d.f.d.r $B \overset{(\beta_B,\beta_C)}{\Longrightarrow} C$.

Si reunimos ahora las relaciones proyectadas, obtenemos:

$$\tilde{m}(r) = \tilde{m}^{*}(r) = \begin{array}{|ccc|} \hline A & B & C \\ \hline a_1 & b_1 & c_3 \vee \tilde{c} \\ \tilde{a} & \tilde{b} & c_3 \vee \tilde{c} \\ a_4 & b_4 & c_4 \\ a_5 & b_5 & c_5 \\ \hline \end{array}$$

por lo que, efectivamente, $r \sqsubseteq \tilde{m}(r)$. En este ejemplo, hemos de resaltar dos puntos:

1. Obsérvese que en la relación $\tilde{m}(r)$ aparece el dato $c_3 \vee \tilde{c}$, pero dicho valor satisface el nivel de granularidad (β) impuesto en el atributo C de la relación r. En definitiva, en el proceso de reconstrucción difusa de la relación original, no se pierde cuantitativamente más de lo que se había asumido al establecer la semejanza entre dos valores (β).

2. En el caso de que A hubiese sido llave difusa de la relación r, entonces, las tuplas t_1 y t_2 no se habrían fundido en la relación $\widetilde{\Pi}_{AB}(r)$, por lo que $r|_{AB} = \tilde{m}(r)|_{AB}$, y por lo tanto estaríamos recuperando exactamente la información original presente en AB, mientras que los consecuentes de r estarían incluidos en los de $\tilde{m}(r)$. Este hecho lo demostraremos formalmente en la próxima sección.

■

◇ *Esquema General del Proceso de Descomposición*

Pasamos a esquematizar brevemente en la figura 5.5 el proceso de descomposición y reunión que hemos detallado a lo largo de esta sección.

$$\boxed{¿\ r \text{ satisface } X \overset{(\alpha,\beta)}{\Longrightarrow} Y\ ?} \xrightarrow{SI}$$

$$\xrightarrow{SI} \boxed{\text{Descomponer } r \text{ en } \widetilde{\Pi}_{XZ}(r) \text{ y en } \begin{cases} \widetilde{r} = \widetilde{\Pi}^{X}_{XY}(r) \text{ o bien} \\ r^* = RULE^{X}_{XY}(r) \end{cases}}$$

$\widetilde{r}$ verifica lo siguiente:

1) X es llave difusa de $\widetilde{r}$ (teorema 4.3.2)

2) $\widetilde{r}$ satisface $X \overset{(\alpha,\beta)}{\Longrightarrow} Y$ (teorema 4.2.11)

r^* verifica lo siguiente:

1) No hay datos de X solapados (proposición 4.2.2)

2) r^* satisface $X \overset{(\alpha,\beta)}{\longrightarrow} Y$ (teorema 4.2.11)

Recuperación de los datos originales:

El teorema 5.2.10 garantiza que $r \sqsubseteq \widetilde{\Pi}_{XZ}(r) \,\widetilde{\sqsupset\!\bowtie}\, \widetilde{r}$

El teorema 5.2.12 garantiza que $r \sqsubseteq \widetilde{\Pi}_{XZ}(r) \,\widetilde{\sqsupset\!\bowtie}\, r^*$

Además, ambas relaciones satisfacen el nivel de granularidad

Figura 5.5. Esquema General de Descomposición

5.3 Proceso de Normalización

Ahora procedemos a estudiar consecuencias inmediatas de la aplicación del teorema fundamental de descomposición difusa. Ello nos llevará a plantear cuales son las formas normales que deberan imponerse en una base de datos relacional difusa. Hemos visto en la sección anterior, que el operador de reconstrucción difusa es, en general, una extensión de la relación original. Esto nos dice que mediante el mismo, podemos realmente recuperar la información que teníamos. Ahora vamos a ver que, dependiendo de la forma normal en la que estuviese la relación original, podremos recuperar incluso más información de la que proporciona la aplicación de los teoremas generales (teoremas 5.2.10 y 5.2.12, páginas 250 y 5.2.12 respectivamente). Para ello, necesitamos introducir previamente el concepto de forma normal difusa.

5.3.1 Formas Normales Difusas

En el apartado 1.2.3 del primer capítulo, se puso de manifiesto la importancia de conseguir mediante el diseño, relaciones que verifiquen ciertas restricciones en cuanto a:

1. Los datos que pudiesen aparecer en una tupla.
2. Las dependencias funcionales que puedan aparecer.

En cuanto al primer punto, se establecía una primera forma normal en la que se imponía que los datos no pudiesen ser conjuntos, sino únicamente valores atómicos (alfanuméricos o numéricos). En la correspondiente versión difusa, sustituiremos los tipos alfanuméricos o numéricos, por una distribución de posibilidad en general, definiendo así la *primera forma normal difusa.* A continuación, extenderemos al caso difuso el segundo punto, introduciendo las demás formas normales difusas.

◇ *Primera Forma Normal Difusa*

Los tipos de datos que hemos permitido representar han sido, en general, una distribución de posibilidad, y como casos particulares los tipos crisp. Así pues, definimos la primera forma normal como sigue:

Definición 5.3.1 *Una relación r se dice que verifica la* **primera forma normal difusa** *(*1FND*) si y solo si:*

$$\forall\, t \in r \;\; \forall\, A \in REL(r),\;\; A(t) \textit{ es una distribución de posibilidad (o un caso degenerado)}$$

Restricción. *Toda relación en un sistema relacional difuso ha de estar en 1FND.*

Nota. Obviamente toda relación en primera forma normal está también en primera forma normal difusa.

Por otra parte, nos preguntamos: ¿ha de imponerse alguna restricción sobre el nivel de granularidad (definición 3.2.16) permitido en los datos que sean difusos? Formalmente, ha sido una restricción necesaria a imponer en los atributos antecedentes y consecuentes de una relación, para poder demostrar los principales resultados de esta memoria. Ahora bien, nosotros pensamos que también debería imponerse para cualquier atributo de una relación, como una restricción de integridad de identidad de los datos. De esta forma, no se permitiría aceptar cualquier imprecisión en los *datos*, sino que éstos deberían cumplir una restricción de granularidad. Por lo tanto, ante un nuevo valor que viole dicha restricción, debería almacenarse el valor `desconocido` como representación de dicha información. En cualquier caso, hemos de enfatizar que en nuestro desarrollo, no afecta en nada aceptar datos violando el nivel de granularidad (para atributos que no están presentes en una dependencia).

Para definir el resto de las formas normales, nos apoyamos en el concepto de dependencia funcional:

◇ *Segunda, Tercera y Forma Normal Difusa de Boyce Codd*

Recordemos brevemente el proceso de diseño básico descrito para el caso clásico en el apartado 1.2.3 del primer capítulo. Este puede aplicarse en dos situaciones diferentes:

1. No se tiene almacenado ningún dato. El experto debe elicitar las restricciones dadas en formas de dependencias entre los distintos atributos. En base a esto, se construyen esquemas relacionales que verifiquen cierta forma normal.

2. La base de datos existe físicamente, pero el esquema no se diseñó adecuadamente, por lo que hay presentes dependencias no deseables, ya que producen problemas de redundancia y actualización en general. En este caso se procede a un proceso de descomposición hasta obtener relaciones en ciertas formas normales.

En el caso clásico, se recomienda llegar en la descomposición hasta un esquema en el que no estén presentes dependencias no triviales u otras distintas a las correspondientes a las llaves candidatas. Obligatoriamente debería llegarse hasta la tercera forma normal, y a veces era deseable hasta FNBC.

En la extensión difusa, realizaremos un estudio análogo pero cambiando la definición de d.f por d.f.d.r. por lo que las formas normales clásicas serán casos particulares de las difusas. Veremos que si nos quedamos en la tercera forma normal, entonces podemos obtener mucha más precisión que la dada por el teorema fundamental de descomposiciones sin pérdidas. Es por ello, que se aconsejará no llegar en el diseño hasta la forma normal (difusa) de Boyce Codd. Esto es, por supuesto, sólo un consejo y se puede llegar más allá, descomponiendo una relación que no esté en FNBC, y siempre estaremos bajo las garantías de pérdidas difusas dadas por el teorema fundamental 5.2.10.

Pasamos a introducir las distintas formas normales, enfatizando el hecho de que únicamente se diferencian de sus análogas versiones del modelo clásico, en la utilización de d.f.d.r en vez de d.f.

Definición 5.3.2 *Un esquema relacional REL se dice que está en* **segunda forma normal difusa** *(*2FND*) si siempre que REL satisfazga una d.f.d.r $X \stackrel{(\alpha,\beta)}{\Longrightarrow} A$ con $A \notin X$,* *entonces o bien A es primo, o bien X no está incluido estrictamente en ninguna llave difusa.*

La segunda forma normal difusa, al igual que la clásica, carece de interés ya que queda englobada por la tercera forma que pasamos a describir:

Definición 5.3.3 *Un esquema relacional REL se dice que está en* **tercera forma normal difusa** *(*3FND*) si siempre que REL satisfazga una d.f.d.r $X \stackrel{(\alpha,\beta)}{\Longrightarrow} A$ con $A \notin X$, entonces o bien A es primo, o bien X contiene a alguna llave difusa.*

Definición 5.3.4 *Un esquema relacional REL se dice que está en* **forma normal difusa de Boyce Codd** *(*FNDBC*) si siempre que REL satisfazga una d.f.d.r $X \stackrel{(\alpha,\beta)}{\Longrightarrow} A$ con $A \notin X$, entonces X contiene a alguna llave difusa.*

Al igual que en el caso clásico, los conceptos de 2FND, 3FND y FNDBC van siendo más restrictivos, por lo que, por ejemplo, toda relación que esté en FNDBC está también en 3FND y 2FND.

5.3.2 Casos Particulares del Teorema Fundamental de Descomposición: Formas Normales Aconsejables

Vamos a ver qué consecuencias pueden obtenerse si se trabaja con la información de cual es la llave difusa de la relación original r que se va a descomponer, y la conexión con las distintas formas normales difusas. A dicha llave la llamaremos a partir de ahora K. Para ello, veamos el siguiente teorema, que se enuncia bajo las mismas condiciones que los teoremas 5.2.10 y 5.2.12, y añade hipótesis sobre la llave difusa considerada:

Teorema 5.3.5 (caso particular del fundamental de descomposición) . *Sea una relación r con esquema $REL = XYZ$, dónde X, Y, Z son conjuntos de atributos con $X \neq \emptyset$. Supongamos que $X \cap Y = \emptyset$, $X \overset{(\alpha,\beta)}{\Longrightarrow} Y$ y que K es llave difusa de r verificando $K \subseteq XZ$. Entonces se verifica lo siguiente:*

i) $r \sqsubseteq \widetilde{m}\ (r)$, $r \sqsubseteq \widetilde{m}^*\ (r)$. *Además, las relaciones anteriores verifican el nivel de granularidad en los mismos atributos en los que se satisface en r.*

ii) K *es llave difusa de* $\widetilde{m}\ (r)$ *y de* $\widetilde{m}^*\ (r)$. *Además:*

$$r|_{XZ} = \widetilde{m}\ (r)|_{XZ} = \widetilde{m}^*\ (r)|_{XZ}$$

Demostración.

La parte *i)* viene dada por el teorema 5.1.6. Veamos entonces el punto *ii)*. Si K es llave difusa de r y verifica $K \subseteq XZ$, entonces no hay valores redundantes en XZ y por tanto $r|_{XZ} = \widetilde{\Pi}_{XZ}\ (r)$, por lo que K seguirá siendo llave difusa en la relación $\widetilde{\Pi}_{XZ}\ (r)$.

