PONTIFICIA UNIVERSIDAD CATÓLICA DE CHILE
FACULTAD DE MATEMÁTICAS

Lógicas Anotadas y Algebrización.

Por

Guillermo Ortíz Rico

Tesis presentada a la Facultad de Matemáticas
de la Pontificia Universidad Católica de Chile,
como un requisito para optar al grado de
Doctor en Ciencias Exactas mención Matemática.

Profesor Guía	:	Renato Lewin	Universidad Católica de Chile
Comisión Informante	:	Dra. Irene Mikenberg	Universidad Católica de Chile
		Dra. Marta Sagastume	Universidad Nacional de la Plata

25 de Abril de 2007
Santiago-Chile

Dedicatoria

A Guille y Cami

Agradecimientos

Es mi deseo dejar constancia del eterno agradecimiento a mi familia, en especial a mis padres quienes un día de septiembre del año 1969 me llevaron a la escuela de donde no he vuelto. Es natural recalcar la gran importancia de mi madre al insistir desde mi pequeña infancia acerca de "mis dones de inteligencia[1]", así como el gran respeto de mi hermano José Orlando por mi libre elección, quién junto a mi padre me permitieron cobrar la independencia necesaria para optar por este fascinante (aunque largo) camino que hoy me acontece.

Naturalmente son muchas las otras personas que me han colaborado en el trabajo académico en sí, a quienes manifiesto mi agradecimiento; mi profesor guía Renato Lewin, por sus siempre oportunos comentarios en la orientación de mi trabajo de tesis. Las profesoras Victoria Marshall e Irene Mikenberg, quienes junto a Renato contribuyeron de una forma muy honesta a mi formación básica en el área de lógica. A mis profesores del programa de Posgrado, dentro de los que quisiera destacar al profesor Pomareda por su especial don de gentes, y al profesor Elgueta, por su particular ejemplo de trabajo y sencillez, que sobrepasa mi admiración y la de muchos de los compañeros a un plano estrictamente personal. A mis amigos y compañeros del programa, en especial a Osvaldo, Hugo y Carlitos quienes sin duda constituyen un valioso aporte en mi trabajo y vida misma.

Ahora digo que agradezco al grupo completo de personas que conforman la Facultad de Matemáticas de la PUC, quienes sin duda de una u otra forma contribuyeron a hacer muy placentera mi estadía en vuestro país, baste con señalar nombres como Virginia, Miriam y la señora Tina.

Al Departamento de Matemáticas de la Universidad del Valle y a la Dirección de Investigación y Post-grado de la Pontificia Universidad Católica de Chile por su financiamiento.

En esta etapa final es apenas justo agradecer a mis estudiantes a quienes represento en el nombre de Jhovanny, y a mis amigos del grupo en extenso de Historia

[1]Que aunque sólo fuesen creencias de mi madre, bastante impulso me han dado.

de la Matemática de Univalle que represento en el nombre de Maribel, quienes en su conjunto han hecho menos difícil la misma, pues han constituido un gran soporte de entusiasmo.

Mis agradecimientos finales, pero no menos importantes, para Daniela y Camilo que me dieron mi título más preciado *Papá de Camilo*.

Índice general

Resumen

El presente trabajo tiene por objeto proporcionar nuevos desarrollos teóricos de las Lógicas Anotadas, que esperamos permitan nuevas posibilidades de aplicaciones. A fín de cuidar que éstas nuevas propuestas sean razonables nos propondremos a garantizar la algebrizabilidad de tales generalizaciones.

En la espera de hacer el trabajo autocontenido daremos tres capítulos iniciales, que permitirán contextualizar en forma amplia y precisa los resultados de la presente investigación. El primero cubrirá el tema de *la paraconsistencia,* el segundo el de *las lógicas anotadas,* y tercero el de *la algebrización.*

En el Capítulo 4, se presenta una revisión general de la programación lógica actual mostrando sus cercanías a las lógicas anotadas, mostrando la necesidad de proponer lógicas anotadas de segunda generación. Realmente nosostros propondremos tres sistemas de lógicas anotadas, que se desarrollan en los capítulos finales. En el final del capítulo 4 nos centramos en el manejo técnico de anotaciones de anotaciones, así como la posibilidad de escoger de manera muy general el álgebra de anotaciones, cuidando siempre tal escogencia a fin de garantizar la algebrizabilidad de los sistemas de lógicas anotadas por proponer.

En el Capítulo 5 se presenta un desarrollo de lógicas anotadas en semigrupos, obteniéndose como resultado importante la algebrización de tales lógicas. Además se muestra que si una lógica es algebrizable, la correspondiente lógica anotada sobre un semigrupo también lo es.

En el Capítulo 6 se presenta un desarrollo de lógicas anotadas fuzzy. Este tipo de ejemplos cobra singular importancia para el presente trabajo, pues muestra la cercanía de nuestros desarrollos teóricos a las aplicaciones. Análogamente al capítulo 5, se demuestra la algebrización de tales lógicas.

En el Capítulo 7 se presenta un ejemplo particular de lógicas anotadas donde las anotaciones son tomadas en Birretículos, y análogamente a los capítulo 5 y 6, se demuestra la algebrización de tales lógicas. Parte de estos desarrollos fué presentado en el *II World Congress On Paraconsistency,* Juquehy São Sebastião, Sao Paulo, Brazil, Mayo 08 - 12 de 2000, trabajo que fue publicado en los proceedings del evento por Marcer Dekker, New York, Abril del 2002.

Capítulo 1

Paraconsistencia

Es nuestro deseo iniciar con un marco muy general, y a pesar de las dificultades propias de satisfacer un deseo de este tipo, creemos que una buena idea es inscribirnos a una verdad de a puño, que es tan nuestra como de Joaquín Salvador Lavado (*Quino*) en su construcción gramatical *"A lo largo del camino hay que hacerse a más y mejores instrumentos conceptuales y estéticos"*.

La paraconsistencia la entenderemos en un sentido muy amplio, como todo aquello que tiene que ver con las contradicciones. Aunque particularmente nos centraremos en aquellas lógicas que permiten contradicciones sin trivializarse.

Ya en este inicio, se hace necesario hacer un llamado obligado a entender, ó por lo menos aceptar, que nos "saldremos" de los marcos paradigmáticos de nuestra cultura occidental que sentencian prohibir las contradicciones. Se trata de abrir una ventana que nos permita observar los mágicos y sorprendentes aportes de la lógica desde los años 60's del siglo XX, que contrasta con el absoluto aristotélico de los últimos 2.400 años.

1.1. Breve recorrido histórico

A fin de contrastar el absoluto aristotélico de los últimos 2.400 años con los últimos 50 años del desarrollo de la lógica, es pertinente hacer un breve recorrido histórico por la sistematización del razonamiento que data del año 400 a.c. con los pueblos griegos, árabes, hindúes y chinos.

En Occidente, el precursor es Aristóteles; aunque hoy conocemos la existencia de trabajos paralelos por parte de los árabes, hindúes y chinos, algunos de los cuales a pesar de su vigencia, seguimos ignorándolos. Digamos que Aristóteles es el primer filósofo europeo en hacer de los métodos del razonamiento un objeto de estudio.

En la cultura romana, nos encontramos con *los comentadores*, que transmiten los textos griegos a la edad media. Emerge como mayor representante Boecio (480-524).

En los siglos VII y VIII, con *la escolástica* aparecen las academias; y en los siglos IX y X, la sobresaliente escuela de Bagdad. El representante más conocido Pedro Abelardo (1079-1142) comentarista de Aristóteles. Porfirio y Boecio.

En el siglo XII, aparecen *las primeras universidades*, y con ellas los textos de Lógica de Guillermo de Shyreswood y Pedro Hispano.

En los siglos XVI y XVII, figuras como Descartes (1596-1650) y Leibniz (1646-1716) dan cuenta de *el ideal Luliano:* que en cada teoría basta un número finito de principios básicos, a partir de los cuales con reglas claras de inferencia se da cuenta de la totalidad del conocimiento.

En el siglo XIX, Bolzano, Boole y Cantor inician el estudio de *los fundamentos de la Matemática* que pasa a Frege, Peano, Russell, Brouwer, Whitehead y Hilbert. Este último es el máximo representante de los formalistas, que buscaban la depuración del método axiomático; creyeron que era posible realizar en las matemáticas el sueño Luliano de Descartes y Leibniz. Pero en 1931 Gödel, con su famoso *Teorema de Incompletitud,* echa por tierra tal sueño.

Es obligado referenciar la magistral obra *From Frege to Gödel: a Source Book in Matehematical Logic,* (1879 - 1931) de *Jean van Heijenoort*, en la cual se consignan los artículos originales traducidos al inglés que dan cuenta exacta del proceso de los desarrollos de los fundamentos de la matemática que hoy nos rige. Señalemos que en esta obra se incluyen artículos tan relevantes (la verdad todos lo son) como el de Kolmogorov (1925) que anticipa la formalización del intuicionismo de Heyting; el de Finsler (1926) que antecede las ideas fundamentales del teorema de incompletitud de Gödel y el de Herbrand (1930) que incluye el pilar fundamental del principio de resolución de Robinson, y por ende el pilar de la demostración automática con sus múltiples aplicaciones a través de la programación lógica que llegan hasta nuestros días.

En esta época de discusión acerca de los fundamentos, tenemos situados los antecedentes de las lógicas paraconsistentes. Inicialmente con los trabajos de Lukasiewicz (1910), quien afirma que la exigencia de consistencia, *el principio de no contradicción,* no es tan fundamental; y también los tan importantes aportes de Vasiliev (1910-1913) quién propone derogar *las leyes de contradicción y el tercero excluido.* Después, en 1948, Jaskowski propone el primer cálculo proposicional discursivo. Sin embargo en 1953 un joven matemático Argentino Florencio González Asenjo, ahora Profesor Emerito de la Universidad de Pittsburgh, presentó en la Universidad de la Plata, en Argentina, una conferencia titulada *La idea de un cálculo de antinomias.* Asenjo desarrolló una teoría sobre la inconsistencia en esa década, pero sus primeras publicaciones aparecen entre 1965 y 1966 ([10]), y en 1958, de forma independiente, Newton da Costa (el hoy denominado verdadero padre de la lógica paraconsistente) presenta una serie de Cálculos Proposicionales Paraconsistentes, Cálculos de Predicados Paraconsistentes y Teorías de Conjuntos Paraconsistentes.

El término *paraconsistente* fue acuñado por el filósofo peruano Francisco Miró Quesada, a petición de Newton da Costa, en el Tercer Simposio Latinoamericano de Lógica Matemática en Campinas Brasil (1976). Ahora, gracias a Decio Krause la lógica paraconsistente tiene su propia sección **03B53** en los *Mathematic Reviews.*

Recientemente se han celebrado una serie de importantes congresos mundiales especializados en el área, como por ejemplo *El primer congreso mundial sobre paraconsistencia*, realizado entre Julio 30 y Agosto 2 de 1997 que se celebró en la Universidad de Ghent, Bélgica, con cerca de 100 expositores de 30 países. Tratando temas variados como Fundamentos Filosóficos, Teorías de Pruebas Formales, Teorías de Modelos, Aplicaciones en Matemáticas y Ciencias de la Computación. Este evento dio lugar a la publicación del libro *Frontiers of Paraconsistent Logic 2000,* y a un número especial de la revista *Logique et Analyse.*

Resultados análogos se consiguieron en el segundo congreso mundial sobre paraconsistencia, realizado en Sao Paulo (Brasil) 2000, así como en los Workshop del verano europeo (Italia 2002), y el III World Congress on Paraconsistency Toulouse, France 2003.

1.2. Paraconsistencia

El término paraconsistencia es usado en un sentido muy amplio, prácticamente concierne a todo aquello que tiene que ver con las contradicciones. Es decir, la paraconsistencia concierne a todos aquellos estudios filosóficos, históricos y científicos relacionados con las contradicciones.

La característica fundamental de la paraconsistencia es proveer teorías inconsistentes no triviales. En la búsqueda de este objetivo es necesario desarrollar lógicas subyacentes que sean tolerantes a ciertas inconsistencias. A este tipo de lógicas es las que llamaremos paraconsistentes.

En esta apreciación pensaremos la inconsistencia como un fenómeno primario, mientras que la paraconsistencia es derivado. Por este motivo el término "paraconsistencia" califica una propiedad de la lógica y no de la teoría, así como el término "inconsistente" califica la teoría y no a la lógica. Sin embargo, es usual encontrar en la literatura errores en ambas direcciones. Se puede concluir que las lógicas paraconsistentes exhiben formalmente maneras de "controlar" ciertas contradicciones.

En un marco muy general es posible asegurar que Occidente ha consolidado sus versiones éticas, conceptuales y estéticas a partir de la no contradicción, dentro de un marco regido por la lógica clásica, que incluye la no contradicción como uno de los pilares fundamentales.

Este hecho, sin ser en un todo responsable, tiene singulares implicaciones cuando se pasa a los *discursos de poder*. Citemos dos ejemplos: primero el del presidente Bush, después del 11 de septiembre del 2001, cuando afirma "el que no está con nosotros está en contra nuestra". Muy clásica la lógica subyacente en este discurso, una lógica de sólo dos posibilidades. Acaso, ¿no es posible que no estemos con ellos, sin estar en su contra? El segundo, que realmente es el mismo en otro contexto, el de los agentes protagónicos del conflicto armado en Colombia donde se asesina a la población civil por no estar a favor de un bando, pues esto los "hace" del otro.

Siendo románticos, estamos seguros de que este mundo sería diferente, y de seguro mejor, si la lógica subyacente al discurso genérico fuese distinta de la clásica. Más aún sea esta una invitación a adoptar una que sea paraconsistente.

En la lógica clásica, de una contradicción se deducen todas las proposiciones, es decir, si al modelar cualquier teoría con la lógica clásica, aparece una contradicción, entonces se infieren todas las proposiciones de tal teoría. Este hecho usualmente se abrevia diciendo que la teoría se vuelve trivial. Así es comprensible el pavor a las contradicciones, y el afán por eliminarlas.

En la lógica clásica lo contradictorio se funde con lo inconsistente, lo que no permite distinguirlos.

Queremos cuestionar a la lógica clásica el hecho de imponer la consistencia como requisito para desarrollar teorías no triviales (en las que no toda fórmula del lenguaje es derivable de los axiomas de la teoría).

Pasemos brevemente a la inteligencia artificial, donde es frecuente que en una base de conocimiento se tenga alguna inconsistencia "local", que bien podría considerarse "irrelevante" en vista del conjunto total de información contenida. ¿Que es lo razonable? Seguir sacando conclusiones interesantes. Pero la lógica clásica la vuelve trivial. Más específicamente, en consultas técnicas con distintos expertos éstos suelen no coincidir sobre un mismo aspecto del conocimiento del dominio. Por ejemplo en una base de datos de diagnóstico médico, se encuentra con la siguiente información; a partir de síntomas observados, dos distintos especialistas consignan opiniones contrarias. Obviamente no hay que eliminar la base de datos, ni tampoco dichas opiniones; por el contrario, es muy importante para el paciente en cuestión, para quizás consultar una tercera opinión. Estos ejemplos ilustran la justificación para desarrollar nuevas lógicas donde no aparezca el límite de la no contradicción, al menos en un sentido tan fuerte.

Las lógicas paraconsistentes son aquellas que rechazan el principio de no contradicción, es decir, admiten contradicciones verdaderas sin ser triviales.

Más aún, en algunas lógicas paraconsistentes existen teoremas de la forma

$$\alpha \wedge \neg\alpha,$$

sin que de éstos se infieran todas las fórmulas, hecho que se contrapone al caso de la lógica clásica (y de la intuicionista por mencionar otra) en la que a partir de un teorema de la forma $\alpha\wedge\neg\alpha$ se deducen todas las fórmulas. Las lógicas paraconsistentes toleran razonablemente las contradicciones.

1.3. Lógicas Paraconsistentes

Aclaremos, a nuestro parecer, cual sería la acepción correcta del término *paraconsistente.*

Señalemos las distintas acepciones que tiene el prefijo *para.*

1. *Contra* como en paradoja (En contra del sentido común).
2. *Más allá de* como en paranormal.
3. *Muy similar* como en paramilitar.

Es así que tal vez esta última acepción sea la más indicada en el significado del prefijo *para* dentro del término paraconsistente[1].

Insistamos, en que el problema fundamental de modelar teorías inconsistentes en la lógica clásica es que en ésta de una contradicción se deduce cualquier proposición, hecho que explica el pavor que las contradicciones generan, pues conducen a la trivialización. Por tanto es natural que en este contexto sea indispensable eliminarlas.

Formulemos la pregunta obligada: ¿es posible desarrollar una lógica que tolere razonablemente las contradicciones? Es decir, en la cual éstas resulten inofensivas, o

[1]Es oportuno mencionar que enel II WCP se mencionó como tal vez más apropiado el término "parainconsistente" para las lógicas hoy conocidas como "paraconsistentes".

al menos poco peligrosas. La pregunta tiene respuesta afirmativa, pues la creación de las lógicas paraconsistentes así lo garantiza.

A fin de precisar un poco más acerca de las lógicas paraconsistentes, presentaremos, a manera de ilustración, las más conocidas: aquellas desarrolladas por Newton da Costa. En rasgos generales sus fundamentos obedecen a los siguientes lineamentos:

1. El principio de no contradicción, $\neg(\alpha \wedge \neg\alpha)$, en general, no debe ser válido.
2. De dos fórmulas contradictorias, α y $\neg\alpha$, en general, no debe existir la posibilidad de deducir una fórmula arbitraria.
3. Mantener los esquemas y reglas de deducción del cálculo clásico que no interfieran con las dos condiciones anteriores.

1.3.1. Axiomas y reglas

$[\rightarrow_1\ ]$ $A \rightarrow (B \rightarrow A)$

$[\rightarrow_2\ ]$ $(A \rightarrow B) \rightarrow ((A \rightarrow (B \rightarrow C)) \rightarrow (A \rightarrow C))$

$[\rightarrow_3\ ]$ A, $A \rightarrow B$ / B

$[\wedge_1\ ]$ $A \wedge B \rightarrow A$

$[\wedge_2\ ]$ $A \wedge B \rightarrow B$

$[\wedge_3\ ]$ $A \rightarrow (B \rightarrow (A \wedge B))$

$[\vee_1\ ]$ $A \rightarrow A \vee B$

$[\vee_2\ ]$ $B \rightarrow A \vee B$

$[\vee_3\ ]$ $(A \rightarrow C) \rightarrow ((B \rightarrow C) \rightarrow (A \vee B \rightarrow C))$

$[\neg_1\ ]$ $B^0 \rightarrow ((A \rightarrow B) \rightarrow ((A \rightarrow \neg B) \rightarrow \neg A))$

$[\neg_2\ ]$ $A^0 \wedge B^0 \rightarrow ((A \rightarrow B)^0 \wedge (A \wedge B)^0 \wedge (A \vee B)^0)$

$[\neg_3\ ]$ $A \vee \neg A$

$[\neg_4\ ]$ $\neg\neg A \rightarrow A$

$[\neg_{(n),1}\ ]$ $B^{(n)} \rightarrow ((A \rightarrow B) \rightarrow ((A \rightarrow \neg B) \rightarrow \neg A))$

$[\neg_{(n),2}\ ]$ $A^{(n)} \wedge B^{(n)} \rightarrow ((A \rightarrow B)^{(n)} \wedge (A \wedge B)^{(n)} \wedge (A \vee B)^{(n)})$

$[\{\neg_{(n),1}\}^{-}]$ $(A \to B) \to ((A \to \neg B) \to \neg A)$

Donde :

- $A^0 = \neg(\alpha \wedge \neg\alpha)$
- $A^1 = A^0$
- $A^n = A^{000\ldots0}$ (n-veces).
- $A^{(1)} = A^1$
- $A^{(n)} = A^1 \wedge A^2 \wedge \ldots \wedge A^n$

La lógica intuicionista positiva, que abreviaremos *LIP*, está determinada por $\to_{1,2,3}$ $\wedge_{1,2,3}$ y $\vee_{1,2,3}$, mientras que:

- $C_\omega = LIP + \neg_3, \neg_4$
- $C_0 = C_\omega + \{\neg_{(n),1}\}^{-}$ que corresponde a la Lógica Proposicional Clásica.
- $C_1 = C_\omega + \neg_1, \neg_2$
- $C_n = C_\omega + \neg_{(n),1}, \neg_{(n),2}$

Intuitivamente, con el símbolo "bolita" se quiere representar las fórmulas que tienen un buen comportamiento, las que tienen un comportamiento clásico. Si una fórmula es clásica (tiene un buen comportamiento) es razonable pensar que no se puede tener ella y su negación. El axioma $\neg_1$ corresponde a *reductio ad absurdum* para formulas clásicas.

Teorema 1.1 *En C_n se cumplen todos los teoremas válidos en LIP, para todo n.*

Teorema 1.2 *En C_1 si $\gamma, \alpha \vdash \beta^0, \beta, \neg\beta$ entonces $\gamma \vdash \neg\alpha$.*

Teorema 1.3 *En C_1 no son teoremas, entre otros, los esquemas:*

- $\alpha \to (\neg\alpha \to \beta)$
- $\neg\alpha \to (\alpha \to \beta)$
- $\alpha \wedge \neg\alpha \to \beta$
- $\neg(\alpha \wedge \beta) \leftrightarrow \neg\alpha \vee \neg\beta$
- $\neg\neg\alpha \leftrightarrow \alpha$

Teorema 1.4 *Si a C_n se agrega el esquema $\neg(\alpha \wedge \neg\alpha)$ como un nuevo axioma se obtiene el cálculo proposicional clásico.*

Teorema 1.5 *En C_1 se cumple que*

- $\vdash ((\alpha \rightarrow \beta) \rightarrow \alpha) \rightarrow \alpha$
- $\vdash \alpha^{00}$
- $\vdash \alpha^0 \rightarrow (\neg\alpha)^0$

Teorema 1.6 *(Arruda) C_1 no es decidible por matrices finitas.*

Teorema 1.7 *(Fidel) C_1 es decidible.*

Teorema 1.8 *Si $\neg^*\alpha$ representa a $\neg\alpha \wedge \alpha^0$, entonces $\neg^*$ cumple todas las propiedades de la negación clásica en C_n.*

Después de este rápido recorrido por las lógicas de Da Costa, creemos que es muy pertinente señalar, incluyendo su prueba, el siguiente resultado que establece técnicamente la heterodoxia de las lógicas de Da Costa.

Teorema 1.9 *No existe una extensión de C_n $(1 \leqslant n < \omega)$ tal que satisfaga la propiedad de sustitución y sea más débil que C_0.*

Demostración

Recordemos *la propiedad de sustitución:* si B es una subfórmula de un teorema A y $C \leftrightarrow B$ entonces al sustituir B por C en algunas o todas las ocurrencias de B en A se obtiene un teorema.

La idea de la prueba es suponer que la propiedad de sustitución se cumple en C_n y colapsarlo a C_0. Para esto es suficiente probar $\vdash B^{(n)}$ ya que por $\neg_{(n),1}$ se obtendrá $\{\neg_{(n),1}\}^-$ y así C_0.

En este camino busquemos una fórmula ϕ tal que $B \leftrightarrow \phi$ y $\vdash \phi^{(n)}$ con lo cual aplicando sustitución se tiene $\vdash B^{(n)}$.

Particularizando $B = C$ en $\vee_3$ se tiene $(A \rightarrow C) \rightarrow ((C \rightarrow C) \rightarrow (A \vee C \rightarrow C))$. Con lo cual si tenemos A tal que $\vdash A \rightarrow C$ tendremos de inmediato $\vdash A \vee C \rightarrow C$.

Sea C la fórmula $B \leftrightarrow ((B \rightarrow A) \rightarrow A)$, observemos que en la fórmula siguiente, $(A \vee (B \leftrightarrow ((B \rightarrow A) \rightarrow A))) \rightarrow (B \leftrightarrow ((B \rightarrow A) \rightarrow A))$ el antecedente es una tautología, y así se tiene el consecuente $B \leftrightarrow ((B \rightarrow A) \rightarrow A)$. Por tanto un buen candidato para ϕ es la fórmula $(B \to A) \to A$, siempre que podamos hallar A tal que $\vdash A \to C$.

Un posible A es $B^{(n)} \wedge (B \wedge \neg B)$. Tomando en $\neg_{(n),1}$ el lugar de A por $\neg C$ se obtiene que $B^{(n)} \to ((\neg C \to B) \to ((\neg C \to \neg B) \to \neg\neg C))$, por permutación de antecedentes

$$\frac{C \to (D \to E)}{D \to (C \to E)}$$

regla válida en LIP y por tanto en C_n, se tiene que: $(\neg C \to B) \to ((\neg C \to \neg B) \to (B^{(n)} \to \neg\neg C))$. Pero por $\to_1$ se tiene además que: $B \to (\neg C \to B)$. De donde por transitividad $B \to ((\neg C \to \neg B) \to (B^{(n)} \to \neg\neg C))$.

Usando permutación de antecedentes nuevamente, tenemos como resultado que $(\neg C \to \neg B) \to (B \to (B^{(n)} \to \neg\neg C))$, y usando además el axioma $\to_1$ obtenemos como resultado que $\neg B \to (\neg C \to \neg B)$. Así $B \to (\neg B \to (B^{(n)} \to \neg\neg C))$.

De donde, aplicando importación

$$\frac{C \to (D \to E)}{C \wedge D \to E}$$

obtenemos

$(B \wedge \neg B) \to (B^{(n)} \to \neg\neg C)$.

De nuevo por importación se obtiene que

$(B \wedge \neg B) \wedge B^{(n)} \to \neg\neg C$, que junto a $\neg_4$ y la transitividad nos permite concluir que efectivamente $\vdash A \to C$.

Ahora veamos que: $\vdash \phi^{(n)}$, con $\phi = (B \to A) \to A$ y $A = B^{(n)} \wedge (B \wedge \neg B)$.

Considerando $\vee_3$ con $C = A^{(n)}$ tenemos: $(A \to A^{(n)}) \to ((A^{(n)} \to A^{(n)}) \to (A \vee A^{(n)} \to A^{(n)}))$, donde el primer antecedente se tiene por la argumentación anterior, y el segundo antecedente es un teorema básico. Es decir, nuestro objetivo se conseguirá sustentando el tercer antecedente.

Veamos que:

$\neg A^i$	$= \neg\neg(A^{i-1} \wedge \neg A^{i-1})$
por $\neg_4$	$\neg\neg(A^{i-1} \wedge \neg A^{i-1}) \to (A^{i-1} \wedge \neg A^{i-1})$
por $\wedge_2$	$(A^{i-1} \wedge \neg A^{i-1}) \to A^{i-1}$

Con lo cual usando inducción se garantiza: $\neg A^i \to \neg A^0$ para todo i, $1 \leqslant i < \omega$. Es decir, $\neg A^i \to \neg(A \wedge \neg A)$.

De otra parte:

por $\neg_4$	$\neg\neg(A \wedge \neg A) \to (A \wedge \neg A)$
por $\wedge_1$	$A \wedge \neg A \to A$

Lo que, usando transitividad, nos permite concluir $\neg A^{i-1} \to A$, que junto a $\neg_3$ garantiza $A \vee A^i$.

Así se obtiene $(A \vee A^i) \wedge (A \vee A^{i-1}) \wedge \ldots \wedge (A \vee A^0)$, por tanto se tiene $A \vee (A^i \wedge A^{i-1} \wedge \ldots \wedge A^0)$, y en correspondencia se tiene $A \vee A^i$.

En LIP es válido el esquema $(D \to E) \to ((C \to D) \to (C \to E))$, así tenemos que $(A \to B^{(n)}) \to ((B \to A) \to (B \to B^{(n)}))$. El antecedente es la condición inicial (ya probada), luego permutando antecedentes obtenemos $B \to ((B \to A) \to B^{(n)})$, que junto a $\to_2$, y $\vee_3$ nos permite concluir $(B \vee B^{(n)}) \to ((B \to A) \to B^{(n)})$.

Como además teníamos $(B \vee B^{(n)})$ se obtiene $(B \to A) \to B^{(n)}$.

De otro lado, usando $\wedge_3$, la condición probada inicialmente, $\neg_{(n),2}$ y transitividad se obtiene $B^{(n)} \to (B \to A)^{(n)}$.

Luego tenemos el esquema: $(B \to A) \to (B \to A)^{(n)}$. Tomando $B = C \to A$ se tiene (i) $((C \to A) \to A) \to ((C \to A) \to A)^{(n)}$.

Por otro lado $(B \to A) \to B^{(n)}$ es un esquema, tomando $B = (C \to A) \to A$ se obtiene (ii) $(((C \to A) \to A) \to A) \to ((C \to A) \to A)^{(n)}$.

De $(i), (ii)$ y $\vee_3$ tenemos que

$$(((C \to A) \to A) \vee ((((C \to A) \to A)) \to A)) \to ((C \to A) \to A)^{(n)},$$

cuyo antecedente es una tautología y así finalizamos la prueba concluyendo que

$$((C \to A) \to A)^{(n)} \quad \blacksquare$$

En correspondencia con la jerarquía de los cálculos proposicionales $C_n, 0 \leqslant n \leqslant \omega$ Da Costa presenta los cálculos de predicados de primer orden $C_n^*, 0 \leqslant n \leqslant \omega$. Estos se obtienen de los C_n respectivos agregando los siguientes axiomas y reglas (sujetas a las restricciones usuales):

[$\forall_1$] $\forall x A(x) \to A(t)$

[$\forall_2$] $\frac{A \to B(x)}{A \to \forall x B(x)}$

[$\forall_3$] $\forall x (A(x))^{(n)} \to (\forall x A(x))^{(n)}$

[$\exists_1$] $A(t) \to \exists x A(x)$

[$\exists_2$] $\frac{A(x) \to B}{\exists x A(x) \to B}$

[$\exists_3$] $\forall x(A(x))^{(n)} \rightarrow (\exists x A(x))^{(n)}$

La jerarquía de los cálculos de predicados de primer orden $C_n^*, 0 \leqslant n \leqslant \omega$ tiene una jerarquía de cálculos de predicados de primer orden con igualdad $C_n^=, 0 \leqslant n \leqslant \omega$ incrementando los axiomas (sujetas a las restricciones usuales):

[$=_1$] $x = x$

[$=_2$] $x = y \rightarrow (A(x) \leftrightarrow A(y))$

Teorema 1.10 *(Béziau)*
Si denotamos

- $T : A \vee \neg A$
- $\varepsilon : (A \wedge \neg A) \rightarrow (\neg(A \wedge \neg A) \rightarrow B)$
- $\alpha : B^0 \rightarrow ((\neg A \rightarrow B) \rightarrow ((\neg A \rightarrow \neg B) \rightarrow A))$
- $\beta : (\neg^* A \rightarrow B) \rightarrow ((\neg^* A \rightarrow \neg^* B) \rightarrow A)$

entonces los sistemas $LIP + T, LIP + \varepsilon, LIP + \alpha$, y $LIP + \beta$ son equivalentes.

A la lógica obtenida en uno de los cuatro casos señalados en la equivalencia anterior la denotaremos por C_i, y agregando los axiomas:

- $A^0 \rightarrow (\neg A)^0$
- $A^0 \rightarrow (A \wedge B)^0$
- $B^0 \rightarrow (A \wedge B)^0$
- $A^0 \rightarrow (A \vee B)^0$
- $B^0 \rightarrow (A \vee B)^0$
- $A^0 \rightarrow (A \rightarrow B)^0$
- $B^0 \rightarrow (A \rightarrow B)^0$

se obtiene C_1^+.

Teorema 1.11 *Los siguientes esquemas se cumplen en C_1^+, pero no en C_1.*

- $\neg(A \vee B) \rightarrow (\neg A \wedge \neg B)$

- $\neg(A \vee \neg B) \rightarrow (\neg A \wedge B)$
- $\neg(\neg A \vee \neg B) \rightarrow (A \wedge B)$
- $\neg(\neg A \vee B) \rightarrow (A \wedge \neg B)$

Mientras que la lógica C_1 no es *simple*, es decir, no se puede definir en ella una relación de congruencia no trivial (Mortensen), la lógica C_1^+ si es simple. La congruencia no trivial está dada por la equivalencia lógica en dicho sistema, agregando la condición de buen comportamiento a las fórmulas equivalentes. A esta equivalencia la denominan *regular*. Formalmente, dos fórmulas A y B se dicen regularmente equivalentes si en el sistema C_1^+ se demuestra que $A \rightarrow B$, $B \rightarrow A$, A^0 y B^0.

Análogamente que para el caso de C_1, la lógica C_1^+ tiene su respectiva lógica de primer orden C_1^{*+} agregando los axiomas $\forall_1, \forall_2, \exists_1, \exists_2$ antes mencionados y

[$\forall_3^+$] $\exists x(A(x))^0 \rightarrow (\exists x A(x))^{(n)}$

[$\exists_3^+$] $\exists x(A(x))^{(n)} \rightarrow (\forall x A(x))^{(n)}$

Teorema 1.12 *Los siguientes esquemas se cumplen en C_1^{*+}, pero no en C_1^*.*

- $\neg(\exists A(x)) \rightarrow (\forall \neg A(x))$
- $\neg(\exists \neg A(x)) \rightarrow (\forall A(x))$

1.4. Teorías de conjuntos paraconsistentes.

Mencionaremos la idea general de cómo se construyen teorías de conjuntos paraconsistentes, lo cual se convierte en un ejemplo relevante, dado que ilustra el uso técnico de las lógicas paraconsistentes, y además nos permite dar un cierto contexto a las mismas.

En términos generales, la escolaridad nos ha brindado un acercamiento, no formal en la mayoría de los casos, a las denominadas *lógica proposicional y lógica de predicados ó lógica de primer orden.*

En particular, en matemáticas, dentro de los cursos básicos de Teoría de Conjuntos se precisa de la reducción de muchas propiedades a la lógica proposicional y de primer orden, las cuales sustentan la verdad, sin discusión alguna, de cada una de las proposiciones en cuestión. Por ejemplo las propiedades asociativas de la unión e intersección de conjuntos se reducen a las propiedades asociativas de la disjunción y la conjunción de la lógica proposicional.

Por lo general no se hace una argumentación amplia y/o explícita que muestre que el desarrollo de esta teoría de conjuntos particular se soporta en la lógica clásica. Esta lógica exige entre otras cosas, *expulsar* conjuntos tipo *Russell;* conjuntos que se pertenecen a sí mismos y que conllevan consigo las dos proposiciones: una que afirman su pertenencia a sí mismos, y la otra que niega tal pertenencia, lo cual les hace conjuntos contradictorios. También se prohíbe la regresión al infinitum a través del axioma de regularidad.

En contraste, cuando se usa una lógica paraconsistente subyacente (como las desarrolladas por Newton da Costa) para desarrollar teorías de conjuntos, cercanas a las conocidas, aparecen unos cimientos más amplios para el edificio matemático. Por ejemplo, en éstas es importante resaltar que *no hay necesidad de expulsar* los conjuntos tipo *Russell* a pesar de ser contradictorios. También es justo mencionar que existen desarrollos de teorías de conjuntos con un axioma de regularidad débil.

En [30] Da Costa et al. se usa la lógica $C_1^=$ para el estudio de teorías de conjuntos paraconsistentes. Primeramente enriquecen el lenguaje con un descriptor tipo Hilbert, proceso por demás contemplado en la lógica clásica, con lo cual definen formalmente dentro de su lenguaje el conjunto de Russell. Además al tenerse dentro de la lógica una negación fuerte se puede incluir dentro de ésta una distinción interesante, ilustremos con el siguiente ejemplo:

- $x \notin y$ para abreviar $\neg(x \in y)$
- $x \notin^* y$ para abreviar $\neg^*(x \in y)$
- $\emptyset$ es definido por $\{x : x \neq x\}$
- $\emptyset^*$ es definido por $\{x : x \neq^* x\}$

Obsérvese que existen dos conjuntos vacíos. Más sin embargo, la noción de que x es un conjunto no vacío puede ser expresada por la fórmula $\exists y(y \in x)$.

Es pertinente que en estas teorías se estudian una serie de propiedades especiales que tiene el conjunto de Russell, como que el conjunto de partes sea un subconjunto de él mismo, que él sea infinito, más aún que su cardinal es fuertemente inaccesible, entre otras.

Además se muestran pasos iniciales que dan pie a una matemática paraconsistente. En este orden de ideas la matemática se reduce, en cierto sentido, a investigar teorías deductivas sobre una lógica dada. Es pertinente resaltar que una deducción, clásica o heterodoxa, no conlleva nada que no se contenga en sus hipótesis iniciales o en su lógica. Luego los sistemas paraconsistentes contienen contradicciones, por que éstas están en sus axiomas, o en la lógica subyacente utilizada. En forma particular, se desarrolla una geometría paraconsistente correspondiente a la geometría clásica del plano afín finito, pero sobre una teoría de conjuntos paraconsistente, con lógica subyacente $C_1^=$.

1.5. Racionalidad

En 1912 el médico ruso Vasiliev pensaba que en toda lógica hay dos niveles de juicio. uno *metalógico* donde residen las leyes del pensamiento, las que no pueden ser derogadas sin que el sistema deje de ser lógico, y un segundo *ontológico*, en donde están las leyes que dependen de la naturaleza de los objetos. A este último pertenecería el principio de no–contradicción, entendido como "un objeto no puede satisfacer o verificar un predicado que lo contradiga". Sin embargo el principio de no–auto–contradicción; "un juicio no puede ser simultáneamente verdadero y falso"pertenece al nivel metalógico y no puede eliminarse. Los juicios del nivel ontológico hablan sobre hechos, los juicios del nivel metalógico hablan sobre conceptos, que expresan leyes atemporales.

Según Vasiliev, hay sólo tres tipos de juicio sobre conceptos. "Todo S es P," "Ningún S es P" y "Algunos S, pero no todos, son P, y los otros son no–P". Luego observa que dadas dos de ellas no pueden ser ambas verdaderas, pero sí pueden ser ambas falsas.[2] Se puede entonces formular un principio del *cuarto excluido* para los juicios sobre conceptos como sigue: Dado un objeto y un predicado podemos formular tres juicios, uno sobre la necesidad del predicado para ese objeto, otro para la imposibilidad de el mismo y un tercero sobre su posibilidad, pero no hay una cuarta opción.

Por otra parte, para los juicios sobre hechos vale la ley del tercero excluido. Estos comentarios hacen de Vasiliev también un precursor de la lógica multivaluada. Debe observarse sin embargo que en ningún momento se ha planteado la posibilidad de un tercer valor de verdad.

Aunque Vasiliev no propuso ningún sistema formal propiamente, hoy tenemos varios sistemas que intentan formalizar estas ideas (Arruda [8]). En ellos hay una cantidad enumerable de letras proposicionales clásicas y un conjunto no–vacío S de letras proposicionales de Vasiliev. El sistema V_1 está dado por el cálculo intuicionista positivo más el principio del tercero excluido, además de:

$$\varphi \to (\neg\varphi \to \psi) \quad \textit{siempre} \quad \textit{que} \quad \varphi \notin S.$$

[2]En la terminología de Aristóteles, ellas son contrarias dos a dos. Vasiliev construye un "triángulo de oposiciones"similar al clásico cuadrado de oposiciones aristotélico.

Hay distinción de dos tipos de letras proposicionales, con axiomas que rigen para unas pero no para las otras, estos sistemas no están cerrados bajo sustituciones.

El sistema V_2 se obtiene de V_1 agregando el axioma válido sólo para letras de Vasiliev:

$$p \wedge \neg p$$

En contraste, el sistema V_3 tiene un solo tipo de letra proposicional, pero consta de dos operaciones que sólo se aplican a las letras proposicionales, de esta forma nuevamente, el sistema no es cerrado bajo sustituciones.

Arruda propone una semántica para la cual los sistemas son correctos y completos. Un método de tablas de verdad de 2 valores (para V_1) y de tres valores (para V_2 y V_3) generaliza el método clásico.

De otro lado Jaśkowski, alumno de Lukasiewicz, estaba motivado por varios problemas relacionados con las contradicciones, en particular la sistematización de teorías, como la dialéctica, que contiene contradicciones, el estudio de teorías con contradicciones causadas por la vaguedad y el estudio de argumentos convincentes que llevan a conclusiones contradictorias, tales como aquellos que surgen en una discusión. Este es el origen del nombre de lógica *discursiva*.

En una discusión, los distintos participantes hacen afirmaciones usando términos con significados vagos o en distinta acepción. Estos son empleados y combinados como si pertenecieran a un cálculo deductivo. Naturalmente sus consecuencias no pueden ser tomadas como una opinión uniforme y deben considerarse más bien como una posibilidad. La aparición de contradicciones no debería extrañar a nadie. Las leyes de esta lógica no pueden ser las mismas que las leyes clásicas. Un ejemplo de esto es que la ley de adjunción, $\varphi, \psi \vdash \varphi \wedge \psi$, no es válida. En efecto, si las afirmaciones φ y ψ han sido hechas por distintos interlocutores, entonces no podemos inferir $\varphi \wedge \psi$, ya que ésta última podría no haber sido dicha por nadie.

Jaśkowski quería un sistema que 1) no se trivializara al aplicarlo a contradicciones, 2) suficientemente rico como para poder tener inferencias correctas, 3) que tuviera una justificación intuitiva. El presenta una solución, el sistema D_2, para un lenguaje proposicional dentro de la lógica modal S5. La siguiente es una axiomatización simplificada de D_2 debida a da Costa y Dubikajtis. Ver [33, 34].

Axiomatización de D_2,

J1 $\quad \Box((\varphi \rightarrow \psi) \rightarrow ((\psi \rightarrow \vartheta) \rightarrow (\varphi \rightarrow \vartheta)))$,

J2 $\quad \Box(\varphi \rightarrow (\neg\varphi \rightarrow \psi))$, J3 $\quad \Box((\neg\varphi \rightarrow \varphi) \rightarrow \varphi)$,

J4 $\quad \Box(\varphi \rightarrow (\varphi \vee \psi))$,

J5 $\quad \Box(\psi \rightarrow (\varphi \vee \psi))$,

J6 $\quad \Box((\varphi \rightarrow \vartheta) \rightarrow ((\psi \rightarrow \vartheta) \rightarrow ((\varphi \vee \psi) \rightarrow \vartheta)))$,

J7 $\quad \Box(\Box(\varphi \rightarrow \psi) \rightarrow \Box(\Box\varphi \rightarrow \Box\psi))$,

J8 $\quad \Box(\Box\varphi \rightarrow \varphi)$,

J9 $\quad \Box(\varphi \rightarrow \Box\Diamond\varphi)$,

R1 $\quad \dfrac{\varphi\,,\ \Box(\varphi \rightarrow \psi)}{\psi}$

R2 $\quad \dfrac{\Diamond\varphi}{\varphi}$

Donde los operadores $\Diamond$ y $\Box$ son los usuales operadores modales unarios. Estos son duales, es decir, $\Diamond\varphi$ y $\neg\Box\neg\varphi$ son equivalentes. La interpretación intuitiva para $\Diamond\varphi$ es "alguien afirma que φ".

