Lógica, Lógicas y Logicidad

María Manzano

10 de Mayo de 2004

Índice general

Índice de figuras

Prefacio

Dentro de la lógica se distinguen tres grandes ramas y una externa que abarca los estudios sobre ella. Se pueden resumir en un cuadro como el que sigue:

1. **Fundamentos:** Esta división recoge los rudimentos y contenidos fundamentales de nuestra disciplina, se podría denominar también *Lógica matemática,* e incluye dominios que igualmente pueden considerarse parte de las matemáticas, aunque sean integrantes de pleno derecho de la lógica moderna, al menos por su origen. Bajo esta categoría se engloban las siguientes subramas: *Lógica Básica, Teoría de la Demostración, Teoría de Modelos, Teoría de la Computación* y *Teoría de Conjuntos.*

2. **Sistemas Lógicos:** Una buena parte de nuestra actividad investigadora consiste en diseñar sistemas lógicos de la más diversa índole. Tradicionalmente se denominaban *lógicas no-clásicas*, muchos de ellos pertenecerían a la clase de *extensiones de la lógica clásica*; llamar no clásicas a unas lógicas cuyos orígenes se remontan a Aristóteles no me parece muy adecuado, por otro lado, no todos los sistemas lógicos son extensiones. Así que he preferido llamarlos simplemente sistemas lógicos. El número de colecciones, apartados y divisiones que aparecen en esta rama es tan inmenso que lo único que cabe hacer es presentar una lista que con el tiempo seguirá llenándose.

3. **Aplicaciones:** Este apartado intenta cubrir dos frentes al mismo tiempo, el de las posibles aplicaciones de la lógica, o de sistemas lógicos, a ámbitos distintos, tales como la *Informática*, la *Lingüística*, la *Economía* o la *Ciencia*, y el de la elaboración de programas informáticos que los implementen y automaticen.

4. **Estudios sobre Lógica:** Este apartado contiene diversos estudios externos que versan sobre la lógica como disciplina: *Historia de la Lógica, Filosofía de la Lógica* y *Pedagogía de la Lógica.*

Puesto que sería imposible hacer en el limitado espacio de un solo libro un repaso de todas las ramas y subramas, lo que he hecho es introducir con un cierto detalle el apartado de *fundamentos* (primera parte), elegir para su presentación varios sistemas lógicos de interés en filosofía, informática e I.A (segunda parte),

centrándome en la tercera en proponer un tratamiento unificador propio[1] y en comparar los resultados obtenidos con los de otros planteamientos diseñados para el mismo fin. Esta última parte la situaría en la rama de *Estudios sobre Lógica*, en particular, en *Filosofía de la Lógica*. aunque su inclusión en *Aplicaciones* estaría igualmente justificada, de hecho la *CUP* publicó el libro mencionado en una colección de informática teórica. La razón que yo aduzco ahora es que aunque la reducción de sistemas resulta extraordinariamente fecunda y con aplicaciones en áreas tales como la demostración automática, también nos ha servido para satisfacer intereses de naturaleza filosófica cuando buscábamos determinar donde reside *la logicidad*, de ahí el título del libro: *Lógica, Lógicas y Logicidad.*

He añadido un capítulo de conclusiones, el 17, en el que analizo lo hecho, incluso me aventuro a identificar qué es o podría ser la logicidad. Por supuesto, me he dejado muchas cosas en el tintero, especialmente en las ramas de *Estudios* y *Aplicaciones*, pero creo que he conseguido así una mayor cohesión y una extensión razonable.

He pretendido que fuera un manual enciclopédico legible, por lo que tiene una concepción unitaria, los diversos capítulos están interconectados de manera que los temas en ellos tratados se complementan, no se producen repeticiones y he procurado que el nivel quedara en lo que nuestros alumnos podrían entender al finalizar el primer ciclo, o en lo que los investigadores del área de Lógica, Metodología y Filosofía de la Ciencia, o de áreas afines (muy en especial las de informática, matemáticas y linguística) no especializados en lógica necesitarían para compartir mi visión de este campo. Las definiciones son precisas y formalmente cuidadas, los teoremas reseñados son los más relevantes, y hay en cada capítulo al menos una demostración de alguno de ellos, frecuentemente con un organigrama. Cada capítulo contiene su bibliografía específica, que he procurado estuviera al alcance del lector potencial de este libro por lo que, siempre que ha sido posible, he elegido libros o capítulos de libros en vez de artículos y en todos los casos seleccionando los que no fueran excesivamente técnicos.

La *Pedagogía de la Lógica* no está ausente, aunque las indicaciones didácticas son casi todas indirectas, y no porque no me preocupe por ella, sino porque creo que la mejor manera de contribuir a la innovación, sistematización y mejora de la tarea educativa es embarcarse en proyectos de esas características, elaborando material apropiado junto a otros docentes e investigadores. Así lo he hecho y los resultados están disponibles en

$$http://aracne.usal.es$$

La *Historia de la Lógica* sólo aparece en capítulos dedicados a otros temas, aunque es un campo que valoro y me puede apasionar. Por mantener más compensado este texto, la he ubicado en introducciones y apéndices. Por ejemplo, en el apartado titulado *La piedra filosofal* (ver sección 11.6) explico porqué creo

[1] El de mi libro *Extensions of First Order Logic*, publicado por la Cambridge University Press en 1996.

que el cálculo lambda configuró casi todos los resultados de Alonzo Church, desde su famosa tesis, pasando por su teorema de indecidibilidad hasta los resultados de completud de Henkin y la teoría de recursión de Kleene.

La *Filosofía de la Lógica* es el motor de la parte de *Tratamientos unificadores*, pero he dejado sin desarrollar muchos temas que me interesan: por ejemplo, el de *La Identidad*, *La Infinitud* o *Las Paradojas*. Como en el caso anterior, incluso estos temas han pasado a formar parte de otros capítulos. Y por supuesto, la inspiración de muchos de los capítulos e incluso de muchos teoremas es radicalmente filosófica; este es el caso de la demostración de la incompletud de la lógica superior, que nosotros hacemos de manera alternativa a la de Gödel utilizando la capacidad expresiva de semejantes sistemas lógicos.

Parte I

Fundamentos

Capítulo 1

Lógica Básica

1.1. Introducción

1.1.1. ¿Qué es la Lógica?

El objetivo fundamental de este capítulo es el de introducir, de manera intuitiva, los conceptos fundamentales de la lógica, y muy particularmente, el concepto de consecuencia, ya que la lógica puede ser definida como el *estudio de la consecuencia*; o lo que es lo mismo, como el estudio de los razonamientos válidos o correctos. Yo la caracterizo como el *estudio de los conjuntos de creencias consistentes* porque pienso que de esta forma es más fácil al comienzo y porque se sabe que los dos planteamientos son equivalentes, como puntualizo insistentemente en un curso de introducción.

En sentido amplio

La *Lógica* es lo que tienen en común ciencias tan dispares como:

MATEMÁTICAS
FILOSOFÍA
LINGÜÍSTICA
INFORMÁTICA
DERECHO
FÍSICA
SOCIOLOGÍA
⋮

Tratándose de disciplinas tan diferentes lo que comparten no puede ser *el tema de estudio,* tampoco *la metodología.*

¿Será tal vez el uso de la ***racionalidad****, la* ***coherencia****, la búsqueda de la* ***consistencia*** *o compatibilidad de las creencias en cada una de estas ciencias?*

La respuesta es que sí, pero también que la *Lógica* es más que eso[1]: Todos nosotros, supuestos seres racionales, empleamos la lógica cuando razonamos, asimilamos o procesamos la información que recibimos del entorno, cualquier tipo de información —somos lógicos porque somos seres humanos—. Tradicionalmente se definía

$$Hombre = Animal{+}Racional$$

y sabemos que el comportamiento racional implica el uso de la lógica como herramienta. Mas allá de las etimologías, atendiendo a los usos propios de las palabras,

$$Racionalidad \implies Lógica$$

En sentido coloquial se usa el adjetivo lógico no sólo para describir las reglas del razonamiento correcto, sino en una gran variedad de casos, más en concordancia con el uso original del *"logos"* de los griegos, relacionándolo con el lenguaje, la doctrina, la estructura del conocimiento, la razón, etc.

Comentario 1 *Durante el siglo XX la lógica fue retomando su extensión y amplitud originales estudiándose en ella no sólo el razonamiento matemático sino también fenómenos de gestión y transmisión de información, de toma de decisiones y de la acción y en general en casi todos los contextos gobernados por reglas. Siguiendo esta línea de extensión del concepto de lógica, nosotros en un curso introductorio nos beneficiamos de las ventajas del razonamiento diagramático, visual. Utilizamos para ello varias aplicaciones informáticas, tanto propias como ajenas*[2]*.*

En sentido estricto

La *Lógica* es también una disciplina en sí misma, una de las grandes ramas del conocimiento.

$$Lógica \;=\; \text{estudio de la } consecuencia$$

—esto es, la que se ocupa de los razonamientos válidos o correctos—

$$Lógica \;=\; \text{estudio de la } consistencia$$

—a saber, los conjuntos de creencias coherentes, *consistentes, satisfacibles*—

Puesto que en el campo de la lógica se cifra no sólo el razonamiento atemporal y estático de la matemática, sino también el temporal del razonamiento aplicado al mundo real, el metateórico de nuestra reflexión sobre la lógica misma, el

[1] En el capítulo 12, sección 12.1, nos volvemos a plantear la pregunta *¿Qué es un sistema lógico?* e intentamos apuntar soluciones a un nivel menos introductorio; sin embargo, en un primer contacto con la materia es mejor sugerir que *hiperdefinir*.

[2] Ver la sección 1.8.

filosófico de nuestra reflexión sobre el pensamiento y el dinámico sobre los resultados de la ejecución de acciones, o los procesos de transmisión de información, no hay una única lógica, sino multitud de ellas.

Comentario 2 *En un curso introductorio sólo nos ocupamos de la lógica clásica, tanto de la proposicional como de la de primer orden; el tránsito de una a otra será pausado, haciendo una parada en la lógica de primer orden de predicados monarios y con una sola variable, usando como cálculo los diagramas de Venn*[3].

En sentido funcional

Para definir *"una" Lógica* se introduce un lenguaje artificial; con alfabeto y reglas *gramaticales* de formación de fórmulas y se atribuye significado a las expresiones del lenguaje mediante interpretaciones *semánticas* o *modelos*. Dichas interpretaciones nos permiten afirmar, en algunos casos, que de ciertos conjuntos de fórmulas —que se toman como hipótesis— se siguen ciertas fórmulas como conclusión. Es decir, que son consecuencia semántica de las hipótesis consideradas.

$$Lógica = Gramática{+}Semántica$$

En algunas ocasiones se puede definir un *cálculo deductivo* para mecanizar el proceso de extraer conclusiones a partir de hipótesis. Por supuesto, se desea que el cálculo sea una réplica mecanizable de dicho proceso; es decir, equivalente —con los mismos resultados—.

$$Semántica \iff Cálculo$$

Comentario 3 *El proceso puede ser el inverso: Introducir primero el lenguaje y las reglas del cálculo, y posteriormente las interpretaciones o modelos. La historia de la lógica está plagada de ejemplos de las dos clases. Simplificando, la visión de la lógica clásica, especialmente la que anima la* ***Teoría de Modelos*** *es la que aquí se ha expuesto; el planteamiento sintáctico alternativo es el que se usó en el pasado para introducir los* ***Sistemas de Lógica Modal*** *y se sigue usando actualmente en algunas lógicas para la informática, la filosofía y la* ***I.A****.*

Podéis consultar el artículo que a este respecto escribió Gladys Palau [23], que empieza así:

> *"En la lógica contemporánea se habla de dos nociones de consecuencia: por un lado, la noción de consecuencia sintáctica, comúnmente identificada con la noción de deducibilidad, representada por el signo deductor* $\vdash$ *; y por el otro, la noción de consecuencia semántica, identificada generalmente con la noción de consecuencia lógica y representada por el signo* $\models$*. Ambas acepciones han dado lugar a distintos enfoques de la lógica que tienen sus defensores y detractores, según sea la concepción filosófica que se sostenga respecto de la lógica."*

[3] Ver la sección 1.8.

Una *Lógica* es una *herramienta* que nos sirve para computar razonamientos, especialmente cuando el rigor y la precisión son imprescindibles. Esto sucede en matemáticas, filosofía, informática, etc. Pero es también un *objeto de estudio*; podemos ver qué propiedades tiene el lenguaje, si el concepto de consecuencia se puede retener mediante las reglas de un cálculo, si hay cálculos más efectivos, incluso si existen algoritmos capaces de suplirlos. Estudiaremos las denominadas metapropiedades de: *corrección, completud [sección 2.4]* y *decidibilidad [sección 3.7]*. Por supuesto, para hacerlo necesitamos tomar cierta distancia, observar los lenguajes formales desde la atalaya del metalenguaje. Estos estudios se conocen con el nombre genérico de *Metalógica*; aquí los he distribuido entre los capítulos de *Teoría de Modelos [capítulo 2]*, *Teoría de la Recursión [capítulo 3]* y *Teoría de la Prueba [capítulo 4]*.

1.1.2. Históricamente

Pasado

El estudio de la lógica se remonta a los filósofos griegos; en el Organon de Aritóteles se estudian los principios del silogismo. A mediados del siglo XIX Boole (1815-1864) creó el primer cálculo lógico, para la lógica proposicional. La lógica en sentido moderno nace a finales del siglo XIX y principios del XX.

Dentro de la lógica se distinguen tres grandes ramas y una rama externa que incluye los estudios sobre ella. Se puede resumir en un cuadro como el que sigue (figura: 1.1)

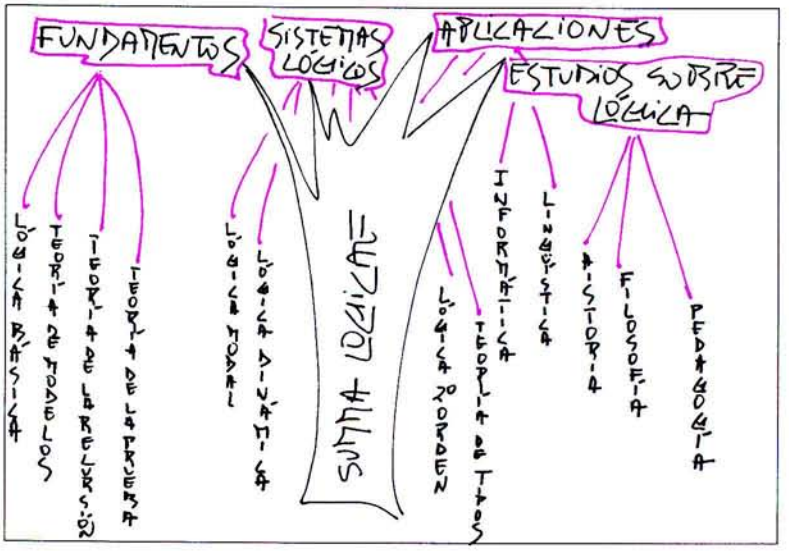


Figura 1.1: *Summa Logicae*

A continuación trazamos muy brevemente el arranque de la historia de lo

que se podría englobar bajo el epígrafe de *Lógica matemática* —yo he elegido un nombre más neutro, *Fundamentos*, pero podría haberlo llamado *lógica* a secas—.

TEORÍA DE LA PRUEBA[4]

Frege (1848-1925), Peano (1858-1932), Russell (1872-1970), Hilbert (1862-1943), Herbrand (1908-1931) y Gentzen (1909-1945) desarrollaron la *Teoría de la Demostración* de la lógica de primer orden. Todos ellos pretendían sistematizar el razonamiento matemático y atacar con la poderosa artillería lógica la fundamentación de la matemática.

Frege es el padre de la lógica moderna, al que debemos gran parte de las distinciones y conceptos en ella usados. El primer cálculo para la lógica de primer orden fue el *Begriffschrift* de Frege. Russell y Whitehead con su *Principia Mathematica* intentaron reducir los conceptos matemáticos —de la aritmética y el álgebra— a conceptos lógicos. Peano axiomatizó la aritmética.

La teoría de la prueba en un sentido mucho más delimitado nació con el denominado *programa de Hilbert*. La idea de Hilbert era la de explotar al máximo la naturaleza finita de las pruebas para proporcionar una fundamentación de la matemática. Podría resumirse su concepción diciendo que preconizaba una axiomatización de las teorías matemáticas de la que pudiera probarse su:

1. *Consistencia.* Es decir, que nunca se podrá demostrar como teoremas de la teoría una sentencia y su negación

2. *Completud.* Es decir, que cada sentencia —del lenguaje en el que se axiomatizó la teoría— sea ella misma o su negación un teorema de la teoría axiomática

3. *Decidibilidad.* Es decir, que exista un procedimiento efectivo mediante el cual, en un número finito de pasos, se determine si una sentencia del lenguaje es o no un teorema de la teoría

Los sistemas de cálculo de Gentzen condujeron a la teoría de la prueba por sus actuales derroteros, ligada inexorablemente a la perspectiva informática. El teorema de Herbrand de 1930 y, posteriormente, el de Robinson se consideran los pilares de la *demostración automática de teoremas*.

TEORÍA DE MODELOS[5]

En el nacimiento de la lógica de primer orden participan decisivamente otro grupo de investigadores cuya orientación apuntaba a la, posteriormente bautizada, *Teoría de Modelos.* Löwenheim (1878-1957), Skolem (1887-1963), Gödel (1906-1978) y Tarski (1901-1983) son los pioneros de otra línea de investigación consistente en el estudio de las estructuras matemáticas considerando las leyes a las que obedecen. Löwenheim y Skolem demostraron teoremas generales acerca

[4] Le he dedicado el capítulo 4.
[5] Le dedico el capítulo 2.

de la infinita variabilidad de la cardinalidad de los modelos de las teorías de primer orden, de la incapacidad manifiesta de esa lógica para caracterizar estructuras infinitas, y para distinguir entre dichas cardinalidades. Gödel demostró la completud del cálculo de la lógica de primer orden. A Tarski le debemos los conceptos fundamentales de la semántica y de la teoría de modelos. A él le cabe además el mérito de haber concebido y dirigido un programa de investigación sistemática en esta disciplina.

En 1931 Gödel demostró que si la aritmética elemental es consistente, no puede ser completa, y que en general el programa de Hilbert es irrealizable. Para demostrar este teorema, conocido como *teorema de incompletud*[6], Gödel introdujo el concepto de recursividad.

Comentario 4 *Estamos usando el término completud de dos formas: (1) completud de una lógica y (2) completud de una teoría. En el primer caso es una propiedad del cálculo; a saber, que es capaz de generar como teoremas a todas las fórmulas válidas. En el segundo caso es una propiedad de una teoría; a saber, la de ser tan potente que toda sentencia del lenguaje (o su negación) se derive de la teoría. En el capítulo siguiente, en la sección 224, explico la divergencia y el parentesco entre ambos usos.*

TEORÍA DE LA RECURSIÓN[7]

¿Cuándo decimos que una función es recursiva?, ¿Qué significa ser recursiva?

Hay varias definiciones precisas, equivalentes entre sí, de este concepto. La noción intuitiva correspondiente es la de ser *efectivamente computable*.

¿Cuándo decimos que una función es efectivamente computable?

Sencillamente, cuando hay un procedimiento efectivo —esto es, un algoritmo— que la computa. Éste debe cumplir una serie de requisitos. Sin embargo no le imponemos restricciones de naturaleza práctica; por ejemplo, en una función sobre los naturales, los argumentos han de serlo, pero de cualquier cardinalidad. El procedimiento ha de ser finito, pero no hay limitación previa, tampoco se prefija la cantidad de papel —o espacio de memoria— del que se dispone para realizar el cálculo. La computabilidad efectiva no es lo mismo que la práctica, lo sería en una situación ideal en la que no importase ni el tiempo ni el espacio de memoria necesario.

Los orígenes de la teoría clásica de la recursión pueden hallarse en Dedekind, cuando en 1988 introduce el estudio de las funciones definibles sobre el conjunto de los números naturales usando ecuaciones y, *recurrentemente*, la inducción sobre los números naturales que él había formulado y precisado. De ahí le viene justamente el nombre.

Por lo que respecta a su estadio presente, cuyo radio de acción cubre la totalidad de las funciones efectivamente computables, los orígenes hay que buscarlos

[6] Ver la sección 3.7 para la incompletud de la teoría de los naturales y la sección 10.5 para la incompletud de la lógica de segundo orden.

[7] Le dedico el capítulo 3.

en el grupo de Princeton; empezó con Church (1903-1995), pero si hay que atribuirle un padre, éste es Kleene. Él fue quien la impulsó, definió y acotó: suyos son los *teoremas de la forma normal* y el de *recursión.*

En cuanto a la definión misma, circulaban varias versiones de este concepto, aunque había cierta resistencia a aceptarlas como definiciones. Varios de estos conceptos aparecieron en los años 30 para caracterizar nociones que en principio parecían diferentes: la primera era la caracterización de Gödel de las funciones definidas mediante recursión, la segunda era la de función definible mediante el operador λ, que Church y Kleene introdujeron, y la tercera era la de función computable mediante una máquina abstracta, las máquinas de Turing. Pronto se demostró que las tres nociones definían las mismas funciones[8].

TEORÍA DE CONJUNTOS[9]

En el último cuarto del siglo XIX se vivió un episodio apasionante de la historia de las matemáticas que las ligaría desde entonces a la historia de la lógica. Primero, George Boole (1815-1864) en su *Mathematical Analysis of Logic* trató de presentar la lógica como parte de las matemáticas. Poco después Gottlob Frege (1848-1925) intentó mostrar que la aritmética era parte de la lógica en su *Die Grundlagen der Arithmetik.* Cantor había demostrado que la totalidad de los números reales comprendidos en el intervalo de extremos 0 y 1 no es numerable, en el sentido de que su infinitud no es de la misma magnitud que la de los números naturales. Como una consecuencia de esa situación, Cantor creó una nueva disciplina matemática entre 1874 y 1897: la *Teoría de Conjuntos.* Su obra fue admirada y condenada simultáneamente por sus contemporáneos. Desde entonces los debates en el seno de la teoría de conjuntos han sido siempre apasionados, sin duda por hallarse estrechamente conectados con importantes cuestiones lógicas.

Según la definición de conjunto de Cantor, éste es *"una colección en un todo de determinados y distintos objetos de nuestra percepción o nuestro pensamiento, llamados los elementos del conjunto".* Frege fue uno de los admiradores de la nueva teoría de Cantor, y dio una definición de conjunto similar.

En 1903 Bertrand Russell demostró que la teoría de conjuntos de Cantor era inconsistente y cuestionó la definición de conjunto en la teoría de Cantor. Pero pronto la teoría axiomática de Zermelo (1908) y refinamientos o nuevas formulaciones de ésta debidos a Fraenkel (1922), Skolem (1923), von Newman (1925) y otros, sentaron las bases para la teoría de conjuntos actual.

Es un hecho que la teoría de conjuntos forma parte de la matemática, es además, la teoría utilizada para fundamentar la aritmética y el resto de teorías de la disciplina. Pero a su vez puede formalizarse en primer orden, convirtiéndose en una más, sujeta a los avatares de cualquiera de ellas.

En esta historia cruzada de las matemáticas, la lógica y los fundamentos de ambas, la teoría de conjuntos permitirá por un lado una fundación *logicista* de las matemáticas; pero por otro lado la teoría de conjuntos considerada como

[8] Véase la sección 11.6.
[9] Le dedico el capítulo 5.

parte de las matemáticas proporciona el metalenguaje, el contexto o substrato de las teorías lógicas. Finalmente, puede ser completamente expresada en un lenguaje de primer orden y sus axiomas y teoremas constituyen una teoría de primer orden a la que pueden aplicarse los resultados generales que se aplican a cualquier teoría de primer orden.

Presente

En la primera mitad del siglo XX la lógica se aplicó mayormente a la fundamentación de la matemática. En la segunda mitad jugó un papel decisivo en la creación y desarrollo de la informática y de los lenguajes de programación, hasta el extremo de poderse caracterizar a la informática así:

$$\textit{Informática} = \textit{Lógica+Ingeniería electrónica}$$

La *Lógica* proporciona los fundamentos para las diversas —cada vez más abundantes— *aplicaciones de la lógica en la informática: verificación de hardware y software, inteligencia artificial, programación lógica, deducción automática, etc.*

Futuro

Pero, como dijimos anteriormente, durante el siglo XX la lógica fue retomando su extensión y amplitud originales estudiándose en ella no sólo el razonamiento matemático sino también fenómenos de gestión y transmisión de información, de toma de decisiones y de la acción y en general en casi todos los contextos gobernados por reglas. Siguiendo esta línea de extensión del concepto de lógica, hay varias líneas de investigación abiertas[10] entre las que cabe destacar: razonamiento con diagramas, lógica dinámica, teoría de juegos.

La *Lógica* es la materia interdisciplinar por excelencia y actúa como núcleo de una ciencia que emerge: la *ciencia de la transmisión de la información.*

$$\text{Triángulo de las Bermudas} = \textit{Lógica, Lenguaje e Informática}$$

Por supuesto la metáfora es que los investigadores se pierden al adentrarse en él.

Por consiguiente, concentrarnos en estudiar los principios que gobiernan la lógica tiene un carácter ejemplificador pues en ella se funden disciplinas en las que son determinantes los aspectos simbólicos del proceso de transmisión de información; esto es, en todas aquellas en las que es conveniente usar lenguajes artificiales.

Empezaremos estudiando la denominada *lógica clásica*, tanto proposicional como de primer orden[11]. Ello será imprescindible tanto si queremos profundizar

[10] Esto constituye una parte importante del proyecto de investigación *Summa Logicae en el siglo XXI.* Véase: http://logicae.usal.es

[11] Los temas más interesantes aparecen distribuídos en los distintos capítulos que constituyen la primera parte de este texto.

después en cualquiera de los campos mencionados, como si la usamos como mera herramienta.

Comentario 5 *La **lógica clásica** se distingue por su rigor y precisión pero carece de matices: la verdad es absoluta, el tiempo está ausente, no existe la ambigüedad. Está especialmente diseñada para caracterizar el razonamiento de las matemáticas y cuando se aplica a ámbitos no matemáticos, se matematizan previamente.*

Comentario 6 *Hay otras lógicas*[12]*: Temporal, modal, dinámica, epistémica, deóntica, multivariada, de orden superior, intuicionista, borrosa, no-monotónica,...*

Resumen 7 *Hemos definido a la lógica de tres maneras diferentes:*

1. Lógica = *estudio de la* consecuencia *(razonamientos válidos o correctos)*
2. Lógica = *estudio de los conjuntos de creencias* ***consistentes***
3. Lógica = ***Gramática + Semántica (+ Cálculo)***

1.2. Consistencia

La *consistencia lógica* o coherencia interna de un conjunto de creencias significa para nosotros *compatibilidad de creencias.*

Hay que distinguir la consistencia lógica, que es una cualidad formal, abstracta, de ciertas virtudes, por otra parte muy estimables, como la lealtad, la justicia o la sinceridad. Por su parte, la inconsistencia no hay que confundirla con la estupidez o la irracionalidad, aunque estén próximas. Hay que distinguirla también, y esto es más difícil, del desacuerdo con la realidad.

$$\begin{array}{l}\text{Consistencia} \neq \text{lealtad}\\ \text{Consistencia} \neq \text{justicia}\\ \text{Consistencia} \neq \text{sinceridad}\\ \text{Inconsistencia} \neq \text{estupidez}\\ \text{Inconsistencia} \neq \text{irracionalidad}\\ \text{Inconsistencia} \neq \text{desacuerdo con la realidad}\end{array}$$

Comentario 8 *Un conjunto de creencias puede muy bien estar en desacuerdo con la realidad y no ser inconsistente, pues no existe incompatibilidad de creencias. Los conjuntos consistentes de creencias se caracterizan porque es siempre posible imaginar una situación (un modelo) en la que todas ellas sean verdaderas, pero puede no ser la del mundo real.*

[12] Muchas de las mencionada se tratan en este volumen, de la mayoría se puede encontrar información en

$$http://logicae.usal.es$$

Todas aparecen en los distintos volúmenes de los diversos manuales enciclopédicos entre los que cabe destacar: [1], [17], [15] y [16]

Comentario 9 *Nadie sostiene a sabiendas creencias inconsistentes. Las leyes lógicas ¿son naturales?, ¿ convencionales?, ¿se adquieren?, etc. Estas preguntas han obtenido respuestas muy variadas a lo largo de la historia. Algunos consideran que las leyes de la lógica son puramente convencionales y que se pueden cambiar, pero la intuición abrumadora y generalizada es que son más fundamentales y estables que las leyes de tráfico e incluso que las de la física.*

La *consistencia* también se puede predicar de una creencia aislada; en tal caso ser consistente es poder ser verdadero en una situación, no necesariamente en todas, ni tan siquiera se exige que lo sea así en la realidad. La *Inconsistencia o Contradicción* es mucho más fuerte: no puede ser verdadero en ninguna situación.

Ejemplo 10 ***¡Políticos!***
Uno de nuestros insignes políticos manifiesta:

- *"Es un error censurar, por violentas, la retransmisión de las corridas de toros porque lo que vemos en la televisión no afecta en absoluto el comportamiento; ni siquiera el de los jóvenes".*
- *"Debería haber más programas y documentales que mostraran nuestras costumbres nacionales (bailes típicos, corridas de toros, concursos de cortar troncos, etc) para así fomentar estas costumbres entre los jóvenes".*

  Suponiendo que dice lo que cree ***¿Son consistentes sus creencias?***

Ejemplo 11 ***El barbero de Las Batuecas***

Hace pocos días me contaron el caso de un hombre llamado Roque, barbero en Las Batuecas. Sólo me habían dicho dos frases cuando exclamé: ¡Imposible!

- *"Roque vive en Las Batuecas"*
- *"Roque afeita a los habitantes de Las Batuecas que no se afeitan a sí mismos y sólo a ellos"*

  ¿Me precipité al no creerme lo que me contaban?

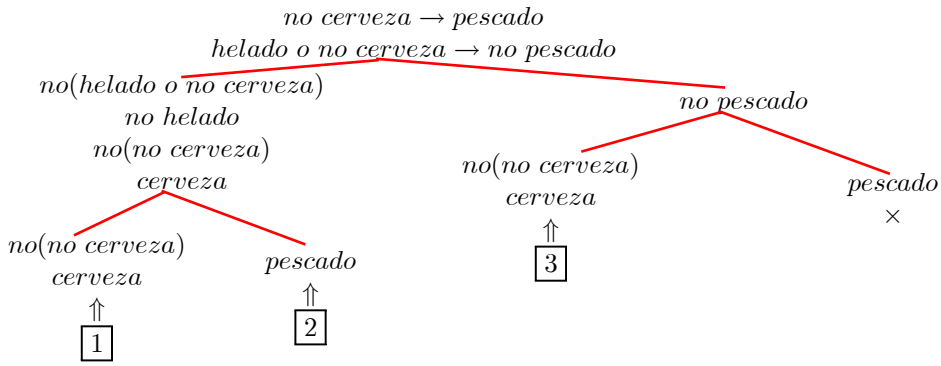
Para verificar la consistencia de un conjunto de creencias lo que necesitamos es ser capaces de describir una situación en la que todas sean verdaderas. Podemos utilizar los tableaux semánticos y colocar las condiciones requeridas en las ramas de un árbol: las abrimos para expresar alternativas y en la misma rama situamos las que deban ser satisfechas simultáneamente.

Ejemplo 12 ***Régimen para una larga vida.*** *Un periodista entrevista a un anciano centenario y éste le revela el secreto de su longevidad, que reside, según él, en su alimentación. El anciano dice:*

- *"Si no bebo cerveza, entonces como pescado"*

- *"No como pescado, si tomo helado o no bebo cerveza"*

  ¿Se puede seguir un régimen así? ¿Podrías hacer el menú de un par de días?



Veamos las ramas abiertas: 1, 2 y 3. En 1 sabemos que el menú debe incluir cerveza pero no helado y el resto se deja al *"gusto del consumidor"*, en 2 debe comer pescado, cerveza y prescindir del helado y en 3 toma cerveza pero no pescado. ¡Menudo amante del lúpulo!

1.3. Enunciados

Puesto que las creencias son inmateriales, intangibles, nos hemos ocupado de su expresión mediante el lenguaje, y mejor aún, como las palabras se las lleva el viento, mediante el lenguaje escrito. Los enunciados que sirven para expresar creencias son los que son susceptibles de ser verdaderos o falsos, aunque no sepamos en un momento dado su valor de verdad.

Por ejemplo, el enunciado
"Pernambuco es un estado de Brasil, cuya capital fue Olinda"
es un enunciado de creencia, que es verdadero en el mundo real, aunque algunos tal vez no lo sepan. Para comprobarlo bastaría consultar un atlas. Sin embargo, lo que lo hace apropiado para expresar creencias es su modalidad enunciativa.

El siguiente enunciado
"Todo entero par mayor que dos es igual a la suma de dos primos"
expresa una creencia, ¡es la famosa conjetura de Goldbach! Pero aunque ha de ser verdadero o falso, no sabemos exactamente cual de los dos valores adoptará, si finalmente alguien consigue demostrar el enunciado o su negación. Se trata de un enunciado, aunque tal vez nunca descubramos su valor de verdad.

Para nosotros lo importante es que sea un enunciado capaz de expresar una creencia.

Es de todos sabido que la relación entre pensamiento y lenguaje plantea muchos problemas, incluso cuando dejamos de lado cuestiones fundamentales

tales como la hipótesis del *determinismo lingüístico*[13].

1. En primer lugar, hay enunciados, tales como las preguntas, las órdenes, las exclamaciones o las dudas que no expresan creencias. Estos enunciados no los emplearemos. Por consiguiente, nos limitaremos al *uso aseverativo* —declarativo o enunciativo— del lenguaje.

2. Por otra parte, un enunciado puede tener más de un significado; la lengua natural está plagada de *ambigüedades léxicas, estructurales, de referencias cruzadas*, etc. No deseamos —ni podríamos— cambiar el lenguaje natural, pues gracias a estas propiedades el lenguaje natural es flexible, con él se puede desde contar chistes hasta hacer *filosofía de la tecnología*. Sin embargo, en lógica necesitamos un lenguaje riguroso, preciso, y habrá que solventar estos problemas creando un lenguaje artificial.

3. Los enunciados precisan ser contextualizados y así el mismo enunciado puede expresar distintas creencias al recibir *distintas contextualizaciones*.

4. En ocasiones no está claro qué pensamiento o creencia expresa una determinada oración; hay *expresiones engañosas*, incluso deliberadamente engañosas.

5. Hay enunciados paradójicos, contradictorios, a los que no puede asignárseles ni el valor verdadero ni el falso. El más antiguo que se conoce es la paradoja de Epiménides el cretense, quien decía que todos los cretenses son mentirosos y que todas sus afirmaciones son mentiras.

Comentario 13 *Introduciremos un lenguaje formal para eludir los problemas de ambigüedad e imprecisiones diversas que caracterizan a la lengua natural. En este lenguaje formal las paradojas serán evitadas; veremos que distinguiendo, como haremos, entre* ***lenguaje*** *y* ***metalenguaje*** *muchas de ellas no pueden reproducirse.*

Ejemplo 14 *Con frecuencia los chistes ocurren porque la frase contiene ambigüedades: léxicas, estructurales, de referencias cruzadas; así ocurre en los siguientes chistes:*

1. *Si nos encuentran, estamos perdidos. (Groucho)*
2. *En una panadería:* "Por favor, una barra de pan, y si tiene huevos, una docena". *(Sale con 12 barras de pan)*

Ejemplo 15 *En la mayor parte de las paradojas hay un problema de autorreferencia.*

[13] Que en el caso que nos ocupa se plantearía si no fue determinante la estructura de las lenguas europeas para el diseño final del lenguaje lógico.

1. *¿Qué sucede con los enunciados del recuadro?*[14]

Barcelona está en China
3+2=7
Hay tres errores en este recuadro

2. *Sócrates, en Troya, dice:* 'Lo que está ahora diciendo Platón en Atenas es falso'. *Platón en Atenas dice:* 'Lo que está ahora diciendo Sócrates en Troya es falso'.
¿Son consistentes los dos enunciados?

1.3.1. Tipos de enunciados

Los enunciados que expresan creencias pueden ser *satisfacibles* —*consistentes*— cuando la creencia expresada lo es; es decir, cuando es verdadera en alguna situación. (En el lenguaje formal que se introducirá después la palabra técnica empleada es *satisfacible* para la propiedad semántica, y *consistente* para la sintáctica de imposibilidad de derivarse una contradicción; evidentemente la una es la contrapartida de la otra.)

Por otra parte, un enunciado que no es verdadero en ninguna situación es *contradictorio.* Los enunciados que son verdaderos en cualquier situación son *tautologías* y los que son verdaderos en algunas situaciones y falsos en otras son *contingentes.*

Los enunciados capaces de describir una situación, y de distinguirla de otras, son contingentes. De esta clase son los enunciados que describen nuestra experiencia, que conforman la mayoría de las ciencias. Las tautologías, al ser verdaderas en toda situación, no pueden describir a ninguna en particular.

¿Describen algo? La respuesta es que sí, que *describen a la propia lógica.* Veremos que esta idea puede ser convenientemente explotada, ya que captar el funcionamiento y naturaleza de las tautologías es captar la esencia de la lógica.

Comentario 16 *Esta tipología se reproduce en el lenguaje formal y tendremos fórmulas* ***satisfacibles, contingentes, contradicciones*** *y* ***tautologías.***

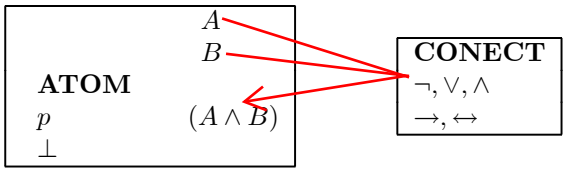
1.4. Lenguaje formal

Para obtener el rigor y precisión deseados, se introduce un *lenguaje formal (lógico).* Se tratará de un lenguaje artificial, con una reglas gramaticales explícitas que nos dicen qué sucesiones de signos del alfabeto son fórmulas y unas reglas semánticas también explícitas, que determinan cuando una fórmula es verdadera bajo una determinada interpretación —en un modelo matemático—. Dependiendo del nivel de abstracción que vayamos a necesitar, de la realidad a tratar y de la naturaleza de dicha realidad en estudio, hay diversos lenguajes posibles.

[14] Esta paradoja se la planteó George Boolos a Ulises Tindón, cuando tenía seis años y éste le dijo que se parecía a la del *Mentiroso. (¡mi niño!)*

En el siguiente capítulo introduciremos el lenguaje de la lógica de primer orden, en éste el de la proposicional, que tendrá las letras $p, q, r, ...$ etc como letras proposicionales; los signos $\bot, \top$ como constantes proposicionales y $, \neg, \wedge, \vee, \rightarrow$ y $\leftrightarrow$ como conectores. Las fórmulas de L_0 se construyen siguiendo unas sencillas reglas de formación, el conjunto formado por ellas —al que llamamos $FORM(L_0)$, o simplemente $FORM$, cuando esté claro por el contexto— es el menor conjunto que se puede generar con su ayuda a partir de sus letras.

- **F1** Las letras sentenciales son fórmulas. Como caso especial $\bot$ y $\top$ lo son.
- **F2** Si A y B son fórmulas, también lo son: $\neg A$, $(A \wedge B)$, $(A \vee B)$, $(A \rightarrow B)$ y $(A \leftrightarrow B)$



Comentario 17 *Adviértase que tal y como hemos definido el conjunto de fórmulas, como el menor conjunto que cumple las reglas* ***F1*** *y* ***F2,*** *si un conjunto* $\mathcal{Q}$ *obedece las mencionadas reglas, entonces* $FORM(L_0) \subseteq \mathcal{Q}$ *lo que significa que todas las fórmulas están en dicho conjunto. Es decir, en nuestra definición de fórmula está embebido un principio de inducción.*

Comentario 18 *El saber encontrar las sufórmulas de una fórmula dada es fundamental para manipular el cálculo deductivo correctamente. La forma más sencilla de presentarlo es mediante árboles genealógicos, que todo el mundo entiende con facilidad.*

1.4.1. Lenguaje y Metalenguaje

En el lenguaje natural utilizamos una serie de recursos para distinguir entre niveles de lenguaje

Ejemplo 19 *‹"'Un famoso poeta es menos inventor que descubridor', dijo Averroes", escribe Jorge Luis Borges›, destaca Deaño.*

Ejemplo 20 *‹Dice Hipólito en su obra* Refutatio omnium haereseum*: "la frase 'el bien y el mal son uno' fue escrita por Heráclito"›, asegura Deaño.*

Y también las comillas nos sirven para indicar cuando usamos o mencionamos una palabra; esto es, cuando nos referimos a un objeto extralingüístico o a la palabra misma.

Ejemplo 21 *Ponemos comillas para distinguir uso y mención.*
Salamanca está bañada por el Tormes
"Salamanca" tiene nueve letras.

Ejemplo 22 *Aquí, sin comillas, no se entiende nada:*
Madrid empieza por m,
termina con t
pero generalmente se escribe con g

Paradojas

Volvamos a la paradoja del mentiroso. La contradicción aparece cuando uno se pregunta sobre la propia afirmación de Epiménides.

¿Es también esta afirmación una mentira?

Una forma fácil de comprobarlo es la siguiente:

Sea p el enunciado: *"Estoy mintiendo"*. Naturalmente, esto es lo mismo que decir: *"No es verdad* p", que podríamos formalizar así: $\neg Verdad\,(p)$. Es decir,

$$p := \neg Verdad\,(p) \tag{1.1}$$

Pero la propiedad semántica de verdad debería ser definida de forma que para cualquier x,

$$x \text{ es verdadera si y sólo si } x$$

es decir,

$$\forall x(Verdad\,(x) \leftrightarrow x)$$

¿Qué sucede cuando consideramos la propia fórmula p*?*

En primer lugar,

$$Verdad\,(p) \leftrightarrow p \tag{1.2}$$

Ahora podemos usar las fórmulas (1.1) y (1.2), reemplazar en (1.2) la fórmula p por su formalización, obteniendo:

$$Verdad\,(p) \leftrightarrow \neg Verdad\,(p)$$

Naturalmente, esto es una contradicción.

Conclusión 23 *Nosotros distinguiremos entre* ***lenguaje*** *y* ***metalenguaje****, la fórmula* $\forall x(Verdad\,(x) \leftrightarrow x)$ *con el significado que se pretende que tenga no puede ser una fórmula del lenguaje objeto. La verdad de un enunciado se expresa en el metalenguaje, nunca en el lenguaje objeto*[15]*.*

[15] Esto no deja de ser una *verdad* a medias, pues en la lógica modal formalizamos el metalenguaje y en lógica de la reflexión también permitimos la autorreferencia. Pero la verdad de estas nuevas fórmulas se establece desde un nuevo nivel metalingüístico, o se crean mecanismos para evitar paradojas.

1.4.2. Interpretación de L_0

Interpretar un lenguaje proposicional es atribuir valores de verdad a sus fórmulas. La definición de este concepto será inductiva, basada en las asignaciones de valores a las letras. Una *asignación* es una función f que otorga un valor de verdad a cada letra proposicional —utilizo V y F, para *lo falso* y *lo verdadero*, respectivamente—

$$f : LS \longrightarrow \{V, F\}$$

Una *interpretación* es una función que da un valor de verdad a cada fórmula

$$\Im : FORM(L_0) \longrightarrow \{V, F\}$$

La definición se hará mediante recursión:

F1. Para letras proposicionales el valor es el de la asignación.

$$\Im(p) = f(p)$$

para $\bot$ y $\top$ tiene un valor fijo

$$\Im(\bot) = F \text{ e } \Im(\top) = V$$

F2. Para fórmulas con conectores se respeta el significado de los mismos:

1. $\Im(\neg C) = V$ syss $\Im(C) = F$
2. $\Im(C \wedge D) = V$ syss $\Im(C) = V$ y $\Im(D) = V$
3. $\Im(C \vee D) =$ syss $\Im(C) = V$ o $\Im(D) = V$
4. $\Im(C \rightarrow D) = V$ syss $\Im(C) = F$ o $\Im(D) = V$
5. $\Im(C \leftrightarrow D) = V$ syss $\Im(C) = \Im(D)$

1.5. Consecuencia lógica

Dijimos que tanto se podía caracterizar a la lógica como el estudio de los conjuntos consistentes de creencias, como el estudio de los razonamientos —o *argumentos*— válidos o correctos. Un argumento es un conjunto de sentencias tales que una de ellas —la *conclusión*— se sigue del resto —las *premisas* o hipótesis—. Lo típico es decir que la misión de la lógica es analizar los conceptos generales, patrones y procedimientos que se usan en los argumentos válidos, y que estos son, hasta cierto punto, independientes de los razonamientos concretos —puesto que aceptamos que hay infinitos razonamientos correctos que siguen el mismo esquema lógico—.

Llamamos relación de consecuencia a la que existe entre la hipótesis y la conclusión de un razonamiento correcto. Definiremos la consecuencia de una

sentencia A a partir de un conjunto de sentencias Γ —y escribiremos $\Gamma \models A$— así:

$$\Gamma \models A \text{ si y sólo si todo modelo de } \Gamma \text{ es también un modelo de } A$$

Por el momento podemos identificar un modelo con una situación particular, capaz de asignar un valor de verdad a cada sentencia.

¿Qué intuición queremos captar con este concepto?, ¿Cómo lo distinguimos de otros conceptos próximos?

El concepto intuitivo, que tendremos que precisar, es que un razonamiento es correcto cuando no se puede imaginar ninguna situación en la que las hipótesis del razonamiento sean verdaderas y la conclusión sea falsa[16]; esto es, cuando el conjunto formado por las hipótesis y la negación de la conclusión es insatisfacible, inconsistente.

Una forma sencilla de verlo es utilizar traducciones del lenguaje natural al formal y, desde éste, retrotraducciones al lenguaje natural. La idea es que si traducimos al lenguaje formal un razonamiento correcto y obtenemos un conjunto de hipótesis Γ y una conclusión A, no importa cómo retrotraduzcamos Γ y A al español; el resultado será siempre un razonamiento correcto. —Esto es, $p \wedge q \models p$ signifiquen lo que signifiquen p y q— Vamos a verlo con algunos ejemplos:

Ejemplo 24 ***(Picasso)*** *Considerad el siguiente argumento (falaz):*

- *Si Picasso nació en Málaga (p), entonces no es cierto que naciera en Francia (¬q).*
- *Picasso no nació en Francia*

  LUEGO
- *Picasso nació en Málaga.*

En este argumento todas las sentencias, tanto las de las hipótesis como la conclusión, son verdaderas, conforme a los hechos; Picasso nació en Málaga y Málaga está en España (que no es Francia, para nada). Pero el argumento no es correcto.

Ejemplo 25 ***(Retrotraducción)*** *Si el esquema lógico anterior fuera correcto; esto es, si*

$$\{(p \to \neg q),\ \neg q\} \models p$$

obtendríamos otro argumento correcto retrotraduciendo al español p y q. Usemos la siguiente:

- *Si Picasso nació en Londres (p), entonces no es cierto que naciera en Francia.(¬q)*

[16] De esta manera no se modeliza el concepto dinámico de prueba, sino el estático de resultado. Sin embargo, se complementa con un cálculo deductivo, que capta mejor el concepto de transformación, de ejecución.

- *Picasso no nació en Francia.*

  LUEGO

- *Picasso nació en Londres.*

  ¿Está claro porqué dudábamos del esquema argumental seguido?

Ejemplo 26 ***La obscuridad de la noche: Una prueba de la Teoría del Big Bang***
El gran descubrimiento de este siglo es que el universo no es inmóvil ni eterno, como supuso la mayoría de los científicos del pasado. El universo tiene una historia, no ha cesado de evolucionar, enrareciéndose, enfriándose, estructurándose. Esta evolución sucede desde un pasado distante que se sitúa, según las estimaciones, hace diez o quince mil millones de años, cuando el universo está completamente desorganizado, no posee galaxias, ni estrellas, ni moléculas, ni tan siquiera núcleos de átomos...Es lo que se ha llamado el BIG BANG. Una de las pruebas indirectas de esta teoría se puede plantear así:

- *Si* ***las estrellas fueran eternas*** (p)*, entonces* ***la cantidad de luz emitida sería infinita*** (q)*.*

- *Si la cantidad de luz emitida fuera infinita, entonces* ***el cielo debería ser extremadamente luminoso*** (r)*.*

- *El cielo es obscuro.*

  LUEGO

- *Las estrellas no existieron siempre.*

Las sentencias anteriores las formalizamos así:

$$(p \to q), (q \to r), \neg r,\ \neg p$$

Para expresar que la última es una consecuencia de las otras tres escribimos:

$$\{(p \to q),\ (q \to r),\ \neg r\} \models \neg p$$

Comentario 27 *En este caso el esquema argumental no levanta sospechas, otra cosa es si aceptáis como verdaderas en el mundo real las hipótesis. Obviamente, el determinarlo no es misión de la lógica. En el presente ejemplo lo sería de la Cosmología.*

Si el esquema anterior corresponde a un razonamiento correcto; es decir, si

$$\{(p \to q),\ (q \to r),\ \neg r\} \models \neg p$$

lo seguirá siendo cuando retrotraduzcamos al castellano p, q y r. Vamos a verlo con otro ejemplo.

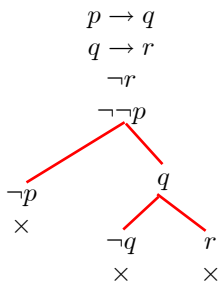
Ejemplo 28 ***(Retrotraducción) Lucrecio, filósofo romano; siglo I antes de Cristo.***
Lucrecio afirmaba que el universo aún estaba en su juventud. Razonó así: He comprobado desde mi infancia, se dijo, que las técnicas se han ido perfeccionando. Han mejorado el velamen de nuestros barcos, inventado armas más y más eficaces, fabricado instrumentos musicales más refinados...¡Si el universo fuera eterno, todos estos progresos habrían tenido tiempo de realizarse cien, mil, un millón de veces¡

- *Si el* ***universo fuera eterno*** (p)*, entonces* ***todos los progresos se habrían realizado ya*** (q).
- *Si todos los progresos se hubieran producido ya,* ***el mundo estaría acabado, no cambiaría*** (r).
- *El mundo cambia.*

  LUEGO

- *El mundo no existe desde siempre.*

Comentario 29 *En este caso el esquema argumental es el mismo, incluso es similar el tema. La lógica nos garantiza que este esquema, al corresponder a un razonamiento válido, seguirá produciéndolos al retrotraducir* p, q *y* r *y ni siquiera tienen que guardar relación con el tema del argumento original. Esto es, si aceptamos las hipótesis como creencias, debemos aceptar la conclusión. En una prueba mediante tableaux lo que hacemos es comprobar la imposibilidad de que se den simultáneamente las hipótesis y la negación de la conclusión.*



Razonamiento concluyente

En la vida cotidiana nuestros razonamientos versan, frecuentemente, sobre hechos: partimos de unas premisas o hipótesis, que pueden ser verdaderas o falsas, y llegamos a una conclusión, que también puede ser verdadera o falsa. Esto es, a diferencia del lógico no estamos aparentemente interesados en todas las realizaciones o modelos de las hipótesis de nuestros razonamientos, sino solamente en lo que acaece en la realidad, en un sólo modelo, o en una colección limitada

de modelos. Esto enmascara tanto los razonamientos válidos con hipótesis falsas como los razonamientos incorrectos con hipótesis y conclusiones verdaderas. Para situar el problema resulta útil la siguiente tabla de doble entrada:

Tipología de razonamientos correctos, clasificados por los valores de verdad de sus hipótesis y conclusión en la realidad

		Conclusión	
		Verdadera	Falsa
Hipótesis	Verdadera	1	2
	Falsa	3	4

Tipología de razonamientos incorrectos, clasificados por los valores de verdad de sus hipótesis y conclusión en la realidad

		Conclusión	
		Verdadera	Falsa
Hipótesis	Verdadera	5	6
	Falsa	7	8

El común de los mortales está interesado mayormente en los razonamientos de tipo 1, que son válidos pero además sus hipótesis son verdaderas, los llamaré *razonamientos concluyentes.* La racionalidad que como humanos se nos supone nos obliga, en principio, a aceptar las conclusiones de estos razonamientos entre nuestras creencias. Por supuesto, para adquirir nuevas creencias precisamos aceptar las conclusiones de los razonamientos cuyas hipótesis aceptamos como creencias; sin embargo, el contrastar dichas hipótesis cae fuera del alcance de la lógica. *¿Hay algo que la lógica pueda hacer al respecto?*

Razonamientos válidos con hipótesis compatibles

En lógica nos interesamos por los razonamientos válidos y estos pueden ser del tipo 1, 3 y 4. Razonamientos de tipo 2 no hay, porque justamente lo que caracteriza a un razonamiento válido es la imposibilidad de que su conclusión sea falsa cuando sus hipótesis son verdaderas. No nos interesa tanto el que la conclusión sea verdad como que el paso entre premisa y conclusión esté justificado.

Sin embargo, aún cuando desde el punto de vista lógico admitamos como válidos algunos razonamientos, nuestra aceptación de las conclusiones de un razonamiento no será la misma si sabemos que las hipótesis son incompatibles. De hecho, nos cuidaremos muy mucho de aceptar entre nuestras creencias un conjunto de hipótesis tal pues sabemos que de él se sigue como consecuencia lógica todo enunciado, que a su vez tendrá que ser admitido también.

Así que siempre que sea posible verificaremos la compatibilidad de nuestras hipótesis[17]; y aunque tal vez no esté en nuestra mano establecer su verdad en el mundo real, al menos sabremos si son consistentes.

Revisión de creencias

Hemos dicho que el principio general de racionalidad nos obliga a aceptar entre nuestras creencias a todas las conclusiones obtenidas mediante razonamientos concluyentes, a todas las consecuencias de nuestras creencias. Se supone que éstas han sido admitidas tras un proceso de evaluación racional. Sin embargo, hay conclusiones que por su inverosimilitud nos hacen revisar nuestras creencias. En los sistemas expertos se suelen implementar mecanismos para el *mantenimiento de la verdad*, diciéndose que la lógica usada es *no monotónica* porque al aumentar las hipótesis disminuyen, en vez de aumentar, las conclusiones. Es una forma de hablar, las hipótesis se reducen como resultado de la revisión de creencias y de ahí que también lo hagan las consecuencias.

1.5.1. Falacias

Los razonamientos incorrectos los descartamos; no garantizan la verdad de la conclusión, ni siquiera cuando sabemos que las hipótesis son verdaderas. Algunos razonamientos falaces los extraemos de la nutrida colección clásica: *Ad Baculum* (apelar a la fuerza), *ad hominem* (contra la persona), *ad populum* (usando en su favor los prejuicios del grupo), *ad verecundiam* (recurriendo al principio de autoridad), *petitio principii* (en círculo), *ignoratio elenchii* (cambiar de tema), etc.

Ejemplo 30 Ignoratio elenchii
"Salamanca es una ciudad muy provinciana"
"No, no es cierto. Salamanca tiene monumentos preciosos y tiene mucha marcha por las noches"

Comentario 31 *Aunque se pueda recurrir a los clásicos como fuente de ejemplos interesantes, no defiendo un planteamiento de* ***Lógica Informal*** *—se suelen limitar a presentar un catálogo de falacias— en un primer acercamiento a la disciplina, sino un planteamiento riguroso, pero con ejemplos bien preparados, interesantes, o al menos divertidos.*

Ejemplo 32 *Razonamiento concluyente.*
El razonamiento consignado es no sólo válido (o correcto), sino también concluyente.
Treinta días tiene Noviembre con Abril, Junio y Septiembre. Veintiocho tiene uno y los demás treinta y uno.
Por lo tanto,

[17] Puede ser inmediato si están expresadas en lógica proposicional, pero tal vez no sea factible en otros casos. Cuanto más potente es la teoría, más complicado es establecer su consistencia; por ejemplo, la consistencia de la *Teoría de Conjuntos* no está demostrada.

Abril tiene treinta días si y sólo si no los tiene Mayo, y si Mayo los tuviera, también los tendría Noviembre.

1.5.2. Definición de conceptos clave

Definición 33 *Una fórmula C es* ***satisfacible*** *syss hay una interpretación $\Im$ tal que $\Im(C) = V$. —Decimos que $\Im$ satisface a la fórmula C ; o también, que $\Im$* ***es modelo*** *de la fórmula C. Escribimos: $\Im \Vdash C$—*

Para conjuntos de fórmulas la definición es similar y la ***insatisfacibilidad*** *es la negación.*

Una fórmula C es ***contingente*** *syss hay tanto una interpretación $\Im$ tal que $\Im(C) = V$ como una interpretación $\Im^*$ tal que $\Im^*(C) = F$*

Definición 34 *Una fórmula C es* ***consecuencia*** *de un conjunto de fórmulas Γ —y escribimos $\Gamma \models C$— syss todo modelo de Γ lo es también de C*

Definición 35 *Una fórmula C es* ***válida*** *—y escribimos $\models C$— syss $\emptyset \models C$*

Definición 36 *Una fórmula C es* ***independiente*** *de un conjunto de fórmulas Γ —y escribimos $\Gamma \not\models C$— syss C no es consecuencia de Γ.*

Definición 37 *Un conjunto Δ de fórmulas es* ***independiente*** *syss para cada $C \in \Delta$ se cumple: $\Delta - \{C\} \not\models C$*

Definición 38 *Dos fórmulas C y D son* ***lógicamente equivalentes*** *si y sólo si*

$$C \models D \quad y \quad D \models C$$

Conforme a las definiciones precedentes las fórmulas se clasifican en satisfacibles e insatisfacibles y dentro de las segundas en válidas y contingentes, conforme al diagrama siguiente (ver figura: 1.2):

1.6. Tableaux semánticos

Para demostrar que nuestras fórmulas están relacionadas de alguna de las maneras arriba mencionadas podemos sistematizar el procedimiento de los tableaux que ya usábamos informalmente, de manera que sirvan para:

1. establecer la *satisfacibilidad* —en su defecto, la insatisfacibilidad— de una fórmula. Al acabar el tableau sabemos si la fórmula tiene o no algún modelo, y en el primer caso nos permite definirlo.

2. para establecer la *satisfacibilidad* —en su defecto, la insatisfacibilidad— de un conjunto finito de fórmulas.

3. para establecer la *validez* de una fórmula (se demuestra que su negación es insatisfacible)

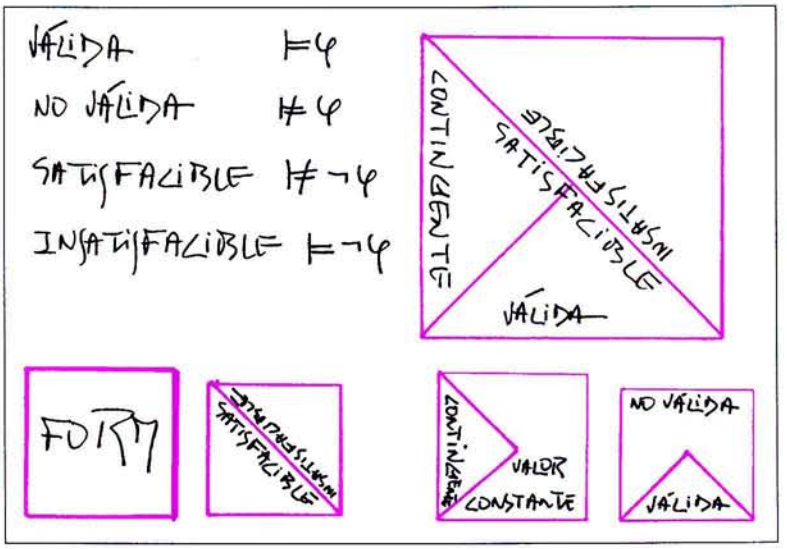


Figura 1.2: *Clasificación de fórmulas*

4. para demostrar *consecuencia* a partir de hipótesis (mostramos que el conjunto formado por las hipótesis y la negación de la conclusión es insatisfacible)

5. para demostrar la *independencia* de una fórmula respecto de un conjunto de fórmulas (vemos que no es consecuencia mostrando un contraejemplo)

Características más sobresalientes de los tableaux

1. consisten básicamente en el despliegue sistemático de las condiciones de verdad de la fórmula —o fórmulas— en estudio

2. tienen el aspecto de árboles —y por eso se les conoce también por ese nombre— cuyas ramas representan las distintas posibilidades; exhaustivas, no necesariamente excluyentes

3. las ramas se cierran cuando en ellas aparecen contradicciones explícitas, entendiéndose que se trata de una posibilidad frustrada. Un árbol completamente desarrollado y con todas las ramas cerradas muestra que la fórmula es insatisfacible

4. una rama abierta y completa permite definir una interpretación que satisface a la fórmula (o fórmulas) del árbol

5. es un procedimiento refutativo: se basa en que

$$\text{“}\Gamma \models C \quad \text{syss} \quad \Gamma \cup \{\neg C\} \text{ es insatisfacible”}$$

¿Qué son?

1. Un procedimiento semántico de búsqueda de un modelo que cumpla ciertos requisitos.
2. Un procedimiento sintáctico de prueba de teoremas

Ambas respuestas son acertadas: la primera permite un tratamiento más intuitivo y es la que usamos en principio, la segunda es evidente, se apreciará en cuanto los definamos.

No obstante, debemos demostrar las metapropiedades de corrección y completud para establecer la equivalencia entre los dos planteamientos. El que intuitivamente parezca convincente que los tableaux demuestran satisfacibilidad/insatisfacibilidad no garantiza por sí solo que sea así en efecto.

Ventajas (como cálculo deductivo)

1. son *'automáticos'* para la lógica proposicional; esto es, proporcionan un procedimiento de decisión que en un número finito de pasos nos dice si la fórmula es válida o no lo es.
2. pueden ser fácilmente implementados en el ordenador —aunque, a menudo, la eficiencia es pobre en comparación con otros sistemas de prueba—
3. son fácilmente generalizables a la lógica de primer orden[18] y a otras lógicas (modal[19], temporal, etc.)
4. su aprendizaje es extremadamente sencillo

Hay otra forma de entenderlos, que desde el punto de vista de la inteligencia artificial es impagable, y que no he visto documentado: como procedimiento de búsqueda de solución a un problema, pudiéndose establecer ciertos filtros. Esto lo explico con detalle en el apartado 1.6.3.

1.6.1. Definiciones

Sea A una fórmula proposicional. Hacemos un *tableau para* A empezando con A y aplicando las reglas de los *tableaux*. Las reglas se encargan de las fórmulas una por una, descomponiéndolas en otras más simples. Las reglas están diseñadas de tal manera que la fórmula *'input'* y las fórmulas *'output'* signifiquen lo mismo. La descomposición se termina cuando o bien se obtienen contradicciones explícitas —tales como B y $\neg B$, $\bot$ o $\neg\top$— o no se pueden aplicar más reglas. Si las reglas llevan en todos los casos a una contradicción, entonces A es contradictoria y concluimos que $\neg A$ es válida. De lo contrario, podemos extraer un modelo de A siguiendo los valores de la rama.

[18] Les dedico la sección 4.6, puede consultarse *Lógica para Principiantes* en

http : //logicae.usal.es

[19] Ver la sección 8.8.

Las reglas de los Tableaux Proposicionales

Hay reglas para cada conectiva y su negación, y una regla especial para cerrar una rama contradictoria.

- α-reglas (α = 'y'):

  1. De $A \wedge B$ se deduce A y B
  2. De $\neg(A \vee B)$ se deduce $\neg A$ y $\neg B$
  3. De $\neg(A \to B)$ se deduce A y $\neg B$
  4. De $\neg\neg A$ se deduce A

- β-reglas (β = 'ramificación'):

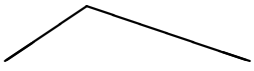
  1. De $A \vee B$ se deduce A y, *en una rama nueva separada,* B
  2. De $\neg(A \wedge B)$ se deduce $\neg A$ y, *en una rama nueva separada,* $\neg B$
  3. De $A \to B$ se deduce $\neg A$ y, *en una rama nueva separada,* B.
  4. De $A \leftrightarrow B$ deducimos A y B y, *en una nueva rama separada,* $\neg A$ y $\neg B$
  5. De $\neg(A \leftrightarrow B)$ deducimos A y $\neg B$ y, *en una nueva rama separada,* $\neg A$ y B

- Regla de cierre:
  Cerrar una rama que tenga A y $\neg A$ (para cualquier A), o $\neg\top$, o $\bot$.

Ejemplo 39 *Empezamos con* $A := \neg(((p \to q) \to p) \to p)$

1. $\neg(((p \to q) \to p) \to p)$

2. $(p \to q) \to p$ *3.* $\neg p$ — *α-regla de $\neg \ldots \to$ en 1*



4. $\neg(p \to q)$ — *6.* p *7.* $\neg q$ — *α-regla de $\neg \ldots \to$ en 4*

rama cerrada(3, 6)

5. p — *β-regla de $\to$ en 2*

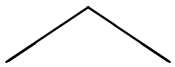
rama cerrada (3, 5)

Vemos que todas las ramas se cierran, por lo tanto este tableau ***está cerrado****.*

Ejemplo 40 *Empezamos con* $B := (p \vee \neg q) \wedge q$. *Esta vez no obtenemos ninguna contradicción.*

1. $(p \vee \neg q) \wedge q$

2. $p \vee \neg q$
3. q *por* α*-regla de* $\wedge$ *en 1*



4. p *5.* $\neg q$ β*-regla de* $\vee$ *en 2*

rama abierta **rama cerrada**

Comprobamos que no se pueden aplicar más reglas en la rama izquierda, el tableau ***no está cerrado****.*

Comentario 41 *Vimos en el ejemplo 40 que a una fórmula dada sólo se puede aplicar una regla —qué regla sea depende exclusivamente de la forma lógica de la fórmula; esto es, de si es una conjunción, o un condicional, etc. Esto es verdad para todas las reglas de los tableaux. La única cuestión aquí, que no es pequeña, es en qué orden tomamos las fórmulas para transformarlas; por lo tanto, en lógica proposicional los tableaux pueden implementarse determinísticamente en un ordenador, aunque la eficiencia pudiera ser pobre. Además, el proceso acaba necesariamente, pues las fórmulas resultantes tienen siempre longitud menor que las originales.*

Tableaux cerrados y teoremas

Definición 42 *Formalmente la deducibilidad se define así:*

1. *Una* ***rama*** *de un tableau es un subconjunto maximal lineal del tableau. (Los ejemplos deberían dejar claro lo que queremos decir.)*

2. *Una rama está* ***cerrada*** *si contiene* B *y* $\neg B$, *para la misma fórmula* B, *o si contiene* $\bot$ *o* $\neg\top$.

3. *Un tableau está* ***cerrado*** *si todas sus ramas están cerradas.*

4. *Si* A *es una fórmula, un* ***tableau para*** A *es un tableau que empieza con* A.

5. *Escribimos* $\vdash A$ *—se lee '*A *es demostrable', o '*A *es un teorema'— si existe un tableau cerrado para* $\neg A$.
   Notemos la $\neg$ *aquí. ¡Los tableaux prueban enunciados por contradicción.!*

6. *Una fórmula* A *es* ***consistente*** *si no hay un tableau cerrado para* A *(syss* $\nvdash \neg A$*).*

7. *Una fórmula A es* ***contradictoria*** *si no es consistente; es decir, si hay un tableau cerrado para A —syss $\vdash \neg A$—*

Conforme a lo dicho, mostramos en el ejemplo 39 que

$$\vdash ((p \to q) \to p) \to p$$

Extraer un modelo a partir de un tableau

Vimos en el ejemplo 40 un tableau para

$$A := (p \vee \neg q) \wedge q$$

con una rama abierta. Este tableau está *'completo'*; es decir, no se pueden aplicar más reglas. Podemos extraer un modelo $\Im$ de A a partir de una rama abierta: la rama tiene p y q, por lo tanto hacemos que $\Im(p) = \Im(q) = V$. Por lo tanto, como se puede comprobar fácilmente $\Im \Vdash A$.

1.6.2. Demostraciones a partir de hipótesis

Definición 43 *Sean A y B fórmulas. Escribimos $A \vdash B$ si hay un tableau cerrado que empieza con las dos fórmulas A y $\neg B$.*

Intuitivamente, $A \vdash B$ *'significa'* que podemos probar B si asumimos A como una hipótesis.

Ejemplo 44 $p \wedge (p \to q) \vdash q$

1. $p \wedge (p \to q)$
2. $\neg q$

3. p $\alpha 1$
4. $p \to q$ $\alpha 1$

5. $\neg p$ $\beta 4$ — *cerrado(3,5)*

6. q $\beta 4$ — *cerrado(2,6)*

1.6.3. Utilizar un tableau para encontrar soluciones

Con frecuencia la situación que se nos plantea no es tanto la de comprobar si un enunciado se sigue de un conjunto de hipótesis, sino más bien la siguiente: Dado un conjunto de hipótesis, queremos extraer conclusiones. En el caso de la lógica proposicional el árbol de las hipótesis nos ayuda a encontrarla. De hecho, para que sea más convincente, lo que hacemos primero es comprobar la compatibilidad de las hipótesis —pues en caso contrario cualquier conclusión es derivable—, para luego usar las ramas abiertas y establecer las coincidencias. Por supuesto, para que el conjunto de conclusiones tenga un tamaño manejable[20]

[20] Este conjunto es de hecho infinito, como puede demostrarse fácilmente, pues si A es una conclusión, también lo son: $A \wedge A$, $(A \wedge A) \wedge A$, $((A \wedge A) \wedge A) \wedge A$, etc

sólo nos interesamos por las fórmulas atómicas. Veámoslo con algún ejemplo concreto, sacado de los archivos de *MAFIA*.

Ejemplo 45 ***Robo de archivos***

Al llegar el Padrino a su despacho notó que alguien había entrado en él, ¡incluso había revuelto sus archivos¡. Pudo comprobar que faltaban algunos documentos comprometedores.
La investigación del caso arroja estos datos:

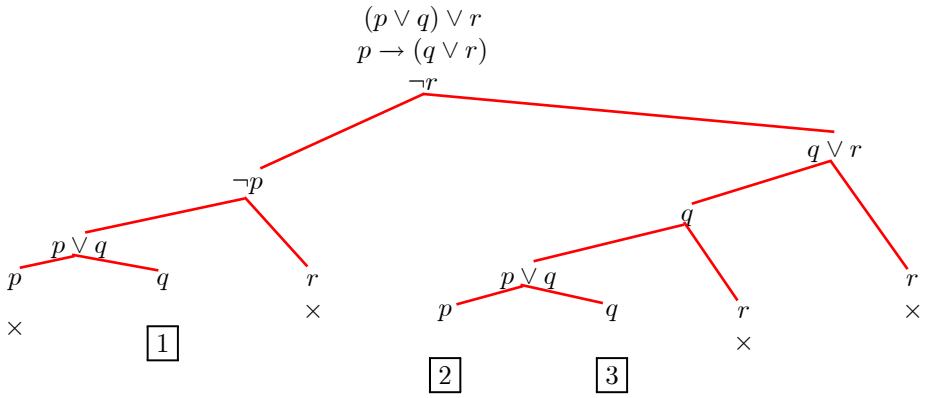
$A :=$ *Nadie más que* P, Q *y* R *están bajo sospecha y al menos uno es traidor.*
$B :=$ P *nunca trabaja sin llevar al menos un cómplice.*
$C :=$ R *es leal.*

1. *Formaliza los enunciados anteriores usando las claves siguientes:* p, q *y* r *que significan, respectivamente,* P *es un traidor,* Q *es un traidor y* R *es un traidor.*

2. *Comprueba si los datos son compatibles.*

3. *Extrae consecuencias de los datos y demuestra que son válidas.*

Solución:
1. La formalización es la siguiente
$A := (p \vee q) \vee r$
$B := p \rightarrow (q \vee r)$
$C := \neg r$
2. Para comprobar que son compatibles hacemos un árbol.



Hemos visto que $\{A, B, C\}$ es satisfacible pues hay tres interpretaciones que hacen a A, B y C simultáneamente verdaderas

$\boxed{1} = \{\neg p, \neg r, q\}$
$\boxed{2} = \{p, q, \neg r\}$

$\boxed{3} = \{q, \neg r\}$

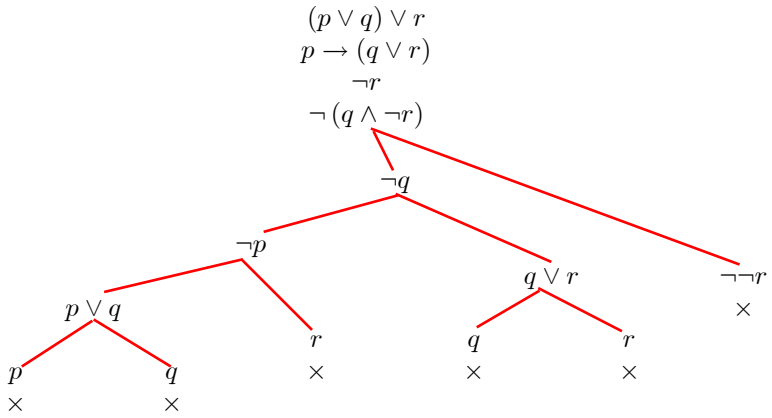
3. Para hallar la conclusión hacemos la intersección

$$\boxed{1} \cap \boxed{2} \cap \boxed{3} = \{q, \neg r\}$$

Ahora veremos que efectivamente

$$\{A, B, C\} \vDash q \wedge \neg r$$

Para demostrarlo hacemos el árbol de $\{A, B, C, \neg (q \wedge \neg r)\}$



1.7. Limitaciones de la lógica proposicional

Pese a su buen comportamiento como cálculo deductivo, al ser la capacidad expresiva de la lógica proposicional extraordinariamente limitada, no nos resulta útil en muchos casos.

Ejemplo 46 *Considerad el siguiente razonamiento:*
$A :=$ *Sólo los viejos y los niños dicen la verdad*
$B :=$ *María Manzano no es una vieja ni es una niña*
LUEGO:
$C :=$ *María Manzano miente*

En lógica proposicional A, B y C se formalizan como letras proposicionales —por ejemplo, p, q y r— y por lo tanto $\{p, q\} \nvDash r$. Sin embargo, el razonamiento es claramente correcto. En el lenguaje de primer orden *FOL* que introduciremos en el próximo capítulo se podría formalizar así:

Ejemplo 47 $A := \forall x(\neg Mx \rightarrow (Vx \vee Nx))$
$B := \neg Va \wedge \neg Na$
LUEGO:
$C := Ma$

En este lenguaje será fácil demostrar la validez del razonamiento.

La lógica de primer orden contiene a la proposicional; es decir, las fórmulas válidas de la proposicional siguen siéndolo en primer orden

$$VAL(PL) \subseteq VAL(FOL)$$

Pero es más potente; esto es,

$$VAL(PL) \subset VAL(FOL)$$

1.7.1. Lenguajes de orden cero, de primero y de segundo orden

En la lógica clásica hay varias categorías de lenguajes: proposicional, de primer orden, de segundo orden, etc. El de primer orden añade al proposicional la capacidad de analizar las fórmulas atómicas mediante relatores, functores y constantes y la cuantificación sobre individuos. El de segundo orden añade al anterior la facultad de cuantificar sobre conjuntos y relaciones.

¿Qué lenguaje necesitamos?

Depende de para qué, veámoslo con un ejemplo:

Ejemplo 48 *(Órdenes). Decimos que una relación* $\boldsymbol{R}$ *definida sobre un conjunto* $\boldsymbol{A}$ *es de orden, si es:*
$A :=$ *Reflexiva*
$B :=$ *Antisimétrica*
$C :=$ *Transitiva*
Cuando además es conectada,
$D :=$ *Conectada*
decimos que $\boldsymbol{R}$ *es un orden lineal.*
Cuando
$E :=$ *Todos los subconjuntos de* $\boldsymbol{A}$ *tienen primer elemento*
decimos que la relación $\boldsymbol{R}$ *es un buen orden.*

¿Qué lenguaje necesitamos para hablar de las relaciones de orden?

- Lenguaje proposicional es insuficiente. Con él podríamos establecer que si falla transitividad, la relación no es de orden

  $$\neg C \vdash \neg((A \wedge B) \wedge C)$$

  O que el lineal es una clase especial de orden

  $$(((A \wedge B) \wedge C) \wedge D) \vdash ((A \wedge B) \wedge C)$$

- En el lenguaje de primer orden con un relator binario R formalizamos:

$A := \forall x Rxx$
$B := \forall xy((Rxy \wedge Ryx) \rightarrow x = y)$
$C := \forall xyz((Rxy \wedge Ryz) \rightarrow Rxz)$
$D := \forall xy(Rxy \vee Ryx)$

La propiedad de ser un orden lineal es axiomatizable

En concreto, $\{A, B, C, D\}$ axiomatiza la propiedad de ser un orden lineal: una estructura $\mathcal{A}$ cualquiera es un orden lineal si y sólo si es un modelo de $\{A, B, C, D\}$.

Estas fórmulas son verdaderas en,

$$\langle \mathbb{N}, \leqslant \rangle, \langle \mathbb{Z}, \leqslant \rangle$$

y en

$$\langle \{\emptyset, \{1\}, \{1, 2\}\}, \subseteq \rangle$$

Cuando además del lenguaje de primer orden contamos con un cálculo deductivo:

- Usamos el cálculo para demostrar propiedades de los órdenes lineales
- Economía de recursos —Valdrán simultáneamente para todas las estructuras que sean órdenes lineales—

¿Se pueden expresar en primer orden todas las propiedades imaginables de las estructuras matemáticas?
¿Sirve la lógica de primer orden para axiomatizar toda la matemática?

La respuesta es que no. En nuestro caso, para expresar la propiedad de ser un buen orden se precisa de la cuantificación sobre propiedades; es decir, de la lógica de segundo orden[21] *SOL*. En *SOL* E se expresa:

$$E := \forall X(\exists y Xy \rightarrow \exists v(Xv \wedge \forall z(Xz \rightarrow Rvz \wedge v \neq z)))$$

Como hemos visto, el lenguaje de la lógica de segundo orden es más expresivo que el de primer orden y éste que el de orden cero. Sin embargo, las propiedades lógicas de estos lenguajes van decreciendo: mientras que la lógica proposicional posee un cálculo deductivo correcto, completo y es decidible, la de primer orden posee un cálculo correcto y completo, pero ya no es decidible, y la de segundo orden ni es decidible ni posee un cálculo completo.

Conclusión 49 *Una lógica es como una balanza (figura: 1.3): en un platillo se pone el poder expresivo de la lógica y en el otro las propiedades lógicas. En la lógica proposicional pesan más las propiedades lógicas, en la de segundo orden la capacidad expresiva, mientras que la de primer orden está más equilibrada. Sabiendo ésto somos nosotros los que decidiremos qué lógica necesitamos, qué virtudes nos interesa conservar.*

[21] La estudiamos con detalle en el capítulo 10.

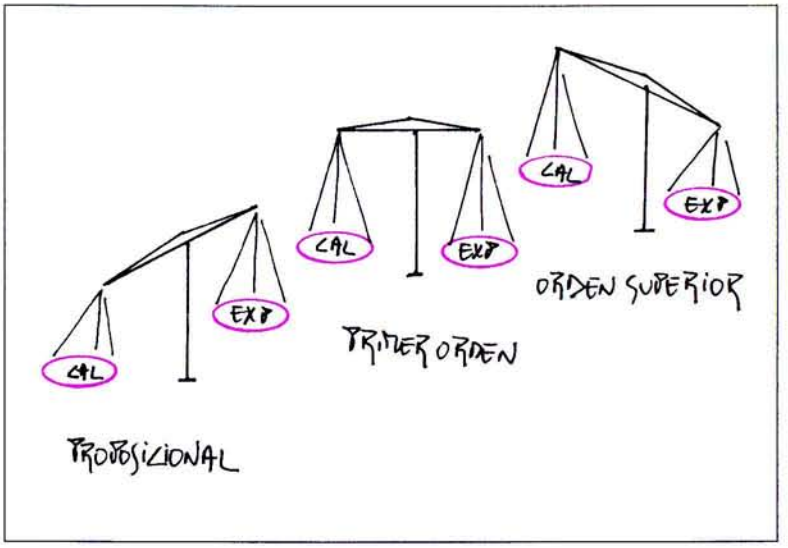


Figura 1.3: *Balanza*

1.8. Aplicaciones informáticas

Los siguientes proyectos de informática se hicieron bajo mi dirección, están disponibles en

$$http://aracne.usal.es$$

en donde se explican con detenimiento.