Por otra parte, aplicando el teorema 4.3.2, tenemos que las relaciones $\widetilde{\Pi}^{X}_{XY}\ (r)$ y $RULE^{X}_{XY}(r)$ satisfacen la d.f.d.r. $X \overset{(\alpha,\beta)}{\Longrightarrow} Y$. Por lo tanto, estamos en las condiciones del teorema 5.1.6 fundamental de conservación de dependencias, tomando como primera relación en la reunión, la relación $\widetilde{\Pi}_{XZ}\ (r)$ y como segunda relación $\widetilde{\Pi}^{X}_{XY}\ (r)$ (o bien $RULE^{X}_{XY}(r)$). Aplicando la parte *ii.c)* de dicho teorema, obtenemos:

$$\left(\widetilde{\Pi}_{XZ}\ (r) \widetilde{\sqsupset\!\bowtie} \widetilde{\Pi}^{X}_{XY}\ (r)\right)\Big|_{XZ} = \widetilde{\Pi}_{XZ}\ (r)$$

$$\left(\widetilde{\Pi}_{XZ}\ (r) \widetilde{\sqsupset\!\bowtie}\ RULE^{X}_{XY}(r)\right)\Big|_{XZ} = \widetilde{\Pi}_{XZ}\ (r)$$

pero como $\widetilde{\Pi}_{XZ}\ (r) = r|_{XZ}$, entonces deducimos:

$$\widetilde{m}\ (r)|_{XZ} = \left(\widetilde{\Pi}_{XZ}\ (r) \widetilde{\sqsupset\!\bowtie} \widetilde{\Pi}^{X}_{XY}\ (r)\right)\Big|_{XZ} = r|_{XZ}$$

$$\widetilde{m}^*\ (r)|_{XZ} = \left(\widetilde{\Pi}_{XZ}\ (r) \widetilde{\sqsupset\!\bowtie}\ RULE^{X}_{XY}(r)\right)\Big|_{XZ} = r|_{XZ}$$

Aplicando ahora la parte *ii.d)* del teorema 5.1.6 obtenemos que K es llave difusa de $\widetilde{m}\ (r)$ y de $\widetilde{m}^*\ (r)$. □

Nota 1. En pocas palabras, el teorema nos dice que, bajo las hipótesis dadas, en el proceso de reconstrucción difusa, sólo hay pérdidas de información en el atributo consecuente (Y) de la dependencia que se aisló al romper el esquema original presente. Además, los datos difusos de dichos atributos se mantienen siempre en los márgenes de granularidad impuestos en la relación original.

Nota 2. Obsérverse que el teorema no nos dice que si K es una llave difusa cualquiera, entonces K es llave de $\widetilde{m}\ (r)$. Sólo se obtiene para aquellas llaves candidatas que estén completamente incluidas en el esquema de la primera proyección: $\rho_1 = XZ$.

Nota 3. Por último hay que destacar que la ecuación:

$$\widetilde{m}^*\ (r)|_{XZ} = \left(\widetilde{\Pi}_{XZ}\ (r) \mathrel{\widetilde{\sqsupset\!\!\bowtie}} RULE^X_{XY}(r)\right)\Big|_{XZ} = r|_{XZ}$$

nos dice también que

$$\widetilde{m}^*\ (r)|_{K} = \left(\widetilde{\Pi}_{K}\ (r) \mathrel{\widetilde{\sqsupset\!\!\bowtie}} RULE^X_{XY}(r)\right)\Big|_{K} = r|_{K}$$

ya que $K \subseteq XZ$. En definitiva, volvemos a obtener los mismos valores de la llave difusa.

Ejemplo 5.3.6 . Consideremos la siguiente relación, en la que existe una d.f.d.r $B \overset{(\beta_B,\beta_C)}{\Longrightarrow} C$ y tiene como llave difusa al atributo A:

$r =$

	A	B	C
t_1	a_1	b_1	c
t_2	a_3	$\widetilde{b}$	c_3
t_3	a_4	b_4	c_4
t_4	a_5	b_5	c_5

Obsérvese que la anterior relación coincide salvo en la segunda tupla con la que aparecía en el ejemplo 5.2.13 visto en la página 261. Suponemos que los datos son los representados en la figura 5.4, y que $A(t_1) \simeq^d_\alpha A(t_2)$. Como $B(t_1) \subseteq B(t_2)$, entonces $B(t_1) \simeq^f_{\beta_B} B(t_2)$. Además, habíamos supuesto que $c \simeq^f_{\beta_C} c_3$, o lo que es lo mismo $C(t_1) \simeq^f_{\beta_C} C(t_2)$. En definitiva, podemos concluir que, efectivamente, la relación anterior satisface la dependencia $A \overset{(\alpha,\beta_{BC})}{\Longrightarrow} BC$. Como no hay valores de A redundantes, y todos ellos satisfacen el nivel de granularidad (esto es obvio ya que todos los $A(t_i)$ son crisp), podemos concluir que A es, efectivamente, llave difusa de r.

Por otra parte, r satisface la dependencia $B \overset{(\beta_B,\beta_C)}{\Longrightarrow} C$, tal y como se vió en el ejemplo 5.2.13. Por lo tanto, procedemos a descomponer r en la forma siguiente:

$$\widetilde{\Pi}_{AB}(r) = \begin{array}{|cc|}\hline A & B \\ \hline a_1 & b_1 \\ a_3 & \widetilde{b} \\ a_4 & b_4 \\ a_5 & b_5 \\ \hline \end{array} \qquad \widetilde{\Pi}^{B}_{BC}(r) = RULE^{B}_{BC}(r) = \begin{array}{|cc|}\hline B & C \\ \hline \widetilde{b} & c_3 \vee c \\ b_4 & c_4 \\ b_5 & c_5 \\ \hline \end{array}$$

Si reunimos ahora las relaciones proyectadas, obtenemos:

$$\widetilde{m}(r) = \widetilde{m}^*(r) = \begin{array}{|ccc|}\hline A & B & C \\ \hline a_1 & b_1 & c_3 \vee c \\ a_3 & \widetilde{b} & c_3 \vee c \\ a_4 & b_4 & c_4 \\ a_5 & b_5 & c_5 \\ \hline \end{array}$$

por lo que, efectivamente, $r \sqsubseteq \widetilde{m}(r)$ y $r \sqsubseteq \widetilde{m}^*(r)$. Además, volvemos a obtener los mismos valores para AB que había en la relación original, tal y como aseguraba el teorema 5.3.5. ■

Ahora pasamos a aplicar el teorema 5.3.5 para concluir las formas normales difusas a las que se ha de llegar en un proceso de diseño de una base de datos relacional difusa.

◇ *Segunda y Tercera Formas Normales Difusas*

Supongamos una relación r que no se encuentra en segunda forma normal difusa (2FND). Entonces, según la definición 5.3.2, debe existir alguna d.f.d.r de la forma $X \overset{(\alpha,\beta)}{\Longrightarrow} A$ con A atributo no primo y X contenido propiamente en alguna llave candidata C. En particular, nos interesa el primer punto, es decir:

$$\nexists\ K \text{ llave candidata} \quad \text{tal que } A \in K \tag{5.44}$$

Tomando como Y el conjunto de atributos que verifican 5.44, obtenemos:

$$\nexists\ K \text{ llave candidata} \quad \text{tal que } Y \subseteq K$$

o lo que es lo mismo

$$K \cap Y = \emptyset \quad \forall\ K \text{ llave candidata}$$

lo que equivale, según la descomposición que estamos realizando en $\rho_1 = XZ$, $\rho_2 = XY$ a que

$$K \subseteq XZ \quad \forall\ K \text{ llave candidata}$$

Por lo tanto, estamos en condiciones de aplicar el teorema 5.3.5 independientemente de la llave candidata K que se elija como llave primaria, obteniendo pérdidas únicamente en los atributos consecuentes.

Ejemplo 5.3.7 . Supongamos un esquema $REL = DXBY$ con llave difusa $K = DX$ y con $X \overset{(\alpha,\beta)}{\Longrightarrow} Y$. Entonces descompondríamos REL en $\rho_1 = DXB$ y $\rho_2 = XY$, tal y como puede apreciarse en la figura 5.6, dónde encerramos bajo un cuadrado con línea continua la llave primaria difusa. Si este esquema correspondía a una relación ya existente, entonces r se reconstruye en alguna de las formas siguientes:

$$r \sqsubseteq \tilde{\Pi}_{DXB}(r) \ \widetilde{\sqsupset\!\bowtie}^{X}\ \tilde{\Pi}_{XY}(r) \quad , \quad r \sqsubseteq \tilde{\Pi}_{DXB}(r) \ \widetilde{\sqsupset\!\bowtie}\ RULE^{X}_{XY}(r)$$

$$REL(r)=REL(\tilde{m}(r)) = REL(\tilde{m}^*(r)) = \boxed{D\ X}\ Y\ B$$

$$\rho_1 = \boxed{D\ X}\ B \qquad \rho_2 = \boxed{X}\ Y$$

Figura 5.6. Violación de una 2FND, y su descomposición

■

Ejemplo 5.3.8 . Supongamos un esquema $REL = DXA_1A_2B$ con llave difusa primaria $K = DX$, llave difusa candidata $C = A_2B$, y verificando $X \overset{(\alpha,\beta)}{\Longrightarrow} A_1A_2$. Considerando el conjunto de atributos no primos (A_1), entonces descompondríamos REL en $\rho_1 = DXA_2B$

y $\rho_2 = XA_1$, tal y como puede apreciarse en la figura 5.7, dónde encerramos bajo un cuadrado con línea quebrada la llave difusa candidata C. Si este esquema correspondía a una relación ya existente, entonces r se reconstruye en alguna de las formas siguientes:

$$r \sqsubseteq \widetilde{\Pi}_{DXA_2B}\,(r) \mathrel{\widetilde{\sqsupset\!\!\bowtie}} \widetilde{\Pi}^{X}_{XA_1}\,(r) \quad , \quad r \sqsubseteq \widetilde{\Pi}_{DXA_2B}\,(r) \mathrel{\widetilde{\sqsupset\!\!\bowtie}} RULE^{X}_{XA_1}(r)$$

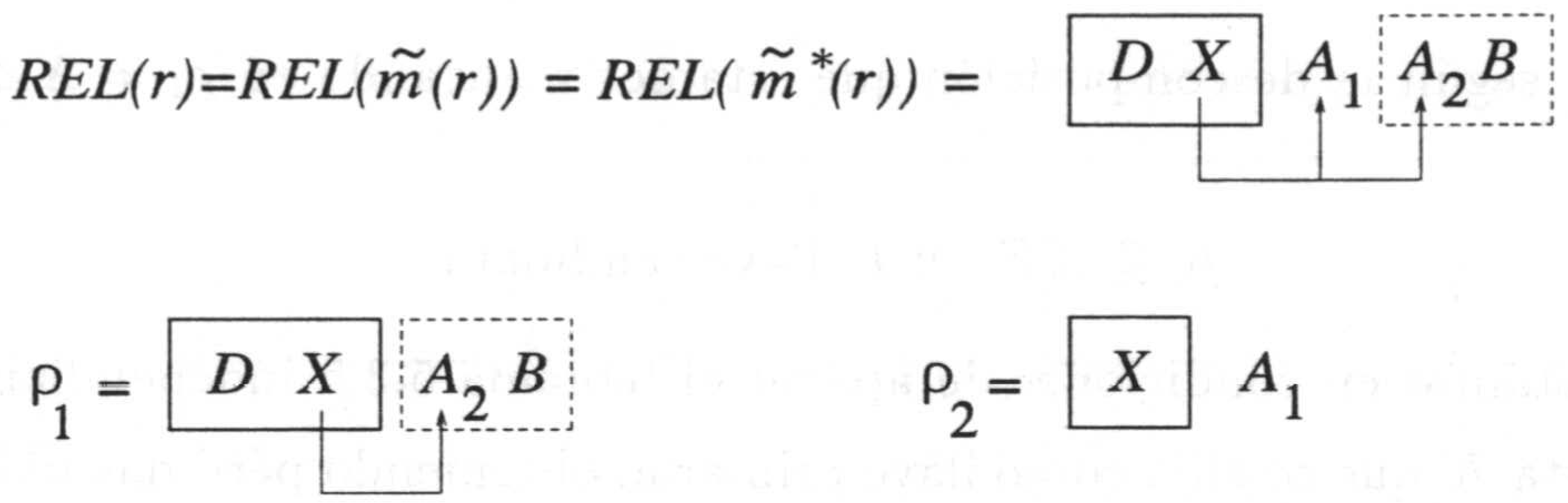

Figura 5.7. Violación de una 2FND, y su descomposición

Obsérvese que al ser A_2 miembro de una llave candidata, la dependencia $X \overset{(\alpha,\beta)}{\Longrightarrow} A_2$ no viola la segunda forma normal, por lo que la relación con esquema ρ_1 no se descompondría.