Es conveniente tener en cuenta que hay otra axiomatización, también debida a da Costa y Dubikajtis, que no recurre a operadores modales.

1.6. La racionalidad de la lógica

Si hay algo que podamos denominar racional es la relación de consecuencia lógica.

Las investigaciones modernas han establecido que la lógica clásica no es adecuada para la formalización de la relación de consecuencia, pues esta lógica no captura ciertas intuiciones. Por ejemplo, que entre las premisas y la conclusión debe haber una relación que permita dar el paso deductivo, lo que hoy se conoce como *relevancia*. Vemos que en el teorema clásico $(\alpha \wedge \neg\alpha) \rightarrow \beta$, no hay ninguna razón que soporte la deducción de β a partir de $\alpha \wedge \neg\alpha$, pues no hay relación entre las premisas y la conclusión.

La manera como funciona nuestra razón es distinta a la que tradicionalmente se supuso. Es tan compleja y profunda que no cabe dentro de los esquemas clásicos. La razón funciona de una manera mucho más amplia y flexible de lo que creyeron los filósofos del conocimiento antes de los recientes desarrollos lógicos.

Asumiendo que la lógica es una disciplina racional y que su desarrollo busca conocimientos bien fundamentados en un contexto universal, podemos asegurar contrario a como se creyó durante mucho tiempo que la lógica (como tal) no es un sistema racionalmente perfecto. Los desarrollos lógicos de sistemas paraconsistentes dan cuenta del esfuerzo por dar sistemas deductivos más racionales.

La racionalidad de la lógica se basa en la posibilidad de efectuar deducciones. La relación de consecuencia constituye el eje de la racionalidad a nivel lógico. Por tanto, el ámbito de la lógica es mucho más amplio de lo que se supuso en el pasado, hoy se pueden establecer consecuencias lógicas dentro de sistemas paraconsistentes.

Del idioma inglés se ha usado la palabra "deviant"para hacer referencia a aquellas lógicas que se "alejan"de la lógica clásica. Pero dada la imprecisión de una traducción directa, como lo es la palabra "desviante", acompañamos al filósofo Miró Quesada [61], en sus precisiones de la expresión *heterodoxia en la lógica*, más explícitamente en lo que hace referencia al *grado de heterodoxia*. En éstos términos podemos afirmar que dentro de la amplia gama de sistemas lógicos, los paraconsistentes se destacan por ser heterodoxos, y dentro de éstos quizás los más, son los sistemas C_n de da Costa.

Los esquemas tradicionales del concepto de razón han sido superados por el desarrollo de la disciplina más racional: la lógica, y con ella la fundamentación de las matemáticas. Para dar cuenta de lo que sucede en el campo de la lógica es inevitable establecer un nuevo concepto de *razón* que conlleva una nueva filosofía del conocimiento. Baste señalar la necesidad de ampliar nuestro concepto habitual de razón, por otro dentro del cual se permitan las contradicciones, y si se quiere el regreso al infinitum en ciertos argumentos.

1.7. Aspectos finales

Consignemos algunas aplicaciones generales de las lógicas paraconsistentes.

1. *Filosóficas:* en la Ontología de los objetos matemáticos formales, si se usa la lógica tradicional (como la clásica, incluso la intuicionista) como lógica subyacente de la ontología, entre los objetos existentes no se encuentran los "inconsistentes", pero si usamos como lógica subyacente una lógica paraconsistente, todo cambia. Pues hay teorías de conjuntos paraconsistentes donde el conjunto de Russell "existe". Luego una ontología fundada en una lógica paraconsistente, puede en principio, contener objetos contradictorios.

   Las nuevas teorías de conjuntos paraconsistentes, las versiones paraconsistentes de la geometría y del cálculo han dado origen a nuevos conceptos y estructuras matemáticas. Así se amplía la matemática tradicional, y se pone en evidencia sus limitaciones.

Los sistemas paraconsistentes en el reino de la lógica está cercano al de las geometrías no euclidianas en el campo de las matemáticas, pero con la ventaja que dentro de buena parte de las lógicas paraconsistentes se puede incluir la clásica, hecho que no sucede con las geometrías no euclidianas.

A manera de conclusion se puede afirmar que las lógicas paraconsistentes constituyen una prueba formal de que la tesis de Hegel es correcta en el nivel abstracto: existen teorías paraconsistentes en las que ciertos objetos tienen propiedades inconsistentes; por ejemplo, pertenecen y simultáneamente no pertenecen a la misma clase. En cuanto a la validez de la tesis de Hegel en conexión con objetos concretos, pareciera que sigue siendo una pregunta abierta.

2. *Informática, inteligencia artificial y bases de datos:* la manipulación de informaciones inconsistentes no puede hacerse a través de la lógica clásica, dadas las trivializaciones. Estas manipulaciones están en los dominios de las lógicas paraconsistentes. De hecho hoy se usan las lógicas anotadas (las cuales son paraconsistentes) que han permitido una programación paraconsistente.

3. *Física Cuántica:* existen fenómenos contradictorios, la dualidad onda partícula. La escuela de Copenhague se pronunció filosóficamente a través de Bohr-Heisemberg con la idea de "complementariedad". Las lógicas paraconsistentes tienen sus aportes con los trabajos de Da Costa y Decio Krause [36].

4. *Robótica:* bajo la dirección de Jair Minoro Abe se ha desarrollado el llamado *para-analizador* que permite hacer aplicaciones en ciencia que manejen incertidumbres, inconsistencias y para-completez [2]. Dentro de este esquema es famoso el primer prototipo de robot paraconsistente (femenino) Emmy [3].

Capítulo 2

Algebrizabilidad

2.1. Introducción

Los inicios de lógica algebraica se pueden ubicar en el siglo XIX con los trabajos en lógica clásica de Boole, de De Morgan, Peirce, y de Schröder, entre otros. Ellos estudiaron las equivalencias lógicas de las proposiciones, más que la veracidad de las mismas. De cierta forma la noción de equivalencia se constituyó en una noción lógica primitiva, explotando la semejanza entre la equivalencia y la igualdad, con lo cual desarrollaron sistemas con un carácter distintivamente algebraico.

Boole desarrolló la teoría moderna de álgebras booleanas, y De Morgan, Peirce y Schröder la teoría de álgebras de relación. La lógica algebraica adquiere independencia con los sistemas lógicos de Frege, Russell y Whitehead. Reforzada por las ideas metamatemáticas de Hilbert, esta tendencia en lógica se enfocó alrededor de las nociones formales de la aserción (validez y demostrabilidad) y la inferencia lógica.

Nos encontramos con dos acercamientos, uno centrado en la noción de la equivalencia lógica y el otro centrado en las nociones de la aserción y de la inferencia.

La conexión entre estas dos maneras de mirar la lógica es establecida por Tarski, quién presenta la conexión exacta entre el álgebra booleana y el cálculo clásico proposicional.

Los trabajos de Tarski originan la necesidad de estudiar los sistemas deductivos en contextos más generales, y por ende podríamos decir que proporcionan el inicio de la lógica algebraica moderna. El aporte fundamental de Tarski es introducir el álgebra de fórmulas del cálculo proposicional, definir la relación:

$$\varphi \sim \psi \quad \text{si y sólo si} \quad \vdash_{CPC} \varphi \leftrightarrow \psi,$$

y demostrar que esta es una relación de equivalencia que respeta las operaciones, en términos modernos, resulta ser una *congruencia.* Además prueba que el álgebra cociente, modernamente llamada el *álgebra de Lindenbaum-Tarski del cálculo proposicional clásico*, es un álgebra de Boole. También establece el resultado recíproco, es decir, muestra la construcción de un sistema deductivo a partir de los axiomas de álgebras de Boole.

A través de la relación establecida por Tarski para el Cálculo Proposicional Clásico y las Algebras de Boole, han surgido a lo largo de la historia un sinnúmero de relaciones similares para otras lógicas no clásicas, mostrándose así que a éstas lógicas les corresponden distintas clases de álgebras. Veamos algunos de los ejemplos más conocidos en la siguiente lista:

Cálculo Proposicional Intuicionista	Algebras de Heyting
Lógicas multi-valuadas	MV-álgebras
Lógica Modal	Algebras Modales
Lógica de Primer Orden	Algebras Cilíndricas.

Las álgebras de Heyting son las primeras nuevas álgebras identificadas con el método de Lindenbaum-Tarski aplicado a un sistema deductivo particular, en este caso el cálculo proposicional intuicionista.

En los textos, ya clásicos en los estudios de algebrización de lógicas, de H. Rasiowa y R. Sikorski [68] y H. Rasiowa [67] se consignan los más importantes avances en la teoría de algebrización. Los sistemas lógicos por ellos considerados presuponen, entre otros, la existencia de una implicación con ciertas propiedades estándar, dejando por fuera de la posibilidad muchos otros sistemas sin tales características.

En 1989 Blok-Pigozzi [16], basado en los trabajos de Czelakowski [24] y de sus propios trabajos previos [18], presentan por primera vez una noción exacta del concepto de una *lógica algebrizable.* No hay lugar a dudas que la publicación de este trabajo ha sido de clara influencia para la aparición de un sinnúmero de otros trabajos que hoy constituyen una nueva línea de investigación conocida como *lógica algebraica abstracta.* En contraste con la lógica algebraica "tradicional", la lógica algebraica abstracta se ocupa de los sistemas deductivos abstractos, más bien que de sistemas específicos, y su propósito principal es estudiar el marco para la algebrización de estos sistemas y develar condiciones generales bajo las cuales amplias clases de sistemas deductivos puedan demostrarse algebrizables. Lo que se quiere es estudiar las características de una tal clase de sistemas deductivos algebrizable a través del estudio de las propiedades de las "algebras" obtenidas en el proceso de algebrización, y viceversa.

Dado un lenguaje $\mathcal{L}$ y un conjunto de variables enumerable V, un sistema deductivo $\mathcal{S} = \langle \mathcal{L}, \vdash_{\mathcal{S}} \rangle$ consiste, en términos generales, de una relación de consecuencia finita y estructural[1] definida sobre el conjunto de las fórmulas $\mathcal{F}m_{\mathcal{L}}(V)$.

De otro lado, dada una clase $\mathbf{K}$ de $\mathcal{L}-$ álgebras, se puede construir un sistema deductivo algebraico $\mathcal{S}_{\mathbf{K}} = \langle \mathcal{L}, \models_{\mathbf{K}} \rangle$ tal que la relación de consecuencia ahora es una relación de consecuencia sobre el conjunto de las ecuaciones $Eq_{\mathcal{L}}(V)$ de tipo $\mathcal{L}$, definida por $E \models_{\mathbf{K}} \phi \approx \psi$ si y sólo si para toda álgebra $\mathbf{A} = \langle A, \mathcal{L}^{\mathbf{A}} \rangle \in \mathbf{K}$, $\vec{a} \in A^{\omega}$, se cumple que si $e_1^{\mathbf{A}}(\vec{a}) = e_2^{\mathbf{A}}(\vec{a})$ para todo ecuación $e_1 \approx e_2 \in E$, entonces $\phi^{\mathbf{A}}(\vec{a}) = \psi^{\mathbf{A}}(\vec{a})$.

Blok-Pigozzi presentan posteriormente la noción de un sistema *k-deductivo* para unificar estas dos nociones. Un sistema k-deductivo consiste en una relación de consecuencia en k-uplas de $\mathcal{L}-$fórmulas. Así, un sistema deductivo en el sentido original es un sistema $1-$*deductivo* y un sistema deductivo algebraico es un sistema $2-$*deductivo*.

Un sistema deductivo $\mathcal{S} = \langle \mathcal{L}, \vdash_{\mathcal{S}} \rangle$ es *interpretable* en un sistema deductivo algebraico $\mathcal{S}_{\mathbf{K}} = \langle \mathcal{L}, \models_{\mathbf{K}} \rangle$ si existe un conjunto finito de n ecuaciones en una variable, $\delta(v) \approx \epsilon(v) = \{\delta_i(v) \approx \epsilon_i(v) : i < n\}$ tal que para todo $\Phi \cup \{\psi\} \subseteq \mathcal{F}m_{\mathcal{L}}(V)$, se cumple que $\Phi \vdash_{\mathcal{S}} \psi$ si y sólo si $\{\delta(\phi) \approx \epsilon(\phi) : \phi \in \Phi\} \models_{\mathbf{K}} \delta(\psi) \approx \epsilon(\psi)$. En este caso diremos que el conjunto $\delta \approx \epsilon$ es un conjunto de *ecuaciones de definición* para $\mathcal{S}$ y $\mathbf{K}$.

Por otra parte, el sistema deductivo algebraico $\mathcal{S}_{\mathbf{K}}$ es *interpretable* en el sistema deductivo $\mathcal{S}$ si existe un conjunto finito $\Delta(v, u) = \{\Delta_j(v, u) : j < m\}$ de m fórmulas en dos variables v, u tales que para todo $E \cup \{\phi \approx \psi\} \subseteq Eq_{\mathcal{L}}(V)$ se cumple que $E \models_{\mathbf{K}} \phi \approx \psi$ si y sólo si $\{\Delta(e_1, e_2) : e_1 \approx e_2 \in E\} \vdash_{\mathcal{S}} \Delta(\phi, \psi)$. En este caso diremos que Δ es un *conjunto de fórmulas de equivalencia* para $\mathcal{S}$ y $\mathbf{K}$.

Nuevamente en aras de unificar estas dos nociones de interpretabilidad, Blok-Pigozzi proporcionan la noción del $k - l -$ *interpretabilidad*. En éstos términos el primer caso considerado arriba es la descripción de $1 - 2 -$ *interpretabilidad* y el segundo de $2 - 1 -$ *interpretabilidad*. Si se quiere esta noción generalizada es inicialmente la inspiración principal para la definición de interpretabilidad a través de las *instituciones* en Voutsadakis [77].

Dos sistemas, uno l-deductivo y otro k-deductivo se dicen *equivalentes* si son interpretables el uno en el otro y además las dos interpretaciones son inversas, es decir, si salimos de una k-fórmula, y aplicamos la k-l-interpretación y a continuación aplicamos la l-k-interpretación al sistema que resulta de l-fórmulas, obtenemos un conjunto de k-fórmulas que son interderivables, con respecto a la relación de la k-consecuencia, con la k-fórmula original y lo mismo sucede si comenzamos con una l-fórmula y aplicamos las interpretaciones en el orden contrario.

[1]Términos que precisaremos un poco más adelante.

Específicamente para los sistemas 1-deductivo y 2-deductivo otra vez, tenemos que el sistema deductivo $\mathcal{S}$ es equivalente al sistema deductivo algebraico $\mathcal{S}_{\mathbf{K}}$ si, para cada $\phi \in \mathcal{F}m_{\mathcal{L}}(V)$ y $\phi \approx \psi \in Eq_{\mathcal{L}}(V)$, los conjuntos de ecuaciones de definición $\{\phi \approx \psi\} \subseteq Eq_{\mathcal{L}}(V)$ y el de fórmulas de equivalencia Δ son inversos el uno del otro, es decir, $\phi \dashv\vdash_{\mathcal{S}} \Delta(\delta(\phi), \epsilon(\phi))$ y $\phi \approx \psi \;=\!\!|\!\!\models_{\mathbf{K}} \delta(\Delta(\phi, \psi)) \approx \epsilon(\Delta(\phi, \psi))$.

Las ideas de Voutsadakis empiezan con la disertación doctoral, escrita bajo supervisión del profesor Don Pigozzi de la Iowa State University, las cuales esencialmente apuntan a extender la algebrizabilidad de Blok-Pigozzi a fin de cubrir la algebrizabilidad de lógicas multisignadas y con cuantificadores. El usa las instituciones como la estructura de soporte en lugar de sistemas deductivos. Más precisamente introduce dentro de la teoría de mónadas del álgebra categórica los conceptos institución algebraica y de institución algebrizable, y presenta la noción de la equivalencia de las instituciones.

Voutsadakis muestra que la lógica ecuacional y la lógica de primer orden (sin términos) se pueden formular dentro del marco de las instituciones, probándose su algebrizabilidad en su versión extendida.

De esta forma nos encontramos con que la teoría general de algebrización de sistemas lógicos ha pasado a ser el centro de las investigaciones, dando paso a la llamada *Lógica Algebraica Abstracta* (abreviadamente LAA).

A riesgo de repetirnos, insistamos que una de las metas de la LAA es descubrir criterios generales para determinar cuándo una clase de álgebras, o una clase de los objetos matemáticos relacionados de cerca con las álgebra como por ejemplo las matrices lógicas o matrices generalizadas, son adecuadas para ser las contrapartes algebraicas de una lógica. A partir de tales criterios, en LAA se desarrollan los métodos para obtener las contrapartes algebraicas. Luego en LAA la generalización del método de Lindenbaum-Tarski juega un papel relevante.

Los teoremas que relacionan características metalógicas de una lógica con las características algebraicas de sus contrapartes algebraicas adquieren interés agregado en el contexto de LAA. Por ejemplo, se conocía que hay una conexión cercana entre el teorema de la deducción y la propiedad de una clase de álgebras de que sus miembros tienen congruencias principales definibles ecuacionalmente de forma uniforme, pero solamente en un contexto más general de LAA es que tal conexión se puede hacer en forma exacta. Más aún, el interés por encontrar el contexto apropiado en el cual esta conexión se pueda establecer en forma exacta es en buena parte una de las mayores motivaciones del desarrollo de LAA.

Hay otros teoremas que relacionan características metalógicas tales como definibilidad (de Beth), la existencia de cálculos de Gentzen, etc. con las características algebraicas tales como la propiedad de que los epimorfismos son sobreyectivos, de extensión de congruencias, etc.

Otra meta importante de LAA es la clasificación de los sistemas lógicos a través de las propiedades de sus contrapartes algebraicas. Cuando se sabe que una lógica dada pertenece a un grupo particular en la clasificación, se espera idealmente tener los teoremas generales que proporcionen la información importante sobre sus características y comportamiento.

Sin embargo, creemos que la LAA es aún bastante joven, así entender algunos puntos en su historia temprana son necesarios para apreciarla completamente.

A manera de resumen digamos que la lógica algebraica puede verse como la parte de la lógica que enfatiza el estudio de las equivalencias de las proposiciones, antes que de su validez. Este hecho se hace específicamente cierto cuando hablamos de lógica algebraica abstracta que tiene por objeto central de estudio la manera en la cual equivalencia y validez se conectan entre si.

Antes de entrar en formulaciones precisas sobre los procesos de algebrización, creemos pertinente detenernos previamente a dar una vista general de lo que entenderemos por una lógica, lo que nos obliga a presentar en forma precisa los conceptos de relación de consecuencia y sistema deductivo.

2.2. Operadores de Consecuencia

Dado un conjunto A, llamaremos un *operador de consecuencia*[2] sobre A a una función $C : \mathcal{P}(A) \longrightarrow \mathcal{P}(A)$ tal que:

(C1) $X \subseteq C(X)$

(C2) $CC(X) = C(X)$

(C3) Si $X \subseteq Y$ entonces $C(X) \subseteq C(Y)$.

Si además un operador de consecuencia sobre un conjunto A satisface

$$[(C4)] \quad C(X) = \bigcup\{C(Y) : Y \subseteq X, Y \mathit{finito}\}$$

diremos que C es *finitario.*

[2]Las condiciones impuestas se deben a Tarski (1930), aunque el nombre aparece posteriormente.

Todo operador de consecuencia tiene asociada una relación $\vdash_C \subseteq P(A) \times A$ definida entre subconjuntos de A y elementos de A. Esta relación está definida para todos $X \subseteq A$ y $a \in A$ en el sentido de que $X \vdash_C a$ si y sólo si $a \in C(X)$.

Las propiedades transmitidas a $\vdash_C$ por las condiciones $(C1)$, $(C2)$ y $(C3)$ sobre C definen lo que es una *relación de consecuencia*. Esas tres condiciones se trasladan en las siguientes dos condiciones sobre $\vdash_C$.

(C1') si $a \in X$, entonces $X \vdash_C a$, y

(C2') si $Y \vdash_C a$ para toda $a \in X$, y $X \vdash_C b$, entonces $Y \vdash_C b$,

las cuales implican la condición de monotonía (*C3'*), si $X \vdash_C a$ y $X \subseteq Y$, entonces $Y \vdash_C a$.

En el otro sentido, toda relación de consecuencia $\vdash$ define un operador de consecuencia $C_\vdash$ estableciendo que $a \in C_\vdash(X)$ si y sólo si $X \vdash a$. Los dos procesos son inversos el uno del otro.

En 1958 Los y Suszko adicionan a las condiciones de Tarski la *estructuralidad* ó invarianza bajo sustituciones. Condición que detallaremos más adelante para los operadores de consecuencia de nuestro interés.

2.3. Sistemas Deductivos

Un *lenguaje proposicional*, o simplemente un *lenguaje*, es un conjunto $\mathcal{L}$ de símbolos cuyos elementos son llamados *conectivos lógicos,* cada uno de los cuales tiene asociado un número natural, su *aridad.* Los conectivos de aridad cero se denominan *constantes.* Los conectivos pueden considerarse como símbolos de operación de un tipo de *similaridad algebraica,* y entonces las fórmulas son el conjunto de términos de este tipo de similaridad sobre el conjunto de variables proposicionales. Por lo tanto tenemos a nuestra disposición el álgebra de términos, que es un álgebra absolutamente libre del tipo $\mathcal{L}$ sobre un sistema enumerable de los generadores $\mathcal{V}$, el conjunto de las variables. La llamaremos el *álgebra de fórmulas* y la denotamos por $\mathcal{F}m$. Así, $\mathcal{F}m$ consiste del conjunto de fórmulas junto con las operaciones de formar las fórmulas complejas asociadas a cada conectivo. Más precisamente, $\mathcal{F}m$ se define recursivamente de la manera usual $\mathcal{V} \subset \mathcal{F}m$, y si α es un conectivo n-ario y $\varphi_1, \varphi_2, \ldots, \varphi_n \in \mathcal{F}m$, entonces $\alpha(\varphi_1, \varphi_2, \ldots, \varphi_n) \in \mathcal{F}m$.

Lema 2.1 *El conjunto $\mathcal{F}m$ es el conjunto mas pequeño que contiene a $\mathcal{V}$ y que es cerrado bajo todos los conectivos lógicos.*

Con esta observación es válido un principio de recursión sobre $\mathcal{F}m$, que nos permite extender recursivamente de manera única toda función $\sigma : \mathcal{V} \longrightarrow \mathcal{F}m$ a la función $\bar{\sigma} : \mathcal{F}m \longrightarrow \mathcal{F}m$ como sigue:

$$\bar{\sigma}(\varphi(v_1, \ldots, v_n)) = \varphi(\sigma(v_1), \ldots, \sigma(v_n)).$$

A dicha función la llamaremos *substitución* y la denotaremos por la misma letra σ.

Para cada $\Gamma \subseteq \mathcal{F}m$ denotaremos $\sigma(\Gamma)$ al conjunto formado por todas las fórmulas de γ pasadas por la sustitución σ, es decir, $\sigma(\Gamma) = \{\sigma(\varphi) : \varphi \in \Gamma\}$.

Una relación de consecuencia $\vdash$ se dice *estructural* si para todos los $\Gamma \subseteq \mathcal{F}m$, y todas $\varphi, \psi \in \mathcal{F}m$, si $\Gamma \vdash \varphi$ y σ es una sustitución, entonces $\sigma(\Gamma) \vdash \sigma(\varphi)$. Haciendo la correspondencia a la propiedad $C4$ para los operadores de consecuencia, si siempre que $\Gamma \vdash \varphi$ entonces existe un subconjunto finito $\Gamma' \subseteq \Gamma$ tal que $\Gamma' \vdash \varphi$ diremos que $\vdash$ es *finitaria.*[3]

Observación 2:1 *La monotonía:*

$$\Gamma \vdash \varphi \qquad y \qquad \Gamma \subseteq \Delta, \; entonces \, \Delta \vdash \varphi$$

es conclusión de las reglas anteriores.

Una *regla de inferencia* sobre $\mathcal{L}$ es un par $\langle \Gamma, \varphi \rangle$, donde $\Gamma \subseteq \mathcal{F}m$, Γ es finito y $\varphi \in \mathcal{F}m$.

Diremos que ψ es *directamente demostrable* a partir de Δ por la regla $\langle \Gamma, \varphi \rangle$, si existe una substitución σ tal que $\sigma(\varphi) = \psi$ y $\sigma(\Gamma) \subseteq \Delta$.

Un *axioma* es una regla de la forma $\langle \emptyset, \psi \rangle$. Cualquier sustitución de un axioma es directamente demostrable a partir de cualquier conjunto de fórmulas Δ. Cada sustitución será una *instancia* del axioma. Una formula tal que $\emptyset \vdash \varphi$ es un *teorema* de $\mathcal{S}$ y lo denotamos simplemente $\vdash \varphi$.

Dado un conjunto de axiomas y de reglas de inferencia en un lenguaje $\mathcal{L}$, entenderemos el *sistema deductivo* $\mathcal{S}$ sobre $\mathcal{L}$, determinado por el par $\langle \mathcal{L}, \vdash \rangle$ donde $\vdash$ es la relación entre conjuntos de fórmulas y fórmulas definida de la manera siguiente:

$\Delta \vdash \psi$ si y sólo si existe una sucesión finita $\langle \varphi_1, \varphi_2, \ldots, \varphi_n \rangle$ de fórmulas de $\mathcal{F}m$, tal que $\varphi_n = \psi$ y para todo $i \leq n$ se cumple alguna de las condiciones siguientes:

- φ_i es una instancia de un axioma.

[3]Cabe señalar que existen ampliaciones a lógicas infinitarias que no serán consideradas aquí.

- $\varphi_i \in \Delta$
- para ciertos $i_1, i_2, \ldots, i_k$ todos menores que i, φ_i es directamente demostrable a partir de $\{\varphi_{i_1}, \ldots, \varphi_{i_k}\}$.

Esta sucesión se llama una *demostración de* ψ *a partir de* Γ. Una fórmula tal que $\vdash \psi$ es un *teorema* de $\mathcal{S}$.

Lema 2.2 *Si* $\mathcal{S} = \langle \mathcal{L}, \vdash \rangle$ *es un sistema deductivo, entonces* $\Delta \vdash \psi$ *si y sólo si* ψ *pertenece al conjunto más pequeño de fórmulas que contiene a* Δ*, a todas las sustituciones de los axiomas y es cerrado bajo pruebas directas.*

A fin de caracterizar las relaciones que llamaremos de consecuencia, presentamos el siguiente resultado estándar en la literatura. Una prueba del mismo se puede encontrar en [16].

Lema 2.3 *Si* $\mathcal{S} = \langle \mathcal{L}, \vdash \rangle$ *es un sistema deductivo, entonces para todo* $\Gamma, \Delta \subseteq \mathcal{F}m$ *y* $\varphi, \psi \in \mathcal{F}m$ *se tiene:*

1. $\Delta \vdash \varphi$ *, para todo* $\varphi \in \Delta$*.*
2. *Si* $\Delta \vdash \psi$ *y* $\Delta \subseteq \Gamma$*, entonces* $\Gamma \vdash \psi$*.*
3. *Si* $\Delta \vdash \psi$ *y* $\Gamma \vdash \varphi$*, para todo* $\varphi \in \Delta$*, entonces* $\Gamma \vdash \psi$*.*
4. *Si* $\Delta \vdash \psi$*, entonces existe* $\Gamma \subseteq \Delta$ *finito tal que* $\Gamma \vdash \psi$*, diremos que* $\mathcal{S}$ *es* finitario.
5. *Si* $\Delta \vdash \psi$*, entonces* $\sigma(\Delta) \vdash \sigma(\psi)$*, para toda substitución* σ*, diremos que* $\mathcal{S}$ *es estructural.*

Cualquier relación que satisfaga 1–5 es la relación de consecuencia de algún sistema deductivo. Podemos así definir un sistema deductivo como un par $S = \langle \mathcal{L}, \vdash \rangle$, donde $\vdash$ es una relación entre conjuntos de fórmulas y fórmulas que verifica 1–5 del lema anterior. Nótese que esta definición no hace alusión alguna a reglas de inferencia ni a axiomas.

En otras palabras un *sistema deductivo* (o sistema lógico), o en forma abreviada, *una lógica,* representa un par $S = \langle \mathcal{F}m, \vdash \rangle$. Usaremos la notación $\Gamma \vdash \Delta$ para indicar que para toda $\varphi \in \Delta$ se tiene que $\Gamma \vdash \varphi$. En esta presentación no se asumen un conjunto de axiomas, junto a ciertas reglas de inferencia para definir el sistema, aunque es conocida la equivalencia entre las dos nociones. Una elección particular de axiomas y reglas de inferencia para un sistema S será llamada una *presentación* de S. De hecho para cada sistema deductivo existen muchas diferentes presentaciones.

Una *teoría* de S, ó equivalentemente, una S-*teoría* es un conjunto de fórmulas Γ cerrado bajo la relación de consecuencia $\vdash_S$, es decir, tal que si $\Gamma \vdash_S \varphi$, entonces $\varphi \in \Gamma$. El conjunto de todas las S-teorías será denotado por ThS.

Ejemplos

1. El Cálculo Proposicional Clásico (CPC) es el sistema deductivo que tiene por lenguaje los conectivos $\mathcal{L} = \{\to, \wedge, \vee, \neg\}$ y su tipo de similaridad es $\langle 2,2,2,1 \rangle$. Las fórmulas, además de las variables proposicionales, son $\varphi_1 \to \varphi_2$, $\varphi_1 \wedge \varphi_2$, $\varphi_1 \vee \varphi_2$ y $\neg\varphi_1$.

2. El lenguaje de las lógicas modales se obtiene agregando al anterior un conectivo unario $\Box$, llamado operador de necesitación de tal modo que si φ es una fórmula, $\Box\varphi$ también lo es.

3. Una primera presentación para el Cálculo Proposicional Clásico.

Axiomas:

A1 $\quad p \to (q \to p)$

A2 $\quad (p \to q) \to ((p \to (q \to r)) \to (p \to r))$

A3 $\quad (p \wedge q) \to p,$

A4 $\quad (p \wedge q) \to q,$

A5 $\quad p \to (q \to (p \wedge q)),$

A6 $\quad p \to (p \vee q),$

A7 $\quad q \to (p \vee q),$

A8 $\quad (p \to z) \to ((q \to z) \to ((p \vee q) \to z)),$

A9 $\quad (\neg p \to \neg q) \to (q \to p).$

Regla: Modus Ponens

MP $\quad p \to q\, , p \vdash_{CPC} q\, .$

4. Una segunda presentación para el Cálculo Proposicional Clásico. Tomando el lenguaje $\mathcal{L} = \{\rightarrow, \neg\}$.

Axiomas:

A1 $\quad p \rightarrow (q \rightarrow p)$

A2 $\quad (p \rightarrow q) \rightarrow ((p \rightarrow (q \rightarrow r)) \rightarrow (p \rightarrow r))$

A3 $\quad (\neg p \rightarrow \neg q) \rightarrow (q \rightarrow p)$

- **Regla:** Modus Ponens

5. El sistema de lógica modal S_4 tiene el lenguaje de CPC (presentación 1 o 2) y el conectivo unario $\Box$.

Axiomas:

A1 Todas las tautologías del CPC son axiomas.

A2 $\quad \Box(p \rightarrow q) \rightarrow (\Box p \rightarrow \Box q)$.

A3 $\quad \Box p \rightarrow p$,

A4 $\quad \Box p \rightarrow \Box\Box p$.

Reglas:

MP $\quad p \rightarrow q\, , p \vdash_{S4} q$,

N $\quad p \vdash_{S4} \Box p$.

6. El sistema $\mathcal{G}$ definido sobre el lenguaje $\mathcal{L} = \{\,\cdot\,, {}^{-1}, e\}$.

G1 $\quad ((p \cdot q) \cdot r) \cdot (p \cdot (q \cdot r))^{-1}$,

G2 $\quad (p \cdot e) \cdot p^{-1}$,

G3 $\quad (e \cdot p) \cdot p^{-1}$,

G4 $\quad p \cdot p^{-1}$,

G5 $\quad p^{-1} \cdot p$.

Reglas:

R1 $\quad p \cdot q^{-1} \vdash_{\mathcal{G}} q \cdot p^{-1}$,

R2 $\quad p \cdot q^{-1} \vdash_{\mathcal{G}} p^{-1} \cdot (q^{-1})^{-1}$,

R3 $\quad \{p \cdot q^{-1}\,,\, q \cdot r^{-1}\} \vdash_{\mathcal{G}} p \cdot r^{-1}$,

R4 $\quad \{p \cdot q^{-1}\,,\, r \cdot s^{-1}\} \vdash_{\mathcal{G}} (p \cdot r) \cdot (q \cdot s)^{-1}$,

R5 $\quad p \vdash_{\mathcal{G}} p \cdot e^{-1}$,

R6 $\quad p \cdot e^{-1} \vdash_{\mathcal{G}} p$.

En los ejemplos 3 y 4, los sistemas deductivos están asociados a la Lógica Proposicional Clásica, en el sentido de que ambos son sistemas correctos y completos:

$$\Gamma \models \varphi \qquad \text{si y sólo si} \qquad \Gamma \vdash_{CPC} \varphi\,.$$

La diferencia entre ambos es el lenguaje; sin embargo, si a la segunda presentación se agregan las definiciones:

$$\begin{aligned} p \vee q &:= \neg p \to q \\ p \wedge q &:= \neg(p \to \neg q), \end{aligned}$$

en él se pueden probar todos los axiomas correspondientes del primer sistema.

2.4. Matrices Lógicas

Las matrices lógicas fueron introducidas formalmente en la tercera década del siglo XX por Lukasiewicz y Tarski, aunque ya aparecían en algunos trabajos anteriores del mismo Lukasiewicz, de Bernays y Post entre otros. Para presentar el concepto de matriz lógica necesitamos precisar el de álgebra sobre un lenguaje dado.

Dado un lenguaje $\mathcal{L}$, diremos que una *$\mathcal{L}$-algebra* es una estructura $\mathbf{A} = \langle A, w\rangle_{w\in\mathcal{L}}$ donde A el universo de $\mathbf{A}$ es no vacío, y w es una operación sobre A de rango k para cada conectivo w de rango k.

Una *$\mathcal{L}$-matriz* es un par $\mathcal{A} = \langle \mathbf{A}, F\rangle$, donde $\mathbf{A}$ es una $\mathcal{L}$-algebra y F es un subconjunto de A; los elementos de F se llaman los *valores designados* de $\mathcal{A}$.

Dada una $\mathcal{L}$-matriz $\langle \mathbf{A}, F\rangle$, cada fórmula φ de $\mathcal{L}$ tiene una única interpretación en $\mathbf{A}$ dependiendo sólo de los valores en $\mathbf{A}$ que son asignados a sus variables. Usando el hecho de que $\mathcal{F}m$ es el álgebra absolutamente libre generada por el conjunto de las variables, la interpretación de una fórmula φ puede ser expresada algebraicamente con $h(\varphi)$, donde $h : \mathcal{F}m \longrightarrow \mathbf{A}$ es el homomorfismo que aplica cada variable en el valor asignado.

Una *valuación* sobre una $\mathcal{L}$-matriz $\mathcal{A} = \langle \mathbf{A}, F\rangle$ es el único homomorfismo que extiende la función (definida en el conjunto de variables proposicionales) $u : P \longrightarrow A$ a $u^* : \mathcal{F}m \longrightarrow \mathbf{A}$.

La fórmula φ es una *consecuencia* de Γ en $\mathcal{A}$, simbólicamente $\Gamma \models_{\mathcal{A}} \varphi$ si para toda valuación $u : P \longrightarrow A$, $u^*(\psi) \in F$, para toda $\psi \in \Gamma$ implica $u^*(\varphi) \in F$.

Para un sistema deductivo (una lógica) S diremos que una matriz $\mathcal{A}$ es una *matriz modelo* de S si $\Gamma \vdash \varphi$ implica $\Gamma \models_{\mathcal{A}} \varphi$, en este caso el subconjunto F es llamado *S-filtro*. Observe que si T es una S-teoría, entonces $\langle \mathcal{F}m, T\rangle$ es una matriz modelo de S. Estas matrices reciben el nombre de *matrices de Lindenbaum* para S. La clase de todas las matrices modelo de una lógica S es denotada por $ModS$.

Una lógica S en un lenguaje $\mathcal{L}$ se dice *completa relativa a la clase de $\mathcal{L}$-matrices* $\mathbf{M}$ si $\mathbf{M} \subseteq ModS$ y para todos $\Gamma \cup \{\varphi\} \subseteq \mathcal{F}m$ tal que $\Gamma \nvdash \varphi$ existe una matriz $\langle \mathbf{A}, F\rangle \in \mathbf{M}$ y un $h \in Hom(\mathcal{F}m, A)$ tal que $h[\Gamma] \subseteq F$, pero $h(\varphi) \notin F$. En tal caso diremos que $\mathbf{M}$ es una *semántica matricial* para S. Es decir, una clase $\mathbf{M}$ de matrices es una semántica matricial para S si $\Gamma \vdash \varphi$ si y sólo si $\Gamma \models_{\mathcal{A}} \varphi$ para toda $\mathcal{A} \in \mathbf{M}$. Por ejemplo, la clase de todas las matrices de Lindenbaum para S y la clase de todas las matrices modelo de S son semánticas matriciales.

Una *congruencia* de una matriz $\mathcal{A} = \langle \mathbf{A}, F\rangle$ es una relación binaria θ sobre A, que es una relación de equivalencia sobre $\mathbf{A}$ y es *compatible* con F, es decir, satisface que para todos $a, b \in A$, $si (a, b) \in \theta$ y $a \in F$, implica que $b \in F$.

Para toda matriz $\mathcal{A} = \langle \mathbf{A}, F\rangle$ existe la más grande congruencia, la cual es llamada congruencia de Leibniz de la matriz $\mathcal{A}$, ó *la congruencia de Leibniz de F* en A, y es denotada por $\Omega_{\mathcal{A}} F$.

2.5. Formalización de la Algebrizabilidad

Un sistema deductivo S sobre un lenguaje $\mathcal{L}$ es *algebrizable* en el sentido de Blok-Pigozzi si existe una clase de $\mathcal{L}$-algebras K tal que la relación de S-consecuencia $\vdash$ y la relación de consecuencia ecuacional $\models_K$ sobre K son interpretables la una en la otra en un sentido fuerte, es decir, dada una fórmula α existe una ecuación $\widehat{\alpha}$ y dada una ecuación $\sigma \approx \tau$ existe una fórmula $\widehat{\sigma \approx \tau}$ tal que:

1. $\Gamma \vdash \alpha$ si y sólo si $\widehat{\Gamma} \models_K \widehat{\alpha}$,

2. $\Sigma \models_K \sigma \approx \tau$ si y sólo si $\widetilde{\Sigma} \vdash \widetilde{\sigma \approx \tau}$,

3. $\alpha \dashv\vdash \widetilde{\widehat{\alpha}}$,

4. $\sigma \approx \tau \mathrel{=\!\!|\!\models_K} \widehat{\widetilde{\sigma \approx \tau}}$.

La cuasivariedad K es llamada la *semántica algebraica equivalente* de S.

Observación 2.2

- *Intuitivamente los items 1 y 4 dicen que toda deducción en S puede convertirse en una consecuencia ecuacional dentro de la cuasivariedad K, en donde podemos usar todas las herramientas de la teoría de modelos y álgebra universal.*

- *Los items 2 y 3 dicen lo mismo en la otra dirección, es decir, las consecuencias ecuacionales en K se convierten en deducciones en el sistema S.*

Dentro de la propuesta de Blok-Pigozzi [16] aparece una muy usada caracterización de algebrizabilidad (Teorema 4.7), que enunciaremos a continuación.

Teorema 2.4 *Un sistema lógico (finitario)*[4] *S es algebrizable si y sólo si existen términos unarios $\delta_i, \epsilon_i, i \in I$ y términos binarios $\Delta_j, j \in J$, con I y J finitos tales que para todas $A, B, C, A_1, A_2, \ldots, A_n, B_1, B_2, \ldots, B_n \in \mathfrak{F}$, y para todo símbolo ω de conectivo $n-ario$ se tiene que:*

[1] $\vdash A\Delta A$

[2] $A\Delta B \vdash B\Delta A$

[3] $A\Delta B, B\Delta C \vdash A\Delta C$

[4] $A_1\Delta B_1, \ldots, A_n\Delta B_n \vdash \omega(A_1, A_2, \ldots, A_n)\Delta\omega(B_1, B_2, \ldots, B_n)$.

[5] $A \dashv\vdash \delta(A)\Delta\epsilon(A)$

Aquí $\vdash A\Delta B$ abrevia, $\vdash A\Delta_j B$, para todo $j \in J$, y análogamente δ y ϵ son abreviaciones. El conjunto Δ se denomina un sistema de fórmulas de equivalencia de S y $\delta \approx \epsilon$ un conjunto de ecuaciones de definición de S.

[4] En la versión original de Blok-Pigozzi aparece la restricción que el sistema deductivo sea finitario, es decir, que la longitud de las cadenas de demostración sean finitas.

La propuesta de Herrmann([48], [49]) difiere principalmente de la de Blok-Pigozzi en que los conjuntos I, J no son necesariamente finitos, es así que en dicha propuesta se refieren a la algebrizabilidad tipo Blok-Pigozzi, como *finitamente algebrizable.* Señalamos que estas dos nociones son muy similares en sus propiedades formales y en los ejemplos naturales de las lógicas que son algebrizables en el sentido más general, la propuesta de Herrmann generaliza adecuadamente la de Blok-Pigozzi. Pero cabe señalar que de las condiciones de finitud para la lógica y para los conjuntos I, J Blok-Pigozzi demuestran que la *semántica algebraica equivalente*, es decir, la clase de algebras que representa la lógica en cuestión, es una cuasivariedad, lo que no se garantiza para la noción de Herrmann. La noción de Czelakowski-Jansana ([26]) *algebrizabilidad débil* es mucho más general y da cuenta inclusive de sistemas lógicos infinitarios (sistemas lógicos dentro de los cuales la longitud de las cadenas de demostración pueden ser infinitas). Como nosotros hemos restringido nuestro sistema a ser finitario nos quedaremos con la noción de algebrizabilidad tipo Herrmann que nos resulta lo suficientemente general según nuestros intereses particulares.