1. Para la enseñanza de la lógica básica, se hicieron los que siguen:
   a) *Calculus Ratiocinator.* [1999]. Jorge Hernández Martín.
   b) *Herramientas de aprendizaje.* [1999]. Rubén Simón Conde.
   c) *Tutor Inteligente de Lógica.* [1999]. José Carlos Blanco García.
2. Para aprender nociones elementales de razonamiento abstracto, dirigido a niños:
   a) *Los Enigmas de Ulises.* [1998]. Aitor Mata Conde.
3. Como manual lúdico-infantil de criptografía:
   a) *El detective Uielss.* [2001]. Nuria Iglesias Malmierca.
   b) *El idioma de los espías.* [2000]. Carlos Zapatel Vidorreta.
4. Para la enseñanza a distancia:

a) *Curso Virtual.* [2001]. Alberto Pérez Rodríguez.

b) *Biblioteca digital: Summa Logicae en el siglo XXI.* [2001]. Iván Marcos Poza.

5. Para la enseñanza del razonamiento con diagramas:

a) *Razonamiento lógico con diagramas de Venn.* [2001]. María Luisa Martín Martín.

b) *Diagramas Alfa de Peirce.* [2001]. Ignacio García Paredes.

c) *Tom's world.* [2000]. Tomás Rodríguez

6. Para la traducción de lógicas:

a) *Traductor de Lógicas: Modal a Multivariada.* 1999. Iván Marcos Poza.

b) *Traductor de Lógicas: Dinámica a Multivariada.* [2000]. María Iglesias Alonso.

c) *Traductor de Lógicas: Multivariada a Primer Orden sin variedades.* [1999]. José Escuadra Burrieza.

d) *Traductor de Lógicas: Modal de Primer Orden a Multivariada y Parcial.* [2001]. Raquel Caño Mateos.

Bibliografía

[1] Abramsky, S, Gabbay, D. y Maibaum, T. [1992-2000]. *Handbook of Logic in Computer Science.* vol 1 a 6. OUP. Oxford. U.K.

[2] Alchourrón, C y otros ed [1995]. *Lógica. Enciclopedia Iberoamericana de Filosofía.* Editorial Trotta. CSIC.

[3] Aracne [2000]. *Lógica para principiantes.* Universidad Nacional de Educación a Distancia. Madrid

[4] Badesa, C., Jané, I. y Jansana, R. [1998]. *Elementos de lógica formal.* Ariel Filosofía. Barcelona.

[5] Allwein, G.y Barwise, J. [1996]. *Logical Reasoning with Diagrams.* Oxford University Press. New York. USA.

[6] Beth, E.W. [1965] *Las Paradojas de la Lógica.* Ed. Castellana de Juan Manuel Lorente. Cuadernos Teorema 4. Valencia, 1978.

[7] Bergmann, M., Nelson, J. [1980]. *The Logic Book.* Random House, New York. USA

[8] Boolos, G. S & Jeffrey, R.C. [1989] *Computability and Logic.* (3rd Ed.). Cambridge University Press. Cambridge. U.K.

[9] Carroll, L. [1972]. *El juego de la lógica.* Alianza Editorial. Madrid

[10] Deaño, A. [1974]. *Introducción a la Lógica Formal.* Alianza Editorial, Madrid. España.

[11] Díaz Estévez, E. [1985]. *"Historia y Filosofía de la Lógica"* en [21].

[12] Barwise, J y Etchemendy, J. [2000]. *Language, proof and Logic.* CSLI. Stanford. Seven Bridges Press. New York. USA.

[13] Church, A. [1956]. *Introduction to Mathematical Logic.* vol I. Princeton: Princeton University Press.

[14] Falguera, J.L. y Martínez Vidal, C. [1999]. *Lógica Clásica de Primer Orden.* Editorial Trotta. Madrid.

[15] D. Gabbay [1994] *What is a Logical System?* Oxford University Press. Oxford U.K.

[16] Gabbay, D. Hogger, G. y Robinson, J. [1993-1998]. *Handbook of Logic in Artificial Intelligence and Logic Programming.* vol 1 a 6. OUP.

[17] Gabbay, D y Guenthner, F. [2001]. *Handbook of Philosophical Logic* 2^{nd} edition. Kluwer Academic Publishers. Dordrecht. Holanda.

[18] Garrido, M. [1974]. *Lógica Simbólica.* Tecnos. Madrid, 1981.

[19] Hodges, W. [1977]. *Logic.* Penguin Books, Middlexes. U.K.

[20] Manzano, M. [2003]. *Lógica para torpes.*Alianza Editorial. Madrid.

[21] Nepomuceno, A. ed. [1995]. *Lógica Formal. Orígenes, métodos y aplicaciones.* Kronos. Sevilla.

[22] Nepomuceno, A. [1995] *"Lógica Formal Elemental"*, en [21].

[23] Palau, G. [2001]. *"La noción abstracta de consecuencia lógica"* en http://logicae.usal.es

[24] Quesada, D. [1985]. *La lógica y su filosofía.* Editorial Barcanova. Barcelona.

[25] Robbin, J. W. [1969]. *Mathematical logic: a first course.* New York: W. A. Benjamin, INC.

[26] Rogers, R. [1971]. *Mathematical logic and formalized theories.* Amsterdam: North Holland.

[27] Sacristán, M. [1964]. *Introducción a la lógica y al análisis formal.* Ariel. Barcelona.

[28] Simpson [1989]. *Esentials of symbolic Logic.* Routledge. London.

[29] Smullyam, R. [1998]. *First-Order Logic.* Dover. Nueva York.

[30] Smullyan, R. [1977]. *What is the name of this book?* Prentice-Hall, Inc. Englewood Cliffs. New Yersey

Capítulo 2

Teoría de Modelos

2.1. Introducción

La teoría de modelos es la rama de la lógica matemática que se ocupa de las relaciones entre los lenguajes formales y sus interpretaciones en sistemas —o modelos— adecuados. El esquema abstracto de la teoría de modelos es así: Tenemos un lenguaje L y una clase de objetos $\mathfrak{K}$, que son los sistemas —también llamados estructuras—, y entre estos dos tipos de realidades tendemos un puente: la noción de verdad (ver figura: 2.1).

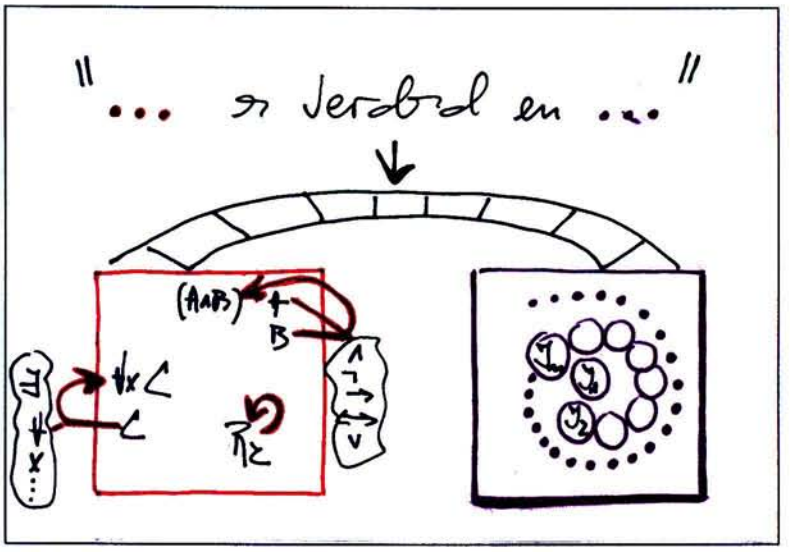


Figura 2.1: *La verdad es el Puente*

Este planteamiento, aparentemente tan simple, proporciona una gran flexi-

bilidad y alcance a la teoría de modelos.

El gran impulsor de las investigaciones en este área fue Tarski[1], que habiendo precisado y definido los conceptos semánticos de verdad y consecuencia, posibilitó esta modernización y generalización de la semántica que es la teoría de modelos. Aunque las raíces estaban echadas ya y algunos de los teoremas que ahora incluimos en ella —como el de Löwenheim-Skolem— habían sido demostrados tiempo ha, no se consolidó como disciplina independiente hasta los años cincuenta. El propio nombre de *Teoría de Modelos* fue utilizado por primera vez por Tarski en 1954. Los pioneros en el estudio de esta disciplina, aparte de los mencionados ya, fueron Gödel, Henkin, A. Robinson, Vaught, Craig y Addison, integrantes casi todos ellos —con excepción de Gödel y Robinson— del recién creado *Group in Logic and Methodology of Science* de Berkeley.

Comentario 50 *En un curso elemental de licenciatura nos ocuparemos exclusivamente de la teoría de modelos de la lógica de primer orden, pero compararemos los resultados obtenidos con los que se conseguirían cambiando dicho lenguaje por otro más potente, como el de segundo orden, o el de teoría de tipos —que forman parte de cursos distintos—. También abordaremos temas de teoría de modelos en los cursos dedicados a las lógicas no clásicas tales como la modal, la dinámica o la multivariada. Por razones pedagógicas he dejado fuera muchos temas que sólo podrían tener cabida en cursos de doctorado.*

2.2. Sistemas o estructuras y lenguaje

Empezaremos introduciendo la noción de sistema, definiendo a estos como un triplete formado por un conjunto no vacío —llamado universo— y una serie de individuos destacados, de funciones y de relaciones definidas sobre el universo del sistema[2]

$$\mathcal{A} = \langle \mathbf{A}, \langle \mathbf{c}_1, ..., \mathbf{c}_r \rangle, \langle \mathbf{f}_1, ..., \mathbf{f}_n \rangle, \langle \mathbf{R}_1, ..., \mathbf{R}_m \rangle \rangle$$

Algunos de estos tienen estructuras conocidas y estudiadas en matemáticas; por ejemplo, son grupos, anillos, órdenes o sistemas de Peano.

Una vez introducidos los sistemas, podemos, si así lo deseamos, estudiarlos sin utilizar el lenguaje formal de primer orden; entraremos entonces en el área conocida como *Álgebra Universal*. Aquí se estudia tanto a ellos mismos, como a ciertas relaciones de similitud entre ellos, tales como la de subsistema, extensión, homomorfismo y todas sus especificaciones —entre ellas, isomorfismo e inmersión—.

De ahora en adelante, mientras no se diga lo contrario, los sistemas que consideremos serán

$$\mathcal{A} = \left\langle \mathbf{A}, \langle \mathbf{f}_i \rangle_{i \in I}, \langle \mathbf{R}_j \rangle_{j \in J} \right\rangle \tag{2.1}$$

[1] En

$$http://logicae.usal.es$$

tenemos una traducción del artículo de Tarski: *La concepción semántica de la verdad y los fundamentos de la semántica.*

[2] Las definiciones de estos conceptos están en la sección 5.2.

$$\mathcal{B} = \left\langle \mathbf{B}, \langle \mathbf{g}_i \rangle_{i \in I}, \langle \mathbf{S}_j \rangle_{j \in J} \right\rangle \tag{2.2}$$

y ambos de tipo o signatura $\langle \mu, \delta \rangle$ —esto es, con el mismo número y grado de funciones y de relaciones, ordenadas y clasificadas por las funciones μ y δ—

Definición 51 *Decimos que* $\mathcal{A}$ *es un* ***subsistema*** *de* $\mathcal{B}$ *syss*

1. $\mathbf{A} \subseteq \mathbf{B}$
2. *Para cada* $i \in I$*: cada función* $\mathbf{f}_i$ *es la restricción de* $\mathbf{g}_i$ *al dominio de* $\mathcal{A}$*,* $\mathbf{f}_i = \mathbf{g}_i \mid \mathbf{A}^{\mu(i)}$*. Esto es, para cada* $\mathbf{x}_1, ..., \mathbf{x}_{\mu(i)} \in \mathbf{A}$

$$\mathbf{f}_i(\mathbf{x}_1, ..., \mathbf{x}_{\mu(i)}) = \mathbf{g}_i(\mathbf{x}_1, ..., \mathbf{x}_{\mu(i)})$$

   en especial, los individuos destacados, que son aquí funciones cero-arias, coinciden

3. *Para cada* $j \in J$*: cada relación,* $\mathbf{R}_j = \mathbf{S}_j \cap \mathbf{A}^{\delta(j)}$

Definición 52 *Una función* H *de* $\mathbf{A}$ *en* $\mathbf{B}$ *es un* ***homomorfismo*** *de* $\mathcal{A}$ *en* $\mathcal{B}$ *syss*

1. *Para cada* $i \in I$ *y cada* $\mathbf{x}_1, ..., \mathbf{x}_{\mu(i)} \in \mathbf{A}$

$$H\left(\mathbf{f}_i(\mathbf{x}_1, ..., \mathbf{x}_{\mu(i)})\right) = \mathbf{g}_i(H(\mathbf{x}_1), ..., H(\mathbf{x}_{\mu(i)}))$$

   en especial, para los individuos destacados,

$$H(\mathbf{f}_i) = \mathbf{g}_i$$

2. *Para cada* $j \in J$ *y cada* $\mathbf{x}_1, ..., \mathbf{x}_{\delta(j)} \in \mathbf{A}$

$$\textit{Si } \left\langle \mathbf{x}_1, ..., \mathbf{x}_{\delta(j)} \right\rangle \in \mathbf{R}_j \textit{ entonces } \left\langle H(\mathbf{x}_1), ..., H(\mathbf{x}_{\delta(j)}) \right\rangle \in \mathbf{S}_j$$

En la figura 2.2 se ve claramente cómo las operaciones de los sistemas se acoplan en un homomorfismo; da lo mismo operar a los elementos originales —$\mathbf{x}$ e $\mathbf{y}$— en el sistema $\mathcal{A}$, con las funciones de éste, que operar sus imágenes mediante el homomorfismo —$H(\mathbf{x})$ y $H(\mathbf{y})$— en el sistema $\mathcal{B}$, con las suyas.

Definición 53 *Una función* H *de* $\mathbf{A}$ *en* $\mathbf{B}$ *es una* ***inmersión*** *de* $\mathcal{A}$ *en* $\mathcal{B}$ *syss*

1. H *es inyectiva*

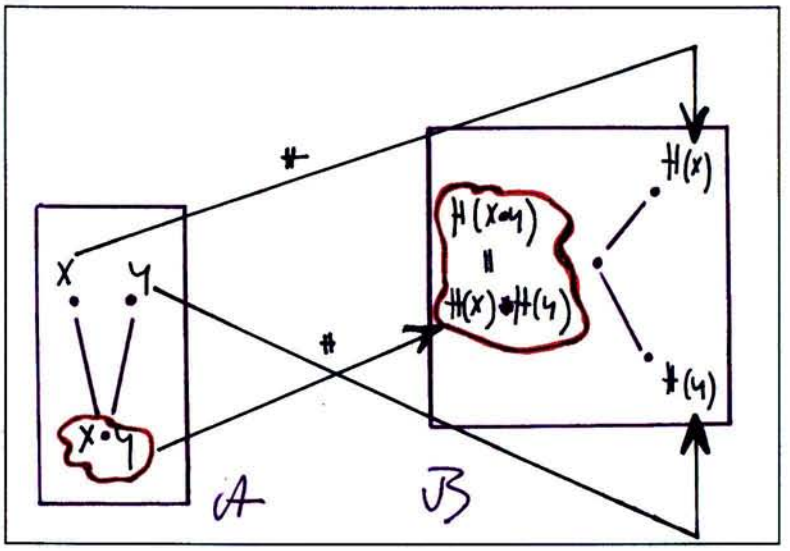


Figura 2.2: *Homomorfismo es acoplamiento*

2. *La condición para funciones es la misma que en el homomorfismo*

3. *Para cada* $j \in J$ *y cada* $\mathbf{x}_1, ..., \mathbf{x}_{\delta(j)} \in \mathbf{A}$

$$\left\langle \mathbf{x}_1, ..., \mathbf{x}_{\delta(j)} \right\rangle \in \mathbf{R}_j \quad syss \quad \left\langle H(\mathbf{x}_1), ..., H(\mathbf{x}_{\delta(j)}) \right\rangle \in \mathbf{S}_j$$

Definición 54 *Una función* H *de* $\mathbf{A}$ *en* $\mathbf{B}$ *es un* ***isomorfismo*** *de* $\mathcal{A}$ *en* $\mathcal{B}$ *syss*

1. H *es biyectiva, el resto de condiciones como en la inmersión.*

Notación 55 *Usamos:* $\cong$ *para isomorfismo,* $\sqsubseteq$ *para subsistema,* $\widetilde{\sqsubseteq}$ *para inmersión y* $\longrightarrow$ *para homomorfismo*

$$\begin{array}{ccccc} & & \mathcal{A} \widetilde{\sqsubseteq} \mathcal{B} & & \\ & \nearrow & & \searrow & \\ \mathcal{A} \cong \mathcal{B} & & \uparrow & & \mathcal{A} \longrightarrow \mathcal{B} \\ & & \mathcal{A} \sqsubseteq \mathcal{B} & & \end{array}$$

La relación existente entre ellas es la marcada en el esquema. La similitud entre estructuras decrece con el sentido de las flechas.

Comentario 56 *Isomorfía es identidad de estructuras (ver figura: 2.3); mientras que para ser totalmente idénticos los objetos que constituyen el universo deberían ser los mismos. Los sistemas* $\mathcal{A}$ *y* $\mathcal{B}$

$$\mathcal{A} = \langle \wp(\{1,2\}), \subseteq \rangle$$

—relación de ser un subconjunto en un conjunto de partes—

$$\mathcal{B} = \langle \{1,2,3,6\}, / \rangle$$

—relación " divide a " *en un conjunto dado—*
los represento mediante diagramas de Hasse; esto es, dos elementos están relacionados cuando hay una línea ascendente entre ellos.

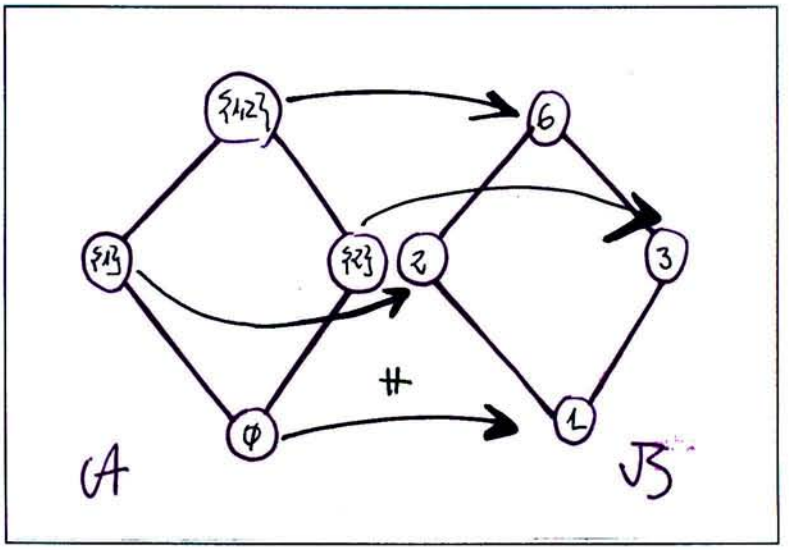


Figura 2.3: *Isomorfía es identidad de estructuras*

La línea divisoria entre álgebra universal y teoría de modelos es difusa. En el libro de Chang-Keisler, sus autores establecen la ecuación siguiente:

Teoría de Modelos = Álgebra Universal + Lógica

2.2.1. Lenguaje

Conforme a la definición precedente, si queremos penetrar en el santuario de la teoría de modelos, necesitamos un lenguaje formal: El que emplearemos en un curso introductorio será el de primer orden con igualdad; lo componen signos y filas de signos. Concretamente, en el alfabeto de nuestro lenguaje formal tendremos signos lógicos, variables, functores y relatores.

- Signos lógicos: $\neg, \wedge, \vee, \rightarrow, \leftrightarrow, \forall$ y $\exists$ e igualdad $=$
- Variables: $x, y, z, ...$
- Constantes individuales: $a, b, c, d, ... \in \overrightarrow{c}$
- Functores: $f_1, ..., f_n \in \overrightarrow{f}$ (pueden ser de diferentes ariedades)
- Relatores: $R_1, ..., R_m \in \overrightarrow{R}$ (pueden ser de distintos grados)

De forma simplificada indicamos que nuestro lenguaje de primer orden es:

$$L\ (\overrightarrow{R}, \overrightarrow{f}, \overrightarrow{c})$$

o sencillamente, L.

Junto a los signos del lenguaje formal tendremos que explicitar cuáles son las combinaciones lícitas de signos: cómo se forman los términos y fórmulas del lenguaje. Las reglas de formación de estas expresiones serán de naturaleza recursiva y proporcionarán un procedimiento de decisión para saber si una determinada sucesión de signos del alfabeto, es o no una fórmula o un término.

Definición 57 *El conjunto $TERM(L)$, de los términos de L es* ***el menor conjunto*** *que se puede generar mediante las reglas:*
T1. *Las variables individuales son términos.*
T2. *Las constantes individuales son términos.*
T3. *Si τ_1,..., τ_n son términos, $f^n\tau_1...\tau_n$ es un término.*

Definición 58 *El conjunto $FORM(L)$, de las fórmulas de L es* ***el menor conjunto*** *que se puede generar a partir de las reglas siguientes:*
F1. *Si τ_1,..., τ_n son términos, $R^n\tau_1...\tau_n$ es una fórmula (denominada* ***atómica****).*
F2. *Si τ_1 y τ_2 son términos, $\tau_1 = \tau_2$ es una fórmula (caso particular de fórmula atómica).*
F3. *Si A y B son fórmulas, también lo son: $\neg A$, $(A \wedge B)$, $(A \vee B)$, $(A \rightarrow B)$ y $(A \leftrightarrow B)$*
F4. *Si A es una fórmula, también lo son: $\forall x\ A$ y $\exists x\ A$.*

Definición 59 *Llamamos* ***expresiones de*** *L al conjunto formado por los términos y las fórmulas de L; es decir,*

$$EXPR(L) = TERM(L) \cup FORM(L)$$

Llamamos sentencias $SENT(L)$ al conjunto formado por las fórmulas que no contienen variables libres; esto es, no afectadas por ningún cuantificador.

Comentario 60 *Adviértase que tal y como hemos definido el conjunto de fórmulas, como el menor conjunto que cumple las reglas* ***F1*** *a* ***F4,*** *si un conjunto $\mathcal{Q}$ las cumple, entonces $FORM(L) \subseteq \mathcal{Q}$, lo que significa que todas las fórmulas están en dicho conjunto. De manera similar para términos. En realidad, nuestra definición de términos y fórmulas, como el menor conjunto de expresiones generadas mediante las reglas* ***T1-T3*** *(resp.* ***F1-F4****), lleva incluido un principio de inducción para términos (y fórmulas)*

Demostraciones por inducción semiótica para todos los términos

Si queremos demostrar que todos los términos tienen una cierta propiedad $\mathcal{T}$, tenemos que demostrarlo en dos pasos:

Básico: (1) Todas las variables individuales y todas las constantes tienen la propiedad $\mathcal{T}$.

Inductivo: (2) Si $\tau_1,...,\tau_n$ tienen la propiedad $\mathcal{T}$, entonces $f^n\tau_1...\tau_n$ tiene la propiedad $\mathcal{T}$.

Demostraciones por inducción semiótica para todas las fórmulas

Si queremos demostrar que todas las fórmulas tienen la propiedad $\mathcal{P}$, tenemos que demostrarlo en dos pasos:

Básico: (1) Todas las fórmulas atómicas tienen la propiedad $\mathcal{P}$.

Inductivo: (2) Si A y B tienen la propiedad $\mathcal{P}$, entonces: $\neg A$, $(A \wedge B)$, $(A \vee B)$, $(A \rightarrow B)$, $(A \leftrightarrow B)$, $\forall x\ A$ y $\exists x\ A$ tienen la propiedad $\mathcal{P}$.

Como he dicho, los objetos de nuestro lenguaje formal son signos y filas de signos. Por otra parte los sistemas —que son los objetos matemáticos de los que hablamos— están formados por conjuntos, relaciones y funciones. En semántica conectamos estos dos tipos de realidades (ver figura: 2.1).

2.3. Semántica

Siguiendo a Tarski, diremos que una sentencia C es verdadera en un sistema $\mathcal{A}$ —o lo que es lo mismo, que $\mathcal{A}$ es modelo de C (notación: $\mathcal{A} \Vdash C$)— si es realmente el caso que se dé C en $\mathcal{A}$.

La explicación tópica y el ejemplo paradigmático es:

"La sentencia 'La nieve es blanca' es verdadera si realmente la nieve es blanca".

Naturalmente, no lo diremos así, sino que precisaremos qué queremos decir al afirmar que se *"dé realmente el caso"*. Para definir el valor de verdad de una fórmula fijamos previamente la interpretación de los signos básicos que aparecen en ella. Sobre dicha interpretación nos fundamentamos para hacer que todos los términos del lenguaje denoten individuos del sistema, y que todas las fórmulas del lenguaje sean verdaderas o falsas en el mismo. Por supuesto, para poder asignar valores de verdad a las fórmulas sin variables libres necesitamos atribuirles elementos del universo: De esta forma una interpretación se define como un par ordenado,

$$\Im = \langle \mathcal{A}, H \rangle$$

siendo

$$H : Var \longrightarrow \mathbf{A}$$

La definición inductiva del valor de verdad de una fórmula la propuso Tarski, aunque ya era entendida y usada la noción antes de ser precisada por él, según dice Mostowski. La gestación del concepto de verdad en un sistema fue larga,

según nos hace notar Hodges[3]. En un primer momento se entendía bien el significado de

$$\textit{la fórmula } \forall x(Rx \to Rx) \quad \textit{es verdadera}$$

pero se tardó mucho más en entender y, sobre todo en definir[4] y precisar, el significado de

$$\textit{la fórmula } \forall xyx(fxfyz = ffxyz) \textit{ es verdadera en un grupo}$$

Esto es, distinguimos entre:

1. ser válida, y escribimos
$$\models \forall x(Rx \to Rx)$$
—que quiere decir, que para cada sistema $\mathcal{A}$ la sentencia es verdadera en él, formalmente: $\mathcal{A} \Vdash \forall x(Rx \to Rx)$, para cada $\mathcal{A}$—

2. ser verdadera en un cierto grupo $\mathcal{G}$
$$\mathcal{G} \Vdash \forall xyx(fxfyz = ffxyz)$$

Esto es, validez es verdad en todo sistema posible.

Resumen 61 *Para interpretar fórmulas necesitamos especificar:*

1. *dominio de cuantificación*
2. *cómo interpretamos las constantes, los functores y los relatores del lenguaje*
3. *Concepto fundamental: verdad en una estructura. (A partir de él se define el de consecuencia.)*

Estructuras de primer orden adecuadas a un lenguaje

Sea nuestro lenguaje de primer orden, $L(\overrightarrow{R}, \overrightarrow{f}, \overrightarrow{c})$

Definición 62 $\mathcal{A}$ *es una estructura adecuada para* $L(\overrightarrow{R}, \overrightarrow{f}, \overrightarrow{c})$ *syss*

$$\mathcal{A} = \left\langle \mathbf{A}, \overrightarrow{R}^{\mathcal{A}}, \overrightarrow{f}^{\mathcal{A}}, \overrightarrow{c}^{\mathcal{A}} \right\rangle$$

donde:

1. $\mathbf{A} \neq \emptyset$ *es el universo o dominio de la estructura.*
2. *Para cada relator n-ario* $R \in \overrightarrow{R}$ *su interpretación es:* $R^{\mathcal{A}} \subseteq \mathbf{A}^n$
3. *Para cada functor n-ario* $f \in \overrightarrow{f}$ *su interpretación es:* $f^{\mathcal{A}} : \mathbf{A}^n \longrightarrow \mathbf{A}$
4. *Para cada* $c \in \overrightarrow{c}$ *su interpretación es:* $c^{\mathcal{A}} \in \mathbf{A}$

[3] Véase el precioso artículo de Hodges, *"Truth in a structure"*

[4] No aparece hasta el artículo de Tarki y Vaugth de 1957, *"Arithmetical extensions of relational systems"*, publicado en el **JSL**

2.3.1. Interpretación de L

Las fórmulas de $L(\overrightarrow{R}, \overrightarrow{f}, \overrightarrow{c})$ se interpretan en una estructura

$$\mathcal{A} = \left\langle \mathbf{A}, \overrightarrow{R}^{\mathcal{A}}, \overrightarrow{f}^{\mathcal{A}}, \overrightarrow{c}^{\mathcal{A}} \right\rangle$$

- Los designadores de L denotan individuos de $\mathbf{A}$
- Las sentencias son verdaderas o falsas en $\mathcal{A}$

Pero para establecer el valor de verdad de una fórmula cualquiera necesitamos previamente asignar valores a las variables.

Definición 63 *Una **asignación** es una función F que otorga un elemento del universo a cada variable; es decir,*

$$F : VAR \longrightarrow \mathbf{A}$$

Definición 64 *Dada una asignación cualquiera F una variable x y un individuo del universo de la estructura $\mathbf{x}$ definimos $F_x^{\mathbf{x}}$ de la siguiente manera:*

$$F_x^{\mathbf{x}} = (F - \{\langle x, F(x)\rangle\}) \cup \{\langle x, \mathbf{x}\rangle\}$$

—esto es, asigna a todas las variables lo que F les asignara, pero a x la manda a $\mathbf{x}$, independientemente de su valor en F—

Definición 65 *Una interpretación $\Im = \langle \mathcal{A}, F\rangle$ es una función tal que*

$$\Im : EXPR(L) \longrightarrow \mathbf{A} \cup \{F, V\}$$

donde

$$\Im[TERM(L)] \subseteq \mathbf{A}$$

y

$$\Im[FORM(L)] = \{V, F\}$$

Para términos

1. ***T1.*** *Para cada variable individual x : $\Im(x) = F(x)$*
2. ***T2.*** *Para cada constante individual a : $\Im(a) = a^{\mathcal{A}}$*
3. ***T3.*** *Para cada término functorial $f\tau_1...\tau_n$*

$$\Im(f\tau_1...\tau_n) = f^{\mathcal{A}}(\Im(\tau_1), ..., \Im(\tau_n))$$

Para fórmulas

4. ***F1***. *Para cada fórmula atómica* $R\tau_1...\tau_n$

$$\Im(R\tau_1...\tau_n) = V \quad syss \quad \langle \Im(\tau_1), ..., \Im(\tau_n) \rangle \in R^{\mathcal{A}}$$

5. ***F2.*** *En especial, cuando es una igualdad,*

$$\Im(\tau_1 = \tau_2) = V \quad syss \quad \Im(\tau_1) = \Im(\tau_2)$$

6. ***F3.*** *Los conectores reciben la interpretación habitual*

7. ***F4.*** *Las fórmulas cuantificadas reciben la siguiente interpretación:*
Una generalización es verdadera cuando lo es para cada elemento del universo

$$\Im(\forall x\ C) = V \quad syss\ para\ cada \quad \mathbf{a} \in \mathbf{A} : \Im^{\mathbf{a}}_x(C) = V$$

Una particularización es verdadera cuando lo es para algún miembro del universo

$$(\exists x\ C) = V \quad syss\ existe\ un \quad \mathbf{a} \in \mathbf{A} \quad tal\ que \quad \Im^{\mathbf{a}}_x(C) = V$$

2.3.2. Conceptos clave

Definición 66 *Dada una interpretación* $\Im$ *tal que* $\Im(C) = V$ *decimos que* $\Im$ *satisface a la fórmula* C *; o también, que* $\Im$ ***es modelo*** *de la fórmula* C*. Usamos la notación* $\Im \Vdash C$

Definición 67 *Un conjunto de fórmulas* Γ *es* ***satisfacible*** *syss hay una interpretación* $\Im$ *tal que* $\Im(G) = V$ *para cada fórmula* $G \in \Gamma$ *—de igual forma cuando haya una sola fórmula—*

Definición 68 *Una fórmula* C *es* ***insatisfacible*** *syss no es satisfacible; es decir, no hay ninguna interpretación* $\Im$ *tal que* $\Im(C) = V$ *—De manera semejante, definimos* Γ *es* ***insatisfacible****—*

Definición 69 *Una fórmula* C *es* ***consecuencia*** *de un conjunto de fórmulas* Γ *—y escribimos* $\Gamma \models C$*— syss todo modelo de* Γ *lo es también de* C*; es decir, toda interpretación que hace verdadera a cada fórmula de* Γ*, hace verdadera a* C

Definición 70 *Una fórmula* C *es* ***válida*** *—y escribimos* $\models C$*— syss* $\emptyset \models C$*; es decir, toda interpretación hace verdadera a* C

Definición 71 *Una fórmula* C *es* ***independiente*** *de un conjunto de fórmulas* Γ *—y escribimos* $\Gamma \not\models C$ *— syss* C *no es consecuencia de* Γ *; es decir, hay modelos de* Γ *que no lo son de* C

Definición 72 *Dos fórmulas* C *y* D *son* ***lógicamente equivalentes*** *si y sólo si*

$$C \models D \quad y \quad D \models C$$

Definición 73 *Dado un conjunto de sentencias* Γ *al de sus consecuencias lo denotamos* $\mathbf{CON}(\Gamma)$

$$\mathbf{CON}(\Gamma) = \{C \in SENT(L) \mid \Gamma \vDash C\}$$

- *Dado un conjunto de sentencias* Γ *la clase de sus modelos la denotamos* $\mathbf{Mod}(\Gamma)$

$$\mathbf{Mod}(\Gamma) = \{\mathcal{A} \mid \mathcal{A} \Vdash C, \text{ para cada } C \in \Gamma\}$$

Comentario 74 *Un ejercicio muy fácil, pero interesante para tomar conciencia de la dificultad de una caracterización categórica de un sistema, es plantear un juego bidireccional entre sistemas y lenguaje: en un sentido, dado un* $\mathcal{A}$ *encontrar sentencias verdaderas en* $\mathcal{A}$*; en el otro, desde esas sentencias verdaderas en* $\mathcal{A}$ *hallar modelos* $\mathcal{B}$ *y comparar con* $\mathcal{A}$*.*

2.3.3. Definibilidad

También se introduce el concepto de relación y función definible en un sistema $\mathcal{A}$ con un lenguaje L

Definición 75 *Sea* $\mathbf{R}$ *una relación* $n-$*aria sobre el universo* $\mathbf{A}$ *del sistema* $\mathcal{A}$ *—esto es,* $\mathbf{R} \subseteq \mathbf{A}^n$*—. Decimos que* $\mathbf{R}$ es definible en $\mathcal{A}$ con L *syss hay una fórmula* C *de* L *con a lo sumo* $x_1, ..., x_n$ *libres, tal que:*

$$\mathbf{R} = \{\langle \mathbf{x}_1, ..., \mathbf{x}_n \rangle \in \mathbf{A}_n \mid \mathcal{A}\ [\mathbf{x}_1/x_1 ... \mathbf{x}_n/x_n] \Vdash C\}$$

Notación 76 *Cuando sabemos que las variables libres de una fórmula están en el conjunto* $\{x_1, ..., x_n\}$ *y tomamos una interpretación* H *en la que les asignamos los valores* $\mathbf{x}_1, ..., \mathbf{x}_n \in \mathbf{A}$ *entonces en vez de*

$$\Im = \langle \mathcal{A}, H \rangle$$

escribimos

$$\mathcal{A}[\mathbf{x}_1/x_1 ... \mathbf{x}_n/x_n]$$

En verdad el valor que H *asigne a las variables ligadas es irrelevante, esto se suele demostrar en primer orden como metateorema sintáctico*[5]*.*

Las relaciones definibles —en un sistema $\mathcal{A}$ con un lenguaje L— están a medio camino entre las relaciones $\mathbf{R}_1, ..., \mathbf{R}_m$ presentes en la estructura

$$\mathcal{A} = \langle \mathbf{A}, \langle \mathbf{c}_1, ..., \mathbf{c}_r \rangle, \langle \mathbf{f}_1, ..., \mathbf{f}_n \rangle, \langle \mathbf{R}_1, ..., \mathbf{R}_m \rangle \rangle$$

que tienen *nombre propio* en el lenguaje L y las que simplemente existen en virtud de la propiedad escasamente descriptiva de *ser un subconjunto de* $\mathbf{A}$[6].

[5] Se le suele denominar *principio de coincidencia*; la demostración detallada de dicho metateorema puede consultarse en [21], página 94.

[6] Este hecho, tiene enormes repercusiones en la lógica de segundo orden: cuando restringimos el universo de cuantificación a los subconjuntos y relaciones definibles del universo de individuos, obtenemos un teorema de completud, como veremos con detalle en la sección 10.6.

Está claro que con un lenguaje numerable en una estructura de universo infinito numerable; por ejemplo, en la de los números naturales,

$$\mathcal{N} = \langle \mathbb{N}, 0, s, +, \cdot, \leq \rangle$$

la mayoría de los subconjuntos de $\mathbb{N}$ no son definibles ya que $\wp(\mathbb{N})$ es supernumerable —su cardinalidad es $\aleph_1$— y sólo tenemos un conjunto numerable de fórmulas para definirlos —esto es, el conjunto de fórmulas es de cardinalidad $\aleph_0$—. Esto hace que pese a que la *inducción aritmética* sea un esquema axiomático en primer orden; esto es, una colección infinita de axiomas de la forma

$$B\left(\frac{c}{x}\right) \wedge \forall x(B \rightarrow B\left(\frac{\sigma x}{x}\right)) \rightarrow \forall x\ B$$

tenga menos poder expresivo que la fórmula de segundo orden correspondiente

$$\forall X(Xc \wedge \forall x(Xx \rightarrow X\sigma x) \rightarrow \forall x\ Xx)$$

porque en el primer caso sólo podemos referirnos a los subconjuntos definibles mediante fórmulas y en el segundo a todos. La verdad es que esto no presentaría mayor problema si el conjunto que nos interesara fuera definible en cualquier modelo $\mathcal{M}$ de la teoría de los naturales, pero no es así. La razón es que el conjunto de los denominados *números estándar*

$$\mathbb{N}(\mathcal{M}) = \{\mathcal{M}(c), \mathcal{M}(\sigma c), \mathcal{M}(\sigma\sigma c), ...\}$$

no es definible en una estructura cualquiera, como se verá luego, pudiéndose incluso demostrar el siguiente teorema:

Teorema 77 *Sea $\mathcal{A}$ un modelo de primer orden de $\mathbf{AP}^1$. Entonces $\mathcal{A}$ es estándar syss el conjunto de los números estándar $\mathbb{N}(\mathcal{A})$ es definible en $\mathcal{A}$.*

Donde

$$\mathbf{AP}^1 = \left\{ \begin{array}{c} \forall x\ c \neq \sigma x \\ \forall xy(\sigma x = \sigma y \rightarrow x = y) \\ \forall x\ x + c = x \\ \forall xy\ x + \sigma y = \sigma(x + y) \\ \forall x\ x \cdot c = c \\ \forall xy\ x \cdot \sigma y = (x \cdot y) + x \end{array} \right\} \cup \{Induc(B) \mid B \in \mathbf{FORM}(L)\}$$

Siendo

$$Induc(B) := B\,(c) \wedge \forall x(B(x) \rightarrow B(\sigma x)) \rightarrow \forall x\ Bx$$

Se podría pensar que, pese a lo dicho, la mayor parte de las relaciones que nos interesan son definibles porque nuestra descripción intuitiva de las mismas puede fácilmente convertirse en definición. Además, es un hecho que todas las relaciones decidibles son definibles. Sin embargo, no es así: hay descripciones que no se pueden plasmar en definiciones.

Un ejemplo de conjunto no definible es el de los números de Gödel de las sentencias verdaderas en $\mathcal{N}$[7].

Otro ejemplo más simple de relación no definible, que pasamos a exponer a continuación, nos lo sugiere la prueba diagonal de Cantor.

[7] Se verá con detalle en la sección 3.7 del próximo capítulo.

Ejemplo 78 *Sean* $\mathbf{X}_1, ..., \mathbf{X}_n, ...$ *los subconjuntos definibles de* $\mathbb{N}$ *y sea*

$$\mathbf{Y} = \{n \mid n \notin \mathbf{X}_n\}$$

Obviamente $\mathbf{Y}$ *no puede ser definible, pues si lo fuera* $\mathbf{Y} = \mathbf{X}_m$ *para algún* m*, de donde se seguiría la contradicción*

$$m \in \mathbf{Y} \text{ syss } m \notin \mathbf{X}_m \ (= \mathbf{Y})$$

Las relaciones sobre los naturales definibles en $\mathbb{N}$ se denominan relaciones aritméticas[8] y un problema que se plantea con frecuencia es el de determinar en qué medida lo son porque aunque todas sean definibles, unas *lo son más que otras*[9].

Hay una gran asimetría entre mostrar que una relación es definible y probar que no lo es. En el primer caso basta con producir la fórmula que la defina; en el segundo usamos un corolario del *Teorema de isomorfía* que afirma que los automorfismos mantienen dentro de la relación a todos sus elementos, si esta es definible (ver figura: 2.4).

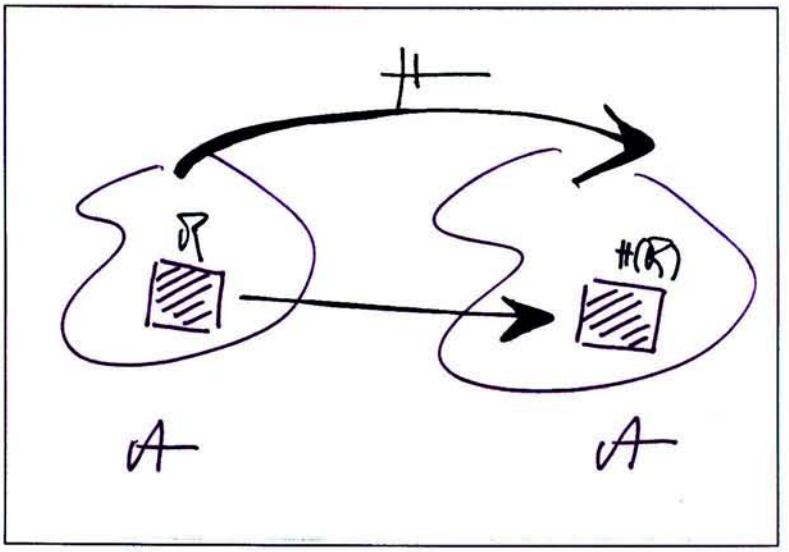


Figura 2.4: *Definibilidad e Isomorfía*

Teorema 79 ***(de isomorfía)*** *Sean* $\mathcal{A}$ *y* $\mathcal{B}$ *dos sistemas isomorfos,* $\mathcal{A} \cong \mathcal{B}$ *y sea* H *el isomorfismo. Para cada* $C \in FORM(L)$, $\mathbf{x}_1, ..., \mathbf{x}_n \in \mathbf{A}$

$$\mathcal{A}[\mathbf{x}_1/x_1 ... \mathbf{x}_n/x_n] \Vdash C \text{ syss } \mathcal{B}[H(\mathbf{x}_1)/x_1 ... H(\mathbf{x}_n)/x_n] \Vdash C$$

[8] Se verán en la sección 3.9 del capítulo 3.
[9] Como la igualdad en la *Granja* de Orwell.

Corolario 80 *Si* H *es un automorfismo sobre* $\mathcal{A}$ *y* $\mathbf{R}$ *es definible en* $\mathcal{A}$*, entonces, para cada* $\mathbf{x}_1, ..., \mathbf{x}_n \in \mathbf{A}$

$$\langle \mathbf{x}_1, ..., \mathbf{x}_n \rangle \in \mathbf{R} \quad syss \quad \langle H(\mathbf{x}_1), ..., H(\mathbf{x}_n) \rangle \in \mathbf{R}$$

(Se ve de forma más gráfica en la figura 2.4.)

Corolario 81 *El conjunto de los números naturales* $\mathbb{N}$ *no es definible en*

$$\mathcal{R}_< = \langle \mathbb{R}, < \rangle$$

Demostración. *Supongamos que lo fuera, y que la fórmula* C *lo definiera*

$$\mathbb{N} = \{\mathbf{x} \in \mathbb{R} \mid \mathcal{R}\,[\mathbf{x}] \Vdash C\}$$

Un automorfismo sobre $R_<$ *es una función creciente; es decir, tal que*

$$si \quad x < y \quad entonces \quad H\,(\mathbf{x}) < H\,(y)\,.$$

La función así definida $H\,(\mathbf{x}) = x^3$ *lo es. De acuerdo con el corolario anterior debería cumplirse*

$$\mathbf{x} \in \mathbb{N} \quad syss \quad H\,(\mathbf{x}) \in \mathbb{N}$$

Pero $5 = \left(\sqrt[3]{5}\right)^3 \in \mathbb{N}$ *mientras* $\sqrt[3]{5} \notin \mathbb{N}$*. Esto es, la función* H *manda a* $\mathbb{N}$ *elementos de* $\mathbb{R}$ *que no son de* $\mathbb{N}$*.* ■

2.3.4. Relación entre sistemas usando L

Cuando ya contamos con el lenguaje formal de primer orden, podemos hablar en él sobre clases de sistemas, y podemos también relacionar a los mismos en función de lo que de ellos podemos decir en el lenguaje de primer orden —que no lo es todo—. Nacen así las relaciones entre sistemas de: *equivalencia elemental, subsistema elemental e inmersión elemental.*

Sean $\mathcal{A}$ y $\mathcal{B}$ sistemas similares.

Definición 82 *Si* $\mathcal{A}$ *y* $\mathcal{B}$ *son del mismo tipo y* L *es el lenguaje adecuado para hablar de ambos, decimos que* $\mathcal{A}$ *es* ***elementalmente equivalente*** *a* $\mathcal{B}$ *si para cada sentencia* C *:*

$$Si \quad \mathcal{A} \quad es\ modelo\ de \quad C \quad entonces \quad \mathcal{B} \quad es\ modelo\ de \quad C$$

Definición 83 *Si* $\mathcal{A}$ *y* $\mathcal{B}$ *son del mismo tipo y* L *es el lenguaje adecuado para hablar de ambos, decimos que* $\mathcal{A}$ *es* ***subsistema elemental*** *de* $\mathcal{B}$ *si:*

1. $\mathcal{A} \sqsubseteq \mathcal{B}$*; es decir, es un subsistema suyo*
2. *Para cada asignación* F *y para cada fórmula* C *:*

$$\Im = \langle \mathcal{A}, F \rangle \quad es\ modelo\ de \quad C \quad syss \quad \Im^* = \langle \mathcal{B}, F \rangle \quad es\ modelo\ de \quad C$$

Definición 84 *Si* $\mathcal{A}$ *y* $\mathcal{B}$ *son del mismo tipo y* L *es el lenguaje adecuado para hablar de ambos, decimos que* H *es una* ***inmersión elemental*** *de* $\mathcal{A}$ *en* $\mathcal{B}$ *syss:*

1. H *es una inmersión de* $\mathcal{A}$ *en* $\mathcal{B}$
2. *Para cada cada fórmula* $C\langle x_1, ..., x_n\rangle$ *y cada* $\mathbf{x}_1, ..., \mathbf{x}_n \in \mathbf{A}$

$$\mathcal{A}[\mathbf{x}_1, ..., \mathbf{x}_n] \Vdash C \quad \textit{syss} \quad \mathcal{B}[\mathbf{x}_1, ..., \mathbf{x}_n] \Vdash C$$

Comentario 85 *De manera que para relacionar sistemas existen fundamentalmente dos modos:*

1. *Al margen del lenguaje formal*
2. *A través de él*
   Como idea general vale la siguiente: las relaciones entre sistemas a veces no pueden ser captadas en toda su profundidad por el lenguaje de primer orden. Por ejemplo, mientras que si dos sistemas son isomorfos, también son elementalmente equivalentes, no vale el recíproco. Es decir, hay sistemas indistinguibles en el lenguaje de primer orden —esto es, que satisfacen las mismas sentencias— que no son iguales, ni tan siquiera isomorfos. La dependencia entre ellas es la marcada en el esquema siguiente.

Notación 86 *Usamos:* $\equiv$ *para equivalencia elemental,* $\prec$ *para subsistema elemental y* $\widetilde{\prec}$ *para inmersión elemental.*

$$\begin{array}{ccccc} & & \mathcal{A}\widetilde{\prec}\mathcal{B} & & \\ & \nearrow & & \searrow & \\ \mathcal{A} \cong \mathcal{B} & & \uparrow & & \mathcal{A} \equiv \mathcal{B} \\ & & \mathcal{A} \prec \mathcal{B} & & \end{array}$$

La demostración de que isomorfía implica equivalencia elemental es sencilla, pero *¿son isomorfos todos los sistemas elementalmente equivalentes?* Demostraremos que no en el teorema 106, al definir un sistema no isomorfo pero elementalmente equivalente a $\mathcal{N}$.

¿Qué teoremas demostramos en teoría de modelos?

Los teoremas de completud, compacidad y Löwenheim-Skolem son bastante característicos y de ellos se derivan consecuencias de mucho calado, como a continuación veremos.

2.4. Completud y algunas de sus consecuencias

2.4.1. Completud del cálculo

Hemos visto que para hablar de un sistema o estructura —o de una clase de ellos— es conveniente introducir un lenguaje lógico adecuado, cuyas fórmulas nos sirvan para describir a las entidades matemáticas que estudiamos. Cada

interpretación selecciona del conjunto de todas las fórmulas a las sentencias verdaderas en ella (ver figura: 2.5), que constituyen lo que más adelante denominaremos *teoría de una estructura*, y que en general son distintas para cada una.

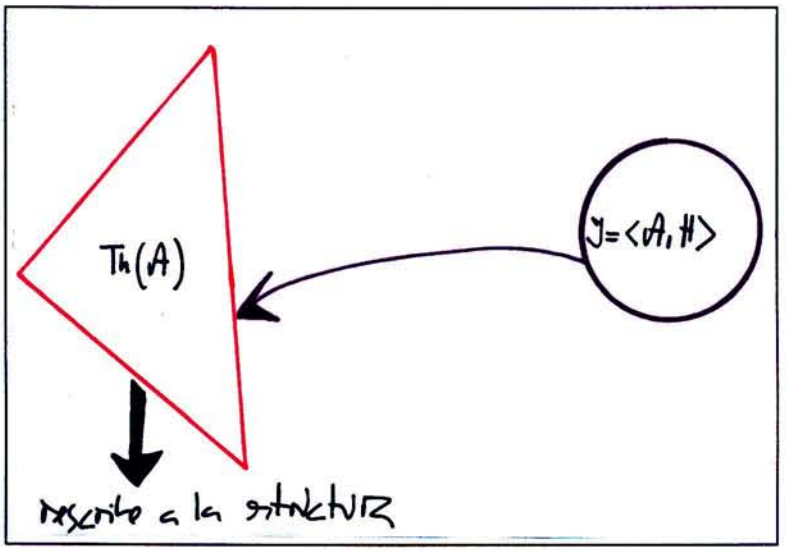


Figura 2.5: *Cada interpretación selecciona una teoría*

Sin embargo, todas las teorías tienen un núcleo común (ver figura: 2.6), el de las fórmulas válidas, *VAL*.

Por ser verdaderas en toda estructura estas sentencias no describen ninguna en particular, sino aquello que es común a todas ellas.

¿Caracterizan algo?

La respuesta es que sí, que describen a la propia lógica. Por consiguiente, si logramos generarlas con facilidad habremos captado la esencia de la lógica, *su perfume* —en el sentido bárbaro y radical de Süskind—. Sin embargo aunque contamos con la noción semántica de verdad es frecuentemente muy difícil el establecerla apelando simplemente a las condiciones de la definición. Mucho más difícil todavía lo es el determinar si una fórmula es consecuencia de un conjunto de ellas, para lo que en principio sería necesario una comprobación sistema a sistema.

Por fortuna hay otro modo de establecer el valor de verdad de una fórmula y de determinar si es consecuencia de otras que no es la mera verificación directa de sus especificaciones semánticas, se trata de inferir o deducir la fórmula en un cálculo deductivo utilizando las otras fórmulas como hipótesis; esto es, de establecer una cadena de razonamiento entre premisas y conclusión. De hecho, esta forma de definir el concepto de consecuencia es incluso más adecuada, ya que refleja el carácter discursivo del razonamiento. Si el cálculo deductivo nos

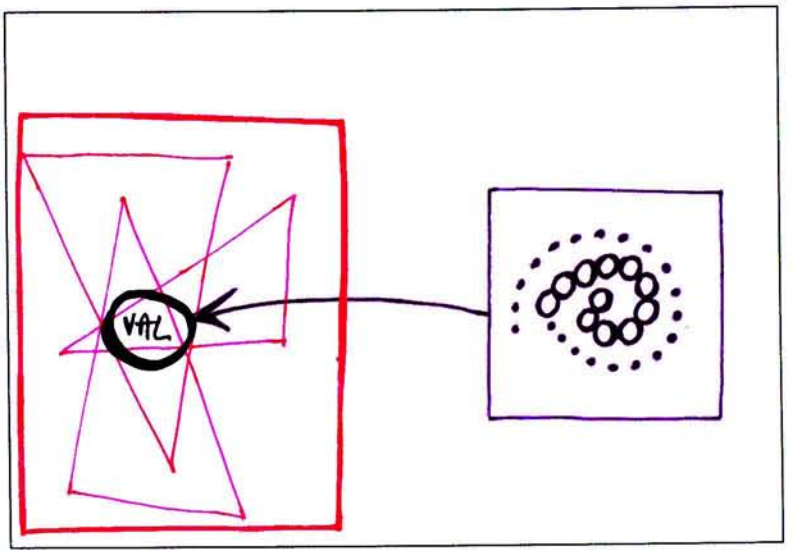


Figura 2.6: *Núcleo común*

va a resultar útil es porque nos ayudará a no equivocarnos; no nos conducirá de hipótesis verdaderas a conclusiones falsas: será un *cálculo correcto*. Además sus reglas permitirán obtener como teoremas a todas las consecuencias de un conjunto dado de hipótesis; esto es, será de aplicabilidad general, lo que denominamos un *cálculo completo*[10].

El problema de la completud lo plantearon inicialmente Post y Hilbert, pero no fue resuelto hasta 1930 por Gödel. La prueba que a continuación comentaré es la de Henkin de 1949. La ventaja de ésta es que es más sencilla y versátil que la original de Gödel, prestándose a extensiones y modificaciones para adaptarse a otros sistemas lógicos, incluso a lenguajes de orden superior[11].

El teorema de completud, junto con el de corrección, establecen la equivalencia entre la noción sintáctica y la semántica de consecuencia, para un cierto lenguaje. Podemos plantear la cuestión así: la noción semántica de verdad sirve para seleccionar del conjunto de todas las sentencias de un cierto lenguaje, a las que son verdaderas en todos los sistemas —a las que llamamos fórmulas lógicamente válidas, *VAL*—. Por otra parte, a nuestro lenguaje formal —de naturaleza puramente sintáctica— podemos incorporarle un cálculo deductivo. Dicho cálculo permitirá deducir unas fórmulas de otras, y nos servirá para generar el conjunto de las sentencias del lenguaje que se pueden deducir sin premisas

[10] Hay quienes prefieren llamar a esto *suficiencia* y reservar el vocablo *completud* para teorías. Prefiero no tomar este tipo de decisiones, que aunque pudieran estar motivadas terminan generando aún más caos terminológico.

[11] Es curioso que la prueba de Henkin de completud de la teoría de tipos —para modelos generales—, precediera a su prueba de completud del cálculo de primer orden. Véase la prueba de completud de la lógica de segundo orden en la sección 10.6.

en el cálculo, a las que llamamos teoremas lógicos *(TEO)* (ver figura: 2.7)

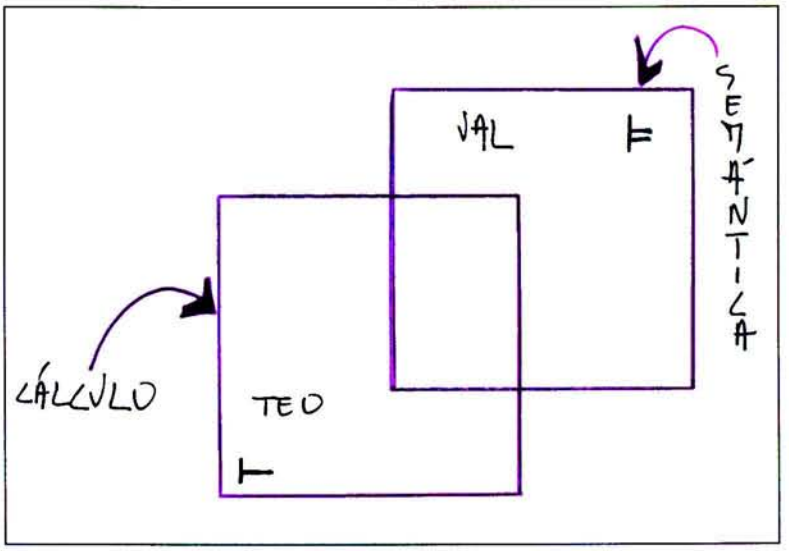


Figura 2.7: *Dos métodos de selección*

¿Coinciden esos conjuntos?

Demostrar que $VAL \subseteq TEO$ es el objetivo del teorema de completud, que $TEO \subseteq VAL$ lo es del de correción. Para cálculos correctos y completos el diagrama aparece así (ver figura: 2.8):

Cálculo deductivo

El cálculo deductivo que introduzco a continuación es *de secuentes,* pero no es el original de Gentzen[12], sino una modificación; es el que utilizo en mi libro de *Teoría de Modelos.* La razón principal de haber escogido uno de esta clase es que la prueba de correción es inmediata, a diferencia de lo que sucede en los cálculos de deducción natural, mientras que las demostraciones en él son relativamente sencillas, a diferencia de lo que sucede en los axiomáticos[13].

1. Regla de introducción de hipótesis *(IH)*

   $\dfrac{}{\Gamma \dashv A}$, siempre que $A \in \Gamma$

2. Regla de monotonía *(M)*

   $\dfrac{\Gamma \dashv A}{\Gamma^* \dashv A}$, siempre que $\Gamma \subseteq \Gamma^*$

[12] Que se puede consultar en la sección 4.5

[13] De todo esto hablaremos extensamente en el capítulo 4.

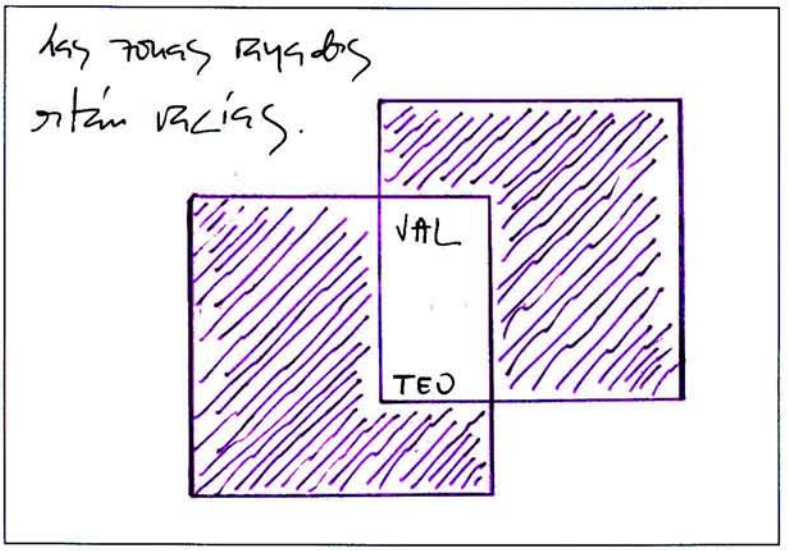


Figura 2.8: *Cálculo correcto y completo*

3. Regla de prueba por casos *(PC)*

$$\frac{\Gamma\ A \dashv B \quad \Gamma\ \neg A \dashv B}{\Gamma\ \dashv B}$$

4. Regla de no contradicción *(NC)*

$$\frac{\Gamma\ \neg A \dashv B \quad \Gamma\ \neg A \dashv \neg B}{\Gamma\ \dashv A}$$

5. Regla de introducción de la disyunción en el antecedente *(IDA)*

$$\frac{\Gamma\ A \dashv B \quad \Gamma\ C \dashv B}{\Gamma\ A \vee C \dashv B}$$

6. Reglas de introducción de la disyunción en la conclusión *(IDC)*

$$\frac{\Gamma \dashv A}{\Gamma \dashv A \vee C} \text{ y } \frac{\Gamma \dashv C}{\Gamma \dashv A \vee C}$$

7. Regla de introducción del particularizador en el antecedente *(IPA)*

$$\frac{\Gamma, A\left(\frac{x}{y}\right) \dashv B}{\Gamma, \exists x\ A \dashv B}, \text{ si la variable no está libre en ninguna de las fórmulas}$$

8. Regla de introducción del particularizador en la conclusión *(IPC)*

$$\frac{\Gamma \dashv A\left(\frac{\tau}{x}\right)}{\Gamma \dashv \exists x\ A}$$

9. Regla de reflexividad de la identidad *(RI)*

$$\frac{}{\dashv \tau = \tau}$$

10. Regla de sustitución de iguales *(SI)*

$$\frac{\Gamma \dashv A\left(\frac{\tau}{y}\right)}{\Gamma\ \tau = \tau^* \dashv A\left(\frac{\tau^*}{y}\right)}$$

Estas reglas están definidas para un lenguaje que tiene tan sólo disyunción, negación y particularización como conectores básicos y particularización como cuantificador, puede extenderse con reglas derivadas[14] de manera que las haya para cada conector y cuantificador, o convertir las fórmulas a las que utilizan este lenguaje económico.

Al crear un cálculo deductivo lo que se pretende es que sus reglas sean capaces de generar a todas las fórmulas lógicamente válidas (VAL), pero sólo a ellas. Los teoremas de completud ($VAL \subseteq TEO$) y de corrección ($TEO \subseteq VAL$) nos aseguran que el objetivo se ha cubierto, y que el conjunto de los teoremas lógicos coincide con el de las fórmulas lógicamente válidas.

De hecho, nosotros demostraremos algo más fuerte —el denominado teorema de completud fuerte, o teorema de completud para consecuencia— que establece que siempre que una fórmula sea consecuencia de un conjunto de fórmulas —conjunto que puede ser infinito—, también será demostrable a partir de ellas con las reglas del cálculo.

Teorema 87 *(completud fuerte)* $\Gamma \vDash A \implies \Gamma \vdash A$

Teorema 88 *(corrección)* $\Gamma \vDash A \impliedby \Gamma \vdash A$

Teorema 89 *(equivalencia)* $\Gamma \vDash A \iff \Gamma \vdash A$

Teorema 90 *(completud débil)* $\vDash A \implies \vdash A$

Teorema 91 *(corrección)* $\vDash A \impliedby \vdash A$

Teorema 92 *(equivalencia)* $\vDash A \iff \vdash A$

Son parte importante de la demostración los siguientes resultados:

Teorema 93 *(de Henkin): Si* $\Gamma \subseteq FORM(L)$ *es consistente, entonces tiene un modelo de universo numerable.*

[14] Puede consultarse [21], páginas 110 a 114.

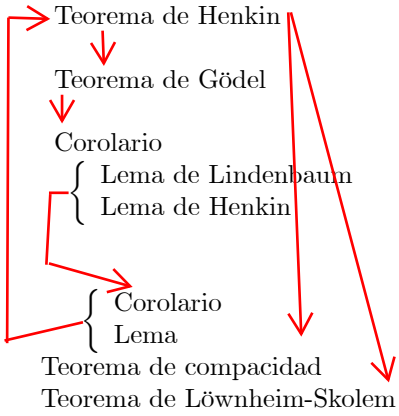
Lema 94 *(de Lindenbaum): Si* $\Gamma \subseteq FORM(L)$ *es consistente y tiene sólo un conjunto finito de variables libres, entonces hay un* Γ^* *máximamente consistente y ejemplificado tal que* $\Gamma \subseteq \Gamma^* \subseteq FORM(L)$

Lema 95 *(Henkin): Si* Γ^* *es máximamente consistente y ejemplificado, entonces tiene un modelo de universo numerable.*

Corolario 96 *Si* $\Gamma \subseteq FORM(L)$ *es consistente y tiene sólo un conjunto finito de variables libres, entonces* Γ *tiene un modelo de universo numerable.*

Lema 97 *Si* $\Gamma \subseteq FORM(L)$ *es consistente y si* $\overline{\Gamma}$ *es un conjunto de sentencias que resulta de sustituir en las fórmulas de* Γ *las variables libres por constantes nuevas y si* $\overline{\Gamma}$ *tiene un modelo de universo numerable, entonces* Γ *también.*

Antes de demostrar el teorema de completud es importante entender la estrategia de demostración (ver esquema), cómo se articulan los diversos lemas que lo componen. Lo primero que hacemos es comprobar que el teorema de Henkin es condición suficiente del de completud de Gödel, evidenciándose que lo realmente importante es demostrar que si un conjunto es consistente, entonces tiene un modelo y consecuentemente, centrar el interés en la construcción del mismo. Visto esto, no es difícil entender el resto del entramado de la prueba ya que el teorema de Henkin se sigue del corolario en cuanto encontremos la manera de prescindir de la condición de que las variables libres constituyan un conjunto finito[15]. A su vez, el corolario se sigue de los lemas de Lindenbaum y del de Henkin de forma obvia.



Teorema 98 *Teorema de Henkin* $\Longrightarrow$ *Teorema de Gödel*

Demostración. *Si* $\Gamma \vDash A$ *entonces* $\Gamma \cup \{\neg A\}$ *es insatisfacible. Entonces* $\Gamma \cup \{\neg A\}$ *es inconsistente, por el teorema de Henkin. En este caso es fácil demostrar que* $\Gamma \vdash A$ *—pues sabemos que* $\Gamma \cup \{\neg A\} \vdash A$*; usaremos las reglas de IH y de PC—* ■

[15] Esta complicación es necesaria cuando se quiere que el teorema valga no sólo para sentencias, sino también para fórmulas abiertas. Así lo demuestro en mi libro de *Teoría de Modelos*, páginas 120 a 137.

Para demostrar el lema de Lindenbaum lo que hago es ordenar la fórmulas del lenguaje y construir inductivamente una cadena infinita de conjuntos consistentes y ejemplificados; algo fundamental, pues sobre dicha ordenación se basa la construcción de la cadena cuya gran unión es máximamente consistente; están todos los posibles, los que no lo convierten en contradictorio y el test de consistencia se va haciendo fórmula a fórmula.

Teorema 99 *(Lindenbaum). Todo conjunto consistente* Γ *cuyo conjunto de variables libres sea finito, puede extenderse a uno máximamente consistente y ejemplificado* Δ

Demostración. *Sea* Γ *un conjunto consistente y sea* $B_1, ..., B_n, ...$ *una enumeración de* $FORM(L)$

Definimos:

$\Delta_0 = \Gamma$

$$\Delta_{n+1} = \begin{cases} \Delta_n, \text{ si } \Delta_n \cup \{B_n\} \text{ es contradictorio} \\ \\ \Delta_n \cup B_n \text{ si es consistente y } B_n \text{ no es una particularización} \\ \\ \Delta_n \cup \left\{B_n, C\left(\frac{y}{x}\right)\right\}, \text{ si es consistente, pero una particularización y la variable es nueva} \end{cases}$$

Hagamos $\Delta = \bigcup_{n \geq 0} \Delta_n$

Utilizando la construcción vemos que:

1. $\Gamma \subseteq \Delta$
2. Δ_n *es consistente, para cada* $n \geq 0$
3. Δ *es máximamente consistente; esto es, para cada* $B \in FORM(L) : B \in \Delta$ *o* $\neg B \in \Delta$ *(no ambos)*
4. Δ *es ejemplificado* ■

A continuación se demuestra el lema de Henkin, que dice que todo conjunto máximamente consistente tiene un modelo. La idea es justamente la de construir el que las fórmulas están describiendo y el *"quid"* de la cuestión está en que los conjuntos máximamente consistentes lo hacen con grandísimo detalle, proporcionándonos algo así como las tablas de las funciones y relaciones del modelo.

¿Qué individuos constituyen el universo?

Sabemos que la naturaleza de los objetos que forman el universo de una estructura es irrelevante —esta filosofía se desprende del teorema de isomorfía, pero en un sentido más general es lo que caracteriza a la lógica, que se ocupa de la *forma*, no tanto del contenido—, por lo que no planteamos ninguna objeción a tomar como individuos a los propios términos del lenguaje. En una primera aproximación construimos

$$\mathcal{B} = \left\langle TERM(L), \langle \mathbf{g}_i \rangle_{i \in I}, \langle \mathbf{S}_j \rangle_{j \in J} \right\rangle$$

donde:

1. Su universo es el conjunto de los términos del lenguaje
2. Para cada functor del lenguaje $f_i : \mathbf{g}_i(\tau_1, ..., \tau_{\mu(i)}) = f_i\tau_1...\tau_{\mu(i)}$, para cada $\tau_1, ..., \tau_{\mu(i)} \in TERM(L)$
3. Para cada relator del lenguaje R_j :

$$\langle \tau_1, ..., \tau_{\mu(i)} \rangle \in \mathbf{S}_j \text{ syss } R_j\tau_1...\tau_{\mu(i)} \in \Delta$$

para cada $\tau_1, ..., \tau_{\mu(i)} \in TERM(L)$.

Este sistema serviría para un lenguaje sin igualdad, pero si la hubiera, habría que hacerla coincidir con lo que al respecto estipule nuestro oráculo Δ. Para ello se establece una relación de equivalencia —la de ser *"iguales a los ojos de* Δ*"*— y en vez de tomar como universo el de los términos, tomamos el cociente; esto es, los elementos no son términos sino clases de términos.

Lo que queda por ver de esta prueba son *pequeñas* comprobaciones, normalmente nada cortas e inductivas.

Comentario 100 *Puesto que el cálculo es de naturaleza finita —es decir , es una sucesión finita de líneas obtenidas conforme a ciertas reglas—, la completud fuerte nos informa de que el problema es siempre reducible a un conjunto finito de hipótesis —las que de hecho se han usado—. Esta simple observación está en la base de la demostración del teorema de compacidad como corolario de completud.*

2.4.2. Teoremas de Löwenheim-Skolem

Estos teoremas hablan del tamaño de los modelos. El más antiguo es el de Löwenheim (1915) que dice que si una sentencia tiene un modelo infinito, entonces tendrá uno numerable.

Teorema 101 *Si* B *tiene un modelo, entonces* B *tiene un modelo numerable*

Nosotros podemos obtener versiones más potentes de este teorema; a saber, su extensión a conjuntos cualesquiera de fórmulas

Teorema 102 *Si* Γ *tiene un modelo, entonces* Γ *tiene un modelo numerable*

y también las versiones *upward* y *downward*

Teorema 103 (downward) *Si* Γ *escrita en un lenguaje de cardinalidad* κ *tiene un modelo , entonces* Γ *tiene un modelo de cardinalidad* $\leq \kappa$

Teorema 104 (upward) *Si* Γ *escrita en un lenguaje de cardinalidad* κ *tiene un modelo infinito, entonces* Γ *tiene un modelo de cardinalidad* λ*, para cada* $\lambda \geq \kappa$

Paradoja de Skolem

Todos estos teoremas nos informan de la incapacidad de la lógica de primer orden para distinguir entre cardinalidades infinitas; por ejemplo la de los naturales y la de los reales. Al considerar este hecho surge de inmediato la llamada *paradoja de Skolem*: Utilizando un lenguaje numerable se puede formalizar una teoría, que incluye a los números reales, en la que la sentencia que afirma que los reales no son numerables es un teorema. Un modelo cualquiera de dicha teoría satisfará la mencionada sentencia. No obstante, por Löwenheim-Skolem sabemos que si dicha teoría tiene un modelo de cualquier cardinalidad, tendrá también uno numerable. Aquí está lo sorprendente: que la sentencia que afirma que los reales no son numerables pueda ser verdadera en un modelo de universo numerable. La paradoja no llega a ser contradicción porque, aunque la sentencia que afirma que los reales no son numerables sea verdadera en un modelo numerable, lo único que implica es que en dicho modelo no hay ninguna función biyectiva de los reales en los naturales —pues esto es lo que significa *ser numerable*—. Los elementos del modelo que representen a los reales pueden ser numerables; la función biyectiva estará fuera del modelo.

2.4.3. Teorema de compacidad

Utilizando la condición de finitud de la deducibilidad, a partir del teorema de completud demostramos el de compacidad como un corolario sencillo. En verdad, algo más es cierto

$$\text{Completud fuerte} \iff \left\{ \begin{array}{c} \text{Completud Débil} \\ + \\ \text{Compacidad} \end{array} \right\}$$

El teorema de compacidad afirma que un conjunto Δ de sentencias tiene un modelo si y sólo si cada subconjunto finito de Δ lo tiene.

Teorema 105 *Δ tiene un modelo syss para cada $\Gamma \subseteq \Delta$ tal que Γ es finito, entonces Γ tiene un modelo.*

Fue inicialmente demostrado por Gödel (1930) como corolario del teorema de completud. Cabe destacar que el enunciado del teorema de compacidad es de naturaleza puramente semántica, y uno tiene la intuición de que puede ser resuelto sin apelar a la noción de deducibilidad: combinando de algún modo los modelos de los conjuntos finitos para construir el modelo del infinito. Esta intuición es correcta y, de hecho, se puede demostrar compacidad utilizando la noción booleana de ultrafiltro —filtro maximal—, y construyendo como modelo del conjunto infinito el ultraproducto.

Nosotros demostramos primero el teorema de completud siguiendo la prueba de Henkin —es decir, construyendo un modelo a partir de constantes— y demostraremos compacidad como corolario de completud. También, para familiarizarnos con otras construcciones de modelos, de un carácter genuinamente

algebraico, demostraremos el teorema de Loś y lo utilizaremos para demostrar nuevamente compacidad. De hecho, hay muchos caminos para llegar al teorema de compacidad[16].

No hay muchas formas distintas de construir modelos, aparte de las mencionadas, y a veces —por ejemplo, en el libro de Chang-Keisler—, se ha utilizado el procedimiento de construcción como criterio unificador para ordenar y exponer la variedad de resultados conocidos en teoría de modelos. Esta es también la idea que aglutina el libro de Hodges [16], en el que para la creación de modelos se usa la *teoría de juegos*.

He comentado algunos de los teoremas más famosos y antiguos de nuestra disciplina, todos formulados y demostrados antes de los años 50. *¿Qué se hizo después?*

2.5. Teorías

Tal vez el cambio más importante se produce cuando nuestro interés se centra en ciertos conjuntos de sentencias que constituyen una teoría. Esto es, conjuntos cerrados bajo la relación de deducibilidad o, lo que es lo mismo en primer orden, cerrados bajo la relación semántica de consecuencia.

Con las teorías tenemos de salida un problema de dimensión ya que el conjunto de sus sentencias es siempre infinito —pues todos los teoremas lógicos son sentencias de cualquier teoría—.

¿Cómo presentar pues una teoría, cómo describir un conjunto infinito?

Hay casos en los que la teoría que nos interesa es lo que veces se denomina teoría de un sistema, o de una clase de sistemas.

Teoría de un sistema

Es decir, tenemos un sistema —por ejemplo $\mathcal{N}$ el de los números naturales con las operaciones aritméticas usuales— y queremos estudiar el conjunto de todas las sentencias verdaderas en $\mathcal{N}$. En estos casos, la descripción que hacemos de la teoría es sencillamente esa, sentencias de primer orden verdaderas en nuestro sistema.

$$Th(\mathcal{N}) = \{C \in \mathbf{SENT}(L) \mid \mathcal{N} \Vdash C\}$$

De forma semejante, cuando se trata de una clase de sistemas.

Teoría axiomática

En otras ocasiones, más felices, para describir una teoría podemos utilizar un conjunto decidible de sentencias, a las que llamamos axiomas, y considerar que las sentencias de nuestra teoría son sus consecuencias lógicas. La primera pregunta que se nos plantea es, *¿se pueden representar axiomáticamente todas las teorías?* La respuesta es que no. Hay teorías que no pueden ser generadas

[16] Hodges cree que empieces por donde empieces, si te afanas con tesón, lo demostrarás.

por ningún conjunto decidible de axiomas, aunque se admitan conjuntos infinitos (pero decidibles) de axiomas.

Teorías axiomatizables

Veamos algunos ejemplos: las teorías de grupos, anillos, cuerpos, retículos, y álgebras de Boole son axiomatizables, y lo que es más, lo son mediante un conjunto finito de axiomas. Las teorías de los cuerpos de característica cero, o la aritmética de Peano de primer orden —es decir, las consecuencias de los axiomas de Peano de primer orden— son axiomatizables, pero no finitamente.

Teorías no axiomatizables

Sin embargo, la teoría de los números naturales —es decir, las sentencias de primer orden verdaderas en el sistema $\mathcal{N}$— no es axiomatizable.

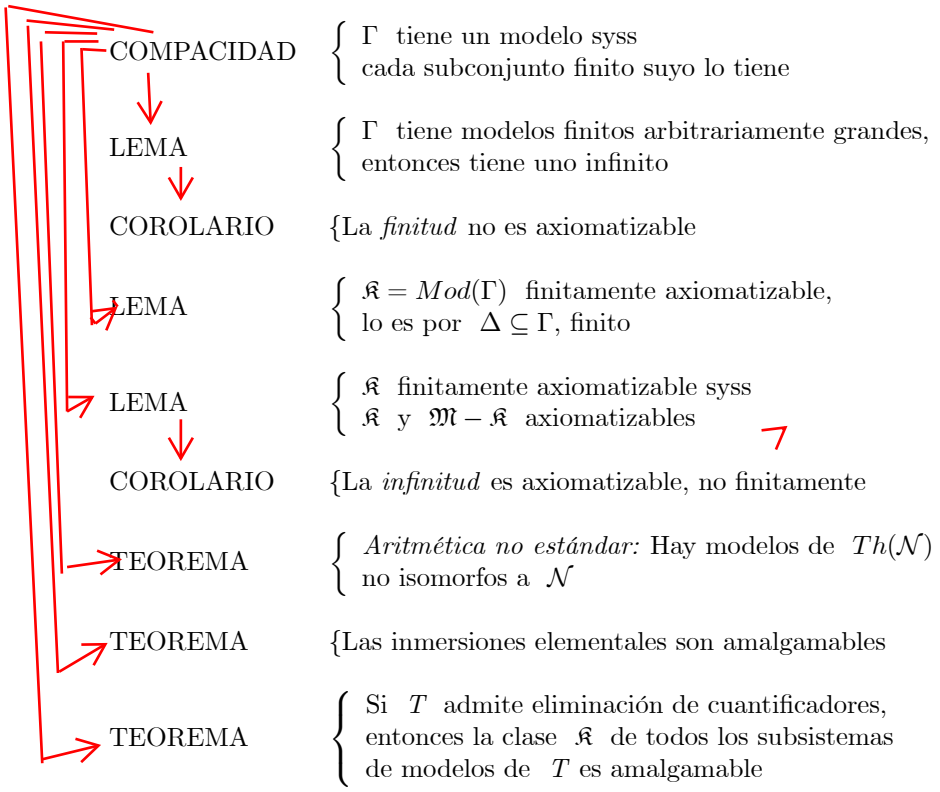
¿Qué es lo que sucede entonces con los números naturales?

Sencillamente, que la aritmética de Peano[17] es un subconjunto propio de la teoría de los números naturales; es decir,

$$\{C \in SENT(L) \mid \mathbf{AP}^1 \vdash C\} \subset Th(\mathcal{N})$$

Hay una gran diferencia entre probar que una teoría es axiomatizable y demostrar que no lo es. En el primer caso basta con dar sus axiomas, mientras que en el segundo tenemos que demostrar que no puede haberlos; para hacer esto último usamos frecuentemente el teorema de compacidad. Dedico en mi libro [21] un capítulo completo a demostrar las consecuencias del teorema de compacidad, siguiendo este esquema:

[17] Los axiomas los dimos en el teorema 77



En particular, vemos que no es axiomatizable la propiedad de finitud, y que la infinitud sólo es axiomatizable con infinitos axiomas. Esta característica distingue a la lógica de primer orden de otras más potentes expresivamente, como la de segundo orden o la de tipos: de hecho un test de compacidad consiste en ver si la infinitud es finitamente axiomatizable.

2.5.1. Modelos no estándar

Como consecuencia directa del teorema de compacidad obtenemos el siguiente resultado:

Teorema 106 *Sea* $\mathcal{N} = \langle \mathbb{N}, 0, s, +, \cdot, \leq \rangle$ *el sistema de los naturales, tal y como lo conocemos intuitivamente. Hay un sistema* $\mathcal{M}$ *que es modelo de* $Th(\mathcal{N})$ *pero que no es isomorfo a* $\mathcal{N}$.