■

Supongamos ahora una relación r que no se encuentra en tercera forma normal difusa (3FND). Entonces, según la definición 5.3.3, debe existir alguna d.f.d.r de la forma $X \overset{(\alpha,\beta)}{\Longrightarrow} A$ con A atributo no primo y X no contenido en una llave candidata. En particular, nos interesa que A es un atributo no primo, por lo que volvemos a estar en las condiciones anteriores (la condición sobre X no interviene en la aplicabilidad del teorema 5.3.5) y volvemos a considerar el teorema 5.3.5, obteniendo pérdidas únicamente en los atributos consecuentes.

Ejemplo 5.3.9 . Supongamos un esquema $REL = DBXY$ con llave difusa $K = D$ y verificando $X \overset{(\alpha,\beta)}{\Longrightarrow} Y$. Entonces descompondríamos REL en $\rho_1 = DBX$ y $\rho_2 = XY$, tal y como puede apreciarse en la figura 5.8. Si este esquema correspondía a una relación ya existente, entonces r se reconstruye en alguna de las formas siguientes:

$$r \sqsubseteq \widetilde{\Pi}_{DBX}\,(r) \mathrel{\widetilde{\sqsupset\!\!\bowtie}} \widetilde{\Pi}^{X}_{XY}\,(r) \quad , \quad r \sqsubseteq \widetilde{\Pi}_{DBX}\,(r) \mathrel{\widetilde{\sqsupset\!\!\bowtie}} RULE^{X}_{XY}(r)$$

■

$$REL(r)=REL(\tilde{m}(r)) = REL(\tilde{m}^*(r)) = \boxed{D}\ B\ \ X\ \ Y$$

$$\rho_1 = \boxed{D}\ B\ \ X \qquad\qquad \rho_2 = \boxed{X}\ Y$$

Figura 5.8. Violación de una 3FND, y su descomposición

◇ *Forma Normal Difusa de Boyce Codd*

Consideremos ahora una relación r que <u>no</u> se encuentra en FNDBC. Entonces, según la definición 5.3.4, debe existir una d.f.d.r del tipo $X \overset{(\alpha,\beta)}{\Longrightarrow} A$ dónde X no contiene a ninguna llave difusa de $REL(r)$. En el caso de que A no fuese primo, entonces la relación tampoco está en tercera forma normal. Así que supongamos que A es primo, es decir, que r $(REL(r))$ no satisface una FNDBC *propia*. Consideremos el conjunto de atributos A que verifican lo anterior y llamémosle Y; entonces tenemos:

$$\exists\ X \overset{(\alpha,\beta)}{\Longrightarrow} Y \quad \text{tal que} \quad \forall\ K \text{ llave candidata, } K \not\subseteq X$$

$$\text{además } \exists\ C \text{ llave candidata} \quad \text{tal que } Y \subseteq C$$

o dicho de otra forma, en general, una relación no está en FNDBC cuando presenta una d.f.d.r con antecedente que no es una llave difusa, y con consecuente un subconjunto de una llave difusa. Podemos distinguir dos casos:

* Existe, además de C, otra llave difusa candidata K. Entonces podemos aplicar el teorema 5.3.5 tomando como llave difusa K, de forma que K será llave difusa de $\tilde{m}\ (r)$, pero en general, C ya no será llave difusa de $\tilde{m}\ (r)$.

Ejemplo 5.3.10 . Consideremos el esquema de la figura 5.9, dónde $REL(r) = KXYQ$, con llave difusa primaria K y llave difusa candidata $C = YQ$. Este esquema podría descomponerse en la forma $\rho_1 = KQX$, $\rho_2 = XY$ y vemos claramente que $K \subseteq \rho_1$ para poder aplicar el teorema 5.3.5, obteniendo pérdidas únicamente en Y. Si este esquema correspondía a una relación ya existente, entonces r se reconstruye en alguna de las formas siguientes:

$$r \sqsubseteq \tilde{\Pi}_{KXQ}\ (r) \overset{X}{\tilde{\sqsupset\!\bowtie}}\ \tilde{\Pi}_{XY}\ (r) \quad , \quad r \sqsubseteq \tilde{\Pi}_{KXQ}\ (r) \ \tilde{\sqsupset\!\bowtie}\ RULE^{X}_{XY}(r)$$

y K sigue siendo llave difusa, pero ahora C ya no es llave difusa de cualquiera de las relaciones anteriores, lo cual es un precio a pagar y que el experto ha de rechazar o aceptar.

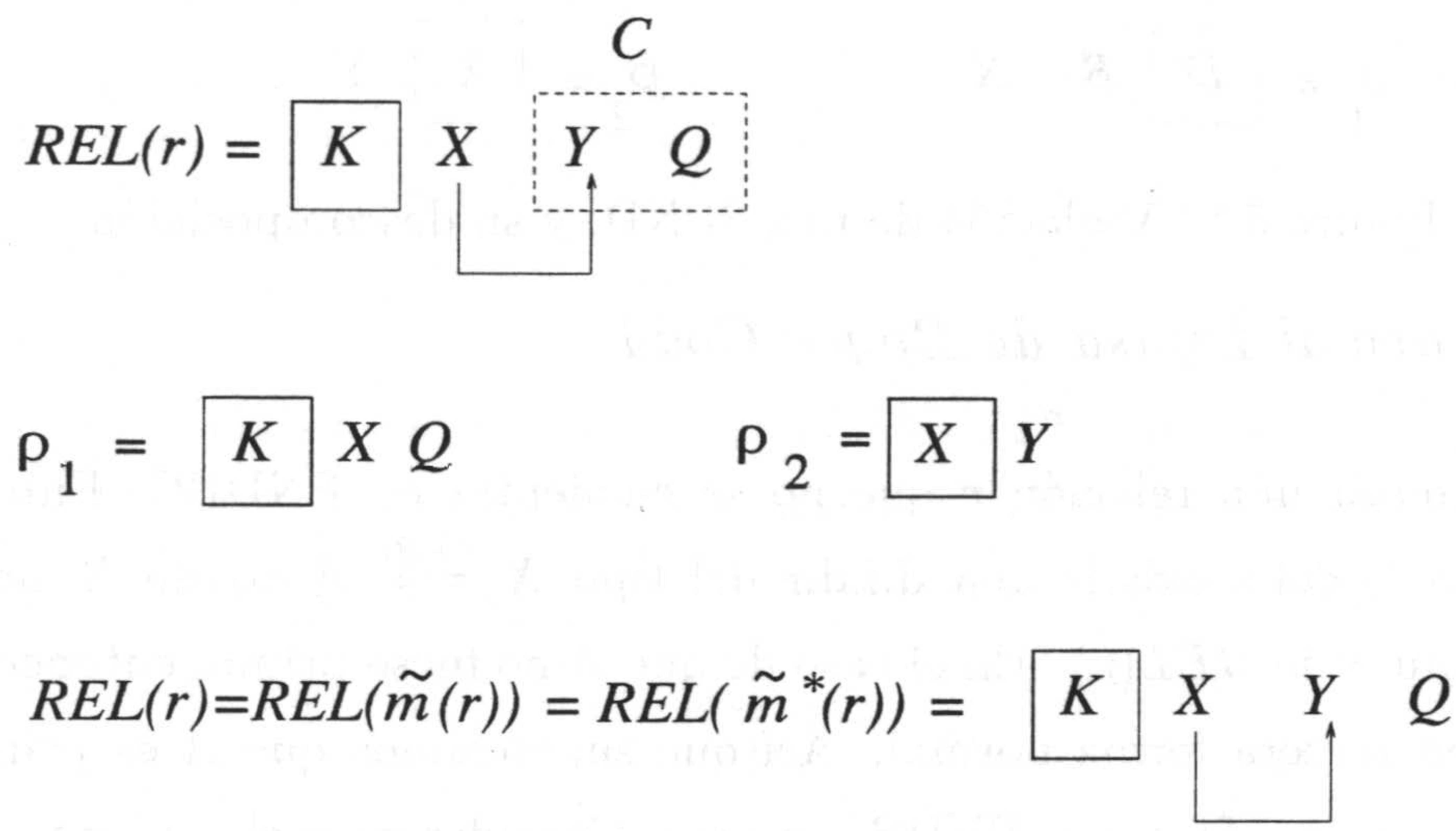

Figura 5.9. Violación de una FNDBC

■

* Sin embargo, puede ser que no exista otra llave difusa candidata, por lo que en este caso, no se verifican las hipótesis del teorema 5.3.5, y por lo tanto únicamente podemos aplicar el teorema 5.2.10 en su forma general. Entonces ya no obtenemos $r|_{\rho_1} = \widetilde{m}\ (r)|_{\rho_1}$ y además, la única llave difusa C deja de serlo en la relación $\widetilde{m}\ (r)$. Es por ello, que no es aconsejable descomponer una relación que no verifica una FNDBC (estrictamente).

Ejemplo 5.3.11 . Considerar el esquema de la figura 5.10, dónde $REL(r) = ZXY$, con única llave difusa primaria $K = ZY$. Este esquema podría descomponerse en la forma $\rho_1 = ZX$, $\rho_2 = XY$ y vemos claramente que $K \not\subseteq \rho_1$, por lo que no podemos aplicar el teorema 5.3.5, y únicamente nos podemos limitar a aplicar el teorema 5.2.10 general. Si este esquema correspondía a una relación ya existente, entonces r se reconstruye en alguna de las formas siguientes:

$$r \sqsubseteq \widetilde{\Pi}_{ZX}\ (r) \sqsupset\!\widetilde{\bowtie}\ \widetilde{\Pi}^{X}_{XY}\ (r) \quad , \quad r \sqsubseteq \widetilde{\Pi}_{ZX}\ (r) \sqsupset\!\widetilde{\bowtie}\ RULE^{X}_{XY}(r)$$

pero ahora $K = Z$ ya no es llave difusa de cualquiera de las relaciones anteriores, lo cual es un precio a pagar y que el experto ha de rechazar o aceptar.

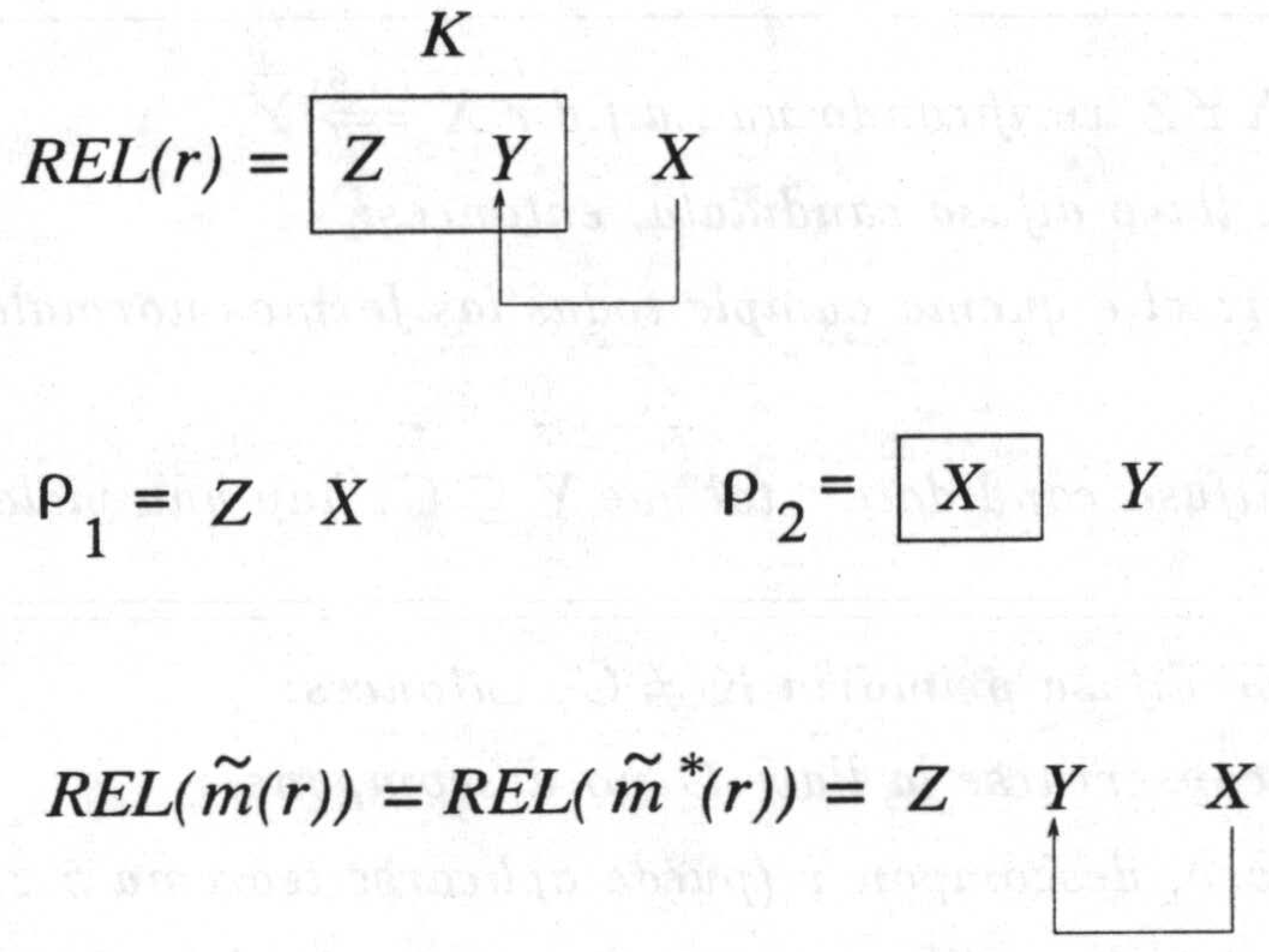

Figura 5.10. Violación de una FNDBC

■

En resumen, podemos dar el siguiente cuadro orientativo en el proceso de diseño de una base de datos relacional difusa:

Sea un esquema $R = XYZ$ *verificando una d.f.d.r* $X \overset{(\alpha,\beta)}{\Longrightarrow} Y$

Si X *contiene una llave difusa candidata, entonces:*

No descomponer: el esquema cumple todas las formas normales difusas

En caso contrario:

Si $\exists$ C *llave difusa candidata tal que* $Y \subseteq C$, *hay una violacion de FNDBC.*

Entonces:

Si $\exists$ K *llave difusa primaria* $K \neq C$, *entonces:*

Si quiere conservarse la llave C *no descomponer.*

En otro caso, descomponer (puede aplicarse teorema 5.2.10).

Si C *es la unica llave difusa, entonces el experto debe decidir si hay que descomponer o no (solo puede aplicarse el teorema 5.3.5).*

En caso contrario, (violacion de 3FND):

Tomar: $Y' = \{A_i \in Y$ *tal que* $A_i \notin C \quad \forall\, C$ *llave difusa candidata*$\}$:

Descomponer R *en* $R - Y'$ *y en* XY'.

Si ya existia fisicamente una instancia r *de* R, *entonces,*

si se quiere recuperar r, *realizar:*

$\tilde{\Pi}_{R-Y'}(r) \;\tilde{\sqsupset\!\bowtie}\; \tilde{\Pi}^{X}_{XY'}(r)$ *o bien* $\tilde{\Pi}_{R-Y'}(r) \;\tilde{\sqsupset\!\bowtie}\; RULE^{X}_{XY'}(r)$

Obsérvese en definitiva, que la máxima a seguir en el proceso de diseño de un esquema relacional es:

Aislar en esquemas independientes los atributos X, Y *que verifiquen* $X \overset{(\alpha,\beta)}{\Longrightarrow} Y$, *siempre y cuando no se parta la llave difusa primaria* K, *es decir, siempre y cuando* $Y \cap K = \emptyset$.

En caso contrario no es deseable la descomposición.

Conclusiones y Líneas de Investigación Futuras

Los resultados presentados en esta memoria pueden resumirse en los siguientes puntos:

1. Hemos planteado los inconvenientes que presentan las aproximaciones existentes a la hora de establecer operadores del álgebra relacional difusa, que se apliquen, no únicamente en consultas, sino también para diseñar la base de datos.

2. Así mismo, se han analizado las desventajas que presentan las extensiones del concepto de dependencia funcional, y se han establecido propiedades genéricas que toda definición de dependencia difusa debería satisfacer.

3. Para establecer una definición de dependencia, coherente con las propiedades anteriores, hemos introducido una primera aproximación que se refleja en la definición 2.5.4.

   Los tres puntos anteriores se han abordado en el segundo capítulo.

4. En el capítulo tercero, hemos planteado la necesidad de comparar los valores de los atributos antecedentes a través de una medida de semejanza débil (definición 3.2.4), mientras que ha de utilizarse una medida fuerte (definición 3.2.14) para comparar los consecuentes. Así mismo, se ha introducido el importante concepto de *nivel de granularidad* (definición 3.2.16), como una restricción de integridad sobre los valores difusos presentes en una base de datos relacional difusa: dicha restricción establece que los datos no pueden ser *demasiado difusos.*

5. Hemos abordado el problema del análisis de la redundancia en un sistema relacional, definiendo operadores de proyección difusa para eliminarla.

6. Se ha introducido una definición de dependencia funcional difusa (*basada en reglas*), de forma que la proyección difusa mantiene dicha dependencia. Esto nos permite un ahorro computacional importante a la hora de comprobar la existencia o no de dicha dependencia. Así mismo, se ha introducido el operador $RULE$ (que también mantiene la dependencia) que actúa compactando aún más los datos que la proyección difusa.

7. Hemos introducido un nuevo operador de reunión difusa que puede aplicarse en un proceso de consulta a la base de datos, y que junto con la proyección difusa, forman un conjunto de operadores del álgebra relacional difusa que aseguran la obtención de descomposiciones *sin pérdidas*. Aplicando este resultado, hemos planteado cuales son las formas normales difusas a las que se habría de llegar en el diseño de una base de datos relacional difusa.

Como líneas de investigación futuras podemos establecer las siguientes:

1. Establecer una equivalencia entre las definiciones de semejanza, bajo condiciones menos restrictivas que las impuestas en los teoremas finales del capítulo tercero.

2. En la definición de dependencia funcional difusa, hemos asumido que los niveles de semejanza eran impuestos por el experto, y pasábamos a comprobar si una relación satisface o no dicha dependencia. Sería interesante desarrollar un mecanismo automático, aplicable sobre una relación arbitraria, que elicite aquellos niveles α, β *mínimos* para los cuales la relación satisfe una (α, β)-d.f.d.r.

3. A la hora de comprobar la existencia de una dependencia, todas las tuplas deben satisfacer las condiciones expuestas en la definición de d.f.d.r. Una posible línea de investigación pasa por considerar dependencias que se verifiquen sobre *la mayoría* de las tuplas, así como dependencias dentro de distintos grupos de tuplas.

4. Actualmente se está desarrollando en lenguaje C, un programa que compruebe la existencia o no de una (α, β)-d.f.d.r en una relación con esquema relacional arbitario. También se están implementando los operadores básicos del álgebra relacional difusa introducidos en esta memoria. En un futuro, pretendemos construir módulos que

desarrollen las anteriores tareas, pero implementándolos en sistemas relacionales gestores de datos difusos generales, como por ejemplo GEFRED (ver Medina et al [58]). Cualquier sistema utilizado, habrá que adaptarlo convenientemente para que pueda soportar las restricciones de integridad difusas expuestas en el capítulo quinto.

5. Completar la teoría de diseño de una base de datos relacional difusa, cuando hay presentes varias dependencias transitivas.

Apéndice. Ejemplo Final

A. Planteamiento del Ejemplo

Supongamos tres clases de plantas, a saber, `Setosa`, `Versicolor` y `Virginica`. Se almacenan en tres relaciones, datos sobre la longitud y anchura de los pétalos y sépalos de distintas plantas. Estas variables conforman el siguiente esquema relacional:

`Long-Pet , Long-Sep , Anch-Pet , Anch-Sep`

La notación que vamos a seguir para representar los datos difusos es la siguiente: Cuando queramos representar el difuso '`Entre` a `y` b' lo notaremos por $a \sim b$ (ver figura 5.11). Otro tipo de dato difuso es el dado en la figura 5.12, que representa la etiqueta '`Aproximadamente` a', y que denotaremos por $\tilde{a}$.

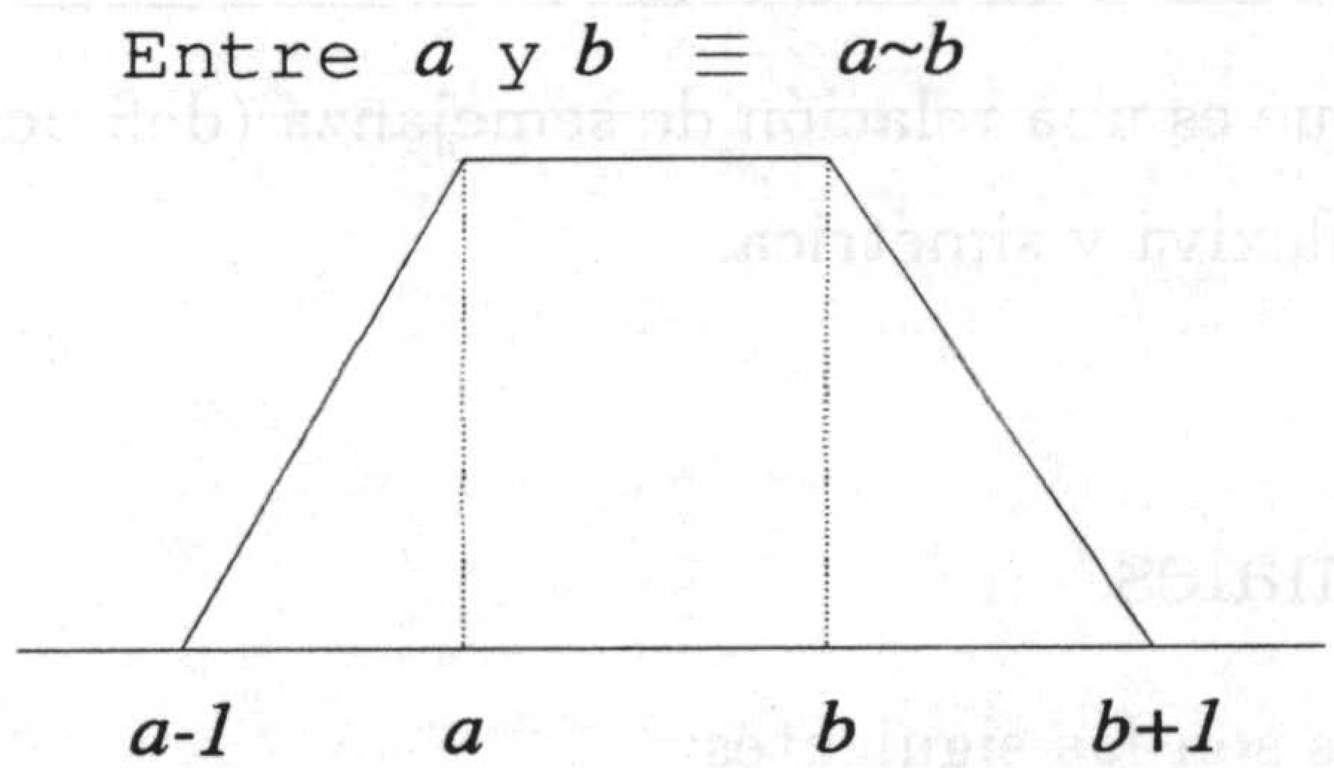

Figura 5.11. Representación del difuso `Entre` a `y` b

Para facilitar la tarea al experto, las relaciones de semejanza entre las longitudes de una clase particular, se construirán atendiendo al siguiente criterio. El experto debe determinar el grado de semejanza η para una diferencia x entre dos longitudes. Por ejemplo, utilizando una unidad de medida *und*:

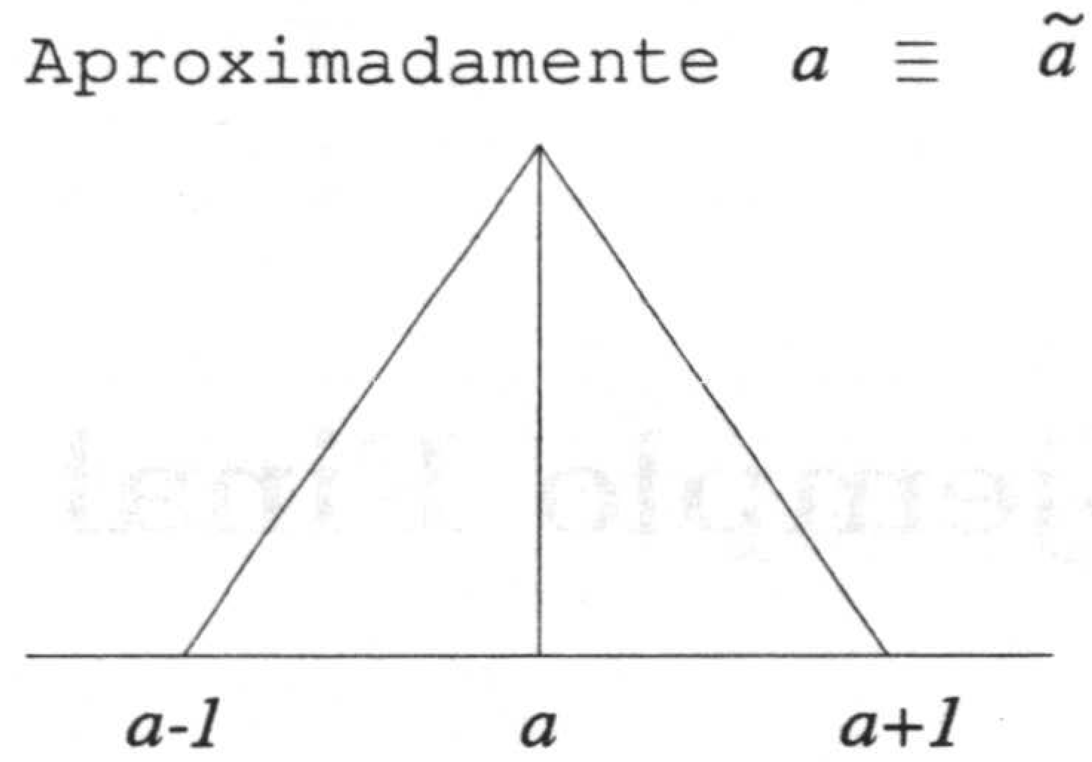

Figura 5.12. Representación del difuso **Aproximadamente** a

Se establece que el grado de semejanza correspondiente a sendas medidas de la anchura de dos pétalos de la clase **Setosa**, *es igual a* $\eta = 0.7$, *cuando dichas medidas difieren en* $x = 4$ *und.*

Claro está, cuanta mayor sea la diferencia entre dichas medidas, será menor el grado de semejanza. Por contra, dos pétalos de igual anchura tendrán semejanza igual a 1. La siguiente función podría ser una representación continua de dicha semejanza (en concreto, habrá que tomar $x = 4$, $\eta = 0.7$):

$$\boxed{R(a,b) = \begin{cases} \frac{1}{|b-a|} x\eta & \text{si } |b-a| \geq x \\ 1 - |b-a| \left(\frac{1-\eta}{x}\right) & \text{si } |b-a| \leq x \end{cases}}$$

Es inmediato comprobar que es una relación de semejanza (definición 2.1.3), es decir, que verifica las propiedades reflexiva y simétrica.

B. Relaciones Originales

Las relaciones originales son las siguientes:

Setosa =

Long − Pet	Long − Sep	Anch − Pet	Anch − Sep
$43 \sim 44$	30	11	$\widetilde{1}$
44	$\widetilde{30}$	13	2
44	32	13	2
46	32	14	$2 \sim 3$
46	$\widetilde{36}$	13	2
46	34	14	3
$47 \sim 48$	32	16	2
48	34	16	2
48	30	14	3
48	31	$14 \sim 16$	2
49	30	14	2
49	36	$\widetilde{14}$	1
50	$34 \sim 36$	14	2
50	34	16	$\widetilde{2}$
$49 \sim 50$	32	$\widetilde{15}$	2
50	30	16	2
50	35	16	6
50	33	14	2
51	35	16	2
51	$\widetilde{38}$	19	$4 \sim 5$
51	38	16	2
51	$32 \sim 33$	17	$\widetilde{5}$
52	41	15	1
52	$\widetilde{34}$	14	2
54	34	15	4

Versicolor =

Long − Pet	Long − Sep	Anch − Pet	Anch − Sep
50	23	$\widetilde{33}$	10
54	30	45	15
55	$25 \sim 26$	44	12
55	24	38	11
$55 \sim 56$	$\widetilde{30}$	45	15
56	25	39	$\widetilde{11}$
57	29	42	13
57	26	35	10
$57 \sim 58$	$\widetilde{28}$	$44 \sim 45$	$\widetilde{13}$
58	27	41	10
58	26	40	12
$\widetilde{59}$	32	48	18
60	$28 \sim 29$	$\widetilde{45}$	15
60	27	51	16
61	$\widetilde{28}$	40	13
61	30	$46 \sim 47$	14
62	$22 \sim 23$	45	15
63	33	47	16
$63 \sim 64$	$\widetilde{32}$	45	$\widetilde{15}$
65	28	46	15
67	31	$\widetilde{44}$	$14 \sim 15$
70	32	47	14

	Long − Pet	Long − Sep	Anch − Pet	Anch − Sep
	49	$\widetilde{25}$	45	17
	58 $\sim$ 59	27	51	19
	58	27	$\widetilde{51}$	19
	59	30	51	$\widetilde{18}$
	61	26 $\sim$ 27	56	14
	61	30	53	18
	62	28	49	18
	62	$\widetilde{34}$	54	22 $\sim$ 23
	63	34	56	24
	63	27	51 $\sim$ 52	18
	63	$\widetilde{25}$	50	19
	63	28	51	15
	64	28	56	$\widetilde{22}$
	64	32	$\widetilde{56}$	23
Virginica =	64	28	56	22
	65	30	55	18
	65	30	53 $\sim$ 54	20
	67	33	57	21
	67	$\widetilde{30}$	55	23
	67	32 $\sim$ 33	57	25
	67	31	56	24
	68	32	$\widetilde{59}$	23
	69 $\sim$ 70	31	59	23
	71	30	59	21
	72	30	58	16
	72	$\widetilde{32}$	60	$\widetilde{18}$
	76	30	66	21
	77	30	61	23
	77	28 $\sim$ 29	67	20 $\sim$ 21
	77	38	67	22
	$\widetilde{79}$	38	64	$\widetilde{20}$

Las relaciones de semejanza asociadas son las siguientes:

– `Relación Setosa`:

$$R_{\texttt{Long-Pet}}(a,b) = R_{\texttt{Long-Sep}}(a,b) = \begin{cases} \frac{1}{|b-a|}0.6 & \text{si } |b-a| \geq 1 \\ 1-|b-a|(1-0.6) & \text{si } |b-a| \leq 1 \end{cases}$$

$$R_{\texttt{Anch-Pet}}(a,b) = \begin{cases} \frac{1}{|b-a|}4 * 0.7 & \text{si } |b-a| \geq 4 \\ 1-|b-a|\left(\frac{1-0.7}{4}\right) & \text{si } |b-a| \leq 4 \end{cases}$$

$$R_{\texttt{Anch-Sep}}(a,b) = \begin{cases} \frac{1}{|b-a|}2 * 0.7 & \text{si } |b-a| \geq 2 \\ 1-|b-a|\left(\frac{1-0.7}{2}\right) & \text{si } |b-a| \leq 2 \end{cases}$$

– `Relación Versicolor`:

$$R_{\texttt{Long-Pet}}(a,b) = R_{\texttt{Long-Sep}}(a,b) = \begin{cases} \frac{1}{|b-a|}0.7 & \text{si } |b-a| \geq 1 \\ 1-|b-a|(1-0.7) & \text{si } |b-a| \leq 1 \end{cases}$$

$$R_{\texttt{Anch-Pet}}(a,b) = \begin{cases} \frac{1}{|b-a|}5 * 0.7 & \text{si } |b-a| \geq 5 \\ 1-|b-a|\left(\frac{1-0.7}{5}\right) & \text{si } |b-a| \leq 5 \end{cases}$$

$$R_{\texttt{Anch-Sep}}(a,b) = \begin{cases} \frac{1}{|b-a|}2 * 0.7 & \text{si } |b-a| \geq 2 \\ 1-|b-a|\left(\frac{1-0.7}{2}\right) & \text{si } |b-a| \leq 2 \end{cases}$$

– `Relación Virginica`: Las relaciones de semejanza asociadas son las siguientes:

$$R_{\texttt{Long-Pet}}(a,b) = R_{\texttt{Long-Sep}}(a,b) = \begin{cases} \frac{1}{|b-a|}0.7 & \text{si } |b-a| \geq 1 \\ 1-|b-a|(1-0.7) & \text{si } |b-a| \leq 1 \end{cases}$$

$$R_{\texttt{Anch-Pet}}(a,b) = \begin{cases} \frac{1}{|b-a|}5 * 0.7 & \text{si } |b-a| \geq 5 \\ 1-|b-a|\left(\frac{1-0.7}{5}\right) & \text{si } |b-a| \leq 5 \end{cases}$$

$$R_{\texttt{Anch-Sep}}(a,b) = \begin{cases} \frac{1}{|b-a|}2 * 0.7 & \text{si } |b-a| \geq 2 \\ 1-|b-a|\left(\frac{1-0.7}{2}\right) & \text{si } |b-a| \leq 2 \end{cases}$$

C. Comprobación de Dependencias y Proyecciones Difusas.

En las tres relaciones, la llave difusa es todo el esquema relacional. Ahora bien, el experto en la materia elicita las siguientes posibles dependencias:

$$\texttt{Long}-\texttt{Pet} \Longrightarrow \texttt{Anch}-\texttt{Pet} \ , \ \texttt{Long}-\texttt{Sep} \Longrightarrow \texttt{Anch}-\texttt{Sep}$$

para cada una de las tres clases de plantas. Los niveles de semejanza asociados a cada atributo son los siguientes:

- Clase `Setosa`:
  - $\alpha_{\texttt{Setosa,Long-Pet}} = \alpha_{\texttt{Setosa,Long-Sep}} = 0.7$
  - $\beta_{\texttt{Setosa,Anch-Pet}} = \beta_{\texttt{Setosa,Anch-Sep}} = 0.7$
- Clase `Versicolor`:
  - $\alpha_{\texttt{Versicolor,Long-Pet}} = \alpha_{\texttt{Versicolor,Long-Sep}} = 0.8$
  - $\beta_{\texttt{Versicolor,Anch-Pet}} = \beta_{\texttt{Versicolor,Anch-Sep}} = 0.6$
- Clase `Virginica`:
  - $\alpha_{\texttt{Virginica,Long-Pet}} = \alpha_{\texttt{Virginica,Long-Sep}} = 0.7$
  - $\beta_{\texttt{Virginica,Anch-Pet}} = \beta_{\texttt{Virginica,Anch-Sep}} = 0.7$

Pasamos por tanto a comprobar si las relaciones satisfacen la definición 4.2.5 de dependencia funcional difusa basada en reglas. Vamos a incluir con detalle, sólo los cómputos efectuados con la clase `Setosa`.