Se conoce que hay ciertos sistemas lógicos paraconsistentes que son algebrizables, dentro de estos podremos citar el caso de la lógica de Sette P_1, resultado obtenido por R. Lewin, I. Mikenberg y G. Schwarze ([54]). Quienes además demostraron que la lógica original de Newton da Costa C_1 no es algebrizable ([55]), y por tanto tampoco lo son todas las C_n. Se conocen un sinnúmero de trabajos relacionados con C_1 en la búsqueda de un mejor comportamiento, modificando sus axiomas y agregando reglas de inferencia. Esto ha dado lugar a una gran variedad de lógicas paraconsistentes que se podrían considerar variaciones de C_1, como las de Bunder ([20]), las de Urbas ([74]), las de Sylvan ([71]) y las de Béziau ([13]). Así mismo hay reiterados intentos de dar una contrapartida algebraica adecuada para las C_n, entre otros los dados por da Costa ([27],[28].[28]), Sette ([69]), Fidel ([42]) y Carnielli-de Alcantara ([22]).

Dentro de un contexto más clásico aún, Béziau en ([14]) retomando las técnicas de algebrización al estilo Tarski-Lindenbaum, da cuenta de limitaciones similares en los procesos de algebrización.

De otra parte, el grupo de R. Lewin desarrolló una versión estructural de la lógica anotada presentada en ([40]) por da Costa, Subrahmanian y Vago (que resulta no ser algebrizable por no ser estructural), el sistema SAL ([56], [57]), para el que demostraron que es algebrizable si y sólo si el retículo de anotaciones es finito.

Para cerrar esta presentación acerca de la algebrizabilidad es muy pertinente resaltar una técnica a través del denominado *operador de Leibniz* que permite determinar muy fácilmente cuando una lógica no es algebrizable en el sentido de Blok-Pigozzi.

Dadas $\mathbf{A}$ una $\mathcal{L}$-algebra y $F \subseteq A$, definimos la relación binaria sobre $\mathbf{A}$:

$$\Omega_A F = \left\{ \begin{array}{ll} \langle a,b\rangle : & \varphi^A(a,\bar{c}) \in F \Leftrightarrow \varphi^A(b,\bar{c}) \in F, \text{ para todo} \\ & \varphi(p,q_0,\ldots,q_{n-1}) \in F \text{ y todo } \bar{c} \in A^n \end{array} \right\}.$$

La relación $\Omega_A F$ se llama la *relación de Leibniz en A sobre F.* El operador sobre las partes de A, que denotaremos como Ω_A, es llamado *operador de Leibniz sobre A.*

Se puede probar que $\Omega_A F$ es la congruencia más grande compatible con F, es decir, para todo $a, b \in A$ tal que $a \in F$ y $\langle a\,,\, b\rangle \in \Omega_A F$ entonces $b \in F$.

La técnica que mencionamos antes se concreta con una importante caracterización de algebrizabilidad de Blok-Pigozzi; un sistema deductivo S es algebrizable si y sólo si el operador de Leibniz es inyectivo, preserva el orden de ThS y preserva uniones de subconjuntos dirigidos en ThS. Esta técnica se uso por Lewin et. al. en [55] para probar que la lógica básica C_1 de Da Costa no es algebrizable, por tanto todas las lógicas C_n de Da Costa tampoco lo son.

Capítulo 3

Lógicas Anotadas

3.1. Introducción a las Lógicas Anotadas

Las Lógicas Anotadas constituyen un tipo de lógicas no clásicas, de relativamente reciente invención, que surgieron dentro del ambiente de la programación lógica. Sus fundamentos han sido objeto de estudio posterior a algunas aplicaciones en la programación lógica. Actualmente sus aplicaciones se derivan principalmente por las diferentes ramas de la inteligencia artificial, particularmente en el manejo de bases de datos inconsistentes.

Si se quiere en rigor las primeras de ellas aparecen inicialmente dentro de una teoría de programación lógica basada en *reglas cuantitativas* y la semántica declarativa asociada ([75]), siendo su gestor inicial Subrahmanian ([70]), quien corrige ciertas deficiencias de los lenguajes presentados en [75], dando un nuevo lenguaje para la *programación lógica cuantitativa*, lo cual constituye el origen de las lógicas anotadas.

Originalmente en [75] se presenta un lenguaje de programación en el cual cada implicación tiene asociado un *factor de atenuación* $\mathbf{c} \in (0, 1]$ sobre las premisas. Intuitivamente, $\mathbf{c}$ afecta (o contribuye) al valor de verdad de la premisa señalada.

Las reglas del lenguaje son expresiones de la forma:

$$A \longleftarrow\!\!\!\text{ⓒ}\!\!\!\longrightarrow B_1 \wedge B_2 \wedge \ldots \wedge B_n,$$

y una interpretación I satisface una instancia cerrada (sin variables libres) de la regla si $I(A) \geqslant \mathbf{c}.\ \text{mín}\ \{I(B_i) : 1 \leqslant i \leqslant n\}$ [1]. Cuando I satisface todas las instancias cerradas de la regla se dice que I satisface la regla.

Dado un conjunto de reglas cuantitativas R, el operador de consecuencia C_R asociado a R se define por

[1]Se asume el mínimo del conjunto vacío como 1.

$$C_R(I)(A) = \bigvee\{\mathbf{c} \cdot min\{I(B_i) : 1 \leqslant i \leqslant n\}\};$$

donde $A \longleftarrow\!\!\text{©}\!\!\longrightarrow B_1 \wedge B_2 \wedge \ldots \wedge B_n$ es una instancia de una regla de R. Una interpretación I se dice un modelo de R si y sólo si $C_R(I) \leqslant I$.

El lenguaje del conjunto de reglas cuantitativas no permite átomos negados en el cuerpo de las reglas; Subrahmanian probó que la existencia de literales[2] negativos en el cuerpo de una regla de un conjunto R implica la no monotonía de C_R, lo cual no garantiza la existencia de un modelo para R.

Otra deficiencia detectada en los lenguajes de conjuntos de reglas cualitativas es el orden de las interpretaciones, así por ejemplo si I_1 e I_2 son tales que para un átomo cerrado $I_1(A) = 0$ e $I_2(A) = 0{,}5$; I_1 dice que A es falso, mientras que I_2 no proporciona información alguna, luego I_2 ofrece menos información sobre A que I_1, pero de acuerdo al orden asociado por los valores de verdad tenemos que $I_1 \leqslant I_2$.

Subrahmanian resuelve estas dificultades en [70], donde introduce un nuevo lenguaje, a través de una variación del de los conjuntos de reglas cualitativas. Allí presenta la noción de *átomo y literal anotado.* Así, si A es un átomo (literal) y $\mu \in [0, 1]$ entonces $\langle A : \mu \rangle$ es un *átomo (literal) anotado.* Intuitivamente, μ puede ser interpretada como el *grado de creencia* en relación con A asociado a un juez (ó agente) calificador.

Las *cláusulas cualitativas* tienen la forma $\langle A : \mu \rangle \leftarrow \langle A_1 : \mu_1 \rangle \wedge \ldots \wedge \langle A_n : \mu_n \rangle$, donde $\langle A : \mu \rangle$ es un átomo anotado y los $\langle A_i : \mu_i \rangle \quad (1 \leqslant n)$ son literales anotados.

Un programa en lógica cualitativa, abreviadamente PLQ, es un conjunto finito de cláusulas cualitativas.

En la interpretación se permitirá negar átomos a través de la incorporación de una "*negación*" en el álgebra de anotaciones. Señalemos el caso particular de la versión original de Subrahmanian $I \models \neg\langle A : \mu \rangle$ si y sólo si $I \models \langle A : \sim \mu \rangle$, donde $\sim$ representa la función de negación, en el caso original $\sim \mu = 1 - \mu$. Sin duda alguna este es uno de los aspectos más relevantes de esta nueva propuesta.

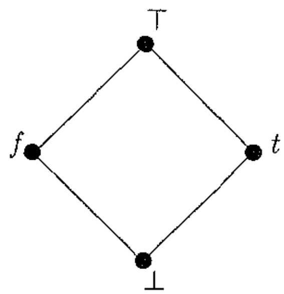
Una interpretación satisface una fórmula si satisface todas las instancias cerradas de ella. Luego I satisface un PLQ si satisface todas sus cláusulas cualitativas.

Como se puede percibir de esta breve introducción las lógicas anotadas surgen como una teoría de la programación lógica. Más aún la idea de Subrahmanian de presentar el conocimiento a través de átomos anotados sigue vigente y continúa su desarrollo.

[2]Letras proposicionales ó negaciones de estas.

En [15] se muestra que estas lógicas son adecuadas para formalizar la programación lógica con ciertas inconsistencias, sin que se vuelvan triviales como en el caso clásico. Si se quiere aquí está la principal aplicación de las lógicas anotadas y su principal motivación.

Técnicamente se define una operación $\sim$ sobre el conjunto de los valores de verdad con la cual se da cuenta de la negación de las anotaciones. Por ejemplo en [15] el conjunto de valores de verdad son tomados en el retículo Four,



Four

y la negación de anotaciones $\sim$ es definida por $\sim f = v$, $\sim v = f$, $\sim \top = \top$ y $\sim \perp = \perp$. Sin embargo, este operador en general puede ser cualquier función definida del conjunto de valores de verdad (que en principio se piensa como un retículo completo, con mínimo $\perp$ *el desconocimiento total* y máximo $\top$ *la inconsistencia total*) en sí mismo.

Es pertinente señalar que en un marco más general que la programación lógica las lógicas anotadas han cobrado su propia vida y continúa su estudio y desarrollo en distintos frentes, obviamente estando lo más cerca posible a las aplicaciones que son su génesis, su esencia está en representar conocimiento a través de *átomos anotados,* donde las anotaciones constituyen objetos tanto sintácticos como semánticos.

Las anotaciones pueden pensarse en un nivel *epistémico* del lenguaje, entendiendo el término epistémico en un sentido más amplio de lo usual, pues además de referirse a *conocimiento*, también ha de incluir lo relativo a *creencia.* Interpretaciones que se aclararán a lo largo del presente trabajo, éstas responden a la intuición de que un átomo anotado $\langle A : \mu \rangle$ puede ser interpretado como que *el valor de verdad de* A *es por lo menos* μ, pero también puede ser considerada como el grado de creencia de un juez (agente calificador), en este caso el átomo anotado $\langle A : \mu \rangle$ puede interpretarse como que *el juez acredita que el grado de veracidad de* A *es por lo menos* μ.

En términos técnicos las lógicas anotadas se obtienen de la lógica intuicionista positiva agregando una negación débil, pero siempre es posible definir dentro de ellas una negación fuerte, con lo cual se constituyen como extensiones de la lógica clásica. Sus versiones originales resultan ser paraconsistentes y paracompletas.

Recordemos que un sistema lógico es paracompleto si y sólo si existe una interpretación paracompleta, una interpretación para la cual existe una fórmula φ en el lenguaje tal que la interpretación no satisface a φ, ni a $\neg\varphi$. De esta forma el principio del tercero excluido no es válido en lógicas anotadas.

3.2. Relevancia de las lógicas anotadas

Las lógicas anotadas han dado cuenta de su importancia en las ciencias de la computación y la inteligencia artificial, pues constituyen un formalismo para el razonamiento en presencia de inconsistencias. Sólo a manera de ilustración señalemos unos cuantos autores, de la ya gran lista de investigadores a nivel mundial que han publicado trabajos acerca de las lógicas anotadas en estos contextos; Newton C.A. Da Costa (Universidad de Sao Paulo), Jair Minoro Abe (Instituto de Estudios Avanzados Universidad de Sao Paulo), Renato A. Lewin (Pontificia Universidad Católica de Chile), V. S. Subrahmanian (Department of Computer Science University of Maryland), A. Nerode (Mathematical Sciences Institute Cornell University), Michael Kifer (Department of Computer Science SUNY at Stony Brook), James J. Lu (Department of Computer Science Bucknell University), Neil V. Murray (Department of Computer Science State University of New York), Erik Rosenthal (Department of Mathematics University), Terrance Swift (Department of Computer Science University of Maryland), Mi Lu (Department of Electrical engineering texas A-M University), Tru H. Cao (School of Information Technology University of Queensland (Australia)), y Thom Früwirth de la Facultad de Informática de la Universidad Ulm (Alemania).

Dada la actualidad de las mismas creemos que el presente trabajo constituye un aporte a los desarrollos teóricos de las mismas, así como una ventana a posibles aplicaciones futuras.

3.3. Observaciones sobre algunas aplicaciones

Supongamos que fijamos un cierto lenguaje de primer orden, que φ es una fórmula de dicho lenguaje, y además que tenemos un conjunto parcialmente ordenado $\langle P, \leqslant_P\rangle$, abreviadamente como es usual nos referiremos a éste como un *copo*. Las anotaciones p las tomaremos en el copo P. En este contexto tenemos que una *fórmula anotada* es un par de la forma $\langle\varphi, p\rangle$.

Ejemplos

1. Sea $P = \langle [0,1], \leqslant \rangle$, donde $\leqslant$ representa el orden usual.

   a) La fórmula anotada $\langle \varphi, 1 \rangle$ sugiere la interpretación de que φ verdadera con certeza total.

   b) La fórmula anotada $\langle \varphi,\ 0.9\ \rangle$ en cambio nos permite las interpretaciones que φ es verdadera con certeza de 0.9 ó se tiene al menos un noventa por ciento de probabilidad de que φ sea verdadera.

   c) La fórmula anotada $\langle \varphi, 0 \rangle$ por lo contrario nos indica que no sabemos nada de la verdad de φ.

2. Sea $\langle P, \leqslant \rangle$ donde P es el conjunto de los intervalos cerrados contenidos en $[0,1]$; y el orden $\leqslant$ está definido por $[a,b] \leqslant [c,d] \Leftrightarrow a \leqslant c$ y $b \leqslant d$. En este caso una fórmula anotada como $\langle \varphi, [0.2, 0.8] \rangle$ nos sugiere la interpretación φ es verdadera con una confianza entre 0.2 y 0.8.

3. Sea $\langle [0,1] \times \mathcal{P}(T) \rangle$; con el orden obtenido del orden usual de los números reales en la primera componente, y de la inclusión de conjuntos en la segunda. Aquí T representa un conjunto de tiempos. Así una fórmula anotada $\langle \varphi, (p, S) \rangle$ nos sugiere la interpretación de que φ es verdadera con una confianza de al menos p en todos los tiempos t en S.

4. Sea $\langle [0,1] \times \mathcal{P}(T) \times \mathcal{P}(L) \rangle$; con el orden análogo al item anterior. Aquí T representa un conjunto de tiempos, y L representa un conjunto de lugares. Si tenemos además que φ representa "está lloviendo", entonces la fórmula anotada $\langle \varphi, (0.9, \{$sábado, domingo $\}, \{$Santiago, Valparaiso $\}) \rangle$ podría interpretarse como que hay un noventa por ciento de probabilidad que llueva entre Santiago y Valparaiso, entre los días sábado y domingo.

La relevancia en la interpretación de las conclusiones de las lógicas anotadas dependen exclusivamente de la destreza del anotador. Estas gozan de un amplio espectro de utilización, dentro del que podemos citar grados de credibilidad, probabilidad, calidad de datos, condiciones atmosféricas, y cualquier otro tipo de inferencias similares (recomendamos [51], [52] y [5]).

Las lógicas anotadas han sido fuertemente utilizadas en el manejo de bases de datos inconsistentes (ver [5], [12]). Insistamos que la Lógica Clásica en presencia de inconsistencias se trivializa, por tanto es necesario el uso de Lógicas que permitan el manejo de inconsistencias sin que se trivialicen, dentro de éstas tenemos las Lógicas de Predicados Anotadas (Annotated Predicate Calculus APC). La razón que se use ampliamente APC es por que las teorías clásicas pueden ser inmersas en APC en diversas formas. Las inmersiones más usadas son aquellas donde las teorías que son inconsistentes clásicamente son consistentes en APC.

Un importante problema es recuperar respuestas a consultas que son consistentes con las restricciones, aunque la base de datos (toda en conjunto) no satisfaga las restricciones. Muy seguramente la mayoría de los datos son consistentes todavía. Intuitivamente una tupla t es una respuesta consistente para una consulta de primer orden $Q(x)$ en una instancia (quizás inconsistente) de una base de datos relacional si es una respuesta (ordinaria) de $Q(x)$ en una reparación minimal (con respecto a la inclusión) de la base de datos.

APC provee un nuevo marco semántico declarativo para estudiar las consultas de bases de datos que son inconsistentes con sus restricciones. Esto se hace a través de una inmersión de la bases de datos y las restricciones en una teoría descrita en APC, para lo cual se requiere un álgebra apropiada de anotaciones.

En [5] se probó que existe una correspondencia entre algunos modelos mínimos de la teoría anotada y las reparaciones de la base de datos inconsistente para restricciones universales. En [12] se amplía a restricciones relacionales, además prueban como una consulta (anotada) es un problema de consecuencia no monótona de la teoría anotada. Sobre la base de la teoría anotada generada, se presentan programas de lógica de primer orden disyuntivos con argumentos anotados para especificar las bases de datos reparadas. También prueban la correspondencia entre algunos bien identificados modelos de los programas y las reparaciones de la instancia de la base de datos original; y se establece un camino para obtener respuestas consistentes de esos modelos. Hay implementaciones de tales programas en DVL.

Recientemente se ha abierto un interesante problema de investigación que relaciona las lógicas anotadas y el nuevo paradigma computacional de los lenguajes de programación con restricciones. Se trata de incrementar la expresividad de éstos lenguajes a través de las lógicas anotadas (ver [45],[66]).

Los anteriores comentarios muestran la amplitud en los usos de las lógicas anotadas, más si se piensa que uno podría anotar con cualquier tipo de conjunto con la estructura que se nos ocurra. Así en las ilustraciones anteriores mostramos ejemplos con conjuntos de anotaciones con estructura de orden, las que nos permiten hacer comparaciones entre los elementos de dicho conjunto, y aprovechando este hecho podríamos introducir una semántica basada en estas comparaciones, es decir, definir una semántica usando la relación de orden dada.

Buscando una presentación más general Da Costa, Subrahmanian y Vago [40] presentan las lógicas P_τ en las cuales se consideran las anotaciones en conjuntos con estructura de retículo. A continuación presentaremos los principales aportes de estas lógicas.

3.4. Las lógicas P_τ

Los sistemas P_τ que presentaremos aquí dan cuenta de la formalización del año 1991 [40], aunque ellas realmente aparecen algunos años antes en una serie de aplicaciones en programación lógica como lo señalamos en nuestra introducción.

3.4.1. El sistema Pτ

Consideremos un retículo completo τ fijo, y una función $\sim\ : \tau \longmapsto \tau$ también fija. Dentro del retículo τ, como es usual usaremos los símbolos $\perp$ y $\top$ para denotar el mínimo y el máximo del retículo. En cuanto a la función $\sim$ la usaremos a fin de representar una negación para las fórmulas anotadas. Tanto el retículo, como la función son en principio absolutamente arbitrarios. De está manera realmente estaremos hablando de infinitos sistemas lógicos, puesto que cada sistema depende del retículo τ y la función $\sim$.

El lenguaje del sistema P_τ consiste de símbolos lógicos $\wedge,\vee,\rightarrow,\neg$, un conjunto $\mathcal{P}$ de variables proposicionales p, q, r,..., constantes de anotación λ,μ,..., para cada elemento del retículo τ y paréntesis. El conjunto $\mathcal{F}$ de las fórmulas de P_τ se define recursivamente.

1. Si p es una letra proposicional y λ es una constante de anotación, p_λ es una fórmula.
2. Si A es una fórmula, entonces $\neg A$ es una fórmula.
3. Si A y B son fórmulas, entonces $(A \wedge B)$, $(A \vee B)$ y $(A \rightarrow B)$ son fórmulas.
4. Una expresión es una fórmula si y sólo si se obtiene usando sólo las reglas antes citadas.

Las fórmulas $\neg\neg\cdots\neg p_\mu$, cuando hay k símbolos de negación, se denotará por $\neg^k p_\mu$, $k \geq 0$ y se llamarán *hiperliterales*. Una *fórmula compleja* es una fórmula que no es un hiperliteral.

La presentación de P_τ está dada por los axiomas y reglas de inferencia que listamos a continuación.

$(\rightarrow_1)$ $A \rightarrow (B \rightarrow A)$,

$(\rightarrow_2)$ $((A \rightarrow (B \rightarrow C)) \rightarrow ((A \rightarrow B) \rightarrow (A \rightarrow C))$,

$(\rightarrow_3)$ $((A \rightarrow B) \rightarrow A) \rightarrow A$,

$(\to_4)$ $\dfrac{A, A \to B,}{B}$

$(\wedge_1)$ $(A \wedge B) \to A$,

$(\wedge_2)$ $(A \wedge B) \to B$,

$(\wedge_3)$ $A \to (B \to (A \wedge B))$,

$(\vee_1)$ $A \to (A \vee B)$,

$(\vee_2)$ $B \to (A \vee B)$,

$(\vee_3)$ $(A \to C) \to ((B \to C) \to ((A \vee B) \to C))$.

Si F y G son fórmulas complejas,

$(\neg_1)$ $(F \to G) \to ((F \to \neg G) \to \neg F)$,

$(\neg_2)$ $F \to (\neg F \to A)$,

$(\neg_3)$ $F \vee \neg F$,

(τ_1) $p_\perp$,

(τ_2) $\neg^k p_\mu \leftrightarrow \neg^{k-1} p_{\sim\mu}$, para $k \geq 1$,

(τ_3) $p_\mu \to p_\lambda$, para $\mu \geq \lambda$,

(τ_4) Si $A \to p_{\mu_j}$ para cada $j \in J$, entonces $A \to p_\mu$, donde $\mu = \bigvee_{j \in J} \mu_j$.

Las definiciones de prueba, consecuencia, $\vdash_{P_\tau}$ y teoremas son definidos en la forma usual. Un conjunto Δ de fórmulas se dice *trivial* si para toda fórmula $A \in \mathcal{F}$, $\Delta \vdash A$. Un conjunto Δ se dice *inconsistente* si existe una fórmula A tal que $\Delta \vdash A$ y $\Delta \vdash \neg A$. Existen conjuntos de fórmulas que son inconsistentes pero no triviales.

Observación 3.1 *1. Los axiomas $\neg_1$, $\neg_2$ y $\neg_3$ dan cuenta que el comportamiento de las fórmulas complejas es similar al de la lógica proposicional clásica. La distancia con la lógica clásica está marcada en el manejo de los átomos anotados, legislado con los axiomas τ_1 al τ_4.*

2. Consideramos muy pertinente señalar que el sistema P_τ no es estructural; así si σ es a sustitución tal que $\sigma(p_\perp) = p_\lambda$, para $\lambda \neq \perp$, se tiene:

$$\vdash_{P_\tau} p_\perp \text{ pero } \nvdash_{P_\tau} \sigma(p_\perp),$$

lo cual se puede verificar a partir del teorema 3.4.

3. *Las fórmulas complejas tienen un comportamiento clásico en el siguiente sentido. Si $\varphi(x_1, \ldots, x_n)$ es una tautología clásica y $A_1, \ldots, A_n$ son fórmulas complejas, entonces $\vdash_{P_\tau} \varphi(A_1, \ldots, A_n)$.*

4. *En la regla τ_4, si $J = \emptyset$ entonces $\bigvee_{j \in J} \mu_j = \perp$ y por los axiomas τ_1, $\rightarrow_1$ y modus ponens $A \rightarrow p_\perp$. Es decir, una consecuencia de la regla τ_4 es uno de los axiomas del sistema P_τ. Así nosotros asumiremos que $J \neq \emptyset$.*

5. *Una particularidad de P_τ está en el axioma τ_4, el cual puede tener una instancia infinita si el retículo es infinito.*

Se obtiene un resultado similar al teorema deducción en el caso clásico, el cual es una consecuencia de los dos primeros axiomas y de que la regla de inferencia es modus ponens, más la verificación para el mismo de la regla τ_4.

Teorema 3.1 Σ, $A \vdash_{P_\tau} B$ *si y sólo si* $\Sigma \vdash_{P_\tau} A \rightarrow B$.

3.4.2. Semántica para el sistema P_τ

Una *interpretación* I para el sistema P_τ es una función $I : \mathcal{P} \longrightarrow \tau$. Para cada interpretación se define una *evaluación* $v_I : \mathcal{F} \longrightarrow \{0, 1\}$ por:

1. Si $p \in \mathcal{P}$, entonces $v_I(p_\mu) = 1$ sii $I(p) \geq \mu$.

2. *a*) $v_I(\neg^k p_\mu) = v_I(\neg^{k-1} p_{\sim\mu})$, para $k \geq 1$,
   b) Si A es compleja, entonces $v_I(\neg A) = 1 - v_I(A)$.

3. Si A, $B \in \mathcal{F}$, entonces

   a) $v_I(A \rightarrow B) = 1$ sii $v_I(A) = 0$ ó $v_I(B) = 1$,
   b) $v_I(A \vee B) = 1$ sii $v_I(A) = 1$ ó $v_I(B) = 1$,
   c) $v_I(A \wedge B) = 1$ sii v_I(A) = 1 y v_I(B) =1.

Observación 3.2 *Si se tiene que $I(p) = \perp$, entonces $v_I(p_\lambda \wedge \neg p_\lambda) = 1$, es decir, el sistema es paraconsistente en el sentido que existen contradicciones que son "verdaderas"bajo ciertas interpretaciones y así $(A \wedge \neg A) \rightarrow B$ no es un teorema en P_τ.*

Sin embargo, si F es compleja, entonces bajo toda interpretación I, siempre tendremos que $v_I(F \wedge \neg F) = 0$. En efecto, si $\varphi(x_1, \ldots, x_n)$ es una contradicción clásica y $A_1, \ldots, A_n$ son fórmulas complejas, entonces $\varphi(A_1, \ldots, A_n)$ es falsa bajo toda interpretación.

Las nociones de consecuencia semántica, $\models$, fórmula valida, etc., se definen en el sentido usual. Finalizamos esta breve presentación listando algunos de los principales resultados de las lógicas $P\mathcal{T}$.

Teorema 3.2 *Si $\Sigma \vdash_{P_\tau} A$, entonces $\Sigma \models A$.*

Lema 3.3 *Si A es no trivial, entonces tiene un modelo.*

Teorema 3.4 *1. Si Σ es finito, entonces $\Sigma \vdash_{P_\tau} A$ sii $\Sigma \models A$.*

2. Si τ es finito, entonces $\Sigma \vdash_{P_\tau} A$ sii $\Sigma \models A$.

Las lógicas $P\mathcal{T}$ están dentro del marco de las lógica proposicionales, sin embargo no es difícil extender estas lógicas proposicionales a lógicas de primer orden.

Para precisar el anterior comentario presentaremos los sistemas Q_τ, que dan cuenta de lógicas anotadas de primer orden.

3.5. Las lógicas anotadas Q_τ

El sistema Q_τ es la lógica anotada básica de primer orden, en la cual igual que en el caso proposicional τ es un retículo completo, al cual en general por efecto de las aplicaciones generalmente se le agrega la condición de ser distributivo. Igualmente $\sim$ es una función unaria sobre τ. El lenguaje $\mathcal{L}$ de Q_τ es un lenguaje de primer orden, que como es lo usual consta de unos símbolos comunes. Los signos de *variables,* los *signos lógicos* para los conectivos, la relación de identidad y los cuantificadores, y los *signos técnicos* ó auxiliares, como son los paréntesis. Sin embargo es usual incluir, por fuera de la formalidad, los signos de puntuación de nuestro lenguaje natural. Así los símbolos comunes los podemos precisar en la siguiente lista.

(i) Variables: $u, v, x, y, z, u_1, u_2, u_3, \ldots$

En general, las últimas letras minúsculas del alfabeto latino, si es necesario con subíndices.

(ii) Conectivos: $\vee, \wedge, \rightarrow, \leftrightarrow, \neg$.

(iii) Cuantificadores: El cuantificador universal $\forall$, y el cuantificador existencial $\exists$.

(iv) Identidad: $\approx$, por simplicidad suprimiremos este símbolo de nuestro lenguaje.

(v) Los paréntesis:) y (.

En cuanto a los símbolos no comunes, tenemos los de *predicado,* para referirnos a las relaciones entre los objetos, los de *función* muy propios de ciertas relaciones en el uso de las matemáticas preferiblemente, y los de *constantes,* para referirse a objetos individuales. El conjunto de todos los símbolos no comunes usualmente se denomina *léxico de primer orden,* y se puede detallar por:

- Un conjunto R de símbolos de relación (ó símbolos de predicado). A cada uno de los cuales se le asigna un natural, denominado su aridad, para indicar el número de "argumentos"que éste permite. Cuando este número sea n diremos que el predicado es *n-ario.*
- Un conjunto F de símbolos de función[3]. Al igual que para los símbolos de predicado, cada uno de los símbolos de función tiene asociado un número natural asociado, para indicar el número de "argumentos"que éste permite. Este número igual que para los símbolos de predicados, recibirá el nombre de aridad de f, y lo recalcaremos diciendo que f es *n-ario.*
- Un conjunto C de símbolos de constantes.

Usaremos la letra L para denotar el léxico formado por los conjuntos R, F, y C.

Los *términos* de L se definen recursivamente por:

1. Toda variable es un término de L.
2. Todo símbolo constante es un término de L.
3. Si f es una función n-aria y $t_1, t_2, ..., t_n$ son términos de L, entonces $f(t_1, t_2, ..., t_n)$ es un término de L.
4. Los términos se forman únicamente por los items anteriores.

Diremos que un término es *cerrado* si no tiene variables. A partir de los términos se definen las fórmulas de primer orden más simples que llamaremos atómicas, para después si definir las fórmulas de primer orden en general.

Las *fórmulas atómicas* de L son las expresiones de la forma $R(t_1, t_2, ..., t_n)$, donde R es un símbolo de relación n-ario; y $t_1, t_2, ...t_n$ son términos de L.

Las *fórmulas de primer orden (f.p.o.)* de L:

1. Toda fórmula atómica es una f.p.o.

[3]En algunas presentaciones de lenguajes de primer orden no aparecen; incluso estos podrían pensarse dentro de los símbolos de predicado.

2. Si α es una f.p.o. entonces $\neg(\alpha)$ también lo es.

3. Si α y β son f.p.o. entonces $(\alpha)\wedge(\beta)$, $(\alpha)\vee(\beta)$, $(\alpha)\rightarrow(\beta)$ y $(\alpha)\leftrightarrow(\beta)$ también lo son.

4. Si α es una f.p.o. y x es una variable entonces $(\forall x)\alpha$ y $(\exists x)\alpha$ también lo son.

5. Las f.p.o. se forman únicamente por los items anteriores.

Si P es un símbolo de predicado de aridad n y $\lambda \in \tau$ entonces al par $\langle P, \lambda\rangle$ lo llamaremos un *predicado anotado*, se denotara por P_λ.

Dado un predicado anotado P_λ de aridad n y n términos $t_1, \ldots, t_n$, un *átomo anotado* es una expresión de la forma $P_\lambda(t_1, \ldots, t_n)$. La noción de fórmula de primer orden es análoga a f.p.o. tomando como base los átomos anotados en lugar de las fórmulas atómicas.

3.5.1. Interpretaciones

Una interpretación para un léxico de primer orden $L = (R,F,C)$ esta dada por un par $I = (D,i)$ donde D es un conjunto no vacío llamado *dominio* de I, e i es una aplicación que asocia:

1. A cada símbolo constante $c \in C$, un elemento c^i en D.

2. A cada símbolo n-ario de función $f \in F$, una función n-aria $f : D^n \longrightarrow D$.

3. A cada símbolo n-ario de relación $P \in R$, una relación[4] n-aria $P^i : D^n \rightarrow \tau$.

Una *asignación* en una interpretación $I = (D,i)$ es una función $A : V \longrightarrow D$, definida del conjunto de las variables en el dominio de la interpretación I, y para la cual usaremos la notación $A(x) = x^A$. Además usaremos el signo $\Box$ para la asignación vacía.

Si $I = (D,i)$ es una interpretación para L y A una asignación en esa interpretación, entonces a cada término t de L se asocia el valor $t^{i,A}$ en D a través de las siguientes indicaciones:

1. Para un símbolo de constante c, $c^{i,A} = c$.

2. Para un símbolo de función n-aria f y términos $t_1, t_2, \ldots, t_n$ sobre L $[f(t_1, t_2, \ldots, t_n)]^{i,A} = f(t_1^{i,A}, t_2^{i,A}, \ldots, t_n^{i,A})$.

[4]En el caso clásico se asocia una relación n-aria P^i, es decir, $P^i \subseteq D^n$.

3. Para una variable x, $x^{i,A} = x^A$.

Observación 3.3 *Si t no tiene variables, y hacemos $t^i = t^{i,\Box}$, entonces $t^{i,A} = t^i$.*

Nuestro siguiente objetivo será atribuirle valores de significado en $\{0, 1\}$ a las fórmulas anotadas de primer orden de L, relativos a una interpretación y una asignación, para lo cual es indispensable la siguiente definición.

Si x es una variable y A, B son asignaciones en $I =$ *(D,i)*, diremos que A y B son *x-variantes* sí, y sólo si, coinciden para toda variable excepto posiblemente en x.

Observación 3.4 *Las asignaciones x-variantes con la asignación vacía $\Box$, son todas asignaciones que modifican a lo más únicamente a la variable x.*

Dada $I =$ *(D,i)* una interpretación para L y A una asignación en I. A cada fórmula anotada de primer orden de L se asocia un único valor de verdad de la siguiente forma:

1. Para un átomo anotado $P_\tau(t_1, t_2, ..., t_n)$ tenemos que $[P_\tau(t_1, t_2, ..., t_n)]^{i,A} = 1$ sí, y sólo si, $P^i(t_1^{i,A}, t_2^{i,A}, ..., t_n^{i,A}) \geqslant \tau$.
2. Para un hiperliteral $\neg^k(P_\mu)$ tenemos que $[\neg^k(P_\mu)]^{i,A} = [\neg^{k-1}(P_{\sim\mu})]^{i,A}$.
3. $\neg[\varphi]^{i,A} = \neg[\varphi^{i,A}]$, cuando φ no es un hiperliteral.
4. $[\varphi \diamond \psi]^{i,A} = \varphi^{i,A} \diamond \psi^{i,A}$, para $\diamond$ un conectivo binario.
5. $[\forall x\varphi]^{i,A} = 1$ sí, y sólo si, $\varphi^{i,B} = 1$, para toda asignación B en I que es x-variante de A.
6. $[\exists x\varphi]^{i,A} = 1$ sí, y sólo si, $\varphi^{i,B} = 1$, para alguna asignación B en I que es x-variante con A.

Observación 3.5 *Usando la notación $\varphi^i = \varphi^{i,\Box}$ se tiene que $(\forall x\varphi)^i = 1$ sí, y sólo si, $\varphi^{i,B} = 1$ para cada asignación B que modifica únicamente la variable x. Además si φ es una sentencia, $\varphi^{i,A} = \varphi^i$.*

Al igual que en el caso clásico tenemos las definiciones estándar de *validez, consecuencia semántica,* y algunas abreviaciones ampliamente usadas como las dos siguientes.

Si φ, ψ son fórmulas de $\mathcal{L}$, entonces $\varphi \leftrightarrow \psi$ abrevia la expresión $(\varphi \to \psi) \wedge (\psi \to \varphi)$, mientras que $\neg\varphi$ abrevia a $(\varphi \to (\varphi \to \varphi)) \wedge \neg(\varphi \to \varphi)))$. La expresión $\neg\varphi$ recibe el nombre de negación fuerte de φ en Q_τ.

Ahora daremos una presentación para el sistema Q_τ. Consideraremos a α, β, γ como fórmulas cualesquiera, φ y ψ denotarán fórmulas complejas y θ es un átomo anotado. Donde t está sujeto a las restricciones usuales, y también al usar la variable x mantenemos las restricciones de libertad estándar.

$(\rightarrow_1)$ $\alpha \rightarrow (\beta \rightarrow \alpha)$

$(\rightarrow_2)$ $(\alpha \rightarrow \beta) \rightarrow ((\alpha \rightarrow (\beta \rightarrow \gamma)) \rightarrow (\alpha \rightarrow \gamma))$

$(\rightarrow_3)$ $\alpha,\ \alpha \rightarrow \beta / \beta$

$(\rightarrow_4)$ $((\alpha \rightarrow \beta) \rightarrow \alpha) \rightarrow \alpha$

$(\wedge_1)$ $\alpha \wedge \beta \rightarrow \beta$

$(\wedge_2)$ $\alpha \wedge \beta \rightarrow \alpha$

$(\wedge_3)$ $\alpha \rightarrow (\beta \rightarrow \alpha \wedge \beta)$

$(\vee_1)$ $\alpha \rightarrow (\alpha \vee \beta)$

$(\vee_2)$ $\beta \rightarrow (\alpha \vee \beta)$

$(\vee_3)$ $(\alpha \rightarrow \gamma) \rightarrow ((\beta \rightarrow \gamma) \rightarrow (\alpha \vee \beta \rightarrow \gamma))$

$(\neg_1)$ $(\varphi \rightarrow \psi) \rightarrow ((\varphi \rightarrow \neg\psi) \rightarrow \neg\varphi))$

$(\neg_2)$ $\varphi \rightarrow (\neg\varphi \rightarrow \alpha)$

$(\neg_3)$ $\varphi \vee \neg\varphi$

$(\exists_1)$ $\alpha(t) \rightarrow \exists x \alpha(x)$

$(\exists_2)$ $\frac{\alpha(x) \rightarrow \beta}{\exists x \alpha(x) \rightarrow \beta}$

$(\forall_1)$ $\forall x \alpha(x) \rightarrow \alpha(t)$

$(\forall_2)$ $\frac{\alpha \rightarrow \beta(x)}{\alpha \rightarrow \forall x \beta(x)}$

(τ_1) $(\theta_\perp) \wedge \neg^k(\theta_\mu) \leftrightarrow \neg^{k-1}(\theta_{\neg\mu})$

(τ_2) $(\theta_{\neg\mu}) \rightarrow (\theta_{\neg\lambda})$ donde $\lambda \leq \mu$

(τ_3) Si $\alpha \rightarrow (\theta_{\mu_j})$ para todo $j \in J$, entonces $\alpha \rightarrow (\theta_\mu)$ donde $\mu = sup\{\mu_j : j \in J\}$

Dado que τ es un retículo completo, el supremo en (τ_3) está bien definido. Más aún, cuando τ es finito la regla (τ_3) puede ser reemplazada por $\bigwedge_{j=1}^{j=n}(\theta_{\mu_j}) \to (\theta_\mu)$ donde $\mu = sup\{\mu_j : 1 \leq j \leq n\}$. La definición de consecuencia sintáctica $\vdash$ se introduce como es usual. Estas lógicas anotadas también resultan ser paraconsistentes.

Resulta más o menos estándar dar una prueba al teorema de *validez* para Q_τ con respecto a la semántica presentada. Es decir, para $\Gamma \cup \{\alpha\}$ un conjunto de fórmulas de Q_τ, se cumple que si α se deduce de Γ a partir del sistema deductivo, entonces α es una consecuencia lógica de Γ.

Capítulo 4

Revisión a las lógicas anotadas

Intentaremos presentar una nueva, y quizás más profunda, revisión a las lógicas anotadas en relación con las ciencias de la computación. Más precisamente, en relación con la programación lógica y la "computación suave" (soft computation). Revisión que justifica la necesidad de nuevos desarrollos teóricos en esta linea de investigación.

Cuando en el año 1989 Subrahmanian, da Costa y Vago diseñan la lógica P_τ sin lugar a dudas querían dar respuesta a una problemática presentada en el seno de la programación lógica. Una mirada a los acontecimientos recientes muestra que la propuesta de P_τ no da solución a la totalidad de las dificultades encontradas en su momento en la programación lógica, más aún tales dificultades han ido en incremento con los desarrollos posteriores en las áreas cercanas a la programación lógica como la computación suave. Esto, sumado al desarrollo de las diferentes lógicas no clásicas multivaluadas en la vecindad mencionada, ha dado pie a gestar una nueva etapa de la programación lógica. Siendo necesario consolidar propuestas actualizadas de los desarrollos lógicos que proporcionen herramientas sólidas que sustenten tal etapa.

No es fácil definir, ni si quiera acotar en forma precisa cuáles son estos problemas a solucionar, pues como tales no están planteados. Por el contrario, sólo la revisión de los desarrollos de esta línea de investigación (programación lógica + computación suave) permite exhibir la aparición de nuevas dificultades. Por tanto, nuestro problema desborda la posibilidad de una solución definitiva. Se trata de acompañar la vanguardia de las aplicaciones de las área de conocimiento en mención con mejores y más apropiados sustentos teóricos. Nuestra propuesta es desarrollar lógicas anotadas de segunda generación que eventualmente podrían soportar algunas aplicaciones. En esta perspectiva se hace necesario cuidar el buen comportamiento de tales lógicas, y tenemos por recurso teórico el de los métodos algebraicos de la lógica, a través de los procesos de algebrización.

En este capítulo reseñamos una serie de artículos con los que se evidencia la aparición de la nueva etapa en la programación lógica antes mencionada. En los mismos se detecta igualmente la necesidad de llevar las lógicas anotadas por nuevos desarrollos. Con el objeto de precisar los comentarios por venir y buscando un carácter autocontenido, hemos decidido hacer un breve recorrido por los principales elementos de la programación lógica hasta poder detallar el principio de resolución de Robinson y precisar el concepto de hiper-resolución. Esto nos permitirá cierta solvencia en nuestras reseñas. Además presentaremos una serie de observaciones generales, producto de esta investigación, acerca de los procesos de algebrización que nos garanticen el buen comportamiento de las lógicas a construir, por lo menos desde el punta vista algebraico.