Demostración. *Sea* L *el lenguaje adecuado para hablar de* $\mathcal{N}$ *y añadámosle una nueva constante* k. *Y sea*

$$\Sigma = Th(\mathcal{N}) \cup \left\{ C_n \mid C_n := \neg k = \sigma \ldots^{(n} \sigma c, \text{ para cada } n \right\}$$

Se demuestra que cada subconjunto finito de Σ *tiene un modelo y por compacidad que también lo tendrá el total. De este modelo se extrae el* $\mathcal{M}$ *mencionado. El*

modelo no es estándar porque aparecerán una serie de elementos que no son ni cero, ni sus siguientes, constituyendo lo que se denominan $\mathbb{Z}-$cadenas (ver figura:2.9) , por su semejanza con los enteros. ■

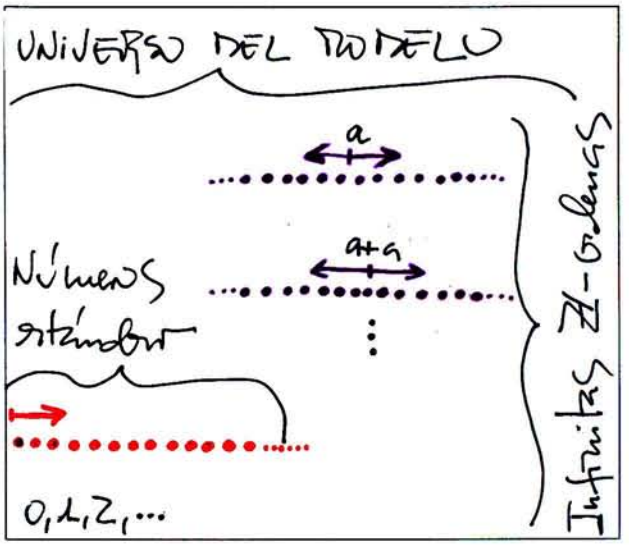


Figura 2.9: $\mathbb{Z}-$*cadenas*

La existencia de modelos no estándar para la aritmética había sido descubierta por Skolem en los años treinta, pero durante mucho tiempo no se les prestó demasiada atención. De hecho, hasta 1949 fueron utilizados como contraejemplos patológicos. En esta fecha Henkin demostró la completud de la teoría de tipos usando modelos no estándar e invirtiendo su consideración. Aunque el sentido de no-estándar en primero y segundo orden no es el mismo, conviene saber que están relacionados, como se verá en la sección 10.7; en primer orden no estándar significa no isomorfo al modelo pretendido —aquí, no isomorfo a $\mathcal{N}$—, en el segundo, que utiliza una nueva semántica en donde los universos de la estructura no contienen necesariamente a todos los subconjuntos posibles. Pues bien, al final de su artículo *"Completeness in the theory of types"* Henkin construye un modelo de la aritmética de Peano doblemente no estándar.

En la década de los cincuenta el estudio de los modelos no estándar se convirtió en una especie de moda y en el Simposio de matemáticas de Varsovia de 1959 se presentaron varios artículos sobre el tema, uno de ellos de Robinson. Las investigaciones se centraban en los modelos de la aritmética y el objetivo de las mismas no era el de las propiedades matemáticas de los números sino el de las propiedades metamatemáticas de los lenguajes formales y los cálculos deductivos. El análisis no estándar lo inventó Robinson en el otoño de 1960. En el prefacio de su libro *Non-standard analysis* afirma que pensó que los conceptos y métodos de la lógica matemática podrían proporcionar un marco adecuado

para desarrollar el cálculo diferencial e integral por medio de números infinitamente grandes e infinitamente pequeños. Este cálculo había sido descrito en esos mismos términos en el siglo XVII por Leibniz y Newton, usando cantidades infinitamente pequeñas pero distintas de cero. Durante el siglo XVIII se desarrolló la técnica del cálculo y se consolidó en el XIX, ya sin infinitesimales. Pero, como dije, en el año 1960 se creó el análisis no estándar en el que reaparecen los infinitesimales, ahora con todo el rigor y precisión exigido por los estándares modernos. La idea clave es la de poder aprovechar los modelos no estándar de los reales.

¿Tiene alguna ventaja el análisis no estándar frente al estándar, aparte del histórico-sentimental de resucitar los infinitesimales de Leibniz?

Hay que decir que el análisis no estándar, a diferencia de la aritmética no estándar se utiliza no sólo en metamatemática, sino también en matemáticas. Concretamente Robinson y Berstein lo usaron para solucionar un problema abierto sobre espacios de Hilbert. No obstante, toda prueba realizada en análisis no estándar puede sustituirse por una estándar, aunque las primeras son más intuitivas y menos artificiosas, dicen los expertos. Por consiguiente, el elegir análisis no estándar o no hacerlo es cuestión de gusto, no de necesidad.

Quien se puso decididamente a favor de él fue Gödel, pronosticando que sería el análisis del futuro. Dijo también que los historiadores de la matemática que nos sucedan considerarán una gran estupidez el no haber sabido dar, 300 años antes el salto de los reales a los infinitesimales, siendo así que es tan natural como el dado para pasar de los naturales a los enteros, o de éstos a los racionales o de los racionales a los reales. Aunque Gödel exageraba, nadie duda hoy en considerar que el análisis no estándar es uno de los mayores inventos de la lógica matemática en la segunda mitad del siglo XX y una agradable consecuencia de los teoremas de compacidad y Löwenheim-Skolem.

2.5.2. Teorías completas

Otra importante propiedad relacionada con la de axiomatizabilidad, que no todas las teorías comparten, es la de completud. Una teoría es completa si para cada sentencia B del lenguaje, o ella, o su negación es deducible en la teoría.

Definición 107 Δ *es completa syss para cada* $B \in \Delta : \Delta \vdash B$ *o* $\Delta \vdash \neg B$

Por supuesto, si definimos a una teoría semánticamente, como el conjunto de las sentencias verdaderas en un cierto sistema, no tiene sentido preguntarse si dicha teoría es completa: ¡Naturalmente que sí!, en un sistema una sentencia cualquiera es o verdadera, o falsa. Sin embargo, no todas las teorías se presentan de este modo, y nos interesa poder caracterizar a una teoría completa mediante procedimiento no sintácticos. La primera cuestión que se puede demostrar es que la completud de una teoría equivale a que sean elementalmente equivalentes todos sus modelos; es decir, que satisfagan simultáneamente las mismas sentencias. Sabemos que equivalencia elemental e isomorfía están relacionados pues lo segundo implica lo primero.

Completud de una lógica versus completud de una teoría

El concepto de completud de una teoría está muy estrechamente vinculado al de equivalencia elemental y al de decisión. En particular, si Δ es una teoría completa la clase de sus modelos está bastante bien definida ya que dos cualesquiera son elementalmente equivalentes. Tal y como se ha definido este concepto, es una propiedad sintáctica, relacionada con el cálculo y sus reglas; la de ser maximal, estar *al completo.* Por otra parte, la completud del cálculo significa la equivalencia entre la noción sintáctica y semántica de consecuencia. Hay quienes prefieren reservar la palabra completud para teorías y usar en los cálculos *suficiencia*. Yo prefiero mantener el vocablo completud en ambos casos, porque son formulaciones de un mismo problema, al menos en su génesis. Me explicaré mejor.

Normalmente estamos interesados en un sistema o en una clase de sistemas — por ejemplo, $\mathcal{N}$ o $\mathfrak{G}$— y en un lenguaje apropiado proponemos un conjunto de axiomas Δ que cifre las características del sistema o de la clase. Por descontado, en Δ ponemos sólo sentencias verdaderas en los sistemas considerados, pero *¿están todas las precisas?*

En el primer caso, *¿se cumple que si* $\mathcal{N} \Vdash B$ entonces $\Delta \vdash B$*?*

En el segundo, *¿se verifica que si* $\mathcal{B} \Vdash B$ para cada $\mathcal{B} \in \mathfrak{G}$ entonces $\Delta \vdash B$*?*

Cuando la respuesta es afirmativa podemos decir que Δ *es completo respecto del modelo* $\mathcal{N}$ en el primer caso, o que lo es respecto de la clase $\mathfrak{G}$ en el segundo.

Suponed que nuestra clase de sistemas es tan amplia que incluye a todos los de un cierto tipo $\langle \mu, \delta \rangle$. Una sentencia verdadera en todos ellos es lo que llamamos fórmula válida y preguntarnos si el conjunto $\varnothing$ de fórmulas es completo en la clase de todos los sistemas es justamente preguntarse si el cálculo es completo en sentido débil. La completud fuerte significa que todo conjunto Γ es completo respecto de la clase formada por todos sus modelos.

2.5.3. Teorías categóricas

¿Cómo relacionar el resultado anterior con la completud de una teoría?

Introduciremos el concepto de categoricidad que establece que una teoría consistente es categórica, si todos sus modelos son isomorfos.

Definición 108 *Sea* Δ *una teoría consistente:* Δ *es categórica syss para cada* $\mathcal{A}, \mathcal{B}$ *: Si* $\mathcal{A} \Vdash \Delta$ *y* $\mathcal{B} \Vdash \Delta$ *entonces* $\mathcal{A} \cong \mathcal{B}$

Ahora contamos con tres procedimientos para saber si una teoría es completa:

1. el sintáctico
2. utilizando equivalencia elemental

3. utilizando categoricidad —porque si T es categórica, todos sus modelos son elementalmente equivalentes y, por tanto, es completa—

No obstante, los resultados obtenidos son de poca aplicabilidad pues por el teorema de Löwenheim-Skolem, toda teoría con modelos infinitos posee modelos no isomorfos —pues los modelos de distinta cardinalidad no pueden ser isomorfos—.

Demostraremos también que toda teoría completa, con un modelo finito, es categórica. Pero aquí terminan las teorías categóricas.

Teorías $\kappa-$*categóricas*

Es por esto por lo que es necesario introducir una noción más amplia, menos exigente que la de categoricidad. Dicha noción es la de *κ-categoricidad*; es decir, que teniendo un modelo de cardinalidad κ, todos los de dicha cardinalidad sean isomorfos. El concepto de κ-categoricidad nos permite establecer otro test de completud de teorías, el llamado test de Vaught.

Teorema 109 ***(Test de Vaught)*** *Sea Γ una teoría de cardinalidad κ que no tiene modelos finitos. Si Γ es $\lambda-$categórica para un $\lambda \leq k$ —siendo λ infinito—, entonces Γ es completa.*

En las teorías con modelos finitos, categoricidad y completud son propiedades que se dan (o dejan de darse) simultáneamente.

¿Pasa lo mismo con la κ-categoricidad en las teorías en las que todos sus modelos son infinitos?

No, veremos que hay teorías completas que no son $\kappa-$categóricas para ningún κ infinito. Por consiguiente, hemos de desarrollar otras técnicas para caracterizar a las teorías completas que escapan a la $\kappa-$categoricidad.

2.6. Otras propiedades

Las teorías completas son interesantes por varios motivos. Veamos uno: Supongamos que $\mathcal{A}$ y $\mathcal{B}$ son dos modelos de una teoría completa T. Si B es una sentencia verdadera en $\mathcal{A}$, también lo será en $\mathcal{B}$. Por consiguiente, las propiedades conocidas, y expresables en primer orden, de un sistema que nos sea familiar $\mathcal{A}$ pueden también aplicarse a otro menos familiar, siempre que ambos sean modelo de la misma teoría completa.

Por otra parte, toda teoría axiomatizable y completa es decidible; es decir, existe un procedimiento efectivo que, dada una sentencia nos dice si está en la teoría o nó.

Para ver cómo podemos demostrar decidibilidad utilizando axiomatizabilidad y completud, obsérvese que los teoremas de una teoría axiomatizable forman un conjunto recursivamente enumerable, al igual que las sentencias refutables en la teoría (negación de los teoremas). Pero, si una teoría es completa, cada sentencia es: o un teorema, o una sentencia refutable. Por consiguiente, tanto el

conjunto de los teoremas, como su complementario son ambos recursivamente enumerables. Por tanto, la teoría es decidible. (ver figura: 2.10)

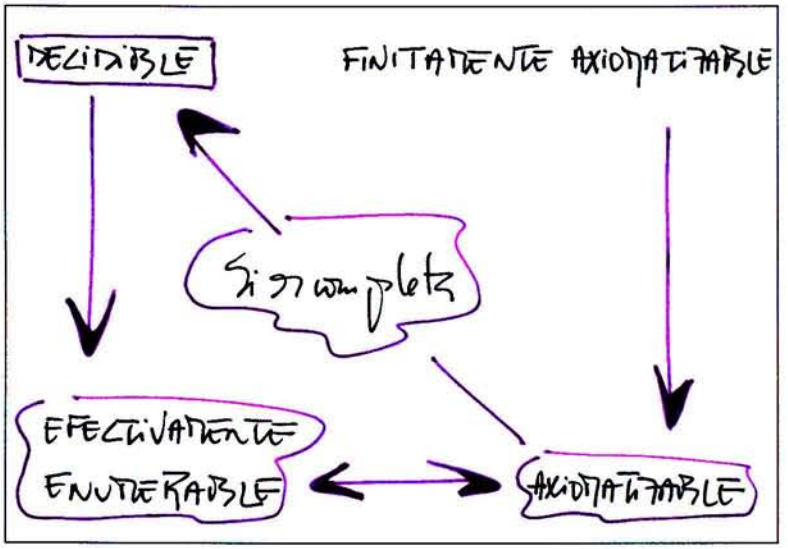


Figura 2.10: *Relación entre propiedades*

Comentario 110 *Este y otros resultados relacionados con la teoría de modelos, como el teorema de indecidibilidad de Church y el de incompletud de Gödel, requieren más teoría de la recursión de la que se puede ofrecer en un curso introductorio de teoría de modelos, por lo que no los demostraremos. Lo que hago es ofrecer un curso optativo dedicado a recursión, cuyo objetivo principal sea el teorema de incompletud. En este texto le dedico el capítulo siguiente.*

Aplicando algunos de los teoremas ya comentados de teoría de modelos junto a los de Gödel y Church se puede demostrar lo siguiente:

1. La teoría de los números naturales $Th(\mathcal{N})$ no es axiomatizable ni decidible. Por supuesto, la aritmética de Peano de primer orden —$\mathbf{CON}(\mathbf{AP})$—, es un subconjunto propio de $Th(\mathcal{N})$. Pero no es éste un problema del conjunto de axiomas $\mathbf{AP}$, no existe ningún otro conjunto Γ de sentencias de primer orden tal que $\mathbf{CON}(\Gamma) = Th(\mathcal{N})$. Además, por el teorema de Löwenheim-Skolem, sabemos que $Th(\mathcal{N})$ tiene modelos no isomorfos, y por compacidad podemos construir uno de esos modelos. Así que $Th(\mathcal{N})$ no es categórica, y menos aún $\mathbf{CON}(\mathbf{AP})$.

2. La teoría de los números reales $Th(\mathcal{R})$ por el contrario, es axiomatizable y decidible, aunque no categórica.

El analizar estos resultados, en cierto modelo sorprendentes, es también misión de la teoría de modelos.

¿Por qué $Th(\mathcal{R})$ *es decidible y finitamente axiomatizable y* $Th(\mathcal{N})$ *no es ninguna de estas cosas?*

Después de todo, los números naturales están incluídos entre los reales. Incluso en el álgebra elemental podemos construir nombres para cada número natural. Aunque esto es evidentemente cierto, en el álgebra elemental no podemos referirnos de manera alguna al conjunto de todos los números naturales. No hay ninguna fórmula B con una variable libre —en el lenguaje de $\mathcal{R}$— que sea satisfecha por un número real si y sólo si es un natural; es decir, los naturales no son definibles en $\mathcal{R}$. Este hecho bloquea la transferencia de los resultados de Gödel y Church, sobre los naturales, a los reales.

Se puede describir a la matemática moderna como a la ciencia de los *objetos abstractos*, sean números reales, estructuras algebraicas, o lo que sea. La lógica matemática le añade a esta ciencia la preocupación acerca del lenguaje en el que puede ser formulada, su capacidad de caracterizar a dichos objetos, la posibilidad de definir objetos nuevos, etc.

Bibliografía

[1] Barwise, J. [1975]. *Admissible Sets and Structures.* Springer-Verlag. Berlín. Alemania.

[2] Barwise, J. [1977]. *Handbook of Mathematical Logic.* North Holland Publishing Company. Amsterdam. Holanda.

[3] Barwise, J. y Feferman, S. eds [1985]. *Model-Theoretic Logics.* Springer-Verlag. Berlín. Alemania.

[4] Bell, J y Slomson, A. [1969]. *Models and Ultraproducts.* North Holland Publishing Company. Amsterdam. Holanda.

[5] Bridge, J. [1977]. *Beginning Model Theory.* Clarendon Press. Oxford. Reino Unido.

[6] Corcoran, J. (ed.), [1983]. *Logic, Semantics and Metamathematics.* Hackett Publishing Company, 1983.

[7] Chang, C. y Keisler, J. [1977]. *Model Theory.* North Holland Publishing Company. Amsterdam. Holanda. (reedición 1990)

[8] van Dalen, D. [1983]. *Logic and Structure.* Springer-Verlag. Berlín. Alemania.

[9] Doets, K. [1996]. *Basic Model Theory.* CSLI. Stanford. USA

[10] Etchemendy, J. [1988]. *"Tarski on Truth and Logical Consequence"*, The Journal of Symbolic Logic, vol.53, n^{o}1, pp.51-79.

[11] Etchemendy, J. [1990]. *The Concept of Logical Consequence.* Harvard University Press.

[12] van Heijenoort, J. [1967]. *From Frege to Gödel: a source book in mathematical logic,* 1879-1931. Cambridge, Mass: Harvard University Press.

[13] Henkin, L. [1949]. *"The completeness of the first order functional calculus".* **JSL.** vol. 14, pp. 159-166.

[14] Henkin, L. [1950]. *"Completeness in the theory of types".* **JSL.** vol. 15. pp. 81-91.

[15] Henkin, L. [1996]. *"The discovery of my completeness proofs"*, Dedicated to my teacher, Alonzo Church, in his 91^{st} year, **Bulletin of Symbolic Logic**, vol. 2, Number 2, June 1996. (presentado el 24 de Agosto de 1993 en el XIX International Congress of History of Science, Zaragoza, Spain).

[16] Hodges, W. 1985. *Building models by games.* Cambridge University Press. Cambridge. Reino Unido.

[17] Hodges, W. [1993]. *Model Theory.* Cambridge University Press. Cambridge. Reino Unido

[18] Kleene, S. [1952]. *Introduction to metamathematics.* Amsterdam: North Holland Publishing Company.

[19] Kreisel, G & Krivine, J.L. [1971]. *Elements of Mathematical Logic.* Amsterdam: North-Holland Publishing Company.

[20] Löwenheim, L. [1915]. *"On the possibilities in the calculus of relatives".* en [12]].

[21] Manzano, M. [1989]. *Teoría de Modelos.* Alianza Universidad Textos. Alianza Editorial. Madrid.

[22] Marraud, H. [1990]. *Teoría de Modelos Elemental.* Ediciones UAM.

[23] Tarski, A. [1930]. *"The Concept of Truth in Formalized Languages"*, en [6].

[24] Tarski, A. [1936]. *"On the Concept of Logical Consecuence",* en [6].

Capítulo 3

Teoría de la Computabilidad

Comentario 112 *La materia condensada en estas ocho secciones constituye un curso tranquilo de segundo ciclo. Los alumnos deben saber ya lógica de primer orden, haber demostrado los teoremas de completud y corrección del cálculo y es mejor que posean nociones muy elementales de Teoría de Modelos. Si esto no fuera así, podría empezarse por ello, introduciendo los conceptos de* ***teoría****,* ***teoría de un sistema*** *o* ***de una clase de sistemas****, y su simétrico, el de* ***modelo de un conjunto de sentencias****. También introduciremos las propiedades de* ***completud de teorías*** *y la de* ***axiomatizabilidad****. Se usará también el concepto de* ***relación definible*** *en un sistema.*

El objetivo fundamental no es tanto el de introducir el concepto básico de *recursividad,* sino el de trazar la interconexión profunda que con la lógica, en el sentido restrictivo de *Ciencia de la Consecuencia*, existe. Este curso yo lo titularía *Recursión para la Metalógica.* Se demostrará el teorema más famoso de nuestra disciplina, el de *incompletud de Gödel*, por lo que conviene situarse históricamente. Es de sumo interés que se destaque el vínculo con la *Teoría de Modelos.*

3.1. Concepto intuitivo: algoritmo

¿Cuándo decimos que una función es recursiva, qué significa ser recursiva?

Hay varias definiciones precisas, equivalentes entre sí, de este concepto. La noción intuitiva correspondiente a ser recursiva es la de ser efectivamente computable. Fijémonos en las funciones parciales n-arias sobre los naturales.

¿Cuándo decimos que una función parcial n-aria f sobre los naturales es efectivamente computable?

Sencillamente, cuando hay un procedimiento efectivo —esto es, un algoritmo— que la computa. Un procedimiento efectivo debe cumplir los siguientes requisi-

tos:

1. Instrucciones precisas, de longitud finita. Estas instrucciones no deben requerir astucia ninguna por parte de la persona o máquina que las siga y no deben incluir el azar —lanzar una moneda al aire y actuar según el resultado, por ejemplo—

2. Si se le da un argumento $\bar{x}$ en el dominio de la función, debe producir $f(\bar{x})$ en un número finito de pasos de cálculo

3. Si se le da un $\bar{x}$ que no esté en el dominio de la función, debe seguir indefinidamente o parar, pero en ningún caso producir un valor $f(\bar{x})$

Como ejemplos de funciones efectivamente computables podemos citar la suma y la multiplicación. Los procedimientos efectivos de cálculo son los que aprendimos en el *cole*. Por otra parte, toda función con dominio finito es efectivamente computable: las instrucciones pueden ser, sencillamente, la lista de los valores. Hay, por supuesto, funciones efectivamente computables más interesantes que las de estos ejemplos.

Cuando tenemos un algoritmo que determina si se cumple o no una propiedad, el conjunto o relación definido por ella se llama *decidible*. Así pues, el concepto intuitivo de computabilidad efectiva para funciones se generaliza para conjuntos. Hay algunos que son decidibles sólo a medias y los llamamos *efectivamente enumerables*: sus elementos pueden ser listados.

Nosotros no imponemos restricciones de naturaleza práctica a los procedimientos efectivos. Cuando se trata de funciones sobre los naturales, los argumentos de las funciones han de ser números naturales, pero no se restringe su tamaño. Por otra parte, aunque el número de pasos ha de ser finito, tampoco ponemos un tope. Además, no se prefija la cantidad de papel —o espacio de memoria— que haya de precisarse para realizar el cálculo. Estas consideraciones son importantes cuando comparamos la computabilidad efectiva con la práctica: la clase de las funciones efectivamente computables la obtenemos en una situación ideal en donde no importa el tiempo ni la memoria requerida.

Aunque la descripción intuitiva del concepto de función efectivamente computable que hemos dado no puede considerarse una definición, cuando se trate de mostrar que una función es efectivamente computable será suficiente con esta presentación informal. Otra cosa muy distinta es cuando se quiera demostrar que no lo es[1].

Las funciones recursivas son las efectivamente computables, pero reservamos el vocablo *"recursiva"* para el concepto matemático; es decir, el definido con precisión.

Comentario 113 *Nosotros definiremos en un principio el concepto de recursividad como ser representable en una teoría consistente y finitamente axioma-*

[1] Veremos en la página 311 del capítulo 11, que fue justamente el deseo de resolver problemas de indecidibilidad lo que movió a un significativo grupo de lógicos a definir la computabilidad efectiva, a introducir matemáticamente el concepto de recursividad.

tizable. En esto seguiremos fielmente el libro de Enderton [9] ya que se adapta bien al nivel de nuestra licenciatura.

Lógica y recursión

¿Por qué es la teoría de las funciones recursivas una parte de la lógica?

Claramente, si no se hubiera inventado la teoría de las funciones recursivas en lógica, se habría hecho, más tarde, en informática. Sin embargo, no fue una casualidad histórica: hay aspectos fundamentales de la lógica que requieren el concepto de procedimiento efectivo. De hecho, donde por vez primera se define con precisión fue en el artículo de Gödel de la incompletud de la lógica superior.

Pero incluso a un nivel mucho más básico la lógica emplea algoritmos; nosotros hemos establecido con claridad requisitos de *efectividad* del lenguaje y del cálculo lógico:

1. Al especificar los signos primitivos —dado un signo hay que poder determinar si es o no un signo primitivo—
2. Al definir fórmulas —dada una sucesión de signos hay que poder determinar si es o no una fórmula—
3. Al especificar los axiomas —dada una fórmula hay que poder determinar si es o no un axioma—
4. Al definir las reglas de inferencia —cuando se propone una fórmula como conclusión de una inferencia inmediata se debe poder determinar si es correcta, conforme a las reglas—

Como hemos visto, en la lógica se construyen cálculos para probar teoremas. Los cálculos han de ser tales que cualquiera pueda comprobar la corrección de la prueba mediante un procedimiento efectivo. El conjunto de las pruebas ha de ser *decidible* y el de los teoremas del cálculo *efectivamente enumerable*. Por consiguiente, si ya sabemos que un conjunto de fórmulas no es recursivamente enumerable —y para ello tendremos que haber definido matemáticamente este concepto—, también conocemos que no hay cálculo capaz de generarlas y no tendremos que hacer esfuerzos inútiles para encontrarlo.

¿Qué se entiende por conjunto efectivamente enumerable?

No hay que confundir *enumerable* con *numerable*. Un conjunto es numerable cuando se puede poner en correspondencia con el conjunto de los números naturales; cuando hay una función inyectiva del conjunto en el de los naturales, distinguiéndose entre finito-numerable e infinito-numerable. La enumerabilidad supone un procedimiento efectivo que liste en un cierto orden los elementos del conjunto. La idea es que, al menos idealmente, se pueda programar un computador que anote exactamente los elementos del conjunto en una lista. No importa que la máquina tenga que dar un número infinito de pasos, lo realmente importante es que el procedimiento pueda ser seguido mecánicamente y sin que se cuele el azar; debe ser lo que denominamos un *procedimiento efectivo* o *algoritmo*.

El concepto de enumerabilidad es más débil que el de decidibilidad. Un conjunto Δ es decidible cuando existe un procedimiento efectivo que, dada una expresión nos diga si pertenece o no al conjunto. Aquí resulta pertinente el ejemplo de la lógica proposicional y el de la de primer orden. En la primera, que es decidible, hay procedimientos para determinar si una fórmula cualquiera es o no es una tautología; las tablas de verdad, por ejemplo. En la segunda, si una deducción no sale, no sabemos si es falta de pericia o imposibilidad real. Es verdad que siempre puedo hacer una prueba de independencia semántica, pero las pruebas de independencia no son efectivas.

¿Cuál es la diferencia entre enumerable y decidible?

No todos los conjuntos enumerables son decidibles porque siguiendo el procedimiento efectivo del listado de Δ podemos encontrar las fórmulas C tales que $C \in \Delta$; pero, *¿qué pasa si* $C \notin \Delta$*?* Por regla general, no hay respuesta, ya que el que la fórmula no haya aparecido aún en el listado no nos garantiza que no vaya a aparecer, nada nos asegura que no esté. Es por esto por lo que para que Δ sea decidible hace falta no solamente que Δ sea enumerable, sino que también su complementario lo sea. Más tarde veremos que el conjunto de las fórmulas lógicamente válidas de primer orden no es decidible.

Por otra parte, una teoría axiomatizable —mediante un conjunto decidible de axiomas, claro— tiene un conjunto efectivamente enumerable de teoremas. Si además de ser axiomatizable es completa, entonces sus teoremas formarán un conjunto decidible.

Estos son sólo unos cuantos ejemplos para subrayar que la teoría de la recursión es de la lógica una parte fundamental.

3.2. Concepto matemático: Tesis de Church

La Teoría de la recursión clásica abarca el estudio de las funciones definidas sobre los naturales, pero la teoría de la recursión actual, al ir desarrollando su potencial propio y sus métodos específicos, ha alcanzado un notable desarrollo abstracto y aplicaciones insospechadas. Los orígenes de la teoría clásica pueden hallarse en Dedekind, cuando en 1888 introduce el estudio de las funciones definibles sobre el conjunto de los números naturales usando ecuaciones y, recurrentemente, la inducción sobre los naturales que él había formulado y precisado. De ahí viene, justamente, el que se adoptara el nombre de Teoría de la recursión.

Por lo que respecta a su estadio presente, cuyo radio de acción cubre la totalidad de las funciones efectivamente computables, los orígenes hay que buscarlos en el grupo de Princeton; empezó con Church, pero si hay que atribuirle un padre, éste es Kleene. El fue quien la impulsó, definió y acotó. Suyos son los teoremas de la forma normal y el de la recursión.

Hay varias maneras equivalentes de matematizar el concepto intuitivo de algoritmo, algunas aparecieron en los años 30 para captar nociones que en un principio parecían diferentes: la primera era la caracterización de Gödel de las funciones definidas mediante recursión, la segunda era la noción de función definible con el operador λ, que Church y Kleene introdujeron, y la tercera,

era la de función computable mediante una máquina abstracta, las máquinas de Turing[2]. Pronto se demostró que las tres nociones definían las mismas funciones.

Si bien estaba claro que todas las funciones definidas mediante cualquiera de los procesos anteriores era efectivamente computable, este resultado seguía sin servir para demostrar que un conjunto es indecidible; necesitamos la conversa. Por su propia naturaleza esta afirmación no puede ser demostrada, y si no se considera adecuado que sea una definición, hay que proporcionarle otro estatus.

Tesis de Church

Por esas mismas fechas Church se preguntó si el concepto de recursividad de Turing se correspondía con la noción intuitiva de computabilidad efectiva. Su respuesta, conocida como *tesis de Church*, es que sí. Church y Kleene definieron también el concepto de λ-*definibilidad*, que se corresponde —según demostró Turing— con el de recursividad[3].

Shoenfield [20] dice que la tesis de Church surgió de manera casual, yo diría algo muy parecido, que surgió de forma natural[4].

Aunque por tratarse de una tesis no se puede demostrar matemáticamente, hay razones poderosas para aceptarla:

1. La primera es una razón basada en la experiencia acumulada: todos los algoritmos conocidos son recursivos.

2. La segunda está basada en la equivalencia entre los tres conceptos introducidos para establecer la noción de algoritmo: resulta tranquilizador que desde planteamientos tan distintos se lleguen a definir exactamente las mismas funciones.

3. La tercera, que fue la que sin duda convenció a Gödel, está relacionada con el análisis que Turing hizo del concepto de computabilidad; no se interesa tanto por la naturaleza de las funciones computables como por el propio proceso de computación. Las funciones se construyen a partir de funciones elementalísimas de las que no cabe duda de su carácter algorítmico mediante procesos que tampoco plantean dudas al respecto.

4. De hecho, el argumento anterior es muy similar al dado por Church al que nos referiremos en 11.6.

5. Finalmente, la razón que convenció a Kleene: la del fracaso del argumento diagonal[5].

[2] La versión mediante máquinas imaginarias fue concebida por Turing en 1936. Su trabajo es anterior a las computadoras e influyó poderosamente en la creación de éstas.

[3] La sección 11.6 está enteramente dedicada a dilucidar cómo actuó el cálculo lambda en la destilación de estos conceptos. Yo confío en su alquimia, llegando incluso a pensar que el cálculo lambda fue la piedra filosofal de Church.

[4] Lo veremos con detalle en la página 312, cuando hayamos introducido el cálculo lambda.

[5] La extrañeza ante semejante aserto, nos movió a Enrique Alonso y a mí a escribir el artículo *Diagonalization and Church's Thesis: Kleene's Homework* .

Ahora se podía demostrar que un conjunto no era decidible pues bastaba con mostrar que no era recursiva la función que cifraba la pertenencia a él. Esto lo hizo Church en 1936 para un problema en el cálculo lambda. En el mismo año, apoyándose en los resultados de Gödel, Church demostró que si fuera decidible la lógica de primer orden, también lo sería el del problema anterior, que había sido resuelto negativamente.

3.3. Enumerabilidad

Comentario 114 *El objetivo fundamental de esta sección sería probar el teorema de enumerabilidad para la lógica de primer orden; es decir, demostrar que el conjunto de las fórmulas lógicamente válidas de esta lógica es efectivamente enumerable. Además de eso se vería la relación existente entre enumerabilidad, axiomatizabilidad y decidibilidad en el caso de teorías de primer orden.*

El teorema de enumerabilidad es una consecuencia del de completud para la lógica de primer orden. Dicho teorema afirma que el conjunto de las fórmulas lógicamente válidas de primer orden es efectivamente enumerable.

La demostración del teorema de enumerabilidad es fácil. Dado que tenemos un cálculo deductivo que nos permite obtener como teoremas todas las fórmulas válidas y sólo ellas —teoremas de corrección y completud del cálculo—, sólo hay que hacer un programa que produzca sistemáticamente todas las deducciones.

Aquí pueden demostrarse las implicaciones precisas para completar el esquema del capítulo precedente, el de la relación entre distintas propiedades de una teoría (ver figura: 2.10) :

Nociones fundamentales

Una vez introducidos los conceptos de *procedimiento efectivo*, *conjunto efectivamente enumerable* y *conjunto decidible*, se verá:

Proposición 115 *Existe un procedimiento efectivo que, dada una fila de signos, nos dice si es una fórmula o no.*

Teorema 116 *Un conjunto de expresiones es decidible si y sólo si tanto él como su complementario —respecto del conjunto de todas las expresiones— son efectivamente enumerables.*

Se introducirá también el concepto de *lenguaje razonable*; es decir, cuando sus constantes forman un conjunto efectivamente enumerable y los relatores y functores n-arios constituyen conjuntos decidibles.

Teorema 117 *(Enumerabilidad) VAL es efectivamente enumerable*

Lo esencial en la demostración del teorema de enumerabilidad es que tanto los axiomas del cálculo —en su defecto, reglas sin premisas— como el de las deducciones en él son decidibles. Basándonos en este hecho se prueba que los

teoremas del cálculo forman un conjunto efectivamente enumerable. Utilizando completud y corrección se prueba que también las fórmulas válidas forman un conjunto efectivamente enumerable. Como corolario de este teorema demostramos que siempre que tengamos un conjunto decidible de fórmulas Γ el conjunto de sus teoremas

$$\{C \mid \Gamma \vdash C\}$$

o, de forma equivalente, el de sus consecuencias

$$\mathbf{CON}(\Gamma) = \{C \mid \Gamma \vDash C\}$$

es efectivamente enumerable.

Otro corolario del teorema de enumerabilidad es que si Γ es decidible, el lenguaje es razonable y para cada sentencia C o bien $\Gamma \vDash C$ o bien $\Gamma \vDash \neg C$, entonces el conjunto de las sentencias que se siguen de Γ es decidible.

Teorías

Utilizamos los conceptos de teoría, teoría de un sistema, teoría de una clase de sistemas y modelo de un conjunto de sentencias definidos en el capítulo precedente junto a los de axiomatizabilidad y axiomatizabilidad finita, y también el de completud de teorías. Se demuestra lo siguiente:

Teorema 118 *Si* $\mathbf{CON}(\Sigma)$ *es finitamente satisfacible, entonces hay un subconjunto finito suyo* $\Sigma_0 \subseteq \Sigma$ *tal que* $\mathbf{CON}(\Sigma_0) = \mathbf{CON}(\Sigma)$. *—Haciendo uso del teorema de compacidad—*

Hecho eso, los corolarios mencionados anteriormente pueden reescribirse:

Corolario 119 *Una teoría axiomatizable en un lenguaje razonable es efectivamente enumerable.*

Corolario 120 *Una teoría completa y axiomatizable en un lenguaje razonable es decidible.*

3.4. El sistema $\mathcal{N}_s$

En esta sección estudiamos el sistema

$$\mathcal{N}_s = \langle \mathbb{N}, 0, s \rangle$$

formado por los números naturales, el cero y la función del siguiente. La teoría de este sistema $Th(\mathcal{N}_s)$ resulta ser axiomatizable —los axiomas los recogemos en Nat_s—, siendo dicha axiomatización completa y decidible, pero no categórica.

Recordaremos no sólo el concepto de completud de teorías, sino también las definiciones equivalentes de dicho concepto que vamos a manejar y algunos de los procedimientos comunes para demostrarlo; distinguiremos completud de

categoricidad e introduciremos el concepto de $\lambda-$categoricidad[6]. En especial, en este tema para demostrar que $\mathbf{CON}(Nat_s)$ es completa vamos a utilizar el denominado test de Vaught[7].

La teoría del sistema $\mathcal{N}_s$ —es decir, las sentencias verdaderas en $\mathcal{N}_s$— es axiomatizable y la axiomatización es completa. Los axiomas son las sentencias del conjunto infinito Nat_s formado por:

1. El cero no es el siguiente de ningún número

$$C_1 := \forall x\ \sigma x \neq c$$

2. La función del siguiente es inyectiva

$$C_2 := \forall xy\ (\sigma x = \sigma y \rightarrow x = y)$$

3. Todo número distinto del cero es el siguiente de otro

$$C_3 := \forall y\ \ (y \neq c \rightarrow \exists x\ y = \sigma x)$$

4. No hay ciclos de longitud 1

$$C_{4,1} := \forall x\ \sigma x \neq x$$

5. No hay ciclos de longitud 2

$$C_{4,2} := \forall x\ \sigma\sigma x \neq x$$

en general, para cada número natural n,

6. no hay ciclos de longitud n

$$C_{4.n} := \forall x\ \sigma^{(n} ... \sigma x \neq x$$

Evidentemente, $\mathcal{N}_s$ es modelo de Nat_s pues en $\mathcal{N}_s$ el cero no es el siguiente de ningún número, la función del siguiente es inyectiva, todo número distinto de cero es el siguiente de otro y no hay ciclos en $\mathcal{N}_s$. Por consiguiente

$$\mathbf{CON}(Nat_s) \subseteq Th(\mathcal{N}_s)$$

Para demostrar que Nat_s axiomatiza $Th(\mathcal{N}_s)$ lo que hacemos es ver que

$$Th(\mathcal{N}_s) = \mathbf{CON}\,(Nat_s)$$

[6] Es decir, daremos un breve repaso a los primeros apartados del capítulo 7 de mi libro de *Teoría de Modelos*.

[7] Haría falta recordar la prueba de dicho teorema, que está basada en el de Löwenheim-Skolem

Para demostrar que vale la igualdad lo que hacemos es ver que $\mathbf{CON}\,(Nat_s)$ es una teoría completa. La razón es que una vez demostrado que $\mathbf{CON}\,(Nat_s)$ es completa, es inmediato ver que

$$Th(\mathcal{N}_s) \subseteq \mathbf{CON}\,(Nat_s)$$

pues si hubiera una sentencia C de $Th(\mathcal{N}_s)$ que no estuviera en $\mathbf{CON}\,(Nat_s)$, por ser ésta completa $\neg C \in \mathbf{CON}\,(Nat_s)$. Pero, puesto que

$$\mathbf{CON}\,(Nat_s) \subseteq T(\mathcal{N}_s)$$

también $\neg C \in Th(\mathcal{N}_s)$. Esto último es imposible, un mismo sistema no puede ser simultáneamente modelo de una sentencia y de su negación.

¿Cómo demostrar que $\mathbf{CON}\,(Nat_s)$ *es completa?*

De hecho, hay dos formas sencillas de hacerlo: Una es aplicando el test de Vaught —es decir, demostrando que la teoría no tiene modelos finitos y que los de la misma cardinalidad son isomorfos—; la otra es demostrar que admite eliminación de cuantificadores —es decir, que dada una fórmula C cualquiera, es posible encontrar una fórmula D sin cuantificadores tal que

$$Nat_s \vdash C \leftrightarrow D$$

Para poder aplicar el test de Vaught, necesitaremos demostrar que $\mathbf{CON}\,(Nat_s)$ no tiene modelos finitos y que los de la misma cardinalidad son isomorfos. Esto no es difícil de probar utilizando la noción de $\mathbb{Z}$-cadena. Veamos qué son las denominadas $\mathbb{Z}$-cadenas.

¿Cómo ha de ser un sistema $\mathcal{A} = \langle \mathbf{A}, c^{\mathcal{A}}, \sigma^{\mathcal{A}} \rangle$ *para ser modelo de* Nat_s?

1. En primer lugar, $\sigma^{\mathcal{A}}$ tiene que ser una función biyectiva de $\mathbf{A}$ en $\mathbf{A} - \left\{c^{\mathcal{A}}\right\}$. Esto se sigue de los tres primeros axiomas de Nat_s.

2. Por el axioma $C_{4.n}$ sabemos que no hay ciclos de longitud n. En $\mathbf{A}$ tiene que haber una parte estándar,

   $$c^{\mathcal{A}} \to \sigma^{\mathcal{A}}\left(c^{\mathcal{A}}\right) \to \sigma^{\mathcal{A}}\left(\sigma^{\mathcal{A}}\left(c^{\mathcal{A}}\right)\right) \to ...$$

   de puntos distintos. En $\mathbf{A}$ podría haber también otros puntos —por ejemplo $\mathbf{a}$—; caso de haberlos, tienen que tener sus siguientes. Pero, puesto que por C_3 todo número que no sea el cero tiene un predecesor, y por C_2 sabemos que es único, $\mathbf{a}$ estará situado en

   $$...* \to * \to \mathbf{a} \to \sigma^{\mathcal{A}}(\mathbf{a}) \to \sigma^{\mathcal{A}}\left(\sigma^{\mathcal{A}}(\mathbf{a})\right) \to ...$$

   a las que llamamos $\mathbb{Z}$-cadenas por su semejanza con los enteros.

En un modelo de Nat_s puede haber cualquier número de $\mathbb{Z}$-cadenas, siendo todas disjuntas y también disjuntas de la parte estándar.

Por otra parte, todo sistema formado de una parte estándar y de cualquier número de $\mathbb{Z}$-cadenas es un modelo de Nat_s.

La noción de $\mathbb{Z}$-cadena nos permite demostrar con facilidad que dos modelos cualesquiera de Nat_s con el mismo número de $\mathbb{Z}$-cadenas son isomorfos; a partir de ahí demostramos que dos modelos supernumerables de Nat_s de la misma cardinalidad son isomorfos. Ahora podemos aplicar el test de Vaught y concluir que $\mathbf{CON}\,(Nat_s)$ es una teoría completa.

Sabiendo que $Th(\mathcal{N}_s)$ es axiomatizable y su axiomatización es completa, aplicando los resultados del tema anterior, es fácil ver que es decidible. Esta prueba de la decidibilidad no es constructiva y habrá casos en los que estemos interesados no sólo en saber que es decidible, sino en conocer el método para poder aplicarlo en casos concretos.

Eliminación de cuantificadores

Hemos dicho que $Th(\mathcal{N}_s)$ es decidible.

¿Cuál es el procedimiento de decisión?.

Para esto es para lo que es interesante demostrar que la teoría admite eliminación de cuantificadores[8]; es decir, que toda fórmula de su lenguaje es equivalente en el marco de la teoría a una sin cuantificadores. El procedimiento de eliminación de cuantificadores es en sí mismo un procedimiento de decisión.

También se demostrará que, como consecuencia de admitir eliminación de cuantificadores, las relaciones definibles en $\mathcal{N}_s$ lo son mediante fórmulas sin ellos y por lo tanto es sencillo comprobar que los únicos subconjuntos de $\mathbb{N}$ definibles en $\mathcal{N}_s$ son los finitos y los cofinitos.

3.5. Relaciones y funciones representables

La Subteoría $\mathbf{CON}(Nat_E)$

En este tema introduzco las nociones de relación y función representable en una teoría y pruebo algunos resultados para el caso particular de la teoría $\mathbf{CON}\,(Nat_E)$, una subteoría finitamente axiomatizable de $Th(\mathcal{N})$. Aunque, de hecho, las nociones las introduzco para demostrar en la sección 3.7 los teoremas de incompletud de Gödel y de indecidibiilidad de Church, creo que son lo suficientemente interesantes como para merecer un tratamiento detallado.

Considero el sistema estándar de los naturales con las operaciones de suma, producto, exponenciación, así como el cero, la función del siguiente y la relación de orden. Sabemos que $Th(\mathcal{N})$ no es ni decidible ni axiomatizable, pero hay una subteoría suya $\mathbf{CON}\,(Nat_E)$ que siendo finitamente axiomatizable, permite representar, en un sentido que precisaré, hechos sobre conjuntos decidibles.

Las nociones de representabilidad y definibilidad describen ambas relaciones sobre los naturales mediante fórmulas; la diferencia es que en la representabilidad lo que interesa es que la fórmula sea deducible a partir de los axiomas de

[8] La demostración detallada está en mi libro de *Teoría de Modelos*, en la página 264 de la edición española.

Nat_E y en la definibilidad que sea verdadera en $\mathcal{N}$. Veremos que representabilidad en $\mathbf{CON}(Nat_E)$ implica definibilidad en $\mathcal{N}$. El interés que tiene la noción de representabilidad es que también implica la de decidibilidad. Concretamente, se demuestra que toda relación representable en una teoría consistente y axiomatizable es decidible. Como caso particular, cuando la teoría sea finitamente axiomatizable, la relación representable en ella seguirá siendo decidible.

¿Vale el recíproco?

Es decir, *¿es equivalente el ser decidible al ser representable en una teoría consistente y finitamente axiomatizable?*

Se trata de una versión de la tesis de Church, que aceptamos. Dicha tesis no es una sentencia matemática, ni tan siquiera metamatemática que pueda ser demostrada, es más bien el criterio de que la noción intuitiva de decidibilidad puede formularse correctamente si la identificamos con la de representabilidad en una teoría consistente y finitamente axiomatizable.

Pues bien, si a todo esto lo llamamos ser recursiva —es decir, identificamos ser recursiva con ser representable en teoría consistente y finitamente axiomatizable— podemos expresar la tesis de Church así: una relación es decidible si y sólo si es recursiva. Ahora la recursividad se convierte en la precisión de la noción intuitiva de decidibilidad.

Hay que insistir aquí en la diferencia entre la computabilidad teórica y la práctica. Y señalar que para que el algoritmo se pueda implementar en un ordenador real, con una memoria limitada, que produzca resultados en un tiempo razonable, no basta con saber que es decidible, sino también cómo crece la complejidad del procedimiento con la complejidad de la fórmula.

Presentación de Nat_E

El sistema en estudio es $\mathcal{N} = \langle \mathbb{N}, 0, s, +, \cdot, E, < \rangle$. En realidad bastaría con tener suma y producto[9] pues el resto de las operaciones pueden definirse en función de ellas; no obstante, se simplifican las pruebas teniéndolos.

Los axiomas de Nat_E son los siguientes:

1. C_1 y C_2 de Nat_s
2. $B_1 := \forall\, xy(x < \sigma y \leftrightarrow x \leq y)$
3. $B_2 := \forall x\ \neg x < c$
4. $B_3 := \forall\, xy(x < y \vee x = y \vee y < x)$

junto con las definiciones habituales de suma, producto y exponenciación.

Puesto que $\mathcal{N}$ es modelo de Nat_E tenemos que $\mathbf{CON}(Nat_E) \subseteq Th(\mathcal{N})$. Demostraremos algunos teoremas de esta teoría y veremos que todas las sentencias sin cuantificadores que son verdaderas en $\mathcal{N}$ son deducibles en Nat_E. Por supuesto, no se puede demostrar que admita eliminación de cuantificadores.

[9] De hecho, en el apéndice 3.10 demuestro el teorema de indecidibilidad de Church apoyándome en la propiedad homónima de la llamada Aritmética de Robinson, que no contiene exponenciación ni orden.

Relaciones representables

Recordaremos el concepto de relación definible en un sistema del capítulo anterior e introduciremos el de relación representable en una teoría y el de relación numeralmente determinada por Nat_E.

Definición 121 *Sea* $\mathbf{R}$ *una relación* $n-$*aria sobre* $\mathbb{N}$ —*esto es,* $\mathbf{R} \subseteq \mathbb{N}^n$—. *Decimos que* $\mathbf{R}$ ***es representable en*** Nat_E *syss hay una fórmula* C *del lenguaje de la aritmética con a lo sumo* $x_1, ..., x_n$ *libres, tal que:*

$$Si \quad \langle \mathbf{x}_1, ..., \mathbf{x}_n \rangle \in \mathbf{R} \quad entonces \quad C\left(\frac{\sigma^{\mathbf{x}_1}}{x_1}, ..., \frac{\sigma^{\mathbf{x}_n}}{x_n}\right) \in \mathbf{CON}\left(Nat_E\right)$$

$$Si \quad \langle \mathbf{x}_1, ..., \mathbf{x}_n \rangle \notin \mathbf{R} \quad entonces \quad \neg C\left(\frac{\sigma^{\mathbf{x}_1}}{x_1}, ..., \frac{\sigma^{\mathbf{x}_n}}{x_n}\right) \in \mathbf{CON}\left(Nat_E\right)$$

Definición 122 *Una fórmula* C *del lenguaje de la aritmética con a lo sumo* $x_1, ..., x_n$ *libres, está* ***numeralmente determinada*** *por* Nat_E *syss para cada* n*-tupla* $\langle \mathbf{x}_1, ..., \mathbf{x}_n \rangle \in \mathbb{N}$ *se cumple una de estas dos cosas:*

$$Nat_E \vdash C\left(\sigma^{\mathbf{x}_1} c, ..., \sigma^{\mathbf{x}_n} c\right)$$
$$Nat_E \vdash \neg C\left(\sigma^{\mathbf{x}_1} c, ..., \sigma^{\mathbf{x}_n} c\right)$$

Fácilmente se demuestra que una fórmula C representa una relación $\mathbf{R}$ en $\mathbf{CON}\left(Nat_E\right)$ si y sólo si:

1. C esta numeralmente determinada por Nat_E .
2. C define a $\mathbf{R}$ en $\mathcal{N}$.

Así mismo se demuestra el teorema principal, que establece que las relaciones representables en una teoría consistente y axiomatizables son decidibles. De este teorema se sigue lo siguiente

Corolario 123 *Toda relación representable en una teoría consistente y finitamente axiomatizable es decidible.*

3.6. Aritmetización de la sintaxis

El objetivo de esta sección es introducir el instrumental necesario para demostrar en la próxima el teorema de incompletud de Gödel; es decir, introduciremos la denominada gödelización del lenguaje.

En especial, veremos:

Primero : Ciertas afirmaciones sobre fórmulas pueden convertirse en afirmaciones sobre números.

Segundo : Algunas de estas afirmaciones sobre números (hechas en metalenguaje) pueden retrotraducirse al lenguaje formal. Además, la teoría $\mathbf{CON}(Nat_E)$ es suficientemente potente como para demostrar en ella las traducciones obtenidas.

Esto nos permitirá construir fórmulas que aunque aparentemente expresan proposiciones sobre números, indirectamente hablan de fórmulas, incluso de sí mismas. En la sección siguiente explotaremos esta capacidad autorreflexiva al demostrar resultados fundamentales sobre indefinibilidad e indecidibilidad.

Para dar cuenta del apartado **Primero** lo que haremos es asignar números a los signos del lenguaje y, sobre ellos, números de Gödel a cada expresión, cada conjunto de expresiones e incluso a cada sucesión de expresiones —como por ejemplo, una deducción—. Lo más interesante de esta asignación de números es que es biunívoca; no sólo sabemos asignar un número de Gödel a cada expresión, sino también, dado un número de Gödel, sabemos exactamente a qué formula corresponde. Esta última propiedad se debe a que los números de Gödel son productos de potencias de primos y la descomposición de un número en factores primos es única.

Para desarrollar el apartado **Segundo** lo que hacemos es demostrar que algunas relaciones y funciones definidas sobre números naturales, que tienen que ver con números de Gödel, son representables en $\mathbf{CON}(Nat_E)$. En especial, veremos que el conjunto de los números de Gödel de las deducciones a partir de un conjunto finito de axiomas es representable en $\mathbf{CON}(Nat_E)$.

Funciones representable: Catálogo

Utilizando el concepto de función representable en $\mathbf{CON}(Nat_E)$, estableceremos los resultados siguientes:

1. La función del siguiente es representable
2. Toda función constante es representable
3. La función proyección es representable
4. Adición, multiplicación y exponenciación son representables

Además de esto demostramos que la clase de las funciones representables está cerrada bajo composición y que es posible encontrar una fórmula que represente al denominado *operador mínimo-cero*.

Números de Gödel

En primer lugar, se asignan números distintos a todos los signos del lenguaje aritmético; es decir, a los signos lógicos, a las variables y a los functores y relatores específicos que representan al: cero, siguiente, suma, producto, exponenciación, menor que. Sea h dicha asignación de números a los signos del lenguaje.

Además de eso, dada una expresión cualquiera

$$\varepsilon := \varrho_0\varrho_1...\varrho_n$$

definimos su correspondiente número de Gödel $\#\varepsilon$ así:

$$\#\varepsilon = \triangleleft h(\varrho_0)...h(\varrho_n)\triangleright = p_0^{h(\varrho_0)+1}...p_n^{h(\varrho_n)+1}$$

Por otra parte, a un conjunto de expresiones le asignamos el conjunto de sus números de Gödel y a una sucesión de expresiones le asignamos también un número de Gödel; a saber,

$$G(\langle\alpha_1...\alpha_n\rangle) = \triangleleft\#\alpha_1...\#\alpha_n\triangleright$$

Hecho esto, se debe probar que son representables en $\mathbf{CON}(Nat_E)$ una gran cantidad de funciones, relacionadas con números de Gödel, que necesitaremos en la demostración del teorema de incompletud. Entre otras, las funciones codificadoras y descodificadoras, los conjuntos de los números de los axiomas y de las deducciones, etc.

Teoremas

Utilizando la maquinaria de la aritmetización se puede demostrar que toda relación recursiva es representable en $\mathbf{CON}(Nat_E)$. Como se recordará, nosotros habíamos definido el ser recursiva diciendo que tenía que ser representable en alguna teoría consistente y finitamente axiomatizable. Lo que aquí se demuestra es que nos podemos limitar siempre a $\mathbf{CON}(Nat_E)$, pues del hecho de que lo sea en cualquier teoría de las características antes dichas, se sigue que lo es en la prepotente teoría $\mathbf{CON}(Nat_E)$.

Teorema 124 *Una relación es recursiva syss es representable en* $\mathbf{CON}(Nat_E)$

Corolario 125 *Toda relación recursiva es definible en* $\mathcal{N}$

Del teorema anterior se sigue fácilmente la equivalencia entre recursividad —ser representable en una teoría consistente y finitamente axiomatizable— y ser representable en $\mathbf{CON}(Nat_E)$, ya que $\mathbf{CON}(Nat_E)$ es consistente y finitamente axiomatizable.

Además, usando resultados anteriores, se verá que toda relación recursiva es definible en $\mathcal{N}$.

Para finalizar demostraremos que si el conjunto de los números de Gödel de un conjunto de axiomas es recursivo, y sus consecuencias semánticas constituyen una teoría completa, entonces el conjunto de sus números de Gödel es también recursivo.

Teorema 126 *Si* $\#\Gamma$ *es recursivo y* $\mathbf{CON}(\Gamma)$ *es completo, entonces* $\#\mathbf{CON}(\Gamma)$ *es recursivo.*

3.7. Teorema de incompletud de Gödel

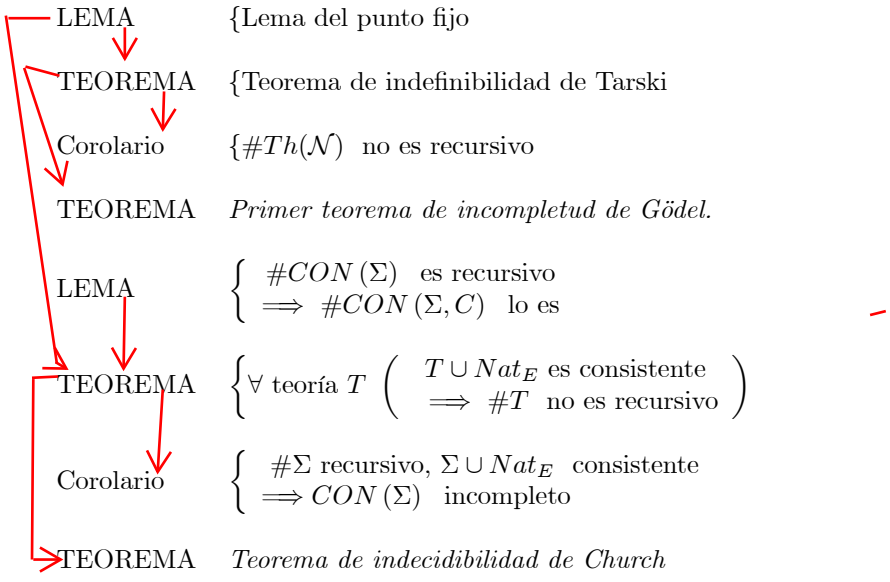
Ahora veremos cómo la maquinaria introducida en el tema de Aritmetización de la sintaxis funciona para producir los teoremas limitativos más famosos de toda la lógica: el de *incompletud de Gödel*, el de *indefinibilidad de Tarski* y el de *indecidibilidad de Church*. Para demostrarlos se usa el lema del punto fijo o de la autorreferencia, que afirma lo siguiente:

Lema 127 *(punto fijo) Dada una fórmula B con una variable libre es siempre posible encontrar una fórmula D tal que*

$$Nat_E \vdash D \leftrightarrow B(\sigma^{\#D}c) \tag{3.1}$$

Dicha fórmula D aunque al igual que B no sea más que una fila de signos del lenguaje de la aritmética, que al ser interpretadas en $\mathcal{N}$ hablan sencillamente de números, podemos pensar que habla acerca de sí misma y que indirectamente dice: *"B es una afirmación verdadera sobre mí"*. El motivo que nos permite pensar en la autorreferencia es que nosotros hemos asignado números a las fórmulas y por lo tanto D, al hablar de su número, habla de sí misma.

Procederé como indica el siguiente organigrama:



Teorema de Tarski, Gödel y Church

Demostraremos en primer lugar el teorema del punto fijo. A partir de él es fácil demostrar el teorema de indefinibilidad de Tarski.

Teorema 128 *(indefinibilidad de Tarski) El conjunto* $\#Th(\mathcal{N})$ *no es definible en* $\mathcal{N}$.
Demostración. *Para demostrar este teorema supondremos que el conjunto* $\#Th(\mathcal{N})$ *fuera definible mediante una fórmula* B. *Aplicando el teorema del punto fijo a* $\neg B$ *encontramos una fórmula* D *equivalente a* $\neg B\left(\sigma\ ^{\#_D}c\right)$. *En particular*

$$\mathcal{N} \Vdash D \leftrightarrow \neg B(\sigma^{\#_D}c) \tag{3.2}$$

¿Qué dice esta fórmula?
Si B *define* $\#Th(\mathcal{N})$, *entonces* D *dice indirectamente* "yo soy falsa". *Entonces,* D *es verdadera en* $\mathcal{N}$ *si y sólo si* $B(\sigma^{\#_D}c)$ *es falsa en* $\mathcal{N}$

$$\mathcal{N} \Vdash D \quad syss \quad \mathcal{N} \nVdash B(\sigma^{\#_D}c)$$

De donde fácilmente se obtiene una contradicción tanto cuando D *sea verdadera en* $\mathcal{N}$ *como en el caso en que sea falsa. Por consiguiente,* B *no puede definir a* $\#Th(\mathcal{N})$. ■

Corolario 129 $\#Th(\mathcal{N})$ *no es recursivo.*
Demostración. *Porque los conjuntos recursivos son definibles en* $\mathcal{N}$ ■

Aquí se podrá utilizar el teorema de indefinibilidad de Tarski para demostrar el teorema de incompletud de Gödel, que expresaremos así:

Teorema 130 *(Incompletud de Gödel) Si* $\Gamma \subseteq Th(\mathcal{N})$ *y* $\#\Gamma$ *es recursivo, entonces* $\mathbf{CON}(\Gamma)$ *no es una teoría completa. —Esto es, no hay una axiomatización completa de* $Th(\mathcal{N})$
Demostración. *Si* $\Gamma \subseteq Th(\mathcal{N})$ *también* $\mathbf{CON}(\Gamma) \subseteq Th(\mathcal{N})$. *Si fuera una teoría completa, valdría la igualdad. Pero, si fuera completa,* $\#\mathbf{CON}(\Gamma)$ *sería recursivo y sabemos que* $\#Th(\mathcal{N})$ *no es recursivo.* ■

De hecho, el enunciado original del Teorema de Gödel no es éste y en su demostración no se utiliza el teorema de Tarski. Él en vez de definir a las funciones recursivas en la forma en que lo hacemos nosotros, las definió directamente, como aquellas que se obtienen por composición y recursión a partir de unas muy simples, las denominadas recursivas primitivas.

Esta demostración tiene la ventaja de ser más simple y, desde el punto de vista docente, se puede abordar directamente después de un curso de lógica de primer orden.

Para finalizar este apartado se demuestra el teorema de la indecidibilidad de $\mathbf{CON}(Nat_E)$ y el teorema de Church (1936) que afirma que el conjunto de las sentencias válidas en el lenguaje de $\mathcal{N}$ no es recursivo, puesto que el de sus números de Gödel no lo es. La idea es probar:

Lema 131 *Si*

$$\#\mathbf{CON}(Nat_E)$$

es recursivo, entonces

$$\#\mathbf{CON}(Nat_E \cup \{D\})$$

también lo es

Teorema 132 *(indecidibilidad fuerte de* $\#\mathbf{CON}(Nat_E)$*). Sea* Γ *una teoría —esto es, cerrada bajo consecuencia—.*
Si $\Gamma \cup \#\mathbf{CON}(Nat_E)$ *es consistente, entonces* $\#\Gamma$ *no es recursivo.*

Corolario 133 *Si* $\#\Gamma$ *es recursivo y* $\Gamma \cup Nat_E$ *es consistente, entonces* $\mathbf{CON}(\Gamma)$ *no es una teoría completa.*

Este corolario es una versión del teorema de Gödel en donde las sentencias en vez de ser verdaderas en $\mathcal{N}$ son consistentes con Nat_E.

Teorema 134 *(indecidibilidad de Church*[10]*) El conjunto de los números de Gödel de las fórmulas válidas (en el lenguaje de la aritmética) no es recursivo.*
Demostración. *En el teorema de la indecidibilidad fuerte de* $\#\mathbf{CON}(Nat_E)$ *tomamos como conjunto* Γ *al de las fórmulas válidas.* ■

En cualquier caso, haremos que se perciba el clima emocional en el que el teorema de Gödel vino a caer[11]. Desde la publicación de los trabajos de Frege, y sobre todo de Russell y Whitehead, los lógicos pensaban que toda demostración intuitivamente correcta podía reconstruirse en el cálculo lógico. También se pensaba que añadiendo axiomas adecuados al cálculo podían construirse teorías —en el sentido lógico— en las que toda sentencia matemática intuitivamente correcta podría deducirse. Como ya comenté en la sección 1.1.2, la idea de Hilbert era explotar al máximo la naturaleza finita de las pruebas para proporcionar una fundamentación de la matemática. El proceso consistía en axiomatizarla y demostrar que la teoría es: *Consistente*, *Completa* y *Decidible.*

No es de extrañar que en este clima el teorema de Gödel fuera considerado como la prueba del fracaso del programa de Hilbert.

3.8. Máquinas de Turing

Hay varios caminos distintos para llegar a la noción de recursividad. Una versión mediante máquinas computadoras imaginarias fue concebido por Turing en 1936.

Intuitivamente una máquina de Turing consiste en una máquina abstracta (una caja negra), una cabeza lectora, un mecanismo de escritura y una cinta infinita en ambas direcciones. La máquina puede estar en cualquiera de los

[10] Una prueba más detallada de este teorema aparece en el apéndice 3.10.
[11] A este respecto el artículo de Gandy [12] es fantástico.

estados internos, que constituyen un conjunto finito. La cinta está dividida en cuadros, llamados cuadros de la cinta.

$\cdots$ □□□ $\cdots$

L (*izquierda*) $\wedge$ (*cabeza lectora*) □ (*máquina*) (*derecha*) **R**

La máquina funciona paso a paso. En un paso dado realiza una acción que depende del estado en el que está y de la inscripción que encuentra en el cuadro en el que está la cabeza lectora. Sólo se pueden escribir dos signos: 0 y 1. Todos los cuadros, excepto un número finito de ellos, tienen ceros. Estas son las acciones que la máquina puede hacer:

1. Escribir 0 en el cuadro (borrar primero lo que hay allí)
2. Escribir 1 en el cuadro (borrar primero lo que hay allí)
3. Ir al cuador inmediato de la derecha, **R**.
4. Ir al cuador inmediato de la izquierda, **L**.
5. Pararse, **S**.

Para expresar esto rigurosamente:

Definición 135 *Una máquina de Turing es una matriz de la forma:*

c_1	0	v_1	d_1
c_1	1	v_2	d_2
c_2	0	v_3	d_3
c_2	1	v_4	d_4
$\vdots$	$\vdots$	$\vdots$	$\vdots$
c_n	0	v_{2n-1}	d_{2n-1}
c_n	0	v_{2n}	d_{2n}
$\vdots$	$\vdots$	$\vdots$	$\vdots$
c_m	0	v_{2m-1}	d_{2m-1}
c_m	1	v_{2m}	d_{2m}
Estados de la máquina	*Estímulos que recibe*	*Pasos que realiza*	*Estados en que queda*

$$c_1, ..., c_m \in \omega,\ v_1, ..., v_{2m} \in \{0, 1, 2, 3, 4\},\ d_1, ..., d_{2m} \in \{c_1, ..., c_m\}.$$

Una fila

$$c_i \quad \varepsilon \quad v_j \quad d_j$$

de la matriz proporciona la siguiente información: cuando la máquina está en el estado c_i y encuentra ε en la cinta, realiza la acción v_j y pasa al estado d_j. Las acciones que realiza son:

$v_j = 0$	escribir 0 en el recuadro
$v_j = 1$	escribir 1 en el recuadro
$v_j = 2$	mover la cinta un cuadro hacia la derecha
$v_j = 3$	mover la cinta un cuadro hacia la izquierda
$v_j = 4$	parar

Es fácil *construir* algunas máquinas sencillas, tales como la máquina de ir hacia la izquierda —se va un cuadro hacia la izquierda y se para—, la de ir hacia la derecha, la de sumar, la de restar, etc.

Cuando se introduce el concepto de función recursiva primitiva se demuestra que todas ellas son computables mediante máquinas de Turing; es decir:

Primero : Todas las funciones iniciales son Turing-computables

Segundo : Los procedimientos de recursión primitva y composición aplicados sobre funciones Turing-computables producen funciones Turing-computables.

3.9. La Jerarquía Aritmética

En matemáticas es bastante común la introducción de mediciones. Así, en teoría de conjuntos vemos que entre los conjuntos infinitos los hay que son *"más infinitos que otros"*. Para precisar la, por así llamar, medida de infinitud, se define la noción de cardinalidad. Normalmente el concepto métrico permite establecer un orden en el ámbito de estudio, tal vez sólo un orden parcial.

Ahora nos ocuparemos de los números naturales. Este caso no es tan particular y restringido como pudiera parecer, pues vimos en temas anteriores que gracias a la aritmetización podemos hablar de números y conjuntos de números refiriéndonos a fórmulas y conjuntos de fórmulas. En principio estamos en disposición de extrapolar los resultados a cualquier entorno que pueda ser codificado mediante números. En especial, los resultados que obtengamos se aplicarán no sólo a conjuntos de números, sino también a conjuntos de fórmulas.

En el ámbito de los números naturales hemos establecido ya una partición: un conjunto puede ser recursivo o no serlo.

Ejemplos de conjuntos recursivos tenemos muchos: el conjunto de los números pares, el conjunto de los números de Gödel de los términos, de las fórmulas, etc.

Ejemplos de conjuntos no recursivos tenemos el de la teoría de los naturales $\#Th(\mathcal{N})$ —es decir, el de los números de Gödel de las sentencias verdaderas en $\mathcal{N}$—, el de las fórmulas válidas en el lenguaje de los naturales —es decir, el conjunto de los números de Gödel de las sentencias lógicamente válidas—, etc. Puesto que el conjunto de los conjuntos recursivos es numerable, en $\wp(\mathbb{N})$ que es supernumerable, casi todos los conjuntos no son recursivos.

Otra medición que se puede establecer en el ámbito de los conjuntos de naturales es la de los grados de irresolubilidad, que indican lo lejos que está un conjunto de ser recursivo.

También se puede usar como escala de medida la definibilidad, y aplicarla a conjuntos que sean definibles en $\mathcal{N} = \langle \mathbb{N}, 0, s, +, \cdot \rangle$ mediante fórmulas de primer orden —no nos ocuparemos de los conjuntos definibles mediante fórmulas de segundo orden—. Recordemos que un conjunto $\mathbf{R} \subseteq \mathbb{N}$ es definible en $\mathcal{N}$ si y sólo si hay una fórmula C, con a lo sumo una variable libre tal que:

$$\mathbf{R} = \{\mathbf{n} \in \mathbb{N} \mid \mathcal{N} \ [\mathbf{n}/x] \Vdash C\}$$

Esto es,

$$\mathbf{R} = \{\mathbf{n} \in \mathbb{N} \mid \ \mathcal{N} \Vdash C\,(\sigma^{\mathbf{n}} c)\}$$

Por consiguiente, $\mathbf{R}$ contiene justamente a los elementos de $\mathbb{N}$ que hacen verdadera a C en $\mathcal{N}$. Para relaciones n-arias usaremos fórmulas con a lo sumo n variables libres. Naturalmente, puesto que la clase de las fórmulas del lenguaje de $\mathcal{N}$ es numerable, el conjunto de los subconjuntos de $\mathcal{N}$ que son definibles es sólo numerables. No obstante y pese a que, como ocurría con la recursión, la mayoría de los elementos de $\wp(\mathbb{N})$ no son definibles, los más interesantes suelen ser definibles. La razón es que si estamos interesados en una cierta relación $\mathbf{R}$ es porque sabemos qué es $\mathbf{R}$ y este conocimiento normalmente nos permite definir a $\mathbf{R}$ en $\mathcal{N}$.

Como ejemplos de relaciones definibles en $\mathcal{N}$ podemos citar: la relación de orden, el conjunto de los números primos, el de los números pares, la relación de divisibilidad y la de exponenciación. En efecto, el orden se define mediante:

$$C := (x = y \vee \exists z \ x + \sigma z = y)$$

mientras que los números primos se definen mediante:

$$D := (x \neq \sigma c \wedge \forall yz \, (x = y.z \rightarrow y = \sigma c \vee z = \sigma c))$$

Como ejemplos de relación no definible[12] podemos citar el de los números de Gödel de las sentencias verdaderas en $\mathcal{N}$.

Si un conjunto es definible en $\mathcal{N}$, entonces nosotros queremos saber en qué grado o medida lo es.

¿Cómo se mide esto?

Para empezar, llamamos *aritméticas* a las relaciones definible en $\mathcal{N}$ mediante fórmulas de primer orden y *analíticas* a las definibles con fórmulas de segundo orden. En la clase de los conjuntos aritméticos podemos intentar establecer una medición —no binaria— de la definibilidad. Puesto que la definiblidad se establece mediante fórmulas, es natural fijarse en ellas para establecer dicha medición. Puede pensarse que una relación definible mediante una fórmula cortita o sin cuantificadores es más definible que otra que precise una fórmula complicadísima. La idea va un poco por ahí, pero es más sutil.

Por supuesto, las relaciones o funciones destacadas en el sistema son las más definibles, ¡tienen incluso nombre propio!; por ejemplo, la suma y el producto. Nosotros hemos escogido un sistema $\mathcal{N}$ en el que la exponenciación no era una

[12] En el capítulo 2 aparece un ejemplo muy sencillo (el 78) de conjunto no definible.

operación destacada, pero esta elección fue bastante arbitraria; podríamos haber escogido un sistema en el que la exponenciación fuera también una operación destacada y no hubiera cambiado casi nada. Sin embargo, la fórmula que define a la exponenciación en un sistema que no la tenga como función destacada es bastante larga y por lo tanto tendríamos a la suma y al producto en distinto lugar en la ordenación por definibilidad que a la exponenciación. Como digo, esto no parece satisfactorio y nos gustaría que la suma, el producto y la exponenciación estuvieran en el mismo paquete; queremos que la medida de la definibilidad esté exenta de estas arbitrariedades. Es por ello por lo que mediremos la definibilidad módulo recursividad.

El siguiente teorema, demostrado por Gödel, puede ser aplicado.

Teorema 136 *Toda relación recursiva sobre* $\mathbb{N}$ *es aritmética.*

Las relaciones aritméticas son las definibles en un sistema con universo $\mathbb{N}$ y que tuviera destacada a todas las relaciones recursivas; esta es la forma natural de medir definibilidad módulo recursividad. Toda relación definible en este sistema lo es, en particular, mediante una fórmula en forma normal prenexa. Lo que se hace es utilizar la alternancia de cuantificadores en dicha fórmula prenexa como criterio métrico.

La jerarquía aritmética que se construye sobre esta base es así:

Δ_1 : relaciones recursivas

Σ_1 : recursivamente enumerables

Π_1 : relaciones cuyo complementario está en Σ_1

En general, por recursión se define,

Las relaciones de Σ_n son dominios de relaciones de Π_{n-1} y las relaciones de Π_n son relaciones cuyo complemento está en Σ_n.

$$\Delta_n = \Sigma_n \cap \Pi_n$$

$$\Sigma_1 = \{\overrightarrow{a} \mid \exists b \ \langle \overrightarrow{a}, b\rangle \in \mathbf{R}\} \quad \mathbf{R} \text{ es recursiva}$$

$$\Pi_1 = \{\overrightarrow{a} \mid \forall b \ \langle \overrightarrow{a}, b\rangle \in \mathbf{R}\} \quad \mathbf{R} \text{ es recursiva}$$

$$\Sigma_2 = \{\overrightarrow{a} \mid \exists c \forall b \ \langle \overrightarrow{a}, b, c\rangle \in \mathbf{R}\} \quad \mathbf{R} \text{ es recursiva}$$

$$\Pi_2 = \{\overrightarrow{a} \mid \forall c \exists b \ \langle \overrightarrow{a}, b, c\rangle \in \mathbf{R}\} \quad \mathbf{R} \text{ es recursiva}$$

Gráficamente la situación es así (ver figura: 3.1)

3.10. Apéndice: Indecidibilidad de *FOL*.

La indecidibilidad de la lógica de primer orden no se demuestra directamente, se suele utilizar algún sistema finitamente axiomatizable del que ya sabemos que es indecidible. Church en su demostración de indecidibilidad hace uso de

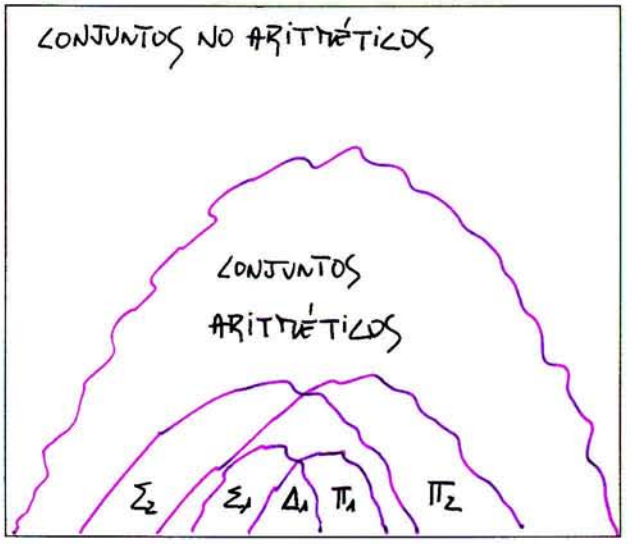


Figura 3.1: *Elementos de* $\wp(\mathbb{N})$

los resultados de otro artículo suyo, normalmente ahora solemos usar el sistema **Q** —denominada la *aritmética de Robinson*—, una subteoría de la aritmética que tiene sólo siete axiomas; a saber:

1. $\mathbf{Q}_1 := \forall xy(\sigma x = \sigma y \rightarrow x = y)$
2. $\mathbf{Q}_2 := \forall x(c \neq \sigma x)$
3. $\mathbf{Q}_3 := \forall x(x \neq c \rightarrow \exists y(x = \sigma y)$
4. $\mathbf{Q}_4 := \forall x(x + c = x)$
5. $\mathbf{Q}_5 := \forall xy(x + \sigma y = \sigma(x + y))$
6. $\mathbf{Q}_6 := \forall x(x \cdot c = c)$
7. $\mathbf{Q}_7 := \forall xy(x \cdot \sigma y = (x \cdot y) + x)$

El aparato conceptual introducido por Gödel nos permite decir que un conjunto de fórmulas es decidible, si el conjunto de los números de Gödel de sus elementos es recursivo. Así, una teoría es decidible si y sólo si el conjunto de los números de Gödel de sus teoremas es recursivo o, de forma equivalente, es recursiva su función característica. Dada una teoría decidible hay un método efectivo para decidir si una sentencia es un teorema: se halla su número de Gödel y se usa ese número como argumento en la función característica. La sentencia será un teorema si y sólo si el valor es 1. Por el contrario, si una teoría no es decidible, y si aceptamos la tesis de Church, no habrá un método efectivo para

decidir si una sentencia es un teorema. Si lo hubiera, la función característica de los números de Gödel de sus teoremas sería efectivamente computable y, por lo tanto, recursiva, usando la tesis de Church.

La prueba de que $\mathbf{Q}$ no es decidible [13] usa los recursos de Gödel —es decir, gödelización y diagonalización— y ciertas propiedades de $\mathbf{Q}$ —en particular, que todas las funciones recursivas son representables en $\mathbf{Q}$—.

El argumento de Church se puede presentar como sigue:

1. Sea
$$\theta := \wedge_{1 \leq i \leq 7} \mathbf{Q}_i$$
la conjunción de los axiomas de $\mathbf{Q}$

2. Por el teorema de la deducción tenemos que:
$$\mathbf{Q} \vdash C \text{ syss } \vdash \theta \rightarrow C$$
(Haciendo uso de los teoremas ya disponibles de completud y corrección, podemos parafrasear este resultado diciendo que un test de validez equivale a un test de pertenencia a la teoría $\mathbf{Q}$: para decidir si C es un teorema, hagamos el test a $\theta \rightarrow C$.)

3. Designemos mediante $\#A$ al número de Gödel de la fórmula A. Sea f una función que a cada número natural n le asigna $\#(\theta \rightarrow D_n)$ —donde, $\#D_n = n$—. Esta función es claramente recursiva. Para cada fórmula C, de número de Gödel n,
$$f(n) = \#(\theta \rightarrow C)$$

4. Sea
$$P_0 = \{\#G \mid\vdash G\}$$
es decir, el conjunto de los números de Gödel de los teoremas lógicos de la lógica de primer orden

5. Sea
$$P = \{\#C \mid \mathbf{Q} \vdash C\}$$
Por lo tanto,
$$P = \{\#C \mid\vdash \theta \rightarrow C\} = \{n \mid f(n) \in P_0\}$$
Pero P no es recursivo, puesto que es el conjunto de los números de Gödel de los teoremas de $\mathbf{Q}$, y sabemos que $\mathbf{Q}$ no es decidible

6. Por consiguiente, P_0 tampoco es recursivo

7. La conclusión es que la lógica de primer orden no es decidible; no hay test de validez.

[13] Una demostración de este teorema puede leerse en Boolos [**?**], en la página 175, basada en resultados demostrados en el libro. Desgraciadamente, este gran lógico también murió recientemente, este mismo año de 1996.

Bibliografía

[1] Alonso, E. [1996]. *Curso de Teoría de la Computación.* Serie Cuadernos de Trabajo. Ediciones de la UAM. Madrid.

[2] Alonso, E. [2002]. *Teoría de la Computación.* Summa Logicae en el siglo XXI en http://logicae.usal.es

[3] Barwise, J. [1977]. *Handbook of Mathematical Logic.* North Holland Publishing Company. Amsterdam. Holanda.

[4] Beth, E. [1965]. *The foundations of mathematics.* Amsterdam: North Holland.

[5] Boolos, G. [1989]. *Computability and Logic.* Cambridge University Press. Cambridge. U.K.

[6] van Dalen, D. [2001]. *"Algorithms and Decision Problems: A Crash Course in Recursion Theory"* en [11]

[7] Davis, M. (ed.) [1965]. *The Undecidable. Basic Papers on Undecidable Propositions, Unsolvable Problems, and Computable Functions.* Raven Press, Hewlett, Nueva York.

[8] Davis, M. [1977]. *"Unsolvable Problems"*, en [3].

[9] Enderton, H. [1972]. *A Mathematical Introduction to Logic.* Academic Press. New York. USA

[10] Enderton, H. [1977]. *"Elements of Recursion Theory"* en [3]

[11] Gabbay, D y Guenthner, F. [2001]. *Handbook of Philosophical Logic* 2^{nd} edition. Kluwer Academic Publishers. Dordrecht. Holanda.

[12] Gandy, R. [1988] . *"The Confluence of Ideas in 1936".* (en Herken [14]).

[13] Heijenoort, J. ed. [1967]. *From Frege to Gödel.* Harvard University Press. Harvard. USA

[14] Herken, R. ed. [1988]. *The Universal Turing Machine: A Half-Century Survey.* Oxford University Press. Oxford. U.K.

[15] Hermes, F. [1969]. *Enumerability, Decidability, Computability.* North Holland Publishing Company. Amsterdam. Holanda.