C.1. Clase Setosa

Empezamos analizando la primera dependencia:

$$\texttt{Long}-\texttt{Pet} \overset{(0.7,0.7)}{\Longrightarrow} \texttt{Anch}-\texttt{Pet}$$

Veamos aquellos antecedentes que son semejantes débiles (definición 3.2.4) a grado $\alpha = 0.7$. En primer lugar, para computar el grado de semejanza débil entre dos datos difusos F y F', debemos calcular el término:

$$R^d(F,F') = \max\left\{R(x^-_{F_\alpha}, x^+_{F'_\alpha}), R(x^+_{F_\alpha}, x^-_{F'_\alpha})\right\}$$

dónde

$$x^-_{F_\alpha} = \min\{x \in F_\alpha\} \ , \ x^+_{F_\alpha} = \max\{x \in F_\alpha\}$$

Dichos valores $x^-_{F_\alpha}$ y $x^+_{F_\alpha}$ se calculan, tal y como se ve en la figura 5.13. Para hallar el grado de semejanza fuerte, haremos:

$$R^f(F, F') = \min\left\{R(x^-_{F_\alpha}, x^+_{F'_\alpha}), R(x^+_{F_\alpha}, x^-_{F'_\alpha})\right\}$$

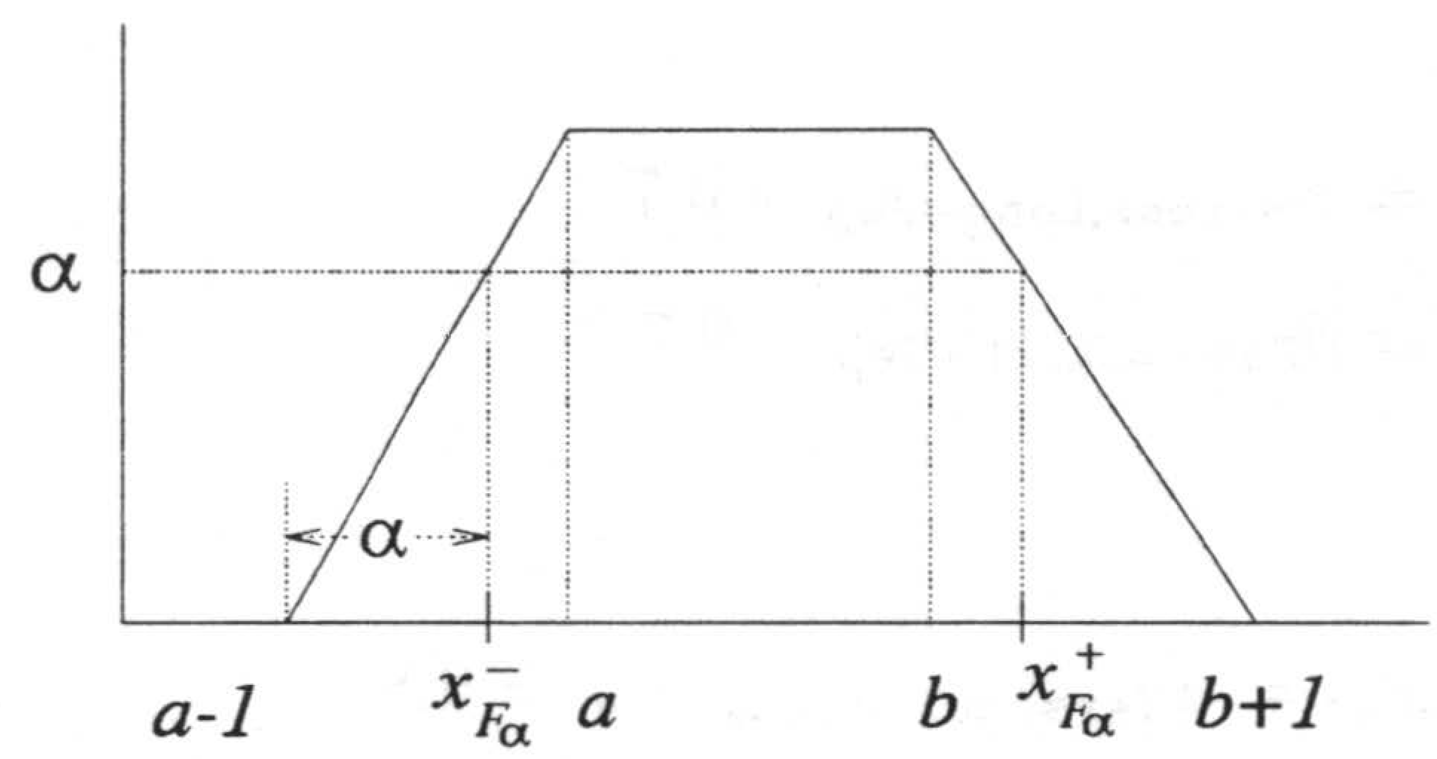

Figura 5.13. Cómputo de $x^-_{F_\alpha}$

Empezamos con las tres primeras tuplas. Como el valor 44 es un crisp incluido en el difuso 43 $\sim$ 44, entonces el grado de semejanza es 1, es decir:

$$44 \subseteq 43 \ , \ 43 \sim 44 \ \Rightarrow \ R^d(43 \sim 44, 44) = 1$$

Como resulta que $R^d(43 \sim 44, 44) \geq \alpha$, entonces tenemos que comprobar si los correspondientes valores de los consecuentes son semejantes fuertes. Efectivamente,

$$R^f(11, 13) = 1 - |b - a|(1 - \frac{1 - 0.7}{4}) = 1 - 2 * 0.075 = 0.85 \geq \beta$$

Realizamos el mismo razonamiento con el resto de las tuplas. Sólo incluimos aquellas tuplas con antecedentes semejantes débiles. Por lo tanto, al ser $R(a, a+1) = 0.6$, que es menor que el umbral impuesto ($\alpha = 0.7$), dos tuplas con valores en antecedentes iguales a a y $a+1$, no influyen en la determinación de la dependencia. Por comodidad, podemos seguir la relación ya proyectada que se detalla en la página 290 (recordar el teorema 4.2.11 de equivalencia)

- $R^d(46, 47 \sim 48) = R(46, 47 - \alpha) = 1 - \alpha\frac{1-\eta}{x} = 1 - 0.7(1 - 0.6) = 0.72 \geq \alpha$

  $R^f(13 \vee 14, 14 \sim 16) = R(13, 16 + \beta) = 1 - (3 + \beta) * 0.075 = 0.722 \geq \beta$

- $R^d(47 \sim 48, 49 \sim 50) = R(48+\alpha, 49-\alpha) = 0.86 \geq \alpha$

  $R^f(14 \sim 16, \tilde{14} \vee \tilde{15} \vee 16) = R(14-\beta, 16) = 0.79 \geq \beta$

- Para el resto de las tuplas, debemos ver que los valores de los consecuentes verifican el nivel de granularidad. Efectivamente:

  $\min\{R^f(16,19), R^f(16,17), R^f(17,19)\} = R(16,19) = 0.77 \geq \beta$

  $R^f(14,15) = R(14,15) = 0.92 \geq \beta$

Por otra parte, los datos que aparecen en la relación `Setosa`, no satisfacen la dependencia

$$\texttt{Long}-\texttt{Sep} \overset{(0.7,0.7)}{\Longrightarrow} \texttt{Anch}-\texttt{Sep}$$

Esto puede comprobarse con las siguientes tuplas:

$$(50\ ,\ 35\ ,\ 16\ ,\ 6)\quad,\quad(51\ ,\ 35\ ,\ 16\ ,\ 2)$$

ya que $R^d(35,35) = 1$, y sin embargo $R^f(6,2) = 0.4 < \beta$.

Así pues, podemos concluir que los datos que aparecen en la relación `Setosa`, corroboran la primera hipótesis del experto, pero no la segunda, es decir:

$$\texttt{Long}-\texttt{Pet} \overset{(0.7,0.7)}{\Longrightarrow} \texttt{Anch}-\texttt{Pet}$$

$$\texttt{Long}-\texttt{Sep} \overset{(0.7,0.7)}{\not\Longrightarrow} \texttt{Anch}-\texttt{Pet}$$

Por lo tanto, la relación original presenta una d.f.d.r dentro de la llave difusa. Así que no se encuentra en FNDBC (pero sí en 3FND), y procedemos a proyectar la relación original en los siguientes esquemas:

$$(\texttt{Long}-\texttt{Pet}, \texttt{Anch}-\texttt{Pet})\quad(\texttt{Long}-\texttt{Pet}, \texttt{Long}-\texttt{Sep}, \texttt{Anch}-\texttt{Sep})$$

Las relaciones obtenidas con la proyección difusa (definición 4.1.7), son las siguientes:

$$\mathtt{Setosa-Pet} = \overset{\sim\mathtt{Long-Pet}}{\Pi}_{\mathtt{Long-Pet,Anch-Pet}}(\mathtt{Setosa}) =$$

Long − Pet	Anch − Pet
$43 \sim 44$	$11 \vee 13$
46	$13 \vee 14$
$47 \sim 48$	$14 \sim 16$
$49 \sim 50$	$\widetilde{14} \vee \widetilde{15} \vee 16$
51	$16 \vee 17 \vee 19$
52	$14 \vee 15$
54	15

Observamos que esta relación sigue satisfaciendo la dependencia

$$\mathtt{Long-Pet} \overset{(0.7,0.7)}{\Longrightarrow} \mathtt{Anch-Pet}$$

sin más que aplicar el teorema 4.2.11. Además, el teorema 4.3.2 garantiza que el atributo `Long-Pet` es una llave difusa de esta relación.

En cuanto a la proyección sobre el segundo esquema, obtenemos la siguiente relación:

$$\mathtt{Setosa-LongPet-Sep}=\widetilde{\Pi}_{\mathtt{Long-Pet,Long-Sep,Anch-Sep}}(\mathtt{Setosa})=$$

Long − Pet	Long − Sep	Anch − Sep
$43\sim44$	30	1
44	$\widetilde{30}$	$\widetilde{1}$
44	32	2
46	32	$2\sim3$
46	$\widetilde{36}$	2
46	34	3
$47\sim48$	32	2
48	34	2
48	30	3
48	31	2
49	30	2
49	36	1
50	$34\sim36$	$\widetilde{2}$
$49\sim50$	32	2
50	30	2
50	35	6
50	33	2
51	35	2
51	$\widetilde{38}$	$4\sim5$
51	38	2
51	$32\sim33$	$\widetilde{5}$
52	41	1
52	$\widetilde{34}$	2
54	34	4

dónde puede apreciarse que se han eliminado dos pares de tuplas redundantes.

Nota. En el caso de que se hubiese satisfecho la dependencia

$$\mathtt{Long-Sep}\overset{(0.7,0.7)}{\Longrightarrow}\mathtt{Anch-Sep}$$

entonces habría que descomponer la relación original en los siguientes esquemas:

$$(\mathtt{Long-Pet,Anch-Pet})\quad(\mathtt{Long-Sep,Anch-Sep})$$

que no tienen atributos en común. Esto se debe a que ambos esquemas describirían entidades que no tienen nada que ver entre sí. En este caso, (como en el correspondiente caso clásico) no tendría sentido aplicar el teorema fundamental de descomposiciones sin pérdidas.

C.2. Clase Versicolor

En el caso de la clase `Versicolor`, se puede comprobar que los datos originales violan ambas dependencias, es decir:

$$\texttt{Long}-\texttt{Pet} \overset{(0.8,0.6)}{\not\Longrightarrow} \texttt{Anch}-\texttt{Pet}$$

$$\texttt{Long}-\texttt{Sep} \overset{(0.8,0.6)}{\not\Longrightarrow} \texttt{Anch}-\texttt{Sep}$$

Por lo tanto, dejamos la relación original tal y como estaba. Como no había tuplas redundantes, no hay que tomar la proyección difusa sobre todo el esquema relacional.