4.1. Programación Lógica

Uno de los temas actuales de mayor interés referidos a las ciencias computacionales, la lógica y las matemáticas mismas es la deducción automática. Siendo su eje articulador el principio de resolución lógica de Robinson que, en términos generales, soporta y sustenta los procedimientos de deducción automática. Queremos hacer un rápido recorrido de las ideas centrales del razonamiento automático; una breve revisión de cómo se llegó a los procedimientos de resolución desde Leibniz hasta Robinson, para después presentar algunos ejemplos de resolución lógica más generales, y citar ciertos sistemas experimentales para la prueba de teoremas.

El uso de sistemas formales permite capturar ciertos esquemas de razonamiento a través de de cálculo simbólico, es decir, éstos razonamientos pueden ser tomados como procedimientos combinatorios, finitos y puramente sintácticos. La idea de que "el razonamiento es como el cálculo", debida a Leibniz, toma forma y se materializa con el uso de las computadoras.

En 1936 Church y Turing probaron que no existe un procedimiento general de decisión para constatar la validez de las formulas de primer orden en lógica. Sin embargo, hay procedimientos de prueba que pueden verificar que una formula es válida si en realidad lo es. Para formulas inválidas, estos procedimientos en general nunca terminarán. Ya en 1930 Herbrand había desarrollado un algoritmo para encontrar una interpretación que pueda falsear una fórmula dada. Sin embargo, si la formula dada es realmente valida, tal interpretación no existirá y su algoritmo va a parar después de un número finito de pruebas. El método de Herbrand es la base de los procedimientos automáticos de prueba más modernos.

En 1960 Gilmore consigue implementar, aunque no muy eficientemente, los procedimientos de Herbrand en una computadora, método que es mejorado sustancialmente por Davis y Putnam en 1960. Sin embargo el avance mayor se realiza a través de Robinson en 1965, al introducir el Principio de Resolución. Éstos son procedimientos de refutación, es decir, en lugar de probar la validez de una formula se prueba que la negación de una formula es insatisfacible.

Con el principio de resolución de Robinson se da origen a la *demostración automática* de teoremas, que gozó de gran interés en los años setenta al inicio de la llamada Inteligencia Artificial (IA). Las aplicaciones más eficientes y de mayor aplicación, dentro del marco de la IA, se inician con el origen de Prolog (1971) en la Universidad de Marsella (Francia). Prolog es un ejemplo de lenguaje basado en la Lógica de Primer Orden y aunque toma su nombre de la expresión "PROgramming in LOGic" hace uso de una estrategia de refinamiento denominada Resolución-SLD, que abrevia "Linear resolution with Selection function for Definite clauses".

La equivalencia entre semántica declarativa y semántica procedimental para programas definidos es básicamente la validez y la completitud de resolución SLD. La SLDNF-resolución esta pensada como la contraparte de un procedimiento de la semántica declarativa. Esquema de resolución que es la base actual de la programación lógica. Es esencialmente una resolución SLD aumentada por la regla de inferencia NAF (Negation as Failure). La definición de SLDNF-resolución es una versión inductiva formal como opuesta a la versión recursiva.

Actualmente la programación lógica ha cobrado un renovado interés al hacer converger las ciencias computacionales, la lógica y las matemáticas. Interés que está respaldado por un sinnúmero de aplicaciones que incluyen las bases de datos y lenguajes de programación. Esto significa que no sólo sirven para demostrar teoremas o argumentos de Matemática o Filosofía, sino que también se pueden aplicar a las áreas de Ingeniería, Control industrial y Computación.

Dentro de los demostradores de teoremas de aplicación experimental, se destacan MACE, FINDER, MGTP, LDPP, SATO, SEM y el sistema OTTER (Organized Techniques for Theorem-proving and Effective Research). Con algunos de los cuales se ha experimentado en la aplicación a la validación y diseño de circuitos, corrección de programas, y en general a problemas de inferencia modelados en lógica de primer orden.

En lo que resta de esta sección haremos un rápido recorrido de los elementos básicos necesarios para presentar el concepto de hiper-resolución.

4.1.1. Conjunto estándar

Una fórmula de primer orden φ está en una forma normal prenexa (FNP) si φ tiene la forma $(Q_1x_1)...(Q_nx_n)M$ donde $Q_1, ..., Q_n$ son cuantificadores ($\forall$ ó $\exists$) y M

no contiene ningún cuantificador, $(Q_1x_1)...(Q_nx_n)$ es llamado el *prefijo* de φ y M la *matriz* de φ.

Para obtener una *forma estándar de Skolem* se parte de una forma normal prenexa y la matriz se pasa a Forma Normal Conjuntiva (FNC). Tendremos entonces una fórmula donde $(Q_1x_1)...(Q_nx_n)$ son cuantificadores; $x_1, ..., x_n$ variables (distintas) y M una matriz en FNC. En seguida se eliminan todos los cuantificadores existenciales mediante el proceso llamado skolemización, que describimos a continuación. Sea Q_m es un cuantificador existencial del prefijo de φ. Si antes de Q_m no hay ningún cuantificador universal, se escoge un símbolo de constante c, que no aparecía antes en la fórmula, se reemplaza por todas las ocurrencias de x_m y se borra Q_mx_m del prefijo. En caso contrario, sean $Q_{s_1}, ..., Q_{s_r}$ todos los cuantificadores universales que aparecen a la izquierda de Q_m. Escogemos un nuevo símbolo de función r-ario, f, se reemplazan todas las ocurrencias de x_m por $f(x_{s_1}, ..., x_{s_r})$ y se elimina Q_mx_m del prefijo. El proceso termina cuando se ha aplicado a todos los cuantificadores existenciales. Constantes y símbolos de función añadidos son llamados *funciones de Skolem.*

Una fórmula de primer orden, φ, no es necesariamente equivalente a su forma estándar de Skolem, S, sin embargo, se demuestra (*teorema de Skolem*) que S es insatisfacible, si y sólo si, φ es insatisfacible.

Un *literal* es un predicado atómico, o la negación de un predicado atómico. Una *cláusula* en la Lógica de Primer Orden es una disjunción finita de cero o más literales. Mientras que una cláusula proposicional es una cláusula en la Lógica Proposicional, es decir, una disjunción finita de letras proposicionales ó negaciones de letras proposicionales.

Diremos que una cláusula es *unitaria,* si solo tiene un literal. La cláusula vacía (que tiene cero literales) se escribirá $\square$, y se entenderá como falsa en cualquier interpretación (ya que una cláusula se interpreta en verdadero si existe al menos un literal que se interpreta en verdadero). Simplifiquemos ahora la representación de una forma estándar de Skolem, de la siguiente manera. Primero, los cuantificadores $(\forall x_1)...(\forall x_m)$ se pueden omitir, pues sabemos que cualquier variable que aparezca está cuantificada universalmente. Segundo, puesto que $\mathrm{M} = C_1 \wedge ... \wedge C_n$, ya que se encuentra en FNC, donde cada C_i es una cláusula. Podemos representar la forma de Skolem mediante el conjunto de sus cláusulas, $S = \{C_1, C_2, ..., C_n\}$. A S lo llamaremos un conjunto estándar de cláusulas (o simplemente, conjunto estándar).

4.1.2. Interpretaciones de Herbrand

Sea S un conjunto estándar de cláusulas, siempre que exista al menos un símbolo de constante en S, se define $H_0 = \{c : c \text{ es un símbolo de constante en } S\}$; en caso contrario se define $H_0 = \{a\}$, con a un símbolo de constante cualquiera. Recursivamente definimos H_{n+1} como la unión entre H_n y el conjunto de todos los términos de la forma $f(t_1, t_2, ..., t_n)$ donde f es un símbolo de función $n-$ario que aparece en S, y $t_1, t_2, ..., t_n \in H_n$. Finalmente el universo de Herbrand (H) de S es la unión de todos los H_n.

Sea S un conjunto estándar y H su universo de Herbrand. La base de Herbrand (B) de S, es el conjunto de todas las instancias básicas de los átomos que aparecen en S, es decir, B es el conjunto de todos los predicados de la forma $P(t_1, ..., t_n)$ donde P es un símbolo de predicado en S y $t_1, ..., t_n$ son términos en H.

Una interpretación de Herbrand es una interpretación que tiene por universo a H tal que todas las constantes de S son interpretadas como ellas mismas, y si f es un símbolo de función n-aria, y $h_1, ..., h_n$ términos de H, la interpretación de $f(h_1, ...h_n)$ es $f(h_1, ..., h_n)$.

Se demuestra (*teorema de Herbrand*) que una fórmula φ es insatisfacible si y sólo si, φ es falsa en todas las interpretaciones de Herbrand.

4.1.3. Resolución en lógica proposicional

Un par de literales L_1 y L_2 se dicen *complementarios* si $L_1 = \neg L_2$ o $L_2 = L_1$. El principio de resolución consiste básicamente en la aplicación, sobre un conjunto de cláusulas, de las siguientes reglas de inferencia:

Regla de Resolución Binaria

Dadas dos cláusulas, C_1 y C_2, si hay un par complementario, digamos P y $\neg P$ donde P es un literal de C_1 y P un literal de C_2, se infiere C_3 así:

$$\begin{array}{l} C_1 : F_1 \vee P \vee G_1 \\ C_2 : F_2 \vee \neg P \vee G_2 \\ \hline C_3 : F_1 \vee G_1 \vee F_2 \vee G_2 \end{array}$$

La cláusula C_3 es llamada una resolvente binaria de C_1 y C_2, y es de nuevo una cláusula, pues F_1, G_1, F_2 y G_2 son cláusulas (posiblemente vacías). El par complementario, P y $\neg P$ no aparece en la conclusión, y se dice que P y $\neg P$ son los literales resueltos.

Regla de Factorización

Dada una cláusula $C : F_1 \vee P \vee F_2 \vee P \vee F_3$ donde P es un literal, C' se infiere así:

$$\frac{C : F_1 \vee P \vee F_2 \vee P \vee F_3}{C' : F_1 \vee P \vee F_2 \vee F_3}$$

C' se denomina un factor de C.

Regla de Resolución

Dadas dos cláusulas C_1 y C_2, llamadas cláusulas padres, la aplicación de la regla de resolución sobre estas cláusulas consiste en uno de los cuatro casos siguientes:

1. Resolución binaria entre C_1 y C_2.
2. Resolución binaria entre C_1 y un factor de C_2.
3. Resolución binaria entre un factor de C_1 y C_2.
4. Resolución binaria entre un factor de C_1 y un factor de C_2.

El resultado es llamado una resolvente de C_1 y C_2.

Esta regla es muy poderosa. De hecho se extiende a la lógica de predicados de primer orden y se muestra la completitud en el siguiente sentido: si un conjunto de cláusulas es insatisfacible, entonces es posible inferir de allí la cláusula vacía.

Deducción

Dado un conjunto S de cláusulas, una deducción de C a partir de S es una secuencia $C_1, C_2, ..., C_k$ tal que $C_k = C$ y cada C_i es bien una cláusula de S, o se obtiene por la aplicación de la regla de resolución a cláusulas anteriores a C_i. Si C es la cláusula vacía entonces a esta deducción se le llama refutación de S. Convendremos en lo que resta de éste capítulo en usar la notación $S \vdash^R C$ para denotar que S es un conjunto a partir del cual se puede deducir C usando la regla de resolución. Además si S y T son conjuntos notamos $S \vdash^R T$ si $S \vdash^R t$ para cada $t \in T$.

Validez de Resolución en la Lógica proposicional

Sean S un conjunto estándar de cláusulas proposicionales y C una cláusula proposicional. Si $S \vdash^R C$ entonces C es es una consecuencia lógica de S $(S \models C)$.

El recíproco no es cierto, puesto que $p \nvdash^R p \vee q$ y $p \models p \vee q$. Pero es cierto en cuanto a satisfacibilidad, es decir, si $\Gamma \subset \mathcal{D}(p_1, ..., p_n)$, el conjunto de todas las cláusulas que contienen sólo las letras proposicionales $p_1, ..., p_n$, entonces Γ es insatisfacible si y solo si $\Gamma \vdash^R \square$.

Ejemplo 4.1 *Dado el conjunto estándar S,* *1.* $P \vee Q$ *de* S

2. $\neg P \vee Q$ *de* S

3. $P \vee \neg Q$ *de* S

4. $\neg P \vee \neg Q$ *de* S

Generamos los siguientes resolventes

5. Q *de* (1) *y* (2)

6. $\neg Q$ *de* (3) *y* (4)

7. $\square$ *de* (5) *y* (6)

Dado que se derivó $\square$*,* S *es insatisfacible.*

4.1.4. Resolución en Lógica de Primer Orden

Dadas dos expresiones, E_1 y E_2, una sustitución σ es un unificador de (E_1, E_2) si y solo si $E_1\sigma = E_2\sigma$. En tal caso E_1 y E_2 se dicen *unificables*. Un unificador θ de (E_1, E_2) es un unificador más general (*umg*) si y solo si para cada unificador σ de (E_1, E_2) existe una sustitución π tal que $\sigma = \theta \circ \pi$. Robinson desarrolló un algoritmo que permite calcular el *umg* siempre que exista, y en caso contrario decidir que no es posible.

Regla de resolución binaria

Dadas dos cláusulas, C_1 y C_2 que no tienen variables en común, si existen dos literales, $P(t_1, ..., t_n)$ y $P(s_1, ..., s_n)$ de C_1 y C_2 respectivamente tales que $P(t_1, ..., t_n)$ y $P(s_1, ..., s_n)$ son unificables, y a través de la regla de resolución binaria se puede inferir la cláusula C_3, que es llamada resolvente binaria de C_1 y C_2. Los literales $P(t_1, ..., t_n)$ y $P(s_1, ..., s_n)$ no aparecen en C_3 y se dice que son los *literales resueltos*.

$$\frac{\begin{array}{l} C_1 : \Gamma_1 \vee P(t_1, ..., t_n) \vee \Delta_1 \\ C_2 : \Gamma_2 \vee \neg P(s_1, ..., s_n) \vee \Delta_2 \end{array}}{C_3 : (\Gamma_1 \vee \Delta_1 \vee \Gamma_2 \vee \Delta_2)\theta}$$

donde θ es el *umg* de $P(t_1, ..., t_n)$ y $P(s_1, ..., s_n)$.

Regla de Factorización

Dada una una cláusula C con dos literales L_1 y L_2 unificables, a través de la regla de factorización se puede inferir C', que se denomina un factor de C.

$$\frac{C : \Gamma_1 \vee L_1 \vee \Gamma_2 \vee L_2 \vee \Gamma_3}{C'' : (\Gamma_1 \vee L_1 \vee \Gamma_2 \vee \Gamma_3)\theta}$$

donde θ es el *umg* de L_1 y L_2.

La regla de resolución es análoga al caso proposicional, e igualmente se tiene validez, es decir, dados S un conjunto estándar y C una cláusula, si de S se deduce C usando resolución, entonces $S \models C$. Además usando árboles semánticos se prueba la completitud: un conjunto estándar S es insatisfacible si y sólo si existe una deducción de la cláusula vacía ($\square$) a partir de S.

Resolución semántica

Uno de los mayores avances en mejorar la implementación de resolución está dado por los métodos que se derivan de resolución semántica. La idea es bastante simple y viene de la observación que generalmente aplicamos resolución a un conjunto inconsistente de fórmulas S; esto significa que ninguna valuación hace verdadero al conjunto de fórmulas.

Ahora supongamos que elegimos una valuación arbitraria; esta valuación divide al conjunto de cláusulas en dos conjuntos no vacíos. Sean S_1 el conjunto que es hecho verdadero por tal valuación y S_2 el conjunto hecho falso, $S_1 \cup S_2 = S$.

Bajo este esquema, para producir una contradicción no tiene sentido hacer resolución entre cláusulas de S_1 o cláusulas de S_2. Sólo podremos producir una contradicción al hacer resolución entre cláusulas de distintos conjuntos. También es posible demostrar que esta resolución es correcta y completa.

El principio de resolución puede generar muchas cláusulas irrelevantes y redundantes, una buena idea para evitar la generación de las cláusulas innecesarias es dividir un conjunto de cláusulas en dos grupos: S_1 y S_2; añadiendo la restricción de que no se pueden resolver entre sí cláusulas del mismo grupo. En resolución semántica, se usa una interpretación para dividir las cláusulas, este hecho es el que le da el nombre de resolución semántica. Veamos algunos procedimientos particulares de resolución semántica.

Una *hiper-resolución* es un caso especial de resolución semántica, en la cual la interpretación escogida falsifica a todas las variables del conjunto de fórmulas. La ventaja de hiper-resolución es que la pertenencia de cualquier fórmula a los conjuntos S_1 o S_2 puede ser determinada sintácticamente. En efecto, las cláusulas que se hacen falsas son todas aquellas que contienen sólo literales positivos. Por otro lado, las cláusulas que contienen al menos un literal negativo se hacen verdaderas. En cada paso de la hiper-resolución, se escogen n cláusulas (llamadas satélites) que sólo contengan literales positivos y una cláusula que contenga, al menos, un literal negativo (llamada núcleo).

4.2. $\mho$-resolución

James Lu et al proponen $\mho$-resolución para ciertas lógicas[1] anotadas, con las anotaciones en retículos denominados ordinarios, y muestran que los retículos ordinarios son equivalentes a los distributivos completos [59]. La regla de $\mho$-resolución es una técnica para desarrollar un procedimiento de consulta de respuestas al estilo SLD, dentro del marco de las programación lógica para un subconjunto de lógicas anotadas. Éstas incluyen las lógicas signadas con las cuales el grupo de Lu ha trabajado durante los años 90's en el marco de representación de conocimiento y programación lógica. Ellos muestran que $\mho$-resolución es completa, y que el $\mho$-operador permite incorporar ciertas técnicas de borrado naturalmente, lo que permite incorporar con efectividad las estrategias de restricción clásicas respectivas.

El fundamento de las técnicas de resolución en el caso clásico está en la noción de par complementario, sin embargo al pasar a trabajar con lógicas no clásicas, aparecen dificultades como que en general tal noción no se tiene en forma inmediata. Pues puede suceder que existan elementos del conjunto anotador que no sean comparables.

En las lógicas de Lu se consideran las anotaciones como ideales. Un ideal es un conjunto de la forma $\{x : x \leq \mu$, para algún $\mu\}$ y así la negación de un átomo anotado es considerada como la fórmula anotada con el complemento de un ideal entonces la $\mho$-resolvente de un átomo anotado con un átomo negado será un átomo negado para poder aprovechar la propiedad de los ideales.

Si A es un subconjunto de un retículo distributivo y completo $\mathcal{L}$, representamos por $\uparrow A$ al conjunto $\{\rho \in \mathcal{B} \mid (\exists \mu \in A) \quad \mu \preceq \rho\}$. Así si $\zeta = Sup\{\mu, \rho\}$ entonces $(\uparrow \mu) \cap (\uparrow \rho) = \uparrow \zeta$.

Diremos que un subconjunto A es *regular* si para algún $\delta \in \mathcal{L}$ se tiene que $A = \uparrow \delta$ ó $A = (\uparrow \delta)'$. En nuestro contexto sus anotaciones sólo afectan los átomos, y sólo permite anotaciones regulares. Es importante observar que dada interpretación I sobre Σ que mapea literales en $\mathcal{L}$, siempre es posible definir una interpretación Σ-consistente I_c por $I_c(A_\lambda) = 1$ si $I(A) \in \uparrow \lambda$ y $I_c(A_\lambda) = 0$ si $I(A) \in (\uparrow \lambda)'$. Este hecho garantiza la cercanía de éstas lógicas al caso clásico.

En cuanto a la negación, ésta está dada por una *involución,* es decir, una función inyectiva $\sim : \mathcal{L} \to \mathcal{L}$ que satisface:

- Si $x \preceq y$ entonces $\sim y \preceq \sim x$, donde $\preceq$ es el orden del retículo.
- $\sim\sim x = x$

[1] La verdad ellos no desarrollan un sistema lógico formal, aunque bien podría desprenderse de sus presentaciones un tal sistema.

En esta presentación se obtiene que una cláusula con negación transfiere la negación a una operación de negación sobre un retículo. Así que dado un átomo A_λ tenemos que $\neg A_\lambda \leftrightarrow A_{\sim\lambda}$, preservándose la idea original de las lógicas P_τ.

El $\mho$-operador es definido por $\mho(\mu, \rho) = Inf\ \mathcal{M}(\mu, \rho)$, donde $\mathcal{M}(\mu, \rho) = \{\gamma \in \mathcal{B} \mid Sup\{\gamma, \rho\} \succeq \mu\}$. Un retículo $\mathcal{L}$ es ordinario si $\mu, \rho \in \mathcal{L}$ implica que $\mho(\mu, \rho) \in \mathcal{M}(\mu, \rho)$.

La resolvente entre las cláusulas $\neg A_\mu \vee E_1$ y $A_\rho \vee E_2$ en el caso de que $\mu \leq \rho$ (se dice que $\neg A_\mu$ y A_ρ son complementarios) está dada por $E_1 \vee E_2$. Dos cláusulas son resolubles cuando la anotación del átomo negado es menor o igual que la anotación del átomo no negado. Para el caso en que se tienen las cláusulas $A_{\mu_1} \vee D_1$ y $A_{\mu_2} \vee D_2$ siendo μ_1 y μ_2 incomparables en el retículo, la resolvente está definida por $A_{\sup\{\mu_1,\mu_2\}} \vee D_1 \vee D_2$. Si tenemos que $\neg A_\mu$ y A_ρ tales que $\mu \not\leq \rho$, entonces la regla de resolución no es aplicable. Para obtener la reducción regular es necesario encontrar una anotación minimal $\mathcal{S}_0$ que contenga a $(\uparrow \mu)' \cap \uparrow \rho$ y así la resolvente de las dos cláusulas tenga a $\mathcal{S}_0$ como su anotación. Lu et al demuestran que tal conjunto minimal sobre las anotaciones esta dado por $(\uparrow \mho(\mu, \rho))'$ por lo que es indispensable que el retículo sea ordinario.

La regla de $\mho$-resolución

Dadas las cláusulas $(\neg A_\mu \vee D_1)$ y $(A_\rho \vee D_2)$ entonces la *$\mho$-resolvente* de las dos cláusulas sobre los literales anotados $\neg A_\mu$ y A_ρ es $\neg A_{\mho(\mu,\rho)} \vee D_1 \vee D_2$. Es decir, la $\mho$-resolvente se obtiene para el par complementario de cláusulas anotadas al calcular $(\uparrow \mho(\mu, \rho))'$. Cuando τ es un retículo ordinario entonces $(\uparrow \mho(\mu, \rho))'$ es el signo regular negativo que contiene $(\uparrow \mu)' \cap \uparrow \rho$.

Además ellos construyen un procedimiento de macro inferencia de hiper-resolución basado en el operador de $\mho$-resolución el cual puede verse como una regla única que ejecuta de forma simultanea varios pasos de resolución como sucede con la regla de hiper-resolución en lógica clásica.

4.3. Programación con restricciones

Uno de los nuevas tendencias paradigmáticas en computación es la *programación con restricciones*, una tecnología de software que permite expresar de forma fácil, declarativa y bastante intuitiva la especificación y la resolución de problemas complejos y reales tales como planificación, optimización, asignación de recursos, cumplimiento de contratos Web y de la industria, entre otros.

Un sistema de restricciones es una estructura $\langle D, \vdash \rangle$, con D un conjunto no vacío de símbolos de restricciones y $\vdash \subset D_o \times D$, donde D_o representa la colección de subconjuntos finitos (más aún en esta sección el símbolo $_o$ indicará finitud), es una relación de consecuencia tal que:

1. Si $P \in u$ entonces $u \vdash P$.
2. Si $u \vdash P$, para todo $p \in v$, y $v \vdash Q$ entonces $u \vdash Q$.

Además $\vdash$ se extiende a $D_o \times D_o$ de la forma usual; $u \vdash v$ si y sólo si $u \vdash P$ para todo $P \in v$. Se denota $u \approx v$ si $u \vdash v$ y $v \vdash u$.

Los hechos (representables en el sistema) se identifican con el conjunto de símbolos de restricción.

Dado un sistema de restricciones $\langle D, \vdash \rangle$. Si $c \subset D$ es tal que si $u \subset_o c$ y $u \vdash P$ entonces $P \in c$, diremos que c es *un elemento*. El conjunto de todos los elementos del sistema se denota por $|D|$. Si $u \subset_o D$, el conjunto $\{P \in D : u \vdash P\}$ se denota por $\overline{u}$. Obviamente $\overline{u} \in |D|$.

El retículo (de las teorías): $(|D|, \subset)$ es un retículo algebraico completo, no es necesariamente finito. Los elementos finitos de $|D|$ son los elementos generados por subconjuntos finitos de D. El conjunto de todos los elementos finitos se denota por $|D|_o$. Se denota: $\uparrow c = \{d \in |D| : c \leq d\}$ y $\downarrow c = \{d \in |D| : d \leq c\}$. *Una restricción finita* es un conjunto finito de símbolos. Si u es una restricción finita se identifica u con $\overline{u} \in |D|_o$.

Si $\mathcal{L}$ es un lenguaje de primer orden con un conjunto enumerable de variables, sean D un conjunto de fórmulas abiertas en dicho lenguaje, y Δ una clase de estructuras. Se define la relación de consecuencia por $\{P_1, P_2, ..., P_n\} \vdash Q$ si y sólo si $M \models \{P_1, P_2, ..., P_n\}$, para toda $M \in \Delta$ implica que $M \models Q$. Este es un sistema de restricciones.

Las lógicas anotadas con restricciones (LAC) presentadas por Frühwirth en [45] extienden los lenguajes de primer orden con cierta clase de predicados distinguidos, *las restricciones,* y una clase de términos (distinguidos) que forman un retículo, *las anotaciones,* las cuales se usan a nivel de fórmula. Semánticamente LAC provee reglas de inferencia para razonar sobre fórmulas anotadas y una teoría de restricciones que define las operaciones de retículo, es decir, las restricciones sobre las anotaciones, y muestran que existen recorridos sistemáticos para obtener un fragmento clausal que sea ejecutable como un programa lógico con restricciones. Presentan un interpretador y un compilador que puede ser implementado con los lenguajes de programación lógica con restricciones.

La idea subyacente al capturar una lógica temporal que sea ejecutable convenientemente es separar los aspectos temporales de los no temporales. En este caso las anotaciones gozan de un estatus especial para designar los aspectos temporales, y si eliminamos las anotaciones estaremos tratando con un una lógica de primer orden estándar. Con el advenimiento de la programación lógica con restricciones, la implementación de lógicas temporales por aplicación dentro de los lenguajes con restricciones viene a ser posible. La idea de tal aplicación es introducir parámetros temporales a través de relaciones y funciones especiales que describan la estructura de tiempo, que bien pueden ser miradas como restricciones y las condiciones asociadas como una teoría de restricciones. Lo que garantiza la separación de los aspectos temporales de los de la lógica misma, así el manejo de la teoría de restricciones puede usar un algoritmo especial, mientras que para los aspectos de la lógica (l.p.o) es suficiente SLD-resolución.

El poder de expresibilidad de éstos nuevos sistemas lógicos es indudable, sin embargo en las presentaciones iniciales aparece limitado por la restricción de anotar solamente los átomos. Por ende es fundamental extender las anotaciones a fórmulas complejas, necesidad que es planteada explícitamente por Frühwirth. En su intento por solucionar estas dificultades ellos proponen tres tipos de anotaciones involucrando conjuntos temporales, así como instantes. Este hecho no permite una caracterización homogénea de las anotaciones, pero bien podría adecuarse su propuesta para lograrlo. Más aún introduce operadores de anotación innecesarios, pues están dados por negaciones estándar.

Para una fórmula de primer orden A y t un instante de tiempo la correspondiente fórmula anotada[2] (f_tA) se entiende como que la fórmula A es verdadera en el instante t. En este caso establece condiciones de homomorfismo para el operador f_t con respecto a la negación y la conjunción. Posteriormente define lo que sería anotar en un conjunto de tiempos, diciendo que es lo mismo que anotar en cada uno de los instantes de tal conjunto, con lo que se pretende capturar variables temporales cuantificadas. Para un conjunto de tiempos I, nos permitimos representar por f_IA como el equivalente a f_tA para todo t en I.

Dentro de los operadores innecesarios está presentar como un nuevo operador la negación de una fórmula negada que ha sido anotada en un conjunto de instantes. Estos hechos generan dificultades innecesarias en su propuesta, puesto que se entra a mostrar propiedades que se podrían recibir directamente de las interpretaciones estándar de la lógica de primer orden.

Las ideas temporales manejadas anteriormente se extienden con la colaboración de Raffaeta [66] a lógicas espacio temporales, agregando a las anotaciones componentes espaciales.

[2] El autor no introduce esta notación, sin embargo creemos que dentro de nuestro trabajo el uso de la notación f_tA nos facilitará establecer algunas observaciones.

4.4. Elementos básicos de lógica borrosa

Para poder entrar a comentar algunos de los nuevos desarrollos en programación lógica fuzzy, necesitamos precisar algunos elementos básicos de lógica fuzzy (ver [63]). Es así que daremos la presentación general de las t-normas y t-conormas, incluyendo algunos importantes resultados con generadores de t-normas. De igual forma haremos presentaciones generales de las negaciones y las implicaciones. Con estos elementos tendremos por reto comentar acerca de ciertas tendencias en al programación lógica difusa. Recordemos que un *conjunto borroso* sobre un conjunto usual U es una función $U \longrightarrow [0,1]$.

4.4.1. Algunas operaciones sobre conjuntos borrosos

- $\chi_{A\cup B}(x) = max\{A(x), B(x)\} = A(x) \vee B(x)$
- $\chi_{A\cap B}(x) = min\{A(x), B(x)\} = A(x) \wedge B(x)$
- $\chi_{A'}(x) = 1 - \chi_A(x)$.

4.4.2. El retículo $[0,1]$

- $x \vee y = max\{x, y\}$
- $x \wedge y = min\{x, y\}$
- $x' = 1 - x$

  El complemento $'$ es una operación tal que $x'' = x$, $\;x \leqslant y \Rightarrow y' \leqslant x' \;\; 0' = 1$, y $\;1' = 0$. (Una *involución* que satisface las leyes de DeMorgan). Es decir, el retículo $[0,1]$ es un *álgebra de DeMorgan.*
- El conjunto de todos los conjuntos borrosos definidos sobre un conjunto U tiene naturalmente estructura de álgebra de DeMorgan, $\langle \mathcal{F}(U), \vee, \wedge, ', 0, 1\rangle$ es un álgebra de DeMorgan, donde $\mathcal{F}(U) = [0,1]^U$.

4.4.3. t-normas

Una t-norma es una función $\triangle : [0,1] \times [0,1] \longrightarrow [0,1]$ tal que:

1. $1\triangle x = x$
2. $x\triangle y = y\triangle x$
3. $x\triangle(y\triangle z) = (x\triangle y)\triangle z$
4. Si $w \leqslant x$ e $y \leqslant z$ entonces $w\triangle y \leqslant x\triangle z$

Ejemplos

- $x\triangle_0 y = \begin{cases} x \wedge y & si \quad x \vee y = 1 \\ 0 & en\ otro\ caso \end{cases}$
- $x\triangle_1 y = 0 \vee (x + y - 1)$
- $x\triangle_2 y = \frac{xy}{2-(x+y-xy)}$
- $x\triangle_3 y = xy$
- $x\triangle_4 y = \frac{xy}{x+y-xy}$
- $x\triangle_5 y = x \wedge y$

Observación:
Si $\triangle$ es una t-norma entonces para todos $x, y \in [0,1]$, se cumple que

$$x\triangle_0 y \leqslant x\triangle y \leqslant x\triangle_5 y.$$

Diremos que una t-norma es *arquimediana* si es continua y para todo $x \in (0,1)$ se cumple que $x\triangle x < x$, y cuando $x\triangle x = x$, diremos que es *idempotente*, la única tal norma es $\triangle_5$.

Generadores

Sea $a \in [0,1)$ y f un isomorfismo de orden de $[0,1]$ a $[a,1]$. Definiendo $\triangle_f$ por

$$x\triangle_f y = f^{-1}(f(x)f(y) \vee a)$$

tenemos que es una t-norma arquimediana.

Cuando $\triangle = \triangle_f$, diremos que f es un *generador* de $\triangle$.

Caracterización de t-normas arquimedianas

Si $\triangle$ es una t-norma arquimediana entonces existen un $a \in [0,1)$ y un isomorfismo de orden $f : [0,1] \longrightarrow [a,1]$ tal que $x\triangle y = f^{-1}(f(x)f(y) \vee a)$ para todos $x, y \in [0,1]$. Más aún, si tenemos que g es un isomorfismo de orden $g : [0,1] \longrightarrow [a,1]$ entonces $x\triangle y = g^{-1}(g(x)g(y) \vee a)$, si y sólo si $f(x) = g(x)^r$ para algún $r > 0$.

Una t-norma se dice *nilpotente* si para todo $a \neq 1$ existe un número natural n tal que $a^n = 0$, donde $a^n = a\triangle a\triangle \ldots \triangle a$ (n veces a). Por ejemplo $\triangle$ definida por $x\triangle y = 0 \vee (x + y - 1)$.

Si f es un generador de una t-norma arquimediana $\triangle$, entonces $\triangle$ es nilpotente si y sólo si $f(0) > 0$.

Las t-normas arquimedianas se clasifican en las nilpotentes y las que no lo son, a las que denominaremos *estrictas*. Por ejemplo $\triangle$ definida por $x\triangle y = xy$ es estricta.

Caracterización de t-normas estrictas

Una t-norma arquimediana $\triangle$ es estricta si y sólo si existe un isomorfismo de orden $f : [0,1] \longrightarrow [0,1]$ tal que $x\triangle y = f^{-1}(f(x)f(y))$ para todos $x, y \in [0,1]$; ó equivalentemente $f(x\triangle y) = f(x)f(y)$ para todos $x, y \in [0,1]$.

En consecuencia: una t-norma arquimediana $\triangle$ es estricta si y sólo si $x\triangle y$ es estrictamente creciente para todo $x, y \in (0,1)$.

Ejemplos

1. La t-norma arquimediana estricta $\triangle$ definida por $x\triangle y = xy$ tiene por generador $f(x) = x$.
2. La t-norma arquimediana estricta $\triangle$ definida por $x\triangle y = \frac{xy}{x+y-xy}$ tiene por generador $f(x) = e^{-\frac{1-x}{x}}$.
3. La t-norma arquimediana nilpotente $\triangle$ que está definida por la relación $x\triangle y = 0 \vee (x+y-1)$ tiene por generador $f(x) = e^{x-1}$.

4.4.4. Negaciones fuertes

Una negación fuerte es una función $\eta : [0,1] \longrightarrow [0,1]$ tal que:

1. $\eta(0) = 1, \quad \eta(1) = 0$.
2. η es decreciente.
3. $\eta(\eta(x)) = x$.

Ejemplos

$\alpha : [0,1] \longrightarrow [0,1]$, definida por $\alpha(x) = 1 - x$.

Si η es una negación fuerte, entonces existe un automorfismo f de $\langle [0,1], \leqslant \rangle$ tal que $\eta = f^{-1}\alpha f$.

Dada $\triangle$ una t-norma arquimediana nilpotente y f un generador de $\triangle$, considerando $\eta_\triangle(x) = \bigvee\{y : y\triangle x = 0\}$ y $\eta_f(x) = f^{-1}(\frac{f(0)}{f(x)})$, entonces $\eta_\triangle = \eta_f$ y es una negación fuerte.

4.4.5. t-conormas

Una t-conorma es el operador dual de alguna t-norma con respecto a una negación fuerte. es decir,

$$x \nabla y = \eta(\eta(y) \Delta \eta(x))$$

Caracterización de t-conormas

Una operación binaria ∇ sobre $[0, 1]$ es una t-conorma si y sólo si

1. $0 \nabla x = x$
2. $x \nabla y = y \nabla x$
3. $x \nabla (y \nabla z) = (x \nabla y) \nabla z$
4. Si $w \leqslant x$ e $y \leqslant z$ entonces $w \nabla y \leqslant x \nabla z$

Ejemplos

- $x \nabla_0 y = \begin{cases} x \vee y & si \quad x \wedge y = 0 \\ 1 & en\ otro\ caso \end{cases}$
- $x \nabla_1 y = 1 \wedge (x + y)$
- $x \nabla_2 y = \frac{xy}{1+xy}$
- $x \nabla_3 y = x + y - xy$
- $x \nabla_4 y = \frac{x+y-2xy}{1-xy}$
- $x \nabla_5 y = x \vee y$

4.4.6. Implicaciones difusas

Hay por lo menos tres formas estándar de obtener implicaciones, que describiremos a continuación.

R-implicaciones asociadas con $\triangle$

Una R-implicación es una función

$$\Rightarrow: [0,1] \times [0,1] \longrightarrow [0,1]$$

de la forma

$$x \Rightarrow y = \bigvee\{z \in [0,1] : x \triangle z \leqslant y\}.$$

Con lo cual tenemos que:

- Para $x \triangle y = x \wedge y$

  $$x \Rightarrow y = \bigvee\{z \in [0,1] : x \wedge z \leqslant y\}$$

  Luego

  $$x \Rightarrow y = \begin{cases} 1 & si \quad x \leqslant y \\ y & si \quad x > y \end{cases}$$

- Para $x \triangle y = xy$

  $$x \Rightarrow y = \bigvee\{z \in [0,1] : xz \leqslant y\}$$

  Luego

  $$x \Rightarrow y = \begin{cases} 1 & si \quad x \leqslant y \\ \frac{y}{x} & si \quad 0 \leqslant y < x \end{cases}$$

- Para $x \triangle y = 0 \vee (x + y - 1)$

  $$x \Rightarrow y = \bigvee\{z \in [0,1] : 0 \vee (x + z - 1) \leqslant y\}$$

  Luego

  $$x \Rightarrow y = 1 \wedge (1 - x + y)$$

∇-implicaciones

Una ∇-implicación es una función

$$\Rightarrow : [0,1] \times [0,1] \longrightarrow [0,1]$$

de la forma

$$x \Rightarrow y = \eta(x) \nabla y$$

Con lo cual se tiene que:

- Para $x \nabla y = x \vee y$ y $\eta(x) = 1 - x$ tenemos que la ∇-implicación está dada por

$$x \Rightarrow y = (1 - x) \vee y$$

- Para $x \nabla y = x + y - xy$ y $\eta(x) = 1 - x$ tenemos que la ∇-implicación está dada por

$$x \Rightarrow y = (1 - x) \nabla y$$

  y por tanto

$$x \Rightarrow y = 1 - x + xy$$

- Para $x \nabla y = 1 \wedge (x + y)$ y $\eta(x) = 1 - x$ tenemos que la ∇-implicación está dada por

$$x \Rightarrow y = 1 \wedge (1 - x + y)$$

$\mathcal{Q}$-implicaciones

Dado un *sistema de DeMorgan:* $\langle \triangle, \nabla, \eta \rangle$ una $\mathcal{Q}$-implicación es una operación binaria

$$\Rightarrow : [0,1] \times [0,1] \longrightarrow [0,1]$$

de la forma

$$x \Rightarrow y = \eta(x) \nabla (x \triangle y)$$

Ejemplos

- Para el sistema de DeMorgan

$$x \triangle y = x \wedge y$$
$$x \nabla y = x \vee y$$
$$\eta(x) = 1 - x$$

la $\mathcal{Q}$-implicación es

$$x \Rightarrow y = (x \wedge y) \vee (1 - x)$$

- Para el sistema de DeMorgan

$$x \triangle y = (x + y - 1) \vee 0$$
$$x \nabla y = (x + y) \wedge 1$$
$$\eta(x) = 1 - x$$

la $\mathcal{Q}$-implicación es

$$x \Rightarrow y = (1 - x) \vee y$$

4.5. Programación lógica fuzzy

El objetivo general de establecer la programación lógica fuzzy es proveer unas bases teóricas sólidas para los sistemas expertos fuzzy. Se espera que los sistemas de programación lógica gocen de propiedades teóricas tales como que el sistema produce solamente respuestas, realmente produce todas las posibles, correctas. Normalmente aparecen nuevas nociones de consecuencia lógica fuzzy, y de modelo mínimo fuzzy. Es habitual mostrar la equivalencia entre el modelo declarativo, mínimo, el de punto fijo y la semántica procedimental.

La idea siempre es obtener teorías que presenten procedimientos mediante los cuales se pruebe o refute un hecho dado. Es decir, que haya equivalencia entre la teoría de modelos y la teoría de la prueba, o lo que es lo mismo se tenga validez y completetitud.

La diferencia semántica fundamental entre el caso clásico y el fuzzy está dada en la interpretación de los símbolos del predicado. En el caso fuzzy los símbolos de predicado se interpretan como relaciones fuzzy de el dominio sobre el intervalo $[0, 1]$. El método de definición de semánticas es conocido como *semántica modelo.* La definición de la consecuencia lógica da otra definición de la semántica que se llama *semántica declarativa.* A través de la introducción de un operador del inferencia. surge una nueva semántica, la denominada de *punto fijo,* la cual es necesaria para conectar la semántica declarativa con la semántica modelo para obtener el procedimiento mecánico de prueba que es la metodología para obtener los teoremas: la *semántica procedimental.* La definición de la semántica de punto fijo, pasa por establecer un teorema de punto fijo para el operador de inferencia sobre un retículo completo. La equivalencia entras éstas semánticas se ha probado en la programación lógica clásica, y se demuestra en lógica fuzzy en [41]. Este hecho conduce a los resultados de la validez y de completez.

4.5.1. Un nuevo principio de resolución

En [50] se introduce un principio de hiper-resolución fuzzy basado en antinomias (AFHR), y se prueba su completez. La regla de inferencia propuesta es basada en la idea de reductio ad absurdum por antinomias, no por la negación de la lógica binaria clásica. La resolución propuesta tiene un rango de perdida de significado, un conjunto especial, donde se incorpora lo que no es verdad y también que no es falso; rango que bien podría no considerarse en el razonamiento. La ventaja de este nuevo principio es que ejecuta la resolución eficazmente, y a la vez muestra la flexibilidad para ciertos razonamiento automatizados.

En contraste con la resolución de lógica clásica, la lógica fuzzy tiene varios estados tales como perfecto (1), muy alto (0.9), alto (0.7), medio (0.5), bajo (0.3), muy bajo (0.1), y ninguno (0) (entendemos los valores en un grado de pertenencia). Un orden natural ($Verdadero$) $perfecto > muy\,alto > alto > medio > bajo > muy\,bajo > ninguno\,(falso)$. Así, el predicado muy alto puede ser "negado" con predicados como bajo ó muy bajo ó ninguno. Para hacer la resolución fuzzy se tienen varios candidatos para entrar a resolver, a menos que decidamos un rango para la "negación," por ejemplo todos los predicados posibles (corto, muy corto o ninguno).