[16] Nagel, E. y Newman, J.R. [1958]. *El Teorema de Gödel.* Ed. castellana de Adolfo Martín. Estructura y Función. Tecnos. Madrid, 1979.

[17] Odifreddi, P. [1989]. *Classical Recursion Theory***.** North Holland Publishing Company. Amsterdam. Holanda.

[18] Rogers, H. [1967]. *Theory of Recursive Functions and Effective Computability.* McGraw-Hill. New York. USA

[19] Shoenfield, J. R. [1967]. *Mathematical Logic.* Reading: Addison-Wesley Publishing Company.

[20] Shoenfield, J. R. [1995]. *"The mathematical work of S. C. Kleene".* **Bulletin of Symbolic Logic**. vol. 1. Núm 1. pp 9- 43.

[21] Smullyan, R. [1993]. *Recursion Theory for Metamathematics.* Oxford Logic Guides 22. Oxford: Oxford University Press.

[22] Tarski, A., Mostowski, A. y Robinson, R. M. [1953]. *Undecidable Theories.* Amsterdam: North Holland Publishing Company.

Capítulo 4

Teoría de la Demostración

4.1. Introducción

En lógica construimos demostraciones en cálculos deductivos: ellas y estos constituyen el objeto de estudio en *Teoría de la Demostración*.

No hemos dejado de estar interesados en la verdad —en sentido matemático; por ejemplo, la de un enunciado aritmético en el modelo de los naturales, o la validez o verdad en todo modelo— pero ahora contamos con un mecanismo para establecerla, el cálculo.

¿Por qué queremos un cálculo?

Hay muchas razones para querer un cálculo y no contentarnos con el mero concepto de consecuencia que proporciona la semántica, que es poco operativo. Se me ocurren las siguientes:

1. *"Implementar"* es más sencillo, y esto es algo difícil de concebir sin el soporte de un cálculo. (Estoy pensando en un demostrador de teoremas.)

2. Un cálculo es un *modelo mejor del proceso mental* que seguimos para extraer conclusiones de ciertas hipótesis que su correspondiente formulación semántica. (No vamos al universo matemático a verificar que cada modelo de las hipótesis lo es también de la conclusión, sino que, sin levantarnos de la silla, transformamos las hipótesis para obtener la conclusión.)

3. Se usan *menos recursos matemáticos* para demostrar en un cálculo que para hacerlo semánticamente.

4. *Convencer* es más sencillo

   y en especial,

5. No tendrás que demostrar que tu prueba lo es en efecto. (Hay incluso un *algoritmo que comprueba* si una deducción lo es conforme a las reglas del cálculo.)

Desde los griegos hasta nuestros días se han desarrollado muchos procedimientos de prueba: la propia silogística, los métodos axiomáticos, los de deducción natural, los de secuentes, los tableaux semánticos y los de resolución, entre otros.

Presentaré varios para poder compararlos, acompañados de algún ejemplo sencillo de deducción en ellos. Por razones obvias de extensión me limitaré en este capítulo a la lógica clásica de primer orden[1] sin igualdad, aunque muchas de las características que quisiera destacar ya aparecen en la proposicional, su decidibilidad la singulariza en exceso.

¿Qué resultados se establecen en Teoría de la Demostración?

Por descontado, si un cálculo se ha introducido como contrapartida sintáctica a la noción de verdad en todo modelo, lo primero que se plantea es si este objetivo se ha alcanzado; esto es, si contamos con los teoremas de corrección y completud. Puesto que estos teoremas ya han sido tratados en la sección 2.4 del capítulo 2 no serán abordados aquí.

¿Existe alguna diferencia entre tener una demostración de un teorema en un cálculo deductivo y saber que es verdadero?

No nos ocuparemos en este capítulo del teorema de incompletud de Gödel, que por cierto responde afirmativamente a la pregunta: hay una gran diferencia entre verdad y demostrabilidad. En un modelo dado, por ejemplo el de la aritmética

$$\mathcal{N} = \langle \mathbb{N}, 0, s, +, \cdot \rangle$$

cada enunciado ha de ser verdadero o falso, pero hay verdades indemostrables. El que haya de ser necesariamente verdadero o falso no significa que nosotros sepamos siempre dilucidarlo: éste es el caso de la *conjetura de Goldbach*[2]. De los resultados de incompletud nos ocupamos en capítulos aparte, los hay tanto de incompletud de teorías de primer orden como de cálculos deductivos de orden superior. Este primer resultado establece que la teoría de los números naturales no es completa, habiendo en $Th(\mathcal{N})$ verdades sin prueba[3].

Pero tanto de los resultados de Gödel como de la capacidad expresiva de la lógica de segundo orden se desprende que ésta es incompleta en este otro sentido: hay fórmulas válidas en SOL que no son demostrables[4].

Los teoremas mencionados, son hasta cierto punto externos: establecen la relación entre demostrabilidad y verdad —o validez—. Hay otros teoremas internos, propios, característicos de esta teoría: son los de *Eliminación del Corte* —el Hauptsatz de Gentzen—, teorema de Herbrand, teoremas de Robinson y teorema de Craig. De todos ellos me ocupo en la última sección de este capítulo; aunque no los demuestro, los enuncio y comento brevemente.

[1] Otros sistemas lógicos, para lógicas no clásicas aparecen en la Parte II y en la Parte III de este volumen.

[2] En la novela *El tío Petrus y la conjetura de Goldbach* su protagonista, tras años de infructuosos intentos de demostrarla, se consuela pensando que la misma es una de esas verdades indemostrables que Gödel señalara.

[3] Lo vimos en 3.7

[4] Se demostratrá en 10.5

Otro aspecto que también está directamente relacionado con las cuestiones aquí planteadas es el siguiente, *¿cómo programamos a una máquina para que use alguno de nuestros cálculos? ¿son implementables?*

Está claro que si la lógica no es decidible, y esto es lo que nos sucede en primer orden, no hay ningún algoritmo que resuelva la pertenencia a VAL. Pero de la misma manera que nosotros probamos teoremas *"a mano"*, usando el cálculo, podríamos enseñar a una máquina a hacerlo. De alguna forma deberíamos trasladarle nuestra experiencia y pericia. Los cálculos que mejor se prestan para ser implementados son los de resolución y los de tableaux, razón por la que los incluyo aquí. En el libro de Melving Fitting [6] se ha elegido Prolog como lenguaje de programación y también está escrito en él el programa *Winke*[5], que implementa una modificación del cálculo de tableaux con corte, que lo convierte en más efectivo.

En teoría automática de teoremas se persiguen dos objetivos: (1) demostrar teoremas —no triviales— y (2) hacerlo automáticamente. La experiencia nos enseña que desgraciadamente estos objetivos son bastante incompatibles; desde los años sesenta se han ido desarrollando demostradores y se observa que cuanto más complejo es el teorema, más tarda el demostrador y el crecimiento tiende a ser exponencial. Así que lo interesante reside en proporcionar al programa procedimientos heurísticos, trasmitirle nuestras estrategias, intuiciones y conocimiento. Sucede con frecuencia que éstas dependen del tema tratado, por lo que lo natural sería diseñar sistemas expertos: de *teoría de grupos* o de *geometría descriptiva*, pongamos por caso.

Gran parte del trabajo hecho en demostración automática se basa en *Resolución*, un método desarrollado por Robinson en los años 60, que está basado en los teoremas de Herbrand.

4.2. Silogística

Empecemos por Aristóteles, que fue el primero que de manera sistemática, trató con una cierta profundidad la relación que se establece entre las sentencias que forman parte de una razonamiento, observando que para estudiar la naturaleza de la deducción hace falta analizar primero la estructura de las que constituyen sus hipótesis y su conclusión. En la lógica tradicional, de Aristóteles a Leibniz, incluso en Boole, ésta se toma de la gramática de las lenguas naturales[6]; es decir, una sentencia se analiza en términos de *sujeto* S y *predicado* P.

Formas

Se distinguen cuatro formas típicas de proposiciones: A, E, I y O
A : Todo S es P (universal afirmativa)

[5] Este programa lo utilizo en clases prácticas, consultad http://www.dcs.kcl.ac.uk/staff/endriss/WinKE/

[6] Nosotros ahora utlilizamos un análisis más rico, basado en la concepción de Frege.

E : Ningún S es P (universal negativa)
I : Algún S es P (particular afirmativa)
O : Algún S no es P (particular negativa)

Estas cuatro formas aparecen relacionadas en el denominado cuadro de Boecio (ver figura: 4.1), así:

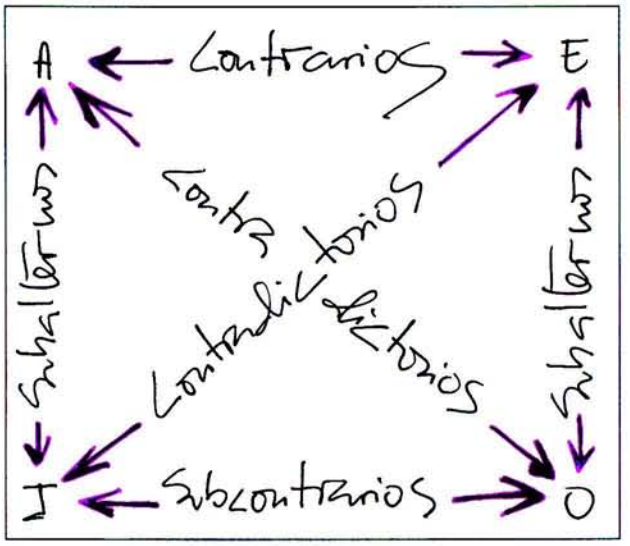


Figura 4.1: *Cuadro de Boecio*

Silogismo

El silogismo categórico es una estructura de proposiciones, propia de la lógica tradicional aristotélica, que se caracteriza:

1. Por tener dos premisas (mayor y menor) y una conclusión
2. Por tener sólo tres términos: Mayor P, medio M y menor S

Figuras

Se llaman figuras del silogismo a las distintas posiciones que ocupa el término medio —que desaparece de la conclusión— y que se expresan así:

Primera figura	Segunda figura	Tercera figura	Cuarta figura
$M \quad P$	$P \quad M$	$M \quad P$	$P \quad M$
$S \quad M$	$S \quad M$	$M \quad S$	$M \quad S$
$S \quad P$	$S \quad P$	$S \quad P$	$S \quad P$

Modos

El modo de un silogismo resulta de la combinación de las formas que contiene. Para cada figura hay sesenta y cuatro modos posibles. Una forma de construirlos es la que sigue:

$$\begin{array}{cccc} A & E & I & O \\ A & A & A & A \\ \hline A & A & A & A \end{array} \qquad \begin{array}{cccc} A & E & I & O \\ E & E & E & E \\ \hline A & A & A & A \end{array}$$

$$\begin{array}{cccc} A & E & I & O \\ I & I & I & I \\ \hline A & A & A & A \end{array} \qquad \begin{array}{cccc} A & E & I & O \\ O & O & O & O \\ \hline A & A & A & A \end{array}$$

Como puede verse, todos los casos que hemos construido concluyen en A; de igual modo en la siguiente vuelta se construyen los que concluyen en E, luego en I y finalmente en O, obteniéndose los sesenta y cuatro modos. Como hay cuatro figuras, el resultado final es de 256 silogismos posibles. Por supuesto, no todos son válidos. La lógica tradicional selecciona de entre ellos a 24, a los que considera silogismos válidos, a muchos de los cuales se les atribuyeron nombres nemotécnicos en el medioevo. Son los siguientes:

Primera figura	Segunda figura	Tercera figura	Cuarta figura
B*A*RB*A*R*A*	C*E*SAR*E*	D*A*RAPT*I*	B*A*MAL*I*P
C*E*LAR*E*NT	C*A*M*E*STR*E*S	F*E*LAPT*O*N	C*A*M*EN*E*S*
D*A*R*II*	F*E*ST*I*N*O*	D*AT*IS*I*	D*I*MAT*I*S
F*E*R*IO*	B*A*RO*C*O	D*IS*AM*I*S	F*E*SAP*O*
AAI	*AEO*	BOCARD*O*	FR*ES*I*SO*N
EAO	*EAO*	F*ER*IS*O*N	*AEO*

Comentario 137 *Un ejercicio interesante es el de comprobar, uno a uno los silogismos, para enmendar, en lo posible la plana a los clásicos. Efectivamente, no todos los seleccionados son razonamientos válidos con los estándares actuales.*

Ejemplo 138 *Como ejemplo de razonamiento siguiendo este esquema, pongamos éste:*

Hipótesis 1. *Los misóginos son impresentables*

Hipótesis 2. *Los que guardan silencio ahora son misóginos*

Conclusión *Los que guardan silencio ahora son impresentables*

1. *Las tres proposiciones que contiene son, respectivamente, de la forma E A E. Es decir, su modo es*
   E
   A
   E
2. *Este silogismos es del tipo CELARENT. Por tratarse de un silogismo de la primera figura, tiene la forma*
   $$\begin{array}{ll} M & P \\ S & M \\ \hline S & P \end{array}$$
3. *Este es su esquema*
   $$\begin{array}{lll} M & E & P \\ S & A & M \\ \hline S & E & P \end{array}$$
4. *En la lógica actual se expresa así:*
   $$\{\forall x(Mx \to \neg Px), \forall x(Sx \to Mx)\} \vdash \forall x(Sx \to \neg Px)$$
   Donde:
   $Mx := x$ *es un misógino*
   $Sx := x$ *guarda silencio ahora*
   $Px := x$ *es presentable*
5. *Para demostrarlo podemos usar el cálculo deductivo de primer orden, o el visual de los diagramas de Venn*

Llevemos a un diagrama de Venn las hipótesis y la negación de la conclusión (ver figura: 4.2)

Por consiguiente (ver figura: 4.3)

4.3. Cálculos axiomáticos

El cálculo de Frege-Hilbert, más comúnmente denominado de Hilbert, o axiomático, se caracteriza por poseer un cierto número de axiomas —en realidad, esquemas axiomáticos— y de reglas. Los axiomas son fórmulas y las reglas son transformaciones de las fórmulas en fórmulas. Una *prueba* en éste —o en cualquier otro axiomático— se caracteriza como sucesión finita de fórmulas, cada una de las cuales es un axioma o se ha obtenido de las líneas anteriores mediante la aplicación de reglas. Una *prueba de una fórmula* A es una prueba en el sentido anterior, cuya última fórmula es A. Escribimos $\vdash_H A$ para denotarlo.

Los axiomas del cálculo proposicional son los siguientes:

Axioma 1 $A \to (B \to A)$

Axioma 2 $(A \to (A \to B)) \to (A \to B)$

Axioma 3 $(A \to B) \to ((B \to C) \to (A \to C))$

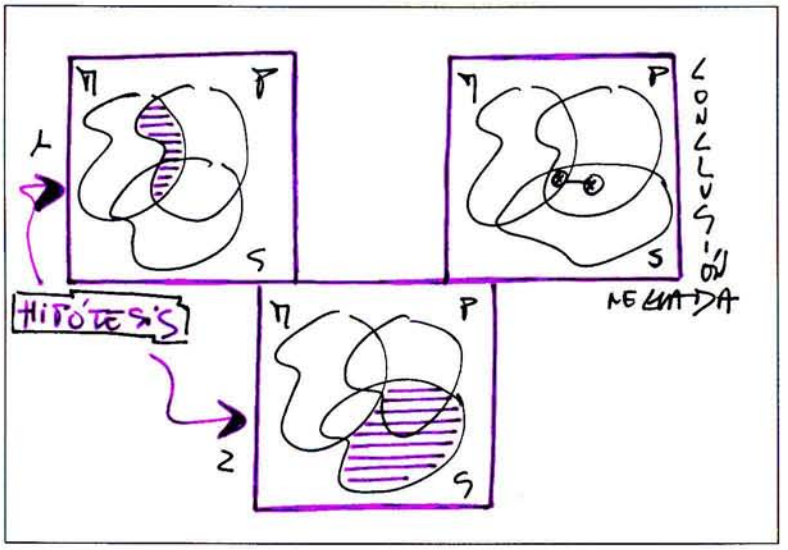


Figura 4.2: *Diagrama de hipótesis y negación conclusión*

Axioma 4 $(A \wedge B) \rightarrow A$

Axioma 5 $(A \wedge B) \rightarrow B$

Axioma 6 $(A \rightarrow B) \rightarrow ((A \rightarrow C) \rightarrow (A \rightarrow (B \wedge C))$

Axioma 7 $A \rightarrow (A \vee B)$

Axioma 8 $B \rightarrow (A \vee B)$

Axioma 9 $(A \rightarrow C) \rightarrow ((B \rightarrow C) \rightarrow ((A \vee B) \rightarrow C))$

Axioma 10 $(((A \rightarrow B) \wedge (A \rightarrow \neg B)) \rightarrow \neg A)$

Axioma 11 $\neg A \rightarrow (A \rightarrow B)$

Axioma 12 $\neg A \vee A$

A los que para la lógica de primer orden se suman

Axioma 13 $\forall x(C \rightarrow C\left(\frac{\tau}{x}\right))$

Axioma 14 $C\left(\frac{\tau}{x}\right) \rightarrow \exists x C x$

Habiéndose definido previamente la sustitución recursivamente y de tal manera que sólo se sustituyan variables libres.

La única regla del cálculo proposicional es *Modus Ponens:*

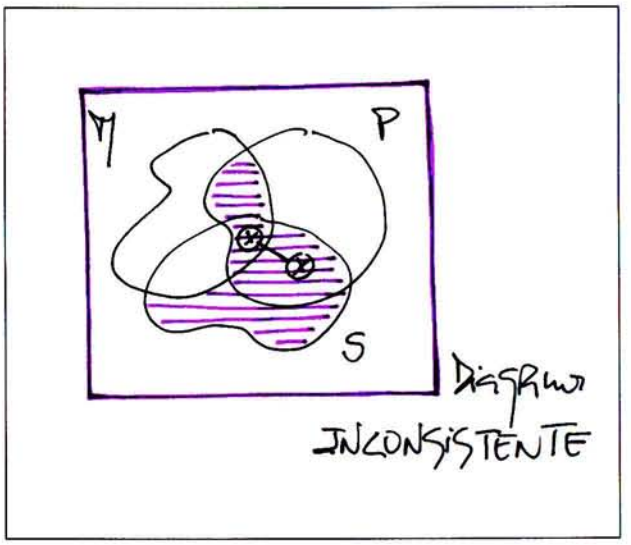


Figura 4.3: *Diagrama inconsistente, razonamineto válido*

Regla 1 Si $\vdash_H A$ y $\vdash_H A \to B$ entonces $\vdash_H B$

Para el de primer orden sin igualdad añadimos estas dos reglas

Regla 2 Si $\vdash_H B \to C\left(\frac{u}{x}\right)$ entonces $\vdash_H B \to \forall x C$

Regla 3 Si $\vdash_H A \to B$ entonces $\vdash_H \exists v(A \to B)$

—en la segunda regla la variable x no está libre en B; en la tercera, v no está libre en B—

Este es un cálculo axiomático para la lógica clásica, suprimiendo el **Axioma 12** obtenemos la lógica intuicionista, suprimiendo también el **Axioma 11** obtenemos la lógica minimal.

Ejemplo 139 *Una demostración de* $p \to p$ *es así:*

1. $p \to (p \to p)$ *Axioma 1*
2. $(p \to (p \to p)) \to (p \to p)$ *Axioma 2*
3. $p \to p$ *MP 1 y 2*

Ejemplo 140 *Una demostración de* $Rb \to (Rb \to \forall x\,(Rb \vee Qx))$ *es:*

1. $Rb \to Rb \vee Qx$ *Axioma 7*
2. $Rb \to \forall x\,(Rb \vee Qx)$ *Regla 2*
3. $(Rb \to \forall x\,(Rb \vee Qx)) \to (Rb \to (Rb \to \forall x\,(Rb \vee Qx)))$

4. $Rb \to (Rb \to \forall x\,(Rb \vee Qx))$ *MP 2 y 3*

Este cálculo está tomado de Gentzen, tal y como lo reproducen Bibel y Ender [2]. Sin embargo, hay presentaciones mucho más escuetas, Gödel redujo considerablemente el número de axiomas y en el libro de Church [4] los axiomas proposicionales se reducen a estos tres:

Axm 1 $A \to (B \to A)$

Axm 2 $(A \to B) \to ((B \to C) \to (A \to C))$

Axm 3 $((A \to \bot) \to \bot) \to A$

El primero se denomina *ley de afirmación del consecuente*, el segundo *ley distributiva del condicional material* y el tercero *ley de la doble negación.*

Hemos usado los signos $\bot$ y $\top$ para lo falso y lo verdadero. Cuando no se tengan como primitivos, se introducen por definición así:

$$\bot :=_{Df} (p \wedge \neg p) \text{ y } \top :=_{Df} \neg\bot$$

Con el cálculo axiomático se da un método para generar el conjunto VAL de las fórmulas válidas a partir del de los axiomas. Pero un cálculo deductivo, además de mecanizar el concepto de validez, debe ofrecer una réplica del de consecuencia; esto es, debemos indicar cómo obtener conclusiones a partir de un conjunto Γ de hipótesis. Una forma sencilla de hacerlo es incorporar a los axiomas de H las fórmulas de Γ de manera que se defina la deducción a partir de hipótesis basada en la deducción de teoremas lógicos :

$$\Gamma \vdash_H A \text{ syss } \vdash_{H\cup\Gamma} A$$

Sin embargo, definiéndolo así, sin más cuidado, pueden fallar los teoremas de corrección y monotonía para fórmulas con variables libres. Lo que se debe hacer es modificar levemente las reglas de manera que las restricciones sobre variables libres afecten también a las fórmulas de Γ.

Por ejemplo, no queremos que se pueda demostrar esto

$$\{Px\} \vdash_H \forall x Px$$

El primer metateorema que se demuestra[7] tanto para éste como para otros cálculos es el de la deducción, que dice lo siguiente:

Teorema 141 *(de la deducción) Si* $\Gamma \cup \{A\} \vdash_H B$ *entonces* $\Gamma \vdash_H A \to B$

Los cálculos axiomáticos son buenos para demostrar metateoremas sobre ellos —el teorema de corrección es casi inmediato— y como hemos visto, permite visualizar a los subcálculos tales como el intuicionista o el minimal de forma

[7] Con detalle puede verse en Marraud [12].

evidente. otra de sus ventajas es que se extienden fácilmente a lógicas no clásicas, como la modal o la dinámica[8], añadiendo axiomas y reglas sobre un lenguaje asímismo extendido, pero conservando el chasis.

Los cálculos axiomáticos son desastrosos para efectuar deducciones en ellos: La razón principal es que puesto que la única regla de inferencia es *Modus Ponens*, para demostrar B debemos encontrar fórmulas A y $A \to B$ de las que deducirla, y la búsqueda de A se efectuará en un conjunto potencialmente infinito, siendo así que no tiene que estar directamente relacionada con B. Por estas mismas razones son inadecuados para la deducción automática y pedagógicamente ingratos.

Muchos de los cálculos que se introducen en este capítulo permitirán una búsqueda sistemática de una prueba de B. Se trata de sistemas que tienen la *propiedad de la subfórmula*; esto es, cada fórmula que aparece en la prueba es muy similar a una subfórmula de B. *Modus Ponens*, que es la regla que hemos apuntado como origen de la dificultad para automatizar el proceso de búsqueda de pruebas, es un caso particular de la regla que en las cálculos de Gentzen se denominan *Reglas de Corte* y que adopta formas diversas.

En Andrews [1] es

$$\text{Si} \vdash B \vee A \quad \text{y} \quad \vdash \neg A \vee C \quad \text{entonces} \quad \vdash B \vee C$$

Otra regla de inferencia, primitiva o derivada, que puede originar circularidad es la de doble negación

$$\vdash A \quad \text{syss} \quad \vdash \neg\neg A$$

Pues bien, usando estas reglas y sin usarlas probamos los mismos teoremas. Este es el denominado *Teorema de eliminación del Corte*, el famoso *Hauptsatz* de Gentzen —Hauptsatz significa enunciado principal—

Comentario 142 *El que todavía se enseñen estos cálculos en el primer año de carrera y se les haga a los alumnos deducir teoremas en ellos me resulta incomprensible, habiendo, como veremos, otras posibilidades. Sin embargo, tanto por razones históricas como porque hay algunas lógicas que sólo cuentan con cálculos axiomáticos, no son completamente prescindibles.*

4.4. Deducción natural

Pese a sus ventajas en el plano metateórico, los axiomáticos no son de manejo intuitivamente fácil, ni reflejan el proceso que sigue alguien que quiera informalmente demostrar un enunciado matemático, donde se utilizan supuestos auxiliares que se puedan luego eliminar; por ejemplo, para demostrar $A \to B$ se supone A y se demuestra B y de esta forma, eliminando el supuesto A se concluye que $A \to B$.

En 1934 se publica *Untersuchungen über das Logische Schliessen* de Gentzen, cuya traducción al francés fue realizada y comentada por Feys y Ladrière. En

[8] Véase 8.5.1 y 9.6.

el prefacio la presentan como una alternativa al estilo Hilbert, al que tachan de artificioso —desde un punto de vista intuitivo— al reducir las deducciones a la transformación de un mínimo conjunto de axiomas. Por el contrario el enfoque de Gentzen es caracterizado por ellos como revelador del uso frecuente y natural de razonar en matemáticas. Ahora se permite introducir la expresión, *tal cosa es verdadera bajo tal suposición*, como elemento formalizado de sus dos sistemas deductivos, tanto en el de *Deducción Natural* como en el de *Secuentes* —implícitamente en los sistemas primeros y explícitamente en los segundos —. La siguiente afirmación de Gentzen fundamenta lo dicho por Feys:

> *Nous voulons édifier un formalisme qui reflète le plus exactement possible les raisonnements logiques qui son réellement utilisés dans les démostrations mathématiques.*

Actualmente se acepta que la presentación de Gentzen es la que mejor refleja los aspectos inferenciales de los sistemas lógicos y por lo tanto es común considerar al cálculo de deducción natural como el más apropiado para una versión inferencial de la lógica.

El primero de los sistemas de Gentzen; se caracteriza por lo siguiente[9]: (1) las reglas de inferencia son más bien reglas de derivación que reglas de demostración —en el sentido de que fueron pensadas para mostrar la validez de inferencias más que para establecer que determinadas fórmulas son teoremas; (2) no hay axiomas sino dos clases generales de supuestos, las premisas o supuestos iniciales y los supuestos adicionales; (3) hay reglas que obligan a la introducción de supuestos adicionales que luego es obligatorio descargar, mientras que no es obligatorio descargar los supuestos iniciales o premisas; (4) los teoremas lógicos se definen como deducciones a partir del conjunto vacío de los supuestos; (5) el significado de las constantes lógicas —al menos en los sistemas de deducción natural estrictos— se fija por medio de reglas de *Introducción* y de *Eliminación* del signo lógico en la conclusión; y (6) una prueba es una sucesión finita de fórmulas donde cada una es o un supuesto inicial, o un supuesto adicional o el resultado de la aplicación de una de las I-reglas o E-reglas a fórmulas anteriores.

En este cálculo *natural* la deducción parte de supuestos que se transforman mediante reglas de inferencia para conseguir nuevas fórmulas; no hay axiomas. Algunas reglas nos garantizan la independencia de una fórmula respecto a los supuestos usados para conseguirla, eliminándose adecuadamente. Son estas las reglas de *descarga de hipótesis.* Para que una fórmula sea un teorema, todas las hipótesis han tenido que ser convenientemente suprimidas.

El formato de las reglas es siempre el mismo, dividiéndose entre reglas de introducción y de eliminación de la conectiva, o del cuantificador.

Reglas

Estas son sus reglas proposicionales:

[9] Esto que sigue lo dice Sundholm en [19], y yo lo comparto.

1. Introducción y eliminación del condicional

$$\dfrac{\begin{array}{c}[A]\\ \vdots\\ B\end{array}}{A \to B}\ I \to \qquad \dfrac{A \qquad A \to B}{B}\ E \to$$

2. Introducción y eliminación de la conjunción

$$\dfrac{A \qquad B}{A \wedge B}\ I\wedge \qquad \dfrac{A \wedge B}{A}\ E1\wedge \qquad \dfrac{A \wedge B}{B}\ E2\wedge$$

3. Introducción de la disyunción

$$\dfrac{A}{A \vee B}\ I\vee \qquad \dfrac{B}{A \vee B}\ I\vee$$

4. Eliminación de la disyunción

$$\dfrac{A \vee B \qquad \begin{array}{c}[A]\\ \vdots\\ C\end{array} \qquad \begin{array}{c}[B]\\ \vdots\\ C\end{array}}{C}\ E\vee$$

5. Introducción y eliminación de la negación

$$\dfrac{\begin{array}{c}[A]\\ \vdots\\ \bot\end{array}}{\neg A}\ I\neg \qquad \dfrac{A \quad \neg A}{\bot}\ E\neg$$

6. Regla intuicionista de lo absurdo

$$\dfrac{\bot}{A}\ \bot \text{ intuicionista}$$

7. Doble negación

$$\dfrac{((A \to \bot) \to \bot)}{A}\ DN$$

Para el cálculo de primer orden sin igualdad incluimos además:

8. Reglas de generalización

$$\frac{A\left(\frac{x}{y}\right)}{\forall x A}\ I\forall \qquad\qquad \frac{\forall x A}{A\left(\frac{\tau}{x}\right)}\ E\forall$$

9. Reglas de particularización

$$\frac{A\left(\frac{\tau}{x}\right)}{\exists x A}\ I\exists \qquad\qquad \frac{\exists x A \qquad \begin{array}{c}\left[A\left(\frac{x}{y}\right)\right]\\ \vdots \\ C\end{array}}{C}\ E\exists$$

La lógica clásica incluye todas estas reglas, la intuicionista elimina la de la doble negación y la minimal ésta y la del absurdo.

En el campo de la deducción automática de teoremas se han implementado cálculos de deducción natural; por ejemplo, *AUTOMATH* y *NuPRL*.

Ejemplo 143 *En la lógica minimal demostramos lo siguiente:*

$$\dfrac{\dfrac{[A]}{\dfrac{B \to A}{A \to (B \to A)}\ I\to}}{}\ I\to$$

Ejemplo 144 *Éste es un teorema de la minimal de primer orden:*

$$\dfrac{\dfrac{\dfrac{\dfrac{[\neg\exists x A]^2 \qquad \dfrac{[A]^1}{\exists x A}\ I\exists}{\bot}\ E\neg}{\neg A}\ I^1\neg}{\forall x \neg A}\ I\forall}{\neg\exists x A \to \forall x \neg A}\ I^2\to$$

Ejemplo 145 *Éste es un teorema de la clásica de primer orden:*

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{[\neg\forall x A]^3 \qquad \dfrac{\dfrac{[\neg\exists x \neg A]^2 \qquad \dfrac{[\neg A]^1}{\exists x \neg A}\ I\exists}{\bot}\ E\neg}{A}\ \bot\ c^1}{\forall x A}\ I\forall}{\bot}\ E\neg}{\exists x \neg A}\ \bot\ c^2}{\neg\forall x A \to \exists x \neg A}\ I^3\to$$

Se demuestra con facilidad que los teoremas del cálculo minimal están contenidos en el intuicionista y éstos en el clásico. Por supuesto, no vale la inclusión contraria.

$$MINIMAL \subset INTUICIONISTA \subset CLÁSICA$$

Al demostrar teoremas en este cálculo se producen pruebas redundantes en las que se introducen fórmulas que luego son eliminadas; sucede especialmente al utilizar la regla del condicional, que es una versión de la de corte. Esto plantea la pregunta de si hay siempre una prueba alternativa, sin corte. La verdad es que sí y su demostración la hizo Gentzen. Para un cálculo similar al que aquí proponemos la demostración de la eliminación del corte, o teorema de normalización se puede consultar en Marraud [12].

El formato de presentación de los cálculos de deducción natural es muy variado, hay quienes utilizan cajas o corchetes anidados, o marcas e interrogantes para indicar en qué estado se encuentra la fórmula; esto es, si ha sido ya descargada. Este cálculo goza de mucho predicamento entre los que confían en el valor propedeútico de las demostraciones formales, como introducción a las informales.

En la sección 11.3.2 se encuentra una reflexión muy interesante —acompañada de reformulación— de la naturaleza de la *deducción natural*. Ésta se lleva a cabo en el cálculo lambda y se origina en 1924, allá por los inicios de la lógica combinatoria, cuando Schönfinkel[10] observó que las inferencias básicas de la lógica son de naturaleza funcional; es decir, que son funciones de fórmulas en fórmulas. Semejantes funciones entre fórmulas lo serán del cálculo lambda: las fórmulas del cálculo de deducción natural son los tipos básicos del cálculo lambda —esta analogía es conocida bajo este epígrafe: *"proposiciones como tipos"*—, las reglas del cálculo de deducción natural son o bien $\lambda-$términos u otras expresiones del $\lambda-$cálculo.

4.5. Cálculo de secuentes

Una aproximación esencialmente distinta a la noción de consecuencia lógica y de sistema deductivo es la que Gentzen [9] ofrece en su *Cálculo de Secuentes*. Según sus propias palabras, éste es un *cálculo logístico* al estilo Hilbert, ya que tiene como objetivo fundamental la demostración de teoremas lógicos —contiene axiomas expresados en términos del cálculo de secuentes— y servirá para determinar qué secuencias son teoremas. La gran diferencia es que sus teoremas lógicos son en el fondo expresiones del metalenguaje: las nuevas fórmulas no son fórmulas sino secuentes. En realidad se asemeja al enfoque antes mencionado de conversión de las inferencias en fórmulas. Se diferencia del de deducción natural porque: (1) introduce el nuevo concepto de secuente; (2) no parte de supuestos, sino de un *secuente básico o axioma* y un conjunto de *reglas de inferencia* que son fundamentalmente *reglas estructurales* y que hacen referencia precisamente a la estructura de los secuentes; (3) introduce un nuevo signo en su lenguaje $\dashv$ para denotar una determinada relación entre antecedente y consecuente del secuente; y por último, (4) precisamente el signo $\dashv$ permite reflejar en su propio lenguaje objeto la noción de consecuencia lógica.

[10] La elaboró Curry en el 1958 y fue redescubierta por Howard en 1980.

Definición 146 *Un* ***secuente*** *es una expresión de la forma:*

$$\Gamma \dashv \Omega$$

donde Γ, Ω *son conjuntos de fórmulas cualesquiera.*

Hay dos tipos de reglas de inferencia: las estructurales propiamente dichas y las de operaciones; tanto unas como otras tienen un esquema similar. Aceptando que la secuencia $\Gamma \dashv \Omega$ refleja en el lenguaje de secuentes la noción de inferencia, las reglas, tanto las estructurales como las de operaciones, toman inferencias como premisas y devuelven inferencias como conclusión. Gentzen las llama *figuras de deducción* —según el caso estructurales u operatorias—. Sin embargo, existen diferencias fundamentales entre las reglas de los cálculos de deducción natural y las reglas de los de secuentes, a saber[11]:

1. pese a que los esquemas operatorios se refieren a cada signo lógico, no hay reglas de eliminación; en su lugar aparecen reglas de *introducción en el prosecuente* y las reglas de introducción de NK —el de deducción natural— son ahora las de *introducción en el postsecuente*

2. las reglas estructurales, en tanto que permiten la permutación, repetición y agregado de fórmulas en el prosecuente y en el postsecuente, generalizan propiedades de la noción de deducción para cualquier tipo de deducción representada en una secuencia, prescindiendo totalmente de los signos lógicos que ocurren en las fórmulas que componen los secuentes; y

3. a diferencia de NK, las reglas de LK permiten más de una fórmula en el postsecuente. Además, análogamente a cómo las reglas operatorias fijan el significado de los signos lógicos, las reglas estructurales fijan el significado del signo $\dashv$ asignándole propiedades similares al signo metalingüístico de deducibilidad $\vdash$

Axiomas

$$\Gamma \dashv \Gamma \qquad \dashv \top \qquad \bot \dashv$$

Reglas estructurales (o figuras de deducción estructurales)

a la derecha —o en el *postsecuente* (o consecuente)—
a la izquierda —o en el *prosecuente* (o antecedente)—

1. Atenuación

$$\frac{\Gamma \dashv \Omega}{\Gamma \dashv \Omega, A} \dashv A \qquad\qquad \frac{\Gamma \dashv \Omega}{A, \Gamma \dashv \Omega} A \dashv$$

[11] Tomado de Palau [14]

2. Contracción

$$\frac{\Gamma \dashv \Omega, A, A}{\Gamma \dashv \Omega, A} \dashv C \qquad \frac{\Gamma, A, A \dashv \Omega}{\Gamma, A \dashv \Omega} C \dashv$$

3. Permutación

$$\frac{\Gamma \dashv \Theta, A, B, \Omega}{\Gamma \dashv \Theta, B, A, \Omega} \dashv P \qquad \frac{\Gamma, A, B, \Phi \dashv \Omega}{\Gamma, B, A, \Phi \dashv \Omega} P \dashv$$

4. Corte (Eliminación)

$$\frac{\Gamma \dashv \Theta, A \qquad A, \Omega \dashv \Phi}{\Gamma, \Omega \dashv \Theta, \Phi}$$

Reglas de operaciones (en el *postsecuente* $\dashv x$ en el *prosecuente* $x \dashv$)

1. Condicional

$$\frac{A, \Gamma \dashv \Omega, B}{\Gamma \dashv \Omega, A \to B} \dashv \to \qquad \frac{\Gamma \dashv \Omega, A \quad B, \Phi \dashv \Theta}{A \to B, \Gamma, \Phi \dashv \Omega, \Theta} \to \dashv$$

2. Conjunción

$$\frac{\Gamma \dashv \Omega, A \quad \Gamma \dashv \Omega, B}{\Gamma \dashv \Omega,\ A \wedge B} \dashv \wedge \qquad \frac{A, \Gamma \dashv \Omega}{A \wedge B, \Gamma \dashv \Omega} \quad \frac{B, \Gamma \dashv \Omega}{A \wedge B, \Gamma \dashv \Omega} \wedge \dashv$$

3. Disyunción

$$\frac{\Gamma \dashv \Omega, A}{\Gamma \dashv \Omega, A \vee B} \dashv \vee \qquad \frac{\Gamma \dashv \Omega, B}{\Gamma \dashv \Omega, A \vee B} \dashv \vee \qquad \frac{A, \Gamma \dashv \Omega \quad B, \Gamma \dashv \Omega}{A \vee B, \Gamma \dashv \Omega} \vee \dashv$$

4. Negación

$$\frac{A, \Gamma \dashv \Omega}{\Gamma \dashv \Omega, \neg A} \dashv \neg \qquad \frac{\Gamma \dashv \Omega, A}{\neg A, \Gamma \dashv \Omega} \neg \dashv$$

Para el cálculo de primer orden se añaden las siguientes, con las restricciones sobre variables habituales

5. Reglas de generalización

$$\frac{\Gamma, A\left(\frac{x}{y}\right) \dashv \Omega}{\Gamma, \forall x A \dashv \Omega} \forall \dashv \qquad \frac{\Gamma \dashv \Omega, A\left(\frac{\tau}{x}\right)}{\Gamma \dashv \Omega, \forall x A} \dashv \forall$$

6. Reglas de particularización

$$\frac{\Gamma \dashv \Omega, A\left(\frac{\tau}{x}\right)}{\Gamma \dashv \Omega, \exists x A} \dashv \exists \qquad \frac{\Gamma, A\left(\frac{x}{y}\right) \dashv \Omega}{\Gamma, \exists x A \dashv \Omega} \exists \dashv$$

La regla $\neg \dashv$ es el *Principio de no Contradicción*; la de $\dashv \neg$ representa al *Principio del Tercio Excluso*, mientras que $\rightarrow \dashv$ es el *Modus Ponens*.

En este cálculo, si exceptuamos la regla de corte, las demás presentan la siguiente característica, llamada *propiedad de subfórmula:*

> *"toda fórmula que aparece en la prueba de un teorema guarda cierta semejanza con la conclusión, al menos con una subfórmula"*

De esto surge que toda demostración $\Gamma \dashv A$ que no use corte, tiene la propiedad de la subfórmula; no se cumple para el caso de corte, en donde aparece en el proceso una fórmula que no tiene conexión con la que se quiere probar. En particular, corte elimina lo que silogísticamente Aristóteles llamó el *término medio*, el que desaparece en la conclusión. Ahora bien, si los axiomas y reglas estructurales —sin Corte— son suficientes para generar todas las afirmaciones que contienen $\dashv$ y además las reglas de operaciones bastan para determinar el significado de los conectores lógicos, entonces se podría esperar *a priori*, que semejante regla fuera redundante. Y esto es precisamente lo que afirma el llamado *Teorema fundamental (Hauptsatz)*, tanto para la intuicionista como para la clásica. En principio, podría pensarse que corte es eliminable como regla primitiva y que es posible derivarla de las restantes, no es así.

El que sea eliminable no quiere decir que no tenga utilidad, siendo aplicada por el propio Gentzen para importantes resultados metateóricos: (1) en la demostración de la corrección de la lógica de primer orden tanto clásica como intuicionista, (2) para demostrar la consistencia de la aritmética sin necesidad de acudir al principio de inducción completa, (3) para dar una solución al problema de la decisión de la lógica intuicionista, (4) para realizar la transformación de un sistema de deducción natural para la lógica intuicionista en uno de secuentes para la misma lógica, y por último, (5) para demostrar la equivalencia entre los sistemas estilo-Hilbert y de secuentes tanto para la lógica intuicionista como para la lógica clásica.

Pero es obvio que de su aplicabilidad no se infiere la necesidad de incluirla entre las reglas estructurales. Lo que en realidad sucede es que de la regla de corte se deriva la de *Modus Ponens* como caso particular, y es ésta la que nos permite caracterizar la deducibilidad.

Demostrar la corrección de este cálculo es muy sencillo; cada secuente expresa consecuencia —cuando $\Gamma \dashv A$ también $\Gamma \vDash A$— y las reglas transforman secuentes correctos en secuentes correctos. Las demostraciones en ellos tienen la misma dificultad que las del de deducción natural, yo no los enseñaría en primero, pero sí en otros cursos[12].

4.6. Tableaux semánticos

Los "Tableaux semánticos" son:

[12] Tanto en mi libro de *Teoría de Modelos* como en el de *Extensions of First Order Logic* los cálculos que he elegido son de esta clase.

(1) Procedimiento sintáctico de prueba de teoremas y (2) Procedimiento semántico de búsqueda de un modelo que cumpla ciertos requisitos. Aunque ambas caracterizaciones son acertadas, la segunda permite un tratamiento más intuitivo, sin olvidar que lo importante es que se trata de un cálculo.

Como tal existe en varios formatos y para lógicas diversas[13]: clásica, temporales, modales, intuicionista, subestructurales, no monotónica, etc. El procedimiento de prueba usado es *refutativo*; esto es, para demostrar que A es válida partimos de su negación y vamos desmembrando la fórmula sintácticamente, normalmente produciendo una serie de alternativas. Esta parte del proceso de construcción del tableau es de *expansión* y se caracteriza por entender las fórmulas como disyunción de conjunciones —que así lo son, de acuerdo con el *teorema de la forma normal*— para ir situándolos en las ramas de un árbol. Frecuentemente, aunque no siempre, las nuevas fórmulas son subfórmulas de las anteriores. Hay finalmente reglas de *cierre* que se establecen al producirse incompatibilidades sintácticas dentro de una misma rama. Si todas las ramas están cerradas, decimos que el tableau *está cerrado*. Un tableau cerrado que parte de la negación de A —o de una expresión que diga que A no es válida, $F\ A$— es una prueba mediante tableaux de A.

Hay otra forma semántica de entender los tableaux, como búsqueda de un modelo que cumpla ciertos requisitos; cada rama puede considerarse como una descripción parcial de un modelo, produciéndose al final del proceso éste o un contraejemplo, al menos en las lógicas decidibles. Este recurso es *implementado* en *demostración automática de teoremas*, estando tales sistemas normalmente inspirados en tableaux o en el sistema deductivo complementario, el de *resolución* que será tratado más adelante, en la sección 4.7.

Hay otra forma de entenderlos, que desde el punto de vista de la inteligencia artificial es impagable, y que no he visto documentado, y es como procedimiento de búsqueda de solución a un problema, pudiéndose establecer ciertos filtros. Esto lo vimos en la sección 1.6.3.

Estas son sus ventajas como cálculo deductivo: (1) Son *'automáticos'* para la lógica proposicional —esto es, proporcionan un procedimiento de decisión que en un número finito de pasos nos dice si la fórmula es válida o no lo es—, (2) Pueden ser fácilmente implementados en el ordenador —aunque, a menudo, la eficiencia es pobre en comparación con otros sistemas de prueba—, (3) Son fácilmente generalizables a la lógica de primer orden y a otras lógicas —modal, temporal, etc—, (4) Su aprendizaje es extremadamente sencillo.

¿Quién da más?

4.6.1. Tableaux proposicionales

Las definiciones son las ya introducidas en la sección 1.6.1: Vimos que hay reglas para cada conectiva y su negación y una regla especial para cerrar una rama contradictoria. Como puede verse en el cuadro, las fórmulas α son las de naturaleza conjuntiva.

[13] Véase el manual, elaborado por varios autores, en el que se desarrollan cálculos de esta índole para lógicas varias [5].

• α-reglas ($\alpha = \alpha_1 \wedge \alpha_2$):

α	α_1	α_2
$A \wedge B$	A	B
$\neg(A \vee B)$	$\neg A$	$\neg B$
$\neg(A \to B)$	A	$\neg B$
$A \leftrightarrow B$	$A \to B$	$B \to A$

Mientras que las β son disyuntivas.

• β-reglas ($\beta = \beta_1 \vee \beta_2$):

β	β_1	β_2
$A \vee B$	A	B
$\neg(A \wedge B)$	$\neg A$	$\neg B$
$A \to B$	$\neg A$	B
$\neg(A \leftrightarrow B)$	$\neg(A \to B)$	$\neg(B \to A)$

Regla de cierre:
Cerrar una rama que tenga A y $\neg A$ —para cualquier A—, o $\neg\top$, o $\bot$.

El teorema de corrección-completud

Resulta que con este procedimiento podemos probar exactamente las fórmulas válidas:

Teorema 147 *Sea A una fórmula proposicional. Entonces*

$$\models A \quad \textit{syss} \quad \vdash A$$

—A es válida syss A es un teorema lógico—

¿Por qué el teorema 147 es verdadero?

Corrección: Ya hemos dicho que las ramas de un tableau para A exploran todas las maneras en que A puede ser verdad en un modelo. Por lo tanto, si $\vdash A$, entonces hay un tableau cerrado para $\neg A$. Por consiguiente, todas las posibilidades han sido indagadas y todas se han cerrado; ninguna nos ha permitido encontrar un modelo de $\neg A$ ya que todas ellas son contradictorias; es decir, $\neg A$ *no puede* ser verdad nunca. Por lo tanto A debe ser siempre verdadera; es decir, es una fórmula válida.

Completud: Si $\nvdash A$, hagamos un tableau *'completo'* para $\neg A$ aplicando *todas las reglas posibles.* Como $\nvdash A$, el tableau de $\neg A$ debe tener al menos una rama abierta. Esta rama es una descripción completa de un modo en el que $\neg A$ puede ser verdadero. Podemos usarla para construir un modelo de $\neg A$. Por lo tanto A no es válida.

Comentario 148 *Por supuesto, lo anterior no constituye una demostración de los teoremas de completud y corrección.*

Decidibilidad algorítmica

Debido a que hay sólo una regla aplicable a cada línea dada de un tableau —la exigida por la forma lógica de la fórmula que está en la línea—, y a que siempre las fórmulas *"output"* de cualquier regla son más simples que las fórmulas *"input"*, se puede programar un ordenador para construir un tableau para cada fórmula dada $\neg A$. El programa terminará en un tiempo finito, o bien porque el tableau se cierra, o porque se ha completado; esto es, no se pueden aplicar más reglas.

- Si el tableau se cierra, sabemos que $\vdash A$
- Si no, podemos extraer un modelo de $\neg A$ a partir de una rama abierta, por lo tanto $\nvdash A$

Consecuentemente, se puede programar un ordenador para decidir en un tiempo finito si se da $\vdash A$ o no, para cualquier fórmula proposicional A.

Veremos que éste no es el caso en la lógica de primer orden.

4.6.2. Tableaux de primer orden

Extenderemos este trabajo a la lógica de primer orden, para lo que sólo se precisa añadir algunas reglas para cuantificadores y para la identidad. De todos modos, la construcción de un tableau, caso de haberlo, ya no será *"automática"*.

Sea L un lenguaje con un número infinito de constantes individulae, y sea A una fórmula. Hacemos un *tableau para* A empezando con A y aplicando *las reglas de los tableaux.* Se conservan las reglas de la lógica proposicional. Añadimos las siguientes:

• γ-reglas:

γ	$\gamma(t)$
$\forall x A$	$A\left(\frac{t}{x}\right)$
$\neg\exists x A$	$\neg A\left(\frac{t}{x}\right)$

—Si t es un término cerrado y x una variable—

• δ-reglas:

δ	$\delta(c)$
$\exists x A$	$A\left(\frac{c}{x}\right)$
$\neg\forall x A$	$\neg A\left(\frac{c}{x}\right)$

—Siendo x es una variable, para cualquier constante $c \in L$ que no haya sido usada aún en la rama—

Las fórmulas γ actúan de manera universal y las δ existencial.

Ejemplo 149 *Este es un tableau para* $\forall x \exists y Pxy$.

1. $\forall x \exists y Pxy$
2. $\exists y Pcy$ $\quad \gamma 1$
3. Pcd $\quad \delta 2$

Ejemplo 150 *Otro tableau para* $\forall x \exists y Pxy$ *Podríamos haber seguido aplicando las reglas:*

1.	$\forall x \exists y Pxy$	
2.	$\exists y Pcy$	$\gamma 1$
3.	Pcd	$\delta 2$
4.	$\exists y Pdy$	$\gamma 1$
5.	$\exists y Pf(c,a,d),y$	$\gamma 1$
6.	Pde	$\delta 4$
7.	$Pf(c,a,d),b$	$\delta 5$

es también un tableau para $\forall x \exists y Pxy$*, y así sucesivamente.*

Definición 151 *Sea* A *una sentencia. Escribimos* $\vdash A$ *si existe un tableau cerrado para* $\neg A$*.*

Ejemplo 152 *Probamos* $\vdash \exists x Px \rightarrow \neg \forall x \neg Px$*.*

1. $\neg(\exists x Px \rightarrow \neg \forall x \neg Px)$

2.	$\exists x Px$	$\alpha 1$
3.	$\neg\neg \forall x \neg Px$	$\alpha 1$
4.	Pc	$\delta 2$
5.	$\forall x \neg Px$	$\alpha 3$
6.	$\neg Pc$	$\gamma 5$

cerrado(4,6)

¿Cómo sabemos cuándo tenemos que parar?

Como en la lógica proposicional intentamos cerrar el tableau. Paramos cuando todas las ramas están cerradas; esto es, contienen una contradicción explícita. Desde luego, puede ocurrir que no seamos lo suficientemente listos como para conseguir cerrarlo.

Las reglas de los tableux no son deterministas, nos dicen qué podemos hacer pero el orden de aplicación de las mismas hace la diferencia entre un tableau eficiente y otro ineficaz. Sin embargo, a diferencia de lo que sucedía en proposicional que el tableau acaba por *"agotamiento"*, en primer orden pueden aparecer ramas infinitas, de manera que nunca se cierre la rama incluso cuando exista un tableau cerrado. La fuente de conflicto es la regla γ. No cabría esperar algo distinto: la lógica de primer orden no es decidible, no podríamos esperar que fueran recursivamente enumerables VAL y su complementario.

4.7. Resolución

Introducción

¿Qué es Resolución?

Procedimiento de prueba alternativo al de tableaux semánticos, extraordinariamente similar al mismo.

Las demostraciones mediante tableaux semánticos se visualizan en árboles: cada rama representa una conjunción iterada y el árbol mismo la disyunción de

sus ramas. Por consiguiente, los árboles explicitan una disyunción generalizada de conjunciones generalizadas. El método de resolución se ocupa de su noción dual: conjunciones generalizadas de disyunciones generalizadas. La representación en este caso no se hará mediante árboles; la disyunción se hará listando los términos en corchetes —esto es, $[A_1, ..., A_n]$ representa $(...(A_1 \vee A_2) \vee ...) \vee A_n)$. Una conjunción de disyunciones se representa mediante una lista que contiene una disyunción por línea

$$\begin{array}{c} [A_{11}, ..., A_{1n}] \\ [A_{21}, ..., A_{2n}] \\ \vdots \\ [A_{p1}, ..., A_{pn}] \end{array}$$

Tenemos reglas para añadir nuevas líneas a una secuencia, denominadas *reglas de expansión*, y un *principio de resolución*. Las reglas de expansión son las mismas que usábamos en los tableaux, que para la lógica proposicional podemos ver así:

$$\frac{\alpha}{\begin{array}{c}\alpha_1\\ \alpha_2\end{array}} \qquad \begin{array}{c}\beta\\ \beta_1 \mid \beta_2\end{array} \qquad \frac{\sigma}{\sigma_1}$$

Estas reglas especifican tanto cómo se añaden nuevas líneas como la forma de derivar nuevas disyunciones de las existentes. Imaginad que tenemos una disyunción

$$D := [D_1, ...D_n]$$

que contiene una fórmula no literal A —recordad que literal aquí significa atómica o disyunción de atómica.

- Si $A := \neg\neg C$; entonces se sigue una nueva disyunción que es exacta a D excepto en lo que respecta a la ocurrencia de A; que ahora es reemplazada por C
- Si A es de tipo β, entonces se sigue una disyunción que coincide con D en todo excepto en que en vez de β tiene dos fórmulas; β_1 y β_2
- Si A es de tipo α, entonces se siguen dos disyunciones: una es como D en todo excepto en que en vez de α tiene α_1 y en la otra se sustituye α por α_2.

En cada caso diremos que la nueva disyunción (o disyunciones) se sigue de D mediante la aplicación de una regla de expansión.

Antes de precisar estos conceptos veámoslo con un ejemplo muy simple:

$$\begin{array}{l} [\neg(p \to (q \to p))] \\ [p] \\ [\neg(q \to p)] \\ [q] \\ [\neg p] \\ [] \end{array}$$

4.7.1. Resolución para la lógica proposicional

Sea A una fórmula proposicional. Hacemos una *expansión mediante resolución de A* empezando con A y aplicando las reglas de expansión y el principio de resolución. La descomposición se termina cuando o bien se obtiene la cláusula vacía, [], o no se pueden aplicar más reglas.

Si las reglas nos llevan a la cláusula vacía, entonces A es contradictoria y concluimos que $\neg A$ es válida. De lo contrario, podemos extraer un modelo de A siguiendo los valores de las fórmulas que aparecen en el desarrollo.

Las reglas de expansión

Hay reglas para cada conectiva y su negación, y una regla especial para cerrar una rama contradictoria. De hecho, son las mismas que en el cálculo de tableaux.

- α-reglas ($\alpha = \alpha_1 \wedge \alpha_2$):

α	α_1	α_2
$A \wedge B$	A	B
$\neg(A \vee B)$	$\neg A$	$\neg B$
$\neg(A \rightarrow B)$	A	$\neg B$
$A \leftrightarrow B$	$A \rightarrow B$	$B \rightarrow A$

- β-reglas ($\beta = \beta_1 \vee \beta_2$):

β	β_1	β_2
$A \vee B$	A	B
$\neg(A \wedge B)$	$\neg A$	$\neg B$
$A \rightarrow B$	$\neg A$	B
$\neg(A \leftrightarrow B)$	$\neg(A \rightarrow B)$	$\neg(B \rightarrow A)$

- $\sigma-$regla ($\sigma = \neg\neg\sigma$):

σ	σ_1
$\neg\neg A$	A

Principio de Resolución

Definición 153 *Decimos que D es el resolvente de D_1 y D_2 sobre A si: D_1 y D_2 son dos disyunciones tales que A es un miembro de D_1 y $\neg A$ lo es de D_2 y D se obtiene así:*
1.- De D_1 se borra A
2.- De D_2 se borra $\neg A$
3.- Se combina el resto en una nueva disyunción
Es decir,

si $D_1 = [C_1, ..., C_{i-1}, A, C_{i+1}, ..., C_n]$

y $D_2 = [E_1, ..., E_{j-1}, \neg A, E_{j+1}, ..., E_m]$

entonces $D = [(D_1 - \{A\}) \cup (D_2 - \{\neg A\})]$

• *Cuando* $D_1 = [A]$ *y* $D_2 = [\neg A]$ *se obtiene la cláusula vacía,* []. *(Se entiende como contradicción)*

Definición 154 *Sea* $\{C_1, ..., C_n\}$ *un conjunto finito de fórmulas proposicionales, entonces:*
$[C_1]$
$[C_2]$
$\vdots$
$[C_n]$
es una ***expansión mediante resolución*** *de* $\{C_1, ..., C_n\}$

Definición 155 *Si* S *es una expansión mediante resolución de* $\{C_1, ..., C_n\}$ *y* D *se obtiene de alguna línea o líneas de* S *mediante una regla de expansión o mediante el principio de resolución, entonces el resultado de añadir* D *a* S *como línea es también una expansión mediante resolución de* $\{C_1, ..., C_n\}$

Definición 156 *Una expansión mediante resolución que contiene la cláusula vacía se llama* ***cerrada***

Definición 157 *Una* ***demostración mediante resolución*** *de* C *es una expansión mediante resolución de* $\neg C$ *que está cerrada.*
C *es un* teorema *del sistema de resolución —escribimos* $\vdash_{res} C$*— si hay una demostración mediante resolución de* C

Definición 158 *Una expansión mediante resolución* S *es satisfacible, si hay una interpretación* $\Im$ *tal que cada línea de* S *es verdadera.*

Ejemplo 159 *Para demostrar que la fórmula* $\neg((\neg p \wedge \neg\neg q)$ *es un teorema hacemos el siguiente desarrollo de* $((\neg p \wedge \neg\neg q) \wedge \neg q)$

$[((\neg p \wedge \neg\neg q) \wedge \neg q)]$
$[(\neg p \wedge \neg\neg q)]$
$[\neg q]$
$[\neg p]$
$[\neg\neg q)]$
[]

Ejemplo 160 *Para demostrar que la fórmula* $((p \to q) \to p) \to p$ *es un teorema hacemos el siguiente desarrollo de* $\neg(((p \to q) \to p) \to p)$

$[\neg(((p \to q) \to p) \to p)]$
$[((p \to q) \to p)]$
$[\neg p]$
$[\neg(p \to q),\ p]$
$[\neg(p \to q)]$
$[p]$
$[\neg q]$
$[]$

Ejemplo 161 *Para demostrar que la fórmula* $(r \leftrightarrow s) \to ((r \wedge s) \vee (\neg r \wedge \neg s))$ *es una tautología partimos de su negación* $\neg((r \leftrightarrow s) \to ((r \wedge s) \vee (\neg r \wedge \neg s)))$

$[\neg((r \leftrightarrow s) \to ((r \wedge s) \vee (\neg r \wedge \neg s)))]$
$[r \leftrightarrow s]$
$[\neg\,((r \wedge s) \vee (\neg r \wedge \neg s))]$
$[\neg(r \wedge s)]$
$[\neg(\neg r \wedge \neg s)]$
$[\neg r,\ \ \neg s]$
$[\neg\neg r,\ \ \neg\neg s]$
$[r \to s]$
$[s \to r]$
$[\neg s,\ \ r]$
$[\neg s]$
$[\neg r,\ \ s]$
$[\neg r]$
$[\neg\neg s]$
$[]$

Para la resolución de primer orden introducimos las reglas γ y δ

γ	$\gamma\,(t)$
$\forall x A$	$A\left(\frac{t}{x}\right)$
$\neg\exists x A$	$\neg A\left(\frac{t}{x}\right)$

δ	$\delta\,(t)$
$\exists x A$	$A\left(\frac{t}{x}\right)$
$\neg\forall x A$	$\neg A\left(\frac{t}{x}\right)$

Corrección y completud

Queremos usar resolución, como antes usábamos tableaux, para probar como teoremas lógicos todas las fórmulas válidas (todas ellas, pero sólo ellas) y para determinar otras propiedades semánticas como satisfacibilidad, consecuencia e independencia.

Puesto que la cláusula vacía, [], no es satisfacible, una expansión mediante resolución cerrada, de una fórmula A muestra que $\neg A$ debe ser válida.

En el ejemplo 159 se demostró que

$$\vdash_{res} \neg((\neg p \wedge \neg\neg q) \wedge \neg q).$$

Mostramos en el ejemplo 160 que

$$\vdash_{res} ((p \to q) \to p) \to p.$$

Resulta que con este procedimiento podemos probar $\vdash_{res}$ exactamente las fórmulas válidas:

Teorema 162 *Sea A una fórmula proposicional. Entonces*

$$\models A \quad \textit{syss} \quad \vdash_{res} A$$

—A es válida syss A es un teorema lógico del sistema de resolución—

Modelos

De manera paralela a como lo hacíamos con tableaux, podemos extraer un modelo a partir de una expansión mediante resolución: veámoslo con un ejemplo

$$A = (p \vee \neg q) \wedge q$$

$[(p \vee \neg q) \wedge q]$
$[p \vee \neg q]$
$[q]$
$[p,\ \neg q]$
$[p]$

Esta expansión está *'completa'*; es decir, no se pueden aplicar más reglas. Podemos extraer un modelo $\Im$ de A a partir de los literales que aparecen en las distintas líneas de la expansión; en este caso tenemos p y q, por lo tanto hacemos que $\Im(p) = \Im(q) = V$. Como se puede comprobar fácilmente $\Im \models A$.

Demostraciones a partir de hipótesis

Por un procedimiento similar al usado con los tableaux, se pueden considerar conjuntos de hipótesis:

Definición 163 *Sean A, B fórmulas. Escribimos $A \vdash_{res} B$ si hay una expansión mediante resolución cerrada de $\{A,\ \neg B\}$.*

Ejemplo 164 *Demostramos que $p \wedge (p \to q) \vdash_{res} q$*

$[p \wedge (p \to q)]$
$[\neg q]$
$[p]$
$[p \to q]$
$[\neg p,\ q]$
$[q]$
$[]$

4.8. Forma normal conjuntiva

Hay cálculos de resolución que actúan sobre fórmulas escritas previamente en la denominada *forma normal conjuntiva*; esto es como una conjunción de disyunciones. A continuación introduzco un algoritmo que puede ser usado para efectuar la reducción.

Definición 165 *Un* ***literal*** *es una fórmula atómica o una fórmula atómica negada.*

Definición 166 *Una fórmula está en* ***forma normal conjuntiva*** *syss es de la forma*

$$A_1 \wedge A_2 \wedge ... \wedge A_n$$

con $n \geq 1$ *donde cada* A_i *—para cada* $i \in \{1, ..., n\}$*— es una disyunción de literales.*

Algoritmo 167 *Cualquier fórmula de* L_0 *se puede expresar en forma normal conjuntiva realizando, sistemáticamente las transformaciones siguientes:*

1. Fase primera: Eliminar los conectores $\rightarrow$ y $\leftrightarrow$ utilizando las equivalencias siguientes:

   a) $(A \leftrightarrow B) \equiv (A \rightarrow B) \wedge (B \rightarrow A)$
   b) $(A \rightarrow B) \equiv \neg A \vee B$

2. Segunda fase: Reducir al máximo el alcance del conector monario de negación, utilizando las equivalencias siguientes tantas veces como sea necesario:

   a) $\neg\neg A \equiv A$
   b) $\neg(A \wedge B) \equiv (\neg A \vee \neg B)$
   c) $\neg(A \vee B) \equiv (\neg A \wedge \neg B)$

3. Tercera fase: Obtener la forma normal conjuntiva usando las equivalencias siguientes, tantas veces como sea necesario:

   a) $(A \vee (B \wedge C)) \equiv ((A \vee B) \wedge (A \vee C))$
   b) $(A \wedge B) \equiv (B \wedge A)$
   c) $(A \vee B) \equiv (B \vee A)$
   d) $((A \wedge B) \vee C) \equiv ((A \vee C) \wedge (B \vee C))$
   e) $((A \vee B) \wedge C) \equiv ((A \wedge C) \vee (B \wedge C))$

4.9. Selección de Meta-Teoremas

En esta sección se concentran los resultados más interesantes de *Teoría de la Demostración*, desde el teorema de Herbrand hasta el de definibilidad de Beth. El *teorema de la existencia de modelo* juega un papel central y permite obtener otros muchos resultados, el inconveniente es que para demostrarlo se utilizan otros teoremas cuya prueba no es constructiva.

Teorema 168 *Sea A una fórmula atómica, B y C fórmulas cualesquiera tales que en C haya ocurrencias de A. Escribimos $C\left(\frac{B}{A}\right)$ para denotar la fórmula que resulta de borrar en C las ocurrencias de A y poner en su lugar B. Para abreviar escribimos también $C(A)$ y $C(B)$ —respectivamente, para la original C y para $C\left(\frac{B}{A}\right)$—*

Teorema 169 *(de reemplazo de fórmulas equivalentes). Sea $\mathcal{A}$ una estructura de primer orden. Si $C(D)$ es una fórmula cualquiera — donde D, que es atómica ocurre— y A, B son fórmulas*

$$\mathcal{A} \Vdash A \leftrightarrow B \quad \textit{entonces} \quad \mathcal{A} \Vdash C(A) \leftrightarrow C(B)$$

Corolario 170 *Si $\vDash A \leftrightarrow B$ entonces $\vDash C(A) \leftrightarrow C(B)$*

Cuando demostramos fórmulas en un cálculo, por ejemplo en el de tableaux, se introducen constantes nuevas o parámetros al usar la regla δ, pero se pueden preprocesar las sentencias de manera que δ sea innecesaria; esto se consigue mediante la *skolemización*.

Teorema 171 *(Skolemización). Sea $C(x)$ una fórmula tal que sus variables libres están entre $x, y_1, ..., y_n$ y sea $D(A)$ una fórmula tal que $D(\exists x C(x))$ es una sentencia y f es un functor $n-$ario que no está en $D(\exists x C(x))$.*

1. *Si todas las estancias de A en $D(A)$ son positivas,entonces*

$$D\left(\frac{\exists x C(x)}{A}\right) \quad \textit{es satisfacible syss lo es} \quad D\left(\frac{C\left(\frac{fy_1 \ldots y_n}{x}\right)}{A}\right)$$

2. *Si todas las estancias de A en $D(A)$ son negativas, entonces*

$$D\left(\frac{\forall x C(x)}{A}\right) \quad \textit{es satisfacible syss lo es} \quad D\left(\frac{C\left(\frac{fy_1 \ldots y_n}{x}\right)}{A}\right)$$

Dada una fórmula es posible encontrar otra que es equivalente a la anterior, pero que tiene todos los cuantificadores al principio a las que llamamos *fórmulas en formas prenexa*. Si les aplicamos skolemización su estructura será muy simple.

Proposición 172 *Hay un algoritmo que convierte cada sentencia A en otra A^* en forma prenexa y que contiene sólo cuantificadores universales, tal que A es satisfacible syss A^* lo es.*

Teorema de Herbrand

En 1930 Herbrand demostró un teorema, que puede entenderse como una versión constructiva del de completud de Gödel, y que constituye la base teórica de los demostradores automáticos de teoremas. Para demostrarlo se introducen las *extensiones de Herbrand*; esto es, para una fórmula cualquiera B se define recursivamente $\varepsilon(B, D)$ así:

1. Si L es un literal $\varepsilon(L, D) = L$
2. $\varepsilon(\neg\neg A, D) = \varepsilon(A, D)$
3. $\varepsilon(\alpha, D) = \varepsilon(\alpha_1, D) \wedge \varepsilon(\alpha_2, D)$
4. $\varepsilon(\beta, D) = \varepsilon(\beta_1, D) \wedge \varepsilon(\beta_2, D)$
5. $\varepsilon(\gamma, D) = \varepsilon(\gamma(t_1), D) \wedge ... \wedge \varepsilon(\gamma(t_n), D)$
6. $\varepsilon(\delta, D) = \varepsilon(\delta(t_1), D) \vee ... \vee \varepsilon(\delta(t_n), D)$

—donde $D = \{t_1, ..., t_n\}$ es un conjunto de términos cerrados—

Por otra parte se definen modelos a partir de constantes, como hicimos en la prueba de completud de Henkin 2.4. Lo que hay que remarcar aquí es que finalmente se puede demostrar que la validez de primer orden se reduce a proposicional.

Teorema 173 *(Herbrand) Una sentencia A es válida syss una extensión de Herbrand de A es una tautología.*

La parte difícil de la demostración es la de ver que si A es válida entonces tiene una extensión de Herbrand que es una tautología, lo que él hizo fue convertir la validez en deducibilidad en una cálculo y establecer cómo a partir de la prueba de A se puede extraer una extensión de Herbrand que sea tautología; es decir, hay una versión constructiva de su teorema[14].

Teorema 174 *Hay una algoritmo que extrae de cada demostración mediante tableaux de una sentencia A una expansión de Herbrand tautológica.*

El teorema de *Corte* de Herbrand se prueba de manera fácil y constructiva, usando tableaux, *¿Puede hacerse algo parecido con otros cálculos deductivos?* Está claro que usando la corrección y completud de los cálculos podemos tomar una fórmula válida, por completud de tableaux sabemos que es demostrable en el cálculo de tableaux y con independencia de si la prueba original que establecía los validez de A fuera en un cálculo de esta clase o axiomático o de secuentes, hacer la conversión en tautología usando el tableau. De tal forma que se usa el cálculo de Hilbert o el de secuentes sólo para saber que hay una prueba, pero no utilizamos la prueba misma. Pese a todo, nos gustaría tener un procedimiento que tradujera las pruebas hechas en el axiomático a tableaux.

[14] La demostración para el cálculo de tableaux puede consultarse en [6]

Como era de esperar no hay problema con los axiomas de Hilbert, que tienen pruebas de tableaux sencillos, pero el escollo se sitúa en la regla de *Modus Ponens*

$$\frac{A \quad A \to B}{B}$$

Sin embargo, partiendo de la formulación simplificada de corte ya dada, a saber:

$$\frac{\Gamma \dashv A \qquad A, \Omega \dashv B}{\Gamma, \Omega \dashv B}$$

no es difícil ver que el siguiente esquema:

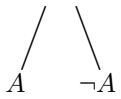
$$\frac{\Gamma \dashv A \qquad A, \Gamma \dashv B}{\Gamma \dashv B}$$

es un caso particular. Si además la secuencia Γ es vacía, se obtiene la de *Modus Ponens:*

$$\frac{\dashv A \qquad A \dashv B}{\dashv B}$$

Luego, la noción de consecuencia caracterizada por la lógica de secuentes, a través de la regla de corte, valida *Modus Ponens* como regla de la lógica clásica.

La versión simple de corte que para tableaux introduce Smullyan es



pudiéndose demostrar lo siguiente:

Proposición 175 *Si se permite la regla de Corte, una prueba mediante tableaux de A y una prueba mediante tableaux de $A \to B$ se convierte en una prueba mediante tableaux de B.*

La cuestión ahora está en ver si se podría prescindir de esta regla sin renunciar a la traducibilidad.

El algoritmo ideado por Gentzen para eliminar la aplicación de corte, junto con la demostración de que tal algoritmo funciona es uno de los pilares de la *Teoría de la Demostración*. El hecho de que se puede eliminar de forma constructiva el uso de la regla de corte tiene también consecuencias en la teoría de la prueba automática de teoremas.

Teorema 176 *(del Corte de **corte**) Todo tableau cerrado en el que se ha usado **Corte** puede convertirse en un tableau cerrado en donde no se ha aplicado **Corte**.*

Teorema 177 *(de interpolación de Craig). Si $A \to B$ es una sentencia válida, entonces tiene un interpolante; esto es, una sentencia C cuyos signos peculiares son comunes a A y B tal que $A \to C$ y $C \to B$ son ambas tautologías.*

De nuevo este teorema tienen una demostración no constructiva —usando el *teorema de la existencia de un modelo*— y una constructiva; tiene una versión más potente, demostrada por Lyndon.

Teorema 178 *(Lyndon). Si $A \to B$ es una sentencia válida, entonces tiene un interpolante C tal que cada relator que ocurre positivamente en C también lo hace en A y B —y lo mismo cuando ocurre negativamente—.*

Para terminar, unas palabras sobre el teorema de definibilidad de Beth.

En ocasiones la información que tenemos es explícita:

"María Manzano disfrutó de una Fulbright en Berkeley en el curso 1977-78"

y en otras implícita:

"La maravillosa profesora de lógica, que pasea una pastora belga llamada Maui, disfrutó de una Fulbright en Berkeley en el curso 1977-78"

Con frecuencia los acertijos se resuelven convirtiendo una caracterización implícita en explícita: Lo que dice el teorema de definibilidad de Beth es que en la lógica clásica de primer orden tales cuestiones siempre tienen solución.

Definición 179 *Sean R un relator $n-$ario y $A(x_1...x_n)$ una fórmula cuyas variables libres están en $\{x_1, ..., x_n\}$ en donde R no ocurre. Decimos que A define explícitamente a R respecto a un conjunto Γ de sentencias si*

$$\Gamma \vDash \forall x_1...x_n \, (A \longleftrightarrow Rx_1...x_n)$$

Definición 180 *Sea R un relator $n-$ario. Decimos que R es* definido implícitamente *en Γ syss lo determina unívocamente en este sentido. Sea R^* es un relator n-ario que no es R ni está en Γ y sea Γ^* el resultado de sustituir R por R^* en Γ. Decimos que Γ determina unívocamente a R si*

$$\Gamma \cup \Gamma^* \vDash \forall x_{1-} x_n \, (Rx_1...x_n \leftrightarrow R^*x_1...x_n)$$

Teorema 181 *(de Beth). Si R es definido implícitamente en Γ entonces R tiene una definición explícita respecto de Γ.*

En cierto modo lo que el teorema de Beth establece es la completud de la lógica de primer orden por lo que respecta a la definibilidad de sus relatores.

Bibliografía

[1] Andrews, P. [1986]. *An Introduction to Mathematical Logic and Type Theory: To Truth Through Proof.* Academic Press. New York. USA

[2] Bibel, W. y Eder, E. [1993]. *Methods and Calculi for Deduction.* en [8].

[3] Craig, W. [1957]. *"Linear reasonning. A new form of the Herbrand-Gentzen theorem"*

[4] Church, a. [1956]. *Introduction to Mathematical Logic.* vol I. Princeton: Princeton University Press.

[5] D'Agostino, M., Gabbay, D., Hähnle, R. y Possega, J. [1999]. *Handbook of Tableau Methods.* Kluwer Academic Publishers. Londres. U.K.

[6] Fitting, M. [1996]. *First-order logic and automated theorem proving,* Springer Graduate Texts in Computer Science. Berlín. Alemania.

[7] Gabbay, D y Guenthner, F. [2001]. *Handbook of Philosophical Logic* 2^{nd} edition. Kluwer Academic Publishers. Dordrecht. Holanda.

[8] Gabbay, D. Hogger, G. y Robinson, J. [1993-1998]. *Handbook of Logic in Artificial Intelligence and Logic Programming.* vol 1 a 6. OUP.

[9] Gentzen, G. [1934]. *"Untersuchungen über das Logische Schliessen"*. Math. Zeitschrift 39.

[10] Heijenoort, J. ed. [1967]. *From Frege to Gödel*. Harvard: Harvard University Press.

[11] Hodges, W. [1997]. *Logic* Pelican (Penguin). Londres. U.K.

[12] Marraud, H. [1998]. *Introducción a la Teoría de los Sistemas Deductivos.* Ediciones de la Universidad Autónoma de Madrid. Madrid.

[13] Marraud, H. y Navarro, P. [1988]. *Sistemas Deductivos Tipo Gentzen.* Ediciones de la Universidad Autónoma de Madrid. Madrid.

[14] Palau, G. [2000]. *La noción abstracta de consecuencia lógica.* en http://logicae.usal.es

[15] Post, E. L. [1921]. *"Introduction to a general theory of elementary propositions". (*en [10]).

[16] Post, E. L. [1927]. *"Finite combinatory processes".* **The Journal of Symbolic Logic**, vol. 2, pp. 103-105.

[17] Robinson, J. [1965]. *"A Machine-Oriented Logic Based on the Resolution Principle"*, J. Assoc. Comput. Mach. 12, pp.23-41.

[18] Smullyan, R. M. [1994]. *First-order logic,* Springer-Verlag, Berlin. (la versión original es de 1968, la del 94 es una edición revisada)

[19] Sundholm, G. [2000]. *"Systems of deduction"* en [7].

[20] Takeuti, G. [1987]. *Proof Theory.* North-Holland. Amsterdam.

Capítulo 5

Teoría de Conjuntos

En las secciones que siguen se presenta primero la teoría intuitiva de conjuntos, basada en la original de Cantor, para seguir con sus problemas de inconsistencia y terminar con la solución axiomática de Zermelo-Fraenkel.

5.1. Teoría intuitiva de conjuntos

5.1.1. La selva de Cantor

La definición inicial de Cantor es informal

> *"Un conjunto es cualquier colección C de objetos determinados y bien distintos x de nuestra percepción o nuestro pensamiento (que se denominan elementos de C), reunidos en un todo".*

Su concepción de la naturaleza de los conjuntos coincide con la de Frege, identificándola con la extensión de un predicado —esto es, con la colección de objetos que satisface el predicado—. Esta idea tan sencilla e intuitiva resulta ser también ingenua porque produce enormes contradicciones de inmediato, como por ejemplo la denominada *paradoja de Russell*. Para poder mostrarla es necesario empezar por formalizar esta teoría que, aparte de los signos para los conjuntos y sus elementos —C, x, etc.—, tendrá los signos de pertenencia $\in$ e igualdad $=$.

- Que x es un elemento del conjunto C se expresa "x *pertenece a* C " (y se formaliza, $x \in C$).
- Que x no es un elemento de C se expresa "x *no pertenece a* C" (y se formaliza, $\neg x \in C$ o $x \notin C$).

Tendremos en cuenta que no es necesario denotar siempre con letras mayúsculas a los conjuntos y con minúsculas a sus elementos, ya que un conjunto

puede ser a su vez un elemento de otro conjunto e incluso podemos considerar que en nuestra teoría no hay objetos que no sean conjuntos.

¿Cómo se determina una colección?

Hay dos modos:

- *Listar los objetos.* De acuerdo con la definición de Cantor un conjunto queda determinado si es posible describir completamente cuales son sus elementos. El procedimiento más sencillo es nombrar a cada uno de ellos, y se llama *definición por extensión*; es conocida la notación de encerrar entre llaves a los elementos del conjunto.

  El inconveniente de este método de listado o enumeración de los elementos del conjunto es que éste debe tener sólo un número finito de ellos y, en la práctica, uno muy pequeño

  ¿Qué hacer cuando la colección es infinita, o cuando es finita pero numerosa?

- *Describir los objetos.* Cuando el conjunto tiene infinitos elementos —como, por ejemplo, el de los número impares— o son demasiado numerosos —como el de todas las palabras que pueden formarse con el alfabeto latino— se utiliza el método de *definición por intensión,* que consiste en la descripción de un conjunto como la *extensión de un predicado,* esto es, mediante una o varias propiedades (el predicado) que caracterizan a los elementos de ese conjunto.

En principio podría tomarse cualquier lengua natural para describir los objetos (español, inglés, italiano, vasco, catalán, etc), sin embargo es preferible utilizar un *lenguaje formal* que ofrezca rigor y precisión. Dicho lenguaje debe ser adecuado y lo bastante rico; esto es, lo suficientemente expresivo como para poder describir *todas* las colecciones matemáticas. Pero también su teoría ha de ser lo suficientemente restrictiva como para limitarse *sólo* a las colecciones de objetos matemáticos. Para expresar predicados utilizaremos el lenguaje $L^{\in}$ de la la lógica de primer orden —que contiene como signos lógicos las conectivas $\neg, \vee, \wedge, \rightarrow, \leftrightarrow$ más los cuantificadores universal $\forall$ y existencial $\exists$— al que se añaden variables, igualdad y el relator binario de pertenencia. Puede ser ampliado mediante definición con los símbolos propios de las operaciones, relaciones o funciones del lenguaje específico de teoría de conjuntos.

5.1.2. La paradoja de Russell

Pero la definición de conjunto como *"colección de objetos 'describible' por un predicado"*, conduce inevitablemente a ciertas contradicciones que se llaman paradojas, siendo la más célebre la conocida como *paradoja de Russell*:

En la teoría cantoriana de conjuntos todas las propiedades permiten definir conjuntos, pero si tomamos como propiedad la de no pertenecerse a sí mismo, llegamos a una contradicción al analizar la clase

$$U = \{x \mid x \notin x\}$$

Evidentemente, para cualquier A se cumple

$$A \in U \leftrightarrow A \notin A$$

Pero, *¿qué sucede con el propio U?, ¿Es U un elemento de U?*

Si U fuera un elemento de U, debería verificar la propiedad que lo define; *i.e.*, debería no pertenecerse a sí mismo, $U \notin U$. Por otra parte, si U no fuera un elemento de U debería no verificar la propiedad que define a esta clase; *i.e.*, $\neg U \notin U$; es decir, $U \in U$. Naturalmente, la expresión resultante,

$$U \in U \leftrightarrow U \notin U$$

es una contradicción.