C.3. Clase Virginica

Respecto a la tercera clase, puede comprobarse que los datos presentes en la relación original, satisfacen la siguiente d.f.d.r:

$$\texttt{Long}-\texttt{Pet} \overset{(0.7,0.7)}{\Longrightarrow} \texttt{Anch}-\texttt{Pet}$$

Sin embargo, no se tiene el mismo resultado para la otra dependencia, es decir:

$$\texttt{Long}-\texttt{Sep} \overset{(0.7,0.7)}{\not\Longrightarrow} \texttt{Anch}-\texttt{Sep}$$

Por lo tanto, debemos descomponer atendiendo a los siguientes esquemas relacionales:

$$(\texttt{Long}-\texttt{Pet}, \texttt{Anch}-\texttt{Pet}) \quad (\texttt{Long}-\texttt{Pet}, \texttt{Long}-\texttt{Sep}, \texttt{Anch}-\texttt{Sep})$$

Tomando las proyecciones difusas, nos queda:

$$\texttt{Virginica} - \texttt{Pet} = \Pi^{\sim\texttt{Long-Pet}}_{\texttt{Long-Pet,Anch-Pet}}(\texttt{Virginica}) =$$

Long − Pet	Anch − Pet
49	45
$58 \sim 59$	$\widetilde{51}$
61	$53 \vee 56$
62	$49 \vee 53$
63	$50 \vee (51 \sim 52) \vee 56$
64	$\widetilde{56}$
65	$53 \sim 54 \vee 55$
67	$55 \vee 56 \vee 57$
68	$\widetilde{59}$
$69 \sim 70$	59
71	59
72	$58 \vee 60$
76	66
77	$61 \vee 67$
$\widetilde{79}$	64

En cuanto a la proyección sobre el segundo esquema, obtenemos la siguiente relación:

$$\texttt{Virginica} - \texttt{LongPet} - \texttt{Sep} = \widetilde{\Pi}_{\texttt{Long-Pet,Long-Sep,Anch-Sep}}(\texttt{Virginica}) =$$

Long − Pet	Long − Sep	Anch − Sep
49	$\widetilde{25}$	17
$58 \sim 59$	27	19
59	30	$\widetilde{18}$
61	$26 \sim 27$	14
61	30	18
62	28	18
62	$\widetilde{34}$	$22 \sim 23$
63	34	24
63	27	18
63	$\widetilde{25}$	19
63	28	15
64	28	$\widetilde{22}$
64	32	23
65	30	18
65	30	20
67	33	21
67	$\widetilde{30}$	23
67	$32 \sim 33$	25
67	31	24
68	32	23
$69 \sim 70$	31	23
71	30	21
72	30	16
72	$\widetilde{32}$	$\widetilde{18}$
76	30	21
77	30	23
77	$28 \sim 29$	$20 \sim 21$
77	38	22
$\widetilde{79}$	38	$\widetilde{20}$

D. Recuperación de los Datos Originales

Veamos ahora cómo recuperamos los datos originales a través de las relaciones proyectadas. Analizamos primero la clase `Setosa`, y luego pasamos a la clase `Virginica`. La relación original estaba en 3FND pero no en FNDBC. Por lo tanto, las pérdidas obtenidas vendrán dadas por la aplicación del teorema 5.3.5, es decir:

$$r \sqsubseteq \widetilde{m}\,(r)$$

En nuesto caso, será:

$$\mathtt{Setosa} \sqsubseteq \widetilde{\Pi}_{\mathtt{Long-Pet,Long-Sep,Anch-Sep}}\,(\mathtt{Setosa}) \mathrel{\widetilde{\sqsupset\!\!\bowtie}}^{\mathtt{Long-Pet}} \Pi_{\mathtt{Long-Pet,Anch-Pet}}\,(\mathtt{Setosa})$$

Efectivamente, realizando la reunión difusa del anterior término, obtenemos la siguiente relación:

Long − Pet	Anch − Pet	Long − Sep	Anch − Sep
$43 \sim 44$	$11 \vee 13$	30	1
44	$11 \vee 13$	$\widetilde{30}$	$\widetilde{1}$
44	$11 \vee 13$	32	2
46	$13 \vee 14$	32	$2 \sim 3$
46	$13 \vee 14$	$\widetilde{36}$	2
46	$13 \vee 14$	34	3
$47 \sim 48$	$14 \sim 16$	32	2
48	$14 \sim 16$	34	2
48	$14 \sim 16$	30	3
48	$14 \sim 16$	31	2
49	$\widetilde{14} \vee \widetilde{15} \vee 16$	30	2
49	$\widetilde{14} \vee \widetilde{15} \vee 16$	36	1
50	$\widetilde{14} \vee \widetilde{15} \vee 16$	$34 \sim 36$	$\widetilde{2}$
$49 \sim 50$	$\widetilde{14} \vee \widetilde{15} \vee 16$	32	2
50	$\widetilde{14} \vee \widetilde{15} \vee 16$	30	2
50	$\widetilde{14} \vee \widetilde{15} \vee 16$	35	6
50	$\widetilde{14} \vee \widetilde{15} \vee 16$	33	2
51	$16 \vee 17 \vee 19$	35	2
51	$16 \vee 17 \vee 19$	$\widetilde{38}$	$4 \sim 5$
51	$16 \vee 17 \vee 19$	38	2
51	$16 \vee 17 \vee 19$	$32 \sim 33$	$\widetilde{5}$
52	$14 \vee 15$	41	1
52	$14 \vee 15$	$\widetilde{34}$	2
54	15	34	4

que puede comprobarse que es una extensión de la relación original `Setosa` que aparece en la página 283.

Con respecto a la clase `Virginica`, tenemos el mismo planteamiento que el dado para la clase `Setosa`, y la reunión difusa de las relaciones proyectadas es la siguiente:

$$\widetilde{\Pi}_{\texttt{Long-Pet,Long-Sep,Anch-Sep}}(\texttt{Virginica}) \overset{\sim\texttt{Long-Pet}}{\widetilde{\sqsupset\!\bowtie}} \widetilde{\Pi}_{\texttt{Long-Pet,Anch-Pet}}(\texttt{Virginica})$$

que puede comprobarse que es una extensión de la relación original `Virginica` que aparece en la página 285:

`Long − Pet`	`Anch − Pet`	`Long − Sep`	`Anch − Sep`
49	45	$\widetilde{25}$	17
$58 \sim 59$	$\widetilde{51}$	27	19
59	$\widetilde{51}$	30	$\widetilde{18}$
61	$53 \vee 56$	$26 \sim 27$	14
61	$53 \vee 56$	30	18
62	$49 \vee 53$	28	18
62	$49 \vee 53$	$\widetilde{34}$	$22 \sim 23$
63	$50 \vee (51 \sim 52) \vee 56$	34	24
63	$50 \vee (51 \sim 52) \vee 56$	27	18
63	$50 \vee (51 \sim 52) \vee 56$	$\widetilde{25}$	19
63	$50 \vee (51 \sim 52) \vee 56$	28	15
64	$\widetilde{56}$	28	$\widetilde{22}$
64	$\widetilde{56}$	32	23
65	$53 \sim 54 \vee 55$	30	18
65	$53 \sim 54 \vee 55$	30	20
67	$55 \vee 56 \vee 57$	33	21
67	$55 \vee 56 \vee 57$	$\widetilde{30}$	23
67	$55 \vee 56 \vee 57$	$32 \sim 33$	25
67	$55 \vee 56 \vee 57$	31	24
68	$\widetilde{59}$	32	23
$69 \sim 70$	59	31	23
71	59	30	21
72	$58 \vee 60$	30	16
72	$58 \vee 60$	$\widetilde{32}$	$\widetilde{18}$
76	66	30	21
77	$61 \vee 67$	30	23
77	$61 \vee 67$	$28 \sim 29$	$20 \sim 21$
77	$61 \vee 67$	38	22
$\widetilde{79}$	64	38	$\widetilde{20}$

Bibliografía

[1] A.V. Aho, C. Beeri, and J.D. Ullman. The Theory of Joins in Relational Databases. *ACM Transactions on Database Systems*, 4(3):297–314, 1979.

[2] A.V. Aho, C. Beeri, and J.D. Ullman. The Theory of Joins in Relational Databases. Corrigendum. *ACM Transactions on Database Systems*, 8(2):287, 1979.

[3] M. Annvari and G.F. Rose. Fuzzy Relational Databases. In Bezdek, editor, *Analysis of Fuzzy Information*, volume II. CRC Press, 1987.

[4] W.W. Armstrong. Dependency Structures of Data Base Relationships. In *Proc. IFIP Congress*, pages 580–583. North Holland. Amsterdam, 1974.

[5] J.F. Baldwin. A Fuzzy Relational Inference Language for Expert Systems. In *Proceedings of the 13th IEEE International Symposium on Multivalued Logic. Kyoto. Japan*, pages 416–423, 1983.

[6] J.F. Baldwin and S.Q. Zhou. A Fuzzy Relational Inference Language. *Fuzzy Sets and Systems*, 14:155–174, 1984.

[7] C. Beeri, R. Fagin, and J.H. Howard. A Complete Axiomatization for Functional and Multivalued Dependencies. In *ACM SIGMOD International Symposium on Management of Data*, pages 47–61, 1977.

[8] P.A. Bernstein. Synthesizing third normal form relations from functional dependencies. *ACM Transactions on Database Systems*, 1(4):277–298, 1976.

[9] J. Biskup. A Formal Approach to Null Values in Database Relations. In H. Gallaire and J.M. Nicolas, editors, *Workshop: Formal Bases for Databases.* Plenum Press, December 1979.

[10] J. Biskup. A Foundation of Codd's Relational Maybe-Operations. *ACM Transactions on Database Systems*, 8(4), 1983.

[11] P. Bosc, M. Galibourg, and G. Hamon. Fuzzy Queryng with SQL: Extensions and Implementation Aspects. *Fuzzy Sets and Systems*, 28:333–349, 1988.

[12] P. Bosc and O. Pivert. Flexible Queries, Discriminated Answers and Fuzzy Sets. In *Fifth IFSA World Congress*, pages 525–528, 1993.

[13] P. Bosc, O. Pivert, and K. Farquhar. Integrating Fuzzy Queries into a Database Management System: the Example of Dec-Information Warehouse. In *International Conference on Information Processing and Management of Uncertainty in Knowledge Based Systems, IPMU'92*, pages 259–263, 1992.

[14] B.P. Buckles and F.E. Petry. Fuzzy Databases and their Applications. In Gupta and Sanchez, editors, *Fuzzy Information and Decision Processes.*, pages 361–371. North-Holland, 1982.

[15] B.P. Buckles and F.E. Petry. A Fuzzy Representation of Data for Relational Databases. *Fuzzy Sets and Systems*, 7:213–226, 1982.

[16] B.P. Buckles and F.E. Petry. Extending the Fuzzy Database with Fuzzy Numbers. *Information Sciences*, 34:145–155, 1984.

[17] B.P. Buckles, F.E. Petry, and H.S. Sachar. A Domain Calculus for Fuzzy Relational Databases. *Fuzzy Sets and Systems*, 29:327–340, 1989.

[18] G. Chen, E.E. Kerre, and J. Vandenbulcke. Fuzzy Functional Dependency and its Axiomatic System in a Fuzzy Relational Data Model. In *International Conference on Information Processing and Management of Uncertainty in Knowledge Based Systems, IPMU'92*, pages 313–317, 1992.

[19] G. Chen, J. Vandenbulcke, and E. Kerre. A General Treatment of Data Redundancy in a Fuzzy Relational Model. *Journal of the American Society for Information Science*, 43(4):304–311, 1992.

[20] G. Chen, J. Vandenbulcke, and E.E. Kerre. A Step Towards the Theory of Fuzzy Relational Database Design. In *Proceedings on Computer, Management 91*, pages 44–47, 1991.

[21] E.F. Codd. A Relational Model of Data for Large Shared Data Banks. *Commun. ACM*, 13(6):377–387, 1970. Aqui se introduce definicion de df y formas normales hasta 3fn.

[22] E.F. Codd. Normalized Data Base Structure: A Brief Tutorial. In *ACM SIGFIDET Workshop on Data Description, Access, and Control. San Diego*, 1971.

[23] E.F. Codd. *Further Normalization of the Data Base Relational Model*, volume 6 of *Courant Computer Science Symposia Series.* Englewood Cliffs, N.J. Prentice Hall, 1972. Aqui se introduce FNBC.

[24] E.F. Codd. Extending the Database Relational Model to Capture more Meaning. *ACM Transactions on Database Systems*, 4(4):397–434, 1979.

[25] E.F. Codd. Missing Information (Applicable and Inapplicable) in Relational Databases. *ACM Sigmod*, 15(4), 1986.

[26] E.F. Codd. More Commentary on Missing Information in Relational Databases (Applicable and Inapplicable Information). *ACM Sigmod*, 16(1), 1986.

[27] E.F. Codd. *The Relational Model for Database Management.* Addison-Wesley, Reading, Mass., 1990.

[28] J.C. Cubero, J.M. Medina, and M.A. Vila. Influence of Granularity Level in Fuzzy Functional Dependencies. In M. Clarke, R. Kruse, and S. Moral, editors, *Symbolic and Quantitative Approaches to Reasoning and Uncertainty. Lecture Notes in Computer Science 747*, pages 73–78. Springer Verlag, 1993.

[29] J.C. Cubero, O. Pons, and M.A. Vila. Weak and Strong Resemblance in Fuzzy Functional Dependencies. In *Submitted to IEEE'94 International Conference.*

[30] J.C. Cubero and M.A. Vila. A New Definition of Fuzzy Functional Dependencies in Fuzzy Relational Databases. *To appear in International Journal of Intelligent Systems.*

[31] C.J. Date. Null Values in Database Management. In C.J. Date, editor, *Relational Database: Selected Writings.* Addison Wesley, 1986.