Dado un umbral de credibilidad, digamos $\beta \in [0,1]$ los valores de verdad aceptados como verdaderos están en $[\beta, 1]$, mientras que los falsos en $[0, \beta]$. Los valores en el intervalo $[1-\beta, \beta]$ son el rango de perdida de significado, pueden ser considerados como verdaderos y falsos a la vez. Usaremos el símbolo $P^a(x)$ para designar la antinomia de $P(x)$. La definición esta dada por $P^a(x) = \neg P(x)_\beta$, lease la antinomia de $P(x)$, la cual tiene un valor de verdad bajo $1-\beta$. Es decir, $\neg P(x) \supseteq P^a(x)$.

Dadas una interpretación I, y un par de las variables complementarias P_i y de $\neg P_i$, diremos que $P_i \wedge P_i^a$ es una contradicción bajo interpretación dada I. Según sea el caso, diremos:

1. Si $T_I(P_i \wedge P_i^a) = 0$ diremos que es una *antinomia completa.*

2. Si $T_I(P_i \wedge P_i^a) = [1-\beta, \beta]$ que no es antinomia.

3. Si $T_I(P_i \wedge P_i^a) = (0, \beta)$ diremos que es una *antinomia incompleta.*

Notaremos por $c = |T_I(P_i) - T_I(P_i^a)|$, *el grado de confianza* de la resolvente fuzzy $R(C_1, C_2) = L_1 \vee L_2$ de las cláusulas $C_1 = P_i \vee L_1$ y $C_2 = \neg P_i \vee L_2$.

Una interpretación I se dice que satisface una fórmula S si de $T_I(S) \geqslant \beta$, y que falsifica a S si $T_I(S) \leqslant 1 - \beta$.

Obsérvese que puede suceder que una interpretación los satisface y falsifica simultáneamente una fórmula S, baste con que $T_I(S) \in [1-\beta, \beta]$. Es decir, todos los valores de verdad en valor de verdad en $[1-\beta, \beta]$ no tienen sentido en esta lógica fuzzy.

La inferencia la regla AFHR considera simultáneamente una cláusula N que contenga por lo menos un literal negativo y las cláusulas A_i, que contienen solamente literales positivos, y entrega una cláusula B (hiper-resolvente) que contiene solamente literales positivos con el grado de confianza c.

Ejemplo

Este es un procedimiento de la resolución de AFHR:

$\neg P \vee \neg Q \vee R \vee S$	$: N$	"Si P y Q entonces R o S"
P (con valor de verdad 0.9) $\vee T$	$: A_1$	
Q (con valor de verdad 0.8) $\vee W$	$: A_2$	
$(T \vee W \vee R \vee S)_{c=0,6}$	$: B$	

donde el grado de confianza c =0.6 se obtiene calculando primero el grado de certeza para P, que está dado por $c_p = |T_I(P) - T_I(P^a)| = |$ 0.9 - 0.1 $| = 0.8$. Luego se hace lo mismo para Q, y obtenemos que $c_q = 0.6$. Así, por simplicidad, usando la $t-$norma *min* para interpretar a $\wedge$ tenemos que $min\{c_p, c_q\}$ es igual 0.6.

A través de la necesidad de incluir el grado de confianza sobre la resolvente es natural pensar que se necesite usar anotaciones de fórmulas complejas, aunque (una vez más) los autores no hacen este tipo de observaciones.

4.5.2. Fuzzy Prolog

En [47] se propone una lenguaje fuzzy con modelos $\mathcal{B}([0,1])$ (los borelianos finitos), que permite incluir diversas propuestas anteriores que utilizan representación de los valores de verdad basadas en unión de sub-intervalos de $[0, 1]$. El sistema prolog se ejecuta interpretando el razonamiento fuzzy como sistema de restricciones que se propagan con las reglas por medio de operadores de la agregación. Presentan la semántica declarativa y procedimental, y se prueba su equivalencia. La puesta en práctica está dada por programación lógica con restricciones sobre números reales.

Esta propuesta se enmarca a través de los siguientes aspectos generales:

1. Un valor de verdad será la unión finita de sub-intervalos de $[0, 1]$. Se incluye el caso particular de intervalos con solamente un elemento.

2. Un valor de verdad será propagado a través de las reglas por medio de los operadores de agregación. Definición que incluye los operadores conjuntivos (como las normas), a los operadores disyuntivos (como las co-normas), a los operadores de promedios (como la media aritmética, promedio casi lineal, etc) y a los operadores híbridos (combinaciones de los operadores antes mencionados).

3. Se combinan el razonamiento clásico, junto al fuzzy en un compilador tipo prolog.

4. Se presentan la semántica declarativa y procedimental para los programas de la lógica fuzzy, y se prueba su equivalencia.

5. Se da una implementación para el lenguaje propuesto, agregando los contenidos fuzzy a un compilador tipo prolog usando programación lógica con restricciones sobre números reales. Los intervalos representan restricciones sobre número reales y los operadores de agregación operan sobre dichas restricciones.

La idea de fondo es proporcionar una poderosa herramienta que incorpore el razonamiento fuzzy para solucionar problemas fuzzy complejos y problemas con incertidumbres, mostrando que su implementación es natural en programación lógica con restricciones sobre los números reales.

En [63] los conjuntos fuzzy sobre $[0, 1]$ se extienden (naturalmente) sobre los intervalos de $[0, 1]$ por funciones $A : X \rightarrow \mathcal{E}([0, 1])$, donde $\mathcal{E}([0, 1])$ denota la familia de todos los sub-intervalos cerrados de $[0, 1]$. Ahora se generaliza al álgebra de los borelianos sobre el intervalo $[0, 1]$. Se tomarán las funciones $A : X \rightarrow \mathcal{B}([0, 1])$, donde un elemento en $\mathcal{B}([0, 1])$ es una unión finita de sub-intervals de $[0, 1]$.

4.5.3. Operadores de agregación

Los operadores de agregación se utilizan para propagar los valores de verdad por medio de reglas fuzzy. Éstos son operadores de la forma $f : [0, 1]^n \rightarrow [0, 1]$ tales que $f(0, ..., 0) = 0$ y $f(1, ..., 1) = 1$, son monótonos y continuos.

Dado $f : [0,1]^n \longrightarrow [0,1]$ un operador de agregación se extiende (ver [63]) a intervalos por la función $F : (\mathcal{E}([0,1]))^n \longrightarrow \mathcal{E}([0,1])$ definida por $F([x_1, y_1], ..., [x_n, y_n]) = [f(x_1, ..., x_n), f(y_1, ..., y_n)]$.

Ahora en [47], dado un operador de agregación sobre intervalos se extienden a los borelianos por la función $\mathcal{F} : (\mathcal{B}([0,1]))^n \longrightarrow \mathcal{B}([0,1])$ definida por $\mathcal{F}(B_1, ..., B_n) = \bigcup\{F(E_1, ..., E_n) : E_j \in B_j\}$.

Esta novedosa propuesta[3], en nuestros términos, intenta desarrollar una lógica con anotaciones únicamente sobre los átomos, y dichas anotaciones tomadas en los borelianos (finitos) del intervalo unidad.

Existen otras propuestas de lógicas temporales con restricciones, donde se exhiben indicios de la necesidad de lógicas anotadas, como por ejemplo la de [76] donde se intenta generalizar las lógicas probabilísticas. En este caso se podría pensar las anotaciones sobre funciones de distribución de probabilidades, lo que hace dispendioso un comentario más preciso. Sin embargo una ligera inspección de la propuesta permite chequear la necesidad de hacer anotaciones sobre formulas complejas.

4.6. Generalidades en Lógicas Anotadas

A fin de fijar ideas en un sentido general, que nos permita ampliar las anteriores presentaciones de las lógicas anotadas, señalaremos algunas observaciones que a la luz de nuestra investigación consideramos relevantes.

4.6.1. Álgebra de interpretación de verdad

Los sistemas lógicos generalmente los presentamos sintácticamente, y dentro de lo posible a cada lógica S se le asocia por lo menos una semántica, que da cuenta de las interpretaciones. Esto se hace siguiendo las ideas del cálculo clásico de primer orden, se fija un dominio D donde se interpretan las letras proposicionales y las constantes (símbolo de conectivo de aridad cero), y cada símbolo de conectivo $n-$ario ω en el lenguaje, se interpreta por una función $n-$aria $\omega^{\mathfrak{D}}$ definida de D^n en D.

Tomando todos los símbolos ω en el lenguaje podemos formar $\mathfrak{D} = \langle D, \omega^{\mathfrak{D}} \rangle$, un álgebra. Nos referiremos a ella como un *álgebra de interpretación de verdad* para S. Generalmente la pensaremos fija. Para nuestros desarrollos posteriores tendremos como requisito que el lenguaje contenga por lo menos un símbolo que nos represente la implicación, habitualmente emplearemos el símbolo $\rightarrow$ como es lo usual.

[3]Los autores no relacionan su presentación con lógicas anotadas, ni presentan un sistema lógico formal.

El álgebra de anotaciones será un álgebra de interpretación de verdad enriquecida por operadores unarios g_σ para cada σ en el dominio de anotaciones. Para efectos de las intuiciones, en general sugerimos pensar esta álgebra como un retículo, aunque propondremos ejemplos donde no lo sea.

En este orden de ideas, se trata de pasar de una lógica, incluyendo una versión interpretativa, a una lógica anotada. Por ejemplo si tenemos una lógica fija, podremos pensar que se hacen observaciones sobre dicha lógica a través de la actuación de las anotaciones. En particular tendremos en esta dirección las anotaciones en un semigrupo, y las observaciones sobre la lógica están dadas por los operadores de anotación que se registran formalmente por una acción del semigrupo sobre el dominio de interpretación.

Digamos que tenemos una presentación de un sistema S fijo. A partir de éste formemos un sistema anotado. Al lenguaje del sistema S le agregamos un símbolo de operación unaria g_σ para cada constante de anotación en el dominio de anotación. Las fórmulas del nuevo sistema se definen en la manera estándar, agregando a las reglas de formación de fórmulas que *si A es una fórmula y σ es una anotación, entonces $g_\sigma A$ es una fórmula.* Las demás definiciones básicas son estándar. Nos referiremos a $g_\sigma A$ como *fórmula anotada*, y si $A = p$, diremos que $g_\sigma p$ es un átomo anotado.

4.6.2. Condiciones generales para la algebrización

Pensemos que al interpretar las fórmulas de la lógica en consideración tenemos un álgebra de *interpretación* $\mathcal{A}$ fija, sobre la cual además están definidas las funciones h y k.

En las siguiente observaciones, unas veces nos conviene tener equivalencias, pero otras será suficiente con implicaciones en una de las direcciones consideradas. Para efectos de simplificar las notaciones nos quedaremos con los símbolos de equivalencia.

Para la negación de fórmulas anotadas, por lo general basta con considerar dos casos de acuerdo al comportamiento de la fórmula a negar. Para cuando dicho comportamiento se considere clásico supondremos los operadores de anotación como homomorfismos. Mientras que el caso no clásico, tendremos consideraciones especiales.

1. A con comportamiento clásica:

$$\neg g_b A \leftrightarrow g_b \neg A$$

2. A con comportamiento no clásica:

$$\neg g_b A \leftrightarrow g_{h(b,\ldots)} \neg A$$

La función h debe tomar valores en el álgebra de anotaciones. Lo estándar ha sido usar una función que represente una "negación," por esto sugerimos pensar inicialmente $h(b, ...) = \sim b$, con lo cual estaremos considerando $\sim$ una función del álgebra de anotaciones en ella misma. Dentro de la mayor parte de nuestras presentaciones consideraremos funciones $\sim$ tal que su doble aplicación sea igual a la identidad.

Como hemos resaltado, uno de nuestros aportes es hacer anotaciones de fórmulas complejas, lo que nos obliga a considerar las anotaciones de anotaciones. Esto lo manejaremos a través de:

$$g_b g_c A \leftrightarrow g_{k(b,c,...)} A$$

donde la función k toma valores en el álgebra de anotaciones. Inicialmente nos parece prudente usar la notación: $k(b, c, ...) = b \otimes c$, que sugiere que dicho proceso se salda a través de operar las anotaciones en otra instancia. Esta idea es lo suficientemente amplia, como veremos en nuestros desarrollos posteriores. En varias de nuestras presentaciones hemos escogido el caso en que $\otimes$ es realmente una operación binaria sobre el algebra de anotaciones, con algunas condiciones mínimas entre los que no puede faltar la asociatividad de dicha operación.

Siguiendo con la idea de anotar fórmulas complejas, necesitamos hacer anotaciones de disyunciones y conjunciones:

$$g_b(A \vee B) \leftrightarrow F_{\vee}(A, B, g_b A, g_b B, ...)$$

$$g_b(A \wedge B) \leftrightarrow F_{\wedge}(A, B, g_b A, g_b B, ...).$$

Este punto no es sencillo de tratar, y se ha hecho de acuerdo a las intuiciones propias de cada lógica desarrollada. Algunas veces los operadores de anotación se comportan como homomorfismos, lo cual nos permite los desarrollos con cierta facilidad. Sin embargo veremos que esto no siempre es posible, pues según el carácter de las anotaciones nos hemos encontrado también con comportamiento de anti-homomorfismos como veremos en el caso de las lógicas anotadas fuzzy.

Por último también necesitamos hacer anotaciones de las implicaciones. Si se define a través de la negación y la disyunción, como se hace en el caso clásico, se reduciría a los items anteriores, sin embargo en general no es el caso, más aún podrá suceder que tenemos varios tipos de implicaciones. Así en términos generales tendremos:

$$g_b(A \rightarrow B) \leftrightarrow F_{\rightarrow}(A, B, g_b A, g_b B, ...)$$

Veamos una serie de requerimientos o exigencias que hemos de asegurar con condiciones sobre cada una de las funciones de anotación antes introducidas.

1. En los casos más cercanos con P_τ, dado $b \in A$

$$g_b A \to g_c A$$

para todo c tal que $R(c, b)$, donde R representa una cierta relación en el álgebra de anotaciones, por ejemplo el orden en el caso de τ.

2. De nuevo en los casos más cercanos con P_τ, dados $b, c \in A$

$$g_b A \wedge g_c A \to g_{b \oplus c} A$$

donde $\oplus$ es una operación en el algebra de anotaciones, que generaliza el habitual $\vee$ de los retículos.

3. Insistamos en que $\otimes$ debe ser por lo menos asociativa, pues esto nos permitirá abordar anotaciones de anotaciones en forma anidada. Sin embargo en ciertos casos necesitaremos de restricciones mayores.

De alguna manera fijar la idea de *par complementario*, para efectos de asegurar aplicaciones a bases de datos a través del método de resolución al estilo Robinson. Una posible opción será usar la misma relación R citada en el item 1. En la literatura ([59]) se "sugiere" el uso de unos nuevos retículos denominados *ordinarios.*

4.6.3. Algebrización a lo Blok-Pigozzi

Dados los antecedentes en los procesos de algebrización de lógicas anotadas, así como los logros del presente trabajo podemos hacer ciertas recomendaciones generales en tales procesos. En este camino nos proponemos fijar que el sistema de fórmulas de equivalencia $A \Delta B$ está dada por:

1. $A \leftrightarrow B$
2. $\neg A \leftrightarrow \neg B$
3. $g_b A \leftrightarrow g_b B$

Condiciones caso (a) : $A \Delta B \vdash \neg A \Delta \neg B$

1. La condición $\neg A \leftrightarrow \neg B$ se obtiene del item 2.
2. Para lograr la condición $\neg\neg A \leftrightarrow \neg\neg B$ será necesario fijar axiomas que garanticen $\neg\neg A \leftrightarrow A$, y junto al resultado del item 1.

3. Para lograr la condición $g_b\neg A \leftrightarrow g_b\neg B$ en general es necesario dividirla en dos casos:

   Caso 1: El comportamiento de A es clásico, en este caso como por item 3 se tiene que $g_bA \leftrightarrow g_bB$ entonces es más o menos razonable esperar como conclusión (clásica) que $\neg g_bA \leftrightarrow \neg g_bB$.

   Caso 2: Si el comportamiento de A no es clásico, es una situación especial que obliga a escoger bien los axiomas de la negación que acompañan la propiedad dada por item 2.

Condiciones caso (b) : $A\Delta B \vdash g_bA \leftrightarrow g_bB$

1. La condición $g_bA \leftrightarrow g_bB$ por item 3.
2. Para lograr la condición $\neg g_bA \leftrightarrow \neg g_bB$ es necesario, nuevamente, dividir en dos casos:

   Caso 1: El comportamiento de A es clásico, por item 3 se tiene que $g_bA \leftrightarrow g_bB$ entonces es razonable tener como conclusión (clásica) que $\neg g_bA \leftrightarrow \neg g_bB$.

   Caso 2: Si el comportamiento de A no es clásico, por item 2 se tiene que $\neg A \leftrightarrow \neg B$, asumiendo que $\neg g_bA \leftrightarrow g_{\sim b}\neg A$, necesitamos como axioma (ó una condición suficiente para el mismo) que: $(A \leftrightarrow B) \rightarrow (g_bA \leftrightarrow g_bB)$.
3. La condición $g_cg_bA \leftrightarrow g_cg_bB$ se puede lograr, a rasgos generales, al establecer dentro de los axiomas que $g_cg_bA \leftrightarrow g_{c\otimes b}B$.

Condiciones caso (c) : *Si* $A\Delta B$ *y* $C\Delta D$ *entonces* $A * C\Delta B * D$, se tiene que cuidar el comportamiento clásico y un axioma del tipo $g_b(A * B) \leftrightarrow g_bA * g_bB$.

Condiciones caso (d) : $A \dashv\vdash \delta(A)\Delta\varepsilon(A)$. Como nuestras lógicas son en su parte positiva la LIP, será oportuno esperar que el caso $\delta(A) = A \wedge A$ y $\varepsilon(A) = A \rightarrow A$ sea relativamente controlable.

4.7. Las lógicas anotadas propuestas

Aunque la problemática en cuestión aparece dispersa, en una serie de artículos que nos muestran algunas de la múltiples necesidades en la representación de conocimiento, creemos firmemente que nos proporciona una amplia línea de investigación. Particularmente el hecho de liberar las anotaciones de los átomos, es decir, construir lógicas anotadas que permitan anotaciones de anotaciones es por ende un aporte en sí mismo. Nuestra propuesta goza de originalidad en este sentido, pues al momento no se han desarrollado lógicas anotadas con estas características. En este trabajo presentaremos tres sistemas lógicos, que gozan de las características antes mencionadas, que tienen buen comportamiento al menos desde el punto de vista algebraico.

4.7.1. Anotaciones en semigrupos

Proponemos una lógica anotada con las anotaciones en un semigrupo, la idea básica es pensar que las anotaciones actúan sobre el álgebra de los valores de verdad. En esta intuición podremos pensar que una vez determinados los valores de verdad en un cierto orden, alguien decide hacer un reordenamiento de los mismos valores. Este reordenamiento está dado por la acción de las anotaciones, que en este ejemplo están en un álgebra particular, un semigrupo, sobre el conjunto de interpretaciones. Por demás podría pensarse que se tiene establecida una cierta lógica, con un buen comportamiento, para representar ciertas situaciones de conocimiento, y se hacen anotaciones sobre tal lógica, entonces es deseable que la nueva lógica, la lógica con las anotaciones, goce del mismo buen comportamiento. Este hecho lo hemos conseguido desde el punto de vista algebraico, lo que nos deja a la vista de un método para generar multitud de lógicas anotadas de este tipo. Esto nos da una ilustración de la idea de hacer anotaciones que respeten el buen comportamiento.

4.7.2. Anotaciones fuzzy

Se proponen lógicas anotadas cercanas a $SP\mathcal{T}$ y al sistema C_n de da Costa, y lo más cercanas posible a las propuestas previamente para aplicaciones de representación de conocimiento. También controlaremos su buen comportamiento desde el punto de vista algebraico, es decir, mostraremos que tal lógica podrá escogerse de tal forma que su algebrizabilidad se garantice. Esperamos que esta propuesta abra nuevas iniciativas a las aplicaciones posibles de las lógicas anotadas.

4.7.3. Anotaciones en birretículos

Se presentan sistemas con las anotaciones en un birretículo entrelazado, estructuras que han sido ampliamente estudiadas y usadas en aplicaciones teóricas de programación lógica, cercanos a P_τ, $SP\mathcal{T}$ y SAL, pero con importantes diferencias en el concepto de "buen comportamiento de las fórmulas", y en la presentación novedosa de anotaciones de anotaciones. Nuevamente la lógica con las anotaciones, goza de buen comportamiento, al menos desde el punto de vista algebraico.

Capítulo 5

Lógicas Anotadas en Semigrupos

Fijaremos $\mathcal{A} = \langle A, F \rangle$ una matriz de interpretación de verdad para el sistema, donde F es un filtro (conjunto de los elementos distinguidos) en A.

Sea $\mathfrak{S}$ un semigrupo con identidad e, que actúa transitivamente sobre A. Fijaremos la acción $*$ del semigrupo $\mathfrak{S}$ en el dominio A, es decir, $* : \mathfrak{S} \times A \longrightarrow A$ es una función fija, usaremos la notación estándar $*(\sigma, a) = \sigma a$. Cabe señalar las condiciones $ed = d$, $\gamma(\sigma d) = (\gamma\sigma)d$ y que la transitividad significa que dados $a, b \in A$ existe un $\sigma \in \mathfrak{S}$ tal que $\sigma a = b$.

Sea ν un elemento, especialmente seleccionado para representar la "negación," en $\mathfrak{S}$ diferente del elemento identidad, es decir, $\nu \in \mathfrak{S}$ y $\nu \neq e$.

Además supondremos que F es un filtro que contiene elementos paraconsistentes, es decir, existe por lo menos un $a \in A$ tal que $a \in F$ y $\nu a \in F$.

El lenguaje $\mathcal{L}$ está formado por un conjunto P de letras proposicionales, los conectivos lógicos $\wedge, \vee, \rightarrow, \neg$ y los operadores de anotación f_λ, uno por cada $\lambda \in \mathfrak{S}$. Las fórmulas están dadas en forma estándar agregando la regla: si A fórmula de $\mathcal{L}$ entonces $f_\lambda A$ también lo es.

5.1. El sistema $OP_{\mathfrak{S}}$

Para cada $a \in A$ existen $\sigma \in \mathfrak{S}$ y $b \in A$ tales que $a = \sigma b$, puesto que la acción es transitiva.

Dada $p \in P$ definimos

$$p^0 = \neg(\bigvee_{\lambda \in \mathfrak{S}} f_\lambda p \wedge \neg f_\lambda p)$$

5.1.1. Los axiomas $OP_{\mathfrak{S}}$

El sistema $OP_{\mathfrak{S}}$ consta de la lógica intuicionista positiva (LIP); ver sección 1.3, en la cual se tiene incluída como única regla de inferencia Modus Ponens; y de los axiomas para las anotaciones y las negaciones, que listamos a continuación.

Axiomas para las anotaciones

[$\mathfrak{S}_1$] $f_e A \leftrightarrow A$

[$\mathfrak{S}_2$] $f_\lambda(A \oslash B) \leftrightarrow f_\lambda A \oslash f_\lambda B$, para todo $\lambda \in \mathfrak{S}$, donde $\oslash \in \{\vee, \wedge, \rightarrow\}$.

[$\mathfrak{S}_3$] $f_\lambda f_\mu A \leftrightarrow f_{\lambda\mu} A$ para todos $\lambda, \mu \in \mathfrak{S}$

El axioma $\mathfrak{S}_1$ nos indica que toda fórmula puede considerarse como una fórmula anotada. El axioma $\mathfrak{S}_2$ asegura que f_σ es un homomorfismo con respecto a los conectivos distintos a la negación. Mientras que $\mathfrak{S}_3$ refleja la intuición que las anotaciones se interpretarán a través de la acción.

Axiomas para la negación

[$\neg_1$] $\neg p^0 \rightarrow (f_\nu p \leftrightarrow \neg p)$

[$\neg_2$] $p^0 \rightarrow (p \vee \neg p)$

[$\neg_3$] $p^0 \rightarrow ((q \rightarrow p) \rightarrow ((q \rightarrow \neg p) \rightarrow \neg p))$

[$\neg_4$] $p^0 \rightarrow (\neg\neg p \leftrightarrow p)$

[$\neg_5$] $p^0 \rightarrow (\neg f_\lambda p \leftrightarrow f_\lambda \neg p)$

Las nociones de demostración (finita en nuestro caso), teorema y relación de consecuencia para el sistema $OP_{\mathfrak{S}}$ son definidas en la forma usual.

Como una consecuencia inmediata de incluir LIP, tenemos el siguiente resultado.

Proposición 5.1 *Son válidos todos los teoremas de la lógica intuicionista positiva.*

Además se pueden mostrar resultados clásicos para aquellas fórmulas marcadas con el símbolo " ^{0} " como por ejemplo reducción al absurdo, que se deduce de los axiomas $\neg_2$ y $\neg_3$.

Proposición 5.2 *Si* $\Gamma, A \vdash A^0 \wedge B^0 \wedge B \wedge \neg B$, *entonces* $\Gamma \vdash \neg A$.

5.2. Semántica para $OP_{\mathfrak{S}}$

Una *interpretación* I para el sistema $OP_{\mathfrak{S}}$ es una función I definida sobre el conjunto de las letras proposicionales P en el álgebra de interpretación de verdad del sistema $\mathbf{A}$, es decir, $I : P \longrightarrow \mathbf{A}$ tal que :

- $I(f_\lambda p) = \lambda I(p)$
- $I(\neg f_\lambda p) = \nu\lambda I(p)$
- $I(\neg p) = \nu I(p)$
- $I(f_\lambda \neg p) = \lambda\nu I(p)$

la cual se extiende recursivamente en forma estándar al conjunto de todos los hiperliterales.

Una *valuación* v_I es una función del conjunto de todos los hiperliterales al conjunto $\{0,1\}$, tal que $v_I(h) = 1$ si $I(h) \in F$ y $v_I(h) = 0$ en otro caso. Esta valuación v_I se extiende al conjunto de todas las fórmulas por:

1. $v_I(\neg A) = 1$ si y sólo si $v_I(A) = 0$.
2. $v_I(A \oslash B) = v_I(A) \circledast v_I(B)$, donde $*$ representa un conectivo binario y $\circledast$ la respectiva operación booleana sobre $\{0,1\}$.
3. $v_I(f_\lambda(A \oslash B)) = v_I(f_\lambda(A)) \circledast v_I(f_\lambda(B))$.
4. $v_I(f_\lambda(\neg A)) = v_I(\neg f_\lambda A)$.

Es importante observar que la paraconsistencia está garantizada por las condiciones iniciales. Puesto que F contiene elementos paraconsistentes, existe $a \in A$ tal que $a, \nu a \in F$. Luego si para alguna letra proposicional p hacemos $I(p) = a$ entonces $I(\neg p) = \nu a$, y así $I(p) = 1$ y $I(\neg p) = 1$. Por tanto, $v_I(p \wedge \neg p) = 1$. Luego nuestro sistema es paraconsistente en un sentido fuerte, pues pueden existir contradicciones verdaderas.

Es oportuno decir que esto no puede suceder con fórmulas clásicas, más aún A^0 implica que no son equivalentes en general las fórmulas $f_\nu A$ y $\neg A$. Ilustremos por ejemplo el caso $A = p \wedge \neg p$ suponiendo que $I(p) = a$ y que $a, \nu a, \nu\nu a \in F$. Entonces $f_\nu A \leftrightarrow f_\nu p \wedge f_{\nu\nu} p$ y por tanto $v_I(f_\nu A) = 1$ mientras que $v_I(\neg A) = 0$ ya que $v_I(A) = 1$, pues $v_I(p) = v_I(\neg p) = 1$.

Las definiciones de validez se definen en la forma estándar, y usaremos la notación $\models_{OP_{\mathfrak{S}}} \varphi$ para decir que φ es universalmente válida.

Lema 5.1 *Para toda valuación v_I se cumple que $v_I(A^0) = 1$ si y sólo si A es compleja.*

Demostración

El caso en que A es un hiperliteral se reduce a considerar los subcasos p, $\neg p$, $f_\lambda p$ y $f_\lambda \neg p$. Veamos el primer subcaso; por la definición de p^0 tenemos que $\neg p^0 \leftrightarrow \bigvee_{\lambda \in \mathfrak{S}} f_\lambda p \wedge \neg f_\lambda p$. Como la acción es transitiva, para todo $a \in \mathbf{A}$ existen $\lambda_a \in \mathfrak{S}$ y $b_a \in \mathbf{A}$ tales que $a = \lambda_a b_a$. Este hecho, junto a la existencia de elementos paraconsistentes nos permite asegurar que $v_I(\neg p^0) = 1$, con lo cual $v_I(p^0) = 0$. Los otros subcasos son análogos.

Para el caso en que A es una fórmula compleja tenemos que $f_\lambda \neg A \leftrightarrow \neg f_\lambda A$, y $v_I(f_\lambda A \wedge f_\lambda \neg A) = v_I(f_\lambda A) \wedge v_I(f_\lambda \neg A) = v_I(f_\lambda A) \wedge v_I(\neg f_\lambda A) = v_I(f_\lambda A) \wedge \neg v_I(f_\lambda A)$, con lo cual $v_I(f_\lambda A \wedge f_\lambda \neg A) = 0$, y por tanto $v_I(A^0) = 1$ ∎

Teorema 5.2 $\Gamma \vdash_{OP_{\mathfrak{S}}} \varphi$ *implica que* $\Gamma \models_{OP_{\mathfrak{S}}} \varphi$.

Demostración

La parte de la lógica intuicionista positiva es igual que en el caso clásico, por las definiciones semánticas dadas.

El axioma $\mathfrak{S}_1$, por inducción se reduce a a verificar que $I(f_e A) = I(A)$, el cual se tiene porque, de la definición de interpretación, $I(f_e A) = eI(A)$, y de la acción de semigrupo $eI(A) = I(A)$.

El axioma $\mathfrak{S}_2$; por los item 3 y 2 de la definición de valuaciones, respectivamente, tenemos: $v_I(f_\lambda(A * B)) = v_I(f_\lambda(A)) \circledast v_I(f_\lambda(B))$ y $v_I(f_\lambda A * f_\lambda B) = v_I(f_\lambda(A)) \circledast v_I(f_\lambda(B))$. Por tanto $v_I(f_\lambda(A * B)) \leftrightarrow v_I(f_\lambda A * f_\lambda B)$.

El axioma $\mathfrak{S}_3$; por inducción se reduce a a verificar que $I(f_\lambda f_\mu A) \leftrightarrow I(f_{\lambda\mu} A)$, lo cual se obtiene de la definición de interpretación, puesto que $I(f_\lambda f_\mu A) = \lambda I(f_\mu A) = \lambda\mu I(A)$ y $I(f_{\lambda\mu} A) = \lambda\mu I(A)$.

En cuanto al axioma $\neg_1$, el caso en que p es una fórmula compleja, se obtiene de forma inmediata por el lema 5.1 y de las definiciones clásicas dadas. Para el caso en que p es un hiperliteral, se reduce a verificar que $I(f_\nu p) = I(\neg p)$, que se tiene por la definición de interpretación.

Los axiomas $\neg_2, \neg_3, \neg_4$ y $\neg_5$ son consecuencia inmediata del lema 5.1. ∎

5.3. Algebrización

El objetivo central de esta sección es mostrar la algebrización de la lógica $OP_{\mathfrak{S}}$, para lo cual usaremos la caracterización de algebrizabilidad de Blok-Pigozzi (ver el teorema 2.4).

Tomaremos como conjunto de ecuaciones de definición $\delta(A) = A \wedge A$ y $\epsilon(A) = A \to A$, y como sistema de fórmulas de equivalencia el conjunto Δ formado por

- $\Delta_{\leftrightarrow}(A, B) = A \leftrightarrow B$
- $\Delta_{\neg}(A, B) = \neg A \leftrightarrow \neg B$
- $\Delta_{\lambda}(A, B) = f_\lambda A \leftrightarrow f_\lambda B$

Antes de presentar el resultado principal de esta sección necesitaremos de algunos resultados previos, en los cuales mantenemos las asignaciones previamente presentadas.

Lema 5.3 $p\Delta q \vdash p^0 \leftrightarrow q^0$

Demostración

Como $p\Delta q$ entonces tenemos que $f_\lambda p \leftrightarrow f_\lambda q$, y se demuestra $\neg f_\lambda p \leftrightarrow \neg f_\lambda q$. Así $p^0 = \neg(\bigvee_{\lambda \in \mathfrak{S}} f_\lambda p \wedge \neg f_\lambda p)$ y obtenemos que $p^0 \leftrightarrow \neg(\bigvee_{\lambda \in \mathfrak{S}} f_\lambda q \wedge \neg f_\lambda q) = q^0$ ∎

Proposición 5.3 $p^0 \leftrightarrow (f_\lambda p)^0$

Demostración

Usando $\neg_4$ tenemos que
$(f_\lambda p)^0 \leftrightarrow \neg(\bigvee_{\rho \in \mathfrak{S}} f_\rho f_\lambda p \wedge \neg f_\rho f_\lambda p)$. De donde gracias a $\mathfrak{S}_3$ podemos afirmar que $(f_\lambda p)^0 \leftrightarrow \neg(\bigvee_{\rho \in \mathfrak{S}} f_{\rho\lambda} p \wedge \neg f_{\lambda\lambda} p)$, y por $\neg_5$ tenemos $(f_\lambda p)^0 \leftrightarrow \neg(\bigvee_{\rho \in \mathfrak{S}} f_{\rho\lambda} p \wedge f_{\rho\lambda} \neg p)$, y por tanto $(f_\lambda p)^0 \leftrightarrow \neg(\bigvee_{\sigma \in \mathfrak{S}} f_\sigma p \wedge f_\sigma \neg p) = p^0$ ∎

Proposición 5.4 $p^0 \leftrightarrow (\neg p)^0$

Demostración

Usando $\neg_4$ tenemos que $(\neg p)^0 = \neg(\bigvee_{\lambda\in\mathfrak{S}} f_\lambda\neg p \wedge \neg f_\lambda\neg p)$. Con lo cual usando los axiomas $\neg_4$ y $\neg_5$ obtenemos que $(\neg p)^0 \leftrightarrow \neg(\bigvee_{\lambda\in S} \neg f_\lambda p \wedge f_\lambda p) \leftrightarrow p^0$ ∎

Teorema 5.4 *El sistema $OP_{\mathfrak{S}}$ es algebrizable.*

Demostración

Los items [1], [2], y [3] son inmediatos.

[4] (a) $A\Delta B \vdash \neg A\Delta\neg B$

Por la hipótesis tenemos:

1. $A \leftrightarrow B$
2. $\neg A \leftrightarrow \neg B$
3. $f_\sigma A \leftrightarrow f_\sigma B$ para todo $\sigma \in \mathfrak{S}$

Del item 2 se tiene de inmediato $(i) : \Delta_{\leftrightarrow}(\neg A, \neg B)$.

Para obtener $(ii) : \Delta_\neg(\neg A, \neg B)$; tenemos dos casos. El primero cuando se tiene A^0, por lema 5.3 se tiene también B^0. Así por el axioma $\neg_4$ tenemos $\neg\neg A \leftrightarrow A$ y $\neg\neg B \leftrightarrow B$, que junto al item 1 de la hipótesis nos permite concluir $\neg\neg A \leftrightarrow \neg\neg B$, es decir, $\Delta_\neg(\neg A, \neg B)$. El segundo caso cuando se tiene $\neg A^0$ por la proposición 5.3 también se tiene $\neg B^0$. Con lo cual usando la proposición 5.4 además tenemos $\neg(\neg A)^0$ y $\neg(\neg B)^0$. Así usando los axiomas $\neg_1$ y $\mathfrak{S}_3$ tenemos que $\neg\neg A \leftrightarrow f_{\nu\nu}A$ y $\neg\neg B \leftrightarrow f_{\nu\nu}B$. Con lo cual usando el item 3 de la hipótesis nos permite concluir que $\Delta_\neg(\neg A, \neg B)$.

Dado $\sigma \in \mathfrak{S}$, para obtener $(iii) : \Delta_\sigma(\neg A, \neg B)$; tenemos también dos casos. El primero cuando se tiene A^0, de nuevo por lema 5.3 se tiene también B^0. Así por el axioma $\neg_5$ $f_\sigma\neg A \leftrightarrow \neg f_\sigma A$ y $f_\sigma\neg B \leftrightarrow \neg f_\sigma B$. Pero por el item 3 de la hipótesis sabemos que $f_\sigma A \leftrightarrow f_\sigma B$, con lo cual se muestra que $\neg f_\sigma A \leftrightarrow \neg f_\sigma B$, y así $f_\sigma\neg A \leftrightarrow f_\sigma\neg B$, es decir, $\Delta_\sigma(\neg A, \neg B)$. El segundo caso cuando se tiene $\neg A^0$, de nuevo, por la proposición 5.3 también se tiene $\neg B^0$. Así usando los axiomas $\neg_1$ y $\mathfrak{S}_3$ tenemos que $f_\sigma\neg A \leftrightarrow f_{\sigma\nu}A$ y $f_\sigma\neg B \leftrightarrow f_{\sigma\nu}B$. Con lo cual usando el item 3 de la hipótesis nos permite concluir que $\Delta_\sigma(\neg A, \neg B)$.

Con $(i), (ii)$ y (iii) se concluye $\neg A\Delta\neg B$.

[4] (b) $A\Delta B \vdash f_\sigma A\Delta f_\sigma B$

De nuevo, por la hipótesis tenemos:

1. $A \leftrightarrow B$

2. $\neg A \leftrightarrow \neg B$

3. $f_\sigma A \leftrightarrow f_\sigma B$ para todo $\sigma \in \mathfrak{S}$

Del item 3 se tiene de inmediato $(i) : \Delta_{\leftrightarrow}(f_\sigma A, f_\sigma B)$.

Para ver $(i) : \Delta_{\neg}(f_\sigma A, f_\sigma B)$, tenemos dos casos. El primero cuando se tiene A^0, entonces se tiene que $A^0 \vdash \neg f_\sigma A \leftrightarrow f_\sigma \neg A$, y por parte [4] (a) $f_\sigma \neg A \leftrightarrow f_\sigma \neg B$, y también que $f_\sigma \neg B \leftrightarrow \neg f_\sigma B$.

Ahora si se da $\neg A^0$, dentro del sistema. Entonces se tiene que $\neg A^0 \rightarrow (\neg A \leftrightarrow f_\nu A)$. Luego como $\neg A^0 \rightarrow \neg (f_\lambda A)^0$, veamos

$\neg f_\lambda A \leftrightarrow f_\nu f_\lambda A \leftrightarrow f_{\nu\lambda} A \leftrightarrow f_{\nu\lambda} B \leftrightarrow f_\nu f_\lambda B \leftrightarrow \neg f_\lambda B$, puesto que también se tiene $(f_\lambda B)^0$. Por tanto tenemos $(ii) : \Delta_{\neg}(f_\sigma A, f_\sigma B)$.

Dado $\sigma \in \mathfrak{S}$, tenemos que $\Delta_\sigma(f_\lambda A, f_\lambda B) = f_\sigma f_\lambda A \leftrightarrow f_\sigma f_\lambda B$. Pero $f_\sigma f_\lambda A \leftrightarrow f_{\sigma\lambda} A$, y $f_\sigma f_\lambda B \leftrightarrow f_{\sigma\lambda} B$, y por itení 3 de la hipótesis $f_{\sigma\lambda} A \leftrightarrow f_{\sigma\lambda} B$. Con lo cual se tiene $(iii) : \Delta_\sigma(f_\lambda A, f_\lambda B)$.

Con $(i), (ii)$ y (iii) se concluye $f_\sigma A \Delta f_\sigma B$.

[4] (c) se concluye del comportamiento de las anotaciones como homomorfismos y la lógica clásica.

[5] $p \dashv\vdash (p \wedge p)\Delta(p \rightarrow p)$

Veamos $(\dashv)$: es conocido porque de los dos primeros axiomas y la regla modus ponens $\vdash p \rightarrow p$ y usando una de las implicaciones de $\Delta_{\leftrightarrow}$, para ser precisos la de izquierda a derecha $\vdash (p \rightarrow p) \rightarrow p \wedge p$. De donde se concluye $\vdash p$ puesto que en la lógica positiva $p \wedge p \rightarrow p$. En el otro sentido de la equivalencia, por axioma $\wedge_3$ tenemos que $p \vdash p \wedge p$, con lo cual $p \vdash (p \rightarrow p) \rightarrow p \wedge p$. Por tanto $p \vdash (p \rightarrow p) \leftrightarrow p \wedge p$.

Los casos $p \vdash (p \wedge p)\Delta_{\neg}(p \rightarrow p)$ y $p \vdash (p \wedge p)\Delta_\sigma(p \rightarrow p)$ son consecuencia de resultados estándar de la lógica intuicionista positiva y el caso anterior ∎

5.4. Anotaciones respetan algebrización

En esta sección impondremos ciertas condiciones a nuestras lógicas para efectos de garantizar su algebrización. Tales condiciones las iremos presentando a través del desarrollo de las mismas. Entre éstas, imponemos para el lenguaje la condición de incluir los símbolos binarios $\rightarrow$ e $\wedge$, el símbolo unario $\neg$; y como es lo usual $A \leftrightarrow B$ abreviará la expresión $(A \rightarrow B) \wedge (B \rightarrow A)$.

El objetivo central de esta sección es mostrar que la algebrización de la lógica S es una condición suficiente para la algebrización de la lógica S_G, obtenida del sistema S agregando anotaciones sobre un semigrupo G. En este punto es muy importante exigir al sistema S que contenga la lógica intuicionista positiva.

Al lenguaje de S le agregamos un símbolo de operación unaria f_σ para cada constante de anotación $\sigma \in G$. Las fórmulas del nuevo sistema se definen en la manera estándar, agregando a las reglas de formación de fórmulas que *si A es una fórmula y σ es una anotación de semigrupo, entonces $f_\sigma A$ es una fórmula.* Las demás definiciones básicas son estándar.

El sistema S_G está dado por la presentación del sistema S, agregando los siguientes axiomas para manejar las anotaciones.