Se ha visto claramente que el concepto de conjunto no es tan sencillo y que identificarlo sin mayor cautela con el de colección resulta problemático. Para evitar la paradoja de Russell y otras de esta naturaleza, es necesario restringir la formación de conjuntos, lo veremos en lo que sigue. Otras paradojas, de hecho las primeras en descubrirse, afectaban a colecciones muy grandes, como por ejemplo la de los ordinales, o la de todos los conjuntos; estas colecciones no podrán pertenecer a la categoría de conjunto en la teoría axiomática de Zermelo.

5.1.3. Solución de las paradojas

Una solución radical al problema de las paradojas es la propuesta en 1903 por Russell, su *Teoría de Tipos*. Observa que en todas las paradojas conocidas hay una componente de reflexividad, de circularidad. Técnicamente se evitan las paradojas al eliminar del lenguaje las formaciones autorreflexivas[1]. Se reconoce que nuestro universo matemático no es plano, sino jerarquizado, por niveles, y que el lenguaje más adecuado para hablar de un universo así debe tener diversos tipos de variables que correspondan a cada nivel; en particular, la relación de pertenencia se da entre objetos de distinto nivel.

En 1908 Zermelo propone como solución la *Teoría Axiomática de Conjuntos*, refinada más tarde por Fraenkel, Skolem y otros. En ella se evita que las colecciones que producían las paradojas puedan ser conjuntos. Una colección de objetos será un conjunto si los axiomas la respaldan; esto es, justifican que lo sea. Dichos axiomas permiten formar conjuntos a partir de conjuntos previamente construidos y postulan la existencia del $\emptyset$ y de al menos un conjunto infinito. Sin embargo, en la teoría de von Neumann se admiten colecciones que no son conjuntos, las denominadas *clases últimas*. En ella se definen las clases mediante propiedades, sin restricción alguna, pero habrá que mostrar que se trata de conjuntos viendo que pertenecen a alguna clase, en los casos oportunos. Las clases últimas, como la clase universal o la de los ordinales, no pertenecen a ninguna otra clase.

[1] La paradoja antes mencionada no puede producirse en la teoría de tipos simple porque la sucesión de signos $X \notin X$ no es una fórmula y el axioma de definición de clases no se aplica para ella. El signo de $\in$ se aplica entre objetos de distinto tipo, $X^{\alpha} \in X^{(0,\alpha)}$

5.1.4. El Universo matemático

La idea intuitivamente más fructífera y también la más extendida es que nuestro *universo matemático*, —esto es, el que contiene todas las colecciones de objetos matemáticos, pero solamente los objetos matemáticos— constituye una jerarquía de conjuntos, la denominada *Jerarquía de Zermelo.*

En la construcción de los conjuntos que la formarán se parte de una colección inicial $\mathcal{M}_0$ de objetos dados y a continuación se construye una colección $\mathcal{M}_1$ de conjuntos de elementos de $\mathcal{M}_0$, después una colección $\mathcal{M}_2$ de conjuntos de objetos de $\mathcal{M}_0$ y $\mathcal{M}_1$, etc.

El supuesto fundamental es que los conjuntos se construyen por niveles y que por lo tanto no nos vienen dados en bloque desde un principio. En cada uno de ellos formamos nuevos conjuntos a partir de los que disponemos hasta ese nivel; es decir, los ya formados[2]. Entre ellos se forma una cadena de inclusiones

$$\mathcal{M}_0 \subseteq \mathcal{M}_1 \subseteq \mathcal{M}_2 \subseteq ...$$

y se considera que nuestros conjuntos son todos los que aparecen en sus eslabones. Así, el universo de conjuntos resultante es una jerarquía.

Para proporcionar mayor precisión debemos responder a las preguntas siguientes:

1. *¿Cuál será nuestra colección de partida, $\mathcal{M}_0$?*
2. *¿Qué conjuntos de objetos de niveles inferiores se toman para formar nuevos niveles en la jerarquía?*
3. *¿Hasta dónde se extiende esta jerarquía?*

Para responder a la primera pregunta debemos considerar si nos interesa tomar objetos que no sean conjuntos o si nos basta con partir de un primer nivel que sea sencillamente el conjunto $\emptyset$. Está claro que así se toman sólo objetos matemáticos, pero habrá que ver también que ello es suficiente y que podremos finalmente contar en el *Universo* con todos los objetos matemáticos.

Una respuesta a la segunda pregunta que parece razonable es que los nuevos conjuntos que vayan siendo admitidos, se puedan describir con nuestro lenguaje formal. Al tomar esta opción formamos la *jerarquía de conjuntos constructibles.* Otra posibilidad es tomar como objetos de un nuevo nivel a todos los posibles. Veremos que esta fue la opción de Zermelo.

Finalmente, la tercera de las preguntas es hasta donde se extiende la jerarquía. La respuesta es que no tiene fin, siempre se pueden construir nuevos niveles.

Comentario 182 *Para precisar un poco más esta representación mental de nuestro Universo matemático es conveniente contar con algunas nociones de teoría de conjuntos básica y con el concepto de ordinal. Por ello sólo volvemos sobre este asunto cuando tengamos el equipamiento necesario.*

[2] Necesitaremos, no obstante, a los números ordinales —$\Omega = Ord$— para disponer los infinitos niveles, pero esto no es un problema grave porque lo que estamos haciendo ahora es proporcionar una imagen intuitiva; no estamos definiéndolos en un sentido técnico o fuerte.

5.2. Algunas definiciones pertinentes

Definiciones de igualdad, inclusión y vacío

1. *Igualdad (Axioma de extensionalidad):* $\forall x(x \in A \leftrightarrow x \in B) \rightarrow A = B$
2. *Inclusión, subconjunto*: $A \subseteq B \leftrightarrow_{Df} \forall x(x \in A \rightarrow x \in B)$
3. *Inclusión estricta*: $A \subset B \leftrightarrow_{Df} (A \subseteq B) \wedge (A \neq B)$
4. *Conjunto vacío:* $\emptyset$ *se define* $\forall x(x \notin \emptyset)$

Definiciones de operaciones algebraicas

1. *Unión*: $\forall x(x \in A \cup B \leftrightarrow (x \in A) \vee (x \in B))$
   por extensionalidad, este conjunto es único, $A \cup B = \{x/\ (x \in A) \vee (x \in B))\}$
2. *Intersección:* $\forall x(x \in A \cap B \leftrightarrow (x \in A) \wedge (x \in B))$
   por extensionalidad, este conjunto es único, $A \cap B = \{x/\ (x \in A) \wedge (x \in B)\}$
3. *Diferencia:* $\forall x(x \in A - B \leftrightarrow (x \in A) \wedge \neg(x \in B))$
   por extensionalidad, este conjunto es único, $A - B = \{x/\ (x \in A) \wedge (x \notin B)\}$

Definiciones de clases unitarias, pares, díadas y conjunto potencia o de las partes de un conjunto

1. *Par desordenado*: $\{x, y\} = \{z/\ (z = x) \vee (z = y)\}$
2. *Clase unitaria*: $\{x\} = \{x, x\}$
3. *Par ordenado*: $\langle x, y\rangle = \{\{x\}, \{x, y\}\}$
4. *Partes de un conjunto:* $\wp(A) = \{C/\ C \subseteq A\}$

Definiciones de gran unión, gran intersección y producto cartesiano

1. *Gran unión*: $\bigcup \mathcal{A} = \{x \mid \exists A\ (A \in \mathcal{A} \wedge x \in A)\}$
2. *Gran intersección:* $\bigcap \mathcal{A} = \{x \mid \forall A\ (A \in \mathcal{A} \rightarrow x \in A)\}$
   Convención: $\bigcap \emptyset = \emptyset$
3. *Producto cartesiano de dos conjuntos:*

$$A \times B = \{z \mid \exists uv(z = \langle u, v\rangle\ \wedge\ u \in A\ \wedge\ v \in B)\}$$

Definiciones de relaciones binarias

1. *Relación binaria:* $\forall R(R$ *es una relación* $\leftrightarrow_{Df} \forall x(x \in R \rightarrow \exists yz\ (x = \langle y, z\rangle))$
   Una relación binaria es una clase de pares ordenados.
2. *Dominio:* $Dom\ R = \{x \mid \exists y\ \langle x, y\rangle \in R\}$
3. *Rango:* $Rang\ R = \{x \mid \exists y\ \langle y, x\rangle \in R\}$
4. *Campo:* $Camp\ R = Rang\ R \cup Dom\ R$

Definición de relación inversa, producto relativo y restricción

1. *Relación inversa:* $R^{-1} = \{\langle x, y\rangle \mid \langle y, x\rangle \in R\}$
2. *Producto relativo:* $R/S = \{\langle x, y\rangle \mid \exists z\ (\langle x, z\rangle \in R\ \wedge\ \langle z, y\rangle \in S\}$
3. *Restricción de una relación a un conjunto:*

$$R \restriction A = \{\langle x, y\rangle \mid \langle x, y\rangle \in R\ \wedge\ x \in A\}$$

Definiciones de propiedades de ciertas relaciones

1. R *es* **reflexiva** *si y sólo si:* $\forall x\ (x \in Camp\ R) \rightarrow \langle x, x\rangle \in R)$
2. R *es* **simétrica** *si y sólo si:* $\forall xy\ (\langle x, y\rangle \in R \rightarrow \langle y, x\rangle \in R)$
3. R *es* **transitiva** *si y sólo si:*

$$\forall xyz\ (\langle x, y\rangle \in R \wedge \langle y, z\rangle \in R \rightarrow \langle x, z\rangle \in R)$$

4. R *es* **irreflexiva** *si y sólo si:* $\forall x\ (x \in Camp\ R \rightarrow \langle x, x\rangle \notin R)$
5. R *es* **asimétrica** *si y sólo si:* $\forall xy\ (\langle x, y\rangle \in R \rightarrow \langle y, x\rangle \notin R)$
6. R *es* **intransitiva** *si y sólo si:*

$$\forall xyz\ (\langle x, y\rangle \in R \wedge \langle y, z\rangle \in R) \rightarrow \langle x, z\rangle \notin R)$$

7. R *es* **antisimétrica** *si y sólo si:* $\forall xy\ (\langle x, y\rangle \in R \wedge \langle y, x\rangle \in R \rightarrow x = y)$
8. R *está* **conectada** *syss:*

$$\forall xy\ (x, y \in Camp\ R \wedge x \neq y \rightarrow \langle x, y\rangle \in R \vee \langle y, x\rangle \in R)$$

9. R *está* **fuertemente conectada** *syss:*

$$\forall xy(x, y \in Camp\ R \rightarrow \langle x, y\rangle \in R \vee \langle y, x\rangle \in R)$$

10. R *es* **euclídea** *syss:* $\forall xyz\ (\langle x, y\rangle \in R \wedge \langle x, z\rangle \in R \rightarrow \langle y, z\rangle \in R)$
11. R *es* **incestuosa** *syss:*

$$\forall xyz\ (\langle x, y\rangle \in R \wedge \langle x, z\rangle \in R \rightarrow \exists u\ (\langle y, u\rangle \in R \wedge \langle z, u\rangle \in R))$$

Definiciones de relaciones de orden

1. R *es una relación de* **orden (parcial)** *si y sólo si* R *es una relación y* R *es reflexiva, antisimétrica y transitiva*

2. R *es un* **orden (parcial)** *sobre* A *si y sólo si* $Camp\ R = A$ *y* R *es una relación de orden*

3. R *es un* **orden lineal si y sólo si** R *es una relación de orden y* R *esta conectada*

4. R *es un* **orden lineal sobre** A *si y sólo si* $Camp\ R = A$ *y* R *es una relación de orden lineal*

5. *Un* **conjunto parcialmente ordenado** *es un par* $\langle A, R\rangle$ *formado por un conjunto* A *y un orden parcial sobre* A.

6. *Un* **conjunto linealmente ordenado** *es un par* $\langle A, R\rangle$ *formado por un conjunto* A *y un orden lineal sobre* A

7. *Sea* $\langle A, R\rangle$ *un conjunto parcialmente ordenado y sea* $Y \subseteq A$.
   Un elemento $a \in Y$ *es un* **elemento minimal** *de* Y *si y sólo si*

   $$\neg\exists x(x \in Y \wedge \langle x, a\rangle \in R)$$

   Un elemento $a \in Y$ *es* **primer elemento** *de* Y *(mínimo de* Y*) si y sólo si*

   $$\forall x(x \in Y \rightarrow \langle a, x\rangle \in R)$$

8. *Un conjunto parcialmente ordenado* $\langle A, R\rangle$ *está* **bien fundado** *si cada subconjunto no vacío de* A *posee elemento minimal.*

9. *Cuando* $\langle A, R\rangle$ *está linealmente ordenado y bien fundado decimos que está* **bien ordenado.**

Definición de funciones, composición y propiedades

1. *Función:* f *es una función si y sólo si* f *es una relación y*

   $$\forall xyz(\langle x, y\rangle \in f \wedge \langle x, z\rangle \in f \rightarrow y = z)$$

2. *Composición:* $f \circ g = g/f$

3. f *es una función de* A *en* B *si y sólo si* f *es una función y* $Dom\ f = A$ *y* $Rang\ f \subseteq B$

4. f *es una función parcial de* A *en* B *si y sólo si* f *es una función y* $Dom\ f \subseteq A$ *y* $Rang\ f \subseteq B$

5. f *es una* **función inyectiva** *si y sólo si* f *es una función y*

$$\forall xyz\ (\langle x,y\rangle \in f \wedge \langle z,y\rangle \in f \rightarrow x = z)$$

6. $f : A \longrightarrow B$ *es* **exhaustiva** *si y sólo si* $Rang\ f = A$

7. $f : A \longrightarrow B$ *es* **biyectiva** *si y sólo si* f *es inyectiva y exhaustiva*

5.3. Buenos órdenes e inducción

En un conjunto bien ordenado todos los subconjuntos no vacíos tienen un primer elemento. El principio de inducción es una consecuencia de esta importante propiedad.

De entre los subconjuntos de números naturales destacan por su interés los que poseen la siguiente propiedad: *si tomamos un número cualquiera del conjunto y le sumamos uno, el resultado también está en el conjunto.*

Llamemos *inductivos* a los subconjuntos de los naturales que exhiban semejante comportamiento y formulemos el principio de inducción así:

> *Si un conjunto* G *de números naturales es tal que: 0 es elemento de* G *y* G *es inductivo, entonces* $G = \mathbb{N}$

Podemos expresarlo de esta manera

$$\mathcal{P}(0)\&\forall n(\mathcal{P}(n) \Rightarrow \mathcal{P}(n+1)) \Rightarrow \forall n\mathcal{P}(n)$$

No voy a entrar en la aplicabilidad de este principio, sino en su justificación.

¿Por qué es verdad?

En la aritmética de Peano se toma como axioma y por lo tanto no tiene sentido preguntárselo. Pero estamos en un contexto mucho más amplio, tanto que tendría que servir de fundamento, incluso de teorías tan potentes como la misma aritmética; aquí debería no ser un principio, sino derivarse de otros más generales. Para contestar a la pregunta resulta útil plantearse esta primero:

¿por qué funciona el principio de inducción?

Si queremos demostrar que todos los números tienen una determinada propiedad $\mathcal{P}$ es obvio que, al tratarse de un conjunto infinito, no podemos comprobar uno a uno que todos los naturales cumplen $\mathcal{P}$; sin embargo, basta con ver que

$$\mathcal{P}(0)\&\forall n(\mathcal{P}(n) \Rightarrow \mathcal{P}(n+1))$$

¿Por qué?

Esta es la razón:

Supongamos que no todos los naturales tuvieran la propiedad $\mathcal{P}$; es decir,

$$\{n \mid \neg\mathcal{P}(n)\} \neq \emptyset.$$

Habría un primer elemento de este conjunto; esto es, habría un m para el que valdría $\neg\mathcal{P}(m)$ pero también, por ser m el primer elemento, valdría $\mathcal{P}(m-1)$.

Esto es justamente lo que queda excluido en una prueba por inducción; porque demostramos

$$\forall n(\mathcal{P}(n) \Rightarrow \mathcal{P}(n+1))$$

Retrocedamos un poco, *¿por qué tiene que haber un primer elemento en el conjunto?*

Lo que hace que funcione el principio de inducción matemática es el buen orden de los naturales que nos garantiza que todo subconjunto tiene primer elemento. Esto es, la seguridad de que $\forall n\mathcal{P}(n)$ se verifica en cuanto somos capaces de probar que vale para el cero y para el siguiente de todos los que tienen la propiedad, se apoya en el principio del buen orden.

¿Se puede extender este método para que sirva no sólo con los conjuntos numerables, sino también con los transfinitos (supernumerables)?

La respuesta es afirmativa, lo hacemos introduciendo a los ordinales.

Teorema 183 *Sea $\langle X, \leq\rangle$ un conjunto bien ordenado. Y sea $E \subseteq X$ tal que:*
(1) el primer elemento de X es elemento de E
(2) para cada $x \in X$, si $\forall y(y < x \rightarrow y \in E)$, entonces $x \in E$
entonces $E = X$

La inducción para los naturales es una consecuencia directa de la inducción para conjuntos bien ordenados cualesquiera.

Comparación de conjuntos bien ordenados. Isomorfismos.

Definición 184 *Sean $\langle X, \leq\rangle$ y $\langle X^*, \leq^*\rangle$ dos conjuntos bien ordenados.*
*$f : X \longrightarrow X^*$ es un **isomorfismo de órdenes** si y sólo si*
(i) f es biyectiva
(ii) $x < y \Rightarrow f(x) <^* f(y)$, para todo $x, y \in X$

Teorema 185 *Sea $\langle X, \leq\rangle$ un conjunto bien ordenado, $Y \subseteq X$ y $f : X \cong Y$. Entonces $x \leq f(x)$, para todo $x \in X$.*

Teorema 186 *Sean $\langle X, \leq\rangle$ y $\langle X^*, \leq^*\rangle$ buenos órdenes. Si $\langle X, \leq\rangle \cong \langle X', \leq'\rangle$, entonces el isomorfismo es único.*

Comentario 187 *El que se trate de un buen orden es esencial en este teorema, no bastaría que el orden fuera lineal.*

5.3.1. Segmento

Definición 188 *Sea $\langle X, \leq\rangle$ un conjunto bien ordenado y $a \in X$. Llamamos* segmento *de X determinado por a al conjunto*

$$X_a = \{x \in X \mid x < a\}$$

Proposición 189 *Sea* $\langle X, \leq \rangle$ *un conjunto bien ordenado. No hay ningún isomorfismo de* X *en un segmento de* X

Proposición 190 *Sea* $\langle X, \leq \rangle$ *un conjunto bien ordenado y sea*

$$A = \{X_a \mid a \in X\}$$

Entonces $\langle X, \leq \rangle \cong \langle A, \subseteq \rangle$

5.3.2. Ordinal

Definición 191 *Un* ordinal *es un conjunto bien ordenado* $\langle X, \leq \rangle$ *tal que* $X_a = a$*, para todo* $a \in X$

Teorema 192 *Sea* $\langle X, \leq \rangle$ *un conjunto bien ordenado. Entonces las tres condiciones son equivalentes*

$$(\forall x, y \in X)(x < y \leftrightarrow X_x \subset X_y \leftrightarrow x \subset y)$$

Teorema 193 *Sea* X *un ordinal. Si* $a \in X$*, entonces* X_a *es un ordinal.*

Teorema 194 *Sea* X *un ordinal. Sea* $Y \subseteq X$*. Si* Y *es un ordinal, entonces* $Y = X_a$*, para algún* $a \in X$*.*

Teorema 195 *Si* X *e* Y *son ordinales, entonces* $X \cap Y$ *es un ordinal.*

Teorema 196 *Sean* X *e* Y *ordinales. Si* $X \neq Y$*, entonces uno es un segmento del otro.*

Teorema 197 *Si* X *e* Y *son ordinales isomorfos, entonces* $X = Y$

Teorema 198 *Sea* $\langle X, \leq \rangle$ *un conjunto bien ordenado tal que para cada* $a \in X$*,* X_a *es isomorfo a un ordinal. Entonces* X *es isomorfo a un ordinal.*

Cada conjunto bien ordenado es isomorfo a un único ordinal.

Comentario 199 *Observaciones acerca de los ordinales:*
Los Ordinales miden la longitud de los conjuntos bien ordenados
Si $\langle X, \leq \rangle$ *es un conjunto bien ordenado,* $Ord(X)$ *es el único ordinal isomorfo a* X*.*
Por otra parte, si X *e* Y *son conjuntos bien ordenados,*

$$X \cong Y \quad syss \quad Ord(X) = Ord(Y)$$

Esta unicidad nos permite usar a los ordinales como "vara de medir" conjuntos bien ordenados; es decir, $Ord(X)$ *es la longitud de* X*.*
Inclusión (y pertenencia) bien-ordena a los ordinales

5.4. Teoría axiomática de conjuntos

Recordemos los componentes de una teoría axiomática:

1. El lenguaje o signos formales de la teoría.
2. Los axiomas, que son proposiciones acerca de los objetos de la teoría y que imponen el funcionamiento de dichos objetos.
3. Los teoremas, que son todas las proposiciones demostrables con herramientas lógicas a partir de los axiomas.

En la teoría de conjuntos axiomática de Zermelo-Fraenkel se usará el lenguaje de la lógica de primer orden. Las variables de dicho lenguaje se referirán a conjuntos; es decir, en la interpretación usual todos los objetos serán conjuntos. Es decir, *existir* será sinónimo de *ser un conjunto.* El lenguaje básico sólo tiene el relator binario de pertenencia, pero se extiende, mediante definiciones pertinentes, para dar cabida a operaciones, como vimos anteriormente.

Los conceptos primitivos de esta teoría son el de conjunto y el de pertenencia. En realidad la mayoría de los axiomas sirven para garantizar la existencia de los conjuntos que nos interesa tener. Por ello la idea de *construcción* es esencial en la teoría axiomática de Zermelo-Fraenkel (que notaremos ZF).

En la versión axiomática de la teoría de conjuntos se respeta la idea fundamental de aceptar que una colección de objetos pueda ser un conjunto, pero se impone la condición extra de que todos los objetos de una colección deben haberse formado antes de definir dicha colección, y de esta manera se evitarán los problemas que conducían a las paradojas. Uno de los axiomas de la teoría —se verá con detalle más adelante— impondrá esta restricción:

> *"Si X es un conjunto ya construido, existe un conjunto Y formado por los elementos de X que satisfacen un predicado $\mathcal{P}$ que los describe —o lo que es lo mismo, una fórmula con al menos una variable libre—".*

Así un predicado describirá un conjunto sólo si los objetos han sido ya construidos —son de otro conjunto X— y además satisfacen el predicado.

Añadiendo esta restricción a la definición de conjunto de Cantor desaparece la paradoja de Russell ya que para que

$$U = \{x \mid x \notin x\}$$

sea un conjunto se debería tener previamente el conjunto X a partir del cual se construyese; es decir,

$$U = \{x \in X \mid x \notin x\}$$

¿Cómo se resuelve la paradoja?

Como se requiere partir de un conjunto ya admitido desaparece la contradicción. Ahora, para cualquier B se verifica:

$$B \in U \leftrightarrow (B \in X \wedge B \notin B)$$

En realidad, puesto que la condición $B \notin B$ la cumplen todos, U será el propio X. No estamos así definiendo a la clase universal, es imposible que exista el conjunto de todos los conjuntos; esa colección no tiene cabida aquí.

Los teoremas de ZF se derivan de los axiomas, pero para que tengan un cierto interés deben mediar definiciones de conceptos y operaciones nuevas. Aunque en principio podría usarse un cálculo deductivo de primer orden, en la práctica resulta desaconsejable pues en él cualquier demostración se alargaría en exceso.

Los axiomas de la teoría ZF son propiedades indemostrables. que se aceptan como verdaderas y que tienen por objeto garantizar que en la Jerarquía de Conjuntos ZF todo lo construido sean conjuntos y así evitar las paradojas.

5.4.1. Axiomas de Extensionalidad y de Separación

Axioma 200 *(de Extensionalidad):*

$$\forall AB\ (A = B \leftrightarrow \forall x(x \in A \leftrightarrow x \in B))$$

Este axioma asegura que el signo lógico $=$ para la igualdad de objetos de la teoría coincide con la intuición de que dos conjuntos son iguales si tienen los mismos elementos[3]*.*

Axioma 201 *(de Separación):* $\forall A \exists B\ \forall x(x \in B \leftrightarrow x \in A \wedge C(x))$
Expresa que si $C(x)$ es una fórmula del lenguaje de la teoría de conjuntos, con a lo sumo la variable x libre y A es un conjunto, entonces la clase (colección)

$$\{x \mid x \in A \wedge C(x)\}$$

es un conjunto. El axioma de separación —llamado también de los subconjuntos— obliga a que los conjuntos estén formados de elementos de conjuntos ya construidos. Hay que observar que más que un sólo axioma es un esquema de axiomas, pues tenemos uno para cada predicado.

5.4.2. Axiomas del Par, de la Unión y de las Partes

Axioma 202 *(del Par)*

$$\forall A \forall B \exists C\ \forall x(x \in C \leftrightarrow x = A \vee x = B)$$

Expresa que si A y B son conjuntos entonces la clase $\{A, B\}$ es un conjunto. En particular, si A es un conjunto entonces la clase $\{A\}$ es un conjunto (por extensionalidad). Este axioma asegura que las colecciones de conjuntos son conjuntos.

[3] En realidad sólo un sentido de la flecha tiene significado conjuntista: la flecha hacia la derecha es una propiedad de la igualdad, que nada tiene de especial en teoría de conjuntos, comparten todas sus propiedades los objetos iguales; la flecha hacia la izquierda indica algo peculiar de la teoría de conjuntos, que sólo es relevante los individuos que pertenezcan a un determinado conjunto, no las propiedaes que les hacen pertenecer a ellos.

Axioma 203 *(de la Unión):* $\forall A \exists B \forall x (x \in B \leftrightarrow \exists y (y \in A \wedge x \in y))$
Expresa que si A *es un conjunto la gran unión de* A, $\bigcup A$, *es un conjunto.*

Axioma 204 *(de las Partes):* $\forall A \exists B \forall x (x \in B \leftrightarrow x \subseteq A)$
Si A *es un conjunto, entonces las partes de* A, $\wp(A)$, *es un conjunto.*

Teoremas

A partir de los cinco primeros axiomas se obtienen los resultados siguientes que nos garantizan que las clases definidas con anterioridad son conjuntos.

Proposición 205 *Si* A *y* B *son conjuntos entonces* $A \cup B$, $A \cap B$, $A - B$, $\{A, B\}$, $\wp(A)$, $\bigcup A$, $\bigcap A$ *son conjuntos.*
Demostración. *Es consecuencia inmediata de los axiomas.* ■

Proposición 206 *Si* A *y* B *son conjuntos entonces* $A \times B$ *es un conjunto.*
Demostración. *Es consecuencia de que*

$$A \times B \subseteq \wp(\wp(A \cup B))$$

y de los axiomas de la unión y de las partes. ■

Como consecuencia de la proposición precedente las relaciones obtenidas a partir de productos cartesianos también serán conjuntos.

Lo que no se puede deducir a partir de los axiomas anteriores es que la imagen por una función de un conjunto sea un conjunto. Necesitamos un nuevo axioma para ello.

5.4.3. Axioma de Reemplazamiento

Axioma 207 ***Axioma de Reemplazamiento***:

$$\forall x \exists! y C(x, y) \rightarrow \forall A \exists B \forall y (y \in B \leftrightarrow \exists x (x \in A \wedge C(x, y))$$

Expresa que la imagen de un conjunto por una función es un conjunto.

5.5. La Jerarquía de Zermelo

5.5.1. Construcción de la Jerarquía

Ahora podemos presentar con rigor la jerarquía de conjuntos de Zermelo Fraenkel, y responder a las preguntas que nos hicimos cuando hablábamos del *universo matemático* y se proponía la construcción de los conjuntos de la teoría partiendo de una colección inicial $\mathcal{M}_0$ de objetos dados y formando a continuación una colección $\mathcal{M}_1$ de objetos de $\mathcal{M}_0$, después $\mathcal{M}_2$ de objetos de $\mathcal{M}_0$ y de $\mathcal{M}_1$ y así sucesivamente. Una posibilidad sería la siguiente: reunir en el

conjunto $\mathbf{A}$ a todos los objetos que sin ser conjuntos queremos que formen parte de ellos, les llamaremos *átomos.* Y procedemos a construir la jerarquía

$$\mathcal{M}_0 \subseteq \mathcal{M}_1 \subseteq \mathcal{M}_2 \subseteq ...$$

poniendo $\mathcal{M}_0 = \mathbf{A}$; en el nuevo nivel $\mathcal{M}_1$ ponemos además a los conjuntos que se pueden formar con los átomos

$$\mathcal{M}_1 = \mathcal{M}_0 \cup \wp(\mathcal{M}_0) = \mathbf{A} \cup \wp(\mathbf{A})$$

El tercer nivel incluye lo anterior y todos los conjuntos de objetos del nivel precedente

$$\mathcal{M}_2 = \mathcal{M}_1 \cup \wp(\mathcal{M}_1)$$

En general

$$\mathcal{M}_{n+1} = \mathcal{M}_n \cup \wp(\mathcal{M}_n)$$

Sin embargo, aún siendo infinita, esta jerarquía no contiene suficientes conjuntos; por ejemplo, aunque $\varnothing \in \mathcal{M}_1$, $\{\varnothing\} \in \mathcal{M}_2$, $\{\{\varnothing\}\} \in \mathcal{M}_3$, no hay ningún conjunto que los reúna a todos,

$$\{\varnothing,\ \{\varnothing\},\ \{\{\varnothing\}\}, ...\}$$

por lo que añadimos un nuevo nivel ω

$$\mathcal{M}_\omega = \mathcal{M}_0 \cup \mathcal{M}_1 \cup \mathcal{M}_2 \cup ...$$

La construcción no termina aquí, pues seguimos añadiendo niveles.

Esta es la descripción habitual de la jerarquía de Zermelo, que se representa normalmente como un cono en cuyo eje se sitúan los ordinales (ver figura: 5.1). A mí me gusta representarlos como una escalera interminable en cuya superficie se hallan los ordinales (ver figura: 5.2).

No obstante, podemos simplificar la jerarquía haciendo que el conjunto $\mathbf{A}$ sea $\varnothing$. Perdemos así la posibilidad de formar conjuntos de sillas, o de calcetines, pero en realidad sólo nos interesan los objetos matemáticos y éstos están, como veremos.

1. *¿Cuál será nuestra colección de partida, $\mathcal{M}_0$?*
   Nuestro origen es $\mathcal{M}_0 = \emptyset$; en el nivel inicial sólo ponemos el conjunto vacío, que denotaremos $\mathcal{V}_0$.
   $$\mathcal{V}_0 = \emptyset$$

2. *¿Qué conjuntos de objetos de niveles inferiores se toman para formar nuevos niveles en la jerarquía?*
   Supóngase que hemos definido ya $\mathcal{V}_\alpha$.
   ¿Qué conjuntos de miembros de $\mathcal{V}_\alpha$ tomaremos para formar $\mathcal{V}_{\alpha+1}$?.
   Simplemente consideraremos $\wp(\mathcal{V}_\alpha)$ el conjunto potencia o de las partes

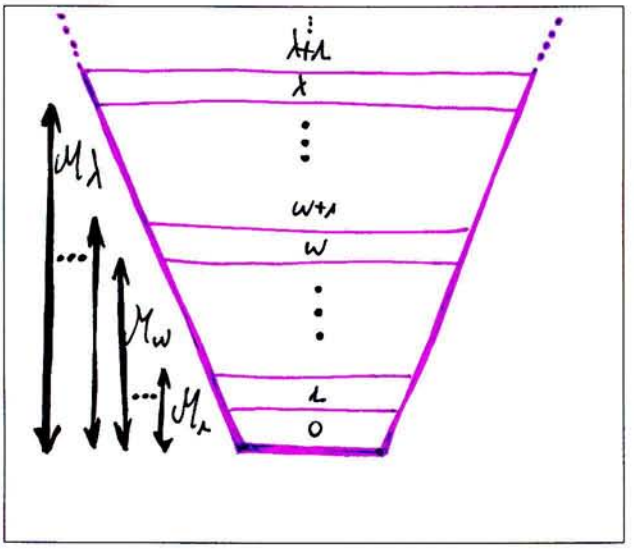


Figura 5.1: *Jerarquía con átomos*

de $\mathcal{V}_\alpha$, cuyos elementos son los subconjuntos de $\mathcal{V}_\alpha$. Por tanto, dado el nivel $\mathcal{V}_\alpha$ formamos

$$\mathcal{V}_{\alpha+1} = \wp(\mathcal{V}_\alpha)$$

Y cuándo λ es un ordinal límite y hemos definido $\mathcal{V}_\alpha$ para cada $\alpha < \lambda$, definimos

$$\mathcal{V}_\lambda = \bigcup_{\alpha<\lambda} \mathcal{V}_\alpha$$

Es decir, en estos momentos recopilamos todos los niveles anteriores.
Esta definición por casos —cero, ordinal sucesor y límite— puede comprimirse, para cualquier α

$$\mathcal{V}_\alpha = \bigcup_{\beta<\alpha} \wp(\mathcal{V}_\beta)$$

3. *¿ Hasta dónde se extiende esta jerarquía?*
La jerarquía de conjuntos no tiene fin; siempre se pueden construir nuevos conjuntos. Para cada ordinal α debe haber un nivel $\mathcal{V}_\alpha$. Sus miembros son conjuntos cuyos elementos están en los niveles previos; en

$$\bigcup_{\beta<\alpha} \mathcal{V}_\beta$$

Finalmente, la jerarquía de Zermelo Fraenkel de todos los conjuntos es:

$$\mathcal{V} = \bigcup_{\alpha\in Ord} \mathcal{V}_a$$

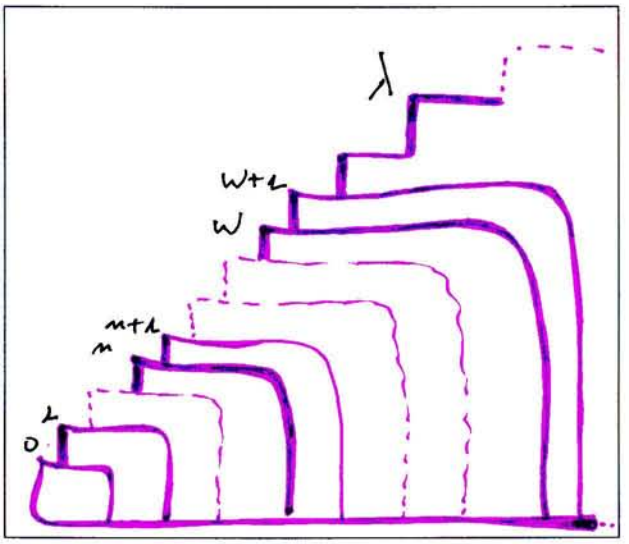


Figura 5.2: *Escalera de bloques*

Como vemos, la construcción se efectúa de manera muy similar a los $\mathcal{M}_\alpha$; mediante recursión sobre los ordinales formamos la cadena de los distintos niveles

$$\mathcal{V}_0 \subseteq \mathcal{V}_1 \subseteq \mathcal{V}_2 \subseteq ...$$

de la forma arriba indicada: $\mathcal{V}_0 = \emptyset$, $\mathcal{V}_1 = \wp(\mathcal{V}_0) = \{\emptyset\}$, $\mathcal{V}_2 = \wp(\mathcal{V}_1) = \{\emptyset, \{\emptyset\}\}$, etc.

De hecho, se simplifica algo para los ordinales sucesores, como los números naturales, pues siempre que tengamos un nivel $\mathcal{V}_n$ construiremos el nivel siguiente

$$\mathcal{V}_{n+1} = \wp(\mathcal{V}_n)$$

Y para construir el nivel $\mathcal{V}_\omega$ tomamos la unión infinita de los niveles previos, como en $\mathcal{M}_\omega$

$$\mathcal{V}_\omega = \mathcal{V}_0 \cup \mathcal{V}_1 \cup \mathcal{V}_2 ...$$

y continuamos,

$$\mathcal{V}_{\omega+1} = \wp(\mathcal{V}_\omega)$$

Comentario 208 *Si comparamos esta jerarquía con la que admite átomos vemos que esta es más estilizada y su construcción más simple (ver figura: 5.3).*

Como se comentó con anterioridad la jerarquía de Zermelo es una buena representación intuitiva de nuestro universo matemático. Sería conveniente que respondiera a todas las expectativas planteadas en apartados precedentes y también, lo cual es extremadamente importante, que el marco axiomático en el que

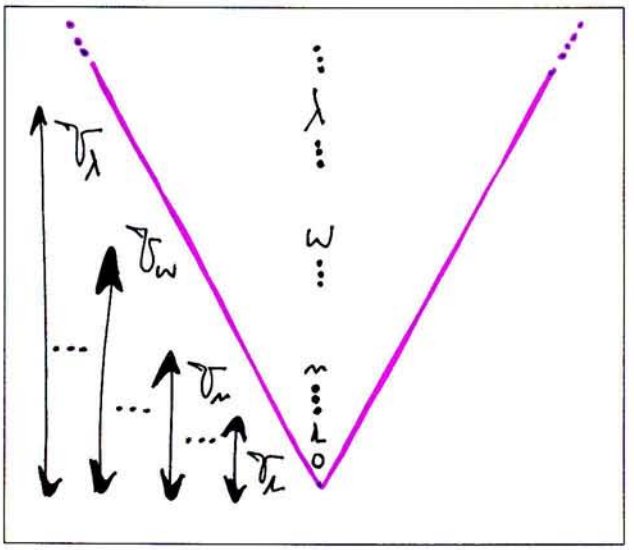


Figura 5.3: *Cono de conjuntos*

nos movemos; a saber, la teoría de Zermelo Fraenkel, garantizara su construcción. De hecho, se puede ver al revés, se eligieron los axiomas justamente para que nuestra construcción pudiera llevarse a término.

- Se observa en primer lugar que en toda la construcción hay una *operación básica*: formar el conjunto potencia o de las partes de un conjunto $\wp(x)$
- Otra de nuestras preocupaciones era que pudieran construirse *sólo* colecciones matemáticas. Esto está garantizado porque hemos optado por empezar con el $\emptyset$, descartando a los átomos, y formamos el universo

$$\mathcal{V} = \bigcup_{\alpha \in Ord} \mathcal{V}_a$$

  mediante construcciones que no introducen objetos que no sean matemáticos. Sin embargo, $\mathcal{V}$ no es un conjunto.

- Finalmente, *todas* las colecciones matemáticas están, porque tenemos el axioma de separación.

5.5.2. Axiomas involucrados en la construcción de la jerarquía

Hemos ya formulado el axioma de separación y dejaremos para un tratamiento posterior los axiomas de elección y constructibilidad.

¿Cuales son los principios usados en la construcción de la Jerarquía?

Claramente

$$\mathcal{V} = \bigcup_{\alpha \in Ord} \mathcal{V}_a$$

no es una fórmula del lenguaje de primer orden que estamos usando. En su construcción usamos lo siguiente:

1. Tomamos como operación básica la de partes, $\wp(x)$. El axioma de las partes de un conjunto

   $$\forall A \exists B(\forall x(x \in B \leftrightarrow \forall y(y \in x \rightarrow y \in A))$$

   permite formar un nuevo conjunto con todos los subconjuntos de uno dado, A. Este axioma nos permite pasar de $\mathcal{V}_\alpha$ a $\mathcal{V}_{\alpha+1}$, haciendo $\mathcal{V}_{\alpha+1} = \wp(\mathcal{V}_\alpha)$

   ¿Qué sucede cuando α es un ordinal límite?

2. Debemos poder formar la unión de colecciones de conjuntos. El axioma de la unión,

   $$\forall A \exists B(\forall x(x \in B \leftrightarrow \exists y(y \in A \wedge x \in y))$$

   dice que dado un conjunto A hay un conjunto cuyos elementos son los elementos de los elementos de A. El axioma de la unión nos permite formar $\mathcal{V}_\alpha$, cuando α es un ordinal límite, haciendo

   $$\mathcal{V}_\alpha = \bigcup_{\beta < \alpha} \mathcal{V}_\beta$$

   Es decir, $\mathcal{V}_\alpha = \bigcup \{\mathcal{V}_\beta \mid \beta < \alpha\}$.
   Pero, *¿sabemos si $\{\mathcal{V}_\beta \mid \beta < \alpha\}$ es un conjunto?*.
   En realidad, podríamos conseguirlo a partir de $\{\beta \mid \beta < \alpha\}$ reemplazando cada β por $\mathcal{V}_\beta$. Para ello necesitaríamos contar en la jerarquía de conjuntos con los ordinales y utilizar el axioma del reemplazamiento. Dejemos de momento de lado a los ordinales, supongamos que ya los tenemos.

3. El axioma de reemplazamiento dice que si tenemos una fórmula $C(x, y)$ tal que a cada conjunto a le asigna un único conjunto b tal que $C(a, b)$ entonces, a partir de un conjunto A cualquiera podemos definir otro B en el que los elementos a de A son reemplazados por los b que cumplen $C(a, b)$

   $$\forall x \exists! y C(x, y) \rightarrow \forall A \exists B \forall y(y \in B \leftrightarrow \exists x(x \in A \wedge C(x, y)))$$

   Retomemos la cuestión anterior,

   ¿Qué necesitamos para poder construir los ordinales?

4. En primer lugar, necesitamos el conjunto vacío $\emptyset$. Para ello añadimos el axioma que dice que hay un conjunto que carece de elementos

   $$\exists B \forall x(x \notin B)$$

5. También necesitamos el axioma de infinitud, que dice que hay un conjunto que contiene al $\emptyset$ y que es inductivo —está cerrado bajo la operación del siguiente—

$$\exists B(\emptyset \in B \wedge \forall y(y \in B \rightarrow y \cup \{y\} \in B))$$

Los dos últimos axiomas, afirman la existencia de ciertos conjuntos. Esto contrasta con el resto, que proporcionan reglas mediante las cuales se forman conjuntos a partir de conjuntos existentes. (En realidad, el axioma del conjunto vacío no es necesario, pues es demostrable a partir de infinitud y de separación.)
¿Faltan más axiomas?

6. El axioma de extensionalidad, que usábamos desde el principio en la teoría básica, expresa el criterio fundamental de la teoría de conjuntos que dice que identificamos los conjuntos que tienen los mismos elementos. No estamos interesados en los predicados que definen a los conjuntos, sino en los objetos que finalmente caen bajo ellos, los que los cumplen. Sobre todo, el principio de extensionalidad nos permite demostrar la unicidad de los conjuntos cuya existencia garantizan otros axiomas.

¿Hemos expresado ya que el universo de conjuntos está formado exclusivamente por los elementos de los distintos niveles?

$$\mathcal{V} = \bigcup_{\alpha \in Ord} \mathcal{V}_\alpha$$

En realidad, cuando se tiene el resto de los axiomas mencionados y el axioma de separación, se puede demostrar que el principio así expresado equivale al siguiente

7. *Axioma de fundación.* Dicho principio dice que $\in$ es una relación bien fundada. Lo expresamos así:

$$\forall x(x \neq \emptyset \rightarrow \exists y(y \in x \wedge y \cap x = \emptyset))$$

Añadamos, pues estos axiomas a la lista

Axioma 209 *(conjunto vacío)* $\emptyset$. $\exists B \forall x(x \notin B)$

Axioma 210 *(de infinitud)* $\exists B(\emptyset \in B \wedge \forall y(y \in B \rightarrow y \cup \{y\} \in B))$

Axioma 211 *(de fundación)* $\forall x(x \neq \emptyset \rightarrow \exists y(y \in x \wedge y \cap x = \emptyset))$

5.6. Los Axiomas de Elección y Constructibilidad

Los axiomas presentados anteriormente permiten justificar que todos los elementos utilizados en la construcción de la Jerarquía de Conjuntos de Zermelo

Fraenkel son conjuntos. Esos axiomas y la teoría de conjuntos que se deriva de ellos se conoce como teoría *ZF*.

Existen extensiones de *ZF* consistentes en añadirle uno o varios axiomas; sólo nos ocuparemos ahora de las que se obtienen con dos de ellos en particular: el de *elección* y el de *constructibilidad*.

5.6.1. Axioma de elección

Su enunciado es muy sencillo. Sea

$$\mathfrak{B} = \{A_i \mid i \in I\}$$

una colección infinita de conjuntos no vacíos y disjuntos dos a dos (ver figura: 5.4)

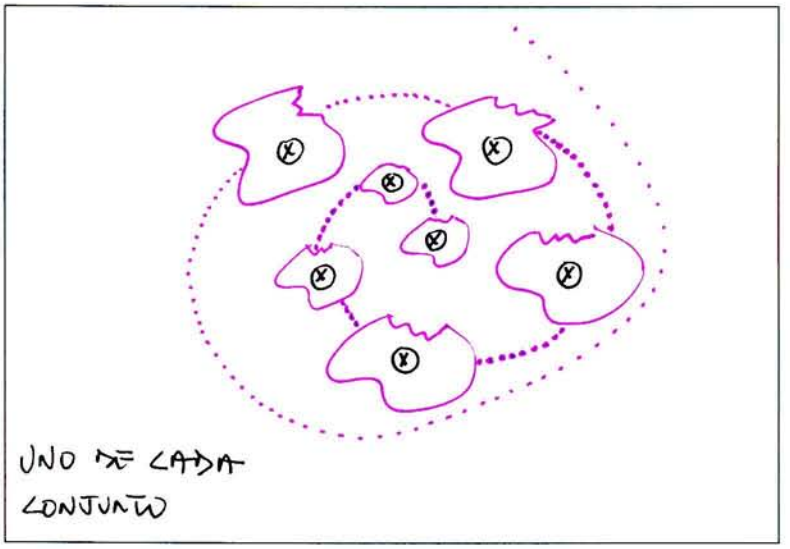


Figura 5.4: *Función de elección*

El axioma establece la existencia de un conjunto que toma un elemento x_i de cada conjunto $A_i \in \mathfrak{B}$. Su formulación podría hacerse así:

> *Dado un conjunto $\mathfrak{B}$ de conjuntos no vacíos y disjuntos, hay un conjunto que tiene un elemento de cada conjunto de $\mathfrak{B}$*

Russell pone el famoso ejemplo de los calcetines: Dada una colección infinita de pares de calcetines, el axioma de elección nos permite elegir uno de cada par. El problema se plantea con los calcetines, pero no con los zapatos, ya que podemos elegir el izquierdo de cada par no necesitando axioma de elección para ello.

¿Por qué un axioma tan sencillo ha originado tanta controversia?

Muy posiblemente se deba a la naturaleza no constructiva del mismo, a que asegura la existencia de un conjunto, pero sin indicar cómo se construye.

La mayor parte de los axiomas hasta ahora introducidos nos indican el modo de formar conjuntos a partir de otros previamente construidos, mediante operaciones de diversa índole, o porque satisfacen ciertos predicados.

La formulación explícita del axioma de elección se atribuye a Zermelo, aunque Peano ya aludió a él y Cantor lo usó, posiblemente sin advertirlo. En 1904 Zermelo demostró que todo conjunto puede ser bien ordenado usando la denominada *función de elección* que de cada subconjunto de un conjunto cualquiera elige un elemento. Esta versión difiere levemente de la mencionada, pero es equivalente. El principio del buen orden no es nada desdeñable, es fundamental para realizar pruebas por inducción sobre conjuntos de cualquier cardinalidad y equivale o implica muchos principios matemáticos básicos, como veremos más adelante.

Tomemos como axioma la primera de las formulaciones y expresémosla en lógica de primer orden.

Axioma 212 *(de elección):* $\forall A(\emptyset \notin A \wedge \forall xy(x \in A \wedge y \in A \wedge x \neq y \rightarrow x \cap y = \emptyset) \rightarrow \exists B \forall z(z \in A \rightarrow \exists! v(v \in z \cap B)))$

Asegura la existencia de un conjunto B obtenido a partir de una colección cualquiera A de conjuntos no vacíos y disjuntos dos a dos. De cada conjunto de A *elige* un único conjunto para poner en B.

Este enunciado no se puede derivar del resto en la teoría *ZF*, y si queremos asegurar la existencia de las llamadas funciones de elección o de la buena ordenación de cualquier conjunto, debe añadirse. La teoría de conjuntos axiomática resultante se conoce con las siglas *ZFC* (*C* del inglés Choice), y es la que en realidad se toma como teoría de Zermelo Fraenkel.

Si *ZFC* fuera consistente tendría un modelo[4] y todo lo que se demostrase formalmente en esta teoría sería verdadero en el modelo. Sin embargo, como consecuencia del teorema de Gödel no hay esperanza de poder demostrar la consistencia de la teoría *ZFC*, ya que para hacerlo tendríamos que efectuar la prueba en una teoría más potente que *ZFC*, cuya consistencia sería aún más difícil de establecer. Otro de los resultados de Gödel muestra que si *ZFC* fuera inconsistente también lo sería *ZF*. Por lo tanto sólo es necesario suponer que *ZF* es consistente para asegurar también que *ZFC* lo es.

[4] Nos gustaría que el universo de Zermelo, junto a la relación habitual de pertenecia

$$\mathcal{U} = \langle \mathcal{V}, \in^{\mathcal{U}} \rangle$$

fuera un modelo de la teoría *ZF*. Sin embargo, para evitar problemas graves de autorreferencia habíamos convenido en exigir que los universoso de las estructuras fueran conjuntos y $\mathcal{V}$ no lo es.

Otras formulaciones del axioma de elección

De entre las numerosas formulaciones del axioma de elección, cabe destacar las siguientes:

1. Para cada relación R hay una función f tal que

$$f \subseteq R \text{ y } Dom(f) = Dom(R)$$

   Esta formulación la usamos para demostrar el criterio de exhaustividad para funciones. A saber, que una función $f : A \longrightarrow B$ es exhaustiva si y sólo si hay una función $g : B \longrightarrow A$ que compuesta con ella produce la identidad sobre B. Puesto que la función f no tiene porqué ser inyectiva, su recíproca normalmente no será función. Para asegurarnos de que existe g usamos *elección.*

2. El producto cartesiano de conjuntos no vacíos es no vacío.

3. Para cada conjunto A hay una función f —llamada *función de elección*— tal que el dominio de f es el conjunto de los subconjuntos no vacíos de A y tal que $f(B) \in B$ para cada $B \subseteq A$.

La demostración de la equivalencia entre estas formulaciones es fácil, y también lo es que todas se derivan de la que hemos tomado, esto es, el axioma número 212. Veamos, pues, que hay siempre una función de elección, usando para ello el axioma.

Proposición 213 *La formulación precedente se sigue del axioma de elección.*
Demostración. *Sea* $\mathfrak{B}$ *una familia de conjuntos no vacíos*

$$\mathfrak{B} = \{X \mid X \in \mathfrak{B}\}$$

Para poder aplicar el axioma de elección necesitaríamos que fueran disjuntos dos a dos, para lo que utilizamos el siguiente truco. Para cada $X \in \mathfrak{B}$ *sea* S_X *el conjunto de pares ordenados* $\langle X, a\rangle$ *tales que* $a \in X$

$$S_X = \{X\} \times X$$

Ahora la colección $\{S_X \mid X \in \mathfrak{B}\}$ *está formada por conjuntos no vacíos y disjuntos. Utilizando el axioma elegimos un* z_X *de cada* S_X *que será de la forma*

$$\langle X, a_X\rangle \quad con \quad a_X \in X$$

y sucede para cada $X \in \mathfrak{B}$*. Hemos obtenido así una función* f *que a cada* $X \in \mathfrak{B}$ *le asigna* $f(X) = a_X$ ■

Importancia del Axioma de Elección

La importancia del axioma de elección es tal que la matemática sin él cambia radicalmente. Cito a continuación tres principios que equivalen o se siguen del de elección, para justificar lo que digo.

1. Equivale al principio del buen orden. $\forall A \exists R(R \ \textit{bien ordena} \ A)$.
   El axioma asegura que es posible definir un buen orden en todo conjunto, y como todo conjunto bien ordenado es isomorfo a un ordinal, será posible *"medir"* los conjuntos usando los ordinales.

2. Implica el *Lema de Zorn: "Todo conjunto ordenado en el que cada cadena posea cota superior, tiene elemento maximal".*

3. El Lema de Zorn implica el *Principio de Hausdorff: "En todo conjunto ordenado cada cadena puede extenderse a una cadena maximal"*

5.6.2. Axioma de Constructibilidad

Dentro de cualquier posible modelo de Zermelo-Fraenkel

$$\mathcal{U} = \langle \mathcal{V}, \in^{\mathcal{U}} \rangle$$

habría siempre un modelo que normalmente sería más pequeño, formado por su parte constructible. La parte constructible de un modelo de ZF es el menor submodelo de ZF que contiene los números ordinales y que continúa siendo modelo de ZF.

Cuando se definió la jerarquía de conjuntos de Zermelo Fraenkel usamos como noción básica la del conjunto de las partes, o potencia de un conjunto

$$\mathcal{V}_{\alpha+1} = \wp(\mathcal{V}_\alpha)$$

Sin embargo la noción de subconjunto es poco descriptiva y cuando nos hacemos preguntas tales como *"qué es un subconjunto arbitrario del conjunto de los números naturales"* y cuántos de ellos hay, no es posible responder sin determinar lo que se entiende por subconjunto. Podemos tomar como noción de conjunto la de una colección descriptible por una fórmula expresada en el lenguaje formal de la teoría de conjuntos. De esta manera obtendremos los conjuntos necesarios en matemáticas, excepto quizás los conjuntos *"no-constructibles"* provenientes del axioma de elección.

Por analogía con la jerarquía de Zermelo $\mathcal{V}_\alpha$, donde α es un ordinal, daremos una imagen intuitiva de la jerarquía constructible $\mathcal{L}_\alpha$. Empezamos como en $\mathcal{V}$ con el conjunto vacío y coincidiendo con los ordinales límite hacemos la recopilación de todo lo anterior tomando la gran unión de todos los niveles ya construidos. La única diferencia afecta al paso crucial $\alpha + 1$ mientras que en $\mathcal{V}_{\alpha+1}$ tomábamos a todos los subconjuntos de $\mathcal{V}_\alpha$, en $\mathcal{L}_{\alpha+1}$ tomamos a una selección de ellos: sólo los subconjuntos definibles de $\mathcal{L}_\alpha$. Así se puede redefinir la jerarquía de conjuntos sustituyendo la noción de *conjunto de las partes*

de un conjunto por la de *conjunto descriptible de las partes* de un conjunto. Indicaremos el nivel $\alpha \in Ord$ de la jerarquía por la nueva notación $\mathcal{L}_\alpha$.

Para 0, entonces

$$\mathcal{L}_0 = \emptyset$$

el primer nivel, como en la jerarquía de Zermelo, sólo tiene el vacío.

Para ordinales sucesores

$$\mathcal{L}_{\alpha+1} = PARAM.DEF(\mathcal{L}_\alpha, L^\epsilon)$$

donde $\mathcal{L}_\alpha = \langle \mathcal{L}_\alpha, \in^{\mathcal{U}} \cap (\mathcal{L}_\alpha)^2 \rangle)$ y se usa el lenguaje de la teoría de conjuntos $L^\in$. Así pues, tomamos todas las colecciones de elementos de $\mathcal{L}_\alpha$ que son definibles con fórmulas del lenguaje de la teoría de conjuntos, admitiéndose usar variables como parámetros.

Y cuando es un ordinal límite, λ

$$\mathcal{L}_\lambda = \bigcup_{\beta<\lambda} \mathcal{L}_\beta$$

Un conjunto $\mathbf{X}$ es *constructible* syss hay un α tal que $\mathbf{X} \in \mathcal{L}_\alpha$. La clase de todos los conjuntos constructibles se denomina *universo constructible* y es denotada por $\mathcal{L}$.

Esto es, la jerarquía constructible de todos los conjuntos es:

$$\mathcal{L} = \bigcup_{\alpha \in Ord} \mathcal{L}_a$$

Si comparamos a $\mathcal{L}$ con la jerarquía de Zermelo vemos que

$$\mathcal{L}_\alpha \subseteq \mathcal{V}_\alpha \text{ para todo } \alpha$$

De hecho, puesto que los conjuntos finitos pueden definirse con facilidad, $\mathcal{L}_n = \mathcal{V}_n$ y por lo tanto $\mathcal{L}_\omega = \mathcal{V}_\omega$.

Normalmente, sin embargo, $\mathcal{L}_{\omega+1} \neq \mathcal{V}_{\omega+1}$ —porque en $\wp(\omega)$ puede haber una cantidad supernumerable de conjuntos y sólo una cantidad numerable de ellos pueden definirse mediante fórmulas de L^ϵ—. También se ve fácilmente que $\mathcal{L}_\alpha \subseteq \mathcal{L}_\beta$ cuando $\alpha \leq \beta$. Pero si comparamos el ritmo de crecimiento de esta jerarquía con la de Zermelo, el de la constructible es bastante lento.

El *axioma de constructibilidad* equivale a la afirmación de que todo conjunto es constructible. Esto significa que sólo aceptamos como conjuntos a los de esta clase. De manera informal expresamos este axioma así:

Axioma 214 *(de Constructibilidad)* $\mathcal{L} = \mathcal{V}$.

Se demuestra que la teoría *ZF* más el axioma de Constructibilidad garantiza la definición de todos los niveles de la jerarquía constructible, pero no se sigue de *ZF*.

La ventaja de añadirlo es que en $ZF + (\mathcal{L} = \mathcal{V})$ podemos definir qué es un subconjunto y efectivamente construir la jerarquía, además en esta teoría el axioma de elección se deriva como teorema.

Teorema 215 *En la teoría* $ZF + (\mathcal{L} = \mathcal{V})$*, cada conjunto puede ser bien ordenado.*

5.7. Clases y Conjuntos en Gödel-Bernays-Neumann

La única diferencia que representa esta jerarquía respecto de la jerarquía de Zermelo es que *GBN* termina, ya que se añade un nivel superior en donde se aceptan como clases todas las colecciones de conjuntos de $\mathcal{V}$; incluso la clase $\Omega = Ord$ de los ordinales o la propia $\mathcal{V}$, formada por todos los conjuntos. Estas son lo que se denominan clases últimas en la presentación habitual.

En *GBN* tenemos un axioma de definición de clases sin restricciones. Intuitivamente el axioma dice que toda subcolección de $\mathcal{V}$ se añade a $\mathcal{V}$ para obtener $\mathcal{V}^+$. De esta forma

$$\mathcal{V}^+ = \mathcal{V} \cup \{\{x \in \mathcal{V} \ / \ \varphi\} / \ \varphi \in FORM(L^\epsilon)\}$$

Comentario 216 *La teoría usualmente tomada como básica es ZFC. Sin embargo es posible obtener muchos resultados matemáticos sólo con ZF. Por otro lado, la teoría de conjuntos constructible puede parecer más natural y algunos matemáticos la adoptan, aunque la jerarquía de conjuntos más aceptada es* $\mathcal{V}$ *y el axioma de constructibidad es muy discutido, como lo sigue siendo el de elección en muchos autores.*

Bibliografía

[1] Alonso, J y otros. [1998]. *Curso prático de Teoría de Conjuntos.* Ediciones la Ñ. Sevilla.

[2] Barwise, J. [1977]. *Handbook of Mathematical Logic.* North Holland Publishing Company. Amsterdam. Holanda.

[3] Devlin, K. (1973) *Aspects of constructibility*. Springer-Verlag. New York.

[4] Devlin, K. (1993) *The Joy of Sets*. Springer-Verlag. New York.

[5] Enderton, H. (1977) *Elements of Set Theory*. Academic Press. New York.

[6] Halmos, P. (1974). *Naive Set Theory*. Springer-Verlag. New York.

[7] Heijenoort, J. ed. [1967]. *From Frege to Gödel.* Harvard University Press. Harvard. USA

[8] Huertas, A y Manzano, M. (2002). *Teoría de Conjuntos.* http://logicae.usal.es

[9] Jech, T. [1985]. *About the Axiom of Choice.* en [2]

[10] Quine, W. [1969]. *Set Theory and its Logic.* Harvard University Press. Cambridge Massachusetts. USA.

[11] Rubin, H y Rubin, J. [1963]. *Equivalents of the Axiom of Choice.* North Holland Publishing Company. Amsterdam. Holanda.

[12] Suppes, P. (1972) *Axiomatic Set Theory*. Dover. New York.

Parte II

Sistemas Lógicos

Capítulo 6

Lógica de cláusulas de Horn

6.1. Introducción

La lógica de las cláusulas de Horn es una parte de la lógica de primer orden. Fue definida por McKinsey [5] en 1943, cuando investigaba problemas de decidibilidad. Entre 1956 y 1970 Mal'tsev demostró que constituye el formalismo ideal para el álgebra universal. En 1970 se demostró que la teoría de la prueba de la lógica de las cláusulas de Horn era especialmente potente. En 1979 Kowalski [4] hizo notar que las sentencias de una teoría de cláusulas de Horn tienen una lectura bastante natural como instrucciones de un computador. Siguiendo dichas instrucciones el computador encuentra las pruebas de los teoremas de cualquier teoría así expresada.

Esta idea se ha usado con enorme éxito en programación lógica, tanto en *Prolog* como en *Datalog*.

6.2. Definición de fórmulas de Horn

El lenguaje es el de primer orden y la formación de fórmulas es la misma.

Definición 217 *Se llaman* ***hechos*** *a las fórmulas atómicas. Si son distintos de* $\perp$ *se llaman estrictas.*

Definición 218 *Se llaman* ***reglas*** *a las fórmulas de la forma*

$$\alpha_1 \wedge ... \wedge \alpha_n \rightarrow \beta$$

donde todas las fórmulas $\alpha_1, ..., \alpha_n$ *y* β *son atómicas (a* β *se le llama la cabeza y a las otras el cuerpo). Se puede decir que un hecho es un caso extremo de regla; cuando su cuerpo es* $\varnothing$.

Definición 219 *El conjunto de las* ***fórmulas básicas de Horn*** *es la unión de los anteriores.*

Definición 220 *Una fórmula básica de Horn es* ***estricta*** *si no contiene el signo para lo falso,* $\perp$

La fórmula

$$\alpha_1 \wedge ... \wedge \alpha_n \rightarrow \perp$$

no es estricta, equivale a

$$\neg\alpha_1 \vee ... \vee \neg\alpha_n$$

Una regla de Horn estricta equivale a

$$\neg\alpha_1 \vee ... \vee \neg\alpha_n \vee \beta$$

donde todas las fórmulas $\alpha_1, ..., \alpha_n$, y β son estrictas.

Definición 221 *Decimos que* $\gamma_1 \wedge ... \wedge \gamma_n$ *es una* ***fórmula de Horn sin cuantificadores****, si cada* γ_i *es una fórmula básica de Horn.*

Definición 222 *Decimos que* $Q\alpha$ *es una* ***fórmula de Horn****, si* Q *es una cadena de cuantificadores y* α *es una fórmula de Horn sin cuantificadores. Decimos que es* ***estricta*** *si no usa* $\perp$ *y que es una sentencia, si no contiene variables libres.*

Definición 223 *Una* ***teoría de Horn*** *es un conjunto de sentencias de Horn*

Ejemplo 224 *Para expresar "cada cadena de caracteres tiene una longitud, que es un número natural" podemos usar esta fórmula*

$$\forall x(cadena(x) \rightarrow \exists y(longitud(xy) \wedge n^o natural(y)))$$

que, aunque no es una sentencia de Horn, se reescribe así:

$$\forall x \exists y((cadena(x) \rightarrow longitud(xy)) \wedge (cadena(x) \rightarrow n^o natural(y)))$$

esta última es una sentencia de Horn estricta

Definición 225 *Una* ***sentencia de Horn universal*** *es una sentencia de Horn que sólo contiene cuantificadores universales.*

Puesto que la fórmula

$$\forall x(\alpha \wedge \beta)$$

es equivalente a

$$\forall x\alpha \wedge \forall x\beta$$

cada sentencia de Horn universal puede escribirse como conjunción de sentencias de la forma

$$\forall x_1 ... x_n \alpha$$

siendo α una fórmula básica

Definición 226 *Se llaman* ***cláusulas de Horn*** *a las fórmulas*

$$\forall x_1 ... x_n \alpha$$

Dependiendo de si la fórmula α *es un hecho o una regla, la cláusula también*

Definición 227 *Una* ***teoría de cláusulas de Horn*** *es un conjunto de sentencias que tienen esta forma*

$$\forall x_1 ... x_k \beta(x_V ... x_k)$$

$$\forall x_1 ... x_k (a_1(x_1 ... x_k) \wedge ... \wedge a_n(x_1 ... x_k) \rightarrow \beta(x_1 ... x_k))$$

6.3. Importancia de estas fórmulas

¿Hay alguna razón que explique porqué la lógica de las cláusulas de Horn es tan útil y sus propiedades tan interesantes?

Hodges responde a esta pregunta así:

Cláusulas de Horn e inducción

Las cláusulas de Horn permiten construir conjuntos inductivamente

Ejemplo 228 *Esta es una teoría de cláusulas de Horn*

$$par(0)$$

$$\forall x(par(x) \rightarrow impar(siguiente(x)))$$

$$\forall x(impar(x) \rightarrow par(siguiente(x)))$$

Esta teoría sirve para formar inductivamente dos conjuntos: el de los pares y el de los impares. Este ejemplo es típico: los hechos de Horn nos dicen qué elementos debemos poner en el conjunto al empezar, las reglas de Horn nos dicen que otros elementos debemos ir añadiendo, en función de los que ya tengamos en ese momento.

Ejemplo 229 *Imaginad una base de datos que almacena la información de la red de autobuses: cómo llegar de una ciudad española a otra utilizando este transporte*

$$\forall x \ enlazada(x, x)$$

$$enlace \ directo(Salamanca, Madrid)$$

$$enlace \ directo(Barcelona, Salamanca)$$

$$\vdots$$

$$\forall xy \ (enlazada(x, y) \rightarrow enlazada(y, x))$$

$$\forall xyz \ (enlazada(x, y) \wedge enlace \ directo(y, z) \rightarrow enlazada(x, z))$$

De esta forma conseguimos los pares de ciudades que están enlazadas mediante autobús, tanto si es directo, como con transbordo.

Subteoría de hechos atómicos

Dada una teoría T cualquiera de cláusulas de Horn, existe un único conjunto S (es el menor de los que existen) formado exclusivamente por hechos atómicos, que deben ser verdaderos para que lo sean las fórmulas de T. Los hechos atómicos de S son descripciones de los conjuntos y relaciones definidos mediante T. En el ejemplo de los autobuses, el conjunto S contiene estas sentencias

$$enlace\ directo(Salamanca, Ávila)$$

$$enlace\ directo(Zamora, Salamanca)$$

$$\vdots$$

En el ejemplo de los pares y los impares el conjunto es infinito.

Esta propiedad de poseer una subteoría atómica con tan buenas prestaciones no la tiene cualquier teoría lógica; basta pensar en cualquiera que contenga esta fórmula

$$p \vee q$$

Elementos de S

El conjunto S (que define el modelo inicial de la teoría T) está formado por las fórmulas atómicas que se deducen a partir de T.

Debido a esta condición y a la mencionada en el apartado anterior, las diferencias entre verdad y demostrabilidad para cláusulas de Horn no es tan clara como en el resto de la lógica de primer orden. Ello permite una simplificación de la metateoría y abre ciertos debates tales como si una *Base de Tipos* es una teoría (un conjunto de sentencias) o un álgebra (un modelo)

Procedimiento de prueba

Se puede demostrar que una sentencia dada está en S siguiendo el procedimiento mediante el cual se definen los conjuntos en la teoría T.

6.4. Niveles de estudio

El conjunto S puede estudiarse a tres niveles diferentes. En el inferior, se analiza su construcción; esto nos lleva a la semántica del punto fijo del Prolog (Emden-Kowalsky). En el segundo se observan los itinerarios de prueba; nos conduce a la definición de un cálculo específico para cláusulas de Horn, directamente emparentado con la metateoría. Finalmente, se describe la colección completa de todas las pruebas y se examina el modo de encontrar entre ellas una concreta, la de un *hecho* preciso. Al traducir estas pruebas *simples* en pruebas mediante resolución nos encontramos cómodamente instalados en la programación lógica.

6.5. Usos destacados

Como se sugirió en el apartado 6.3, las cláusulas de Horn aparecen cuando se selecciona a los elementos de un conjunto mediante la aplicación de reglas. Por ejemplo, el conjunto de las oraciones de una gramática (de una lengua natural cualquiera, pero que posea estructura de frases) se obtiene al aplicar las reglas a un *léxico básico.* Se puede escribir el léxico básico como un conjunto de *hechos* y las reglas como *cláusulas* de Horn. (No es una casualidad el hecho de que el creador del Prolog, Alain Colmenauer, fuera un investigador de lingüística computacional.)

El resultado apunta a otro mucho más general: todo conjunto recursivamente enumerable puede ser definido mediante una teoría de cláusulas de Horn.

Puesto que el conjunto de los teoremas lógicos de la lógica de primer orden es recursivamente enumerable, la lógica de cláusulas de Horn puede servir para simular toda la lógica de primer orden. Siendo esto así resulta que el problema de la decisión para la lógica de primer orden y para su subteoría de cláusulas de Horn es de la misma *complejidad computacional.*

Sin embargo, hay ciertos conjuntos de sentencias de Horn que, desde el punto de vista de la teoría de la decidibilidad, se comportan mucho mejor que sus sentencias asociadas de primer orden.

La lógica ecuacional es un caso particular de cláusulas: las identidades son *hechos* de Horn y los axiomas de sustitución son *leyes de Horn.* Justamente aquí es donde se establece el vínculo entre el álgebra universal y nuestras cláusulas. Las reglas de reescritura son las que mayormente se utilizan en la lógica ecuacional; en nuestra lógica se añaden a éstas las proposicionales.

6.6. Conversión

En el ejercicio 224 vimos que aún cuando las sentencias de primer orden no sean de Horn, algunas de sus equivalentes podrían serlo.

¿Cómo sabemos cuando se puede reescribir una sentencia de primer orden dada en una de Horn y cuando no?

A veces la conversión no es tan inmediata, requiriendo un cambio de lenguaje.

Ejemplo 230 *Aunque esta fórmula no es de Horn*

$$\forall x(p(x) \to \exists y(y = g(x) \wedge r(y)) \vee q(x))$$

es lógicamente equivalente a esta otra

$$\forall x \exists y(((y \neq g(x) \wedge \neg q(x)) \to \neg p(x)) \wedge ((\neg r(y) \wedge \neg q(x)) \to \neg p(x)))$$

que aunque tampoco lo es, se convierte al cambiar $\neg p, \neg q$ *y* $\neg r$ *por* $\widetilde{p}, \widetilde{q}$ *y* $\widetilde{r}$ *e introducir* $D(x, y)$ *por* $x \neq y$*. El resultado es la fórmula*

$$\forall x \exists y(((D(y, g(x)) \wedge \widetilde{q}(x)) \to \widetilde{p}(x)) \wedge ((\widetilde{r}(y) \wedge \widetilde{q}(x)) \to \widetilde{p}(x)))$$

A este procedimiento se le denomina *renombrado* y a la función correspondiente *función de renombrado.*

No siempre se puede convertir un conjunto de sentencias en sentencias de Horn mediante este procedimiento.

Ejemplo 231 *Este conjunto* $\left\{ \begin{array}{c} \forall x(p(x) \vee q(x) \vee r(x)), \\ \forall x(p(x) \vee \neg q(x) \vee \neg r(x)), \\ \forall x(\neg p(x) \vee q(x) \vee \neg r(x)), \\ \forall x(\neg p(x) \vee \neg q(x) \vee r(x)) \end{array} \right\}$ *no es convertible.*

La respuesta a la pregunta planteada sobre si hay un procedimiento general para determinar convertibilidad nos la proporciona el siguiente teorema.

Teorema 232 *(Lewis 1978). Sea* T *un conjunto de cláusulas* $\gamma_V, ..., \gamma_m$ *donde cada* γ_i *es de la forma*

$$\forall \overline{x}(\alpha_{V,h} \vee ... \vee \alpha_{k_h,h})$$

Sea T^{Lewis} *la teoría formada por todas las sentencias de la forma*

$$\forall \overline{x}(\alpha_{i,h} \vee ... \vee \alpha_{j,h})$$

donde $V \leq h \leq m$ *y* $V \leq i \leq j \leq k_h$. *Las siguientes condiciones son equivalentes:*

1. *Existe una función de renombrado que convierte* T *en un conjunto de cláusulas de Horn*
2. *La Teoría* T^{Lewis} *es consistente.*

6.7. El cálculo simple de Horn

Por supuesto, cualquier cálculo de primer orden serviría, pero éste refleja mejor el comportamiento de las cláusulas de Horn.

Definición 233 *Una* T***-inferencia*** *es un diagrama de sentencias de la forma*

$$\frac{\alpha_1(\overline{\mu})...\alpha_n(\overline{\mu})}{\beta(\overline{\mu})}$$

siendo

$$\forall \overline{x}(\alpha_1(\overline{x}) \wedge ... \wedge \alpha_n(\overline{x}) \rightarrow \beta(\overline{x}))$$

una sentencia de T *y* $\overline{\mu}$ *una secuencia de términos cerrados. Cuando* $n = 0$ *tenemos*

$$\overline{\beta(\overline{\mu})}$$

siendo

$$\forall \overline{x}\beta(\overline{x})$$

un hecho.

Definición 234 *Una* $T-$***prueba*** *de una sentencia atómica* γ *a partir de una teoría de cláusulas de Horn* T *se define como un árbol formado por* $T-$*inferencias que cumple:*

1. *Las* $T-$*inferencias de inicio no tienen premisas (provienen de hechos)*
2. γ *es la sentencia final*
3. $\bot$ *no ocurre jamás en el árbol, excepto, tal vez, al final.*

Ejemplo 235 *Sea* $T = \left\{ \begin{array}{c} \forall x p(x) \\ \forall x(p(x) \rightarrow q(x)) \\ \forall x(p(x) \wedge q(x) \rightarrow r(x)) \\ \forall x(r(f(x)) \rightarrow \bot) \\ r(c) \end{array} \right\}$ *la deducción es así:*

$$\dfrac{\dfrac{\dfrac{}{p(f(c))} \qquad \dfrac{\dfrac{}{p(f(c))}}{q(f(c))}}{r(f(c))}}{\bot}$$

Notación 236 *En el caso anterior decimos que* $\bot$ *es deducible de* T *en el cálculo simple de Horn y usamos esta notación*

$$T \vdash_{simple} \bot$$

6.8. Modelo inicial

Cuando una teoría de cláusulas de Horn es sintácticamente consistente, tiene modelos muy simples; en particular tiene un modelo inicial, $\Im$

Definición 237 *Sea* T *una teoría de cláusulas de Horn y* $\Im$ *un modelo. Decimos que* $\Im$ *es un* **modelo inicial** *de* T *si se verifica lo siguiente:*

1. *Cada elemento del universo de* $\Im$ *tiene esta forma* $\mu^{\Im}$ *(es la denotación de un término cerrado)*
2. *Para cada sentencia atómica se cumple:*

$$\Im \vDash \alpha \quad syss \quad T \vdash_{simple} \alpha$$

3. $\Im$ *es un modelo de* T

Usos de los modelos iniciales

Una de las aplicaciones de los modelos iniciales es de índole psicológica. En deducción automática es importante subrayar la relación lógica entre fórmulas. De manera natural tratamos de usar el significado de la fórmula. Es difícil comprender cómo se puede usar el significado de una fórmula sin dar un modelo de ella; una interpretación semántica. Éste es justamente el problema: lo normal es que las relaciones lógicas entre fórmulas no sean privativas de un modelo, de una interpretación particular, sino de todas en conjunto. $\Gamma \models \alpha$ significa que todas las interpretaciones que son modelo de Γ lo son también de α. Toda teoría de cláusulas de Horn sintácticamente consistente tiene la siguiente propiedad: ser deducible a partir de Γ equivale a ser verdadera en el modelo inicial: Por lo tanto podemos asociar la deducibilidad a partir de Γ con la verdad en su modelo inicial.

Con todos los reparos hacia el *psicologicismo* es interesante señalar las tesis de Johnson-Laird y Byrne; según ellos cuando las personas normales deducen algo lo que hacen es representarse modelos esquemáticos de las premisas y examinarlos. La dificultad de una deducción (medida en el tiempo que se tarda en hacerla o en la cantidad de gente que se equivoca) está directamente relacionada con la cantidad de modelos que deben ser examinados: Si ello es cierto, deducir sentencias atómicas a partir de cláusulas de Horn es especialmente sencillo ya que basta con examinar un modelo.

Estos autores analizan veintisiete silogismos y los ordenan por su dificultad (estadística de los que se equivocan al hacerlo). De entre ellos destacan seis, que son básicamente razonamientos con cláusulas de Horn, y que se encuentran entre los nueve más fáciles de todo el bloque.

6.9. Codificación de la lógica de primer orden

¿Cual es la relación existente entre la lógica de primer orden y la de las cláusulas de Horn?

Hemos visto que no todas las teorías son de Horn, ni convertibles en ellas mediante alguna función de renombrado (teorema 232). Sin embargo, puesto que todo conjunto recursivamente enumerable puede ser definido mediante una teoría de cláusulas de Horn, parece que hay esperanzas de poder usar esta teoría incluso para determinar validez. De hecho, hay dos casos en los que la suerte nos acompaña:

1. La pregunta de si una sentencia cualquiera de primer orden es lógicamente válida puede convertirse en la pregunta sobre la inconsistencia de una cláusula de Horn asociada a ella.

2. Para cada teoría de primer orden existe una teoría de cláusulas de Horn que tiene los mismos modelos (o "casi")

Bibliografía

[1] Gabbay, D., Hogger, C.J. y Robinson, J. A. eds [1996]. Handbook of Logic in Artificial Intelligence and Logic Programming. vol 1. Oxford University Press, Oxford.

[2] Hodges, W. [1996] *"Logical features of Horn clauses".* En [1].

[3] Horn, A. [1951]. *"On sentences which are true of direct unions of algebras".* Journal of Symbolic Logic, 16, 14-21.

[4] Kowalsky, R. [1979]. *Logic for problem solving.* North Holland. Amsterdam.

[5] McKenzie, J. C. C. [1943]. *"The decision problem for some classes of sentences without quantifiers".* Journal of Symbolic Logic, 8, 61-76.

[6] Smullyan, R. [1956] *"On definability by recursion".* Bulletin of the American Mathematical Society, pp 601.

[7] Dellunde, P. [1996]. *Contributions to the model theory of equality-free logic.* Tesis doctoral, Barcelona.

Capítulo 7

Lógica Multivariada

7.1. Introducción

7.1.1. Ejemplos

En muchas de las ramas de la matemática, de la filosofía, de la I.A. y de la informática formalizamos enunciados relativos a diversos tipos de objetos. Por consiguiente, tanto los lenguajes lógicos utilizados, como las estructuras matemáticas que los interpretan son multivariadas o heterogéneas; esto es, el conjunto de las variables del lenguaje toma valores sobre diversos universos o dominios.

Son numerosos los ejemplos de materias que utilizan fórmulas y estructuras multivariadas:

1. En *geometría*, por tomar un ejemplo clásico y sencillo, usamos distintos universos para *puntos*, *líneas*, *ángulos*, *triángulos*, etc

2. En la teoría de *espacios vectoriales* tenemos universos distintos para *vectores* y *escalares*. Además de eso, podemos incluir universos para *subespacios*, *métricas* y *aplicaciones lineales*.

3. En *teoría de grupos* las estructuras poseen distintos universos para *elementos* del grupo, *subgrupos*, *subgrupos normales*, *homomorfismos*, etc.

4. En la lógica de segundo orden *SOL* veremos que hay universos para *individuos*, para *conjuntos* de esos elementos básicos, para *relaciones* binarias entre ellos, etc.

5. En *teoría de tipos* la jerarquía corresponde a toda la del *universo matemático finito* que contiene en sus distintos niveles a: *individuos*, *conjuntos de individuos*, *conjuntos de conjuntos de individuos*, etc.

6. En *computación* utilizamos invariablemente estructuras multivariadas: lo típico es tener universos de *datos*, *números naturales* y *operadores boolea-*

nos. Podemos añadir otros para *números reales*, *cadenas de caracteres*, *matrices*, etc.

7. Cuando razonamos sobre programas los situamos en universos para ellos y añadimos otros para *estados* y para *tiempo*.

¿Qué lenguaje y qué lógica es el adecuado en cada uno de estos casos? La respuesta es que la lógica multivariada es la que mejor les cuadra.