[32] C.J. Date. EXISTS Is Not Exists! (Some Logical Flaws in SQL). In C.J. Date, editor, *Relational Database Writings 1985-1989.* Addison Wesley, 1989.

[33] C.J. Date. NOT is Not Not! (Notes on Three Valued Logic and Related Matters). In C.J. Date, editor, *Relational Database Writings 1985-1989.* Addison Wesley, 1989.

[34] C.J. Date. *An Introduction to Data Bases Systems. Vol I.* Addison-Wesley, Reading, Mass., 1990.

[35] M. Delgado. On a Definition of Fuzzy Function. In *Proceedings of the First World Conference on Mathematics at the Service of Man. Barcelona*, pages 426–448, July 1977.

[36] M. Delgado. Una Nueva Definición de Aplicación Difusa. *Stochastica*, IV(1):75–80, 1980.

[37] M. Delgado, J.L. Verdegay, and M.A. Vila. Decision Making Models. *International Journal of Intelligent Systems.*

[38] C. Delobel and R.C. Casey. Decomposition of a Database and the Theory of Boolean Switching Functions. *IBM J. Res. Devel.*, 17(5):370–386, 1972.

[39] R. Demolombe. Uncertainty in Intelligent Data bases. Technical report, ONERA / CERT. Toulouse, 1992.

[40] D. Dubois and H. Prade. *Fuzzy Sets and Systems: Theory and Applications.* Academic Press, N.Y., 1979.

[41] D. Dubois and H. Prade. *Possibility Theory.* Plenum Press, N.Y., 1988.

[42] D. Dubois and H. Prade. Generalized Dependencies in Fuzzy Data Bases. In *International Conference on Information Processing and Management of Uncertainty in Knowledge Based Systems, IPMU'92*, pages 263–267, 1992.

[43] R. Elmasri and S.B. Navathe. *Fundamentals of Database Systems.* The Benjamin/Cummings Publishing Co., 1989.

[44] R. Fagin and M.Y. Vardi. The Theory of Data Dependencies: A Survey. In *Mathematics of Information Processing. Proceedings of Symposia in Applied Mathematics*, volume 34, pages 19–71. American Mathematical Society. Providence. Rhode Island, 1986.

[45] J. Grant. Null Values in a Relational Data Base. *Inf. Process. Lett.*, 6(5):156–157, 1979.

[46] J. Grant. Partial Values in a Tabular Database. *Inf. Process. Lett.*, 9(2):97–99, 1979.

[47] I.J. Heath. Technical report, IBM Internal Memo, 1971.

[48] I.J. Heath. Unacceptable File Operations in a Relational Database. In *ACM SIGFIDET Workshop on Data Description, Access, and Control. San Diego*, 1971.

[49] T. Imielinski and W. Lipski. Incomplete Information in Relational Databases. *Journal of ACM*, 31(4), 1984.

[50] J. Kacprzyk and A. Ziolkowski. Database Queries with Fuzzy Linguistic Quantifiers. *IEEE Transactions on Syst., Man, and Cyb.*, SMC-16(3), 1986.

[51] A. Kiss. On Fuzzy Relational Databases. In *Mathematical Sciences. Past and Present*, volume 3, pages 1183–1193, 1990.

[52] A. Kiss. λ-Decomposition of Fuzzy Relational Databases. In *Annales Univ. Budapest, Sect. Comp.*, volume 12, pages 133–142. 1991.

[53] T.L. Kunii. DATAPLAN: An Interface Generator for Database Semantics. *Information Sciences*, 10:279–298, 1976.

[54] M. Lacroix and A. Pirotte. Generalized Joins. *ACM Sigmod*, 8(3):14–15, 1976.

[55] W.Jr. Lipski. On Semantic Issues Connected with Incomplete Information Databases. *ACM Transactions on Database Systems*, 4(3):262–296, 1979.

[56] K.C. Liu and R. Sunderaman. Indefinite and Maybe Information in Relational Databases. *ACM Transactions on Database Systems*, 15(1), 1990.

[57] J.M. Medina, J.C. Cubero, O. Pons, M. Prados, and M.A. Vila. Un Modelo de Base de Datos Difusa Aplicado a InformaciónMédica. In *Proc. of the First Spanish Congress on Fuzzy Logic and Technology*, pages 1–6, 1991.

[58] J.M. Medina, O. Pons, and M.A. Vila. Gefred: A Generalized Model for Fuzzy Relational Databases. *To appear in Information Sciences.*

[59] N. Mouaddib. The Nuanced Identification in a Data Base - The Nuanced Relational Division. In *International Conference on Information Processing and Management of Uncertainty in Knowledge Based Systems, IPMU'92*, pages 267–271, 1992.

[60] H. Nakajima, T. Sogoh, and M. Arao. Development of an Efficient Fuzzy SQL for Large Scale Fuzzy Relational Database. In *Fifth IFSA World Congress*, pages 517–520, 1993.

[61] M.A. Vila O. Pons, J.M. Medina. Handling Imprecise Medical Information in the Framework of Logic Fuzzy Databases. *To appear in International Journal of Fuzzy Systems.*

[62] S.C. Park, C.S. Kim, and D.S. Kim. Fuzzy Querying in Relational Databases. In *Fifth IFSA World Congress*, pages 533–536, 1993.

[63] O. Pons, M.A. Vila, and M. Delgado. Inferencia a Partir de una Base de Datos Difusa, Utilizando Reglas Difusas. In *Tercer Congreso en Tecnologías y Lógica Fuzzy. Santiago de Compostela*, 1993.

[64] H.B. Potoczny. On Similarity Relations in Fuzzy Relational Databases. *Fuzzy Sets and Systems*, 12:231–235, 1984.

[65] H. Prade and C. Testemale. Generalizing Database Relational Algebra for the Treatment of Incomplete/Uncertain Information and Vague Queries. *Information Sciences*, 34:115–143, 1984.

[66] H. Prade and C. Testemale. Fuzzy Relational Databases: Representational Issues and Reduction Using Similarity Measures. *Journal of the American Society for Information Science*, 38(2):118–126, 1987.

[67] H. Prade and C. Testemale. Representation of Soft Constraints and Fuzzy Attribute Values by Means of Possibility Distributions in Databases. In Bezdek, editor, *Analysis of Fuzzy Information, Vol II*, pages 213–229. CRC Press, 1987.

[68] K. Raju and A. Majumdar. The Study of Joins in Fuzzy Relational Databases. *Fuzzy Sets and Systems*, 21:19–34, 1987.

[69] K. Raju and A. Majumdar. Fuzzy Functional Dependencies and Loss Less Join Decomposition on Fuzzy Relational Database Systems. *ACM Transactions on Database Systems*, 13(2):129–166, 1988.

[70] J. Rissanen. Independent Components of Relations. *ACM Transactions on Database Systems*, 2(4):317–325, 1977.

[71] E.A. Rundensteiner, L.W. Hawkes, and W. Bandler. On Nearness Measures in Fuzzy Relational Data Models. *International Journal of Approximate Reasoning*, 3:267–298, 1989.

[72] S. Shenoi and A. Melton. Proximity Relations in the Fuzzy Relational Database Model. *Fuzzy Sets and Systems*, 31:285–296, 1989.

[73] S. Shenoi and A. Melton. An Extended Version of the Fuzzy Relational Database Model. *Information Sciences*, 52:35–52, 1990.

[74] S. Shenoi, A. Melton, and L.T. Fan. Functional Dependencies and Normal Forms in the Fuzzy Relational Database Model. *Information Sciences*, 60:1–28, 1992.

[75] L. Siklossy and J.L. Lauriere. Removing Restrictions in the Relational Database Model: An Application of Problem Solving Techniques. In *Proc. National Conf. in Artificial Intelligence, Pittsburg.*

[76] M. Sugeno. *Theory of Fuzzy Integrals and its Applications.* PhD thesis, Tokio Institute of Technology, 1974.

[77] R.C. Tripathy and P.C. Saxena. Multivalued Dependencies in Fuzzy Relational Databases. *Fuzzy Sets and Systems*, 38:267–279, 1990.

[78] J.D. Ullman. *Principles of Database and Knowledge-Base Systems, vol. I.* Computer Science Press., 1988.

[79] M. Umano. Freedom-0: A Fuzzy Databases System. In M.M. Gupta and E. Sanchez, editors, *Fuzzy Information and Decision Processes.*, pages 339–347. North Holland, 1982.

[80] M. Umano. Retrieval from Fuzzy Databases by Relational Algebra. In *Proc. of IFAC Symposium on Fuzzy Information Knowledge Representation and Decision Analysis (Marseille-France)*, pages 1–6. Pergamon Press, 1983.

[81] M. Umano and S. Fukami. Perspectives of Fuzzy Databases. *Japanese Journal of Fuzzy Theory and Systems*, 3(1):75–91, 1991.

[82] M. Umano and S. Miyamoto. Recent Development of Fuzzy Database Systems and Applications. In *Fifth IFSA World Congress*, pages 537–540, 1993.

[83] Y. Vassiliou. Functional Dependencies and Incomplete Information. *VLDB*, pages 260–269, 1980.

[84] M.A. Vila, J.C. Cubero, J.M. Medina, and O. Pons. The Generalized Selection: an Alternative Way for the Quotient Operations in Fuzzy Relational Databases. In *To be presented at International Conference on Information Processing and Management of Uncertainty in Knowledge Based Systems, IPMU'94.*

[85] M.A. Vila, J.C. Cubero, J.M. Medina, and O. Pons. A Logic Approach to Fuzzy Relational Databases. *To appear in International Journal of Intelligent Systems.*

[86] M.A. Vila, J.C. Cubero, J.M. Medina, and O. Pons. Logic and Fuzzy Relational Databases: a new Language and a new Definition. In P. Bosc and J. Kacprzyk, editors, *Fuzzy Sets and Possibility Theory in Database Management Systems.* Physica Verlag, 1993.

[87] M.A. Vila, J.C. Cubero, J.M. Medina, and O. Pons. On the Use of a Logical Definition of Fuzzy Relational Databases. In *Second IEEE International Conference on Fuzzy Systems, S. Francisco*, pages 489–499, 1993.

[88] L. Wei-Yi. Extending the Relational Modelo to deal with Fuzzy Values. *Fuzzy Sets and Systems*, 60:207–212, 1993.

[89] L. Weiyi. The Reduction of the Fuzzy Data Domain and Fuzzy Consistent Join. *Fuzzy Sets and Systems*, 50:89–96, 1992.

[90] E.A. Wong. Statistical Approach to Incomplete Information in Database Systems. *ACM Transactions on Database Systems*, 7(3):470–488, 1982.

[91] R.R. Yager. Fuzzy Quotient Operators. In *International Conference on Information Processing and Management of Uncertainty in Knowledge Based Systems, IPMU'92*, pages 317–323, 1992.

[92] K.B. Yue. A More General model for Handling Missing Information in Relational Data Bases using a 3-valued Logic. *SIGMOD Record*, 20(3), 1991.

[93] L.A. Zadeh. Fuzzy Sets. *Information Control*, 8:338–353, 1965.

[94] L.A. Zadeh. Similarity Relations and Fuzzy Orderings. *Information Sciences*, 3:177–200, 1971.

[95] L.A. Zadeh. The Concept of a Linguistic Variable and its Application to Approximate Reasoning, I. *Information Sciences*, 8:199–249, 1975.

[96] L.A. Zadeh. The Concept of a Linguistic Variable and its Application to Approximate Reasoning, II. *Information Sciences*, 8:301–357, 1975.

[97] L.A. Zadeh. The Concept of a Linguistic Variable and its Application to Approximate Reasoning, III. *Information Sciences*, 9:43–80, 1976.

[98] L.A. Zadeh. Fuzzy Sets as a Basis for a Theory of Possibility. *Fuzzy Sets and Systems*, 1:3–28, 1978.

[99] L.A. Zadeh. PRUF-A Meaning Representation Language for Natural Languages. *Int. J. Man-Machine Studies*, 10:395–460, 1978.

[100] L.A. Zadeh. A Theory of Approximate Reasoning. In J. Hayes, D. Michi, and L.I. Mikulich, editors, *Machine Intelligence*, volume 9, pages 149–194. Halstead Press, 1979.

[101] M. Zemankova-Leech and A. Kandel. *Fuzzy Relational Databases: a Key to Expert Systems*. Rheinland: Verlag TUV., 1984.