5.4.1. Axiomas de las anotaciones de semigrupo

Para cada $\sigma, \gamma \in G$:

[G_1] $f_e A \leftrightarrow A$.

[G_2] $f_\sigma \omega(A_1, A_2, \ldots, A_n) \leftrightarrow \omega(f_\sigma A_1, f_\sigma A_2, \ldots, f_\sigma A_n)$, para todo símbolo de conectivo n−ario ω, $\omega \neq \neg$, en el lenguaje de S.

[G_3] $f_\sigma f_\gamma A \leftrightarrow f_{\sigma\gamma} A$.

[G_4] $(A \leftrightarrow B) \rightarrow (f_\sigma A \leftrightarrow f_\sigma B)$.

El axioma G_1 nos indica que toda fórmula puede considerarse como una fórmula anotada. El axioma G_2 asegura que f_σ es un homomorfismo con respecto a los conectivos distintos a la negación. Mientras que G_3 refleja la intuición que las anotaciones se interpretarán a través de la acción, y G_4 la monotonía de la acción con respecto a la implicación.

5.4.2. Axiomas de la negación

Para cada $\sigma \in G$:

[$\neg_1$] $\neg p^0 \rightarrow (f_\nu p \leftrightarrow \neg p)$

[$\neg_2$] $p^0 \rightarrow (p \vee \neg p)$

[$\neg_3$] $p^0 \rightarrow ((q \rightarrow p) \rightarrow ((q \rightarrow \neg p) \rightarrow \neg p))$

[$\neg_4$] $p^0 \rightarrow (\neg\neg p \leftrightarrow p)$

$[\neg_5\,]\ p^0 \to (\neg f_\lambda p \leftrightarrow f_\lambda \neg p)$

Partiremos de que S es un sistema algebrizable, con ecuaciones de definición $\delta \approx \epsilon$, y un sistema de fórmulas de equivalencia Δ al que le imponemos la condición de incluir $\Delta_\leftrightarrow$, que representa al conjunto de todos las fórmulas de la forma $A \leftrightarrow B$, cuando A y B son fórmulas del nuevo sistema.

Observación 5.1 *Si Δ es un sistema de fórmulas de equivalencia para S entonces $\Delta_\leftrightarrow \subseteq \Delta$.*

Formamos el conjunto $\Delta' = \Delta \bigcup \{\Delta_\sigma : \sigma \in G\}$, donde la expresión $A\Delta_\sigma B$ representa el conjunto de todas las fórmulas de la forma $f_\sigma A \leftrightarrow f_\sigma B$, cuando σ recorre todo G.

Teniéndo las condiciones antes citadas obtenemos el siguiente resultado.

Teorema 5.5 *La lógica S_G es algebrizable con sistema de fórmulas de equivalencia Δ' y con ecuaciones de definición $\delta \approx \epsilon$.*

Demostración

[1] $\vdash A\Delta' A$

El resultado se tiene pues $\vdash f_\lambda A \leftrightarrow f_\lambda A$ es un teorema de lógica intuicionista positiva.

[2] $A\Delta' B \vdash B\Delta' A$

Inmediato de la hipótesis y la simetría de la $\leftrightarrow$.

[3] $A\Delta' B, B\Delta' C \vdash A\Delta' C$

Inmediato.

[4] [a] $A_1\Delta' B_1, \ldots, A_n\Delta' B_n \vdash \omega(A_1, A_2, \ldots, A_n)\Delta'\omega(B_1, B_2, \ldots, B_n)$.

Por hipótesis $\vdash \omega(A_1, A_2, \ldots, A_n)\Delta\omega(B_1, B_2, \ldots, B_n)$, y dentro de ésta tenemos que $\vdash \omega(A_1, A_2, \ldots, A_n) \leftrightarrow \omega(B_1, B_2, \ldots, B_n)$ de donde, usando la monotonía de la acción con respecto a la implicación, se puede mostrar en el sistema que $\vdash \omega(A_1, A_2, \ldots, A_n)\Delta_\sigma\omega(B_1, B_2, \ldots, B_n)$ con lo que se concluye le resultado.

[4] [b] $A\Delta' B \vdash f_\sigma A\Delta' f_\sigma B$.

$f_\sigma A\Delta f_\sigma B$ se obtiene por la hipótesis $A\Delta B$ y el axioma G_2. De otro lado sabemos que $f_\gamma f_\sigma A \leftrightarrow f_{\gamma\sigma} A$ por G_3. Así $f_\gamma f_\sigma A \leftrightarrow f_{\gamma\sigma} B$ por la hipótesis $\Delta_{\gamma\sigma}$. Luego aplicando de nuevo el axioma G_3 se obtiene que $A\Delta' B \vdash f_\sigma A\Delta' f_\sigma B$.

[5] $A \dashv\vdash \delta(A)\Delta'\epsilon(A)$

$\delta(A)\Delta'\epsilon(A) \vdash A$ inmediato de la hipótesis.

De otro lado por hipótesis $A \vdash \delta(A)\Delta\epsilon(A)$, y por tanto $A \vdash \delta(A) \leftrightarrow \epsilon(A)$, con lo cual se concluye el resultado ■

Corolario 5.6 *Si S es un sistema finitamente algebrizable y $|G|$ es finito entonces el sistema S_G es finitamente algebrizable.*

Capítulo 6

Lógicas Anotadas Fuzzy

Las lógicas fuzzy gestadas por Zadeh han dado origen a multitud de aplicaciones, así como a un amplio espectro de estudios; dentro de los más significativos, las lógicas multivaluadas. Aquí nos interesaremos por aquellas que han sido usadas para modelar la inferencia automática a través del principio de resolución, más precisamente nos centraremos a las presentadas por Lee [53], y Weigert et. al. [78]. Esta última referencia ha sido recientemente reformulada por James Lu et. al. en [58], presentando un marco unificador entre los principios de resolución de lógicas llamadas signadas (que preceden su versión de lógicas anotadas), las anotadas y el de Weigert. La idea fundamental a desarrollarse es dar un sistema lógico formal que recoja la presentación de Weigert y que generalize el concepto de satisfactibilidad de Lee. Para este efecto creemos que el camino más natural son las lógicas anotadas, que surgen como alternativa a la lógica clásica, la que como mencionamos en la introducción resulta ineficaz para el razonamiento de bases de datos inconsistentes, pues en la lógica clásica de una contradicción se deduce cualquier cosa. Las lógicas anotadas fueron propuestas originalmente por V. S. Subrahmanian, luego estudiadas formalmente a través del sistema $P\mathcal{T}$ [40]. Ahí se demuestra que estos sistemas son paraconsistentes, y son propuestos como una base teórica para el razonamiento de bases de datos con inconsistencias. Después R. Lewin et. al. [57], introducen una version estructural $SP\mathcal{T}$ (es importante señalar que el sistema original $P\mathcal{T}$ no es estructural y por tanto no algebrizable), y dan una prueba formal que el sistema $P\mathcal{T}$ se puede interpretar en $SP\mathcal{T}$.

Nosotros desarrollaremos una lógica anotada cercana a $SP\mathcal{T}$ y al sistema C_n de da Costa, obviamente que satisfaga el objetivo propuesto, lo más cercana posible a la propuesta de Weigert y cerca del énfasis de Lee. Además controlaremos su buen comportamiento desde el punto de vista algebraico, es decir, mostraremos que tal lógica podrá escogerse de tal forma que su algebrizabilidad se garantice. Por demás cabe recalcar que este ejemplo nos abre una ventana a la amplitud de las presentaciones y aplicaciones posibles de las lógicas anotadas.

6.1. El álgebra de anotaciones

Fijaremos como álgebra de anotaciones a

$$\daleth = ([-1, 1], \sqcup, \sqcap, \otimes, \rightsquigarrow)$$

donde $\sqcup$, y $\sqcap$ son las operaciones binarias definidas por

$$x \sqcup y = max\,\{x, y\}$$

$$x \sqcap y = min\,\{x, y\}$$

para $x, y \in \daleth$.

Tenemos una relación de orden estándar sobre $\daleth$ dada por

$$x \leq y \text{ si y sólo si } \quad x \sqcap y = y$$

para $x, y \in \daleth$.

Además las operaciones $\otimes$, y $\rightsquigarrow$ están dadas por

$$x \otimes y = xy$$

$$x \rightsquigarrow y = \begin{cases} 1 & \text{si } x \leq y \\ y & \text{en otro caso} \end{cases}$$

para $x, y \in \daleth$.

Es importante notar que $\otimes$ corresponde al producto habitual, y $\rightsquigarrow$ es una implicación intuicionista.

6.2. El Sistema $OP_\daleth$

El lenguaje del sistema $OP_\daleth$ estará dado por los símbolos $\wedge, \vee, \rightarrow, \Box, \top$, un conjunto P de letras proposicionales $p,q,r,...$, un símbolo de operación unaria f_x para cada constante de anotación $x \in [-1, 1]$, y paréntesis.

Como es usual el conjunto $\mathcal{F}m$ de fórmulas de $OP_\daleth$ está dado recursivamente:

1. $\Box$ y $\top$ son fórmulas.

2. Si p es una letra proposicional, entonces p es una fórmula.

3. Si A y B son fórmulas, entonces $A \wedge B$, $A \vee B$ y $A \rightarrow B$ son fórmulas.

4. Si A es una fórmula y x es una constante de anotación, entonces $f_x A$ es una fórmula.

5. Una expresión es una fórmula si, y sólo si, se obtiene por la aplicación de las reglas antes citadas.

Nos referiremos a $f_x A$ como *anotación fuzzy*, esto en vista de que al menos en nuestra semántica va a "coincidir" con la presentación del operador fuzzy de Weigert. Intuitivamente, la anotación fuzzy dada por fórmula $f_x A$ puede pensarse como *una creencia sobre el grado de veracidad* de la proposición A. Más aún proponemos se piensen las anotaciones como generalizaciones de la negación. Con el símbolo $\Box$ intentamos representar la indecision total, es decir, aquello sobre lo cual no se puede determinar grado alguno de veracidad o falsedad. Mientras que, con el símbolo $\top$ se intentara capturar lo absolutamente verdadero.

Antes de presentar el sistema, queremos dar unas definiciones previas, dentro de estas, la definición de la negación como una anotación particular, insistiendo en nuestra propuesta de ver las anotaciones como generalización de negaciones. También resaltamos la presentación del símbolo A^0 (como ya es usual al estilo Da Costa en este tipo de lógicas) para enfatizar más adelante el carácter clásico de la proposición A.

Definición 6.1 *Se definen*

1. $\neg A = f_{-1} A$

2. $A \leftrightarrow B = (A \rightarrow B \wedge B \rightarrow A)$

3. $A^0 = A \vee \neg A$

4. $\bot = \neg \top$

6.2.1. Los axiomas $OP_{\daleth}$:

El sistema $OP_{\daleth}$ está constituido por la lógica intuicionista positiva (LIP), anexando los axiomas que describimos a continuación.

[$\daleth_1$] $f_0(A) \leftrightarrow \Box$.

[$\daleth_2$] $f_1 A \leftrightarrow A$.

[$\daleth_3$] $f_x(f_y A) \leftrightarrow f_{x \oslash y} A$ para todos $x, y \in \daleth$.

[$\daleth_4$] $f_x(A \vee B) \leftrightarrow f_x A \vee f_x B$, para $x > 0$.

[$\daleth_5$] $f_{-1}(A \vee B) \leftrightarrow f_{-1}A \wedge f_{-1}B$.

[$\daleth_6$] $f_x(A \wedge B) \leftrightarrow f_xA \wedge f_xB$, para $x > 0$.

[$\daleth_7$] $f_{-1}(A \wedge B) \leftrightarrow f_{-1}A \vee f_{-1}B$.

[$\daleth_8$] $f_x(A \rightarrow B) \leftrightarrow ((f_xA \rightarrow f_xB) \rightarrow f_x\top)$, para $x > 0$.

[$\daleth_9$] $f_{-1}(A \rightarrow B) \rightarrow (((f_{-1}B \rightarrow f_{-1}A) \rightarrow f_{-1}A) \rightarrow f_{-1}B)$.

[$\daleth_{10}$] $f_{-1}(A \rightarrow A) \leftrightarrow \perp$.

[$\daleth_{11}$] $(f_xA)^0 \rightarrow (A)^0$.

Es oportuno notar que los operadores de anotación no se comportan como homomorfismos. El axioma $\daleth_1$ nos da una anotación modulativa que se aclarará un tanto mejor en la sección de la semántica; mientras que el axioma $\daleth_2$ nos indica que toda fórmula puede considerarse como una fórmula anotada. El axioma $\daleth_{11}$ nos dice que si una fórmula anotada goza de buen comportamiento es porque antes de la anotación ya lo poseía.

6.2.2. La reglas de inferencia de $OP_\daleth$:

Para todas A y B fórmulas de $OP_\daleth$:

[$R_{\neg(1)}$] $\dfrac{A^0}{(A \wedge \neg A) \rightarrow B}$

[$R_{\neg(2)}$] $\dfrac{A^0 \wedge B^0}{(A \wedge B)^0 \wedge (A \vee B)^0 \wedge (A \rightarrow B)^0}$

[$R_{\neg(3)}$] $\dfrac{A^0 \leftrightarrow B^0}{(A \rightarrow B) \rightarrow (\neg A \rightarrow \neg B)}$

Las nociones de demostración (finita en nuestro caso), teorema y relación de consecuencia para el sistema $OP_\daleth$ son definidas en la forma usual.

Observación 6.1 *La regla $R_\neg(1)$ corresponde a la regla de reductio ad absurdum para aquellas fórmulas con comportamiento clásico.*

Como el sistema $OP_\daleth$ contiene la lógica intuicionista positiva tenemos de inmediato el siguiente resultado.

Proposición 6.1 *Son válidos todos los teoremas de la lógica intuicionista positiva.*

Proposición 6.2 *Las siguientes equivalencias son válidas en el sistema* $OP_{\daleth}$:

$[\daleth_{12}] \qquad A^0 \leftrightarrow (\neg A)^0$

$[\daleth_{13}] \qquad A^0 \leftrightarrow (A^0)^0$

Demostración

$[\daleth_{12}]$: Tenemos que $(\neg A)^0 = \neg A \vee \neg\neg A = \neg A \vee f_{-1}f_{-1}A$, y por $[\daleth_3]$ y $[\daleth_2]$ se obtiene que $(\neg A)^0 \leftrightarrow \neg A \vee A$. De donde por resultados clásicos de la lógica intuicionista positiva se concluye el resultado.

$[\daleth_{13}]$: Tenemos que $(A^0)^0 = (A)^0 \vee \neg(A^0.)$ Pero $\neg(A^0) = \neg(A \vee \neg A)$, que por definición de negación, y los axiomas $[\daleth_5]$ y $[\daleth_3]$ nos conduce a que $\neg(A^0) = \neg A \vee A$. De donde por resultados clásicos de la lógica intuicionista positiva se concluye la prueba ■

6.2.3. Deducción restringida

Si se restringen las demostraciones al uso exclusivo de la regla de modus ponens como única regla de inferencia, se tiene un teorema de la deducción (restringido).

Proposición 6.3 *Si en la prueba de* Σ, $A \vdash B$ *se usa* MP *como única regla de inferencia, entonces* $\Sigma \vdash A \to B$.

Demostración
Se obtiene de los axiomas$\to_1, \to_2$ y que la única regla de inferencia usada es Modus Ponens. ■

Observación 6.2 *El teorema de la deducción no es válido en general, por ejemplo es inmediato de la regla* $R_{\neg}(1)$ *que* $A^0 \vdash (A \wedge \neg A) \to B$, *mas sin embargo, como veremos más adelante, no es cierto que* $\vdash A^0 \to ((A \wedge \neg A) \to B)$.

En nuestro sistema se demuestra la ley de Peirce para fórmulas con comportamiento clásico, dicha prueba es debida a M. Guillaume [35].

Proposición 6.4 $A^0 \vdash (((A \to B) \to A) \to A)$.

Es pertinente resaltar que al ser probada la ley de Peirce para fórmulas con comportamiento clásico entonces serán válidos todos los teoremas de la lógica intuicionista sólo para todas las fórmulas con comportamiento clásico.

Proposición 6.5 *En nuestro sistema se cumple que:*

1. $(A^0 \wedge B^0) \vdash (A \rightarrow B) \rightarrow ((A \rightarrow \neg B) \rightarrow \neg A)$.

2. $A^0 \vdash (A \rightarrow (\neg A \rightarrow B)$.

Es necesario recalcar que al ser probada la ley de reductio at absurdum para fórmulas con comportamiento clásico entonces serán válidos todos los teoremas de la lógica clásica sólo para todas las fórmulas con comportamiento clásico. En efecto, tenemos el siguiente resultado, sustentado en que los axiomas para los conectivos binarios corresponden a la lógica intuicionista positiva y el comportamiento clásico de las proposiciones simbolizadas por A^0, que está dado por las reglas de inferencia adicionales a modus ponens.

Proposición 6.6 *Si φ $(x_1, x_2, \ldots, x_n)$ es una tautología clásica, entonces:*

1. $\{A_1^0, A_2^0, \ldots, A_n^0\} \vdash \varphi(A_1, A_2, \ldots, A_n)$.

2. *Si la tautología φ $(x_1, x_2, \ldots, x_n)$ no contiene negaciones, entonces en nuestro sistema $\vdash \varphi(A_1, A_2, \ldots, A_n)$.*

6.3. Semántica para $OP_{\daleth}$

Una *interpretación* I para el sistema $OP_{\daleth}$ es una función definida sobre el conjunto de las letras proposicionales $I : P \longrightarrow \daleth$, la cual se extiende recursivamente en forma estándar al conjunto de todas las fórmulas $\mathcal{F}m$ por:

$I_1 :$ $I\ (\Box) = 0.$

$I_2 :$ $I\ (\top) = 1.$

$I_3 :$ $I\ (A \wedge B) = I(A) \sqcap I(B)$

$I_4 :$ $I\ (A \vee B) = I(A) \sqcup I(B)$

$I_5 :$ $I\ (A \rightarrow B) = I(A) \rightsquigarrow I(B)$

$I_6 :$ $I\ (f_x A) = x \otimes I(A)$, para cada $x \in \daleth$.

Si se quiere una interpretación intuitiva, se puede pensar en $I(A)$, para $A \in \mathcal{F}m$, como el factor de confiabilidad de la fórmula A bajo la interpretación I. Intuitivamente una anotación en una fórmula se puede interpretar como la confiabilidad de precisión de el valor de verdad de la fórmula. Así por ejemplo $f_1 A$ asegura la certeza (total) de la asignación del valor de verdad de A, $f_{0,9}A$ asegura que la certeza del valor de A es casi correcta, y $f_{0,1}A$ asegura que la asignación del valor de verdad es casi errada. Es necesario resaltar la diferencia entre la la confiabilidad de precisión de el valor de verdad de la fórmula, y el dado por una interpretación I. Así por ejemplo si $I(A) = -1$ entonces el valor de verdad para $I(f_{-1}A) = 1$; lo que indica la anotación f_{-1} es que el valor $I(A) = -1$ es totalmente errado.

Observación 6.3 *De la definición de interpretación se infiere que:*

I_7 : $I\,(f_{-1}A) = -I(A)$, *para cada fórmula A que no contenga el conectivo $\rightarrow$.*

$$I_8 : \; I\,(f_{-1}(A \rightarrow B)) = \begin{cases} -1 & \text{si } I(A) \leqslant I(B), \\ -I(B) & \text{en otro caso.} \end{cases}$$

Observación 6.4 *Tenemos tres tipos de anotaciones especiales f_{-1}, f_0 y f_1. En efecto $I(f_{-1}A) = (-1)\otimes\, I(A)$, es decir, f_{-1} dice que el valor de verdad asignado para $I(A)$ es errado (totalmente) y lo invierte, con lo cual lo que tenemos es que la anotación f_{-1} da cuenta de la negación. Mientras, $I(f_0A) = (0)\otimes\, I(A) = 0$, es decir, la anotación f_0 es un operador neutro, que nos dice que hay total incertidumbre acerca del valor de verdad $I(A)$. Pero $I(f_1A) = (1)\otimes\, I(A) = I(A)$, es decir, la anotación f_1 asegura que hay total certeza acerca del valor de verdad $I(A)$.*

En concordancia con el concepto de satisfacibilidad fuzzy de Lee, fijaremos un cierto nivel de confiabilidad, que tomaremos mayor o igual que cero. Recordemos que estamos usando el intervalo $[-1, 1]$ en lugar del usual $[0, 1]$, con lo cual el "cero" nuestro representa el "un medio" de Lee.

Definición 6.2 *Aquí daremos algunos términos necesarios para lo que resta del presente trabajo:*

1. *Fijaremos como umbral de confiabilidad (de verdad) a un número $\mathrm{u} \in [-1, 1]$ que sea mayor ó igual que cero.*

2. *Para cada interpretación I y para cada fórmula φ diremos que I u-satisface φ si $I(\varphi) \geq \mathrm{u}$, en tal caso usaremos la notación $I \models_{\mathrm{u}} \varphi$.*

3. *Diremos que una fórmula φ es u-válida si $I \models_{\mathrm{u}} \varphi$, para toda interpretación I; y que φ es válida si φ es u-válida para todo $\mathrm{u} \in \mathbb{I}$ que sea mayor o igual que cero, y usaremos la notación $\models \varphi$.*

4. *En la manera usual se define que φ es una u-consecuencia lógica de Γ, lo cual será denotado por $\Gamma \models_{\mathrm{u}} \varphi$, si para toda interpretación I que tal que para toda $\gamma \in \Gamma$, $I \models_{\mathrm{u}} \gamma$, implica que $I \models_{\mathrm{u}} \varphi$. Definimos análogamente $\Gamma \models \varphi$.*

Observación 6.5 *El fijar un* $u \geqslant 0$ *como umbral de confiabilidad nos da un marco mucho más general que el trabajo de Lee, que fija un sólo valor el "un medio."*

Afirmación 6.1 *Una fórmula* φ *es válida si y sólo si es* $1-$*válida.*

Afirmación 6.2 *Para toda interpretación* I *se tiene que* $I(A^0) = |I(A)|$.

El siguiente resultado, aunque no es difícil de probar, es similar al encontrado en [40], es así que nos permitiremos citarlo sin su prueba.

Proposición 6.7 *Si* $A_1, A_2, \ldots, A_n$ *son fórmulas complejas, y* $\varphi\,(x_1, x_2, \ldots, x_n)$ *es una tautología clásica, entonces* $\models \varphi(A_1, A_2, \ldots, A_n)$*. Además, si en* $\varphi\,(x_1, x_2, \ldots, x_n)$ *no ocurren negaciones, se tiene entonces el importante resultado* $A_1, A_2, \ldots, A_n \models \varphi(A_1, A_2, \ldots, A_n)$.

Teorema 6.3 $\Sigma \vdash A$*, entonces* $\Sigma \models A$.

Demostración
Usando la afirmación 6.1 necesitamos probar que dada una interpretación I tal que $I(A) = a$, $I(B) = b$, y $I(C) = c$, todas las fórmulas φ que sean axiomas tienen valor $I(\varphi) = 1$, y que además la validez se preserva por las reglas de inferencia. Luego por inducción se concluye el resultado. En este camino veamos:

[$\rightarrow_1$] $A \rightarrow (B \rightarrow A)$
Si $a \geqslant b$ el resultado es trivial. Para $a < b$ se tiene que $b \rightsquigarrow a = a$, así la fórmula $\rightarrow_1$ tiene interpretación 1.

[$\rightarrow_2$] $(A \rightarrow B) \rightarrow ((A \rightarrow (B \rightarrow C)) \rightarrow (A \rightarrow C))$
Si $a \leq c$ la conclusión es inmediata. Consideremos el caso $c < a$, dentro del cual hay dos subcasos a considerar:
Subcaso $a \leq b$; aquí tenemos,

$$1 \rightsquigarrow ((a \rightsquigarrow c) \rightsquigarrow c) = c \rightsquigarrow c = 1.$$

Subcaso $b < a$; dentro de éste hay que considerar dos nuevos casos:

caso $b < c$ en el cual,

$$b \rightsquigarrow ((a \rightsquigarrow 1) \rightsquigarrow c) = b \rightsquigarrow (1 \rightsquigarrow c) = b \rightsquigarrow c = 1.$$

caso $c < b$,

$$b \rightsquigarrow ((a \rightsquigarrow c) \rightsquigarrow c) = b \rightsquigarrow (c \rightsquigarrow c) = b \rightsquigarrow 1 = 1.$$

$[\wedge_1]$ $A \wedge B \rightarrow A$
Consecuencia del hecho que $min\,\{a,b\} \leq a$.

$[\wedge_2]$ $A \wedge B \rightarrow B$
Análogo al caso $\wedge_1$.

$[\wedge_3]$ $A \rightarrow (B \rightarrow (A \wedge B))$
Si $a = b$ inmediato.
Para $a < b$, entonces $min\,\{a,b\} \leq b$, y así $a \rightsquigarrow b = 1$.
Para $b < a$, entonces $min\,\{a,b\} \leq a$, y así $a \rightsquigarrow 1 = 1$.

$[\vee_1]$ $A \rightarrow A \vee B$
Consecuencia del hecho que $a \leq max\,\{a,b\}$.

$[\vee_2]$ $B \rightarrow A \vee B$
Análogo al caso $\vee_1$.

$[\vee_3]$ $(A \rightarrow C) \rightarrow ((B \rightarrow C) \rightarrow (A \vee B \rightarrow C))$
Veamos dos casos los otros son análogos,
Caso $b < a < c$, así $a \sqcup b = a$ y por tanto $a \sqcup b \rightsquigarrow c = 1$, lo que hace inmediato el resultado.
Caso $c < b < a$, así $a \sqcup b = a$ y por tanto se tiene $c \rightsquigarrow (c \rightsquigarrow c) = 1$.

$[\daleth_1]$ $f_0(A) \leftrightarrow \Box$.
Es consecuencia inmediata de que para toda interpretación I se tiene por definición $I\ (\Box) = 0$, y que $I(f_0(A)) = 0\otimes\ I(A) = 0$.

$[\daleth_2]$ $f_1 A \leftrightarrow A$.
Es consecuencia inmediata de que para toda interpretación I se tiene por definición $I\ (f_1 A) = 1\otimes\ I(A) = I(A)$.

$[\daleth_3]$ $f_x(f_y A) \leftrightarrow f_{x\otimes y}A$ para todos $x, y \in \daleth$.
Para cuando $x > -1$ y $y > -1$ por definición I_6 se tiene que $I(f_x(f_y A)) = x\otimes I(f_y A) = x \otimes y \otimes I(A) = I(f_{x\otimes y}A)$, para toda interpretación I. Los casos cuando $x = -1$ ó $y = -1$ y A no tiene ocurrencias de $\rightarrow$ son similares al anterior. Pero los casos cuando $x = -1$ ó $y = -1$ y A tiene ocurrencias de $\rightarrow$ se obtiene n del hecho que si $A = B \rightarrow C$ entonces por item I_8 para toda interpretación I se tiene que

$$I(f_{-1}(B \rightarrow C)) = \begin{cases} -1 & \text{si } I(B) \leqslant I(C) \\ (-1)I(\mathrm{C}) & \text{en otro caso.} \end{cases}$$

$[I_4]$ $f_x(A \vee B) \leftrightarrow (f_x A \vee f_x B)$, para $x > 0$.
Es cierto ya que para $x > 0$, $x \otimes max\,\{a, b\} = max\,\{x \otimes a, x \otimes b\}$.

$[I_5]$ $f_{-1}(A \vee B) \leftrightarrow f_{-1}A \wedge f_{-1}B$.
Es cierto ya que $(-1) \otimes max\,\{a, b\} = min\,\{(-1) \otimes a, (-1) \otimes b\}$.

$[I_6]$ $f_x(A \wedge B) \leftrightarrow f_x A \wedge f_x B$, para $x > 0$.
Es cierto ya que para $x > 0$, $x \otimes min\,\{a, b\} = min\,\{x \otimes a, x \otimes b\}$.

$[I_7]$ $f_{-1}(A \wedge B) \leftrightarrow f_{-1}A \vee f_{-1}B$.
Es cierto ya que $(-1) \otimes min\,\{a, b\} = max\,\{(-1) \otimes a, (-1) \otimes b\}$.

$[I_8]$ $f_x(A \rightarrow B) \leftrightarrow ((f_x A \rightarrow f_x B) \rightarrow f_x \top)$, para $x > 0$.
Es consecuencia de que

$$a \rightsquigarrow b = \begin{cases} 1 & \text{si } a \leq b \\ b & \text{en otro caso} \end{cases}$$

y así

$$x \otimes (a \rightsquigarrow b) = \begin{cases} x & \text{si } a \leq b \\ x \otimes b & \text{en otro caso,} \end{cases}$$

y que

$$((x \otimes a \rightsquigarrow x \otimes b) \rightsquigarrow x) = \begin{cases} 1 \rightsquigarrow x = x & \text{si } a \leq b \\ x \otimes b \rightsquigarrow x = x \otimes b & \text{en otro caso.} \end{cases}$$

$[I_9]$ $f_{-1}(A \rightarrow B) \rightarrow (((f_{-1}B \rightarrow f_{-1}A) \rightarrow f_{-1}A) \rightarrow f_{-1}B)$.
Se obtiene como conclusión de

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

lo que implica que

$$((-b \rightsquigarrow -a) \rightsquigarrow -a) \rightsquigarrow -b = -b;$$

y del hecho que

$$(-1) \otimes (a \rightsquigarrow b) = \begin{cases} -1 & \text{si } a \leq b \\ -b & \text{en otro caso.} \end{cases}$$

$[I_{10}]$ $f_{-1}(A \rightarrow A) \leftrightarrow \perp$.
Consecuencia de la definición I_8.

$[I_{11}]$ $\vdash (f_x A)^0 \rightarrow (A)^0$.
Como $|x| \leqslant 1$ entonces $|xa| \leqslant |a|$ ∎

6.3.1. La reglas de inferencia de $OP_{\daleth}$:

Para todas A y B fórmulas de $OP_{\daleth}$:

$[R_{\rightarrow}]$ $\dfrac{A, A \rightarrow B}{B}$

Si $a = 1$, y $a \rightsquigarrow b = 1$ entonces $a \leq b$, y por tanto $b = 1$.

$[R_{\neg(1)}]$ $\dfrac{A^0}{(A \wedge \neg A) \rightarrow B}$

Si $|a| = 1$, entonces $a = 1$ ó $a = -1$ así $a \sqcap -a = -1$, y por tanto $a \sqcap -a \rightsquigarrow b = 1$ para todo b.

$[R_{\neg(2)}]$ $\dfrac{A^0 \wedge B^0}{(A \wedge B)^0 \wedge (A \vee B)^0 \wedge (A \rightarrow B)^0}$

Si $|a| = |b| = 1$, entonces $a = 1$ ó $a = -1$, y $b = 1$ ó $b = -1$ así $a \sqcap b = 1$ ó $a \sqcap b = -1$, $a \sqcup b = 1$ ó $a \sqcup b = -1$, y $a \rightsquigarrow b = 1$ ó $a \rightsquigarrow b = -1$ por tanto $|a \sqcap b| = |a \sqcup b| = |a \rightsquigarrow b| = 1$.

$[R_{\neg(3)}]$ $\dfrac{A^0 \leftrightarrow B^0}{(A \rightarrow B) \rightarrow (\neg A \rightarrow \neg B)}$

Como

$$a \rightsquigarrow b = \begin{cases} 1 & \text{si } a \leq b \\ b & \text{en otro caso} \end{cases}$$

$$-b \rightsquigarrow -a = \begin{cases} 1 & \text{si } a \leq b \\ -a & \text{en otro caso,} \end{cases}$$

es suficiente chequear el segundo caso. Pero por la hipótesis $A^0 \leftrightarrow B^0$ gracias a la 6.2 se obtiene que $|a| = |b|$, entonces, como estamos en el otro caso, $-a = b$, lo que concluye el resultado ∎

Observación 6.6 *En el artículo de Weigert se ocupa una implicación débil $\Rightarrow$, definida por $A \Rightarrow B = \neg A \vee B$; en contraste, nosotros hemos completado la lógica con una implicación intuicionista (positiva) la cual tiene mejores propiedades en cuanto al manejo axiomático que podemos hacer de la misma, y así proporciona una mayor amplitud a nuestro sistema. Es pertinente recalcar que nos referimos a una implicación positiva, puesto que en nuestro sistema la negación no es intuicionista; recalquemos esta afirmación mostrando que el axioma faltante dentro de los propuestos para completar una de las más típicas axiomatizaciones de la lógica intuicionista no se cumple en nuestro sistema.*

Proposición 6.8 $\nvdash \neg A \rightarrow (A \rightarrow B)$.

Demostración

Por el teorema 6.3 basta con ver el caso $a = 0 > b$

$$0 \rightsquigarrow (0 \rightsquigarrow b) = 0 \rightsquigarrow b = b \quad \blacksquare$$

Proposición 6.9 $\nvdash A \vee \neg A$.

Demostración

De nuevo por el teorema 6.3 basta con ver el caso que $|a| = 0$ $\blacksquare$

6.4. Algebrización para $OP_\daleth$

Como lo habíamos indicado desde la introducción, nos interesa cuidar del buen comportamiento de nuestro sistema desde el punto de vista algebraico. Es así que en esta sección se establece la algebrización de la lógica $OP_\daleth$.

Para el sistema lógico $OP_\daleth$ escogemos $A\Delta B$ formado por la unión de los conjuntos $\{\Delta^0(A,B) = A^0 \leftrightarrow B^0\}$ y $\{\Delta_x(A,B) = f_xA \leftrightarrow f_xB : x \in [-1,1]\}$. Así tenemos que el item [4] del teorema de caracterización de algebrizabilidad de Blok-Pigozzi, teorema 2.4, se transforman en:

[4][a] $A\Delta B \vdash \Delta^0(A,B)$.

[4][b] $A\Delta B \vdash \Delta_x(A,B)$.

[4][c] $A\Delta B, C\Delta D \vdash A * C\Delta B * D$, donde $*$ representa a $\wedge$ ó $\vee$ ó $\rightarrow$.

Además consideremos: $\delta(A) = A \wedge A$ y $\epsilon(A) = A \rightarrow A$.

Observación 6.7 *Obsérvese que el conjunto* $\Delta = \{\Delta^0\} \bigcup \{\Delta_x : x \in [-1,1]\}$ *no es finito.*

Teorema 6.4 *La lógica $OP_\daleth$ es algebrizable. Es decir,* Δ *y* $\delta \approx \epsilon$ *satisfacen las condiciones [1] al [5] del teorema 2.4.*

Demostración

La prueba de los items [1],[2] y [3] son inmediatas dado que $\Delta_1(A,B)$ es equivalente con $A \leftrightarrow B$. Demostraremos [4] y [5]. Veamos [4];

Caso [4][a] $A\Delta B \vdash A^0\Delta B^0$.

1. $A\Delta B \vdash \Delta^0(A^0, B^0)$.

   Como $(A^0)^0 \leftrightarrow A^0, \quad B^0 \leftrightarrow (B^0)^0$ por $\daleth_{12}$, es casi inmediato obtener el resultado de la hipótesis $A\Delta^0 B$.

2. $A\Delta B \vdash \Delta_x(A^0, B^0)$.
   Caso $x = 0$ es trivial.

   Caso $x > 0$ **:** Por definición de A^0 tenemos que $f_x A^0 = f_x(A \vee \neg A)$. Así por $\daleth_4$ se tiene que $f_x A^0 \leftrightarrow (f_x A \vee f_x \neg A)$. De otro lado por hipótesis Δ_x tenemos que $f_x A \leftrightarrow f_x B$, y por $\vee_1$ se obtiene que $f_x B \rightarrow f_x B \vee f_x \neg B$ y por $\daleth_4$ se tiene que $f_x B \vee f_x \neg B \leftrightarrow f_x(B \vee \neg B)$, con lo que podemos concluir que $[1] : f_x A^0 \rightarrow f_x B^0$.
   Ahora $f_x \neg A = f_x f_{-1} A$. Así por $\daleth_3$ se tiene que $f_x \neg A \leftrightarrow f_{-x} A$, con lo cual usando la hipótesis Δ_{-x} tenemos que $f_x \neg A \leftrightarrow f_{-x} B$. Por $\vee_2$ se obtiene que $f_{-x} B \rightarrow f_x B \vee f_{-x} B$ y por $\daleth_3$ se tiene que $f_x \neg A \rightarrow f_x B \vee f_{-x} B$, con lo que a partir de la definición de $\neg$ y $\daleth_4$ podemos concluir que $[2] : f_x \neg A \rightarrow f_x B^0$.
   De [1] y [2] usando $\vee_3$ concluimos que $(f_x A \vee f_x \neg A) \rightarrow f_x B^0$. De donde por $\daleth_4$ se obtiene que $f_x A^0 \rightarrow f_x B^0$; y en forma simétrica se tiene que $f_x B^0 \rightarrow f_x A^0$, con lo cual tenemos $\Delta_x(A^0, B^0)$.

   Caso $x = -1$ **:** Por definición de A^0 tenemos que $f_{-1} A^0 = f_{-1}(A \vee \neg A)$. Así por $\daleth_5$ se tiene que $f_{-1} A^0 \leftrightarrow (f_{-1} A \wedge f_{-1} f_{-1} A)$; y por $\daleth_3$ se obtiene que $f_{-1} A^0 \leftrightarrow (f_{-1} A \wedge A)$, o equivalentemente, $f_{-1} A^0 \leftrightarrow (\neg A \wedge A)$. Con lo cual usando las hipótesis y algunas tautologías de la lógica intuicionista positiva tenemos que $f_{-1} A^0 \leftrightarrow (B \wedge \neg B)$, con lo cual por definición de B^0 tenemos que $f_{-1} A^0 \leftrightarrow f_{-1} B^0$.

Caso [4][b] $\quad A\Delta B \vdash f_x A \Delta f_x B$

1. $A\Delta B \vdash \Delta^0(f_x A, f_x B)$.
   Como por definición de $(f_x A)^0 = f_x A \vee \neg f_x A$, el resultado se tiene con una prueba similar a la del caso $x > 0$ del item $[4][a](2)$.

2. $f_y f_x A \leftrightarrow f_y f_x B$.
   Por $\daleth_3$ tenemos que $f_y f_x A \leftrightarrow f_{y \otimes x} A$. Luego usando la hipótesis $\Delta_{y \otimes x}$ se obtiene que $f_y f_x A \leftrightarrow f_{y \otimes x} B$, de donde aplicando de nuevo $\daleth_3$ se tiene que $f_y f_x A \leftrightarrow f_y f_x B$.

Caso [4][c] $\quad A\Delta B, C\Delta D \vdash A * C \Delta B * D$, donde $*$ representa a $\wedge$ ó $\vee$ ó $\rightarrow$.

Subcaso $[* = \wedge]$ **:**

1. $A\Delta B, C\Delta D \vdash \Delta^0(A \wedge C, B \wedge D)$
Por definición tenemos que $(A \wedge C)^0 = (A \wedge C) \vee \neg(A \wedge C)$. Así por $\daleth_7$ se tiene que $(A \wedge C)^0 \leftrightarrow (A \wedge C) \vee (\neg A \vee \neg C)$. De otro lado por $\vee_1, \vee_2$ se tiene que $A \wedge C \vdash A, C$. Con lo cual usando la hipótesis Δ_1 se concluye que $A \wedge C \vdash B \wedge D$. En forma simétrica $B \wedge D \vdash A \wedge C$, por tanto se obtiene que $\vdash A \wedge C \leftrightarrow B \wedge D$. Una prueba análoga usando la hipótesis Δ_{-1} proporciona que $\vdash \neg A \wedge \neg C \leftrightarrow \neg B \wedge \neg D$. De donde usando argumentos similares a los del $[a](2)$ se obtiene que $((A \wedge C) \vee (\neg A \vee \neg C)) \leftrightarrow ((B \wedge D) \vee (\neg B \vee \neg D))$, es decir, $(A \wedge C)^0 \leftrightarrow (B \wedge D)^0$.

2. $A\Delta B, C\Delta D \vdash \Delta_x(A \wedge C, B \wedge D)$

   Caso $x = 0$ es trivial.

   Caso $x > 0$: Por $\daleth_6$ se tiene que $f_x(A \wedge C) \leftrightarrow (f_x A \wedge f_x C)$. De donde por analogía al item anterior se concluye que $f_x(A \wedge C) \leftrightarrow f_x(B \wedge D)$.

   Caso $x = -1$: Por $\daleth_7$ se tiene que $f_{-1}(A \wedge C) \leftrightarrow (f_{-1}A \vee f_{-1}C)$. De donde por analogía al item $[a](2)$ se concluye que $f_{-1}(A \wedge C) \leftrightarrow f_{-1}(B \wedge D)$.

Subcaso $[* = \vee]$: Análogo al subcaso $* = \wedge$.

Subcaso $[* = \rightarrow]$:

1. $A\Delta B, C\Delta D \vdash \Delta^0(A \rightarrow C, B \rightarrow D)$
Por definición tenemos la igualdad $(A \rightarrow C)^0 = (A \rightarrow C) \vee \neg(A \rightarrow C)$; y que $(B \rightarrow D)^0 = (B \rightarrow D) \vee \neg(B \rightarrow D)$. Usando las hipótesis no es difícil ver que $(A \rightarrow C) \leftrightarrow (B \rightarrow D)$, con lo cual es casi inmediato concluir que $(A \rightarrow C)^0 \leftrightarrow (B \rightarrow D)^0$.

2. $A\Delta B, C\Delta D \vdash \Delta_x(A \rightarrow C, B \rightarrow D)$

   Caso $x = 0$ es trivial.

   Caso $x > 0$: De $\daleth_8$ tenemos $f_x(A \rightarrow C) \leftrightarrow ((f_x A \rightarrow f_x C)) \rightarrow f_x\top)$. De donde como por hipótesis $f_x A \leftrightarrow f_x B$, y $f_x C \leftrightarrow f_x D$ se concluye que $(f_x A \rightarrow f_x C) \leftrightarrow (f_x B \rightarrow f_x D)$, con lo cual la conclusión del resultado se deriva de tautologías positivas.

   Caso $x = -1$: A partir de nuestras hipótesis se puede demostrar fácilmente que $(A \rightarrow C) \rightarrow (B \rightarrow D)$, y por el item 1 de este subcaso $[* = \rightarrow]$ tenemos que

$(A \to C)^0 \leftrightarrow (B \to D)^0$. Luego por la regla $R_{\neg(3)}$ se obtiene que $f_{-1}(A \to C) \to f_{-1}(B \to D)$. Usando la forma simétrica se obtiene que $\Delta_{-1}(A \to C, B \to D)$.