7.1.2. Comparación con la lógica de primer orden sin variedades

La lógica de primer orden *FOL* es sólida y equilibrada, con muchas propiedades interesantes: tiene un cálculo deductivo correcto y completo, es compacta y disfruta de las propiedades de Löwenheim-Skolen, entre otras.

Su lenguaje y sus modelos, tal y como se suelen presentar en la mayoría de los libros de texto, son *univariados*: contienen sólo un universo de objetos y el lenguaje formal sólo usa variables individuales para referirse y cuantificar sobre ellos. Por consiguiente podemos intentar, y así se ha hecho con frecuencia, reducir y codificar nuestras estructuras y lenguajes multivariados en estructuras y lenguajes univariados. Ello es posible y es de hecho lo que propuso en 1952 Hao Wang [18], uno de los primeros estudiosos de esta lógica.

Así que la reducción de la lógica multivariada *MSL* a la univariada *FOL* es un resultado no sólo bien conocido desde antiguo, sino también el planteamiento que normalmente se hace en los libros de texto. El proceso se lleva a cabo a dos niveles: hay una *traducción sintáctica* de las fórmulas multivariadas a las univariadas —conocida como *relativización de cuantificadores*— y una *conversión semántica* de estructuras —conocida como *unificación de dominios*—. Lo que nunca se suele decir en los libros de texto es el precio que debe pagarse, algo sobre lo que hablaré después.

Para traducir tomamos un lenguaje de primer orden sin variedades —esto es, con una sola clase de variables— con los mismos signos de operación que tuviéramos en la multivariada y le añadimos tantos relatores monarios como variedades hubiera. Cada fórmula cuantificada sobre una variedad i

$$\forall x^i \varphi(x^i)$$

será reemplazada por una fórmula cuantificada condicional, en cuyo antecedente decimos sobre qué variedad se restringe la cuantificación

$$\forall x(Q^i x \to \varphi(x)^*)$$

La nueva estructura univariada obtenida mediante unificación de dominios tendrá un solo universo constituido por la unión de todos los universos de la que se reduce, las relaciones de la estructura multivariada pasan a serlo de la nueva univariada y las funciones de la multivariada se extienden para que puedan serlo de la univariada, añadiendo valores arbitrarios para los nuevos elementos.

Voy a comentar brevemente a qué precio pagamos esta reducción.

1. *Naturalidad*
   Las estructuras que queremos estudiar son multivariadas y el lenguaje más adecuado para llevar a término la investigación debería reflejar esa diversidad. El principio básico de *FOL* es por lo tanto inadecuado y así *perdemos naturalidad* cuando forzamos la conversión.

2. *Interpolación de Craig*
   En 1967 Solomon Feferman [6], el primer lógico que desarrolló la lógica multivariada en sí misma[1], señaló que por lo que respecta al *teorema de interpolación de Craig* el de la multivariada es mejor porque en este caso se demuestra una versión mejorada. Sin embargo, al realizar la reducción aunque se conservan las demostraciones de otros teoremas como los de compacidad y Löwenhein-Skolem, pudiéndose *"arrastrar"* el resultado, sin necesidad de repetir la prueba, no sucede lo mismo con el de Craig. El teorema puede demostrarse desde la teoría de modelos y desde la teoría de la prueba, tanto en el caso de la *FOL* como de la *MSL*. Puesto que una de las posibilidades es derivarlo del teorema de completud del cálculo Gentzen sin regla de corte, cuando falla interpolación la esperanza de encontrar un cálculo de esta clase se reduce. De esta manera el teorema de Craig nos sirve de test para evaluar las bondades de una lógica desde el punto de vista de su teoría de la prueba. Por lo ya comentado en este mismo apartado es de suponer que el cálculo de *MSL* ofrezca buenas prestaciones, mejores incluso que el correspondiente de *FOL*.
   En opinión de Ebbinghaus:
   It is especially with interpolation that many-sortedness pays. As seen in Feferman [1974], the many-sorted version of the interpolation theorem together with its possible refinements is a powerful tool even for one-sorted model theory, offering for instance elegant proofs of various preservation theorems.
   También piensa:
   Interpolation properties seem to indicate some kind of balance between syntax and semantics. This can be seen, for instance, from the work of Zucker [1978] or from the fact that interpolation implies Beth's definability theorem, according to which implicit definitions can be made explicit. Hence we may expect that interpolation properties fail if syntax and semantics are not in an equilibrium.

3. *Interpretabilidad*
   Hook [9] demuestra que una teoría multivariada puede ser interpretada en otra teoría multivariada sin que las correspondientes teorías de primer orden lo sean entre sí. Señala:
   A theory can be proved consistent by exhibiting an interpretation in a known consistent theory. A many-sorted theory, therefore, may be useful in a consistency proof for which the corresponding one-sorted theory would

[1] Sin aplicar el *"procedimiento matemático"* del chiste, causa de tantas simplificaciones y empobrecimientos; esto es, la *"reducción al caso anterior"*.

not suffice. (Even if another consistency proof is known, the proof using interpretations has the advantage of being finitary and purely syntactic.)

4. *Eficiencia deductiva*
   Las deducciones en el cálculo multivariado son más cortas que las correspondientes deducciones en el de primer orden obtenido mediante reducción. Se evitan conclusiones inútiles, tales como los teoremas de primer orden que no tienen contrapartida multivariada porque no son traducción de ninguna de estas fórmulas. La razón por la que aparecen fórmulas nuevas es que al hacer la traducción añadimos al lenguaje tantos relatores monarios como variedades. Por consiguiente, desde el punto de vista de la *deducción automática de teoremas*, cuyo objetivo principal es obtener conclusiones lógicas con eficacia y rapidez, evitando resultados indeseados, la reducción a la univariada es inaceptable.

Es evidente que aunque una se reduce a la otra tienen las propiedades diferentes ya señaladas; su naturalidad, la fuerza del teorema de interpolación, la eficiencia del cálculo nos inclinan a la multivariada. Además, la interpretabilidad entre teorías no siempre se preserva al pasar a la univariada. Por otra parte es verdad, aunque sea obvio, que la lógica univariada está contenida en la multivariada.

Las dos tienen un cálculo deductivo completo en sentido fuerte y se aplican los resultados de compacidad y Löwemheim-Skolem.

Por lo que respecta a su teoría de modelos, nociones tales como subsistema, imagen homomórfica, producto directo y reducido pueden también definirse para la multivariada y muchos de los teoremas se generalizan sin dificultad. La equivalencia elemental se preserva en la reducción; esto es, dos estructuras multivariadas que son elementalmente equivalente con el lenguaje multivariado siguen siéndolo cuando, mediante relativización de cuantificadores y unificación de dominios, pasamos al lenguaje y estructuras univariadas.

¿Es la lógica multivariada una extensión en el sentido propio y estricto de la de primer orden?

En 1969 Lindström descubrió que la lógica de primer orden es la más potente que satisface simultáneamente compacidad y Löwenheim-Skolem. También demostró que es así mismo la más potente que teniendo una sintaxis finita, retiene las metapropiedades de completud y Löwenheim-Skolem. Desde este punto de vista *MSL* no puede ser considerada una extensión propia de *FOL*.

Tal vez sea conveniente hacer una precisión terminológica. Hay varios sistemas lógicos cuyas estructuras, como las de la multivariada poseen varios universos: la lógica de segundo orden, la de tercer orden, la teoría de tipos; pero su semántica estándar es muy específica y difiere notablemente de la multivariada. Se cuelan en ella nociones nada inocentes de la teoría de conjuntos, tales como la ya mencionada de *"subconjunto"* y esto hace que la capacidad expresiva se dispare y la deductiva caiga bajo mínimos. Se trata aquí de extensiones de *FOL* en sentido estricto y son incompletas, no compactas y no poseen la propiedad de Löwenheim-Skolem.

Por otra parte, hay lógicas a las que etiquetamos como *extensiones* pero que lo son sólo parcialmente ya que al poseer simultáneamente todas las propiedades mencionadas, no lo son en el sentido del teorema de Lindström. Lo cierto es que en sentido estricto, tampoco cae en esta categoría la lógica multivariada; la cuantificación cambia, pero no se extiende. En primer orden cuantificamos sobre los elementos de un determinado dominio, mientras que en multivariada lo hacemos sobre los universos obtenidos mediante catalogación o incluso estratificación del mismo.

7.1.3. Usos de la lógica multivariada.

Aunque la lógica multivariada no sea una extensión en sentido propio de la de primer orden (univariada), atendiendo al resto de propiedades se la prefiere en aplicaciones tanto en informática como en lingüística. Por lo que respecta a la informática es ampliamente usada, sirvan los ejemplos siguientes:

1. Tipos abstractos de datos
2. Semánticas y lógicas de verificación de programas
3. Definición de lenguajes de programación
4. Álgebras para distintas lógicas
5. Bases de datos
6. Lógica dinámica
7. Semántica de lenguajes naturales
8. Solución de problemas computarizada
9. Representación del conocimiento
10. Programación lógica y deducción automática

Márkusz [15] hace un repaso detallado del uso de la lógica multivariada en informática, especialmente por el grupo de Hungría; el libro que editan Meinke y Tucker en el 1993, *Many-sorted logic and its applications* reúne una serie de trabajos en esa dirección.

La lógica multivariada se ha usado en matemáticas como procedimiento para buscar modelos no estándar; esto es esencialmente lo que se hace para conseguir los modelos generales de Henkin para la lógica superior, responsables de su teorema de completud.

Puede resultar también útil para entender la lógica dinámica de dos formas distintas:

1. la lógica dinámica proposicional se traduce a la lógica multivariada por un procedimiento claramente deudor al de Henkin
   y está también

2. la lógica dinámica no-estándar, heredera igualmente de los modelos generales de Henkin.

De la reducción de otras lógicas a *MSL* hablaré con detalle en el capítulo 13.

7.2. Lenguaje y estructuras

Nuestro primer objetivo es clasificar a las estructuras matemáticas mediante su signatura de manera que dos estructuras tengan la misma signatura siempre que y sólo cuando se pueda usar el mismo lenguaje para hablar de ellas. Antes de definir con precisión los conceptos de signatura y estructura, digamos informalmente qué son. Aquí las estructuras son concebidas como multivariadas, de manera que contienen más de un universo o dominio de objetos sobre los que toman valores los conjunto de variables. Usamos un conjunto de índices para distinguir los universos: para cada $i \in SORT$, $\mathbf{A}_i$ es el universo de variedad i. En particular, exijimos que $0 \in SORT$ y hacemos que $\mathbf{A}_0 = \{V, F\}$, $\mathbf{A}_0$ es el universo de valores de verdad —variedad booleana—. De hecho, esta presentación nos permitiría tener más de un valor de verdad; por ejemplo, tres $\mathbf{A}_0 = \{T, F, U\}$, añadiendo un valor indefinido[2].

Una estructura tiene operaciones de distintos tipos: para $i_0, ..., i_n \in SORT$, una operación $n-$aria de tipo $\alpha = \langle i_0, ..., i_n \rangle$ es una función cuyo dominio es el producto cartesiano de universos $\mathbf{A}_{i_V} \times ... \times \mathbf{A}_{i_n}$ en $\mathbf{A}_{i_0}$. Cuando $\mathbf{A}_{i_0}$ es $\mathbf{A}_0$, decimos que es una relación $n-$aria de tipo $\langle 0, i_V, ..., i_n \rangle$. Incluimos aquí las funciones veritativas e identificamos las constantes de la estructura con funciones de $\{\emptyset\}$ en $\mathbf{A}_i$; esto es, de tipo $\langle i \rangle$, que simplificamos como i.

También podemos tener relaciones $n-$arias sin tipo: funciones de

$$(\bigcup_{i \in SORT - \{0\}} \mathbf{A}_i)^n$$

en $\mathbf{A}_0$. La identidad es una de ellas. Si no tuviéramos esta modalidad de relaciones tendríamos una identidad fraccionada por tipos[3].

7.2.1. Signatura

Una signatura es un par ordenado

$$\Sigma = \langle SORT, FUNC \rangle$$

tal que

1. $SORT$ es un conjunto de índices, con $0 \in SORT$

[2] En Huertas ([10] y [11]) se usa el tercer valor para definir una lógica multivariada y parcial que sirve de lógica subyacente al hacer la traducción de la modal de primer orden con la semántica de *huecos de valor de verdad*.

[3] De la identidad hablaremos en la sección 11.4.

2. $FUNC : OPER.SYM \longrightarrow S_\omega(SORT) \cup \omega^+$ es una función cuyos valores son números naturales distintos de cero, o están en el conjunto formado por las sucesiones finitas de elementos de $SORT$. Por ejemplo:

   a) Las conectivas toman tienen estos valores:

   $$FUNC(\neg) = \langle 0, 0 \rangle \,,\ FUNC(\wedge) = \langle 0, 0, 0 \rangle$$

   b) La identidad es binaria, sin tipos:

   $$FUNC(E) = 2$$

   c) Hay relaciones $n-$arias establecidas en la unión de tipos:

   $$FUNC(R) = n$$

   d) El resto tienen tipo:

   $$FUNC(f) = \langle i_0, ..., i_n \rangle$$

   distinguiéndose los predicados —$i_0 = 0$—
   de las funciones —$i_0 \neq 0$—

Estructuras

Una estructura es un par ordenado

$$\mathcal{A} = \left\langle \langle \mathbf{A}_i \rangle_{i \in SORT} \,, \langle f^{\mathcal{A}} \rangle_{f \in OPER.SYM} \right\rangle$$

donde:

1. $\mathbf{A}_i$ es el universo de variedad i, que ha de ser no vacío.
   Tomamos $\mathbf{A}_0 = \{V, F\}$
2. Cuando $FUNC(f) = \langle i_0, i_V, ..., i_n \rangle$, entonces

   $$f^{\mathcal{A}} : \mathbf{A}_{i_V} \times ... \times \mathbf{A}_{i_n} \longrightarrow \mathbf{A}_{i_0}$$

   Cuando $FUNC(f) = n$

   $$f^{\mathcal{A}} : \left(\bigcup_{i \in SORT - \{0\}} \mathbf{A}_i \right) \longrightarrow \mathbf{A}_0$$

   Algunos signos de $OPER.SYM$ tienen una interpretación fijada de antemano; por ejemplo, queremos que los conectores tengan la interpretación usual y que la igualdad denote identidad.

$$\neg^{\mathcal{A}}: \begin{array}{ccc} \mathbf{A}_0 & \longrightarrow & \mathbf{A}_0 \\ V & \longmapsto & F \\ F & \longmapsto & V \end{array} \qquad \vee^{\mathcal{A}}: \begin{array}{ccc} \mathbf{A}_0 \times \mathbf{A}_0 & \longrightarrow & \mathbf{A}_0 \\ \langle V, V\rangle & \longmapsto & V \\ \langle V, F\rangle & \longmapsto & V \\ \langle F, V\rangle & \longmapsto & V \\ \langle F, F\rangle & \longmapsto & F \end{array}$$

$E^{\mathcal{A}}(\mathbf{x}, \mathbf{y}) = V$ syss $\mathbf{x} = \mathbf{y}$

Comentario 238 *Cuando* $FUNC(f) = \langle i\rangle$ *entonces* $f^{\mathcal{A}} \in \mathbf{A}_i$. *Y cuando* $FUNC(f) = \langle 0, i_V, ..., i_n\rangle$ *entonces* $f^{\mathcal{A}}$ *es una función característica que se identifica con*

$$\left\{\bar{\mathbf{x}} \in \mathbf{A}_{i_V} \times ... \times \mathbf{A}_{i_n} \mid f^{\mathcal{A}}(\bar{\mathbf{x}}) = V\right\}$$

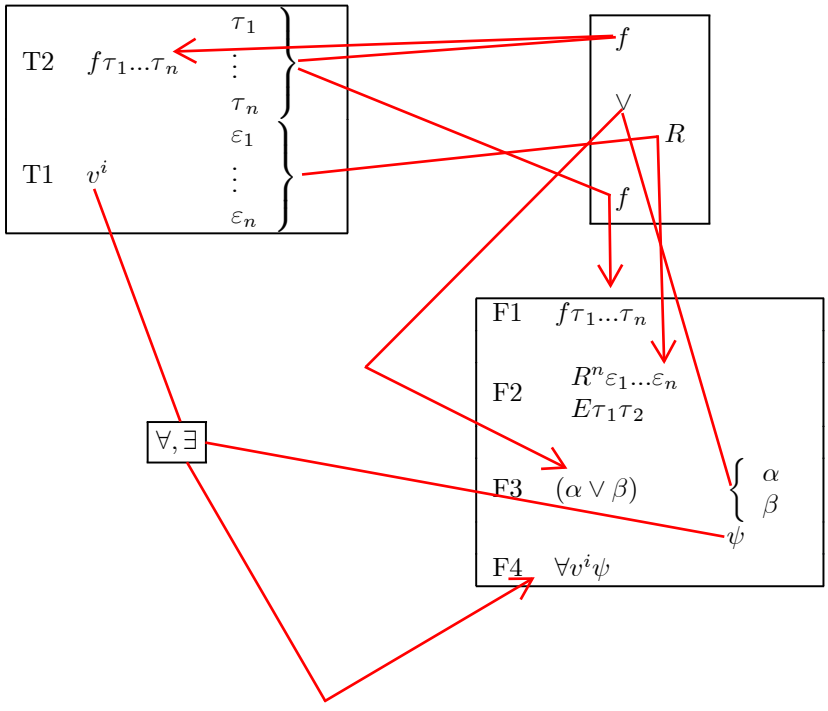
es decir, se trata de una relación $n-$*aria cuyas* $n-$*tuplas respetan la distinción de variedades. Finalmente, cuando* $FUNC(f) = n$ *entonces* $f^{\mathcal{A}}$ *es una relación* $n-$*aria establecida en el batiburrillo de la unión de tipos.*

Lenguaje

El alfabeto lo componen los siguientes signos

- Todos los signos del lenguaje, excepto los cuantificadores y las variables, son los que están incluídos en $OPER.SYM$
- Cuantificadores: $\forall, \exists$
- Variables: $v_0^i, v_1^i, v_2^i, ...$ para cada $i \in SORT - \{0\}$

Usando los signos del alfabeto se forman las expresiones del lenguaje; esto es, términos y fórmulas. Las reglas aparecen gráficamente representadas en el esquema siguiente:



7.3. Semántica

Asignación

Para poder definir las interpretaciones de las fórmulas precisamos de las asignaciones, que respetarán la clasificación en variedades

$$M : (\bigcup V_i)_{i \in SORT - \{0\}} \longrightarrow (\bigcup \mathbf{A}_{iI})_{i \in SORT - \{0\}}$$

—donde $M(v^i) \in \mathbf{A}_i$ para cada $i \in SORT$—

Interpretación

Finalmente definimos la interpretación

$$\Im = \langle \mathcal{A}, M \rangle$$

de expresiones de la manera habitual. En particular:

1. (E1) $\Im(v^i) = M(v^i)$
2. (E2) $\Im(f\varepsilon_1 ... \varepsilon_n) = f^{\mathcal{A}} \Im(\varepsilon_1) ... \Im(\varepsilon_n)$
3. (E3) $\Im(\exists v^i \varphi) = V$ syss $\left\{ \mathbf{a} \in \mathbf{A}_i \mid \Im^{\mathbf{a}}_{v^i}(\varphi) = V \right\} \neq \emptyset$

Consecuencia y validez

Las definiciones son las usuales. Las expresiones

$$\Gamma \vDash \varphi \quad \text{y} \quad \vDash \varphi$$

—denotan consecuencia y validez—

7.3.1. Metateoremas semánticos

De manera muy semejante a cómo se demuestran en primer orden[4], en la multivariada obtenemos:

Lema 239 *(de coincidencia). Sea $\mathcal{A}$ una estructura y M_1 y M_2 asignaciones sobre ellas*
Si

$$M_1 \restriction LBR(\varepsilon) = M_2 \restriction LBR(\varepsilon)$$

entonces

$$\langle \mathcal{A}, M_1 \rangle (\varepsilon) = \langle \mathcal{A}, M_2 \rangle (\varepsilon)$$

[4] El detalle puede encontrarse en [13].

Lema 240 *(de sustitución). Para cada expresión* ε *e interpretación* $\Im$ *:*

$$\Im_{v^i}^{\Im(\tau)}(\varepsilon) = \Im(\varepsilon\frac{\tau}{v^i})$$

—donde τ *es de tipo* i*—*

Lema 241 *(de sustitución de iguales). Para cada expresión* ε *e interpretación* $\Im$ *se verifica:*

$$\Im(\varepsilon\frac{\tau}{v^i}) = \Im(\varepsilon\frac{t}{v^i})$$

siempre que $\Im(\tau) = \Im(t)$ *y "*v^i *es del mismo tipo que* τ *syss* v^i *es del mismo tipo que* t*"*

Teorema 242 *(de isomorfía). Sean* $\mathcal{A}$ *y* $\mathcal{B}$ *con universos disjuntos. Si*

$$\mathcal{A} \cong^h \mathcal{B}$$

entonces

$$h\left\langle \mathcal{A}, M\right\rangle (\varepsilon) = \left\langle \mathcal{B}, h \circ M\right\rangle (\varepsilon)$$

7.4. Cálculo deductivo

Se puede extender un cálculo deductivo de primer orden; por ejemplo, el que contiene las reglas de[5]: Introducción de hipótesis *IH*, monotonía *M*, prueba por casos *PC*, no contradicción *NC*, introducción del disyuntor en el antecedente *IDA*, introducción del disyuntor en el consecuente *IDC*. A este cálculo se le añaden las siguientes:

1. Introducción del particularizador en el antecedente *IPA*

$$\frac{\Omega \quad \varphi\left(\frac{y^i}{x^i}\right) \dashv \psi}{\Omega \quad \exists x^i\varphi \dashv \psi}$$

   $y^i \notin LBR\left(\Omega \cup \left\{\exists x^i\varphi, \psi\right\}\right)$ x^i e y^i son del mismo tipo.

2. Introducción del particularizador en el consecuente *IPC*

$$\frac{\Omega \quad \dashv \varphi\left(\frac{\tau}{x^i}\right)}{\Omega \quad \dashv \exists x^i\varphi}$$

   x^i y τ son del mismo tipo

3. Reflexividad de la igualdad *RI*

$$\frac{}{\tau = \tau}$$

[5] Las reglas están en la página 56. Este es también el cálculo que uso en mi libro *Teoría de Modelos*.

4. Sustitución de iguales *SI*

$$\frac{\Omega \quad \dashv \varphi\left(\frac{\tau}{x^i}\right)}{\Omega \quad \tau = t \dashv \varphi\left(\frac{t}{x^i}\right)}$$

se cumple: "x^i es del mismo tipo que τ syss x^i es del mismo tipo que t"

7.4.1. Propiedades sintácticas

Se definen las propiedades siguientes de la forma habitual; a saber,

Definición 243 $\Delta \subseteq FORM$ *es* ***contradictorio*** *syss*

$$\Delta \vdash \varphi \quad \textit{para toda} \quad \varphi \in FORM$$

Definición 244 $\Delta \subseteq FORM$ *es* ***consistente*** *syss no es contradictorio*

Definición 245 $\Delta \subseteq FORM$ *es* ***máximamente consistente*** *syss* Δ *es consistente y siempre que* $\varphi \in FORM$ *y* $\varphi \notin \Delta$ *entonces* $\Delta \cup \{\varphi\}$ *es contradictorio*

Definición 246 $\Delta \subseteq FORM$ *es* ***ejemplificado*** *syss para cada particularización* $\exists x^i \varphi$,

$$\textit{si} \quad \exists x^i \varphi \in \Delta \quad \textit{entonces} \quad \varphi\left(\frac{\tau}{x^i}\right) \in \Delta$$

para algún τ *del mismo tipo que* x^i

7.4.2. Teoremas sobre consistencia

Se demuestra fácilmente que todo subconjunto de un conjunto consistente lo sigue siendo:

Teorema 247 *Si* Δ *es consistente y* $\Gamma \subseteq \Delta$ *entonces* Γ *es consistente*

Y también, que si un conjunto contiene a otro contradictorio, también él lo será.

Teorema 248 *Si* Δ *es contradictorio y* $\Delta \subseteq \Gamma$ *entonces* Γ *es contradictorio*

etc...

7.4.3. Teoremas sobre consistencia máxima

Sea Δ un conjunto de fórmulas máximamente consistente. Se puede demostrar que la pertenecia a un conjunto de esta clase lleva de alguna forma *"interiorizada"* la interpretación estándar de los signos lógicos y metalógicos.

Proposición 249 *Si* $\Delta \vdash \varphi$ *entonces* $\varphi \in \Delta$

Proposición 250 *Si* $\vdash \varphi$ *entonces* $\varphi \in \Delta$

Proposición 251 $\neg\varphi \in \Delta$ *syss* $\varphi \notin \Delta$

Proposición 252 $\varphi \vee \psi \in \Delta$ *syss* $\varphi \in \Delta$ *ó* $\psi \in \Delta$

etc...

Lema 253 *Finitud de las consistencia:* Δ *es consistente syss cada subconjunto finito de* Δ *lo es también.*

7.4.4. Corrección

Teorema 254 *Corrección del cálculo. Si* $\Gamma \vdash \varphi$ *entonces* $\Gamma \models \varphi$

Corolario 255 *Si* Γ *tiene un modelo entonces* Γ *es consistente*

7.5. Metateoremas de completud, compacidad y Löwenheim-Skolem

La demostración directa de estos teoremas sigue exactamente el mismo esquema y se sirve de los mismos argumentos que usamos en la sección 2.4 al demostrarlos para la de primer orden sin variedades[6]. Arrojando los resultados esperados.

Lema 256 *(Lindenbaum) Si* Γ *es consistente y tiene sólo un conjunto finito de variables libres, entonces hay un* Γ^* *máximamente consistente y ejemplificado que lo contiene* $\Gamma \subseteq \Gamma^*$ *y está escrito en el mismo lenguaje*

Lema 257 *(Henkin) Si* Γ^* *es máximamente consistente y ejemplificado, entonces* Γ^* *tiene un modelo numerable*

Corolario 258 *Si* Γ *es consistente y el conjunto de sus variables es finito, entonces* Γ *tiene un modelo numerable*

Lema 259 *Si* Γ *es consistente y* $\Gamma \subseteq \overline{\Gamma}$ *, siendo* $\overline{\Gamma}$ *un conjunto de sentencias que resulta de sustituir en* Γ *las variables libres por constantes nuevas y si* $\overline{\Gamma}$ *tiene un modelo numerable, entonces* Γ *tiene un modelo numerable*

[6] La demostración de este teorema para la lógica multivariada se halla desarrollada en detalle en [13], páginas 245-257.

Teorema 260 *Si* Γ *es consistente, entonces* Γ *tiene un modelo numerable*

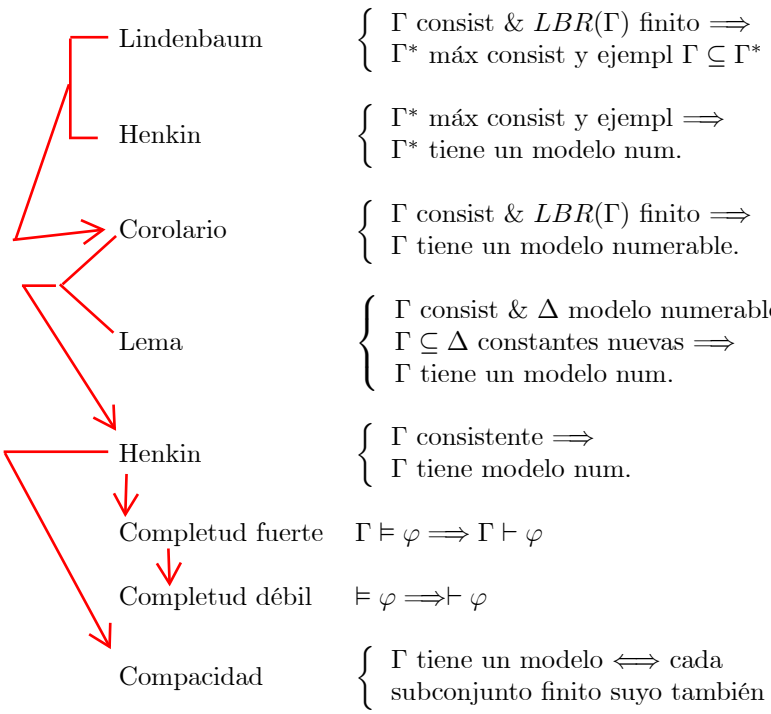
Teorema 261 *(Completud fuerte de* MSL*) Si* $\Gamma \vDash \varphi$ *entonces* $\Gamma \vdash \varphi$

Teorema 262 *(Completud débil de* MSL*) Si* $\vDash \varphi$ *entonces* $\vdash \varphi$

Teorema 263 *(Compacidad) Si* Γ *tiene un modelo, entonces cada subconjunto finito suyo también*

Teorema 264 *(Löwenheim-Skolem) Si* Γ *tiene un modelo, entonces tiene un modelo de universo numerable*

El esquema *argumental* es el que sigue:



7.6. Reducción de *MSL* a *FOL*

Para reducir la lógica multivariada a la de primer orden sin variedades es preciso definir una traducción sintáctica y una conversión de estructuras.

Sea L un lenguaje multivariado de signatura Σ y $OPER.SYM$ el conjunto de sus signos de operación. Vamos a definir un lenguaje de primer orden sin

variedades L^* añadiendo un relator unario Q_i para cada $i \in SORT - \{0\}$. Las variables de L se consideran variables de L^*[7], su signatura es Σ^*.

De hecho, la traducción que definimos a continuación es la relativización de cuantificadores a los nuevos relatores.

Definición 265 *$TRANS$ es una función que asigna a las expresiones de L expresiones en L^*, su traducción. La definición se hará mediante inducción:*

1. *(E1) Para cada variable de variedad i*

$$TRANS(x_i) = x_i$$

2. *(E2) Para las expresiones $\varepsilon\ \tau_{i_1}...\tau_{i_n}$*

$$TRANS(f\ \tau_{i_1}...\tau_{i_n}) = fTRANS(\tau_{i_1})...TRANS(\tau_{i_n})$$

$$TRANS(R\tau_{i_1}...\tau_{i_n}) = RTRANS(\tau_{i_1})...TRANS(\tau_{i_n})$$

3. *(E3) Para fórmulas cuantificadas*

$$TRANS(\exists x_i \varphi) = \exists x_i (Q_i x_i \wedge TRANS(\varphi))$$

Definición 266 *(Conversión de estructuras). Sea*

$$\mathcal{A} = \langle \langle \mathbf{A}_i \rangle_{i \in SORT - \{0\}}, \langle f^{\mathcal{A}} \rangle_{f \in OPER.SYM} \rangle$$

una estructura multivariada de signatura Σ. Vamos a construir una multivariada $\mathcal{A}^*$ por el método de unificación de dominios. Definimos:

$$\mathcal{A}* = \langle \bigcup_{i \in SORT - \{0\}} \mathbf{A}_i,\ \langle f^{\mathcal{A}*} \rangle_{f \in OPER.SYM - \{\neg, \vee\}},\ \langle Q_i^{\mathcal{A}*} \rangle_{i \in SORT - \{0\}} \rangle$$

donde

1. *El universo de $\mathcal{A}^*$ es la unión de los de $\mathcal{A}$*

2. *Para cada $f \in OPER.SYM$ con $FUNC(f) = \langle i_0, i_1, ..., i_n \rangle$ y $i_0 \neq 0$, entonces $f^{\mathcal{A}^*}$ es una extensión de $f^{\mathcal{A}}$ tal que*

$$Dom(f^{\mathcal{A}^*}) = (\bigcup_{i \in SORT - \{0\}} \mathbf{A}_i)^n \quad y \quad f^{\mathcal{A}^*} \restriction (\mathbf{A}_{i_1} \times ... \times \mathbf{A}_{i_n}) = f^{\mathcal{A}}$$

en donde los nuevos valores se eligen arbitrariamente. La única diferencia entre $f^{\mathcal{A}}$ y $f^{\mathcal{A}^*}$ es que el dominio de la segunda es considerablemente mayor que el de la primera y hay que especificar nuevas tuplas
Para cada $R \in OPER.SYM$ tal que o bien $FUNC(R) = \langle 0, i_1, ..., i_n \rangle$ o bien $FUNC(R) = n$, se define

$$R^{\mathcal{A}^*} = \{ \langle x_1, ..., x_n \rangle \in (A^*)^n / R^{\mathcal{A}}(x_1...x_n) = V \}$$

—en particular, $E^{\mathcal{A}^*} = E^{\mathcal{A}}$ y cuando la estructura es normal, la relación es la identidad—

[7] Esto no sería posible si la cardinalidad del conjunto $SORT$ fuera superior a $\aleph_0$. En tal caso la definición precisa de la traducción sería algo más compleja, pero posible.

3. *Cada* $Q_i^{\mathcal{A}^*} = \mathbf{A}_i$

Se pueden demostrar los teoremas que siguen:

Teorema 267 *Sea* $\mathcal{A}$ *una estructura normal de signatura* Σ *y* L *un lenguaje adecuado. Para cada sentencia* φ *se cumple*

$$\mathcal{A} \text{ es un modelo de } \varphi \text{ syss } \mathcal{A}^* \text{ es un modelo de } TRANS(\varphi)$$

Demostración. *Sea* $\mathcal{A}$ *una estructura normal de signatura* Σ *y* $\mathcal{A}^*$ *la conseguida mediante unificación de dominios. Lo que haremos es una reformulación más general del teorema, del que éste se seguirá sin dificultad. En realidad lo que se demuestra es que para cada asignación* M *sobre* $\mathcal{A}$

$$\langle \mathcal{A}, M\rangle(\epsilon) = \langle \mathcal{A}*, M\rangle(TRANS(\epsilon))$$

para cada término o fórmula del lenguaje multivariado, ϵ ■

Una estructura sin variedades $\mathcal{B}$ de signatura Σ^* no es siempre convertible en una multivariada. Hay dos problemas que pueden bloquear el proceso:

1. Las relaciones $Q_i^{\mathcal{B}}$ correspondientes a los nuevos relatores Q_i —aquellos que no estaban en $OPER : SYM$— podrían ser conjuntos vacíos
2. Para cada $f \in OPER.SYM$, $f^{\mathcal{B}}$ es simplemente una operación sobre $\mathcal{B}$ y no hay ninguna razón para que los valores de $f^{\mathcal{B}} \restriction (Q_{i_1}^{\mathcal{B}} \times ... \times Q_{i_n}^{\mathcal{B}})$ estén todos en $Q_{i_0}^{\mathcal{B}}$

Lo que haremos es formular en el lenguaje de primer orden tres condiciones que al cumplirse permiten que sus modelos sean fácilmente convertibles a sistemas multivariados. El conjunto Π lo constituyen todas las fórmulas de las siguientes formas:

1. $\exists x\, Q_i x$, para cada $i \in SORT - \{0\}$
2. $\forall x_1...x_n(Q_{i_1}x_1 \wedge ... \wedge Q_{i_n}x_n \rightarrow Q_{i_0} f\, x_1...x_n)$, donde $FUNC(f) = \langle i_0, i_1, ..., i_n\rangle$, $f \in OPER.SYM$
3. $Q_i c$, para cada $c \in OPER.SYM$ con $FUNC(c) = \langle i\rangle$

De manera que ahora un modelo $\mathcal{B}$ de Π se convierte fácilmente en una estructura multivariada $\mathcal{B}^{\blacktriangledown}$

Definición 268 *Sea* $\mathcal{B}$ *un modelo univariado de* Π *se define*

$$\mathcal{B}^{\blacktriangledown} = \langle \langle \mathbf{B}_i^{\blacktriangledown}\rangle_{i \in SORT}\ ,\ \langle f^{\mathcal{B}^{\blacktriangledown}}\rangle_{f \in OPER.SYM}\rangle$$

donde:

1. *Para cada* $i \in SORT - \{0\}$, $\mathbf{B}_i^{\blacktriangledown} = Q_i^{\mathcal{B}}$ *y* $\mathbf{B}_0^{\blacktriangledown} = \{V, F\}$

2. *Para cada* $f \in OPER.SYM$ *con* $FUNC(f) = \langle i_0, i_1, ..., i_n \rangle$

$$f^{\mathcal{B}^\blacktriangledown} = f^{\mathcal{B}} \restriction (Q^{\mathcal{B}}_{i_1} \times ... \times Q^{\mathcal{B}}_{i_n})$$

3. *Para cada* $f \in OPER.SYM$ *con* $FUNC(f) = \langle i \rangle$

$$f^{\mathcal{B}^\blacktriangledown} = f^{\mathcal{B}}$$

4. *Para cada* $R \in OPER.SYM$ *tal que* $FUNC(R) = \langle 0, i_1, ..., i_m \rangle$ *y cada* $\mathbf{x}_1 \in Q^{\mathcal{B}}_1, ..., \mathbf{x}_m \in Q^{\mathcal{B}}_m$

$$R^{\mathcal{B}^\blacktriangledown}(\mathbf{x}_1, ..., \mathbf{x}_m) = V \quad syss \quad \langle \mathbf{x}_1, ..., \mathbf{x}_m \rangle \in R^{\mathcal{B}}$$

5. *Para cada* $R \in OPER.SYM$ *tal que* $FUNC(R) = m$ *y cada*

$$\mathbf{x}_1 \in \bigcup_{i \in SORT - \{0\}} Q^{\mathcal{B}}_i, ..., \mathbf{x}_m \in \bigcup_{i \in SORT - \{0\}} Q^{\mathcal{B}}_i$$

se cumple

$$R^{\mathcal{B}^\blacktriangledown}(\mathbf{x}_1, ..., \mathbf{x}_m) = V \quad syss \quad \langle \mathbf{x}_1, ..., \mathbf{x}_m \rangle \in R^{\mathcal{B}}$$

Teorema 269 *Si* $\mathcal{B}$ *es una estructura univariada que es modelo de* Π*, entonces* $\mathcal{B}^\blacktriangledown$ *es una estructura multivariada. Además, para cada sentencia multivariada* φ *se cumple*

$$\mathcal{B}^\blacktriangledown \Vdash \varphi \quad syss \quad \mathcal{B} \Vdash TRANS(\varphi)$$

Demostración. *Sea* $\mathcal{B}$ *un modelo de* Π*. Queremos demostrar que* $\mathcal{B}^\blacktriangledown$ *es una estructura multivariada. De hecho:*

1. $\langle B^\blacktriangledown_i \rangle_{i \in SORT}$ *es una familia de conjuntos no vacíos.*
(Se sigue de que $\exists x Q_i x \in \Pi$*)*

2. $\langle f^{\mathcal{B}^\blacktriangledown} \rangle_{f \in OPER.SYM}$ *es una familia de funciones. Veámoslo:*
Cuando $RANK(f) = \langle i_0, ..., i_n \rangle$ *con* $i_0 \neq 0$*, entonces, por definición de* $\mathcal{B}^\blacktriangledown$ *restringimos el dominio de la función al nuevo dominio estratificado. Puesto que* $\mathcal{B}$ *es modelo de*

$$\forall x_1 ... x_n (Q_{i_1} x_1 \wedge ... \wedge Q_{i_n} x_n \rightarrow Q_{i_0} f\ x_1 ... x_n)$$

los valores de la función $f^{\mathcal{B}^\blacktriangledown}$ *están en* $Q^{\mathcal{B}}_{i_0}$
Cuando $FUNC(f) = \langle 0, i_1, ..., i_n \rangle$ *o cuando es una constante individual o un conector, las definiciones pertinentes, junto a los axiomas de la forma* $Q_i c$ *hacen que la estructura cumpla los requisitos necesarios para la conversión.*
El enunciado que se demuestra ahora mediante una sencilla prueba por inducción es el siguiente:
Para cada asignación multivariada M *se cumple:*

$$\langle \mathcal{B}^\blacktriangledown, M \rangle(\epsilon) = \langle \mathcal{B}, M \rangle(\mathbf{TRANS}(\epsilon))$$

Por consiguiente, $\mathcal{B}^\blacktriangledown \Vdash \varphi$ *syss* $\mathcal{B} \Vdash TRANS(\varphi)$

■

Teorema 270 *Sea* $\Gamma \cup \{\varphi\} \subseteq SENT(L)$. *Se verifica lo siguiente:*

$$\begin{array}{rrl} & \Gamma & \models \quad \varphi \quad \textit{en lógica multivariada} \\ \textit{syss} & \Pi \cup TRANS(\Gamma) & \models \quad TRANS(\varphi) \quad \textit{en univariada} \end{array}$$

—*donde* $TRANS(\Gamma) = \{TRANS(\psi)/\ \psi \in \Gamma\}$)—
Demostración. [⇒] *Sea* $\mathcal{B}$ *una estructura univariada de* $\Pi \cup TRANS(\Gamma)$. *Entonces* $\mathcal{B}^{\blacktriangledown}$ *es un modelo multivariado de* Γ. *Puesto que la hipótesis es que* $\Gamma \models \varphi$, $\mathcal{B}^{\blacktriangledown}$ *es también modelo de* φ. *Así que* $\mathcal{B}$ *es modelo de* $TRANS(\varphi)$ [⇐] *Sea* $\mathcal{A}$ *un modelo multivariado de* Γ. *Se vé claramente que* $\mathcal{A}^*$ *es un modelo univariado de* Π *y por lo tanto de* $TRANS(\Gamma)$. *Puesto que nuestra hipótesis es que*

$$\Pi \cup TRANS(\Gamma) \models TRANS(\varphi)$$

se sigue que $\mathcal{A}^*$ *es un modelo de* $TRANS(\varphi)$. *Por lo tanto,* $\mathcal{A}$ *es un modelo de* φ. ■

El teorema anterior nos permite inferir los tres teoremas siguientes de sus homónimos de la lógica de primer orden sin variedades: compacidad, enumerabilidad y Löwenheim-Skolem[8]. Pero nosotros ya los hemos demostrado directamente para la multivariada, así que en nuestro caso ha sido más un ejercicio de fidelidad histórica que de necesidad de obtener los resultados.

En el capítulo 13 realizamos la traducción inversa, de otras lógicas a la multivariada, y veremos cómo transferir propiedades del cálculo *MSL* a otros.

7.7. Implementaciones informáticas

Los siguientes proyectos de informática se hicieron bajo mi dirección, están disponibles en

$$http://aracne.usal.es$$

en donde se explican con detenimiento.

Todos ellos automatizan la traducción de lógicas:

1. *Traductor de Lógicas: Modal a Multivariada.* 1999. Iván Marcos Poza.
2. *Traductor de Lógicas: Dinámica a Multivariada.* [2000]. María Iglesias Alonso.
3. *Traductor de Lógicas: Multivariada a Primer Orden sin variedades.* [1999]. José Escuadra Burrieza.
4. *Traductor de Lógicas: Modal de Primer Orden a Multivariada y Parcial.* [2001]. Raquel Caño Mateos.

[8] See Enderton [1972], pp. 281.

Bibliografía

[1] Barwise, J. y Feferman, S. [1985]. *Model-Theoretic Logics.* Springer-Verlag. Berlín.

[2] Bramer, M, ed. [1986]. *Expert Systems 86.* Cambridge University Press. Cambridge. U.K.

[3] Cohn, A. [1986]. *"Many-sorted logic = unsorted logic + control"* en [2].

[4] Ebbinghaus, H.D. [1985]. *"Extended logics: the general framework"* en [1].

[5] Enderton, H. B. [1972]. *A Mathematical Introduction to Logic.* New York: Academic Press.

[6] Feferman, S. [1967]. *"Lectures on proof theory".* En **Proceedings of the Summer School in Logic, Leeds, 1967.** Lecture Notes in Mathematics, vol 70. Springer-Verlag. Berlín.

[7] Feferman, S. [1974]. *"Application of many-sorted interpolation theorems,* En **Proceedings of the Tarski Symposium.** Proc. Symps. in Pure Math., vol. XXV, pp 205-223,. Amer. Math. Soc. Providence.

[8] Gogen, J.A. y Meseguer, J. [1985]. *"Completeness of many-sorted equational logic".* Journal of Logic Programming, vol 1 (3), 179-210.

[9] Hook, J.L. [1985]. *"A note on interpretation of many-sorted theories".* Journal of Symbolic Logic, vol 50 (2), 372-374.

[10] Huertas, A. [1993]. *"El tercer valor de verdad en la lógica modal de predicados". Actas I Congreso de la Sociedad de Lógica, Metodología y Filosofía de la Ciencia.* Departamento de Reprografía de la UNED Madrid, pp. 80-83. Madrid.

[11] Huertas, A. [1994]. *Modal Logic and Non-Classical Logic,* tesis doctoral de la Universidad de Barcelona.

[12] Manzano, M. [1993]. *"Introduction to Many-sorted Logic"* en [12], páginas 1-86.

[13] Manzano, M. [1996]. *Extensions of first order logic*. Number 19 in Cambridge Tracts in Theoretical Computer Science. Cambridge University Press. Cambridge. U.K.

[14] Markusz, Z. [1981]. *"Knowledge representation of design in many-sorted logic",* Proc. Seventh. Int. Conf. on A.I. IJCAI-81 (Vancouver). 264-269.

[15] Markusz, Z. [1983]. *On first order many sorted logic.* Parts I y II. Computer and Automation Institute, Hungarian Academy of Science, Budapest, Tmulmányok 151/1983.

[16] Meinke, K. y Tucker, V. [1993]. *Many-sorted logic and its applications.* John Wiley & Sons. Chichester. U.K.

[17] Walther, C. [1983]. *A many-sorted calculus based on resolution and paramodulation.* Pitman. Londres.

[18] Wang, H. [1952]. *"Logic of many-sorted theories".* JSL 17 (2). 105-16

[19] Zucker, J.I. [1978]. *"The adequacy problem for classical logic".* Journal of Philosophical Logic. vol 7. 517-535.

Capítulo 8

Lógica Modal

8.1. Introducción

El término *"modalidad"* significa: *forma o manera de ser o de manifestarse una cosa*. En el caso de la lógica son los enunciados los que aparecen calificados modalmente, añadiendo a su alcance descriptivo o denotativo una nueva dimensión hasta cierto punto autorreflexiva. (En lógica modal escribimos $\Box\varphi$ para indicar que *"φ es una verdad necesaria"*.)

Para interpretar las fórmulas modales se precisan *contextos intensionales*, de referencias múltiples, que pueden ser de naturaleza temporal o incluir estados o situaciones diversos.

La oposición *"intensional"* versus *"extensional"* se encuentra en toda la tradición lógica, desde sus inicios hasta nuestros días. Sin embargo, los sistemas lógicos básicos son extensionales en la concepción de Frege: la lógica proposicional se ocupa del valor de verdad de las proposiciones, sin tener en cuenta otros aspectos del significado; la de predicados trata sólo las extensiones de los predicados —esto es, el conjunto de los objetos que los cumplen— y olvida otras características de los conceptos que corresponden a esos predicados. En el *Begriffsschrift* de Frege se critican implícitamente las nociones intensionales diciendo, por ejemplo, que en *"necesariamente A"* la información lógicamente relevante es *"A es verdadero"*, lo demás no pasa de ser percepción psicológica, semejante a cuando decimos *"lo creo firmemente"*.

Hay diversos sistemas lógicos en los que se toman los fenómenos intensionales como objeto de estudio. El ejemplo paradigmático nos lo proporciona la lógica modal, pero son también de esta clase la lógica epistémica —que se interesa por los enunciados de creencia y de conocimiento—, la deóntica —que califica los enunciados como obligatorios o simplemente permitidos—, la temporal —siempre, alguna vez— y la dinámica[1].

[1] A la que dedicamos el capítulo 9.

8.1.1. Historia

La historia de la lógica modal comienza en la época clásica, incluyendo el trabajo de Aristóteles, los megáricos (Diodoro Cronos) y los estoicos.

Los lógicos modales crearon un formalismo capaz de captar situaciones dinámicas, de relativizar la verdad. Desde sus inicios se destacó la relación entre nociones modales y temporales, siendo debatida por los megáricos y los estoicos. El tratamiento sistemático es de principios del siglo XX, con Lewis a la cabeza, aunque participan también Łukasievicz y Carnap. En esta fase se desarrollan los cálculos modales, pero la semántica está sólo apuntada, sugerida. Con los trabajos de Kanger, Hintikka, Prior y Kripke alcanza también ella una estructuración y sistematización notable. Con los de Lemmon, Scott y Segerberg la prueba de completud de los cálculos modales se lleva a término. Hay un cambio de perspectiva con van Benthem, Thomason, Goldblatt y otros, conocido como *teoría de la correspondencia*, que consiste fundamentalmente en tomarse en serio las intuiciones obvias sobre la lectura clásica de las fórmulas modales, desarrollándose también una nueva semántica modal. Con posterioridad a 1975 su uso en aplicaciones tanto a la Inteligencia Artificial como en informática teórica ha hecho que se diversifique enormemente.

Para Aristóteles el tema principal es la diferencia entre *necesidad* y *posibilidad*.

Escribamos $\Box\varphi$ para expresar que *"φ es una proposición necesaria"* y $\Diamond\varphi$ para expresar que *"φ es una proposición posible"*. Aristóteles ve la contradicción existente entre los pares

$$(\Box\varphi \text{ y } \neg\Box\varphi) \text{ por un lado, y } (\Diamond\varphi \text{ y } \neg\Diamond\varphi) \text{ por otro} \tag{8.1}$$

pero piensa que no son contradictorios

$$(\Box\varphi \text{ y } \neg\Diamond\varphi) \text{ ni } (\Diamond\varphi \text{ y } \neg\Box\varphi) \tag{8.2}$$

—porque aunque $\neg\Diamond\varphi$ es lo mismo que $\Box\neg\varphi$, y $\neg\Box\varphi$ es lo mismo que $\Diamond\neg\varphi$ los otros pares podrían darse a la vez—

NECESARIO	POSIBLE
$\Box\varphi$	$\Diamond\varphi$
$\neg\Diamond\neg\varphi$	$\neg\Box\neg\varphi$

IMPOSIBLE	INNECESARIO
$\neg\Diamond\varphi$	$\neg\Box\varphi$
$\Box\neg\varphi$	$\Diamond\neg\varphi$

El denominado cuadro de Boecio de la silogística (ver página 104) aparece ahora así modificado (ver figura: 8.1)

Aristóteles acepta como principio válido el siguiente:

$$\alpha := \Box\varphi \rightarrow \Diamond\varphi$$

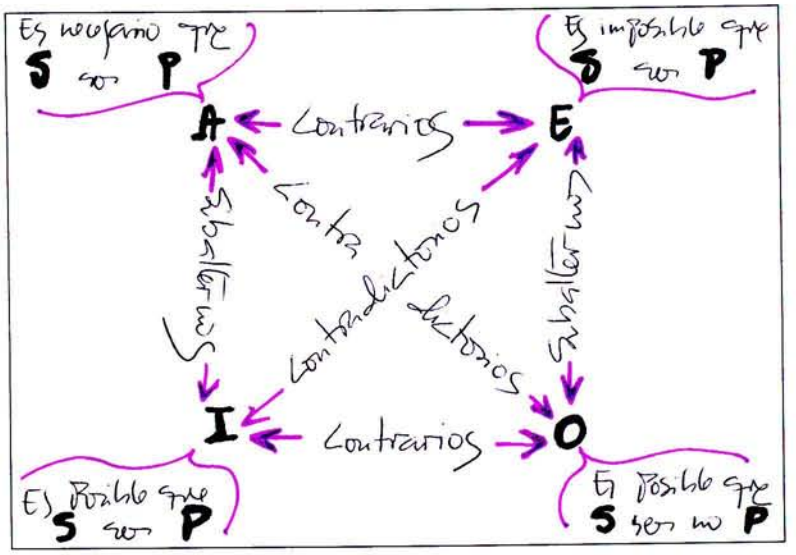


Figura 8.1: *Cuadrado modal*

y tiene dudas sobre este otro:

$$\beta := \Diamond\varphi \to \Diamond\neg\varphi$$

Por supuesto, comprende que α y β no pueden ser verdaderos simultáneamente. En α y β se usan dos sentidos distintos de *"posible"*. En el primer caso, el sentido de *"posible"* es el de que sea *"verdadera en alguna situación"*. En el segundo caso *"posible"* significa *"contingente"* —esto es, verdadero en unos casos y falso en otros—.

La formalización de enunciados modales es a veces problemática. El ejemplo tradicional, de Aristóteles, es el de la Batalla naval.

Dice el almirante:

"Si doy la orden de ataque (p), entonces, necesariamente, habrá una batalla naval mañana (q). Si no la doy, entonces, necesariamente, no la habrá. Yo doy la orden o no la doy. Luego: O bien es necesario que haya una batalla naval mañana o es necesario que no la haya."

La conclusión sorprende, *¿Es correcto el razonamiento?*

Formalización 1

- Hipótesis: $\{p \to \Box q, \neg p \to \Box\neg q, p \vee \neg p\}$
- Conclusión: $\Box q \vee \Box\neg q$

Formalización 2

- Hipótesis: $\{\Box(p \to q), \Box(\neg p \to \neg q), p \vee \neg p\}$

- Conclusión: $\Box q \vee \Box \neg q$

Comentario 271 *En la primera de las propuestas la conclusión se sigue deductivamente de las hipótesis, sin embargo la formalización de las dos primeras hipótesis no es plausible. En la segunda sí que lo es, pero la conclusión no se sigue de las hipótesis.*

La *relación entre nociones modales y temporales* es muy antigua. Diodoro define:

1. *lo posible* $\Diamond$ como aquello que *es o será*
2. *lo imposible* $\neg\Diamond$ como aquello que *siendo falso no será verdadero*
3. *lo necesario* $\Box$ como aquello que *es verdadero y no será falso*
4. *lo no necesario* $\neg\Box$ como aquello que *es o será falso*

Las denominadas *paradojas de la implicación material* a las que se responsabiliza del nacimiento de la lógica modal fueron anticipadas por los filósofos medievales. Las más importantes son:

1. *"una proposición verdadera es implicada por cualquier proposición"*

$$q \rightarrow (p \rightarrow q)$$

2. *"toda proposición falsa implica cualquier proposición"*

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

3. *"todas las proposiciones están conectadas por implicación"*

$$(p \rightarrow q) \vee (q \rightarrow p)$$

Ninguna de ellas es contradictoria, si aceptamos el uso clásico —extensional— del condicional material.

p	q	$p \rightarrow q$
1	1	1
1	0	0
0	1	1
0	0	1

¿Capta este condicional el sentido de la implicación?

Si así fuera, sorprende en todos los casos la debilidad del vínculo entre antecedente y consecuente de una implicación ya que sería de desear una relación mucho más fuerte, de imposibilidad de antecedente sin consecuente —o tal vez

más constructiva: indicando cómo q se obtiene de p al operarlo de una cierta manera para que $p \rightarrow q$ sea una implicación aceptable—.

El pistoletazo de salida de la carrera modal en el siglo XX se produce con las investigaciones de Lewis de 1912 cuyo origen hay que buscarlo justamente en el descontento creado por la interpretación de la implicación en el sistema de Russell y Frege; esto es, en la tabla de verdad del condicional —también denominado *implicación material*— que posibilita las paradojas y que hace que la fórmula

$$\alpha \rightarrow \beta$$

sea equivalente a

$$\neg(\alpha \wedge \neg\beta)$$

y que él considera que debería tener la potencia de un

$$\neg\Diamond(\alpha \wedge \neg\beta)$$

o, de forma equivalente, de un

$$\Box(\alpha \rightarrow \beta)$$

Para solventar el problema se introduce un nuevo operador binario intensional, no extensional, para la denominada *implicación estricta*

$$\alpha \hookrightarrow \beta$$

o, de forma equivalente, se la define con el operador de necesariedad.

Una observación semejante es extensiva al tratamiento de otras conectivas; por ejemplo, en los enunciados α y β

- α : John Lenon murió o John Lenon es una piedra
- β : Eros no me ama o soy amada

la fuerza de la disyunción es muy distinta. En α es meramente *"material"* su verdad en el mundo real depende del hecho empírico de que Lenon muriera. En β no hace falta saber en qué mundo estamos, ni quien es Eros ni quien soy yo. Así que se propone una nueva disyunción no extensional, paralela a la del operador $\hookrightarrow$.

La pregunta crucial es, *¿debemos expresar en el lenguaje objeto las diferencias entre la disyunción de α y β?*

Lewis piensa que sí y por ello introduce sus conocidos sistemas modales, desde *S1* a *S5*. El procedimiento es sólo sintáctico, mediante axiomas y reglas de inferencia que seleccionan del conjunto de todas las fórmulas a sus teoremas lógicos modales. Todavía la semántica es intuitiva, no está matematizada, no hay tampoco una lectura única de los operadores modales, lo que origina en consecuencia varias lógicas o sistemas modales.

La interpretación de las conectivas como funciones veritativas es ciertamente fructífera. Sin embargo, *¿cómo interpretar* $\Diamond\varphi$ *y* $\Box\varphi$*?*, ninguna de las conectivas monarias sobre $\{1, 0\}$ serviría para ello. Una posibilidad, que apunta Łukasiewicz, es tomar más valores de verdad. Así:

φ	$\Box\varphi$
1	1
1/2	0
0	0

φ	$\Diamond\varphi$
1	1
1/2	1
0	0

Dejando a un lado el tratamiento algebraico, descrito por algunos como *sintaxis disfrazada,* el que primero proporcionó una semántica a las fórmulas modales fue Carnap. Algunos consideran que en él se funden tres corrientes fundamentales: Frege, Leibniz y Wittgenstein. De Frege[2] le viene su interés por la semántica, incluyendo la distinción entre *intensión* y *extensión.* De Leibniz la interpretación de *necesariedad* como *verdad en todo mundo posible*, y *posibilidad* como *verdad en algunos*, proporcionando de esta forma una semántica para *S5.* De Wittgenstein tomó la idea de un *mundo de hechos atómicos* y de *descripciones de estados.* Veremos cómo se reelaborará esta intuición en los denominados *modelos canónicos*, que se usan para probar completud. En la semántica de Carnap tenemos:

- U : descripciones de estados
- s : un estado
- p : proposición atómica

Definiéndose

$$U, s \Vdash p \quad \text{syss} \quad p \in s$$

y para los operadores modales

$$U, s \Vdash \Box\varphi \quad \text{syss para todo} \quad t \in U : U, t \Vdash \varphi$$

El sistema *S5* de Lewis es completo con la semántica de Carnap.

La semántica actualmente más usada es la de Kripke: La razón principal es que es muy versátil, permite poner énfasis en las distintas relaciones de accesibilidad, que proporcionan la *"clave semántica"* para la modalidad. Se definen los modelos de la lógica modal como tripletes

$$\mathcal{A} = \left\langle \mathbf{W}, \mathbf{R}, \langle p^{\mathcal{A}} \rangle_{p \in \mathbf{ATOM}} \right\rangle \tag{8.3}$$

donde:

W : estados, mundos, situaciones, puntos,...

R : relación de accesibilidad

[2] De hecho Carnap fue uno de los escasísimos discípulos de Frege.

$p^{\mathcal{A}}$: conjunto de mundos donde p es verdadera

¿Qué significado intuitivo tienen las fórmulas?
Esta semántica permite muchas lecturas de los operadores modales:

- *Necesario (Lebniz)*: Una verdad es necesaria cuando lo es en todo mundo posible
  $\Box\varphi$ significa: φ es necesario en todo mundo posible
  $\mathbf{W}$: conjunto de mundos
  $\mathbf{R}$: relación de accesibilidad entre mundos.
  $\langle s,t\rangle \in \mathbf{R} : t$ es una alternativa a s —todas las verdades necesarias de s se verifican—

- *Necesario en Física* ($\neq$ Necesario en Lógica)
  $\Box\varphi$: φ es consecuencia de las leyes de la física
  $\langle s,t\rangle \in \mathbf{R} : t$ es una alternativa científica a s.
  Por ejemplo, en nuestro mundo es verdadera $\Box(x < c)$
  c : velocidad de la luz
  x : velocidad de un objeto material
  sin embargo, no es una verdad lógica.

- *Necesario en lógica deóntica*
  $\Box\varphi$ significa: φ es obligatorio
  $\langle s,t\rangle \in \mathbf{R} : t$ es una alternativa moral a s.

- *Necesario en lógica temporal*
  $\Box\varphi$ significa: *"en todo momento futuro,* φ"
  $\Diamond\varphi$ significa: *"en algún momento futuro,* φ"
  $\mathbf{W}$: momentos (tiempo) —$\mathbf{W}$ puede ser $\mathbb{N}$, $\mathbb{Z}$, $\mathbb{Q}$, $\mathbb{R}$—
  $\langle s,t\rangle \in \mathbf{R} : t$ es posterior a s.

- *Lógica dinámica*
  $\Box\varphi$ significa: cada ejecución del programa que termina, lo hace en un estado en donde vale φ
  $\Diamond\varphi$ significa: hay una ejecución del programa que terminaen un estado en donde vale φ
  $\mathbf{W}$: estados posibles de un proceso de computación
  $\langle s,t\rangle \in \mathbf{R}$: hay una ejecución del programa que empieza en estado s y termina en estado t

8.1.2. Necesario en lógica: El sistema S5

Es muy fácil entender el planteamiento en el caso de la lógica *S5*, la que capta y sistematiza la *necesariedad lógica*. Observemos nuestras fórmulas clásicas desde la atalaya del metalenguaje; podremos en él expresar las características de las expresiones clásicas pertinentes al nivel —esto es, su *validez, satisfacibilidad, etc*— usando la lengua natural, utilizando signos como abreviaturas

—recuérdese que $\models$ no es un signo del lenguaje objeto— o introduciendo el lenguaje modal.

EN ESPAÑOL	SIGNOS	LENGUAJE MODAL
φ válida φ tautología	$\models \varphi$	φ necesaria $\Box\varphi$
φ no válida	$\not\models \varphi$	φ innecesaria $\neg\Box\varphi$ $\Diamond\neg\varphi$
φ contradicción φ insatisfacible	$\models \neg\varphi$	φ imposible $\Box\neg\varphi$ $\neg\Diamond\varphi$
φ satisfacible $\neg\varphi$ no es válida	$\not\models \neg\varphi$	φ posible $\Diamond\varphi$
φ contingente	$\not\models \varphi$ y $\not\models \neg\varphi$	$\Diamond\neg\varphi \wedge \Diamond\varphi$

De esta forma, en la lógica modal (*PML*) hemos formalizado el metalenguaje (español o español con abreviaturas) que empleábamos para hablar del lenguaje formal de la lógica proposicional (*PL*).

La ventaja es que ahora podremos no sólo expresar nuestros juicios sobre las fórmulas de *PL*, sino que también los podremos verificar mediante un cálculo deductivo; el de la lógica modal *S5*

Ejemplo 272 *En la figura de clasificación de fórmulas se ve claramente que las verdades necesarias constituyen un subconjunto del de las posibles y que las imposibles son un subconjunto de las no necesarias.*

En lógica modal lo expresamos así:

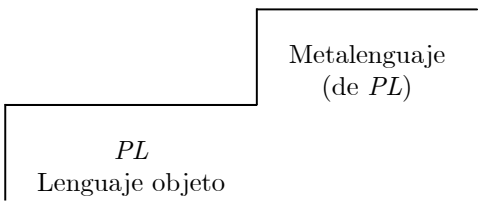
$$\Box\varphi \rightarrow \Diamond\varphi$$

$$\Box\neg\varphi \rightarrow \neg\Box\varphi$$

Incluso, y esta es la ventaja principal, podremos demostrarlo en un cálculo deductivo.

Vamos a explicar de manera gráfica lo que hacemos al definir *S5.*

Para expresar ciertas características de las fórmulas de la lógica proposicional, usamos habitualmente el metalenguaje.



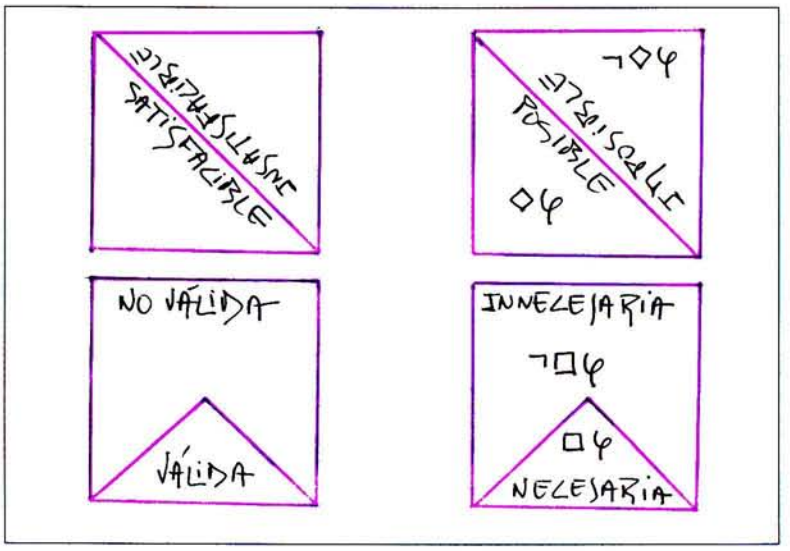
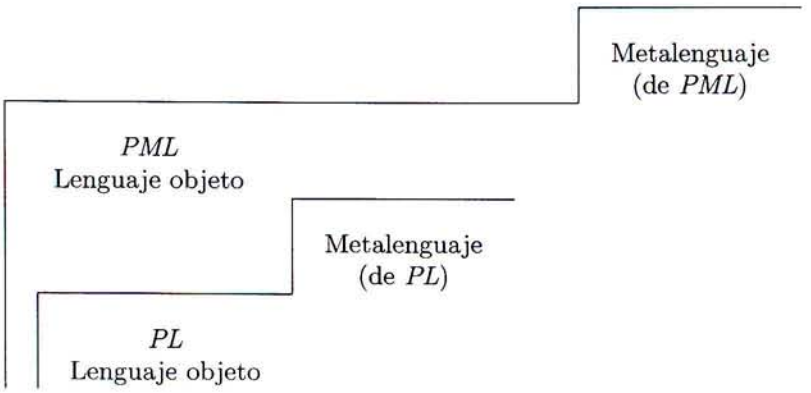


Figura 8.2: *Clasificación de fórmulas*

Si ahora formalizamos este metalenguaje obtenemos un nuevo lenguaje objeto *PML* en el que no sólo podremos establecer las propiedades de *PL*, también las podremos computar en un cálculo. Por supuesto, para expresar validez y satisfacibilidad de las fórmulas de este lenguaje nuevo necesitamos subir otro nivel.



Resumen 273 *Hay lecturas distintas del término* "necesario" *correspondientes a la lógica, la física, la moral, etc.*

$\Box\varphi$: φ *es necesariamente verdadera*
φ *es siempre verdadero*
φ *es obligatoriamente verdadero*
φ *es sabido*
φ *es creído*
φ *es demostrable (en la Aritmética)*
φ *es verdadero tras ejecutar el programa*
etc.
y en consecuencia el catálogo de lógicas modales es amplio:
Normales (K, T, $S4$, $S5, ...$*), temporales, dinámicas, epistémicas, deónticas, de la demostrabilidad, etc.*
Por consiguiente, no hay una única lógica modal, sino multitud de ellas. Ello se concreta a nivel sintáctico y semántico.

1. ***Sintácticamente****: distintos axiomas para las distintas lógicas*

2. ***Semánticamente:*** *distintas propiedades de la relación de accesibilidad*

Comentario 274 *De hecho, los axiomas de una lógica modal intentan caracterizar las propiedades de la relación de accesibilidad que les es propia. Y no sólo eso: el lenguaje modal puede entenderse como el lenguaje ideal para expresar propiedades de las relaciones binarias, como se verá en lo que sigue. Es también un lenguaje muy equilibrado: es bastante expresivo, sin dejar de ser decidible.*

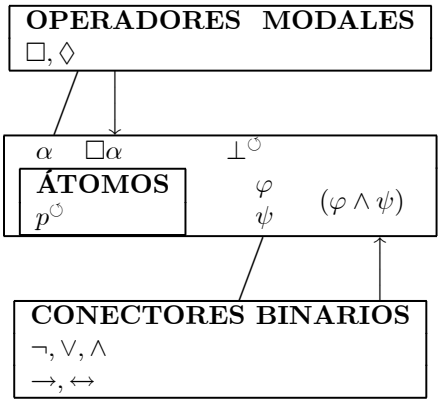
8.2. Lenguaje y Semántica

8.2.1. El lenguaje de la lógica modal

La lógica modal proposicional añade los operadores modales $\Box$ y $\Diamond$ al alfabeto clásico, y una nueva regla de formación para ellos.

$$\langle fórmula\rangle := \langle \mathbf{ATOM}\rangle \mid \bot \mid \neg\langle fórmula\rangle \mid \langle fórmula\rangle \wedge \langle fórmula\rangle \mid \langle fórmula\rangle \vee \langle fórmula\rangle \mid \langle fórmula\rangle \rightarrow \langle fórmula\rangle \mid \langle fórmula\rangle \leftrightarrow \langle fórmula\rangle \mid \Box\langle fórmula\rangle \mid \Diamond\langle fórmula\rangle$$

Es decir, además de las atómicas (regla **F1),** tenemos las fórmulas formadas mediante la regla **F2**: $\bot$, $\neg\varphi$, $(\varphi \wedge \psi)$, $(\varphi \vee \psi)$, $(\varphi \rightarrow \psi)$, $(\varphi \leftrightarrow \psi)$ y las que usan la regla modal **F3**: $\Box\varphi$ y $\Diamond\varphi$.



Definición 275 FORM *es el menor conjunto que contiene a las fórmulas atómicas y está cerrado bajo* ***F2*** *y* ***F3***

Ejemplo 276 *Como ejercicios de formalización de enunciados de la lengua natural podemos plantear los siguientes:*

1. *Nada es completamente relativo:* $\neg\Box(\Diamond\alpha \wedge \Diamond\neg\alpha)$
2. *Es posible que no me entiendas, pero no es así necesariamente:* $\Diamond\neg p \wedge \neg\Box\neg p$
3. *Si puede que llueva, entonces necesariamente puede llover:* $\Diamond q \rightarrow \Box\Diamond q$
4. *Puede que me suspendan, pero tal esto vez no sea necesario:* $\Diamond r \wedge \Diamond\neg\Box r$

8.2.2. Modelos de Kripke

Como se comentó con anterioridad, en un modelo de Kripke se nos da: un conjunto $\mathbf{W} \neq \emptyset$ de estados, mundos, situaciones, puntos,...; una relación binaria, de accesibilidad entre ellos, $\mathbf{R} \subseteq \mathbf{W} \times \mathbf{W}$; y, para interpretar a cada fórmula atómica, un subconjunto de mundos, $p^{\mathcal{A}} \subseteq \mathbf{W}$.

Dado un modelo de Kripke

$$\mathcal{A} = \left\langle \mathbf{W}, \mathbf{R}, \langle p^{\mathcal{A}} \rangle_{p \in \mathbf{ATOM}} \right\rangle \tag{8.4}$$

definimos por inducción sobre la construcción de fórmulas una función de interpretación

$$\Im : \mathbf{FORM}(PML) \longrightarrow \wp(\mathbf{W}) \tag{8.5}$$

que a cada fórmula $\varphi \in \mathbf{FORM}(PML)$ le asigna como interpretación $\Im(\varphi) \subseteq \mathbf{W}$ —conjunto de los mundos donde φ es verdadera—

- **F1** Para fórmulas atómicas: $\Im(p) = p^{\mathcal{A}}$

- **F2** Para fórmulas obtenidas con operadores booleanos a partir de fórmulas cualesquiera: $\Im(\bot) = \emptyset$, $\Im(\varphi \wedge \psi) = \Im(\varphi) \cap \Im(\psi)$, $\Im(\varphi \rightarrow \psi) = \mathbf{W} - (\Im(\varphi) \cap (\mathbf{W} - \Im(\psi))$, de forma similar para el resto de conectivas
- **F3** Para fórmulas obtenidas con operadores modales a partir de fórmulas cualesquiera:

$$\Im(\Diamond\varphi) = \{s \in \mathbf{W} \mid \exists t(t \in \mathbf{W} \ \& \ \langle s,t\rangle \in \mathbf{R} \ \& \ t \in \Im(\varphi))\}$$

$$\Im(\Box\varphi) = \{s \in \mathbf{W} \mid \forall t(t \in \mathbf{W} \ \& \ \langle s,t\rangle \in \mathbf{R} \Rightarrow t \in \Im(\varphi))\}$$

Comentario 277 *En lógica modal la interpretación de una fórmula no es un valor de verdad, sino el conjunto formado por todos los estados o mundos donde la fórmula es verdadera. A continuación veremos cómo podemos usar las interpretaciones para definir el operador de satisfacibilidad y la relación de consecuencia*

8.2.3. Marcos de Kripke

Llamamos marcos de Kripke a las estructuras relacionales. En el caso de la lógica multimodal hay varias relaciones, en el que nos ocupa, una sola, pudiéndose decir que son todos de la forma

$$\mathcal{M} = \langle \mathbf{W}, \mathbf{R} \rangle$$

donde $\mathbf{W} \neq \varnothing$ y $\mathbf{R} \subseteq \mathbf{W} \times \mathbf{W}$. Podemos interpretar las fórmulas modales usando estos marcos, siempre que definamos además una asignación

$$H : \mathbf{ATOM} \longrightarrow \wp(\mathbf{W})$$

que con cada fórmula atómica asocia un subconjunto de $\mathbf{W}$. De manera que

$$\mathcal{A} = \langle \mathcal{M}, H \rangle$$

será un modelo de Kripke —hacemos $H(p) = p^{\mathcal{A}}$— sobre el que se define la interpretación $\Im$ de la manera antes dicha. Escribiremos $\Im_H$ cuando la referencia explícita a la asignación sea relevante —algunos autores escriben $[|\varphi|]_H$ en vez de $\Im_H(\varphi)$—.

Claramente con cada marco de Kripke se crean tantos modelos distintos como funciones H se puedan definir, y cada modelo contiene un marco.

8.2.4. Estructuras por mundos (o puntos)

Dado un modelo de Kripke

$$\mathcal{A} = \left\langle \mathbf{W}, \mathbf{R}, \langle p^{\mathcal{A}} \rangle_{p \in \mathbf{ATOM}} \right\rangle$$

y un elemento de su universo $s \in \mathbf{W}$ diremos que

$$\left\langle \mathbf{W}, \mathbf{R}, \langle p^{\mathcal{A}} \rangle_{p \in \mathbf{ATOM}}, s \right\rangle$$

—que abreviadamente escribiremos $\langle \mathcal{A}, s \rangle$— es la estructura de Kripke $\mathcal{A}$ en el mundo s. La interpretación $\Im_s$ de una fórmula modal en una de estas estructuras utiliza la definida para el modelo y extrae de él un valor de verdad, F o V. Como era previsible su definición es:

$$\Im_s(\varphi) = V \text{ syss } s \in \Im(\varphi)$$

8.2.5. Tres *"satisfacciones"*

Para cada conjunto de letras proposicionales **ATOM** hemos definido tres clases distintas de estructuras asociadas. Tenemos los marcos $\mathcal{M} = \langle \mathbf{W}, \mathbf{R} \rangle$, que pueden convertirse en modelos al unírseles asignaciones $\mathcal{A} = \langle \mathcal{M}, H \rangle$ y tenemos también las estructuras por puntos $\langle \mathcal{A}, s \rangle$. Es importante no confundirlas, pues cada una juega su papel. Hay, por lo tanto tres relaciones de satisfacción correspondientes.

Sea $\mathcal{A} = \langle \mathcal{M}, H \rangle$ un modelo de Kripke, $\Im$ su interpretación asociada, $\mathcal{M} = \langle \mathbf{W}, \mathbf{R} \rangle$ su marco y $s \in \mathbf{W}$.

1. φ *es verdadera en estado* s *del modelo* $\mathcal{A}$: $\mathcal{A}, s \Vdash \varphi \Longleftrightarrow_{Df} s \in \Im(\varphi)$
   Esta es la relación básica de satisfacción.

2. φ *es válida (o verdadera) en el modelo* $\mathcal{A}$
   $\mathcal{A} \Vdash \varphi \Longleftrightarrow_{Df} \forall s \in \mathbf{W}$ ($\mathcal{A}$,$s \Vdash \varphi$) —i.e. $\Im(\varphi) = \mathbf{W}$—

3. φ *es válida (o verdadera) en el marco* $\mathcal{M}$
   $\mathcal{M} \Vdash \varphi \Longleftrightarrow_{Df} \forall H$ ($\langle \mathcal{M}, H \rangle \Vdash \varphi$) —i.e. $\forall H$ ($\Im_H(\varphi) = \mathbf{W}$)—
   Cuando queramos distinguirlas explícitamente podemos usar superíndices
   $\Vdash^p$ satisfacción por puntos, la primera
   $\Vdash^v$ satisfacción por valoraciones o asignaciones, la segunda
   $\Vdash^m$ satisfacción por marcos, la tercera

4. φ *es válida:* $\models \varphi \Longleftrightarrow_{Df} \forall \mathcal{A}\ \mathcal{A} \Vdash \varphi$ (para todo modelo)

5. φ *es marco-válida:* $\models^m \varphi \Longleftrightarrow_{Df} \forall \mathcal{M}\ \mathcal{M} \Vdash^m \varphi$ (para todo marco)
   $\Longleftrightarrow_{Df} \forall \mathcal{M} H\ \langle \mathcal{M},\ H \rangle \Vdash^v \varphi$ (para todo marco y todo modelo sobre él construido)

6. φ *es válida en la clase de modelos* $\mathfrak{D}$
   $\models_{\mathfrak{D}} \varphi \Longleftrightarrow_{Df} \forall \mathcal{A} \in \mathfrak{D}$ [$\mathcal{A} \Vdash \varphi$] (para todo modelo de la clase)

7. φ *es válida en la clase de marcos* $\mathfrak{F}$
   $\models^m_{\mathfrak{F}} \varphi \Longleftrightarrow_{Df} \forall \mathcal{M} \in \mathfrak{F}$ [$\mathcal{M} \Vdash^m \varphi$] (para todo marco de la clase)

8. Γ *es satisfacible en estado* s *del modelo* $\mathcal{A}$
   $\mathcal{A}$,$s \Vdash \Gamma \Longleftrightarrow_{Df} s \in \bigcap \Im(\varphi)$

9. Γ *es válido en el modelo* $\mathcal{A}$
   $\mathcal{A} \Vdash \Gamma \Longleftrightarrow_{Df} \bigcap_{\varphi\in\Gamma} \Im(\varphi) = \mathbf{W}$

10. φ *es consecuencia de* Γ *en la clase de modelos* $\mathfrak{D}$
    $\Gamma \models_{\mathfrak{D}} \varphi \Longleftrightarrow_{Df}$ *Para todo* $\mathcal{A} \in \mathfrak{D}$ y $s \in W$: *si* $s \in \bigcap_{\gamma\in\Gamma} \Im(\gamma)$ *entonces* $s \in \Im(\varphi)$

11. φ *es consecuencia de* Γ
    $\Gamma \models \varphi \Longleftrightarrow_{Df}$ *Para todo modelo* $\mathcal{A}$ *y* $s \in W$: *si* $s \in \bigcap_{\gamma\in\Gamma} \Im(\gamma)$ *entonces* $s \in \Im(\varphi)$

8.2.6. Ejercicios

Se puede ver que las siguientes fórmulas son válidas en todo modelo y en todo marco, pero que no expresan condición alguna sobre la relación de accesibilidad.

1. $\Box\neg\bot$
2. $\Box(p \vee \neg p)$
3. $\Box(p \to (q \to p))$
4. $\Box(\alpha \to \beta) \to (\Box\alpha \to \Box\beta)$
5. $\Box(\alpha \wedge \beta) \leftrightarrow (\Box\alpha \wedge \Box\beta)$
6. $\Box(\alpha \to \beta) \to (\Diamond\alpha \to \Diamond\beta)$
7. $\Diamond(\alpha \vee \beta) \leftrightarrow (\Diamond\alpha \vee \Diamond\beta)$
8. $\Diamond(\alpha \to \beta) \to (\Box\alpha \to \Diamond\beta)$

Sin embargo, se pueden poner como ejemplos los de las fórmulas siguientes, que no son válidas.

1. $D := \Box\varphi \to \Diamond\varphi$
   tomando: $\mathcal{A} = \left\langle \mathbf{W}, \mathbf{R}, \langle p^{\mathcal{A}} \rangle_{p\in\mathbf{ATOM}} \right\rangle$, $\mathbf{W} = \{1\}$, $\mathbf{R} = \emptyset$ y $p^{\mathcal{A}} = \emptyset$
2. $T := \Box\varphi \to \varphi$
   tomando (ver figura: 8.3): $\mathcal{A} = \left\langle \mathbf{W}, \mathbf{R}, \langle p^{\mathcal{A}} \rangle_{p\in\mathbf{ATOM}} \right\rangle$, $\mathbf{W} = \{0, 1\}$, $\mathbf{R} = \{\langle 0, 1\rangle\}$ y $p^{\mathcal{A}} = \{1\}$
3. $B := \varphi \to \Box\Diamond\varphi$
   tomando (ver figura: 8.4): $\mathcal{A} = \left\langle \mathbf{W}, \mathbf{R}, \langle p^{\mathcal{A}} \rangle_{p\in\mathbf{ATOM}} \right\rangle$, $\mathbf{W} = \{0, 1\}$, $\mathbf{R} = \{\langle 0, 1\rangle\}$ y $p^{\mathcal{A}} = \{0\}$
4. $4 := \Box\varphi \to \Box\Box\varphi$
   tomando (ver figura: 8.5): $\mathcal{A} = \left\langle \mathbf{W}, \mathbf{R}, \langle p^{\mathcal{A}} \rangle_{p\in\mathbf{ATOM}} \right\rangle$, $\mathbf{W} = \{0, 1, 2\}$, $\mathbf{R} = \{\langle 0, 1\rangle, \langle 1, 2\rangle\}$ y $p^{\mathcal{A}} = \{1\}$

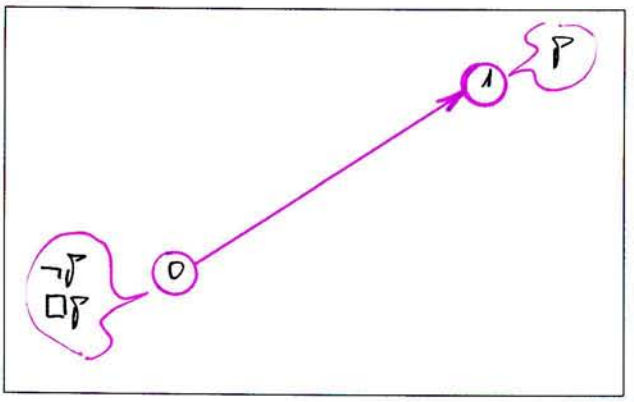


Figura 8.3: $\mathbf{R}$ *no es reflexiva*

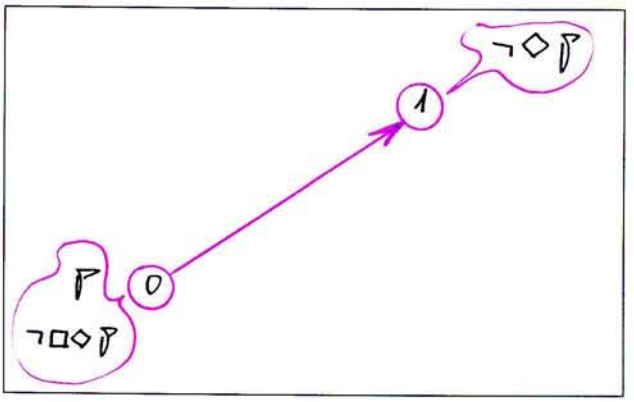


Figura 8.4: $\mathbf{R}$ *no es simétrica*

5. $5 := \Diamond\varphi \rightarrow \Box\Diamond\varphi$
   tomando: $\mathcal{A} = \left\langle \mathbf{W}, \mathbf{R}, \langle p^{\mathcal{A}} \rangle_{p \in \mathbf{ATOM}} \right\rangle$, $\mathbf{W} = \{0, 1, 2\}$, $\mathbf{R} = \{\langle 0, 1\rangle, \langle 0, 2\rangle\}$ y $p^{\mathcal{A}} = \{1\}$

8.3. Bisimulación

El poder expresivo de un lenguaje se mide por su capacidad de distinguir entre modelos distintos o, lo que es lo mismo, por cuales les resultan indistinguibles. En la lógica de primer orden vimos —el teorema 79— que la isomorfía produce invarianza; esto es, si $\mathcal{A}$ y $\mathcal{B}$ son dos sistemas isomorfos y $\mathbf{h}$ el isomorfismo, entonces las fórmulas de primer orden no nos permiten distinguir entre $\mathbf{x} \in \mathbf{A}$ y $\mathbf{h}(\mathbf{x}) \in \mathbf{B}$

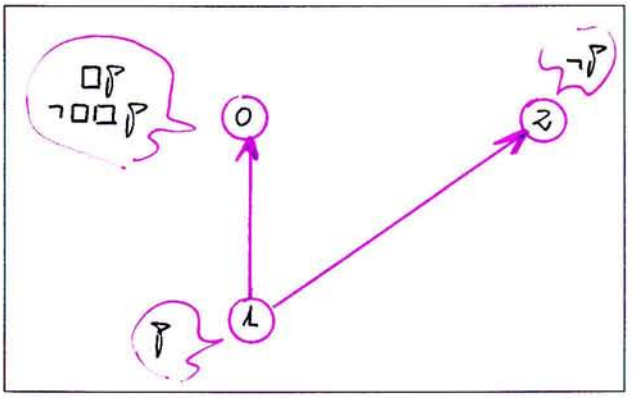


Figura 8.5: $\mathbf{R}$ *no es transitiva*

La verdad es que si juzgamos a nuestro lenguaje modal por su capacidad de caracterizar los modelos de Kripke, el lenguaje de primer orden discrimina mejor que él. Sin embargo, el aumento del poder expresivo se paga en primer orden con la decidibilidad, mientras que el paso de proposicional a modal no nos hacer perder la decidibilidad.

La equivalencia semántica adecuada al lenguaje modal es la *bisimulación*. Se entenderá mejor este concepto si consideramos que los modelos de Kripke representan procesos: sus mundos son estados que pueden poseer ciertas propiedades $p, q, r, ...$ siendo la relación de accesibilidad una transición entre estados.

Definición 278 *Dados los modelos* $\mathcal{A} = \left\langle \mathbf{W}, \mathbf{R}, \langle p^{\mathcal{A}} \rangle_{p \in \mathbf{ATOM}} \right\rangle$ *y* $\mathcal{B} = \left\langle \mathbf{W}^*, \mathbf{R}^*, \langle p^{\mathcal{B}} \rangle_{p \in \mathbf{ATOM}} \right\rangle$ *decimos que la relación* $\mathbf{E} \subseteq \mathbf{W} \times \mathbf{W}^*$ *es una bisimulación si siempre que* $\langle s, t \rangle \in \mathbf{E}$ *se cumple:*

1. s *y* t *satisfacen las mismas fórmulas atómicas:* $s \in p^{\mathcal{A}}$ *syss* $t \in p^{\mathcal{B}}$, *para cada* $p \in \mathbf{ATOM}$
2. *Si* $\langle s, u \rangle \in \mathbf{R}$ *entonces hay un* v *tal que* $\langle t, v \rangle \in \mathbf{R}^*$ *y* $\langle u, v \rangle \in \mathbf{E}$
3. *viceversa*

Gráficamente, lo que sucede es esto (véase la figura: 8.6)

Ejemplo 279 *Veámoslo en algunos casos.*

1. *Sea* $\mathcal{A} = \langle \{0\}, \{\langle 0, 0 \rangle\}, \langle \varnothing \rangle \rangle$ *y* $\mathcal{B} = \langle \{1, 2\}, \{\langle 1, 2 \rangle, \langle 2, 1 \rangle\}, \langle \varnothing \rangle \rangle$ *y sea* $\mathbf{E} = \varnothing$
2. *Sea* $\mathcal{A} = \langle \{1, 2, 3, 4\}, \{\langle 1, 2 \rangle, \langle 1, 3 \rangle, \langle 3, 4 \rangle\}, \langle \varnothing \rangle \rangle$ *y* $\mathcal{B} = \langle \{a, b, c\}, \{\langle a, b \rangle, \langle b, c \rangle, \langle a, c \rangle\}, \langle \varnothing \rangle \rangle$ *y*

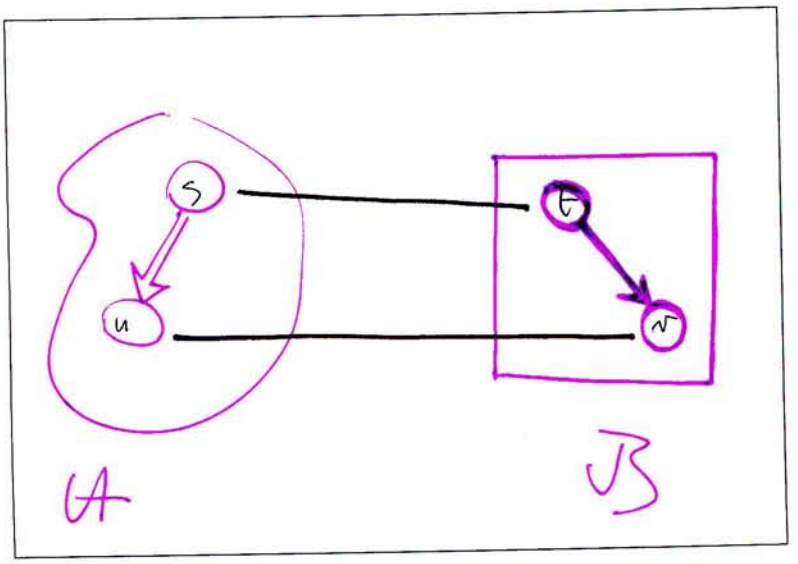


Figura 8.6: *Bisimulación*

sea $\mathbf{E} = \{\langle 1, a\rangle, \langle 3, b\rangle, \langle 4, c\rangle\}$
Esta relación $\mathbf{E}$ *es de bisimulación (ver figura: 8.7)*

3. *No pueden serlo* $\mathcal{A} = \langle \{1,2,3,4\}, \{\langle 1,2\rangle, \langle 2,3\rangle, \langle 2,4\rangle\}, \langle \{3\}, \{4\}\rangle\rangle$ *donde* $p^{\mathcal{A}} = \{3\}, q^{\mathcal{A}} = \{4\}$ *y*
$\mathcal{B} = \langle \{a,b,c,d,e\}, \{\langle a,b\rangle, \langle a,c\rangle, \langle b,d\rangle, \langle c,e\rangle\}, \langle \{d\}, \{e\}\rangle\rangle$
donde $p^{\mathcal{A}} = \{d\}, q^{\mathcal{A}} = \{e\}$*Esta relación* $\mathbf{E}$ *no es de bisimulación (ver figura: 8.8)*

Teorema 280 *Para cada bisimulación* $\mathbf{E}$ *entre* $\mathcal{A}$ *y* $\mathcal{B}$*, si* $\langle s,t\rangle \in \mathbf{E}$ *se cumple:*

$$\mathcal{A}, s \Vdash \varphi \text{ syss } \mathcal{B}, t \Vdash \varphi \tag{8.6}$$

Este teorema se puede usar para demostrar que son indefinibles en el lenguaje modal ciertas propiedades de la relación de accesibilidad, por ejemplo *irreflexividad*. Lo que haríamos es definir un modelo en donde la propiedad sea cierta en algún mundo y otro en donde no sea cierta, pero que estén relacionados mediante la relación $\mathbf{E}$.

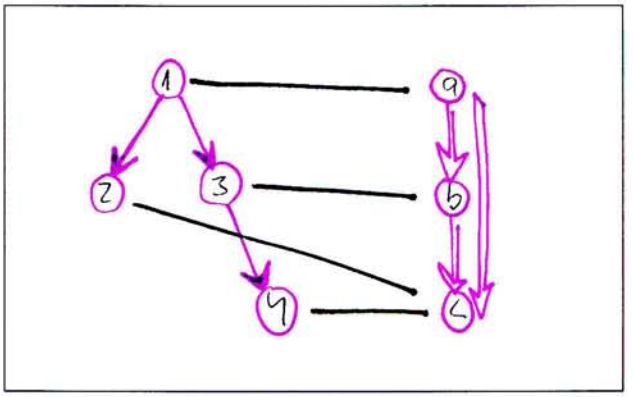


Figura 8.7: *Se dá bisimulación*

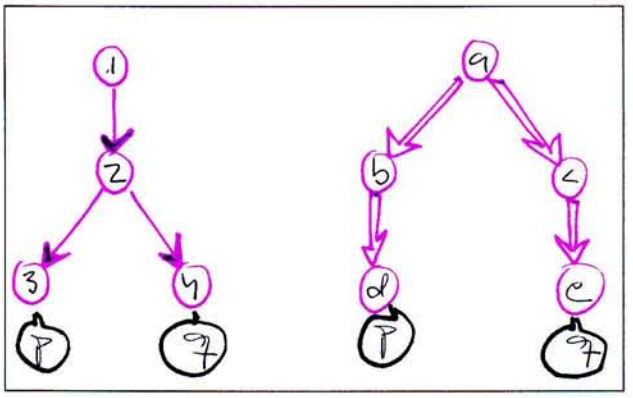


Figura 8.8: *No se dá bisimulación*

8.4. Propiedades de R

Hemos dicho que el lenguaje modal es especialmente adecuado para expresar propiedades de relaciones, veámoslo. Destaquemos algunas de las introducidas en la sección 5.2:

- *Reflexividad:* $\forall s \in \mathbf{W}\ (\langle s, s\rangle \in \mathbf{R})$
- *Simetría*[3]
- *Transitividad*[4]

Veamos algunos ejemplos (figura: 8.9)

[3] Definida en el capítulo 5.
[4] Definida en el capítulo 5.

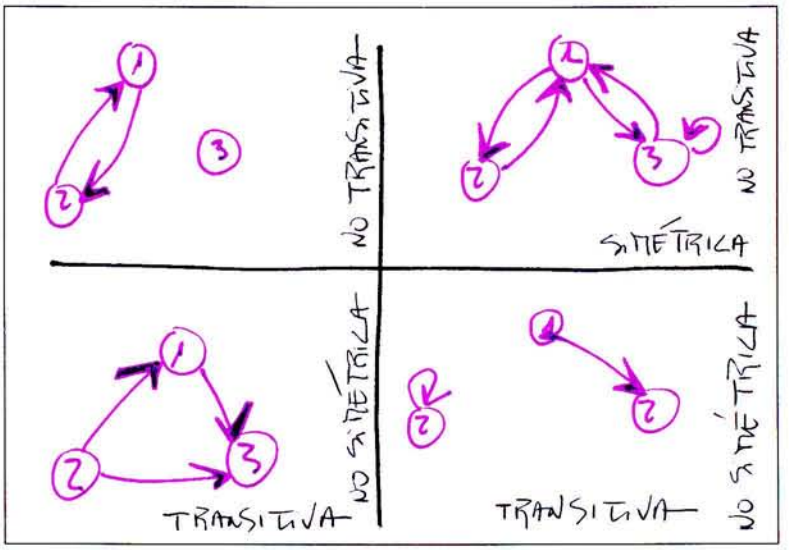


Figura 8.9: *Relaciones simétricas y relaciones transitivas*

- *Serial:* $\forall s\ \exists t\ \langle s,t\rangle \in \mathbf{R}$
- *Euclídea:* $\forall stu\ (\langle s,t\rangle \in \mathbf{R}\ \&\ \langle s,u\rangle \in \mathbf{R} \Rightarrow \langle t,u\rangle \in \mathbf{R})$

Veamos algunos ejemplos (figura: 8.10)

8.4.1. Propiedades de R y fórmulas modales

Con la semántica de modelos se pueden demostrar las siguientes implicaciones:

1. D es válida en la clase $\mathfrak{D}$ de modelos de Kripke con **R** serial
2. T es válida en la clase $\mathfrak{E}$ de modelos de Kripke con **R** reflexiva
3. 4 es válida en la clase $\mathfrak{F}$ de modelos de Kripke con **R** transitiva
4. B es válida en la clase $\mathfrak{G}$ de modelos de Kripke con **R** simétrica
5. 5 es válida en la clase $\mathfrak{H}$ de modelos de Kripke con **R** euclídea

Habiendo, sin embargo, modelos de estas fórmulas que no cumplen la condición asociada; esto es, se establece una correspondencia mediante un condicional, no un bicondicional.

Comentario 281 *El estudiar con detalle estas correspondencias es hoy día imprescindible para entender el papel central que la lógica clásica juega en este marasmo de sistemas que es la lógica actual. Lo abordamos decididamente en el capítulo 13.*

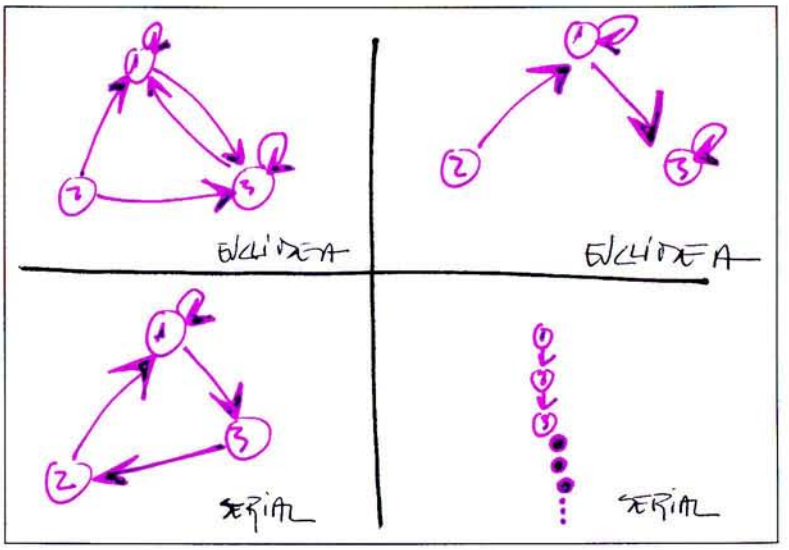


Figura 8.10: *Relaciones euclídeas y relaciones seriales*

Cuando se toma la semántica de marcos para nuestras fórmulas el vínculo entre propiedades de la relación de accesibilidad y las fórmulas pasa a ser bidireccional, pudiéndose demostrar:

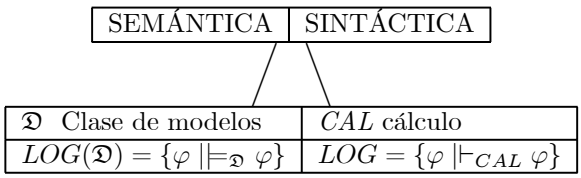
1. D es válida en la clase $\mathfrak{D}$ de marcos de Kripke syss $\mathbf{R}$ es serial
2. T es válida en la clase $\mathfrak{E}$ de marcos de Kripke syss $\mathbf{R}$ es reflexiva
3. 4 es válida en la clase $\mathfrak{F}$ de marcos de Kripke syss $\mathbf{R}$ es transitiva
4. B es válida en la clase $\mathfrak{G}$ de marcos de Kripke syss $\mathbf{R}$ es simétrica
5. 5 es válida en la clase $\mathfrak{H}$ de marcos de Kripke syss $\mathbf{R}$ es euclídea

Comentario 282 *Esta es la razón principal que justifica el cambio de la semántica de modelos a la de marcos. Sin embargo, la opción que yo propongo —en el comentario 387— nos permite mantener la primera, caracterizando en una teoría multivariada a la relación.*

8.5. Lógicas modales proposicionales

Hemos introducido ya el lenguaje modal e interpretado sus fórmulas usando la semántica de Kripke, pero todavía no hemos definido con precisión ninguna lógica modal. *¿Cómo se define?, ¿Qué es una lógica modal?*

En otro lugar —concretamente, en la sección 12.1— entraré en el debate sobre la naturaleza de los sistemas lógicos, ahora indicaré que hay dos procedimientos para definir una lógica modal: el semántico y el sintáctico.



Para no comprometerse con ninguno de estos planteamientos y permitir un acercamiento neutro al tema, definiremos una lógica modal como un conjunto de fórmulas modales que contiene a las tautologías (escritas modalmente, $\Box\varphi$ y $\Diamond\varphi$ cuentan como atómicas) y está cerrado bajo *modus ponens*.

Definición 283 *Una* ***lógica modal*** $\mathfrak{L}$ *es un conjunto tal que:*

$$\{\varphi \in FORM \mid\vdash_{PC} \varphi\} \subseteq \mathfrak{L} \subseteq FORM$$

$\mathfrak{L}$ *está cerrado bajo* MP
(es decir, $\alpha \in \mathfrak{L}$ & $\alpha \to \beta \in \mathfrak{L} \Rightarrow \beta \in \mathfrak{L}$*)*

Definición 284 *(Teoremas lógicos)* $\varphi \in \mathfrak{L}$ *syss* $\vdash_{\mathfrak{L}} \varphi$

Definición 285 *Deducibilidad a partir de hipótesis:* $\Gamma \vdash_{\mathfrak{L}} \varphi$
syss hay $\{\gamma_1, ..., \gamma_n\} \subseteq \Gamma$ *tal que*

$$(\gamma_1 \wedge ... \wedge \gamma_n) \to \varphi \in \mathfrak{L}$$

Se puede demostrar que el operador de consecuencia $\vdash_{\mathfrak{L}}$ así introducido verifica las propiedades consiguientes; esto es, las características de un operador clásico de consecuencia:

1. Monotonía: $\Gamma \subseteq \Delta$ & $\Gamma \vdash_{\mathfrak{L}} \varphi \Rightarrow \Delta \vdash_{\mathfrak{L}} \varphi$
2. MP Generalizado: $\Gamma \vdash_{\mathfrak{L}} \varphi$ & $\Gamma \vdash_{\mathfrak{L}} \varphi \to \psi \Rightarrow \Gamma \vdash_{\mathfrak{L}} \psi$
3. Deducción: $\Gamma \cup \{\varphi\} \vdash_{\mathfrak{L}} \psi \Rightarrow \Gamma \vdash_{\mathfrak{L}} \varphi \to \psi$

8.5.1. Lógicas modales normales

A las lógicas modales normales se las suele definir sintácticamente

Definición 286 *El cálculo* $\{K, Df_{\Diamond}\}_{(N)}$ *de la lógica* $\mathbf{K}$ *contiene todas las tautologías de la lógica clásica, la regla de* MP *y los siguientes axiomas y reglas:*

1. $K := \Box(\varphi \to \psi) \to (\Box\varphi \to \Box\psi)$
2. $Df_\Diamond := \Diamond\varphi \leftrightarrow \neg\Box\neg\varphi$
3. $(N) : \vdash \varphi \Rightarrow \vdash \Box\varphi$ *(Denominada regla de necesariedad)*

Definición 287 *La lógica* **K** *(denominada* ***lógica modal minimal****) es la menor lógica modal normal que contiene los esquemas axiomáticos* **K** *y* $Df_\Diamond$ *y está cerrada bajo* (N)

$$\mathbf{K} = \left\{\varphi \mid \vdash_{\{K, Df_\Diamond\}_{(N)}} \varphi\right\}$$

Otras lógicas modales normales se obtienen al añadirle alguno de los siguientes esquemas axiomáticos: D, T, B, 4 ó 5. Las más conocidas son:

1. **K***T*4 Esta lógica es el resultado de añadir a la lógica minimal los axiomas T y 4 (También conocida como $S4$)
2. **K***T*5 Esta lógica es el resultado de añadir a la lógica minimal los axiomas T y 5 (También conocida como $S5$)

Ejercicio 288 *Como ejercicio de deducción en la lógica* $S5$ *se puede demostrar:*
Todas las lógicas modales normales obtenidas añadiendo a **K** *algunos de los axiomas* D, T, B, 4 *y* 5 *están contenidas en* $S5$.

8.6. Completud y corrección

Tanto cuando se define la lógica como un conjunto de fórmulas válidas en una clase de modelos de Kripke, como cuando nos valemos de un cálculo, es preciso complementar la presentación con su otra dimensión semántica o sintáctica, según el caso.

Para demostrar la equivalencia entre la semántica y la sintaxis de una lógica modal se demuestran los teoremas de corrección y completud[5].

Definición 289 *Una lógica modal* B *es* ***correcta respecto de la clase de modelos*** $\mathfrak{D}$ *syss para cada* $\varphi \in \mathbf{FORM}$:

$$\vdash_B \varphi \Rightarrow \models_{\mathfrak{D}} \varphi$$

Definición 290 *Una lógica modal* B *es* ***completa respecto de la clase de modelos*** $\mathfrak{D}$ *syss para cada* $\varphi \in \mathbf{FORM}$:

$$\models_{\mathfrak{D}} \varphi \Rightarrow \vdash_B \varphi$$

Definición 291 *Una lógica modal* B ***está determinada por la clase de modelos*** $\mathfrak{D}$ *syss para cada* $\varphi \in \mathbf{FORM}$

$$\vdash_B \varphi \iff \models_{\mathfrak{D}} \varphi$$

[5] Una prueba alternativa de estos teoremas se establece en el entorno de la lógica multivariada como marco unificador [13.4].

8.6.1. Corrección

Teorema 292 *(corrección de* $\mathbf{K}$*). La lógica* $\mathbf{K}$ *es correcta en la clase formada por todos los modelos de Kripke.*

Demostración. *Hay que demostrar que todas las tautologías son válidas en la clase de todos los modelos de Kripke, que los esquemas axiomáticos* K *y* $Df_\Diamond$ *son válidos y que tanto* MP *como* N *preservan la validez.*

1. *Es claro que las tautologías son válidas, pues la interpretación de los conectores es la clásica.*

2. *Para ver que* $\vDash Df_\Diamond$ *tenemos que demostrar que si* $\mathcal{A}$ *es un modelo de Kripke cualquiera y* $\Im$ *su interpretación asociada, se cumple que*

$$\Im(\Diamond\varphi \leftrightarrow \neg\Box\neg\varphi) = \mathbf{W}$$

   Es decir, que

$$\Im(\Diamond\varphi) = \Im(\neg\Box\neg\varphi)$$

3. *Para ver que* $\vDash K$ *tenemos que demostrar que en toda interpretación* $\Im$ *sobre un modelo de Kripke* $\mathcal{A}$ *se cumple que*

$$\Im((\Box(\varphi \to \psi) \to (\Box\varphi \to \Box\psi)) = \mathbf{W}$$

   Es decir, que

$$\Im(\Box(\varphi \to \psi)) \subseteq \Im(\Box\varphi \to \Box\psi)$$

4. *Para demostrar que la regla* MP *preserva la validez hemos de ver que para todo modelo de Kripke* $\mathcal{A}$ *se cumple lo siguiente: Si* $\mathcal{A} \vDash \varphi \to \psi$ *y* $\mathcal{A} \vDash \varphi$ *entonces* $\mathcal{A} \vDash \psi$.

5. *Para demostrar que la regla* N *preserva la validez hemos de ver que para todo modelo de Kripke* $\mathcal{A}$ *se cumple lo siguiente: Si* $\mathcal{A} \vDash \varphi$ *entonces* $\mathcal{A} \vDash \Box\varphi$.

■

Corrección de otras lógicas modales normales

Para demostrar la corrección de las lógicas modales normales obtenidas con D, T, B, 4 y 5 usamos las propiedades asociadas —esto es, las especificadas en la sección 8.4—. Haciéndolo así es sencillo obtener lo siguiente:

Teorema 293 *La lógica* $\mathbf{K}D$ *es correcta en la clase* $\mathfrak{D}$ *de modelos de Kripke con* R *serial*

Teorema 294 *La lógica* $\mathbf{K}T$ *es correcta en la clase* $\mathfrak{E}$ *de modelos de Kripke con* R *reflexiva*

Teorema 295 *La lógica* $\mathbf{K}B$ *es correcta en la clase* $\mathfrak{F}$ *de modelos de Kripke con* R *simétrica*

Teorema 296 *La lógica* $\mathbf{K}4$ *es correcta en la clase* $\mathfrak{G}$ *de modelos de Kripke con* R *transitiva*

Teorema 297 *La lógica* $\mathbf{K}5$ *es correcta en la clase* $\mathfrak{H}$ *de modelos de Kripke con* R *euclídea*

Usando estos resultados demostramos la corrección de lógicas $\mathbf{K}S_1...S_n$

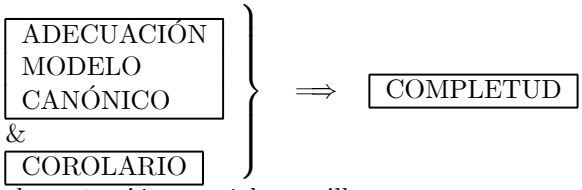
Teorema 298 *Corrección de* $\mathbf{K}S_1...S_n$
Demostración. *Sean* $S_1, ..., S_n$ *esquemas axiomáticos válidos, respectivamente en las clases de modelos* $\mathfrak{C}_1, ..., \mathfrak{C}_n$, *entonces la lógica* $\mathbf{K}S_1...S_n$ *es correcta en la clase de modelos* $\mathfrak{C}_1 \cap ... \cap \mathfrak{C}_n$.
Si $S_1, ..., S_n$ *son esquemas axiomáticos válidos, respectivamente, en las clases de modelos* $\mathfrak{C}_1, ..., \mathfrak{C}_n$, *entonces cada uno de los esquemas* S_i $(i \in \{1, ..., n\})$ *es válido en la clase de modelos* $\mathfrak{C}_1 \cap ... \cap \mathfrak{C}_n$. *Por otra parte, hemos visto que la lógica minimal es correcta en la clase de todos los modelos de Kripke. Por consiguiente, lo seguirá siendo en una clase más restringida de modelos.* ■

Comentario 299 *El teorema utilizado afirmaba que si una clase de modelos era, respectivamente, serial, reflexiva, transitiva, simétrica o euclídea, entonces dicha clase validaba, respectivamente los esquemas* $D, T, 4, B, 5$. *No demostramos (porque no es válido) el teorema con una doble implicación; es decir, que todo modelo de, respectivamente,* $D, T, 4, B, 5$ *sea, respectivamente, serial, reflexivo, transitivo, simétrico o euclídeo. Habitualmente sucede que si un modelo no es, respectivamente, serial, reflexivo, transitivo, simétrico o euclídeo, entonces fallará, respectivamente,* $D, T, 4, B, 5$. *Pero habrá que verlo en cada ocasión porque para la semántica de modelos esta implicación no es necesariamente el caso.*

8.7. Modelos canónicos: completud

Para demostrar los teoremas de completud y de equivalencia de las lógicas modales normales usamos este esquema (ver figura: 8.11):

Antes de comenzar con la demostración de la completud del cálculo comprobamos que el organigrama funciona; esto es,



La demostración es así de sencilla:

Supongamos que $\mathcal{A}_B \in \mathfrak{D}$ y que $\mathcal{A}_B \vDash \varphi \Longleftrightarrow \vdash_B \varphi$, para cada $\varphi \in \mathbf{FORM}$

Queremos demostrar que $\models_{\mathfrak{D}} \varphi \Rightarrow \vdash_B \varphi$. Supongamos que $\models_{\mathfrak{D}} \varphi$. Es decir, que $\mathcal{A} \vDash \varphi$, para cada $\mathcal{A} \in \mathfrak{D}$. Puesto que $\mathcal{A}_B \in \mathfrak{D}$, tenemos que $\vdash_B \varphi$

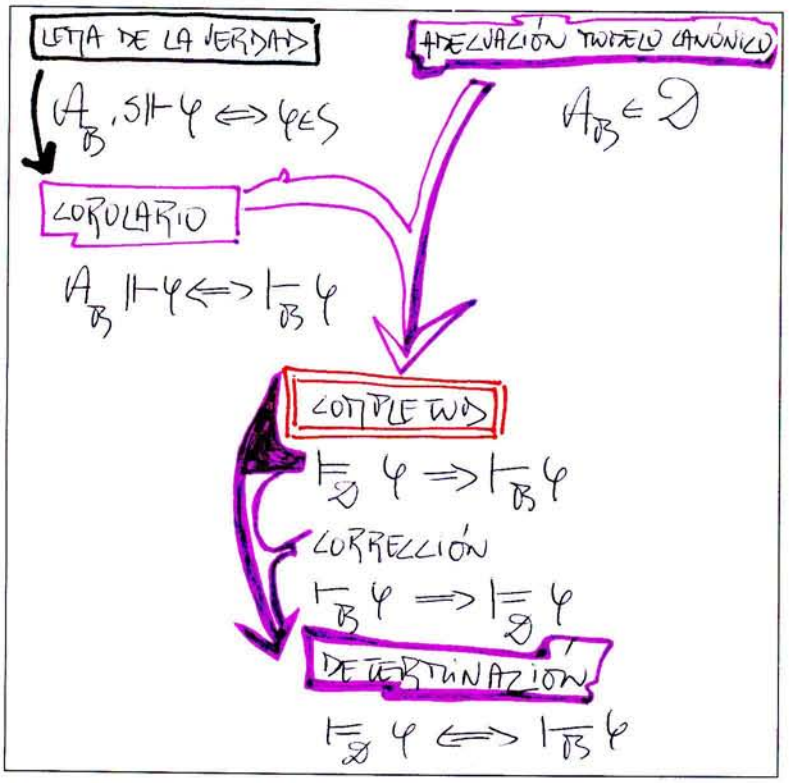


Figura 8.11: *Completud, modelo canónico*

8.7.1. Deducibilidad y consistencia

Definición 300 Γ *es* $B-$consistente *syss* $\Gamma \nvdash_B \bot$

Definición 301 Γ *es* $B-$contradictorio *syss no es* $B-$*consistente*

Utilizando la definición del operador de consecuencia y las definiciones previas se demuestra fácilmente que:

Proposición 302 *(MP generalizado)* $\Gamma \vdash_B \varphi \,\&\, \Gamma \vdash_B \varphi \to \psi \;\Rightarrow\; \Gamma \vdash_B \psi$

Proposición 303 *(Deducción)* $\Gamma \cup \{\varphi\} \vdash_B \psi \;\Rightarrow\; \Gamma \vdash_B \varphi \to \psi$

Proposición 304 $\Gamma \subseteq B \;\Longrightarrow (\Gamma$ *es* $B-$consistente $\iff B \neq \mathbf{FORM})$

Proposición 305 Γ *es* $B-$consistente $\iff$ *hay una fórmula* φ *tal que* $\Gamma \nvdash_B \varphi$

Proposición 306 Γ *es* $B-$consistente $\iff$ *no hay ninguna fórmula* φ *tal que* $\Gamma \vdash_B \varphi \wedge \neg\varphi$

Proposición 307 $\Gamma \vdash_B \varphi \iff \Gamma \cup \{\neg\varphi\}$ *no es* $B-$*consistente*

Proposición 308 Γ *es* $B-$consistente $\Longrightarrow$ *para cada* $\varphi : \Gamma \cup \{\varphi\}$ *o* $\Gamma \cup \{\neg\varphi\}$ *es consistente*

Definición 309 *Sea* $\Gamma \subseteq \mathbf{FORM}$. B *es* $B-$máximamente consistente $\iff$ Γ *es* $B-$consistente *y para cada* $\varphi \in \mathbf{FORM} : \varphi \in \Gamma$ *o* $\neg\varphi \in \Gamma$

Ejemplo 310 *Sea* $\mathcal{A} = \left\langle \mathbf{W}, \mathbf{R}, \langle p^{\mathcal{A}} \rangle_{p \in \mathbf{ATOM}} \right\rangle$ *un modelo de* B *y* $s \in \mathbf{W}$. *El conjunto* $\{\varphi \mid \mathcal{A}, s \models \varphi\}$ *es máximamente consistente.*

Ejercicio 311 *Sea* Γ *un conjunto* $B-$*máximamente consistente. Proponemos como ejercicios los siguientes:*

1. $\Gamma \vdash_B \varphi \implies \varphi \in \Gamma$
2. $\varphi \notin \Gamma \implies \Gamma \cup \{\varphi\}$ *es* $B-$*contradictorio*
   $(\Gamma \subseteq \Delta$ *y* Δ *es* $B-$*consistente* $\implies \Gamma = \Delta)$
3. $\neg\varphi \in \Gamma \iff \varphi \notin \Gamma$, *para cada* $\varphi \in FORM$
4. $B \subseteq \Gamma$
5. $\bot \notin \Gamma$
6. $\varphi \to \psi \in \Gamma \iff (\varphi \in \Gamma \Longrightarrow \psi \in \Gamma)$
7. $\varphi \wedge \psi \in \Gamma \iff (\varphi \in \Gamma$ *y* $\psi \in \Gamma)$
8. $\varphi \vee \psi \in \Gamma \iff (\varphi \in \Gamma$ *ó* $\psi \in \Gamma)$
9. $\varphi \leftrightarrow \psi \in \Gamma \iff (\varphi \in \Gamma \iff \psi \in \Gamma)$

8.7.2. Teorema de Lindenbaum

Como en la lógica clásica, se puede demostrar el teorema de Lindenbaum.

Teorema 312 *Todo conjunto consistente* Γ *puede extenderse a uno máximamente consistente* Δ

Demostración. Sea Γ un conjunto $B-$consistente y sea $\varphi_1, ..., \varphi_n, \cdots$ una enumeración de **FORM**

Definimos:

$\Delta_0 = \Gamma$

$$\Delta_{n+1} = \begin{cases} \Delta_n \cup \{\varphi_n\}, \text{ si } \Delta_n \vdash_B \varphi_n \\ \Delta_n \cup \{\neg\varphi_n\}, \text{ en caso contrario} \end{cases}$$

Hagamos $\Delta = \bigcup_{n \geq 0} \Delta_n$

Se puede demostrar que:

1. Δ_n es $B-$consistente, para cada $n \geq 0$
2. Para cada $\varphi \in \mathbf{FORM} : \varphi \in \Delta$ ó $\neg\varphi \in \Delta$ (no ambos)
3. $\Delta \vdash_B \varphi \implies \varphi \in \Delta$

■

8.7.3. Modelo canónico

Sea B una lógica modal normal, consistente. Llamaremos $\mathcal{A}_B$ al modelo canónico de B que definimos así:

$$\mathcal{A}_B = \left\langle \mathbf{W}_B,\ \mathbf{R}_B,\ \langle p^{\mathcal{A}_B} \rangle_{p \in \mathbf{ATOM}} \right\rangle$$

1. $\mathbf{W}_B = \{s \subseteq \mathbf{FORM} \mid s$ es máximamente $B-$consistente$\}$
2. $\mathbf{R}_B \subseteq \mathbf{W}_B \times \mathbf{W}_B$ se define así: $\langle s,t\rangle \in \mathbf{R}_B$ syss $\{\varphi \mid \Box\varphi \in s\} \subseteq t$
3. $p^{\mathcal{A}_B} = \{s \in \mathbf{W}_B \mid p \in s\}$, para todo $p \in \mathbf{ATOM}$

Vemos cómo en los modelos canónicos reaparecen las ideas de Carnap: el universo lo forman las descripciones de estados.

¿Y qué descripción mejor y más pormenorizada podríamos tener que la de un conjunto máximamente consistente?

Un conjunto tal contiene todas las sentencias capaces de caracterizar el modelo, recoge todas sus verdades. La relación de accesibilidad: *"t es una alternativa a s"* se establece cuando t satisface las verdades necesarias de s

Proposición 313 *Sea B una lógica modal normal, consistente. $\mathcal{A}_B$ su modelo canónico. Para cada $\varphi \in \mathbf{FORM}, s \in \mathbf{W}_B$:*
$\Box\varphi \in s \Longleftrightarrow$ (*para cada* $t \in \mathbf{W}_B : \langle s,t\rangle \in R_B \Longrightarrow \varphi \in t$])

Lema 314 ***(de la verdad)*** *Sea B una lógica modal normal, consistente. $\mathcal{A}_B$ su modelo canónico.*
Para cada $\varphi \in \mathbf{FORM}, s \in \mathbf{W}_B$ se verifica: $\mathcal{A}_B, s \models \varphi \iff \varphi \in s$

Corolario 315 *Sea B una lógica modal normal, consistente. $\mathcal{A}_B$ su modelo canónico. Para cada $\varphi \in \mathbf{FORM}$ se verifica: $\mathcal{A}_B \models \varphi \iff \vdash_B \varphi$*

Teorema 316 *(Completud de $\mathbf{K}$) La lógica $\mathbf{K}$ es completa en la clase de todos los modelos de Kripke. Para cada $\varphi \in \mathbf{FORM}$ se verifica: $\models \varphi \implies \vdash_K \varphi$*

Teorema 317 *(Completud de $\mathbf{K}D$) La lógica $\mathbf{K}D$ es completa en la clase $\mathfrak{D}$ de modelos de Kripke con $\mathbf{R}$ serial. Para cada $\varphi \in \mathbf{FORM}$ se verifica: $\models_{\mathfrak{D}} \varphi \implies \vdash_{KD} \varphi$ (basta mostrar la adecuación del modelo canónico; es decir, $\mathcal{A}_{\mathbf{K}D} \in \mathfrak{D}$)*

Teorema 318 *(Completud de $\mathbf{K}T$) La lógica $\mathbf{K}T$ es completa en la clase $\mathfrak{E}$ de modelos de Kripke con $\mathbf{R}$ reflexiva. Para cada $\varphi \in \mathbf{FORM}$ se verifica: $\models_{\mathfrak{E}} \varphi \implies \vdash_{\mathbf{K}T} \varphi$*

Teorema 319 *(Completud de $\mathbf{K}4$) La lógica $\mathbf{K}4$ es completa en la clase $\mathfrak{F}$ de modelos de Kripke con $\mathbf{R}$ transitiva. Para cada $\varphi \in \mathbf{FORM}$ se verifica: $\models_{\mathfrak{F}} \varphi \implies \vdash_{\mathbf{K}4} \varphi$*

Teorema 320 *(Completud de* $\mathbf{K}B$*) La lógica* $\mathbf{K}B$ *es completa en la clase* $\mathfrak{G}$ *de modelos de Kripke con* $\mathbf{R}$ *simétrica. Para cada* $\varphi \in \mathbf{FORM}$ *se verifica:* $\models_{\mathfrak{G}} \varphi \implies \vdash_{KB} \varphi$

Teorema 321 *(Completud de* $\mathbf{K}5$*) La lógica* $\mathbf{K}5$ *es completa en la clase* $\mathfrak{H}$ *de modelos de Kripke con* R *euclídea. Para cada* $\varphi \in \mathbf{FORM}$ *se verifica:* $\models_{\mathfrak{H}} \varphi \implies \vdash_{\mathbf{K}5} \varphi$

Teorema 322 *(Completud de* $\mathbf{K}S_1...S_n$*) Sean* $S_1, ..., S_n$ *esquemas axiomáticos válidos, respectivamente en las clases de modelos* $\mathfrak{C}_1, ..., \mathfrak{C}_n$*, entonces la lógica* $\mathbf{K}S_1...S_n$ *es completa en la clase de modelos* $\mathfrak{C}_1 \cap ... \cap \mathfrak{C}_n$*. Para cada* $\varphi \in \mathbf{FORM}$ *se verifica:* $\models_{\mathfrak{C}_1 \cap ... \cap \mathfrak{C}_n} \varphi \implies \vdash_{\mathbf{K}S_1...S_n} \varphi$

8.8. Apéndice: Tableaux para la lógica modal

Como se comentó con anterioridad en la sección 4.3 es fácil comprobar si una prueba hecha en el cálculo axiomático o en uno de deducción natural es correcta, pero hacerla no lo es en absoluto. Posiblemente el problema resida en la regla de *Modus Ponens* y sobretodo en la ausencia de la denominada *propiedad de la subfórmula*. Ni los sistemas axiomáticos ni los de deducción natural la tienen, pero hay otros sistemas muy sencillos que sí que la tienen: los de Resolución, los Sistemas de Gentzen y los Tableaux Semánticos de Smullyan.

Para las lógicas modales hay distintos tipos de tableaux. Hemos elegido uno que incluye nombres para los mundos posibles como parte del mecanismo deductivo, de forma que la relación de accesibilidad entre mundos queda reflejada en las características sintácticas de sus propios nombres. Lo que haremos es introducir *prefijos*.

La idea intuitiva es que un prefijo nombra un mundo posible. Si escribimos

$$\xi \ \varphi$$

siendo ξ un prefijo, queremos decir que φ es verdadera en el mundo cuyo nombre es ξ. Los mundos de un modelo pueden estar relacionados mediante la relación de accesibilidad, o no estarlo. El sistema de prefijos que elegiremos nos permitirá saber cuando dos mundos están relacionados, mediante ciertas características sintácticas de sus nombres.

Empezaremos por la lógica K, luego consideraremos otras lógicas modales.

8.8.1. Tableaux semánticos para la lógica K

Definición 323 *Un* ***prefijo*** *para* K *es una sucesión finita y no vacía de números naturales; por ejemplo,* $\langle 1, 3, 2, 1, 4 \rangle$*. El mundo real es* $\langle 1 \rangle$

Definición 324 *Una* ***fórmula con prefijo*** *es de la forma* $\xi \ \varphi$*, donde* φ *es una fórmula y* ξ *un prefijo.*

Si n es un número natural y ξ es un prefijo, ξn es el resultado de añadir n al final: por ejemplo, si $\xi = \langle 1,3,2,1,4\rangle$ y $n = 3$, entonces $\xi n = \langle 1,3,2,1,4,3\rangle$

Definición 325 *Decimos que un prefijo de la forma ξn es $\mathbf{K}-$accesible desde ξ.*

Un tableau tiene forma de árbol, donde cada nudo contiene una fórmula con prefijo. Daremos reglas de inicio del tableau, reglas de inferencia y reglas para terminar. Creo que resultará útil intercalar con la presentación de las reglas las razones que justifican la corrección de las mismas; servirá tanto de motivación intuitiva, como de justificación formal.

La definición de satisfacibilidad refleja la concepción que tenemos de un tableau como una disyunción de sus ramas y una rama como la conjunción de sus nudos.

Definición 326 *Un **tableau es** $K-$**satisfacible** si alguna de sus ramas es $\mathbf{K}-$satisfacible. Y una rama es $\mathbf{K}-$satisfacible si el conjunto de las fórmulas con prefijo que lo constituyen lo es. Finalmente, un conjunto de fórmulas Γ es $\mathbf{K}-$satisfacible si hay un modelo*

$$\mathcal{A} = \left\langle \mathbf{W}, \mathbf{R}, \langle p^{\mathfrak{M}} \rangle_{p \in ATOM} \right\rangle$$

y una función F de los prefijos de Γ en $\mathbf{W}$ tal que

1. *Si ξ, χ están en Γ y el prefijo χ es accesible desde ξ, entonces*

$$\langle F(\xi), F(\chi)\rangle \in \mathbf{R}$$

2. *Si ξ $\varphi \in \Gamma$, entonces $\mathcal{A}, F(\xi) \Vdash \varphi$.*

Es decir, un conjunto de fórmulas con prefijo es $\mathbf{K}-$satisfacible si describe parcialmente un mundo. Ahora que hemos introducido esta terminología, podemos presentar las reglas de los tableaux.

Reglas para K

Una demostración de φ comienza con $\langle 1\rangle$ $\neg\varphi$.

El procedimiento de los tableaux es refutativo: Para demostrar φ suponemos que puede haber algún modo de hacer que φ falle, y obtenemos de ello una contradicción y concluimos que φ. Es fácil ver que si φ no es $\mathbf{K}-$válida, el conjunto $\{\langle 1\rangle$ $\neg\varphi\}$ es $\mathbf{K}-$satisfacible y por lo tanto, el tableau con que comienza es $K-$satisfacible.

Tenemos también reglas de inferencia, que son reglas de extensión de sus ramas. Las clásicas son:

$\neg\neg$ **Doble negación** Una rama que contiene ξ $\neg\neg\varphi$ puede extenderse a ξ φ

$\neg\rightarrow$ **Negación del condicional** Una rama que contiene $\xi\ \neg(\varphi\rightarrow\psi)$ puede extenderse dos nudos con $\xi\ \varphi$ y $\xi\ \neg\psi$. (Es una regla de tipo α)

$\rightarrow$ **Condicional** Una rama que contiene $\xi\ (\varphi\rightarrow\psi)$ puede bifurcarse en dos ramas, con $\xi\ \neg\varphi$ y $\xi\ \psi$. (Es una regla de tipo β)

Estas reglas se presentan esquemáticamente en la tabla siguiente

$$\frac{\xi\ \neg\neg\varphi}{\xi\ \varphi} \qquad \frac{\xi\ \neg(\varphi\rightarrow\psi)}{\begin{array}{c}\xi\ \varphi\\ \xi\ \neg\psi\end{array}} \qquad \frac{\xi\ (\varphi\rightarrow\psi)}{\xi\ \neg\varphi \quad || \quad \xi\ \psi}$$

A continuación introducimos las reglas propiamente modales, pero para ello necesitamos introducir cierta terminología.

Definición 327 *Un prefijo ξ está restringido en una rama siempre que sea un segmento inicial (propio o impropio) de alguno de los prefijos de la rama.*

Definición 328 *Un prefijo está disponible en una rama si ocurre en la rama.*

$\Box$ **Regla de necesariedad** Una rama que contiene $\xi\ \Box\varphi$ puede extenderse a $\xi n\ \varphi$ para cualquier ξn disponible.

$\neg\Box$ **Negación de necesariedad** Una regla que contiene $\xi\ \neg\Box\varphi$ puede extenderse a $\xi n\ \neg\varphi$ siempre que ξn no esté restringido en la rama

Es decir,

$$\frac{\xi\ \Box\varphi}{\xi n\ \varphi}$$

para cualquier ξn disponible

$$\frac{\xi\ \neg\Box\varphi}{\xi n\ \neg\varphi}$$

siempre que ξn no esté restringido en la rama; es decir, no sea un segmento inicial propio o impropio de ninguno de los prefijos de la rama.

De nuevo la motivación informal está clara: Si $\Box\varphi$ es verdadera en un mundo cuyo nombre es ξ, entonces φ es verdadera en todo mundo accesible desde ξ; en particular, en ξn. También, si $\neg\Box\varphi$ es verdadera en el mundo cuyo nombre es ξ, $\neg\varphi$ debe ser verdadera en algún mundo accesible desde él; podemos darle un nombre, pero este mundo no debe estar condicionado, debe estar sin utilizar en el árbol. Es por ello por lo que la regla de la negación de la necesariedad respeta la condición de que el prefijo no debe estar restringido. Formalmente es fácil verificar que cada una de las reglas anteriores preserva la $\mathbf{K}-$satisfacibilidad. Es decir, si el conjunto de las fórmulas de la rama es $\mathbf{K}-$satisfacible antes de aplicarla, lo seguirá siendo después.

Esto completa la presentación de las reglas de extensión. Ahora indicamos las de terminación.

Definición 329 *Una rama de un tableau está cerrada si contiene* $\xi\ \varphi$ *y* $\xi\ \neg\varphi$ *para una fórmula cualquiera,* φ*. O si contiene* $\xi\ \bot$ *o* $\xi\ \neg\top$

Definición 330 *Un tableau está cerrado, si cada rama lo está.*

Definición 331 *Un tableau cerrado que empieza por* $\langle 1\rangle\ \neg\varphi$ *constituye una prueba (por refutación) de* φ*.*

Estar cerrado es la condición sintáctica que equivale en los árboles a ser contradictorio. Formalmente, un tableau cerrado no puede ser $\mathbf{K}-$satisfacible. Ahora bien, si φ no es $\mathbf{K}-$válida, entonces, como observamos antes, $\{\langle 1\rangle\ \neg\varphi\}$ es $\mathbf{K}-$satisfacible y por lo tanto nuestro árbol inicial será $\mathbf{K}-$satisfacible. Por contraposición diremos algo equivalente: Si hemos construido un tableau cerrado, φ debe ser $\mathbf{K}-$válida. Una prueba mediante tableaux garantiza la $\mathbf{K}-$validez; el procedimiento *es correcto.*

Reglas derivadas de inferencia En vez de tratar el resto de las conectivas y modalidades como definidas en términos de $\neg$, $\rightarrow$ y de $\Box$, es más cómodo proporcionar reglas derivadas para ellas.

$$\frac{\xi\ (\varphi\wedge\psi)}{\begin{array}{c}\xi\ \varphi\\ \xi\ \psi\end{array}} \qquad \frac{\xi\ \neg(\varphi\wedge\psi)}{\xi\ \neg\varphi \quad || \quad \xi\ \neg\psi}$$

$$\frac{\xi\ \neg(\varphi\vee\psi)}{\begin{array}{c}\xi\ \neg\varphi\\ \xi\ \neg\psi\end{array}} \qquad \frac{\xi\ (\varphi\vee\psi)}{\xi\ \varphi \quad || \quad \xi\ \psi}$$

$$\frac{\xi\ (\varphi\leftrightarrow\psi)}{\begin{array}{c}\xi\ (\varphi\rightarrow\psi)\\ \xi\ (\psi\rightarrow\varphi)\end{array}} \qquad \frac{\xi\ \neg(\varphi\leftrightarrow\psi)}{\xi\ \neg(\varphi\rightarrow\psi) \quad || \quad \xi\ \neg(\psi\rightarrow\varphi)}$$

$$\frac{\xi\ \neg\Diamond\varphi}{\xi n\ \varphi} \quad \text{para cualquier } \xi n \text{ disponible}$$

$$\frac{\xi\ \Diamond\varphi}{\xi n\ \neg\varphi} \quad \text{siempre que } \xi n \text{ no esté restringido en la rama}$$

—es decir, no sea un segmento inicial propio o impropio de ninguno de los prefijos de la rama—

Completud

Voy a señalar simplemente la idea general de una demostración de completud, que es sencilla.

Suponed que intentamos construir una demostración de φ. Empezamos el tableau con $\langle 1\rangle \ \neg\varphi$, entonces aplicamos las reglas del tableau. Si lo hacemos de una forma consistente, cada regla que puede ser aplicada lo habrá sido. No es difícil indicar un procedimiento que asegure que no se cometen errores. Imaginad que el procedimiento no termina en un tableau cerrado. Sea θ una rama no cerrada. Por construcción, en la rama θ cada regla aplicable lo ha sido.

Construyamos un modelo

$$\mathcal{A} = \left\langle \mathbf{W}, \mathbf{R}, \langle p^{\mathcal{A}} \rangle_{p \in ATOM} \right\rangle$$

como sigue:

$\mathbf{W}$ es el conjunto de los prefijos de θ. $\langle \xi, \chi\rangle \in R$ syss χ es $\mathbf{K}-$accesible desde ξ. Hagamos $p^{\mathcal{A}} = \{\xi \mid \xi \ p \text{ está en } \theta\}$. Esto determina un modelo $\mathcal{A}$. Un argumento inductivo prueba que si $\xi \ \varphi$ está en θ entonces

$$\mathcal{A}, \xi \Vdash \varphi$$

Puesto que $\langle 1\rangle \ \neg\varphi$ está en θ (es su primer punto), $\mathcal{A}, \langle 1\rangle \Vdash \neg\varphi$. Por consiguiente, φ no es válida.

Convirtiendo el argumento anterior por contraposición obtenemos: Si φ es $\mathbf{K}-$válida, entonces φ tiene una prueba mediante tableaux. De hecho, se obtiene una prueba mediante cualquier tableau correcto que se construya. Hay que señalar que el método anterior proporciona un procedimiento de decisión para la lógica proposicional, $\mathbf{K}$.

8.8.2. Lógica $S4$

Resulta fácil extender las ideas anteriores de forma que sirvan para otras lógicas modales. Por ejemplo, obtenemos la lógica $S4$ manteniendo todo el sistema anterior de la lógica $\mathbf{K}$ y modificando las reglas de los operadores modales, que es sustituida por la siguiente:

Si $\xi \ \Box\varphi$ está en una rama, entonces $\chi \ \varphi$ puede ser añadido para cualquier prefijo χ que tenga a ξ como segmento inicial, propio o impropio (la regla derivada para $\neg\Box$ debe modificarse de forma similar)

Ejemplo 332 *En este sistema demostramos que* $\vdash_{S4} \Box p \to \Box\Diamond\Box\Diamond p$ *así:*

$\langle 1\rangle \quad \neg(\Box p \rightarrow \Box\Diamond\Box\Diamond p)$
$\langle 1\rangle \quad \Box p$
$\langle 1\rangle \quad \neg\Box\Diamond\Box\Diamond p$
$\langle 11\rangle \quad \neg\Diamond\Box\Diamond p$
$\langle 11\rangle \quad \neg\Box\Diamond p$
$\langle 111\rangle \quad \neg\Diamond p$
$\langle 111\rangle \quad \neg p$
$\langle 111\rangle \quad p$
$\times$

8.8.3. Otros sistemas modales

Para poder presentar sistemas modales para la variedad de lógicas modales proposicionales que hemos estado estudiando, es conveniente introducir una terminología nueva

Recordad que en la lógica $\mathbf{K}$ decíamos que un prefijo χ era accesible desde ξ si χ era el resultado de añadir a ξ un número adicional al final. Ahora queremos considerar otras relaciones de accesibilidad sobre prefijos, como hicimos antes con $S4$.

Definición 333 *La relación de accesibilidad sobre prefijos satisface:*

1. *La condición general: si ξn es accesible desde ξ, para cualquier n*
2. *La condición inversa: Si ξ es accesible desde ξn, para todo n*
3. *La condición de identidad (reflexividad): si ξ es accesible desde sí mismo*
4. *La condición de transitividad: si χ es accesible desde ξ, siempre que ξ sea un segmento inicial propio de χ*
5. *La condición universal (total): si cualquier prefijo es accesible desde cualquier otro.*

Observad que la condición de transitividad incluye la condición general como un caso particular: Para cada lógica se definirá una noción correspondiente de accesibilidad de prefijos

LÓGICA	CONDICIÓN DE ACCSESIBILIDAD
$\mathbf{K}$, D	general
T	general, identidad
$\mathbf{K}B$, DB	general, inversa
B	general, identidad, inversa
$\mathbf{K}4$, $D4$	general, transitividad
$S4$	general, identidad, transitividad
$S5$	universal

Definición 334 *Un prefijo χ es una extensión simple de ξ si χ es ξ con un sólo número añadido al final.*

Daremos las reglas básicas para el operador de necesariedad $\Box$ las del operador de posibilidad $\Diamond$ tienen una formulación similar

$\neg\Box$ **Negación de necesariedad** Una rama que contiene ξ $\neg\Box\varphi$ puede extenderse a χ $\neg\varphi$ siempre que χ sea una extensión simple de ξ, y no esté restringido en la rama

$\Box$ **Regla de necesariedad** Una rama que contiene ξ $\Box\varphi$ puede extenderse a χ φ para cualquier χ accesible desde ξ siempre que:

1. para las lógicas **K**, **K**B y **K**T, el prefijo χ debe estar disponible en la rama.
2. para las lógicas D, T, DB, B, $D4$, $S4$ y $S5$, el prefijo χ debe ser una extensión simple, no restringida de ξ o estar disponible en la rama.

8.8.4. Sistema para la lógica $S5$

Para la lógica modal $S5$ la relación de accesibilidad ha degenerado en la relación universal (de trivialidad, pues todos están relacionados con todos) y por consiguiente se puede abandonar el sistema de prefijos como sucesiones finitas e introducir otros más simples. Para esta lógica, pero sólo para ella, podemos hacer que estos prefijos sean números naturales. Las reglas adecuadas para los operadores modales se introducen a continuación:

$\neg\Box$ **Negación de necesariedad** Una rama que contiene n $\neg\Box\varphi$ puede extenderse a k $\neg\varphi$ siempre que k sea nuevo en la rama.

$\Box$ **Regla de necesariedad** Una rama que contiene n $\Box\varphi$ puede extenderse a k φ para cualquier número natural k

8.8.5. Prueba a partir de hipótesis

No hemos hablado de pruebas a partir de hipótesis, pero se puede hacer fácilmente. Si L es cualquiera de las lógicas antes mencionadas, entonces

$$\Gamma \vdash_L \Omega \to \varphi$$

si y sólo si hay un tableau cerrado para $\langle 1\rangle$ $\neg\varphi$, usando las reglas de los tableaux antes mencionadas, junto a las siguientes:

Regla global ξ γ puede añadirse al final de cualquier rama, para cualquier $\gamma \in \Gamma$

Regla local $\langle 1\rangle$ ψ puede añadirse al final de cualquier rama, para cualquier $\psi \in \Omega$,

Bibliografía

[1] van Benthem, J.F.A.K. [1975]. *"A Note on Modal Formulas and Relational Properties"*. **JSL** 40 : 55-58.

[2] van Benthem, J.F.A.K. [1976]. *"Modal Reduction Principles"* **JSL** 41 : 301-312.

[3] van Benthem, J.F.A.K. [1978]. *"Two Simple Incomplete Modal Logics"*. **Theoria** 44. 25-37.

[4] van Benthem, J. [1988] *A manual of Intensional Logic*, CSLI, Stanford. USA

[5] Boolos, G. [1979]. *The Unprovability of Consistency.* Cambridge University Press. Cambridge.

[6] Bull, R.y Segerberg, K. [2001] *"Basic Modal Logic"* en pp. 1-88.In Handbook of Philosophical Logic, Vol. II (Eds, Gabbay, D. and Guenther, F.) Reidel, Dordrecht,

[7] Chellas, B. [1980] *Modal Logic: an Introduction*, Cambridge University Press, Cambridge. U.K.

[8] Fine, K. [1980 y 1981] *"First-Order Modal Theories"* —I Sets, II Propositions, III Facts—. Parte I en **Nous** 15, 1981, 177-205; Parte II en **Studia Logica** 39, 1980, 159-202; Parte III en **Synthese,** 1981, 43-122.

[9] Fine, Kit. [1974] *"An Incomplete Logic Containing S4"*. **Theoria** 60 : 23-29.

[10] Fine, K. [1978, 81]. *"Model Theory for Modal Logic"*, Partes I-III. Part I y II en **Journal of Philosophical Logic** 7, 1978, 125-156, 277-306; Parte III en *Journal of Philosophical Logic* 10, 1981, 293-307.

[11] Fine, K.[1975]. *"Vagueness, Truth and Logic"*. **Synthese** 30, 265-300.

[12] Fitch, F.B. [1973]. *"A Correlation Between Modal Reduction Principles and Properties of Relations"*. **Journal of Philosophical Logic** 2. 97-101.

[13] Fitting, M. [1993] *"Basic Modal and Temporal Logics"*. In Handbook of Logic in Artificial Intelligence and Logic Programming(Eds, Gabbay, D., Hogger, C. J. and Robinson, J. A.) Oxford University Press, . In Handbook of

Logic in Artificial Intelligence and Logic Programming, edited by D. Gabbay, C.J. Hogger and J.A. Robinson: Oxford University Press, 1993.

[14] Fitting, M. [1983]. *Proof Methods for Modal and Tense Logics*. Reidel. Dordrecht.

[15] Gabbay, D y Guenthner, F. [2001]. *Handbook of Philosophical Logic* 2^{nd} edition. Kluwer Academic. Holanda. vol I-IV

[16] Gabbay, D. [1976]. *Investigations in Model and Tense Logics, with Applications to Problems in Linguistics and Philosophy.* . Reidel. Dordrecht.

[17] Gabbay, D., C.J. Hogger, and J.A. Robinson, eds. [1993]. *Handbook of Logic in Artificial Intelligence and Logic Programming*. Oxford University Press, Oxford.

[18] Gamut, L. T. F. [1991]. *Logic, Language and Meaning.* vol 2, Intensional Logic and Logical Grammar. The University of Chicago Press. London. U.K.

[19] Gardies, J.L. [1979]. *Lógica del Tiempo.* Ed. castellana de J. Ordóñez y P. Martínez-Freire. Paraninfo. Madrid, 1979.

[20] Goldblatt, R. [1986]. *Logic of Time and Computation*, Center for the Study of Languege and Information. Stanford. USA

[21] Goldblatt, R. [1980]. *"Diodorean Modality in Minkowski Space-Time".* **Studia Logica** 39, 219-236.

[22] Haack, S. [1978]. *Filosofía de las Lógicas*. Ed. castellana de Amador Antón, 1982. ed. Madrid: Cátedra.

[23] Hintikka, J. [1969]. *Models for Modalities*. Reidel. Dordrecht.

[24] Hintikka, J. [1975]. *The Intentions for Intentionality and Other New Models for the Modalities.* Reidel. Dordrecht.

[25] Huertas, A. [1994]. *Modal Logic and Non-Classical Logic*. Universidad de Barcelona, tesis doctoral.

[26] Hughes, G. E. and Cresswell, M. J. [1968 (reimpreso con correcciones 1972)] *An introduction to modal logic*, Methuen, London. U.K.

[27] Hughes, G. E. and Cresswell, M. J. [1984] *A Companion to Modal Logic*, Methuen, London. U.K.

[28] Jansana, R. [1990]. *Una Introducción a la Lógica Modal.* Tecnos. Madrid.

[29] Kneale, W, y M. [1961]. *El Desarrollo de la Lógica*. Trducido por Javier Muguerza, 1972, Colección Estructura y Función. Tecnos. Madrid.

[30] Kripke, Saul A. [1959]. *"A completeness theorem in modal logic".* J. Symbolic Logic 24, 1-14.

[31] Kripke, Saul A. [1963]. *"Semantical analysis of modal logic I. Normal modal propositional calculi.".* **Zeitschrift fur mathematische Logik und Grundlagen der Mathematik.** 9. 67-96.

[32] Kripke, Saul A. [1963]. *"Semantical Considerations on Modal Logic". Acta Philosophica Fennica* 16 : 83-94.

[33] Lemmon, E.J., and Dana S. Scott. [1977]. *"The 'Lemmon Notes': an Introduction to Modal Logic.".* En **American philosophical quarterly Monograph Series**, edited by Nicholas Rescher. Basil Blackwell. Oxford.

[34] Manzano, M. [1996] *Extensions of First Order Logic*, Cambridge University Press, Cambridge, U.K.

[35] Mints, G. [1992]. *A short introduction to modal logic.* CSLI, Stanford. USA

[36] Prior, A. N (ed). [1976]. *Historia de la Lógica.* Translated by Amador Antón y Esteban Requena. Tecnos. Madrid.

[37] Sahlqvist, H. [1973]. *"Completeness and Correspondence in the First and Second Order Semantics for Modal Logic".* University of Oslo, .

[38] Sahlqvist, H. 1975 *"Completeness and Correspondence in the First and Second Order Semantics for Modal Logic."* In Proceedings of the 3rd Scandinavian Logic Symposium, 110-143: North-Holland, .

[39] Segerberg, K. [1971]. *An Essay in Classical Modal Logic.* Filosofiska Studier. Uppsala University.

[40] Thomason, R. [1984]. *"Combinations of Tense and Modality".* En [15]

[41] Thomason, S.K. [1974]. *"An Incompleteness Theorem in Modal Logic".* **Theoria** 60. 30-34.

Capítulo 9

Lógica Dinámica

9.1. Lógica de Hoare-Floyd

En informática los objetos de interés prioritario son los programas.

¿Qué es un programa?

Un programa o algoritmo es un procedimiento efectivo que nos permite calcular ciertos valores a partir de ciertos datos. Por ejemplo, se puede escribir un programa que dado un cierto n calcula el valor $F(n)$ de la serie de Fibonacci.

¿Cómo sabremos que el programa es correcto, más allá de las pocas —o muchas— comprobaciones que podamos hacer?

La idea es crear un formalismo externo que nos permita verificar que el programa realiza las operaciones deseadas.

En el año 1969 Hoare introdujo un método axiomático para probar la corrección de programas, basado en un método anterior, el de Floyd, que ha tenido un gran impacto en el diseño y verificación de programas. También ha sido usado como un medio de especificar la semántica de los lenguajes de programación.

Semántica de programas

En lógica clásica —y muy especialmente en teoría de modelos— se parte de una realidad matemática conocida y para hablar de ella se introduce un lenguaje formal. El puente entre el lenguaje formal y la realidad matemática es la noción semántica de verdad y, basado en ella, la noción de consecuencia. El cálculo deductivo, un conjunto de reglas de deducción, es una réplica mecanizable del concepto intuitivo de consecuencia.

En el caso de la semántica de los lenguajes de programación, pensad que a diferencia de lo que sucede en la lógica clásica, especialmente si se la mira desde la perspectiva de la teoría de modelos, el formalismo y sus reglas constituyen la realidad inmediata y en lo que hay que aplicarse es en buscarle una semántica, unos modelos. Es evidente que el modelo real de lo que sucede, el *«mundo»* al que se refieren las fórmulas de nuestros programas no es otro que el constituido

por la *CPU*, su memoria y registros, pero este *«mundo»* es de una complejidad tal que rebasa los límites de un análisis matemático riguroso. A causa de esta complejidad, se han hecho ciertos intentos de recurrir a la teoría de sistemas , considerándose el holismo como la postura filosófica a adoptar[1].

Las lógicas de programas, al menos en sus origen, pretendían resolver el problema de la semántica de programas y, por consiguiente, el de las expresiones que expresan propiedades de programas. Para los filósofos esta situación —la de encontrarnos con unos sistemas formales de reglas y axiomas de un lenguaje de nuestra invención, pero sin semántica precisa— no nos resulta del todo ajena ya que plantea similitudes con la de la lógica modal en la que la definición de diversos sistemas modales precedió a la definición precisa de su semántica. No deja de ser curioso que, aunque fuera por otros motivos, la lógica de programas recurriera a la modal para solventar sus problemas.

El significado o la semántica de un lenguaje de programación es un asunto mucho más complicado que su sintaxis y rápidamente planteó preguntas acerca de la fundamentación del software.

Sintaxis, semántica y cálculo

Una de las cuestiones primordiales era la de la correción parcial de un programa S respecto del input p y el output q, expresada mediante la fórmula $\{p\}\ S\ \{q\}$. Intuitivamente esta fórmula expresa que si p es verdadera antes de ejecutar el programa S y si el programa S termina, entonces q es verdad tras la ejecución de S. Ejemplos sencillos de fórmulas de corrección de programas son:

$$\{x = 0\}\ x : x + 1\ \{x = 1\}$$

$$\{x \geqq 0\}\ WHILE\ x \succ o\ DO\ x : x - 1\ OD\ \{x = 0\}$$

Tres preguntas fundamentales se plantean:

1. *¿Cuál es la sintaxis de los lenguajes en donde escribimos* S, p *y* q*?*
2. *¿Cuál es el significado de la fórmula* $\{p\}\ S\ \{q\}$*?, ¿cuándo decimos que esta fórmula es verdadera?*
3. *¿Podemos encontrar cálculos adecuados en los que demostrar* $\{p\}\ S\ \{q\}$*?*

La primera pregunta suele contestarse diciendo que p y q son fórmulas de primer orden escritas en el lenguaje de la aritmética. Los programas S se formulan en un lenguaje que suele incluir asignaciones —por ejemplo; $x := x + 1$—, composición secuencial

$$S_1; S_2$$

condicionales

$$IF\ b\ THEN\ S_1\ ELSE\ S_2\ FI$$

[1] Ver, por ejemplo [14].

programas del tipo while

$$WHILE\ b\ DO\ S\ OD$$

y otro tipo de construcciones tales como procedimientos recursivos, programas no deterministas, etc.

Para responder a la segunda hay diversos métodos. Hemos dicho que podemos considerar que el computador real —su *CPU*, memoria, etc.— es el universo en donde interpretar la fórmulas $\{p\}\ S\ \{q\}$. Puesto que esto es imposible a causa de la complejidad del sistema, se plantean varias soluciones. La primera es reemplazar el computador real por la máquina abstracta; es decir, un objeto matemático con estados, estructura de control, etc. La semántica así obtenida se denomina *operativa* y su característica es que aparecen secuencias de computación; esto es, secuencias de estados que describen en la máquina abstracta la ejecución del programa. Las ventajas de este método sobre el *denotativo* es que se mantiene próxima a la semántica intuitiva. Su desventaja es que no es suficientemente abstracta y es con frecuencia tediosa.

En la semántica denotacional se evita el uso de máquinas abstractas empleándose un modelo matemático con conjuntos, funciones y operadores (funciones de funciones); esto es, un modelo extensional en donde no aparecen secuencias de computación. Habitualmente estos modelos son finitos o contienen el modelo estándar de los naturales, son una extensión suya. Es decir, no es relevante el significado de las fórmulas en modelos arbitrarios. Los programas se interpretan como funciones (o relaciones) entre estados iniciales y finales sin usar estados intermedios y las funciones de funciones sirven para interpretar los operadores programas.

La tercera de las cuestiones es cómo generar cálculos en los que derivar fórmulas de corrección parcial,

$$\{p\}S\{q\}$$

Por ejemplo, considerad que el lenguaje de los programas incluye: (1) *asignaciones* $x =: s$ —donde s es un término de la aritmética—, (2) *composición secuencial* $S_1; S_2$ y (3) programas tipo *while* $WHILE\ b\ DO\ S\ OD$.

Un cálculo deductivo para este lenguaje incluirá el axioma.

$$\left\{p\frac{t}{x}\right\}\ x =: t\ \{p\}$$

la regla de la composición secuencial de programas

$$\frac{\{p\}\ S_1\ \{q\} \qquad\qquad \{q\}\ S_2\ \{r\}}{\{p\}\ S_1; S_2\ \{r\}}$$

y la regla de programas tipo while

$$\frac{\{p \wedge b\}\ S\ \{p\}}{\{p\}\ WHILE\ b\ DO\ S\ OD\ \{p \wedge \neg b\}}$$

Enfoques de este tipo son los que fructificaron a raíz de los trabajos de Floyd y de Hoare a finales de los años sesenta. El que ellos iniciaron es posiblemente el programa de investigación en informática teórica más importante que ha existido, ha sido enormemente prolífico y son incontables los métodos —lógicas, se suelen llamar— de corrección de programas que han aflorado pues, frecuentemente, se diseñaba una lógica específica para cada tipo de programa —asignaciones, composición, programas tipo while, etc.— Desde los métodos particulares y relativamente rupestres, hasta las lógicas dinámicas o las temporales, todos pueden considerarse enmarcados en el programa de Hoare-Floyd.

Su programa de investigación está basado en la lógica formal y tiene la pretensión de encarar los fundamentos de la programación —hay quienes lo comparan al programa de Hilbert, y no maliciosamente, pensando en su irrealizabilidad— . Su propuesta suele presentarse diciendo que definen la semántica de los lenguajes de programación axiomáticamente; esto es, mediante unas lógicas de corrección de programas. En estas lógicas el comportamiento de los programas se expresa mediante fórmulas adecuadas y se usan los axiomas y reglas del cálculo de la lógica creada para demostrar la corrección de los programas, como un teorema de cálculo introducido.

El cálculo retorna al software como un programa verificador de programas para así ayudar al programador a comprobar si su programa original, el escrito primitivamente en lenguaje de programación, se comporta como se esperaba. De este programa de investigación han surgido ideas muy interesantes, útiles y bonitas sobre construcción de lenguajes de programación lógica, y sobre especificación y verificación de software. También han contribuido notablemente a fomentar el interés por los métodos de la lógica formal en el diseño del software.

Metateoría de las lógicas de programas

A mediados de la década de los setenta se empezaron a estudiar las propiedades metalógicas de estas lógicas, se intentaron demostrar teoremas de corrección y completud para ellas —no son propiamente teoremas de completud, pues no se prueba para todas las fórmulas y no interesan todos los modelos—. Enseguida se advirtió que el denominado método de Floyd-Hoare, era incompleto en el sentido de que no todas las fórmulas que expresan corrección parcial de programas —que sean verdaderas en los modelos estándar— podían derivarse con las reglas del método (*lógica*) propuesto. Esto originó una frenética actividad investigadora para tratar de solucionar el problema. En unos casos se aumentaba la potencia del cálculo, permitiendo pruebas de longitud infinita, en otros se añadían modelos de computación no estándar, manteniéndose la exigencia de que las pruebas fuesen finitas.

9.2. Introducción a la lógica dinámica

Heredera del método de corrección de programas de Hoare-Floyd que acabamos de ver, pero con la pretensión de proporcionar no eso sólo, sino toda

una lógica en donde expresar y demostrar un gran abanico de propiedades de programas, nació a mediados de los setenta, la *lógica dinámica*. Hay una versión proposicional, una de primer orden y una de primer orden interpretada[2].

En esta introducción la analizaremos de estas tres formas:

1. Como una de las lógicas de programas, planteando los objetivos de este área y diciendo por consiguiente para qué sirve

2. Como una extensión de la lógica modal; ubicándola, pues, históricamente

3. Directamente, presentando su lenguaje y su semántica

9.2.1. La lógica dinámica es una de las lógicas de programas

Como he dicho anteriormente, las lógicas de programas se crean para expresar y probar propiedades de los programas de computación. Nos interesan las propiedades de *corrección*, *equivalencia de programas* y la de *tener fin*, entre otras. Conviene también tener un cálculo deductivo para verificar nuestros razonamientos sobre programas.

La filosofía de esta lógica se suele resumir diciendo que un programa es un objeto dinámico, capaz de hacer pasar al computador de un estado a otro. Un estado puede concebirse como el contenido de los registros de memoria usados por el programa.

La idea fundamental de la lógica de programas es que como resultado de la ejecución de un programa se altera el valor de verdad de las fórmulas. El caso intuitivamente más sencillo es cuando tenemos variables y asignaciones (*un programa*). Una fórmula puede ser verdadera para ciertos valores, pero si estos valores se modifican mediante asignaciones en el transcurso del programa, la fórmula posiblemente modificará su valor de verdad. Cuando manejamos programas es conveniente usar variables que puedan ir tomando diversos valores sobre ciertos universos en cualquier momento de la computación; a diferencia de lo que sucede en matemáticas, una misma variable puede tomar distintos valores en el transcurso del programa y, en particular , distinto valor inicial y final. Por consiguiente, el valor de verdad de las fórmulas en donde aparezca sufrirá variaciones.

¿Por qué no se utiliza la lógica clásica cuyos lenguajes y cálculos conocemos tan bien? La respuesta habitual a esta pregunta es que precisamos de un lenguaje en donde se pueda expresar el cambio de valor de verdad de una fórmula como resultado de una acción (*en particular, la ejecución de un programa*), en donde la verdad pueda ser relativizada. En la lógica clásica la verdad de una fórmula en un modelo dado es absoluta. No sucede así en la modal, en donde la verdad es inestable, depende del mundo, situación o estado en donde se evalúa. No es, por

[2] Hay también una lógica dinámica general, que se inspira en la anterior y que no sólo se aplica al estudio de los programas, sino a la *dinámica de la transmisión de la información* en un sentido mucho más amplio, es la que presenta van Benthem en [13].

consiguiente, de extrañar que la lógica propuesta por este fin, la denominada lógica dinámica, sea una generalización de la lógica modal.

9.2.2. La lógica dinámica es una extensión de la lógica modal

Ya vimos en el capítulo 8 que la lógica modal es la lógica de la *necesariedad* y de la *posibilidad*. Comienza su historia con una etapa primitiva, no sistemática, que incluye los trabajos de Aristóteles, los megáricos, los estoicos y algunos medievales. Para Leibniz son *necesarias* las sentencias verdaderas en todo mundo posible y *posibles* los son las que valen en alguno.

La fase sistemática se sitúa en el siglo XX y comienza, más o menos independientemente, con Lewis, Łukasiewicz y Carnap. Ellos desarrollaron diversos cálculos modales, pero la semántica, aunque ampliamente debatida, necesitaba ser sistematizada. El proceso desemboca en lo que Bull y Segerberg [**?**] denominan *«revolución industrial»* que corresponde a la de Kripke, Prior, Kanger y Hintikka, cuando se define con precisión la semántica.

En la actualidad, la lógica modal se ha diversificado enormemente y a los objetivos tradicionales han venido a sumarse:

1. Problemas filosóficos muy elaborados (Fine),
2. Una ampliación de la teoría de modelos con nociones modales (Mortimer),
3. La interpretación de los operadores modales como demostrabilidad (Boolos, Solovay),
4. Nuestra lógica dinámica (Andréka, Harel, Kozen, Meyer, Németi, Pratt) y
5. Los planteamientos de la denominada *"gramática de Montague"*.

¿Cómo se modifica en la lógica dinámica la semántica de los mundos posibles? Como veremos, el esquema abstracto de la lógica modal y, en particular, los modelos de Kripke, le sientan perfectamente a la lógica dinámica. $\mathbf{W}$ será ahora el conjunto de los estados relevantes para los programas en consideración. Con cada programa b asociamos una relación binaria de accesibilidad de forma que el par $\langle s, t\rangle$ está en ella si hay un estado t al que se llega desde s mediante el programa b; es decir, una computación que transforma el estado s en estado t. Así, un programa es concebido como un conjunto de pares de estados: iniciales y finales. Puesto que vamos a admitir programas no deterministas —es decir, en donde el estado inicial no determina unívocamente el estado final— la relación no necesita ser una función.

Como es de suponer, asociando a cada programa una relación de accesibilidad, se tienen que manejar simultáneamente muchas de ellas y los que haremos es colocar dentro de $\Box$ y de $\Diamond$ el programa concreto al que nos referimos; es decir, escribiremos $[b]$ y $\langle b\rangle$. Además de esta extensión de la lógica modal

consistente en considerar conjuntamente diversas relaciones de accesibilidad, en lógica dinámica está permitida formar programas compuestos a partir de programas simples; es decir, las relaciones se combinan mediante unión, composición y *«loop»* para formar nuevas relaciones.

El lenguaje de la lógica dinámica que voy a presentar es el proposicional.

9.2.3. Lenguaje y semántica de *PDL*

Con las fórmulas y programas atómicos se construyen inductivamente las fórmulas y programas de la dinámica usando para ello: los conectores, los operadores modales, los operadores de programas y el operador de test.