Ahora para concluir veamos [5] : $A \dashv\vdash \delta(A)\Delta\epsilon(A)$;

Primero: $A \vdash \delta(A)\Delta\epsilon(A)$

1. $A \vdash \delta(A)\Delta^0\epsilon(A)$

   Tenemos $\delta(A) = A \wedge A$ y $\epsilon(A) = A \to A$, y además ocupando $\to_1$ y $\to_2$ concluimos que $\vdash A \to A$. Con lo cual se deriva que $\vdash (A \wedge A)^0 \to (A \to A)$. De donde con la utilización de $\vee_2$ no es difícil concluir el siguiente resultado $(A \to A) \to ((A \to A) \vee \neg(A \to A))$. De donde por definición de $(A \to A)^0$ y el uso de tautologías clásicas positivas se obtiene que $\vdash (A \wedge A)^0 \to (A \to A)^0$.

   De otra parte como $A \vdash (A \wedge A)$ entonces nuevamente usando $\vee_2$ se obtiene que $A \vdash (A \wedge A) \to \neg(A \wedge A)$. De nuevo por definición de $(A \wedge A)^0$ y tautologías clásicas positivas se concluye que $\vdash (A \to A)^0 \to (A \wedge A)^0$.

   Con lo que se concluye el resultado.

2. $A \vdash \delta(A)\Delta_x(\delta(A), \epsilon(A))$

   Caso $x = 0$: trivial.

   Caso $x > 0$: $A \vdash f_x(A \wedge A) \leftrightarrow f_x(A \to A)$
   Como $f_x(A \wedge A) \leftrightarrow (f_xA \wedge f_xA)$ entonces por $\wedge_1$ es inmediato demostrar que $f_x(A \wedge A) \to f_xA$. Por la hipótesis se demuestra que $f_xA \to f_x\top$. Por lo cual $f_x(A \wedge A) \to f_xA$. De donde teniendo que $f_xA \to f_xA$ es un teorema del sistema, se obtiene que $f_x(A \wedge A) \to ((f_xA \to f_xA) \to f_x\top$; y así $f_x(A \wedge A) \to f_x(A \to A)$.
   Del otro lado Por la hipótesis se demuestra que $f_x\top \to f_x(A \wedge A)$, y como se tiene que $f_xA \to f_xA$ es un teorema, entonces se concluye el siguiente resultado $((f_xA \to f_xA) \to f_x\top) \to f_x(A \wedge A)$, lo que equivale por $\beth_9$ a que, $f_x(A \to A) \to f_x(A \wedge A)$.

   Caso $x = -1$: $A \vdash f_{-1}(A \wedge A) \leftrightarrow f_{-1}(A \to A)$
   Por item 2 tenemos que $(A \wedge A) \leftrightarrow (A \to A)$; y por el item 1 se obtiene que $(A \wedge A)^0 \leftrightarrow (A \to A)^0$. De donde por analogía al caso $x = -1$ del item 1, usando la regla $R_{\neg(3)}$ se obtiene que $f_{-1}(A \wedge A) \leftrightarrow f_{-1}(A \to A)$.

Segundo: $\delta(A)\Delta\epsilon(A) \vdash A$

De éste lado tenemos, $\delta(A)\Delta\epsilon(A) \vdash (A \rightarrow A) \rightarrow (A \wedge A)$, pero en nuestro sistema $\vdash A \rightarrow A$ y $(A \wedge A) \vdash A$ por $\wedge_1$, y por tanto $\delta(A)\Delta\epsilon(A) \vdash A$ ■

Capítulo 7

Las Lógicas Anotadas $OP_{\mathcal{BL}}$

Es conocido que las bases de datos grandes pueden ser inconsistentes por razones varias. Sin embargo, muchas de estas posibles inconsistencia podrían ser no relevantes dentro de los conocimientos que se deben recuperar del interior de la base de datos. Como es bien sabido las teorías clásicas inconsistentes no tienen modelos, por tanto las lógicas clásicas no son formalismos apropiados para hacer razonamientos con bases de datos inconsistentes. En la búsqueda de solucionar esta dificultad, Subrahmanian [70] introdujo las lógicas anotadas, y fueron posteriormente estudiadas en [15], donde es probado que son sistemas paraconsistentes y que pueden servir de una base de los lenguajes de programación para razonamiento con bases de datos que tienen inconsistencias. Los aspectos fundamentales desde el punto de vista de la teoría de modelos y pruebas formales se desarrollaron en [40] y [1]. Sin embargo, existen varios tipos de axiomas (algunos para fórmulas complejas, otros para fórmulas atómicas y otros para fórmulas arbitrarias), estos sistemas no son estructurales en el sentido que sus operadores de consecuencia no son cerrados bajo sustituciones. La principal dificultad de los sistemas no estructurales es que no se les puede encontrar una contraparte algebraica por los métodos estándar.

Es importante señalar que nuestro mayor interés está en la algebrizabilidad de las lógicas anotadas. Las primeras relaciones entre lógicas anotadas y algebrizabilidad (al estilo Blok-Pigozzi [16]) se presentaron en [56] y [57], donde aparecen versiones estructurales $SP_{\mathcal{T}}$ y SAL respectivamente de las lógicas anotadas. Los sistemas $SP_{\mathcal{T}}$ son algebrizables si y sólo si τ es finito.

Nosotros ahora presentaremos los sistemas $OP_{\mathcal{BL}}$ y $COP_{\mathcal{BL}}$. En este caso las anotaciones pertenecen a un birretículo entrelazado ([6], [43], [46]). Estos sistemas siguen siendo muy cercanos a P_τ, $SP_{\mathcal{T}}$ y SAL, pero con importantes diferencias en el concepto de "buen comportamiento de las fórmulas", y en la presentación novedosa de anotaciones de anotaciones. Además, ciertos axiomas son simplificados. Los sistemas $OP_{\mathcal{BL}}$ son algebrizables cuando $\mathcal{BL}$ es finito. Por lo tanto los sistemas $COP_{\mathcal{BL}}$ también son algebrizables si $\mathcal{BL}$ es finito. Algunos de los resultados obtenidos para las lógicas anotadas OP_{BL} dieron origen al artículo ([64]).

7.1. Birretículos

Aquí innovaremos un poco dentro de las lógicas anotadas tomando las anotaciones en un álgebra $\mathbf{A} = \langle A, \wedge, \vee, \sim, \otimes, \oplus \rangle$ de tipo $(2, 2, 1, 2, 2)$. Para tener un manejo más simple de las anotaciones, escogemos, dentro de este tipo de álgebras, las llamadas birretículos entrelazados, ya que estas han sido ampliamente estudiadas y usadas en aplicaciones teóricas de programación lógica ([46], [42], [6] y [7]).

Incluiremos los desarrollos básicos de birretículos sólo para efectos de tener un texto autocontenido, ya que la mayor parte de lo que escribiremos aparece en la literatura antes citada. Los birretículos son álgebras con dos estructuras de retículo por separado. Han sido usadas como base para semánticas denotacionales en sistemas de inferencia en inteligencia artificial, y en bases de conocimiento con programación lógica ([46] y [42]).

Definición 7.1 *Un birretículo es una estructura $\mathcal{BL} = \langle B, \leq_1, \leq_2, \sim \rangle$ tal que B es un conjunto no vacío que contiene por lo menos dos elementos diferentes; $(B, \leq_1)$, $(B, \leq_2)$ son retículos completos*[1] *y $\sim$ es una operación unaria*[2] *sobre B tal que:*

1. *Si $a \leq_1 b$, entonces $\sim a \geq_1 \sim b$,*
2. *Si $a \leq_2 b$, entonces $\sim a \leq_2 \sim b$,*
3. $\sim\sim a = a$.

Como es usual, nosotros usamos $\wedge$ y $\vee$ para las operaciones del retículo que corresponden a $\leq_1$, mientras que $\otimes, \oplus$ para las operaciones que corresponden a $\leq_2$. Mientras $\wedge$ y $\vee$ pueden ser asociadas con las intuiciones usuales de "mínimo" y "máximo", $\otimes$ y $\oplus$ se pueden interpretar como operadores de "consenso" y de "credibilidad" ("aceptación") respectivamente; $a \otimes b$ es lo más que a y b pueden coincidir, mientras que $a \oplus b$ acepta el conocimiento combinado de a y b [3]. Así, un birretículo puede efectivamente ser visto como $\langle B, \wedge, \vee, \sim, \otimes, \oplus \rangle$ un álgebra del tipo $(2, 2, 1, 2, 2)$. *Las constantes de anotación* son los elementos de B que en principio pueden pensarse como factores de confianza, o como grados de credibilidad o de verdad. Usaremos la notación f y v para el mínimo y el máximo de B con respecto

[1] Esta condición aparece en la definición original de Ginsberg [46]. Sin embargo otros autores han eliminado tal condición de completitud.

[2] Fitting elimina la operación unaria. En tal caso nosotros diremos *birretículo sin negación*.

[3] Esta idea aparece originalmente en Fitting [43].

a $\leq_1$, y respectivamente $\perp$ y $\top$ para notar el mínimo y el máximo de B con respecto a $\leq_2$.

Notemos que la negación respeta el orden $\leq_2$, dando cuenta de la intuición que $\leq_2$ corresponde a diferencias en nuestro conocimiento acerca de las fórmulas y no a sus grados de verdad. Mientras que la negación invierte $\leq_1$, es decir, la negación invierte la noción de verdad, en contraste la negación con respecto a $\leq_2$ dice que no conocemos más ni conocemos menos de $\neg p$ que lo que conocemos de p.

Definición 7.2 *Un birretículo se dice distributivo [46] si todas las doce posibles leyes distributivas concernientes a $\wedge, \vee, \otimes$ y $\oplus$ se verifican. Es llamado entrelazado [42] si cada uno de $\wedge, \vee, \otimes$ y $\oplus$ es monótono con respecto a $\leq_1$ y $\leq_2$.*

Proposición 7.1 *[42] Todo birretículo distributivo es entrelazado.*

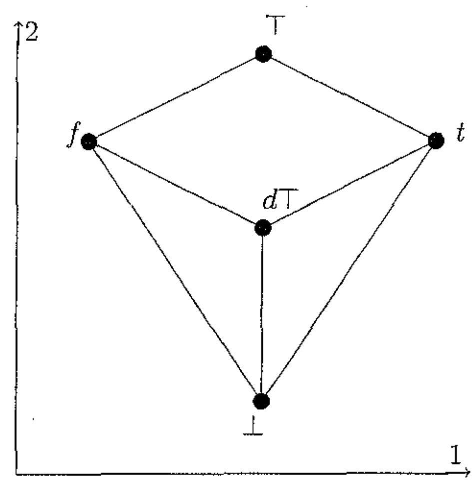
Proposición 7.2 *Sean $\mathcal{BL} = (B, \leq_1, \leq_2, \sim)$ un birretículo, y $a, b \in B$. Entonces*

1. $\sim (a \wedge b) = \sim a \vee \sim b \qquad \sim (a \vee b) = \sim a \wedge \sim b$
   $\sim (a \otimes b) = \sim a \otimes \sim b \qquad \sim (a \oplus b) = \sim a \oplus \sim b,$
2. $\sim f = t \qquad \sim t = f \qquad \sim \perp = \perp \qquad \sim \top = \top.$

Proposición 7.3 *Si $\mathcal{BL} = (B, \leq_1, \leq_2, \sim)$ es un birretículo entrelazado, y $a, b \in B$. Entonces $\perp \wedge \top = f$; $\perp \vee \top = t$; $f \otimes t = \perp$; $f \oplus t = \top$.*

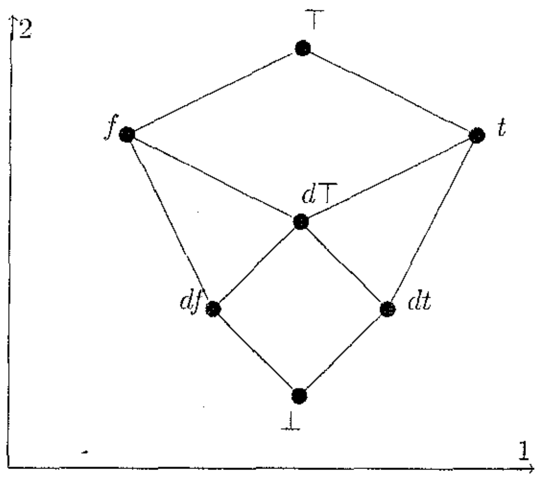
Ejemplos *Five* y *Default*[4]

Figura 1
Birretículo *Five*.



[4] *Five* es considerada en [46] y *Default* es considerada en [6].

Figura 2
Birretículo *Default.*



Ahora presentamos una sencilla e importante técnica de construcción de birretículos, la cual se debe a Ginsberg.

Definición 7.3 *Supongamos que* $(C, \leq)$ *y* $(D, \leq)$ *son retículos. Damos dos ordenes sobre* $C \times D$, $\leq_2$ *y* $\leq_1$, *como sigue;*
$(c_1, d_1) \leq_2 (c_2, d_2)$ *Si* $c_1 \leq c_2$ *y* $d_1 \leq d_2$
$(c_1, d_1) \leq_1 (c_2, d_2)$ *si* $c_1 \leq c_2$ *y* $d_2 \leq d_1$. *Entonces se define*
$BL(C, D) = (C \times D, \leq_2, \leq_1)$.

Proposición 7.4 *Si* $(C, \leq)$ *y* $(D, \leq)$ *son retículos completos, entonces* $BL(C, D)$ *es un birretículo entrelazado sin negación.*

Proposición 7.5 *Dado que* $(C, \leq)$ *sea un retículo completo, entonces* $BL(C, C)$ *es un birretículo entrelazado.*

Un valor de verdad $(x, y) \in BL(C, C)$ puede entenderse intuitivamente como que x representa el grado de credibilidad a favor, e y el grado de credibilidad en contra. Usando esta intuición en las lógicas anotadas originales se han presentado importantes aplicaciones a robótica ([38]).

Ejemplos: Ahora veamos algunos resultados en *Four* [5] y *Nine,* las pruebas aparecen en [6]. *Four* aparece en la figura tres, es el birretículo más pequeño no-degenerado.

[5] *Four* fue introducida por Belnap [11].

Four y *Nine*, aparecen en la figura cuatro, son distributivos, mientras que *Default* no lo es. Además, si denotamos el retículo de dos elementos $\{0, 1\}$ por *Two*, y el de tres elementos $\{0, \frac{1}{2}, 1\}$ por *Three*, entonces *Four* es isomorfo a $BL(Two, Two)$ y *Nine* es isomorfo a $BL(Three, Three)$.

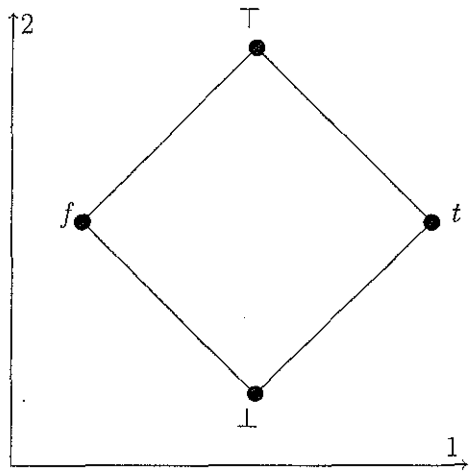


Figura 3: Birretículo *Four*.

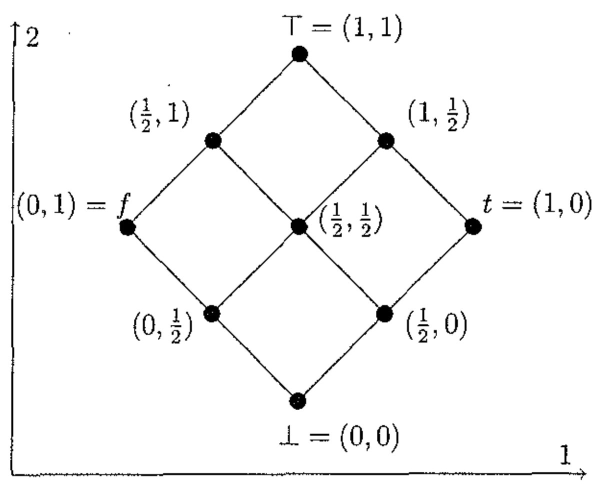


Figura 4: Birretículo *Nine*.

Default fue introducida por Ginsberg ([46]) como una herramienta para el razonamiento no monótono. Los valores de verdad que tienen el prefijo "d" en sus nombres representan valores por asunciones de default, por ejemplo dt es el verdadero por default. Las negaciones de $\top, t, f, \top$ son idénticas a las de *Four*, y $\sim df = dt$, $\sim dt = df$. $\sim d\top = d\top$. Así *Default* no es entrelazado, por ejemplo $f <_t df$, mientras que $f \otimes d\top = d\top$, $df \otimes d\top = df$, pero $d\top \not<_t df$.

7.1.1. Bifiltros

En una próxima sección buscaremos una semántica matricial para nuestros sistemas lógicos, es así que se hace necesario establecer los valores designados para dichas matrices. Es usual encontrar que en las presentaciones algebraicas, el conjunto de valores designados forma un filtro relativo al orden natural de los valores de verdad. Entonces, los análogos naturales a los filtros, para el caso de las birretículos el conjunto de los valores designados corresponda a los bifiltros.

Definición 7.4 *Un bifiltro de un birretículo $\mathcal{BL} = (B, \leq_1, \leq_2, \sim)$ es un subconjunto no vacío $F \subset B$, $F \neq B$, tal que $a \wedge b \in F$ si y sólo si $a \in F$ y $b \in F$; y $a \otimes b \in F$ si y sólo si $a \in F$ y $b \in F$.*

Ejemplo 7.5 *Los birretículos Four, Five y Default tienen un sólo bifiltro: $\{\top, t\}$; mientras que Nine tiene dos: $\{\top, (1, \frac{1}{2}), t\}$, y $\{\top, (\frac{1}{2}, 1), (\frac{1}{2}, \frac{1}{2}), (1, \frac{1}{2}), (\frac{1}{2}, 0), t\}$.*

Proposición 7.6 *[6](Propiedades básicas de bifiltros)*
Sea F un bifiltro de un birretículo $\mathcal{BL} = (B, \leq_1, \leq_2, \sim)$, entonces :

1. *F es cerrado hacia arriba con respecto a $\leq_1$ y $\leq_2$.*
2. *$t, \top \in F$, mientras que $f, \perp \notin F$.*

7.2. Los Sistemas $OP_{\mathcal{BL}}$

Como ya lo habíamos anunciado, aquí las anotaciones están en un birretículo entrelazado. A partir de este momento, salvo manifestación expresa de otra cosa, la expresión "birretículo" se interpretará como "birretículo entrelazado". Estos sistemas están basados en APC (annotated predicate calculus) y los sistemas originales de Da Costa ([27] y [28]). Se define en ellos el concepto de "fórmula con buen comportamiento" al estilo Da Costa, lo que se contrapone a las presentaciones de $SP_{\mathcal{T}}$ y SAL que manejan símbolos adicionales, que implican la inclusion de un gran número de axiomas extras. Además hemos simplificado algunos axiomas, y en cierto sentido logrado una mejor manipulación de los sistemas lógicos en consideración. Como aporte fundamental se permitirán anotaciones de anotaciones. En los sistemas $SP_{\mathcal{T}}$

y SAL las anotaciones de anotaciones se consideran como "updating", es decir, cada nueva anotación elimina la anterior. Nosotros necesitaremos introducir un operador de anotación por cada constante de anotación, estos son homomorfismos con respecto a los conectivos binarios, pero no con respecto a la negación. En las interpretaciones requeriremos de una operación extra, en principio usaremos el "$\oplus$" del birretículo. Esta idea da cuenta de una más general que es interpretar las anotaciones en álgebras tan generales como se quiera.

7.2.1. El lenguaje de $OP_{\mathcal{BL}}$

Sea $\mathcal{BL}$ un birretículo (fijo), el lenguaje $\mathcal{L}$ de $OP_{\mathcal{BL}}$ está formado por los siguientes símbolos primitivos:

1. Un conjunto enumerable P de letras proposicionales.
2. Conectivos lógicos $\wedge, \vee, \rightarrow, \neg$,
3. Una función de anotación unaria g_b por cada $b \in \mathcal{BL}$
4. Símbolos auxiliares: paréntesis, coma.

El conjunto $\mathcal{F}$ de fórmulas es definido recursivamente por:

1. Si p es una letra proposicional, entonces p es una fórmula.
2. Si A es una fórmula, entonces $\neg$A es una fórmula.
3. Si A y B son fórmulas, entonces $A \wedge B, A \vee B$, y $A \rightarrow B$ son fórmulas.
4. Si A es una fórmula y b es una constante de anotación, entonces $g_b A$ es una fórmula.
5. Una expresión es una fórmula si y sólo si se obtiene de la aplicación de un número finito de las reglas anteriores.

Intuitivamente la fórmula $g_b A$ se interpreta como *el grado de evidencia* de A es al menos b con respecto a $\leq_2$ y $\leq_1$. También puede representar el grado de credibilidad ó de incertidumbre asociado a A por un juez de razonamiento.

Para cada $n \in \omega$, inductivamente se define $\neg^n$ por $\neg^0 A = A$ y $\neg^{n+1} A = \neg(\neg^n A)$, y las fórmulas de la forma $\neg^{k_n} g_{b_n} \neg^{k_{n-1}} g_{b_{n-1}} \ldots \neg^{k_1} g_{b_1} \neg^{k_0} p$, se denominan *hiperliterales*, donde $k_i \in \omega$ y $b_i \in B$ para $0 \leq i \leq n$. Una fórmula que no es un hiperliteral se denomina *una fórmula compleja*.

Antes de presentar el sistema formal, siguiendo el estilo de da Costa definimos "el buen comportamiento de fórmulas" y la abreviación usual para la equivalencia. La expresión A^0 abrevia la fórmula $\neg(g_{\top}A \wedge g_{\top}\neg A)$, intuitivamente el símbolo "bolita" intenta representar el comportamiento clásico de las fórmulas, es decir, si una fórmula es clásica (tiene un buen comportamiento), no puede tenerse la proposición que la afirma y la niega a la vez. En cuanto a la expresión $A \leftrightarrow B$ como es lo usual abrevia la fórmula $(A \to B) \wedge (B \to A)$.

7.2.2. Axiomas de $OP_{\mathcal{BL}}$

La presentación del sistema $OP_{\mathcal{BL}}$ está dada por los axiomas de la lógica intuicionista positiva, junto a los que listamos a continuación.

Axiomas para $\neg$

[$\neg_1$] $(A^0 \wedge A \wedge \neg A) \to B$

[$\neg_2$] $A^0 \to (A \vee \neg A)$

[$\neg_3$] $\neg\neg A \leftrightarrow A$

[$\neg_4$] $\neg(A^0) \to (\neg g_b A \leftrightarrow g_{\sim b}\neg A)$

[$\neg_5$] $A^0 \to (\neg g_b A \leftrightarrow g_b \neg A)$

[$\neg_6$] $(A \wedge B)^0 \wedge (A \vee B)^0 \wedge (A \to B)^0$

Es importante recalcar que en este sistema tendremos como única regla de inferencia a modus ponens. Además se puede observar $\neg_1$ y $\neg_2$ simplemente trasladan los respectivos axiomas de $P\mathcal{T}$, intuitivamente corresponde a reductio ad absurdum y tercero excluido para fórmulas clásicas. Mientras que $\neg_4$ corresponde a una nueva forma de negación, la cual está dada por nuestra particular elección de las anotaciones para estos sistemas $g_b a = b \oplus a$, así $\sim (b \oplus a) = \sim b \oplus \sim a$; y $\neg_5$ corresponde al comportamiento clásico.

Axiomas para las anotaciones

Para todos $b, c \in \mathcal{BL}$:

[$\mathcal{BL}_1$] Si $b \leq_2 c$ y $b \leq_1 c$ entonces $g_b A \to g_c A$.

[$\mathcal{BL}_2$] $g_b(A * B) \leftrightarrow g_b A * g_b B$, donde $*$ representa $\wedge$ ó $\vee$ ó $\to$.

[$\mathcal{BL}_3$] $g_b g_c A \leftrightarrow g_{b \oplus c} A$.

[$\mathcal{BL}_4$] $\neg(A^0) \to ((A \leftrightarrow B) \to (g_c A \leftrightarrow g_c B))$.

$[\mathcal{BL}_5\,]\;\; (g_bB \wedge g_cB) \rightarrow g_{b\vee c}B.$

Intuitivamente $\mathcal{BL}_2$ asegura que g_b es un homomorfismo con respecto a los operadores binarios, pero $\neg_4$ asegura que en general no es un homomorfismo con respecto a la negación. De otro lado $\mathcal{BL}_3$ refleja el hecho que la operación g_b se interpretará en $\mathcal{BL}$ por $g_ba = b \oplus a$.

Las nociones de prueba, $\vdash$ relación de consecuencia (entre otras) en los sistemas $OP_{\mathcal{BL}}$ se definen como es lo usual. De nuevo haciendo uso de la lógica intuicionista positiva, junto al hecho de ser modus ponens la única regla de inferencia se tienen los siguientes resultados en la manera estándar.

Proposición 7.7 $\Sigma, A \vdash B$ *si y sólo si* $\Sigma \vdash A \rightarrow B$.

Proposición 7.8 *Si* $\varphi\ (x_1, x_2, \ldots, x_n)$ *es una tautología clásica, entonces*

1. $\{A_1^0, A_2^0, \ldots, A_n^0\} \vdash \varphi(A_1, A_2, \ldots, A_n)$.
2. *Si* $\varphi\ (x_1, x_2, \ldots, x_n)$ *no contiene negaciones, entonces* $\vdash \varphi(A_1, A_2, \ldots, A_n)$.

Dentro de éstos resultados, resaltemos que hay ciertos teoremas que habitualmente aparecen en otras lógicas anotadas como axiomas, por ejemplo la ley de Peirce.

Proposición 7.9 *Para cada fórmulas* A, B *y* C *en* $\mathcal{F}$,

1. $\vdash B^0 \rightarrow ((A \rightarrow B) \rightarrow ((A \rightarrow \neg B) \rightarrow \neg A))$.
2. $\vdash A^0 \rightarrow (A \rightarrow (\neg A \rightarrow B))$.
3. $\vdash ((A \rightarrow B) \rightarrow A) \rightarrow A$.

Insistimos que en los sistemas $OP_{\mathcal{BL}}$ la ley de Peirce es probada, y por lo tanto lo son todos los teoremas del cálculo proposicional clásico positivo.

Proposición 7.10 *(Reductio ad absurdum)*
Si $\Gamma, A \vdash A^0 \wedge B^0 \wedge B \wedge \neg B$, *entonces* $\Gamma \vdash \neg A$.

Demostración

De manera trivial $\Gamma, A \vdash B^0$, por tanto por la proposición 7.7 $\Gamma \vdash A \rightarrow B^0$, $\Gamma \vdash A \rightarrow B$ y $\Gamma \vdash A \rightarrow \neg B$. Así por la proposition 7.9 item uno y la proposición 7.7, $\Gamma \vdash A \rightarrow \neg A$. En el otro sentido $\Gamma \vdash \neg A \rightarrow \neg A$. Por tanto usando $A^0, \neg_2$ y $\vee_3$ se concluye que $\Gamma \vdash \neg A$ ∎

Proposición 7.11 $A^0, B^0, A \rightarrow \neg B \vdash B \rightarrow \neg A$

Demostración

Puesto que $A^0, B^0, A \to \neg B, B, A \vdash B^0, B, \neg B$ entonces por la proposición 7.7 $A^0, B^0, A \to \neg B, B \vdash \neg A$ y $A^0, B^0, A \to \neg B \vdash B \to \neg A$ ■

Teorema 7.6 $\vdash (A^0)^0$.

Demostración

Puesto que $A^0 = \neg((g_\top A \wedge g_\top \neg A)$, así $\neg A^0 = \neg\neg(g_\top A \wedge g_\top \neg A)$ y entonces $\neg A^0 \leftrightarrow (g_\top A \wedge g_\top \neg A)$. Luego $\neg(A^0)^0 \leftrightarrow (g_\top A^0 \wedge g_\top \neg A^0)$. Pero por $\neg_4$, si $\neg(A^0)^0$ entonces $g_\top \neg A^0 \leftrightarrow \neg g_\top A^0$ y por $\neg_6$ $(g_\top A^0)^0$. Luego $\neg(A^0)^0 \vdash g_\top A^0, \neg g_\top A^0, (g_\top A^0)^0$ y puesto que por $\neg_6$ $(\neg(A^0)^0)^0$, por reductio ad absurdum se concluye $\vdash \neg\neg(A^0)^0$, así por $\neg_3$ $\vdash (A^0)^0$ ■

Ahora veamos que para una fórmula con buen comportamiento, se cumple el resultado clásico que de ella y su negación se deduce cualquier otra fórmula.

Corolario 7.7 $A^0 \wedge \neg A^0 \vdash B$.

Demostración

Consecuencia del teorema 7.6 y $\neg_1$ ■

Ahora probaremos que "el buen comportamiento de las fórmulas" se preserva por negaciones y anotaciones.

Proposición 7.12 $\vdash A^0 \leftrightarrow (g_b A)^0$.

Demostración

[$\Rightarrow$] Por $\neg_3$ $\neg(g_b A)^0 \leftrightarrow g_\top(g_b A) \wedge g_\top(g_b(\neg A))$ así por $\neg_4$ y $\mathcal{BL}_3$ se concluye $\neg(g_b A)^0 \leftrightarrow g_\top A \wedge \neg g_\top A$. Así por $\neg_5$ $A^0 \vdash \neg(g_b A)^0 \to \neg A^0$. Además por el teorema 7.6 $(A^0)^0$ y por $\neg_6$ $(\neg(g_b A)^0)^0$. Entonces por la proposición 7.11 $\vdash A^0 \to \neg\neg(g_b A)^0$ así por $\neg_3$ $\vdash A^0 \to (g_b A)^0$.

[$\Leftarrow$] Por $\neg_3$ $\neg(g_b A)^0 \leftrightarrow g_\top(g_b A) \wedge g_\top(g_b(\neg A))$. Además por $\neg_5$ $g_\top(g_b A) \leftrightarrow \neg(g_\top g_b A)$ y por $\mathcal{BL}_3$ $g_\top(g_b A) \leftrightarrow \neg(g_\top A)$ así por $\neg_4$ $g_\top(g_b A) \leftrightarrow \neg(g_\top \neg A)$. Por lo tanto $\neg(g_b A)^0 \leftrightarrow \neg A^0$. Así $(g_b A)^0, \neg A^0 \vdash \neg(g_b A)^0$. Así por la proposición 7.7 $(g_b A)^0 \vdash \neg A^0 \to \neg(g_b A)^0$. Además por el teorema 7.6 $((g_b A)^0)^0$ y por $\neg_6$ $(\neg(g_b A)^0)^0$. Así por la proposición 7.11 $\vdash (g_b A)^0 \to \neg\neg A^0$ así por $\neg_3$ $\vdash (g_b A)^0 \to A^0$ ■

Proposición 7.13 $\vdash A^0 \leftrightarrow (\neg A)^0$.

Demostración

[$\Rightarrow$] Por $\neg_3$ $\neg(\neg A)^0 \leftrightarrow g_{\top}(\neg A) \wedge g_{\top}(\neg(\neg A))$. Puesto que $g_{\top}(\neg(\neg A)) \leftrightarrow g_{\sim\top}(\neg(\neg A))$ entonces por $\neg_4$ se concluye $\neg(\neg A)^0 \leftrightarrow g_{\top}(\neg A) \wedge \neg g_{\top}(\neg A)$. Así por $\neg_5$ y $\neg_3$ se concluye $\vdash \neg(\neg A^0) \rightarrow \neg A^0$ así por la proposición 7.11 y el teorema 7.6 $\vdash A^0 \rightarrow \neg\neg(\neg A)^0$ así por $\neg_3$ $\vdash A^0 \rightarrow (\neg A)^0$.

[$\Leftarrow$] Por $\neg_3$ $\neg A^0 \leftrightarrow g_{\top}A \wedge g_{\top}(\neg A)$. Por lo tanto $\mathcal{BL}_4$ y $\neg_3$ $g_{\top}(g_b A) \leftrightarrow \neg(g_{\top}g_b A)$ y por $\mathcal{BL}_3$ $\neg A^0 \leftrightarrow \neg(\neg A)^0$. Así $(\neg A)^0, \neg A^0 \vdash (\neg A)^0, \neg(\neg A)^0$. Entonces por el teorema 7.6 y la proposición 7.7 $\vdash (\neg A)^0 \rightarrow \neg\neg A^0$, así por $\neg_3$ tenemos $\vdash (\neg A)^0 \rightarrow A^0$ ■

Lema 7.8 *Todo hiperliteral es equivalente a* p, $\neg p$, $g_b p$, *ó* $g_b \neg p$, *para alguna letra proposicional* p.

Demostración

Puesto que los únicas operaciones que intervienen en la formación de hiperliterales son las de negación y anotación, entonces después de aplicar tales operaciones se obtienen fórmulas que son equivalentes a uno de las cuatro casos. Probemos el teorema por inducción sobre la complejidad de la fórmula. Sea A un hiperliteral. Si $A = g_b B$ para algún hiperliteral B, entonces cuando $B \leftrightarrow p$, ó $B \leftrightarrow \neg p$ el resultado es inmediato. Cuando $B \leftrightarrow g_b p$ o $B \leftrightarrow g_b \neg p$ por $\mathcal{BL}_3$ el resultado se obtiene.

Si $A = \neg B$ para algún hiperliteral B, entonces cuando $B \leftrightarrow p$, el resultado es inmediato. Si $B \leftrightarrow \neg p$, de $\neg_3$ nosotros conseguimos que A es equivalente a p. Cuando $B \leftrightarrow g_b p$, si $\neg(p)^0$ de $\neg_4$ se consigue que A es equivalente a $g_{\sim b}\neg p$, y de forma análoga si $(p)^0$ de $\neg_5$ se consigue que A es equivalente a $g_b \neg p$. Para cuando $B \leftrightarrow g_b \neg p$, si $\neg(\neg p)^0$ de $\neg_4$ y $\neg_3$ se consigue que A es equivalente a $g_{\sim b} p$, y en forma análoga si $(\neg(p))^0$ de $\neg_5$ y $\neg_3$ se consigue que A es equivalente a $g_b p$ ■

7.3. Semántica para $OP_{\mathcal{BL}}$

Continuando con la proximidad a P_τ se define una semántica bivalente para $OP_{\mathcal{BL}}$ sobre $\{0,1\}$. Como es usual, 0 y 1 representa falsedad y verdad en el metalenguaje.

Una *interpretación* I para el sistema $OP_{\mathcal{BL}}$ es una función $I : P \longrightarrow \mathcal{BL}$. Cada interpretación I define una única *valuación* $v_I : \mathcal{F} \longrightarrow \{0,1\}$, tal que:

$v_1 :$ $v_I(p) = 1$ si y sólo si $I(p) \geq_2 t$ y $I(p) \geq_1 \top$.

$v_2 :$ $v_I(\neg p) = 1$ si y sólo si $I(p) \geq_2 f$ y $I(p) \leq_1 \top$.

$v_3 :$ $v_I(g_b p) = 1$ si y sólo si $I(p) \oplus b \geq_2 t$ y $I(p) \oplus b \geq_1 \top$.

v_4 : $v_I\ (g_b\neg p) = 1$ si y sólo si $I(p) \oplus \sim b \geq_2 f$ y $I(p) \oplus \sim b \leq_1 \top$.

v_5 : $v_I\ (\neg g_b A) = v_I\ (g_{\sim b}\neg A)$, para todo hiperliteral A.

v_6 : $v_I\ (\neg A) = 1 - v_I(\ A\ )$, para toda fórmula compleja A.

v_7 : $v_I\ (g_b g_c A) = v_I\ (g_{b\oplus c} A)$, para todos $b, c \in \mathcal{BL}$.

v_8 : $v_I\ (g_b(A * B)) = v_I\ ((g_b A) * (g_b B))$, donde $*$ representa $\wedge$, $\vee$, ó $\rightarrow$.

v_9 : $v_I\ (g_b\neg C) = v_I(\neg g_b C\ )$, donde C es una fórmula compleja.

v_{10} : $v_I(A * B) = v_I(A) * v_I(B)$, donde $*$ en el lado izquierdo representa $\wedge$, $\vee$, ó $\rightarrow$, y $*$ en el otro la correspondiente operación del álgebra de Boole en *Two*.

Intuitivamente v_1 y v_2 dicen que los valores designados (ó verdaderos) corresponden al más pequeño bifiltro en la birretículo. Cuando se trata de un birretículo entrelazado estos corresponden a un intervalo $[t, \top]$, particularmente en *Nine* estos son $\{(1,0), (1, \frac{1}{2}), (1,1)\}$. Con los items v_4 y v_5 se enfatiza la diferencia entre la negación de una fórmula compleja y la de un hiperliteral. El caso de una fórmula compleja, la negación es clásica, mientras que la de los hiperliterales obedece a la elección particular para las anotacines que hemos fijado antes.

Las nociones de relación de consecuencia semántica $\models$ y fórmula válida se definen en la forma usual.

Observación 7.1

- *Si I es una interpretación tal que $I(p) = \top$, $v_I(p \wedge \neg p) = 1$. Así estos sistemas son paraconsistentes en un sentido fuerte, es decir, se pueden fijar ciertas interpretaciones que garantizan la existencia de contradicciones "verdaderas." Con esto, $(A \wedge \neg A) \rightarrow B$ no es un teorema de $OP_{\mathcal{BL}}$.*
- *La relevancia del axioma $\neg_4$ se enfatiza aquí, puesto que si $c \in \mathcal{BL}$ y I es una interpretación tal que $I(p) = c$ puede suceder que $v_I\ (g_b\neg p) = 1$ y $v_I(\neg g_b p) = 0$, por ejemplo, si $c \geq_2 b$ y $c \geq_1 b$.*

Proposición 7.14 *A es una fórmula compleja si y sólo si $v_I(A^0) = 1$.*

Demostración

Si A es un hiperliteral, por lema 7.8 hay sólo cuatro casos.

Caso 1: $A = p$. Por v_3, las propiedades de $\oplus$ y $\top$ en el birretículo se obtiene que $v_I(g_\top p) = 1$. Análogamente usando v_4 en lugar de v_3 se obtiene que $v_I(g_\top\neg p) = 1$.

así $v_I(g_\top p \wedge g_\top \neg p) = 1$ y por lo tanto $v_I(A^0) = 0$.

Caso 2: $A = \neg p$. Por $\neg_3$ y v_{10} se reduce al caso 1.

Caso 3: $A = g_b p$. Por v_5, $\neg_3$ y v_{10} se consigue que $v_I(g_\top \neg g_b p) = v_I(g_\top p)$ y por caso 1 se conoce $v_I(g_\top p) = 1$, que implica $v_I(A^0) = 0$.

Caso 4: $A = g_b \neg p$. Por v_3, v_{10}, $\mathcal{BL}_3$, las propiedades de $\oplus$ y el lugar de $\top$ en el birretículo se obtiene que $v_I(g_\top p) = 1$. Análogamente adicionando v_8 se obtiene que $v_I(g_\top \neg g_b \neg p) = 1$, así $v_I(A^0) = 0$. Ahora si A es una fórmula compleja, de v_6 se consigue la identidad $v_I(A^0) = 1 - v_I(g_\top A \wedge g_\top \neg A)$. Puesto que por v_6 $v_I(g_\top \neg A) = 1 - v_I\ (g_\top A)$ entonces $v_I(g_\top A \wedge g_\top \neg A) = 0$. Por lo tanto tenemos que $v_I(A^0) = 1$ ■

También tenemos los siguientes resultados que enmarcan el comportamiento clásico de las fórmulas complejas.

Proposición 7.15 *Si $\varphi\ (x_1, x_2, \ldots, x_n)$ es una tautología clásica y $A_1, A_2, \ldots, A_n$ son fórmulas complejas, entonces $\models \varphi(A_1, A_2, \ldots, A_n)$; además si $\varphi\ (x_1, x_2, \ldots, x_n)$ no contienen negaciones, entonces $A_1, A_2, \ldots, A_n \models \varphi(A_1, A_2, \ldots, A_n)$.*

Proposición 7.16 *Si una interpretación I para los sistemas $OP_{\mathcal{BL}}$ es tal que cumple la propiedad $v_I(A) = 1$, entonces $v_I(g_b A) = 1$.*

Teorema 7.9 *Si $\Sigma \vdash A$, entonces $\Sigma \models A$.*

Demostración
No necesitamos verificar los axiomas de $\wedge, \vee, \rightarrow$ y la regla MP, puesto que estos corresponden al caso clásico, que no contienen negaciones, y además las valuaciones se definieron para que cumplieran dichos requerimientos. Más aún $\neg_1$, $\neg_6$ y $\neg_2$ son consecuencia de la proposición 7.14. En cuanto a $\neg_3$ es estándar, mientras que $\neg_4$ se deduce de la proposición 7.14, y el item v_4. Además $\neg_5$ se deduce de la proposición 7.14 y item v_5.

Para $\mathcal{BL}_1$, por lema 7.8 es suficiente ver los cuatro casos correspondientes a los hiperliterales;
Caso 1: Si $A = p$ entonces $v_I(g_b p) = 1$ si y sólo si, $I(p) \oplus b \geq_2 t$ y $I(p) \oplus b \geq_1 \top$ y entonces $v_I(g_b p) = 1$.
Casos 2, 3, 4 : $A = \neg p$, $A = g_c p$ y $A = g_c \neg p$ se reduce al caso previo.
Si A es una fórmula compleja, $v_I(g_b A) = 1$ entonces $v_I(g_c A) = 1$ por inducción y los items v_7 y v_8.

$\mathcal{BL}_2$ se deduce de items v_8 y v_{10} y $\mathcal{BL}_3$ se prueba por v_7.

Para $\mathcal{BL}_4$, sea I una interpretación tal que $v_I(A) = a$ y $v_I(B) = b$. Si $a = b = 1$ entonces por v_1 se consigue $a \geq_2 t$ y $a \geq_1 \top$, $b \geq_2 t$ y $b \geq_1 \top$. Entonces $a \oplus c \geq_2 t$ y $a \oplus c \geq_1 \top$, $b \oplus c \geq_2 t$ y $b \oplus c \geq_1 \top$. Así de v_3 se concluye que $v_I(g_c A) = 1$.

Para $\mathcal{BL}_5$, sea I una interpretación tal que $v_I(A \to g_b B) = 1$, $v_I(A \to g_c B) = 1$ y $v_I(A \to f_{b \vee c} B) = 0$. Entonces $v_I(A) = 1$ y $v_I(g_{b\vee c} B) = 0$. El caso cuando B es un hiperliteral se reduce al caso $B = p$. Por lo tanto $I(p) \not\geq (b \oplus c)$, pero $v_I(g_b p) = 1$ y $v_I(g_c p) = 1$. Así $I(p) \geq b$ y $I(p) \geq c$, y entonces $I(p) \geq (b \oplus c)$, que es una contradicción. Cuando B es compleja, asumamos que $v_I(g_b A) = 1$, así por la proposición 7.16 $v_I(g_b B) = 1$, y por lo tanto $v_I(g_{b\vee c} B) = 1$ ∎

Estos sistemas no son completos, puesto que $\models_{OP_{\mathcal{BL}}}$ no es estructural, y así $\vdash_{OP_{\mathcal{BL}}}$ no lo es. Baste chequear que $\models g_\top p$, pero si hacemos $\sigma(p) = C \wedge \neg C$ donde C es una fórmula compleja, entonces $\not\models g_\top \sigma(p)$. Para lograr la completitud necesitaremos modificar el sistema $OP_{\mathcal{BL}}$ ligeramente, luego consideraremos una semántica matricial.

7.4. La lógica $COP_{\mathcal{BL}}$

La búsqueda de un sistema completo nos obliga a incluir una nueva regla de inferencia y dos nuevos axiomas. Con la regla queremos asegurar una fórmula aceptada (probada) tiene todos los niveles de credibilidad. El primero de los axiomas garantiza que todos los hiperliterales gozan de un cierto grado de evidencia máximo. El otro refleja el comportamiento de la negación con respecto a un nivel de credibilidad dado. Estos acercan nuestros sistemas a SAL [57]. Para simplificar convendremos a partir de este momento, salvo mención explícita de lo contrario, que el símbolo $\geq$ representa conjuntamente a $\geq_2$ y $\geq_1$.

Para toda fórmula A y $b, c \in \mathcal{BL}$, se define $[A]_b^c$ a continuación.

$$[A]_b^c = \begin{cases} A \wedge \neg A & \text{Si } c \geq b \text{ y } \sim c \geq b \\ A \wedge \neg(\neg A \wedge \neg A) & \text{Si } c \geq b \text{ y } \sim c \not\geq b \\ \neg A \wedge \neg(A \wedge A) & \text{Si } c \not\geq b \text{ y } \sim c \geq b \\ \neg(A \wedge A) \wedge \neg(\neg A \wedge \neg A) & \text{Si } c \not\geq b \text{ y } \sim c \not\geq b \end{cases}$$

Además, se define $d_b(A) = g_b A \wedge \bigwedge_{c \not\geq b} \neg(g_c A \wedge g_c A)$.

Es importante resaltar que este término es verdadero bajo una interpretación si y sólo si el grado de credibilidad de A es menor o igual a b. Obsérvese que $g_c A \wedge g_c A$, al ser una fórmula compleja, tiene un comportamiento clásico con respecto a las negaciones.

Las lógicas $COP_{\mathcal{BL}}$ están basadas en los sistemas $OP_{\mathcal{BL}}$, con la adición de una nueva regla y dos nuevos axiomas.

7.4.1. Nuevos axiomas para las anotaciones

$[\mathcal{BL}_6]$ $\neg(A^0) \to \bigvee_{b\in\mathcal{BL}} d_b(A)$

$[\mathcal{BL}_7]$ $d_b(A) \to \bigvee_{c\in\mathcal{BL}} [A]_b^c$

Intuitivamente $\bigvee_{b\in\mathcal{BL}} d_b(A)$ es verdadero si y sólo si el máximo grado de credibilidad de A es b. El axioma $\mathcal{BL}_6$ dice que todos los hiperliterales tienen un grado máximo de credibilidad.

Análogamente $\bigvee_{c\in\mathcal{BL}} [A]_b^c$ da la relación entre el grado de credibilidad b y el grado de credibilidad c y $\sim c$. Esta relación depende de la particularidad del birretículo empleado.

En cuanto al axioma $\mathcal{BL}_7$ dice que si A tiene un grado de credibilidad máximo b, entonces A y $\neg A$ se comportan con alguna credibilidad dada c.

7.4.2. Nueva regla de inferencia para $COP_{\mathcal{BL}}$

$[R_2]$ $\dfrac{A}{g_b A}$ para todo $b \in \mathcal{BL}$.

Observación 7.2 *Los sistemas $COP_{\mathcal{BL}}$ también son correctos con respecto a la semántica definida en la sección 4.*

7.5. Semántica matricial para $COP_{\mathcal{BL}}$

El objetivo fundamental de esta sección será dar una semántica matricial para los sistemas $COP_{\mathcal{BL}}$. Las definiciones básicas de matrices lógicas están en la sección 2.4.

7.5.1. Validez

Definimos una clase $\mathbf{M}$ de matrices y probaremos que las matrices de $\mathbf{M}$ son matrices modelos para $COP_{\mathcal{BL}}$. Sean $\mathcal{BL}$ un birretículo, F un bifiltro de $\mathcal{BL}$ y sean 0 y 1 objetos que no están en $\mathcal{BL}$. Se define $\widehat{\mathcal{BL}} = \mathcal{BL} \cup \{0,1\}$ y $\widehat{F} = F \cup \{1\}$. Se extienden los ordenes de $\mathcal{BL}$ a $\widehat{\mathcal{BL}}$ como sigue.

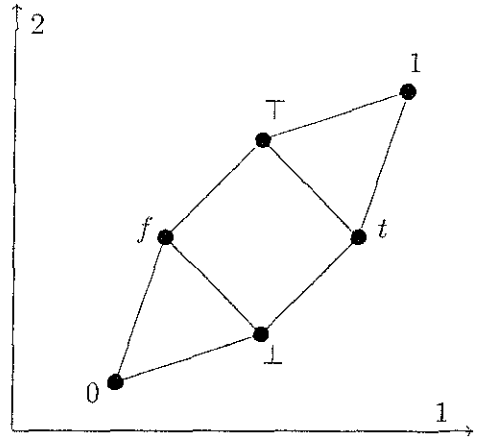


Figura 5: los ordenes de $\widehat{\mathcal{BL}}$

Además se definen las siguientes operaciones sobre $\widehat{\mathcal{BL}}$:

1. $a \vee b = \begin{cases} 1 & \text{si } a \in \widehat{F} \text{ ó } b \in \widehat{F} \\ 0 & \text{en otro caso.} \end{cases}$

2. $a \wedge b = \begin{cases} 1 & \text{si } a \in \widehat{F} \text{ y } b \in \widehat{F} \\ 0 & \text{en otro caso.} \end{cases}$

3. $a \to b = \begin{cases} 1 & \text{si } a \notin \widehat{F} \text{ ó } b \in \widehat{F} \\ 0 & \text{en otro caso.} \end{cases}$

4. Si $a \in \mathcal{BL}$ entonces $\neg a = \sim a$, $\neg 1 = 0$ y $\neg 0 = 1$.

5. $g_b a = \begin{cases} b \oplus a & \text{si } a \neq 0 \text{ y } a \neq 1 \\ a & \text{si } a = 0 \text{ ó } a = 1 \end{cases}$

6. $a^0 = \neg(g_\top a \wedge g_\top \neg a)$.

La clase $\mathbf{M}$ es el conjunto de las matrices $\mathcal{M} = \langle \widehat{\mathcal{BL}}, \widehat{F} \rangle$.

Proposición 7.17 *Si* $a \in \widehat{\mathcal{BL}}$ *entonces* $a^0 = \begin{cases} 0 & si\ a \in \mathcal{BL} \\ 1 & si\ a \in \{0,1\} \end{cases}$

Demostración

Por definición $a^0 = \neg(g_\top a \wedge g_\top \neg a)$, así $a^0 = 0$ si y sólo si $(g_\top a \wedge g_\top \neg a) = 1$ si y sólo si $\top \oplus a, \top \oplus \neg a \in F$ si y sólo si $a \in \mathcal{BL}$, puesto que si $a \in \{0,1\}$ entonces $(g_\top a \wedge g_\top \neg a) = 0$.
En el otro sentido $a^0 = 1$ si y sólo si $(g_\top a \wedge g_\top \neg a) = 0$ si y sólo si $g_\top a = 0$ ó $g_\top \neg a = 0$ si y sólo si $a = 0$ ó $\neg a = 1$ si y sólo si $a \in \{0,1\}$ ∎

Proposición 7.18 *Sea $\mathcal{M} = \langle \widehat{\mathcal{BL}}, \widehat{F} \rangle \in \mathbf{M}$ y $u : P \longrightarrow \widehat{\mathcal{BL}}$ tal que $u(p) \in \mathcal{BL}$ entonces para todo hiperliteral B con la letra proposicional p, $u^*(B^0) = 0$ si y sólo si $u^*(B) \in \mathcal{BL}$. Además, si $u(p) \in \{0,1\}$ entonces $u^*(p^0) = 1$.*

Teorema 7.10 *Si una matriz $\mathcal{M} = \langle \widehat{\mathcal{BL}}, \widehat{F} \rangle \in \mathbf{M}$ entonces $\mathcal{M}$ es una matriz modelo de $COP_{\mathcal{BL}}$.*

Demostración

Los axiomas $\rightarrow_1, \rightarrow_2, \wedge_1, \wedge_2, \wedge_3, \vee_1, \vee_2, \vee_3, \neg_3$, y MP son triviales, puesto que por definición tienen un comportamiento clásico, y las operaciones se definieron para que cumplieran estos requerimientos.

[$\neg_1$] $A^0 \wedge A \wedge \neg A \rightarrow B$

Si $a^0 \in \widehat{F}$ entonces $a \in \{0,1\}$ así $a^0 \wedge a \wedge \neg a = 0$. De donde el resultado se obtiene inmediatamente.

[$\neg_2$] $A^0 \rightarrow (A \vee \neg A)$

Si $a^0 \in \widehat{F}$ entonces $a \in \{0,1\}$. De donde el resultado se obtiene inmediatamente.

[$\neg_4$] $\neg(A^0) \rightarrow (\neg g_b A \leftrightarrow g_{\sim b} \neg A)$

Si $\sim a^0 \in \widehat{F}$ entonces $a^0 \in \mathcal{BL}$. Por lo tanto $\neg g_b A \leftrightarrow \sim (b \oplus a)$. Con lo cual tenemos que $g_{\sim b} \neg A \leftrightarrow \sim b \oplus \sim a$. Así por la proposición 7.2 se concluye el resultado.

[$\neg_5$] $A^0 \rightarrow (\neg g_b A \leftrightarrow g_b \neg A)$

Caso 1: Si $a = 0$ entonces $g_b a = 0$. Así $\neg g_b a = 1$ y $g_b \neg a = g_b 1$, así $g_b \neg a = 1$.

Caso 2: Si $a = 1$ entonces $g_b a = 1$. Así $\neg g_b a = 0$ y $g_b \neg a = g_b 0$, así $g_b \neg a = 0$.

[$\neg_6$] $(A \wedge B)^0 \wedge (A \vee B)^0 \wedge (A \rightarrow B)^0$

Consecuencia de la definición de la operación "bolita".

[$\mathcal{BL}_1$] $g_b A \rightarrow g_c A$ para todo $b, c \in \mathcal{BL}$ tal que $b \leq_2 c$ y $b \leq_1 c$.

Consecuencia de las propiedades básicas de los bifiltros.

[$\mathcal{BL}_2$] $g_b(A * B) \leftrightarrow g_b A * g_b B$, donde $*$ representa $\wedge$, ó $\vee$ ó $\rightarrow$.

Estos son inmediatos.

[$\mathcal{BL}_3$] $g_b g_c A \leftrightarrow g_{b \oplus c} A$ para todos $b, c \in \mathcal{BL}$.

Consecuencia de la asociatividad de $\oplus$ en la birretículo.

$[\mathcal{BL}_4]$ $\neg(A^0) \to ((A \leftrightarrow B) \to (g_b A \leftrightarrow g_b B))$, para todo $b \in \mathcal{BL}$.

Si $\neg a^0 \in \widehat{F}$ entonces $\neg a^0 = 1$. Así $a^0 = 0 \leftrightarrow a \in \mathcal{BL}$, y por tanto esta propiedad se reduce a $c \oplus a \in \widehat{F}$ si y sólo si $b \oplus a \in \widehat{F}$, esto se tiene puesto que F es un bifiltro.

$[\mathcal{BL}_5]$ $(g_b B \wedge g_c B) \to f_{b \vee c} B$ para todos $b, c \in \mathcal{BL}$.

Esto se tiene, puesto que F es un bifiltro.

$[\mathcal{BL}_6]$ $\neg(A^0) \to \bigvee_{b \in \mathcal{BL}} d_b(A)$

Sea el conjunto $S = \{b : b \in F \text{ y } b \oplus a \in F\}$. Entonces $S \neq \phi$ puesto que $\top \in S$. Si hacemos $b_0 = infS$, entonces $(d_{b_0}(A))^{\mathcal{M}} = b_0 \oplus a \wedge \bigwedge_{c \not\geq b_0} \neg(c \oplus a \wedge c \oplus a)$,

y por tanto $d_{b_0}(A)^{\mathcal{M}} = 1$, y así $(\neg(A^0) \to \bigvee_{b \in \mathcal{BL}} d_b(A))^{\mathcal{M}} = 1$.

$[\mathcal{BL}_7]$ $d_b(A) \to \bigvee_{c \in \mathcal{BL}} [A]_b^c$

Si $d_b(a) \in \widehat{F}$ entonces $a \in \mathcal{BL}$ y $a \notin F$. Puesto que $g_b a = b \oplus a$ entonces $b \geq b_0$. En el otro sentido $c \not\geq b$ implica que $\neg(g_c a \wedge g_c a) = 1$. Entonces $b_0 \geq b$ así $b = b_0$. Además puesto que $a \notin F$ entonces

$$[A]_{b_0}^a = \begin{cases} \neg A \wedge \neg(A \wedge A) & \text{si } \sim a \geq b_0 \\ \neg(A \wedge A) \wedge \neg(\neg A \wedge \neg A) & \text{si } \sim a \not\geq b_0 \end{cases}$$

Por lo tanto $([A]_{b_0}^a)^{\mathcal{M}} = 1 \in \widehat{F}$. Así se consigue que $\bigvee_{c \in \mathcal{BL}} [A]_b^c \in \widehat{F}$, y por tanto $(d_b(A) \to \bigvee_{c \in \mathcal{BL}} [A]_b^c)^{\mathcal{M}} = 1 \in \widehat{F}$.

Veamos que se cumple la regla R_2 : Si $a \in \widehat{F}$. Consideremos dos casos.

Caso 1: $a = 1$. En este caso $g_b a = 1$ por definición, y así trivialmente $g_b a \in \widehat{F}$. Luego R_2 es válida.

Caso 2: $a \in F$. Entonces $g_b a = b \oplus a \geq a$, así $g_b a \in F$ puesto que F es un bifiltro, y de nuevo R_2 es válida ■

7.5.2. Semántica Matricial

Definiremos ahora una matriz M y probamos que es una semántica matricial para $COP_{\mathcal{BL}}$. Sea $\mathcal{BL}$ un birretículo (fijo) tal que sus bifiltros son los conjuntos

$\{F_1, F_2, ..., F_k\}$ y sean 0 y 1 objetos que no están en $\mathcal{BL}$.
Se define $\mathsf{F} = (\bigcup_{1 \leq i \leq k} \{i\} \times F_i) \cup \{1\}$ y $\mathsf{B} = (\bigcup_{1 \leq i \leq k} \{i\} \times \mathcal{BL}) \cup \{0, 1\}$.

Se extiende el orden de $\mathcal{BL}$ a B como sigue;

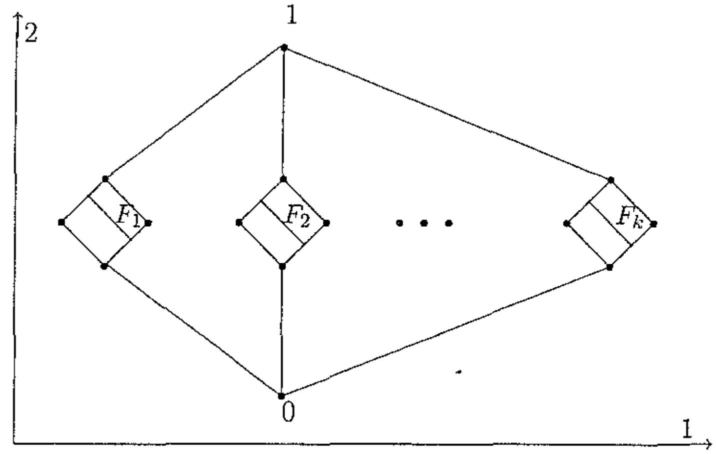


Figura 6: Los ordenes para la semántica matricial.

Además se define las siguientes operaciones sobre B :

1. $\mathsf{a} \vee \mathsf{b} = \begin{cases} 1 & \text{si } \mathsf{a} \in \mathsf{F} \text{ ó } \mathsf{b} \in \mathsf{F} \\ 0 & \text{en otro caso.} \end{cases}$

2. $\mathsf{a} \wedge \mathsf{b} = \begin{cases} 1 & \text{si } \mathsf{a} \in \mathsf{F} \quad y \quad \mathsf{b} \in \mathsf{F} \\ 0 & \text{en otro caso.} \end{cases}$

3. $\mathsf{a} \rightarrow \mathsf{b} = \begin{cases} 1 & \text{si } \mathsf{a} \notin \mathsf{F} \text{ ó } \mathsf{b} \in \mathsf{F} \\ 0 & \text{en otro caso.} \end{cases}$

4. Si $\mathsf{a} = (i, a)$ donde $a \in \mathcal{BL}$ entonces $\neg\mathsf{a} = (i, \sim a)$, $\neg 1 = 0$ y $\neg 0 = 1$.

5. $g_b\mathsf{a} = \begin{cases} (i, b \oplus a) & \text{si } \mathsf{a} = (i, a) \\ \mathsf{a} & \text{si } \mathsf{a} = 0 \text{ ó } \mathsf{a} = 1 \end{cases}$

6. $\mathsf{a}^0 = \neg(g_\top \mathsf{a} \wedge g_\top \neg\mathsf{a})$.

Si la matriz $M = \langle \mathsf{B}, \mathsf{F} \rangle$, se obtiene el siguiente resultado, cuya prueba es similar a la del teorema 7.10.

Teorema 7.11 *La matriz* M *es una matriz modelo de* $COP_{\mathcal{BL}}$.

Como es el caso usual todo conjunto no trivial de fórmulas se puede extender a un conjunto maximal no trivial consistente de fórmulas, así nos restringiremos a presentar la completitud para un conjunto maximal no trivial consistente de fórmulas.

Teorema 7.12 *Sea* Γ *un conjunto maximal consistente no trivial de fórmulas. Entonces existe una valuación* v*:* $P \longrightarrow B$ *tal que para toda fórmula* A, $v^*(A) \in \mathsf{F}$ *si y sólo si* $A \in \Gamma$.

Demostración

Para cada p letra proposicional en P hay sólo dos posibilidades $p^0 \notin \Gamma$ ó $p^0 \in \Gamma$. Cuando $p^0 \notin \Gamma$, hacemos las escogencias a_i el menor elemento de $\{b : g_b(p) \in \Gamma\}$, F_i el más pequeño bifiltro de $\mathcal{BL}$ que contiene a_i, y b_i el menor elemento del conjunto $\{b : [p]_b^{\mathsf{a}_i} \in \Gamma\}$.

Ahora definimós la valuación v por

$$v(\mathrm{p}) = \begin{cases} 1 & \text{si } p^0 \in \Gamma \text{ y } p \in \Gamma \\ 0 & \text{si } p^0 \in \Gamma \text{ y } \neg p \in \Gamma \\ (i, b_i) & \text{si } p^0 \notin \Gamma \end{cases}$$

Por el principio de inducción es suficiente probar el resultado para hiperliterales. Pero por lema 7.8 basta ver los siguientes cuatro casos:

Caso 1: $A = p$. entonces $v^*(p) = v(p)$. Cuando $p^0 \in \Gamma$ entonces $v^*(A) \in \mathsf{F}$ es equivalente a $v(p) \in \mathsf{F}$ que equivale a $v(p) = 1$, lo que es equivalente a $p \in \Gamma$.
Ahora cuando $p^0 \notin \Gamma$, entonces $v^*(A) \in \mathsf{F}$ es equivalente a $v(p) \in \mathsf{F}$ que equivale a $(i, b_i) \in \mathsf{F}$, que equivale a $b_i \in F_i$. Por lo tanto $\mathsf{a}_i \geqslant b_i$, y así

$$[p]_{b_i}^{\mathsf{a}_i} = \begin{cases} p \wedge \neg p & \text{si } \sim \mathsf{a}_i \geq b_i \\ p \wedge \neg(\neg p \wedge \neg p) & \text{si } \sim \mathsf{a}_i \not\geq b_i \end{cases}$$

Entonces $[p]_{b_i}^{\mathsf{a}_i} \rightarrow p$ y como $[p]_{b_i}^{\mathsf{a}_i} \in \Gamma$, por MP se obtiene que $p \in \Gamma$, y por lo tanto $v^*(A) \in \mathsf{F}$ implica que $A \in \Gamma$.
Para el recíproco $v^*(A) \notin \mathsf{F}$ es equivalente a $b_i \notin F_i$, por tanto $\mathsf{a}_i \not\geq b_i$ y así

$$[p]_{b_i}^{\mathsf{a}_i} = \begin{cases} \neg p \wedge \neg(p \wedge p) & \text{si } \sim \mathsf{a}_i \geq b_i \\ \neg(p \wedge p) \wedge \neg(\neg p \wedge \neg p) & \text{si } \sim \mathsf{a}_i \not\geq b_i \end{cases}$$

Entonces $[p]_{b_i}^{\mathsf{a}_i} \rightarrow \neg(p \wedge p)$ y como $[p]_{b_i}^{\mathsf{a}_i} \in \Gamma$, por MP se obtiene que $\neg(p \wedge p) \in \Gamma$, y por lo tanto $p \notin \Gamma$. Así $p \in \Gamma$ implica que $v^*(p) \in \mathsf{F}$, es decir, concluimos el resultado $A \in \Gamma$ implica que $v^*(p) \in \mathsf{F}$.

Caso 2: $A = \neg p$. Análogo al caso 1.

Caso 3: $A = g_c p$. Cuando $p^0 \in \Gamma$ entonces $v\ (p) = 0$ ó $v\ (p) = 1$, y por lo tanto $g_c v\ (p) = v\ (p)$. Luego, como $v\ ^*(A) \in \mathsf{F}$ si y sólo si $v\ ^*(g_c p) \in \mathsf{F}$ si y sólo si $g_c\ v\ (p) \in \mathsf{F}$ si y sólo si $v\ (p) = 1$, esto es equivalente a $p \in \Gamma$, que por R_2 es equivalente a que $g_c p \in \Gamma$.

Cuando $p^0 \notin \Gamma$ entonces $v\ (p) = (i, b_i)$, y por tanto $v\ ^*(g_c p) = (i, c \oplus b_i)$.

Así $v\ ^*(g_c p) \in \mathsf{F}$ si y sólo si $g_c\ v\ (p) \in \mathsf{F}$ si y sólo si $c \oplus\ v\ (p) \in F_i$ si y sólo si $c \oplus b_i \in F_i$ si y sólo si $g_c p \in \Gamma$, es decir, $v\ ^*(A) \in \mathsf{F}$ si y sólo si $A \in \Gamma$.

Caso 4: $A = g_c \neg p$ es análogo al caso 3 ∎

Ahora como consecuencia del teorema 7.12 y que todo conjunto no trivial de fórmulas está contenido en un conjunto maximal de fórmulas, tenemos el siguiente resultado.

Corolario 7.13 *Si Γ es un conjunto no trivial de fórmulas de $COP_{\mathcal{BL}}$ entonces $\Gamma \vdash \varphi$ si y sólo si $\Gamma \models_M \varphi$.*

7.6. Algebrización de $OP_{\mathcal{BL}}$

Usaremos nuevamente la caracterización de Blok-Pigozzi [16] ya acostumbrada.

Teorema 7.14 *Dado un birretículo finito $\mathcal{BL}$, $\delta(A) = A \wedge A \approx A \to A = \epsilon(A)$ es un conjunto de ecuaciones de definición para la lógica $OP_{\mathcal{BL}}$ y*

1. $\Delta_{\leftrightarrow}(A, B) = A \leftrightarrow B$
2. $\Delta_{\neg}(A, B) = \neg A \leftrightarrow \neg B$
3. $\Delta_b(A, B) = g_b A \leftrightarrow g_b B$

son un sistema de fórmulas de equivalencia para la lógica $OP_{\mathcal{BL}}$, así los sistemas $OP_{\mathcal{BL}}$ son algebrizables.

Demostración

Las pruebas de [1], [2] y [3] son inmediatas. Veamos [4];

Caso [a] $A \Delta B \vdash \neg A \Delta \neg B$

1. $\neg A \leftrightarrow \neg B$ se deduce por $\Delta_{\neg}$.

2. $\neg\neg A \leftrightarrow \neg\neg B$, puesto que $\neg\neg A \leftrightarrow A$, se sigue de $\Delta_{\leftrightarrow}$.

3. $g_b \neg A \leftrightarrow g_b \neg B$, para fórmulas complejas se deduce de tautologías clásicas y para hiperliterales por el axioma $\mathcal{BL}_4$.

Caso [b] $A \Delta B \vdash g_b A \Delta g_b B$

1. $g_b A \leftrightarrow g_b B$ se sigue de Δ_b.

2. $\neg g_b A \leftrightarrow \neg g_b B$, para fórmula complejas se deduce de tautologías clásicas y para hiperliterales por axioma $\mathcal{BL}_4$, junto con $\Delta_{\leftarrow}$.

3. $g_c g_b A \leftrightarrow g_c g_b B$, para fórmula complejas se deduce de tautologías clásicas y el axioma $\mathcal{BL}_2$. Para hiperliterales, puesto que por axioma $\mathcal{BL}_3$ $g_c g_b A \leftrightarrow g_{c\oplus b} A$. de $\Delta_{\leftarrow}$ se prueba que $g_{c\oplus b} A \leftrightarrow g_{c\oplus b} B$, y así $g_c g_b A \leftrightarrow g_c g_b B$.

Caso [c] se deduce de $\mathcal{BL}_2$ y tautologías clásicas.

Veamos el item [5]; $\delta(A) = A \wedge A$ y $\epsilon(A) = A \to A$. En el otro sentido $\to_1$ y $\to_2$ implica que $\vdash A \to A$, y así $\vdash (A \wedge A) \to (A \to A)$, y trivialmente $A \vdash (A \wedge A) \to (A \to A)$. Así por $\wedge_3$ se deduce $A \vdash (A \wedge A)$, entonces $A \vdash (A \to A) \to (A \wedge A)$, y entonces $A \vdash (A \wedge A) \leftrightarrow (A \to A)$. Es decir, $A \vdash \delta(A) \leftrightarrow \epsilon(A)$, puesto que $\delta(A)$ y $\epsilon(A)$ son fórmulas complejas y por tautologías clásicas se tiene $A \vdash \delta(A)\Delta\epsilon(A)$. En el otro sentido $\delta(A)\Delta\epsilon(A) \vdash (A \to A) \to (A \wedge A)$, pero $(A \wedge A) \vdash A$ por $\wedge_1$, y así $\delta(A)\Delta\epsilon(A) \vdash A$ ∎

Teorema 7.15 *Si $\mathcal{BL}$ es finito, entonces la lógica anotada $OP_{\mathcal{BL}}$ es algebrizable.*

Observación 7.3 *El teorema 7.15 es válido también para los sistemas $COP_{\mathcal{BL}}$, puesto que es una generalización de $OP_{\mathcal{BL}}$.*

Las lógicas $OP_{\mathcal{BL}}$ y $COP_{\mathcal{BL}}$ son algebrizables cuando $\mathcal{BL}$ es finito, con lo cual podemos afirmar que estas gozan de un buen comportamiento, al menos desde el punto de vista algebraico.

Bibliografía

[1] Abe, J. M., *Fundamentos da Lógica Anotada*, PhD Thesis, Universidade de São Paulo, 1992.

[2] Abe, J. M. and da Silva Filho, J. I., *Manipulating Conflicts and Uncertainties in Robotics,* Multiple-Valued Logic and Soft Computing, V. 9 (2003), 147-169.

[3] Abe, J. M. and da Silva Filho, J. I., *Emmy: a paraconsistent autonomous mobile robot,* Logic, Intelligence Artificial and Robotics, Proc. 2nd Congress of Logic Applied to Technology LAPTEC 2001, Eds. J. M. Abe and J. I. Silva Filho, Frontiers in Artificial Intelligence and Its Applications, IOS Press, Amsterdan, Ohmsha, Tokyo, Editores, Vol. 71, pp. 53-61, 2001.

[4] Anderson, A. y Belnap, N. D., *Entailment: the Logic of Relevance and Necessity,* Princeton Univ. Press, 1975.

[5] Arenas, M., Bertossi, L., and Kifer, D., *Applications of Annotated Predicate Calculus to Quering Inconsistent Databases*, Springer Lectures Notes in Artificial Intelligence 1861, 2000, 926-941.

[6] Arieli, O. and Avron, A., *Reasoning with logical bilattices*, Journal of Logic Language and Information Volume 5 No. 1, (1996), 25-63.

[7] Arieli, O. and Avron, A., *A Model-Theoretic Approach for Recovering Consistent Data from Inconsistent Knowledge Bases*, Journal of Automated Reasoning Volume *22* (1999), 263-309.

[8] Arruda, A. I., *Imaginary Logic of Vasiliev,* in Non–Classical Logics, Model Theory and Computability. (Eds. A. I. Arruda, N. C. A. da Costa y R. Chuaqui), North Holland, Amsterdam, 1977, 1-24.

[9] Arruda, A. I., *A Survey of Paraconsistent Logic*, in Mathematical Logic in Latin America. (Eds. A. I. Arruda, R. Chuaqui y N. C. A. da Costa), North Holland, Amsterdam, 1980, 1-41.

[10] Asenjo, F. G., *A calculus of antinomies,* Notre Dame Journal of Formal Logic Vol. 7, (1966), 103-105.

[11] Belnap, N. D., *A useful four-valued logic,* Modern Uses of Multiple-Valued Logic, J. Michael Dunn and G. Epstein editors, D. Reidel, Boston 1977, 8-37.

[12] Barceló, P. and Bertossi, L., *Logic Programs for Querying Inconsistent Databases,* Springer Lectures Notes in Artifitial Intelligence 2562 (2003), 208-222.

[13] Béziau, J.Y., *Nouveaux résultats et nouveau regard sur la logique paraconsistantes C1*, Logique et analyse, 141-142 (1993), 45-58.

[14] Béziau, J.Y., *Idempotent Full Paraconsistent Negations are not Algebraizable,* Notre Dame Journal of Formal Logic, Volume *39*, Number 1, Winter (1998), 135-139.

[15] Blair, H. A. and Subrahmanian, V. S., *Paraconsistent Logic Programming,* Theorical Computer Science, Vol. 68, (1989), 135-154.

[16] Blok, W. J. and Pigozzi, D., *Algebrizable Logics*, Memoirs of the A.M.S., *77* Nr. 396 (1989).

[17] Blok, W. J. and Pigozzi, D., *Algebraic Semantics for Universal Horn Logic without Equality,* Universal Algebra and Quasigroup Theory, A. Romanowska, J. D. H. Smith (eds.) Heldermann, Berlin, 1992, 1-56.

[18] Blok, W. J. and Pigozzi, D., *Protoalgebraic Logics*, Studia Logica *45* (1986), 337-369.

[19] Bobenrieth, A., *Inconsistencias ¿Por qué no?*, Premios Nacionales Colcultura, Santa Fé de Bogotá, Tercer Mundo Editores, 1996.

[20] Bunder, M., *A new hierarchy of paraconsistent logics*, Proceedings of the Third Brazilian Conference on Mathemathical Logic, Edited A.I. Arruda and others, Sociedade Brasiliera de Logica, Sao Paulo 1980, 13-22.

[21] Burris, S., and Sankappanavar, H. P., *A Course in Universal Algebra,* Springer-Verlag, New York, 1981.

[22] Carnielli, W. A. and de Alcantara, L.P., *Da Costa Algebras,* Studia Logica 43 (1984), 79-88

[23] Czelakowski, J., *Consequence Operations. Foundational Studies*, Reports of the Research Project: Theories, Models, Cognitive Schemata, Institute of Philosophy and Sociology, Polish Academy of Sciences, Warszawa, 1992.

[24] Czelakowski, J., *Equivalential logics I, II*, Studia Logica 40 (1981), 227-236 and 335-372.

[25] Czelakowski, J., *Reduced Products of Logical Matrices*, Studia Logica 39 (1980), 19-43.

[26] Czelakowski, J. and Jansana, R., *Weakly Algebrizable Logics*, The Journal of Symbolic Logic *65* (2000), 641-668.

[27] da Costa, N.C.A., *Opérations non monotones dans les treillis*, C.R. Acad. Sc. Paris t. 263 (1966), 429-432.

[28] da Costa, N.C.A., *On the theory of inconsistent formal systems*, Notre Dame Journal of Formal Logic XI (1974), 497-510.

[29] da Costa, N.C.A. and Alves, E. H., *A semantical analysis of the calculi* C_n, Notre Dame Journal of Formal Logic XVIII (1977), 621-630.

[30] da Costa, N.C.A., Béziau, J.Y. and Bueno, O., *Paraconsistent Set Theory (en Portugues)*, Campinas, Centro de Lógica, Epistemología e Historia da Ciência (Colec. CLE 23), 1998.

[31] da Costa, N.C.A. y Alves, E. H., *Une Semantique pour le Calcul* C_1, Comptes Rendues Acad. Sciences de Paris 283 A (1976), 729-731.

[32] da Costa, N.C.A. y Alves, E. H., *A Semantical Analysis of the Calculi* C_n, Notre Dame Journal of Formal Logic 18 (1977), 621-630.

[33] da Costa, N.C.A. y Dubikajtis, L., *Sur la Logique Discoursive de Jaśkowski*, Bull. Acad Polonaise des Sciences Math., Astr. et Phys. 16 (1968), 551–557.

[34] da Costa, N.C.A. y Dubikajtis, L., *On Jaśkowski's Discussive Logic*. En Non–Classical Logics, Model Theory and Computability. (Eds. A. I. Arruda, N. C. A. da Costa y R. Chuaqui), North Holland, Amsterdam, 1977, 37-56.

[35] da Costa, N.C.A. yand Guillaume, M., *Négations composées et loi de Peirce dans les systèmes* C_n, Portugaliae Math., 24 (1965), 201-210.

[36] da Costa, N.C.A. and Krause, D., *Complementarity and Paraconsistency,* in S. Rahman; J. Symons; J. P. van Bendengen (eds.), Logic, Epistemology, and the Unity of Sciense, Kluwer Ac. Press, 2004.

[37] da Costa, N.C.A. y Lewin, R. A., *Lógica Paraconsistente*. En Enciclopedia Iberoamericana de Filosofía, Volumen 7 "Lógica". Consejo Superior de Investigaciones Científicas, Instituto de Filosofía. Madrid, 1995, 185-204.

[38] da Costa N. *Lógica Paraconsistente Aplicada.* Editora Atlas, Sao Paulo, 1999.

[39] da Costa, N.C.A., Abe, J. M. and Subrahmanian, V.S., *Remarks on Annotated Logic.* Zeitschrift fur Math. Logic 37 (1991) 561-570.

[40] da Costa, N.C.A., Subrahmanian, V.S. and Vago, C., *The Paraconsistent Logics* $P\tau$, Zeitschrift fur Math. Logic 37 (1991), 139-148.

[41] Ebrahim, R. *Fuzzy logic programming,* Fuzzy Sets and Systems 117 (2001) 215-230.

[42] Fidel, M., *La decibilidad de los cálculos Cn.*

[43] Fitting, M. C., *Bilattices and the Semantics of Logic Programming,* Journal of Logic Programming 11 (1991), 91-116.

[44] Font, J. and Jansana, R., *A General Algebraic Semantics for Deductive Systems,* Lecture Notes in Logic 7, Springer Verlag, 1996.

[45] Früwirth, T., *Temporal Annotated Constraint Logic Programming,* Journal of Symbolic Computation Volume 22, Number 5-6, November 1996, pp. 555-583.

[46] Ginsberg, M., *Bilattices and Modal Operators,* Proc. 1990 Intl. Conf. On Theoretical Aspects of Reasoning About Knowledge, Morgan Kaufmann 1990, 273-287

[47] Guadarrama, S., Muñoz, S. and Vaucheret, C. *Fuzzy Prolog: a new approach using soft constraints propagation,* Fuzzy Sets and Systems 144 (2004), 127-150

[48] Herrmann, B., *Equivalential and Algebrizable Logics,* Studia Logica 57 (1997), 419-436.

[49] Herrmann, B., *Characterizing Equivalential and Algebrizable Logics and Definability by the Leibniz Operator,* Studia Logica 58 (1997), 305-323.

[50] Kim, C., Kim, D. and Park, J. *A new fuzzy resolution principle based on the antonym,* Fuzzy Sets and Systems 113 (2000) 299-307

[51] Kifer, M. and Lozinskii, E. L., *A Logic for Reasoning with Inconsistency,* Journal of Automated Reasoning Marcel Dekker, Inc., 9(2)(1992), 335-368.

[52] Kifer, M. and Subrahmanian, V. S., *Theory of Generalized Annotated Logic Programming and its Applications,* Journal of Logic Programming Volume 12 (1992), 335-367.

[53] Lee R. C. T., *Fuzzy Logic and the resolution principle,* Journal of the Association for Computing Machinery, Volume 19, No. 1 (1972), 109-119.

[54] Lewin R. A., Mikenberg I. F., Schwarze M. G., *Algebraization of Paraconsistent Logic P1*, Journal of Non Classical Logic 7, 1-2(1990), 79-80.

[55] Lewin R. A., Mikenberg I. F., Schwarze M. G., *C1 is not Algebraizable*, Notre Dame Journal of Formal Logic 32 (1991), 609-611.

[56] Lewin R. A., Mikenberg I. F., Schwarze M. G., *Matrix Semantics for Annotated Logics*, In Models, Algebras and Proofs, (Caicedo, X. y Montenegro, C. Eds.), Marcel Dekker, Lecture Notes in Pure and Applied Mathematics 203, Proceedings of the X SLALM, Bogotá 1995, 1999, 279-293.

[57] Lewin R. A., Mikenberg I. F., Schwarze M. G., *On the Algebraizability of Annotated Logics*, Studia Logica 57 (1997), 359-386.

[58] Lu, J.J., Murray, N.V. and Rosenthal, E., *A Framework for Automated Reasoning in Multiple-Valued Logics*, Computer Sciense Technical Report 1997, Bucknell University.

[59] Lu J., *Deduction in multiple-value Logics.* Department of mathematics and computer science Emory University. Atlanta USA, 2003.

[60] Mendelson, E., *Introduction to Mathematical Logic*, Chapman & Hall/CRC, 1997.

[61] Miró Quesada, F., *La Lógica Paraconsistente y el Problema de la Racionalidad de la Lógica*, Congreso Latinoamericano de Lógica, Caracas 1983, 693-622.

[62] Mortensen, C., *Every Quotient Algebra for C_1 is Trivial*, Notre Dame Journal of Formal Logic 21 (1980), 694-700.

[63] Nguyen, H. and Walker, E., *A First Course in Fuzzy Logic,* Chapman-Hall CRP Press, Inc., London, 2000.

[64] Ortíz, G., *The Annotated Logics OP_{BL}*, Lecture Notes in Pure and Applied Mathematics, *228* (2002), 411-433.

[65] Petrich, M., *Introduction to Semigroups,* Charles E. Merrill Publishing Company, Columbus, Ohio 1973.

[66] Raffaeta, A., *Spatio-Temporal Annotated Constraint Logic Programming,* Lecture Notes in Computer Science, Volume 1990 (2001), 259-273.

[67] Rasiowa, R., *An Algebraic Approach to Non-Classical Logics*, North–Holland Pub. Co., Amsterdam, 1974.

[68] Rasiowa, R. and Sikorski, R., *The Mathematics of Metamathematics*, Polska Akademia Nauk, Warsawa, 1963.

[69] Sette, A.M., *Sobre as algebras e hyper-reticulados Cω*. Tese do Mestre em Matemática, Campinas 1971.

[70] Subrahmanian, V. S., , *On the Semantics of Quatitative Logics Programs*, In Proc. 4^{th} IEEE Symposium on Logic Programming, pp. 173 - 182, San Francisco, 1987.

[71] Sylvan, R., *Variations on da Costa C Systems and Dual-Intuitionistic Logics*, Studia Logica XLIX,1 (1990), 47-65.

[72] Tarski, A., *Logic, Semantics, and Metamathematics, papers from 1923 to 1938*, Hackett Pub. Co., Indianapolis, Indiana, (1983).

[73] Tarski, A., *Grundzüge des Systemenkalküs. Erster Teil.*, Fundamenta Mathematica, 25 (1935), 503-526.

[74] Urbas, I. *Paraconsistency and the C-Systems of da Costa*, Notre Dame Journal of Formal Logic, Volume*30*, Number 4, Fall (1989), 583-597.

[75] Van Emden, M. H., *Quatitative deduction and its Fixpoint Theory*, Journal of Logic Programming, Vol. 1, (1986), 37-53.

[76] Viedma, A. et. al. *Fuzzy temporal constraint logic: a valid resolution principle*, Fuzzy Sets and Systems 117 (2001)

[77] Voutsadakis, G., *Categorical Abstract Algebraic Logic: Categorical Algebrización of Equational Logic*, Logic Journal of the IGPL, 12 (4) (2004), 313-333, Oxford University Press.

[78] Weigert, T. J., Tsai, J. P. and Liu, X., *Fuzzy Operator Logical and Fuzzy Resolution*, Journal of Logic Automated Reasoning *10* (1993), 59-78.