En primer lugar, tenemos fórmulas atómicas. Además de eso, a partir de fórmulas cualesquiera, los conectores nos permiten formar nuevas fórmulas mediante el procedimiento normal. Mientras que los programas se combinan mediante sus operadores para formar nuevos programas —la unión y la composición de programas son operadores binarios, mientras que el *loop* lo es monario—. A partir de una fórmula se forma un programa que es el *test* de esa fórmula ($\varphi?$). Y con una fórmula, un programa y un operador modal se forma una nueva fórmula —por ejemplo, $\langle a\rangle\varphi$—. Son estas últimas las fórmulas características de la lógica dinámica, las que nos permiten expresar propiedades de programas.

¿Qué significado intuitivo atribuimos a estas fórmulas y programas?

$(a;b)$	significa	«*Haz a seguido de b*»
$(a \cup b)$	significa	«*Haz a o b*»
$a*$	significa	«*Haz a un número finito, pero no precisado de veces*»
$[a]\,\varphi$	significa	«*a es parcialmente correcto respecto del OUTPUT* φ»
$\langle a\rangle\,\varphi$	significa	«*a es totalmente correcto respecto del OUTPUT* φ»

Naturalmente, esto no constituye una definición aceptable, se trata simplemente de indicar qué interpretación intuitiva queremos darle a las expresiones de *PDL*, la definición, mucho más precisa, será por inducción sobre la construcción de fórmulas.

¿Qué modelos usa su semántica?

Se trata de una ampliación de la modal: Los modelos contienen un universo de mundos o de estados, una familia de subconjuntos del universo de estados y una familia de relaciones sobre mundos. Aquí, en vez de tener una relación de accesibilidad, tenemos una para cada programa atómico ya que los programas son concebidos como objetos dinámicos que permiten que el computador pase de un estado a otro; es decir, cumplen la misma misión que las relaciones de accesibilidad entre mundos en la modal. Al interpretar las fórmulas y programas

del lenguaje se hace de manera que toda fórmula represente el conjunto de los estados en donde es verdad y que cada programa se interprete como el conjunto de pares de estados iniciales y finales entre los que el programa nos lleva.

En particular, la interpretación del loop es la menor clausura reflexiva y transitiva de la relación sobre la que se aplica.

¿Es posible generar mediante un cálculo las fórmulas válidas?

Para la lógica dinámica sentencial hay un cálculo deductivo completo, que toma como axiomas los de la lógica sentencial no modal, los que regulan el funcionamiento de la composición de programas y los de la lógica minimal **K**. A diferencia de lo que sucede en lógica modal donde los axiomas describen la relación de accesibilidad, los axiomas dinámicos reflejan las características de los operadores de programas. De entre ellos cabe destacar los que describen el funcionamiento del *loop*, especialmente el que es conocido como axioma de inducción.

$$\varphi \wedge [a^*]\,(\varphi \rightarrow [a]\,\varphi \rightarrow [a^*]\,\varphi)$$

Justamente este axioma es el responsable de alguna de las metapropiedades más singulares de *PDL*, su *incompacidad*. El cálculo de la lógica dinámica sentencial es *correcto* y *completo*, pero *en sentido débil*, pues el teorema de compacidad falla. Es decir, es completo para validez

$$\models \varphi \Longrightarrow \vdash \varphi$$

—es siempre cierto—
pero no es completo para consecuencia

$$\Gamma \models \varphi \Longrightarrow \Gamma \vdash \varphi$$

—no siempre es cierto—.

Este cálculo es también *decidible*, aunque el tiempo empleado en calcular si una fórmula es o no un teorema, crece exponencialmente con la complejidad de la fórmula.

Resumen 335 *Estas son sus propiedades más sobresalientes:*

1. *PDL = Metalenguaje para hablar sobre programas*
2. *Programas = Relaciones entre estados*
3. *Estados = Contenidos registros de memoria*

Comentario 336 *Estas son algunas de sus características:*

1. *PDL es* ***dinámica:*** *la verdad no es absoluta, depende de los estados*
2. *PC (lógica clásica) es* ***estática***
3. *PML (lógica modal) es* ***dinámica***
4. *PDL es* ***multimodal:*** *se usan simultáneamente infinitas relaciones de accesibilidad*

9.3. El lenguaje de la lógica dinámica

Con las fórmulas y programas atómicos —conjuntos $\Phi_0 = \mathbf{ATOM.PROP}$ y $\Pi_0 = \mathbf{ATOM.PROG}$— se construyen inductivamente las fórmulas y programas de la dinámica usando para ello: los conectores $\neg$, $\vee$, $\wedge$, $\rightarrow$ y $\leftrightarrow$; los operadores modales $\Box$ y $\Diamond$; los operadores de programas $\cup$, $*$, y $;$ y el operador de test $?$.

$$\langle fórmula\rangle := \langle \mathbf{ATOM.PROP}\rangle \mid \bot \mid \neg\langle fórmula\rangle \mid \langle fórmula\rangle\wedge\langle fórmula\rangle \mid$$

$$\langle fórmula\rangle\vee\langle fórmula\rangle \mid \langle fórmula\rangle \rightarrow \langle fórmula\rangle \mid \langle fórmula\rangle \leftrightarrow \langle fórmula\rangle \mid$$

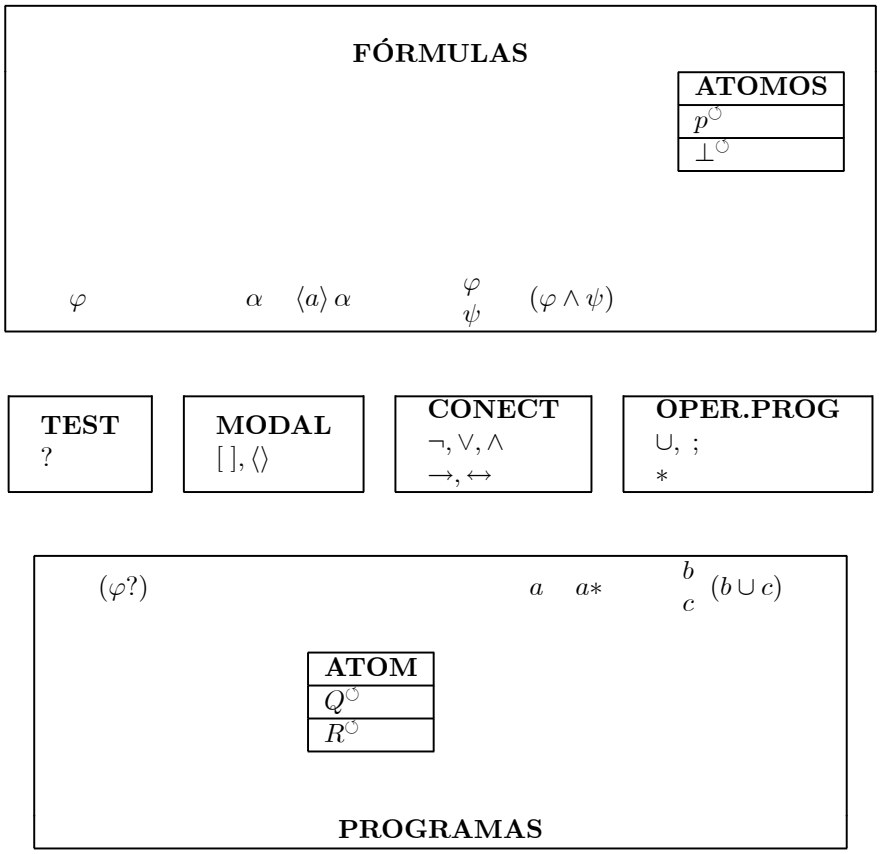
$$[\langle programa\rangle]\,\langle fórmula\rangle \mid \langle\langle programa\rangle\rangle\,\langle fórmula\rangle$$

$$\langle programa\rangle := \langle \mathbf{ATOM.PROG}\rangle \mid \bot \mid \langle programa\rangle\cup\langle programa\rangle \mid \langle programa\rangle^* \mid$$

$$\langle programa\rangle\,;\langle programa\rangle \mid \langle fórmula\rangle?$$

Es decir, además de las atómicas (regla **F1),** tenemos las fórmulas formadas mediante la regla **F2**: $\bot$, $\neg\varphi$, $(\varphi \wedge \psi)$, $(\varphi \vee \psi)$, $(\varphi \rightarrow \psi)$, $(\varphi \leftrightarrow \psi)$ y las que usan la regla modal **F3**: Si φ es una fórmula y a es un programa, también son fórmulas $[a]\,\varphi$ y $\langle a\rangle\,\varphi$.

Estipulamos que son programas: (1) los programas atómicos (regla **P1),** (2) los que se forman mediante la regla **P2:** Si a y b son programas, también lo son: $(a\cup b)$, a^* y $(a;b)$ y (3) los obtenidos a partir de fórmulas usando la regla del operador de test **P3:** Si φ es una fórmula, entonces $(\varphi?)$ es un programa



9.4. Modelos de Kripke

Para interpretar las fórmulas dinámicas se definen estructuras de Kripke de la forma

$$\mathcal{A} = \left\langle \mathbf{W}, \langle Q^{\mathcal{A}} \rangle_{Q \in \mathbf{ATOM.PROG}}, \langle p^{\mathcal{A}} \rangle_{p \in \mathbf{ATOM.PROP}} \right\rangle \tag{9.1}$$

donde: $\mathbf{W} \neq \emptyset$ es un conjunto de estados; $Q^{\mathcal{A}} \subseteq \mathbf{W} \times \mathbf{W}$ son relaciones entre estados iniciales y finales; y $p^{\mathcal{A}} \subseteq \mathbf{W}$ conjunto de estados donde p es verdadera.

9.4.1. Interpretación

Dada una estructura de Kripke $\mathcal{A}$ inductivamente se define la función

$$\Im : \mathbf{EXPR} \longrightarrow \wp(\mathbf{W}) \cup \wp(\mathbf{W} \times \mathbf{W}) \tag{9.2}$$

que a cada $\varphi \in \mathbf{FORM}$ le asigna el conjunto de los estados donde φ es verdadera: $\Im(\varphi) \subseteq \mathbf{W}$ y a cada $a \in \mathbf{PROG}$ la relación $\Im(a) \subseteq \mathbf{W} \times \mathbf{W}$

Como en lógica modal:

- **F1:** $\Im(p) = p^{\mathcal{A}}$
- **F2:** $\Im(\bot) = \emptyset$, $\Im(\varphi \wedge \psi) = \Im(\varphi) \cap \Im(\psi)$,

  $\Im(\varphi \to \psi) = \mathbf{W} - (\Im(\varphi) \cap (\mathbf{W} - \Im(\psi)))$ y de forma semejante para el resto de las conectivas
- **F3:** $\Im(\langle a \rangle \varphi) = \{s \in \mathbf{W} \mid \exists t(t \in \mathbf{W} \ \& \ \langle s,t \rangle \in \Im(a) \ \& \ t \in \Im(\varphi))\}$

  $\Im([a] \varphi) = \{s \in \mathbf{W} \mid \forall t(t \in \mathbf{W} \ \& \ \langle s,t \rangle \in \Im(a) \Rightarrow t \in \Im(\varphi))\}$

Para interpretar los programas se define inductivamente una relación binaria formada por los pares de estados iniciales y finales entre los que el programa nos lleva.

- **P1:** $\Im(Q) = Q^{\mathcal{A}}$
- **P2:** $\Im(a \cup b) = \Im(a) \cup \Im(b)$, $\Im(a;b) = \Im(a) \circ \Im(b)$

  $\Im(a^*) = (\Im(a))^* = \left\{ \begin{array}{c} \langle s,t \rangle \in \mathbf{W} \times \mathbf{W} \mid \exists k \exists s_0,..,s_k (s_0 = s \& s_k = t \ \& \\ \forall i (i \in \{1,...,k\} \Longrightarrow \langle s_{i-1}, s_i \rangle \in \mathcal{A}(a))) \end{array} \right\}$

  Dada una relación $\mathbf{R}$, su clausura reflexiva y transitiva es $\mathbf{R}^* = \bigcup_{n \geq 0} R^n$
- **P3:** $\Im(\varphi?) = \{\langle s,s \rangle \in \mathbf{W} \times \mathbf{W} \mid s \in \Im(\varphi)\}$

9.4.2. Ejemplos

1. Sea (ver figura: 9.1) $\mathcal{A} = \left\langle \mathbf{W}, \langle Q^{\mathcal{A}} \rangle_{Q \in \mathbf{ATOM.PROG}}, \langle p^{\mathcal{A}} \rangle_{p \in \mathbf{ATOM.PROP}} \right\rangle$ con

   $\mathbf{W} = \{b, e\}$, $Q_1^{\mathcal{A}} = \{\langle b,b \rangle\}$ y $Q_2^{\mathcal{A}} = \{\langle b,e \rangle\}$

   De las definiciones precedentes se desprenden las siguientes interpretaciones de fórmulas y programas:

   $\Im(\bot) = \emptyset$, $\Im(\top) = \mathbf{W}$, $\Im(Q_1 \cup Q_2) = \{\langle b,b \rangle, \langle b,e \rangle\}$,

   $\Im([(Q_1 \cup Q_2)] \bot) = \{e\}$, $\Im(\langle Q_1 \rangle \top) = \{b\}$,

   $\Im(\langle Q_2 \rangle [(Q_1 \cup Q_2)] \bot) = \{b\}$,

   $\Im([Q_1^*] (\langle Q_1 \rangle \top \wedge \langle Q_2 \rangle [(Q_1 \cup Q_2)] \bot) = \{b, e\}$
2. Sea (ver figura: 9.2) $\mathcal{B} = \left\langle \mathbf{W}, \langle Q^{\mathcal{B}} \rangle_{Q \in \mathbf{ATOM.PROG}}, \langle p^{\mathcal{B}} \rangle_{p \in \mathbf{ATOM.PROP}} \right\rangle$ y sea

   $\mathbf{W} = \{b_i \mid i \geq 0\} \cup \{e_i \mid i \geq 0\}$ con $Q_1^{\mathcal{B}} = \{\langle b_i, b_{i+1} \rangle \mid i \geq 0\}$ y

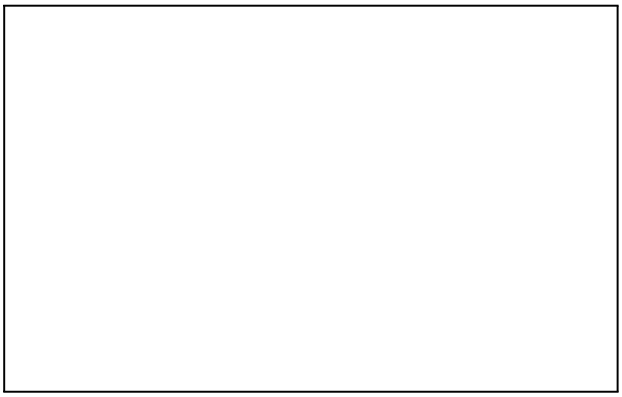


Figura 9.1: *Modelo Kripke dinámico, 1*

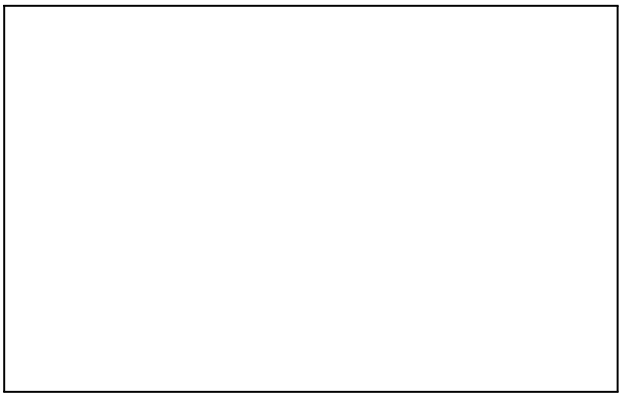


Figura 9.2: *Modelo Kripke dinámico, 2*

$Q_2^{\mathcal{B}} = \{\langle b_i, e_i\rangle \mid i \geq 0\}$

En la interpretación correspondiente: $\Im(Q_1^*) = \{\langle b_i, b_j\rangle \mid i \leqq j\}$,
$\Im([Q_1 \cup Q_2]\,\bot) = \{e_i \mid i \geq 0\}$, $\Im(\langle Q_2\rangle\,[Q_1 \cup Q_2]\,\bot) = \{b_i \mid i \geq 0\}$,
$\Im(\langle Q_1\rangle\,\top = \{b_i \mid i \geq 0\}$, $\Im([Q_1^*]\,(\langle Q_1\rangle\,\top \wedge \langle Q_2\rangle\,[(Q_1 \cup Q_2)]\,\bot)) = \{b_i \mid i \geq 0\}$

3. Sea (ver figura: 9.3) $\mathcal{C} = \left\langle \mathbf{W}, \left\langle Q^{\mathcal{C}}\right\rangle_{Q\in\mathbf{ATOM.PROG}}, \left\langle p^{\mathcal{C}}\right\rangle_{p\in\mathbf{ATOM.PROP}}\right\rangle$ y sea

$\mathbf{W} = \{0,1,2,3\}$, $p_1^{\mathcal{C}} = \{2,3\}$, $p_2^{\mathcal{C}} = \{0,1,3\}$
$Q_1^{\mathcal{C}} = \{\langle 3,0\rangle, \langle 0,3\rangle, \langle 2,1\rangle, \langle 1,2\rangle\}$, $Q_2^{\mathcal{C}} = \{\langle 3,2\rangle, \langle 2,3\rangle, \langle 1,0\rangle, \langle 0,1\rangle\}$

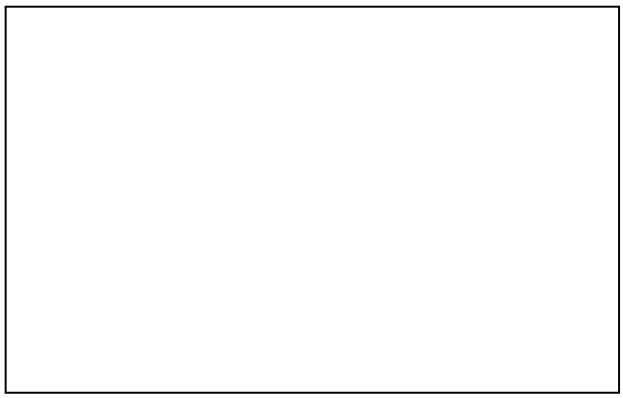


Figura 9.3: *Modelo Kripke dinámico, 3*

$\Im(\langle Q_2\rangle\, p_1) = \{3,2\}\,,\ \ \Im(\langle Q_2\rangle\, p_2) = \{0,1,2\}\,,$
$\Im([Q_1^*]\, p_2) = \{0,3\}\,,\ \ \Im(p_2 \to [Q_1^*]\, p_2) = \{0,2,3\}$

9.4.3. Satisfacibilidad, validez y consecuencia

Sea $\mathcal{A} = \left\langle \mathbf{W}, \langle Q^{\mathcal{A}}\rangle_{Q\in\mathbf{ATOM.PROG}}, \langle p^{\mathcal{A}}\rangle_{p\in\mathbf{ATOM.PROP}} \right\rangle$ un modelo de Kripke, $\Im$ su interpretación asociada y $s \in \mathbf{W}$.

Definición 337 φ *es* ***verdadera en estado*** s ***del modelo*** $\mathcal{A}$

$$\mathcal{A},s \Vdash \varphi \Longleftrightarrow_{Df} s \in \Im(\varphi)$$

Definición 338 φ *es* ***verdadera*** *(o válida)* ***en*** $\mathcal{A}$

$\mathcal{A} \Vdash \varphi \Longleftrightarrow_{Df} \mathcal{A},s \models \varphi$, *para cada* $s \in \mathbf{W}$ $-i.e.\ \Im(\varphi) = \mathbf{W}-$

Definición 339 φ *es* ***válida***
$\models \varphi \Longleftrightarrow_{Df} \mathcal{A} \models \varphi$, *para todo modelo* $\mathcal{A}$

Definición 340 Γ *es* ***satisfacible en estado*** s ***del modelo*** $\mathcal{A}$

$$\mathcal{A},s \Vdash \Gamma \Longleftrightarrow_{Df} s \in \bigcap_{\gamma\in\Gamma} \Im(\gamma)$$

Definición 341 Γ *es* ***válido*** *en* $\mathcal{A}$

$$\mathcal{A} \Vdash \Gamma \Longleftrightarrow_{Df} \bigcap_{\gamma\in\Gamma} \Im(\gamma) = \mathbf{W}$$

Definición 342 φ *es* ***consecuencia local*** *de* Γ
$\Gamma \models_L \varphi \Longleftrightarrow_{Df}$ *Para toda estructura de Kripke* $\mathcal{A}$ *y* $s \in \mathbf{W}$*:*
si $s \in \bigcap_{\gamma\in\Gamma} \Im(\gamma)$ *entonces* $s \in \Im(\varphi)$

Definición 343 φ *es* ***consecuencia global*** *de* Γ
$\Gamma \models_G \varphi \Longleftrightarrow_{Df}$ *Si* $\bigcap_{\gamma\in\Gamma} \Im(\gamma) = \mathbf{W}$ *entonces* $\Im(\varphi) = \mathbf{W}$

9.5. Definición de nuevos programas y de sus propiedades

En el lenguaje de la lógica dinámica se pueden componer nuevos programas a partir de los básicos usando los operadores. He aquí algunos ejemplos:

1. IF φ THEN a ELSE b $\equiv_{Df} ((\varphi?; a) \cup (\neg\varphi?; b))$
2. WHILE φ DO a $\equiv_{Df} ((\varphi?; a)*; \neg\varphi?)$
3. ABORT $\equiv_{Df} \bot?$
4. REPEAT a UNTIL φ $\equiv_{Df} (a; (\neg\varphi?; a)*)$
5. SKIP $\equiv_{Df} \top?$

Para expresar ciertas propiedades de los programas usamos también este lenguaje:

1. $[a]\,\varphi$: Correción parcial programa a, sentido fuerte
2. $\langle a\rangle\,\varphi$: Correción total programa a, sentido débil.
3. $[a]\,\varphi \wedge \langle a\rangle\,\varphi$: Correción total programa a, sentido fuerte
4. $\varphi \rightarrow [a]\,\psi$: Corrección programa a respecto INPUT φ y OUTPUT ψ
5. $[a]\,\bot$: Programa a no tiene fin
6. $\langle a\rangle\,\top$: Programa a termina
7. $\langle a\rangle\,\varphi \leftrightarrow \langle b\rangle\,\varphi$: Equivalencia de programas a y b respecto φ
8. $\langle a\rangle\,\varphi \rightarrow [a]\,\varphi$: Lo cumplen los programas deterministas, para cada φ

9.6. Cálculo

PDL es una lógica modal normal que contiene a la lógica **K** —esto es, las tautologías, K, Df$\Diamond$, MP y N— y los siguientes axiomas:

1. **A1** $:= \langle a; b\rangle\,\varphi \leftrightarrow \langle a\rangle\,\langle b\rangle\,\varphi$
2. **A2** $:= \langle a \cup b\rangle\,\varphi \leftrightarrow \langle a\rangle\,\varphi \vee \langle b\rangle\,\varphi$
3. **A3** $:= \langle a*\rangle\,\varphi \leftrightarrow (\varphi \vee \langle a\rangle\,\langle a*\rangle\,\varphi)$
4. **A4** $:= \langle\varphi?\rangle\,\psi \leftrightarrow (\varphi \wedge \psi)$

5. **A5** $:= [a*]\,(\varphi \to [a]\,\varphi) \to (\varphi \to [a*]\,\varphi)$

Ejercicio 344 *Usando las reglas del cálculo es sencillo demostrar los siguientes teoremas:*

1. $\vdash_{PDL} [([a]\,\varphi)?; a]\,\varphi$
2. $\vdash_{PDL} [([a]\,\varphi)?; a]\,\varphi \leftrightarrow \neg\,\langle([a]\,\varphi)?; a\rangle\,\neg\varphi$
3. $\vdash_{PDL} \neg\,\langle([a]\,\varphi)?\rangle\,\langle a\rangle\,\neg\varphi \leftrightarrow \neg([a]\,\varphi) \wedge \langle a\rangle\,\neg\varphi)$
4. $\vdash_{PDL} \neg([a]\,\varphi) \wedge \langle a\rangle\,\neg\varphi) \leftrightarrow \neg([a]\,\varphi) \vee \neg\,\langle a\rangle\,\neg\varphi$
5. $\vdash_{PDL} \neg([a]\,\varphi) \vee \neg\,\langle a\rangle\,\neg\varphi \leftrightarrow \neg([a]\,\varphi) \vee [a]\,\varphi$
6. $\vdash_{PDL} [([a]\,\varphi)?; a]\,\varphi \leftrightarrow \neg([a]\,\varphi) \vee [a]\,\varphi$
7. $\vdash_{PDL} \neg([a]\,\varphi) \vee [a]\,\varphi$
8. $\vdash_{PDL} [([a]\,\varphi)?; a]\,\varphi$

9.7. Metateoría de la lógica dinámica

9.7.1. Incompacidad

La lógica dinámica no es compacta, ni en sentido local ni global.

Teorema 345 PDL *no es compacta en sentido local*

Demostración. Para demostrar que no es compacta en sentido local veremos que hay un Σ y una φ tales que $\Sigma \vDash_L \varphi$ pero no hay ningún subconjunto finito suyo $\Sigma^* \subseteq \Sigma$ tal que $\Sigma^* \vDash_L \varphi$.

Sea $\Sigma = \{p, [Q]\,p, [Q;Q]\,p, ..., [Q^n]\,p, ...\}$ y $\varphi := [Q^*]\,p$

Claramente $\Sigma \vDash_L \varphi$ pero $\Sigma^* \nvDash_L \varphi$ para cualquier $\Sigma^* \subseteq \Sigma$ finito. Para verlo, sea n el mayor exponente de las fórmulas $\sigma \in \Sigma^*$ y sea $\mathcal{A} = \langle \mathbb{N}, p^{\mathcal{A}}, Q^{\mathcal{A}}\rangle$ donde $p^{\mathcal{A}} = \{0, 1, ..., n\}$ $Q^{\mathcal{A}}$ es la función del siguiente. Sea $\Im$ la interpretación definida sobre $\mathcal{A}$.

$$\underbrace{0}_{p} \Longrightarrow \underbrace{1}_{p} \Longrightarrow \underbrace{2}_{p} \Longrightarrow \underbrace{\cdots}_{p} \Longrightarrow \underbrace{n}_{p} \Longrightarrow \underbrace{n+1}_{\neg p} \Longrightarrow$$

Se ve fácilmente que $0 \in \Im(\sigma)$, para cada $\sigma \in \Sigma^*$ $0 \in \bigcap_{\sigma\in\Sigma^*} \Im(\sigma)$ pero $0 \notin \Im([Q^*]\,p)$ ■

Teorema 346 PDL *no es compacta en sentido global*

Demostración. Para demostrar que no es compacta en sentido local veremos que hay un Σ y una φ tales que $\Sigma \vDash_G \varphi$ pero no hay ningún subconjunto finito suyo $\Sigma^* \subseteq \Sigma$ tal que $\Sigma^* \vDash_G \varphi$.

Sea $\Sigma = \{q \to p, q \to [Q]\, p, q \to [Q;Q]\, p, ..., q \to [Q^n]\, p, ...\}$ y $\varphi := q \to [Q^*]\, p$

Claramente $\Sigma \vDash_G \varphi$ pero $\Sigma^* \nvDash_G \varphi$ para cualquier $\Sigma^* \subseteq \Sigma$ finito. Para verlo, sea $q \to [Q^n]\, p \notin \Sigma^*$ y sea $\mathcal{A} = \langle \mathbb{N}, p^{\mathcal{A}}, q^{\mathcal{A}}, Q^{\mathcal{A}} \rangle$ donde $p^{\mathcal{A}} = \mathbb{N} - \{n\}$, $q^{\mathcal{A}} = \{0\}$ $Q^{\mathcal{A}}$ es la función del siguiente. Sea $\Im$ la interpretación definida sobre $\mathcal{A}$.

$$\begin{array}{ccccccccccc} 0 & \longrightarrow & 1 & \longrightarrow & 2 & \longrightarrow & \cdots & \longrightarrow & n & \longrightarrow & n+1 & \longrightarrow \\ \underbrace{} & & \underbrace{} & & \underbrace{} & & & & \underbrace{} & & \underbrace{} & \\ q & & \neg q & & \neg q & & & & \neg q & & \neg q & \\ p & & p & & p & & & & \neg p & & p & \end{array}$$

Se ve fácilmente que

$$\mathcal{A}, s \Vdash q \to [Q^m]\, p, \text{ para cualquier } s \neq 0$$

$$\mathcal{A}, 0 \Vdash q \to [Q^m]\, p, \text{ para cualquier } m \neq n$$

$$\mathcal{A}, s \nVdash q \to [Q^n]\, p$$

■

9.7.2. Completud

Para demostrar la completud lo natural sería construir el modelo canónico

$$\mathcal{A}_{PDL} = \left\langle \mathbf{W}_{PDL}, \langle Q^{PDL} \rangle_{Q \in \mathbf{ATOM.PROG}} \langle p^{PDL} \rangle_{p \in \mathbf{ATOM.PROP}} \right\rangle$$

tomando como universo el conjunto formado por todos los conjuntos máximamente consistentes y como relaciones las definidas así

$$\langle s, t \rangle \in Q^{PDL} \text{ syss } \{\varphi \mid [Q]\, \varphi \in s\} \subseteq t$$

siendo

$$p^{PDL} = \{s \in \mathbf{W}_{PDL} \mid p \in s\}$$

Pero esta estructura es un modelo no estándar de *PDL* pues aunque todos los teoremas de *PDL* son verdaderos en él y falsas las fórmulas que no son teoremas, la interpretación del *loop* no es la estándar; esto es, no es la menor clausura reflexiva y transitiva sobre la relación.

Teorema 347 *$\mathcal{A}_{PDL}$ cumple todas las condiciones propias de los modelos de Kripke para la lógica dinámica [9.4] excepto la del* loop

$$Q^{*\mathcal{A}_{PDL}} \subseteq (Q^{\mathcal{A}_{PDL}})^*$$

Para demostrar completud se usará la técnica del *filtrado*, pero antes modificaremos el modelo canónico para obtener uno estándar, que haga falsa cierta fórmula $\varphi \in \mathbf{FORM}$. Para conseguirlo colapsaremos $\mathcal{A}_{PDL}$ usando un conjunto Γ adecuado, que contiene a φ. Las reglas de clausura de Γ son

1. Γ está cerrado bajo formación de subfórmulas
2. $[\varphi?]\,\psi \in \Gamma \Longrightarrow \varphi \in \Gamma$
3. $[a;b]\,\varphi \in \Gamma \Longrightarrow [a]\,[b]\,\varphi \in \Gamma$
4. $[a \cup b]\,\varphi \in \Gamma \Longrightarrow [a]\,\varphi \in \Gamma$ y $[b]\,\varphi \in \Gamma$
5. $[a^*]\,\varphi \in \Gamma \Longrightarrow [a]\,[a^*]\,\varphi \in \Gamma$

A los conjuntos Γ que cumplen estas condiciones se les llama *cerrados.*

Lema 348 *(Fisher y Ladner: 1979) Si Γ es el menor conjunto cerrado que contiene a φ entonces Γ es finito.*

Habiendo demostrado la finitud de Γ se realiza una $\Gamma-$filtración del modelo canónico. Hacemos que $\mathbf{ATOM}.\mathbf{PROP}_\Gamma = \mathbf{ATOM} \cap \Gamma$ y que $\mathbf{ATOM}.\mathbf{PROG}_\Gamma$ sea el menor conjunto de programas que incluya:

1. los programas atómicos que aparecen en Γ
2. todos los test $\varphi?$ de fórmulas de $\varphi \in \Gamma$
3. que esté cerrado bajo $\cup$, ; y $*$

El modelo que definamos

$$\mathcal{A}_\Gamma = \left\langle \mathbf{W}_\Gamma, \langle Q^{\mathcal{A}_\Gamma} \rangle_{Q \in \mathbf{ATOM}.\mathbf{PROG}_\Gamma} \langle p^{\mathcal{A}_\Gamma} \rangle_{p \in \mathbf{ATOM}.\mathbf{PROP}_\Gamma} \right\rangle$$

cumple lo deseado.

Teorema 349 *$\mathcal{A}_\Gamma$ es una $\Gamma-$filtración de $\mathcal{A}_{PDL}$*

Basándonos en ello, podemos demostrar

Lema 350 *(del filtrado). Para cada $\varphi \in \Gamma$*

$$\mathcal{A}_{PDL}, s \Vdash \varphi \quad syss \quad \mathcal{A}_\Gamma, |s| \Vdash \varphi$$

Corolario 351 *$\mathcal{A}_\Gamma$ es un modelo estándar.*

A partir de estos resultados es fácil demostrar que *PDL* tiene la propiedad de los modelos finitos y es decidible.

9.7.3. *SOLO*: Una teoría de segundo orden equivalente a *PDL*

En [12] defino una traducción de las fórmulas dinámicas a fórmulas multivariadas —esencialmente de segundo orden— y una extensión de las estructuras dinámicas a estructuras multivariadas. Utilizando el nuevo lenguaje se define la teoría *SOLO* —Second Order Logic *Oraculum*— que suple sintáctica y semánticamente a *PDL*, pudiéndose demostrar

$$\vDash \varphi \Longleftrightarrow SOLO \vDash TRANS(\varphi)$$

siendo $TRANS$ la función traductora, recursivamente definida.

Los detalles del planteamiento están en el libro mencionado, la idea general de la traducción a una lógica marco, las distintas etapas del proceso traductológico y el ejemplo de la lógica modal se encuentra en el presente trabajo en el capítulo 13.

Comentario 352 ***"Hacia el infinito, y más allá":*** *Al estudiar las propiedades metalógicas de las lógicas de programas nos situamos a un nivel de abstracción inmediatamente superior a cuando las usamos para demostrar propiedades de programas.*
En este nivel doblemente metateórico podemos acometer el estudio y la comparación de las diversas lógicas de programas, en particular , por lo que respecta a su capacidad expresiva, y muy especialmente , por su potencia para demostrar la corrección de programas.

Bibliografía

[1] van Benthem, J. [1991]. *Language in Action. Categories, Lambdas and Dynamic Logic*, Amsterdam, North-Holland.

[2] Bull, R. Segerberg, K. [1984]. *"Basic Modal Logic".* En [4]

[3] Clarke, E. y Kozen, D. editores. [1984]. *Proceedings of the Logics of Programming Workshop*, volumen 164 de Lecture Notes in Computer Science, páginas 221–256. Springer-VerlagGoguen, J.A. y Burstall, R.M. Lecture Notes in Computer Science. vol. 164. Springer-Verlarg. Berlín. Alemania.

[4] Gabbay, D y Guenthner, F. [84]. *Handbook of Philosophical Logic Handbook of Philosophical Logic* 2^{nd} edition [2001]. Kluwer Academic Publishers. Dordrecht. Holanda.

[5] Goldblatt, R. [1992]. *Logics of Time and Computation*, Center for the Study of Language and Information. Stanford. USA

[6] Harel, D. [2001]. *"Dynamic Logic".* en [4].

[7] Kozen D. y Parikh, R. [1981]. *"An elementary proof of the completeness of PDL".* Theoretical Computer Science. vol 14 pp 113-118

[8] Manzano, M. [1986]. *"Nuevos hechos, viejos lenguajes".* **V Congrés Català de Lógica.** Barcelona PPU. pp. 17-22

[9] Manzano, M. [1989]. *"SOLO: A second order theory equivalent to dynamic logic".* En Logic Colloquim'87. Resumen en **JSL** vol 54.

[10] Manzano, M. [1990]. *"Lógica dinámica".* **Agora**, vol 9, pp 31-43.

[11] Manzano, M. [1991]. *"La bella y la Bestia. (perdón, lógica e Informática)".* En *Arbor.* Madrid.

[12] Manzano, M. [1996]. *Extensions of First Order Logic***.** Cambridge Tracts in Theoretical Computer Science. Cambridge University Press. Cambridge. U.K.

[13] van Benthem, J. [1996] *Exploring Logical Dynamics*, CSLI, Stanford. USA

[14] Sloman, A. [1978]. *The computer revolution in philosophy.* The Harvester Press. Sussex.

Capítulo 10

Lógica de Segundo Orden

10.1. Introducción

La lógica de segundo orden *(SOL)* se distingue de la de primer orden en que posee variables relacionales además de las individuales, y ambas pueden cuantificarse. Ya que fue Frege el precursor en el uso de las variables relacionales, la lógica de segundo orden cuenta ya más de cien años, aunque una clara diferenciación entre ella y la de primer orden se hizo esperar y supuso el esfuerzo conjunto de otros lógicos. La distinción estaba implícita en el trabajo de Russell, pero no fue explícita hasta el de Hilbert y Ackermann [28]. De hecho, la lógica de primer orden es sólo un fragmento del lenguaje altamente expresivo de Frege [21] y Russell [42]. Durante mucho tiempo fue considerado un *"estudio esotérico"*, limitado al ámbito meramente filosófico; actualmente recibe reconocimiento por su utilidad en aplicaciones y por su importancia en fundamentación de la *informática teórica*[1].

En la lógica de segundo orden podemos decir: *"para todos los individuos,* φ *se cumple"*, como en la lógica de primer orden, y formalizarlo así $\forall x\varphi$. También podemos expresar: *"para toda propiedad se verifica* φ", algo que no nos estaba permitido en lógica de primer orden, y lo escribimos así $\forall X\varphi$. Mediante $\forall X^2\varphi$ indicamos: *"para todas las relaciones binarias,* φ *se cumple"*. Y de ahí en adelante...

Por consiguiente, las estructuras de segundo orden deben poseer distintos dominios de cuantificación: el dominio de individuos $\mathbf{A}$, donde tomarán valores las variables individuales del lenguaje; el dominio de relaciones unarias $\mathbf{A}_1$, sobre el que toman valores las variables predicativas unarias; el dominio de relaciones binarias $\mathbf{A}_2$, y así sucesivamente.

Si queremos que nuestra lógica de segundo orden sea estándar, la fórmula $\forall X\varphi$ deberá significar lo siguiente: *"para todo subconjunto de* $\mathbf{A}$ *se verifica* φ*"* Al hacer este tipo de aseveraciones estamos tomando el concepto de subconjunto

[1] Una buena visión nos la ofrece Daniel Leivan en [32]

de la teoría de conjuntos subyacente que usamos como metalenguaje. Lo cual nos obliga a incluir en $\mathbf{A}_1$ todos los subconjuntos de $\mathbf{A}$, incluso aquellos tan abstrusos que eluden todo intento de descripción o definición. Considérese el caso en el que $\mathbf{A}$ fuera un conjunto de dimensión infinita —sea su cardinalidad α— y sea $\wp(\mathbf{A})$ el conjunto potencia de $\mathbf{A}$, que es a su vez $\mathbf{A}_1$; sabemos que éste contiene 2^α miembros —por el teorema de Cantor—, mientras que nuestro lenguaje formal sólo cuenta con $\aleph_0$ fórmulas. Así pues, mediante un cálculo simple constatamos que la mayor parte de los subconjuntos que tomamos en $\mathbf{A}_1$ no pueden ser definidos. Como veremos más adelante, si añadimos a nuestro universo de conjuntos $\mathbf{A}_1$, todos los caracterizados de un modo tan poco descriptivo como el de ser un subconjunto de $\mathbf{A}$, que es por otro lado la definición estándar de estructura, se genera una lógica *no absoluta*[2] —*i.e.* una lógica cuyo concepto de verdad depende directamente de la teoría de conjuntos subyacente—.

10.2. Sintaxis y Semántica

Así pues, por lo que al alfabeto se refiere, la única diferencia existente entre la de primer orden y la de segundo orden es que en la última añadimos variables para conjuntos y relaciones —se pueden poner las últimas letras mayúsculas del alfabeto latino y utilizar superíndices que indiquen el grado—. Las nuevas variables relacionales aparecerán en fórmulas atómicas; por ejemplo $X\tau$, y en fórmulas cuantificadas del estilo mencionado anteriormente; por ejemplo, $\forall X\varphi$, $\forall X^2\varphi$, etc.

Para presentar el lenguaje formal especificamos primero su signatura

$$\Sigma = \langle VAR, FUNC\rangle$$

—que nos indica cuántas clases de variables tenemos y los tipos de los signos—. Tendremos dos de ellos —SOL y $\lambda - SOL$— siendo su única diferencia el signo de abstracción lambda.

1. Conectores: $\neg, \wedge, \vee, \rightarrow, \leftrightarrow$
   —de tipos: $\langle 0,0\rangle, \langle 0,0,0\rangle$ donde el tipo cero es el de los valores de verdad[3]—
2. Cuantificadores: $\forall, \exists$
3. Abstractor: λ
4. Paréntesis: $), ($
5. Signos de igualdad: E, E_1, E_2

[2] Esto nos permitirá demostrar incompletud en la sección 10.5.

[3] El tipo $\langle 0,0\rangle$ significa que la conectiva asigna valores de verdad a valores de verdad y el $\langle 0,0,0\rangle$, que simplifica a $\langle 0, \langle 0,0\rangle\rangle$, que otorga valores de verdad a pares de valores de verdad.

—de tipos: $\langle 0,1,1\rangle$, $\langle 0,\langle 0,1\rangle\langle 0,1\rangle\rangle$, $\langle 0,\langle 0,1,1\rangle$, $\langle 0,1,1\rangle\rangle$[4]—

6. Falsedad: $\bot$

7. Signos en $OPER.CONS$. Dicho conjunto puede incluir functores y relatores de cualquier grado o ariedad.

8. Variables individuales y predicativas:
   $x, y, z, x_1, x_2, x_3, ...$
   $X_1^1, X_2^1, X_3^1, ...X^1, Y^1, Z^1...$
   —de tipo $\langle 0,1\rangle$—
   $X_1^2, X_2^2, X_3^2, ...X^2, Y^2, Z^2...$
   —de tipo $\langle 0,1,1\rangle$—
   etc...

Con los signos del lenguaje se genera el conjunto de sus expresiones,

$$EXPR = TERM \cup PRED \cup FORM$$

Las novedades respecto de FOL son los predicados con λ

$$\lambda x_1 ... x_n \varphi$$

y la cuantificación de variables predicativas:

$$\forall X^n \varphi$$

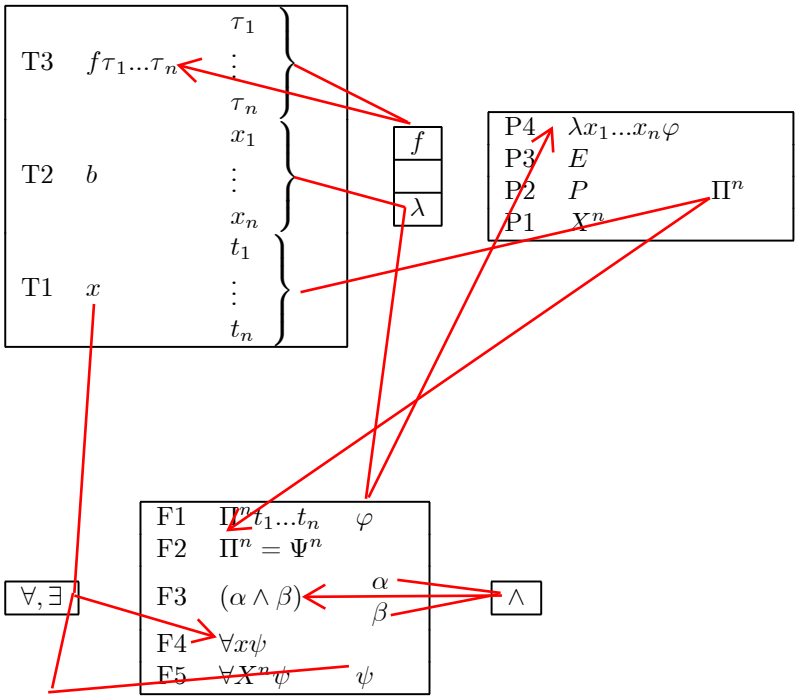
Incluyo tres reglas para formación de términos; a saber, T1: variables, T2: constantes —que son signos en $OPER.CONS$ de tipo 1, que abrevia a $\langle 1\rangle$— y T3: términos functoriales.

Cuatro son las reglas de formación de predicados; a saber, P1: variables, P2: relatores —que son signos en $OPER.CONS$ de tipo $\langle 0,1,...,1\rangle$—, P3: igualdad y P4: predicados con λ.

Cinco son las reglas de formación de fórmulas; F1: atómicas, F2: igualdad entre relatores, F3: conectores booleanos, F4: cuantificación individual y F5: cuantificación predicativa.

La formación recursiva de términos, predicados y fórmulas se puede apreciar en el gráfico:

[4] El tipo $\langle 0,1,1\rangle$ significa que otorga valores de verdad a pares de individuos, mientras que $\langle 0,\langle 0,1\rangle,\langle 0,1\rangle\rangle$ otorga valores de verdad a pares de conjuntos de individuos, y así sucesivamente. Observad que en vez de conjuntos consideramos funciones características.



Semántica estándar

Debemos, por consiguiente, otorgar referencia a las nuevas variables. Una variable de conjunto toma valores en el conjunto de las partes del universo de individuos, mientras que una variable relacional binaria toma valores en el conjunto de las partes del producto cartesiano de dicho universo. Así, en la denominada semántica estándar, en un sistema cuyo universo de individuos sea $\mathbf{A}$, el universo de conjuntos será $\wp(\mathbf{A})$, el de relaciones binarias $\wp(\mathbf{A}^2)$, etc.

Definición 353 *Decimos que* $\mathcal{A}$ *es estándar siempre que*

$$\mathcal{A} = \left\langle \mathbf{A},\ \langle \mathbf{A}_n \rangle_{n\geq 1},\ \left\langle C^{\mathcal{A}} \right\rangle_{C\in OPER.CONS} \right\rangle \tag{10.1}$$

donde:

1. $\mathbf{A} \neq \emptyset$ —*conjunto no vacío como universo de individuos*—
2. $\mathbf{A}_n = \wp(\mathbf{A}^n)$ —*esto es, el conjunto de todas las relaciones* $n-$*arias como universo de relaciones* $n-$*arias*—

   IMPORTANTE!!! *El* concepto de subconjunto *se toma acríticamente de la metateoría de conjuntos, como un concepto* "lógico".
3. *Si* $C := R$ *es un relator* $n-$*ario* —*esto es, de tipo* $\langle 0, 1, ...^n, 1\rangle$— *entonces* $R^{\mathcal{A}} \subseteq \mathbf{A} \times ... \times \mathbf{A}$ *es una relación sobre* $\mathbf{A}$. *O, lo que es lo mismo, su función característica correspondiente* $R^{\mathcal{A}} : \mathbf{A} \times ... \times \mathbf{A} \longrightarrow \{V, F\}$

Si $C := f$ *es un functor* $n-$*ario —esto es, de tipo* $\langle 1, 1, ...^n, 1\rangle$ *— entonces* $f^{\mathcal{A}} : \mathbf{A} \times ... \times \mathbf{A} \longrightarrow \mathbf{A}$ *es una función sobre* $\mathbf{A}$

Definimos asignaciones, para que las variables adquieran valor referencial

$$M : V \cup (\bigcup V_n)_{n\geq 1} \longrightarrow \mathbf{A} \cup (\bigcup \mathbf{A}_n)_{n\geq 1}$$

que respetan el tipo de las variables; esto es: $M(x) \in \mathbf{A}$ y $M(X^n) \in \mathbf{A}_n$

Se definen las interpretaciones

$$\Im = \langle \mathcal{A}, M\rangle$$

Y, basadas en ellas, la *denotación y satisfacción* de términos y fórmulas, de forma similar a como se hace en la lógica de primer orden. Sólo se añade

$$\Im(\lambda x_1 ... x_n \varphi) = \left\{\langle \mathbf{a}_1, ..., \mathbf{a}_n\rangle \mid \Im^{\mathbf{a}_1 ... \mathbf{a}_n}_{x_1 ... x_n} \text{ sat } \varphi\right\}$$

Consecuencia y validez reciben las definiciones y notación usuales:

$$\Gamma \vDash \varphi \text{ y } \vDash \varphi$$

Así que la lógica de segundo orden se nos presenta en un precioso paquete de teoría de conjuntos y con frecuencia, como los niños, pasamos más rato jugando con la caja que con el sofisticado regalo (ver figura: 10.1).

Figura 10.1: *Paquete-regalo*

10.3. Capacidad expresiva

A causa de la cuantificación sobre conjuntos y relaciones la lógica de segundo orden estándar tiene mucho poder expresivo; incluso excesivo, como veremos luego.

A modo de ejemplo:

1. El *Axioma de Inducción* puede formularse del modo siguiente, y retener todo su poder expresivo:

$$\forall X(Xc \wedge \forall x(Xx \rightarrow X\sigma x) \rightarrow \forall x\ Xx)$$

   Esta fórmula dice: *Toda propiedad que valga para el cero y para el siguiente de cualquier número que la tenga, es una propiedad de todos los números.* La aritmética de Peano de segundo orden AP^2 la forman este axioma, junto al de *inyectividad* de la función del siguiente y la exigencia de que el *cero* no sea siguiente de ningún número.

2. La *Identidad entre Individuos* puede introducirse por definición, y no ser, como en la lógica de primer orden un concepto lógico, primitivo; es decir, tomado directamente de la metateoría. La definición comúnmente aceptada es la de Leibniz, que en *SOL* presenta el siguiente aspecto:

$$\forall xy(x = y \leftrightarrow \forall X(Xx \leftrightarrow Xy))$$

   Esta fórmula dice: *"Dos individuos son iguales si, y sólo si, comparten todas sus propiedades".*

3. El concepto intuitivo de *la mayoría de los R son S* —i.e., la mayor parte de los elementos que tienen la propiedad R tienen también la propiedad S —, puede expresarse en lógica de segundo orden con dos relatores monarios para R y S del modo siguiente:

$$\neg\exists X^2(\forall x(\exists y\ X^2xy \leftrightarrow Rx \wedge Sx) \wedge \forall x(\exists y\ X^2yx \rightarrow Rx \wedge \neg Sx)$$
$$\wedge\forall xyz(X^2xy \wedge X^2xz \rightarrow y = z) \wedge \forall xyz(X^2xy \wedge X^2zy \rightarrow x = z))$$

   Esta fórmula dice (ver figura: 10.2): *"no hay ninguna función inyectiva de $R \cap S$ en $R - S$"*. Se acepta que esta formulación logra captar la idea intuitiva de: *"la mayor parte de los R son S"*, puesto que está diciendo que el conjunto $R \cap S$ es *"mayor"* que el conjunto $R - S$.

4. Tanto la *finitud* como la *infinitud* pueden formularse mediante un único enunciado. Por ejemplo, la finitud se escribiría:

$$\forall F(\forall xy(Fx = Fy \rightarrow x = y) \rightarrow \forall x\exists y\ x = Fy)$$

   —o, lo que es lo mismo, cada función inyectiva $f : \mathbf{A} \longrightarrow \mathbf{A}$, sobre la totalidad del universo de individuos $\mathbf{A}$ es también exhaustiva—.

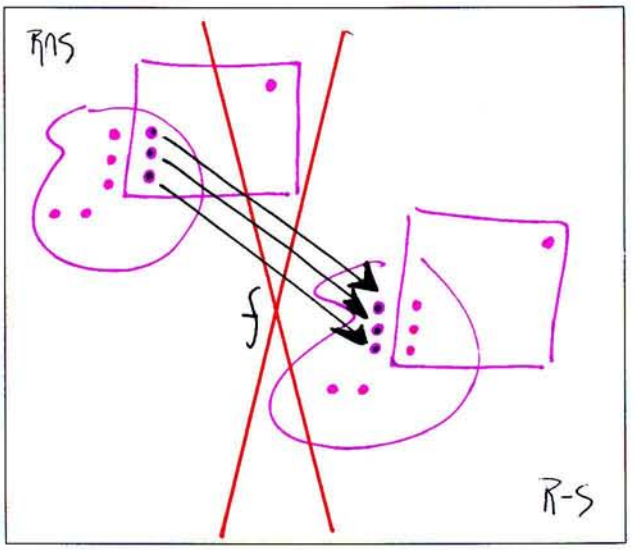


Figura 10.2: *Ninguna función así*

De hecho, estamos empleando variables funcionales en esta definición, pero podríamos sustituirlas fácilmente por variables relacionales, sin más que explicitar que nuestra relación binaria no es más que una función cuyo dominio es todo el universo.
También puede ser modificada para indicar que un conjunto cualquiera del universo es finito —no solamente que lo es la totalidad—.
Por supuesto, para expresar la *infinitud* negamos la fórmula.

$$\varphi_\infty :=_{Df} \exists F(\forall xy(Fx = Fy \to x = y) \wedge \neg\forall x\exists y\ x = Fy)$$

5. Basada en la formulación de finitud del universo o dominio, podemos expresar que un conjunto cualquiera dentro del dominio es finito mediante la fórmula

$$\begin{gathered}\varphi_{fin}(Z) := \forall X^2(\forall x(Zx \leftrightarrow \exists y\ X^2xy) \\ \wedge\forall x(\exists y\ X^2yx \to Zx) \wedge \forall xyz(X^2xy \wedge X^2xz \to y = z) \\ \wedge\forall xyz(X^2xy \wedge X^2zy \to z = x) \to \forall x(Zx \to \exists y\ X^2yx))\end{gathered}$$

que indica: *"Si* $Dom(\mathbf{X^2}) = \mathbf{Z}$, $Rang(\mathbf{X^2}) \subseteq \mathbf{Z}$ *y* $\mathbf{X^2}$ *es función inyectiva, entonces* $Rang(\mathbf{X^2}) = \mathbf{Z}$*"*. Esto es, toda función inyectiva definida sobre $\mathbf{Z}$ es exhaustiva.
Claramente, se verifica que

$$\langle \mathcal{A}, M\rangle \text{ sat } \varphi_{fin}(Z) \text{ syss } M(Z) \text{ es finito}$$

6. Los axiomas del *buen orden* también se formalizan fácilmente. Si $\leq$ es un orden, la fórmula

$$\forall X(\exists y\ Xy \rightarrow \exists u(Xu \wedge \forall z(Xz \rightarrow u \leq z)))$$

indica que todos los subconjuntos no vacíos tienen un primer elemento. Ya vimos en la sección 1.7.1 que para axiomatizar los órdenes nos basta con *FOL*, es para caracterizar a los buenos órdenes para lo que necesitamos *SOL.*

7. El *Axioma de Comprehensión*, que dice que todas las relaciones definibles existen, se formula así

$$\exists X^n \forall x_1 ... x_n (X^n x_1 ... x_n \leftrightarrow \varphi)$$

donde X^n no está libre en φ.

8. La propiedad de *ser numerable* puede ser formulada en segundo orden sin más que afirmar: *"Un conjunto es numerable si y sólo si hay un orden lineal tal que cada elemento tiene a lo sumo un número finito de predecesores".*

$$\varphi_{\leq \omega} := \exists Y(\forall x \neg Yxx \wedge \forall xyz(Yxy \wedge Yyz \rightarrow Yxz) \wedge$$
$$\forall xy(Yxy \vee Yyx \vee x = y) \wedge \forall x \exists X(\varphi_{fin}(X) \wedge \forall y(Xy \leftrightarrow Yyx))$$

Naturalmente, la propiedad de ser supernumerable es la negación de lo anterior

$$\varphi_{>\omega} := \neg \varphi_{\leq \omega}$$

9. Otra propiedad interesante de la lógica de segundo orden estándar es que los *números reales* pueden ser caracterizados hasta isomorfía. Lo único que hay que hacer es tomar los axiomas de primer orden para los cuerpos ordenados y añadirle lo siguiente,

$$\forall ZY(\forall xy(Zx \wedge Yy \rightarrow x \leq y) \rightarrow \exists z \forall xy(Zx \wedge Yy \rightarrow x \leq z \wedge z \leq y))$$

que es una versión simplificada del axioma del corte de Dedekin —que dice que siempre que cortemos a los reales en dos hay un elemento en el corte.—
Esta formulación tiene que funcionar porque sabemos que el cuerpo ordenado de los reales es el único —hasta isomorfismo— cuerpo completo ordenado. Por consiguiente obtenemos una fórmula $\varphi_{\mathbb{R}}$ tal que

$$\mathcal{A} \text{ es un modelo de } \varphi_{\mathbb{R}} \text{ si y sólo si } \mathcal{A} \cong \langle \mathbb{R}, 0, 1, +, \cdot, \leq \rangle$$

10. Incluso la *Hipótesis del Continuo*, CH, puede ser formulada en segundo orden. Semejante φ_{CH} debe decir: *"Si el dominio es de la misma cardinalidad que $\mathbb{R}$, entonces cada subconjunto del mismo es o bien numerable, o de la misma cardinalidad que todo el dominio".* Siendo así

$$\varphi_{CH} \text{ es válida syss } CH$$

Como es bien sabido, esta es la conjetura que el propio Cantor formuló para contestar a los interrogantes relacionados con la cardinalidad de los números reales —también denominados *"el continuo"*—, que de forma esquemática podemos plantear así:
¿Hay cardinalidades intermedias entre $\aleph_0$ —*el primer cardinal infinito, los números naturales*— *y la cardinalidad del continuo,* $|\mathbb{R}|$?
¿Tiene un conjunto de la misma cardinalidad que $\mathbb{R}$ *subconjuntos supernumerables de cardinalidad inferior a la de* $\mathbb{R}$*; es decir, inferior a* $|\mathbb{R}|$?
La cuestión surgió cuando Cantor probó que los reales no son numerables; esto es, $|\mathbb{R}| > \aleph_0$. La hipótesis del continuo CH expresa la respuesta del propio Cantor, que no hay cardinalidades entre $\aleph_0$ y $|\mathbb{R}|$.
Es fácil ver que hay una fórmula de segundo orden φ_{CH} que la formaliza. Queremos que la fórmula φ_{CH} diga: *"cualquier subconjunto de un conjunto cuya cardinalidad sea* $|\mathbb{R}|$ *es numerable o de la misma cardinalidad que los reales"*.
¿Cómo se obtiene φ_{CH}?
Lo primero que hacemos es modificar ligeramente la definición de numerabilidad para con ella poder expresar que un cierto conjunto en el universo es numerable; y no sólo que la totalidad del universo lo sea. Obtenemos así una $\varphi_{\leq\omega}(X)$ tal que

$$\langle \mathcal{A}, M\rangle \text{ sat } \varphi_{\leq\omega}(X) \text{ syss } M(X) \text{ es numerable.}$$

Necesitamos después una fórmula para indicar que el universo del sistema es de la misma cardinalidad que $\mathbb{R}$. Para obtener dicha fórmula eliminamos en $\varphi_{\mathbb{R}}$ los relatores y functores, poniendo en su lugar variables, y cuantificamos existencialmente la fórmula resultante[5]. Con la nueva $\psi_{\mathbb{R}}$ expresamos la propiedad de ser de la misma cardinalidad que $\mathbb{R}$

$$\mathcal{A} \text{ es un modelo de } \psi_{\mathbb{R}} \text{ syss } \mathcal{A} \text{ es de la misma cardinalidad que } \mathbb{R}$$

La hipótesis del continuo tiene también esta lectura:

$$2^{\aleph_0} = \aleph_1$$

—puesto que $|\mathbb{R}| = 2^{\aleph_0} = |\wp(\aleph_0)|$ y tomamos $\aleph_1$ como el primer ordinal después de $\aleph_0$—

11. La *hipótesis generalizada del continuo* GCH dice: $2^{\aleph_\beta} = \aleph_\alpha$, para cada α, $\alpha = \beta + 1$. GCH puede expresarse fácilmente diciendo: *"entre la cardinalidad de un conjunto infinito cualquiera y la del conjunto de sus partes no hay cardinalidades intermedias"*.

[5] Aunque el lenguaje no tenga variables funcionales se puede hacer lo mismo con variables relacionales añadiéndosele la condición que expresa que son funcionales y que el dominio cubre todo el universo.

También la hipótesis generalizada del continuo se puede formular en *SOL* mediante φ_{GCH}

$$\varphi_{GCH} :=_{Df} \forall XY(\inf(X) \wedge Y \sim \wp(X) \rightarrow \forall Z(Z \subseteq Y \rightarrow Z \preceq X \vee Z \sim Y))$$

que puede escribirse completamente en segundo orden. Esta fórmula dice explícitamente: *"Cada subconjunto* $\mathbf{Z}$ *de un conjunto* $\mathbf{Y}$ —esto es, $\mathbf{Z} \subseteq \mathbf{Y}$—, *que sea equipotente con el conjunto potencia de un conjunto infinito* $\mathbf{X}$ —es decir, $\mathbf{Y} \sim \wp(\mathbf{X})$—, *es equipotente con dicho conjunto potencia* —esto es, $\mathbf{Z} \sim \mathbf{Y}$— *o de igual o menor potencia que el referido conjunto infinito —así que* $\mathbf{Z} \preceq \mathbf{X}$—"
La única parte que no tiene una formulación tan obvia en *SOL* es

$$\mathbf{Y} \sim \wp(\mathbf{X})$$

Para hacerlo introducimos una relación binaria $\mathbf{U}$ entre $\mathbf{X}$ y $\mathbf{Y}$ que contiene, en un cierto sentido, una función biyectiva de $\wp(\mathbf{X})$ en $\mathbf{Y}$: A cada subconjunto $\mathbf{Z}$ de $\mathbf{X}$ le corresponde un único elemento de $\mathbf{Y}$ que es la imagen mediante la relación $\mathbf{U}$ de todos los elementos de $\mathbf{Z}$. Además, el recorrido de $\mathbf{U}$ es $\mathbf{Y}$. Todo esto se expresa mediante la fórmula (ver figura: 10.3)

$$\exists U(U \subseteq X \times Y \wedge \forall Z(Z \subseteq X \rightarrow \exists! z(Yz \wedge \forall u(Zu \leftrightarrow Uuz) \wedge Y = Rec(U))))$$

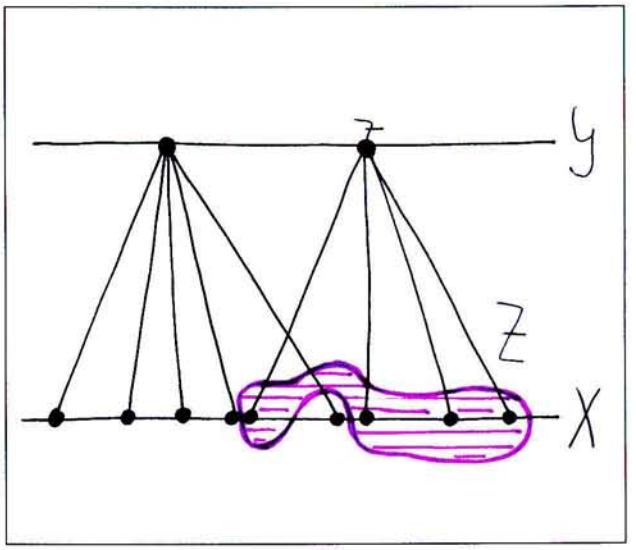


Figura 10.3: *Biyectabilidad,* $\mathbf{Y} \sim \wp(\mathbf{X})$

Puesto que ni GCH ni su negación $\neg GCH$ pueden probarse en la teoría de conjuntos de Zermelo-Fraenkel, la validez de la fórmula de segundo orden que la expresa no puede ni establecerse ni refutarse en el marco de la teoría de conjuntos de Zermelo-Fraenkel. Es por esto por lo que digo que el poder expresivo de la lógica de segundo orden es desmesurado.

Comentario 354 *Un lenguaje que pueda expresar más de lo que la teoría de conjuntos de Zermelo-Fraenkel pueda decidir no es* ***estable****. Cuando esto sucede no hay esperanza de encontrar un cálculo deductivo completo para la lógica asociada a dicha semántica. En una sección próxima veremos cómo utilizando este hecho se puede demostrar la incompletud de la lógica de segundo orden con semántica estándar.*

No hace falta ir a la matemática para encontrar ejemplos de pensamientos para los que se precisa la lógica de segundo orden como lenguaje para formularlos. He aquí algunos ejemplos coloquiales:

1. *"Hay gente para todo".*
$$\forall X \exists y \; Xy$$

2. *"Hay al menos una característica compartida por todos los regímenes autoritarios, tanto de izquierdas como de derechas"*. Puede formularse así:
$$\exists X \forall z (Az \wedge (Lz \vee Rz) \rightarrow Xz)$$

3. *"Hay mujeres capaces de estar enamoradas de hombres muy diferentes, que no comparten cualidad alguna"*. Podemos seleccionar esta formalización
$$\exists x (Mx \wedge \exists z \exists y (Hz \wedge Hy \wedge z \neq y \wedge Axz \wedge Axy \wedge \neg \exists X (Xz \wedge Xy)))$$
El problema es que la mayor parte de ellos son trivialmente verdaderos u obviamente falsos porque el sentido intuitivo es algo más sutil; la cuantificación no es total, elimina obviedades.

4. Si decimos *"Napoleón tiene todas las cualidades de un buen general"*, tenemos problemas cuando intentamos formalizarlo así
$$\forall X (\exists x (Gx \wedge Xx) \rightarrow Xn)$$
porque todo, incluso los buenos generales, tienen la propiedad de ser ellos mismos y Napoleón no puede ser ningún otro general. (Pensad en lo que sucede cuando la propiedad es una clase unitaria cuyo miembro es un general, otro que Napoleón.)

5. Cuando tomamos *"Matemáticos y filósofos comparten al menos una propiedad"*, y lo formalizamos así
$$\forall x \forall y (Mx \wedge Py \rightarrow \exists Z (Zx \wedge Zy))$$
quedamos también insatisfechos porque lo formalizado es trivialmente verdadero. (Pensad en la propiedad de ser o bien un matemático o bien un filósofo.)

10.4. Propiedades metalógicas

Como subproducto del poder expresivo de la lógica de segundo orden obtenemos las siguientes contrapartidas de teoría de modelos:

1. AP^2 es categórica. Es decir, la aritmética de Peano formulada en segundo orden es categórica o, lo que es lo mismo, dos modelos cualesquiera de Peano son isomorfos.
   (La demostración original la hizo Dedekind[6].)

   $$\mathcal{A},\ \mathcal{B} \in \mathbf{Mod}(AP^2) \Rightarrow \mathcal{A} \cong \mathcal{B}$$

   (Donde AP^2 es la aritmética de Peano de segundo orden.)

2. La lógica de segundo orden no es compacta, esto es, falla el teorema.
   Este resultado es una consecuencia directa del hecho de que la lógica de segundo orden sea capaz de expresar infinitud. Pensad en el conjunto infinito de fórmulas

   $$\Phi := \{\varphi_n / \ n \geq 2\}$$

   diciendo que hay al menos n elementos en el universo

   $$\varphi_n := \exists x_1 ... x_n \bigwedge_{i \neq j} x_i \neq x_j$$

   y en la fórmula φ_∞ que expresa que el universo es infinito. Fácilmente se comprueba que $\Phi \models \varphi_\infty$ pero que $\Gamma \not\models \varphi_\infty$, para cada subconjunto finito Γ de Φ.

3. *El teorema de Löwenheim-Skolem también falla.*
   Este resultado se sigue del hecho de que el concepto de supernumerabilidad puede ser expresado en segundo orden: la fórmula que indica que el universo es supernumerable no puede tener modelos numerables, como se seguiría conforme al mencionado teorema de Löwenheim-Skolem.

4. Por consiguiente, en lógica de segundo orden estándar *jamás encontraremos un cálculo completo en sentido fuerte* —esto es, que cumpla: si $\Gamma \models \varphi$, entonces $\Gamma \vdash \varphi$—. La razón es que compacidad, que se demuestra fácilmente a partir de completud fuerte, no es un metateorema de *SOL*

5. Pero sabemos aún más, el conjunto de las fórmulas válidas es tan intratable que también *completud en sentido débil falla* —esto es, no se cumple: Si $\models \varphi$, entonces $\vdash \varphi$—. Este resultado se sigue del teorema de incompletud de Gödel junto al primer apartado de esta sección, la categoricidad de la aritmética; no obstante, nuestra demostración será otra.

Comentario 355 *Aunque estos teoremas son propios de* SOL *pueden usarse como test de completud, compacidad y demás en otras lógicas.*

[6] La demostración detallada se encuentra en el capítulo tercero de [37].

Comentario 356 *Pero no necesitamos el teorema de Gödel para darnos cuenta de que un cálculo deductivo completo es aquí inalcanzable. La observación se hace pensando en fórmulas como* GCH*, fórmulas cuya validez depende de la metateoría de conjuntos que tomemos. ¿ Cómo se podría definir un cálculo para generar un conjunto cambiante de fórmulas*[7] *?*

Ingenuamente se podría pensar que añadiendo GCH a los axiomas se podría reparar la situación. Pero, por el teorema de Gödel, sabemos que no es ése el caso. No es posible dar un conjunto completo de axiomas para la teoría de conjuntos; esto es, tal que para cada fórmula φ del lenguaje de teoría de conjuntos o ella o su negación $\neg\varphi$ es derivable a partir del conjunto de axiomas. De hecho, hay un recurso inagotable de fórmulas como GCH.

10.5. Incompletud de la lógica de segundo orden

10.5.1. Presuposiciones, conceptos clave y resultados previos utilizados en nuestra demostración

En el próximo apartado haré una demostración esquemática de la incompletud de SOL basada, como la de Gödel, en la capacidad expresiva de la lógica de segundo orden. Para poder realizar la prueba hemos de ser conscientes de los presupuestos ontológicos y semánticos que necesariamente aceptamos cuando asumimos la lógica clásica. Además, en la prueba se emplean resultados, trucos y técnicas de la teoría de conjuntos. Finalmente, usamos el poder expresivo de la lógica de segundo orden y una presentación de SOL en teoría de conjuntos especialmente creada para esta demostración. Así que la prueba se basa en los puntos siguientes:

1. *Supuestos ontológicos.* En particular, admitimos que estamos situados en un *universo matemático* que constituye nuestro entorno y que introducimos nuestro lenguaje formal para hablar acerca de los sistemas o estructuras en él situados. No es necesario admitir la existencia de un único universo matemático en nuestra cosmología, pero tenemos que aceptar que sólo uno constituye nuestra referencia inmediata en un momento dado y que cuando establecemos la semántica de nuestras fórmulas sólo se puede hablar sobre los conjuntos situados en alguno de sus sistemas. Además, no es posible tener como dominio al universo en su totalidad; no se puede cuantificar sobre todo él. $\mathcal{U} = \langle \mathcal{V}, \in^{\mathcal{U}} \rangle$ no es un sistema, en parte porque $\mathcal{V}$ no es un conjunto. El motivo de esta restricción es que utilizamos la semántica de Tarski; es decir, distinguimos perfectamente entre lenguaje objeto y metalenguaje. Y esto es así porque no queremos contradecir el teorema de Church de la indefinibilidad de la verdad —que es una reproducción de la paradoja del mentiroso—.

[7] Ya lo decía Heráclito: *"No es posible decir nada verdadero acerca de las cosas que cambian."*

2. *Teoría de conjuntos subyacente* (ver figura: 10.4). Aceptamos la de *Zermelo-Fraenkel* con axioma de elección ZFC como fundamentación matemática de las pruebas que realizamos en el metalenguaje. Los axiomas de ZFC están escritos en un lenguaje de primer orden L^{ϵ} cuyo único signo peculiar es el relator binario de pertenencia, $\in$. Este lenguaje formal no es nuestro lenguaje objeto, es parte del metalenguaje y sus fórmulas pueden ser consideradas como meras abreviaturas de las expresiones correspondientes en español. Puesto que son parte del metalenguaje, son verdaderas o falsas y cuando cuantificamos sobre conjuntos, lo hacemos sobre la clase de todos los conjuntos. Observamos que las fórmulas de ZFC son fórmulas de primer orden y que para *FOL* tenemos muy desarrollada la maquinaria deductiva pues disponemos de un cálculo correcto y completo así como de una cantidad considerable de resultados de *Teoría de Modelos.*

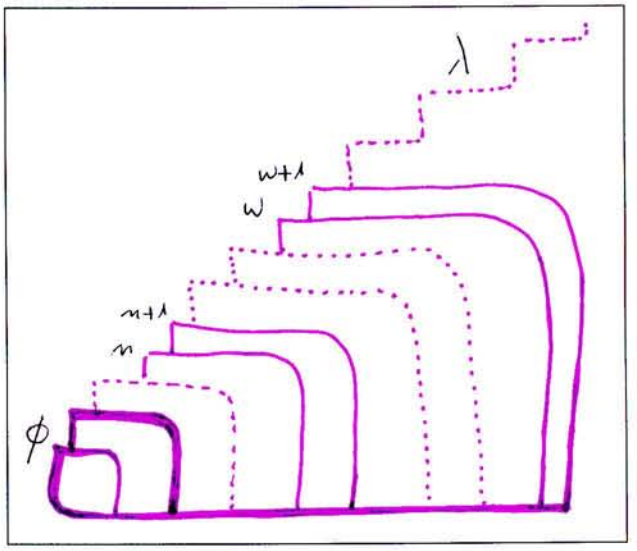


Figura 10.4: *Escalera de conjuntos*

Pero, *¿Se puede definir verdad y consecuencia del metalenguaje en el mismo metalenguaje?, ¿no estaremos cayendo sin querer en el abismo de las paradojas?* Además, *¿cómo expresamos el cambio de universo matemático?*

3. *Metateoría de conjuntos.* Lo que podemos hacer es usar ciertos trucos, técnicas y resultados utilizados en la metateoría de conjuntos. Especialmente, las artimañas usadas en las demostraciones de consistencia relativa y los resultados de Gödel y Cohen acerca de la independencia de la hipótesis del continuo y de su negación de ZFC. Esto quiere decir, que si suponemos la consistencia de ZFC habrá modelos de $ZFC + \neg GCH$ (Cohen) y

también modelos de $ZFC + GCH$ (Gödel).
En particular, podemos considerar que ZFC es simplemente una teoría de primer orden y que por lo tanto, al haber formalizado el metalenguaje, éste se convierte en lenguaje objeto. Por lo que necesitamos un nuevo metalenguaje, más potente. Pensamos que nuestro mundo ahora contiene clases y que L^{ϵ} es un lenguaje formal para referirnos a este nuevo mundo. En particular lo que hacemos es tomar perspectiva, situarnos a un nivel superior. Tomamos la teoría de conjuntos de Gödel, Bernays y Neumann GBN como fundamentación matemática en el metalenguaje. Este procedimiento es utilizado en pruebas de consistencia relativa. Para no correr riesgos, usamos un cálculo deductivo correcto, pero no utilizamos el teorema de completud para él. La diferencia entre GBN y ZFC es importante: en GBN podemos aceptar sistemas cuyos universos son clases, incluso puede admitirse como sistema a

$$\mathcal{U} = \langle \mathcal{V}, \in^{\mathcal{U}} \rangle$$

donde $\mathcal{V}$ es la clase de todos los conjuntos que consideramos fija en un momento dado, aunque desconocida. $\mathcal{U}$ no es un sistema en el sentido usual, porque su universo no es un conjunto, sino una clase última, pero puede ser considerado como tal en GBN. GBN se convierte así en el metalenguaje para hablar de ZFC. (ver figura: 10.5)

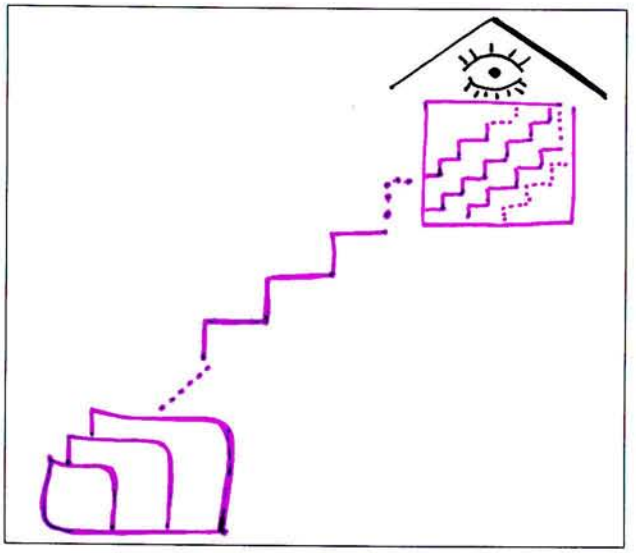


Figura 10.5: *Divina perspectiva*

4. Usaremos también la enorme y abrumadora capacidad expresiva de la lógica de segundo orden, especialmente su habilidad para expresar pro-

piedades del universo matemático. En particular, las que nos permiten distinguir un universo matemático de otro; esto es, fórmulas que son independientes de ZFC[8]. Nosotros nos servimos en la prueba de la fórmula φ_{GCH} que expresa la hipótesis generalizada del continuo. En particular, se usa el hecho

$$\models_{\mathfrak{S}.\mathfrak{S}} \varphi_{GCH} \text{ syss } GCH$$

(Es decir, La fórmula de segundo orden φ_{GCH} es válida en estructuras estándar si y sólo si GCH vale en *"el"* universo matemático; esto es, en el que estemos.)

5. En la demostración se desarrolla la lógica de segundo orden en el marco de la teoría de conjuntos: Las fórmulas de segundo orden se asimilan a conjuntos y podemos, consiguientemente, hablar de ellas en el lenguaje conjuntista L^ϵ de la misma forma en la que hablamos de cualquier otro conjunto; Los conceptos de sistema estándar y los de satisfacibilidad y validez se introducen fácilmente en este mismo marco; Se puede describir el concepto de secuente y el de deducibilidad de SOL y desarrollar la sintaxis de SOL en él. Finalmente, incluso el teorema de completud de SOL puede formularse mediante una sentencia de L^ϵ. Caso de ser completa, la prueba de completud puede hacerse con recursos limitados, usando la base axiomática de teoría de conjuntos, ZFC.

6. Pero no sólo sucede que los recursos de teoría de conjuntos usados para desarrollar la sintaxis no son muy profundos, sino que también las fórmulas de L^ϵ que expresan derivabilidad son extremadamente simples; son fórmulas cuya verdad no se alteraría por el paso de un universo matemático a otro que contuviera más conjuntos.

Teorema 357 *(Incompletud de SOL). No existe ningún cálculo correcto y completo para la lógica de segundo orden estándar (suponiendo que ZFC sea consistente).*

Demostración. *Vamos a suponer que ZFC sea consistente y que tenemos un cálculo correcto* CAL *para* SOL*. Aceptemos también los supuestos iniciales mencionados anteriormente. Esquemáticamente, la demostración es como sigue:*

1. *ZFC es consistente*

2. *Hay un modelo $\mathcal{U} = \langle \mathcal{W}, \in^{\mathcal{U}} \rangle$ de $ZFC + \neg GCH$ (Cohen)*

   Sea $\mathcal{L} \subseteq \mathcal{W}$ su parte constructible. Entonces, por el teorema de Gödel,

3. *$\mathcal{L} = \langle \mathcal{L}, \in^{\mathcal{L}} \rangle$ es un modelo de GCH*
   —donde $\mathcal{L} \sqsubseteq \mathcal{U}$; esto es, $\in^{\mathcal{L}} = \in^{\mathcal{U}} \cap (\mathcal{L} \times \mathcal{L})$—

[8] Yo no sé si esto será muy decepcionante para los cantorianos convencidos, pero yo identifico a los distintos universos matemáticos con los diversos conjuntos máximamente consistentes que extienden ZFC. Suponemos, como es habitual, que ZFC es consistente.

4. $\models_{\mathcal{S}.\mathcal{S}} \varphi_{GCH}$ *syss* GCH *es verdadera en "el" universo matemático (capacidad expresiva de* SOL *con semántica estándar)*

   Por consiguiente, mientras que $\mathcal{L}$ *sea nuestro universo matemático,*

5. $\models_{\mathcal{S}.\mathcal{S}} \varphi_{GCH}$ *(de los apartados 3 y 5)*

   Usando el lenguaje L^{ϵ},

6. $\mathcal{L}$ *es un modelo de* $\ulcorner \models_{\mathcal{S}.\mathcal{S}} \varphi_{GCH} \urcorner$

7. *El cálculo* CAL *para* SOL *no sólo es correcto, sino también completo. (supuesto inicial)*

   Por lo tanto,

8. $ZFC \vdash \ulcorner$**CAL** *es completo con* $\mathcal{S}.\mathcal{S}\urcorner$ *(por 7 y porque hemos hecho un desarrollo conjuntista de* SOL*)*

   Por consiguiente, si todas las fórmulas válidas son derivables, eliminando el cuantificador, obtenemos

9. *Un teorema de* ZFC

$$ZFC \vdash \ulcorner \models_{\mathcal{S}.\mathcal{S}} \varphi_{GCH} \urcorner \rightarrow \ulcorner \vdash_{CAL} \varphi_{GCH} \urcorner$$

   Puesto que $\mathcal{L}$ *es un modelo de* ZFC *—por el teorema de Gödel— y suponemos corrección en el metalenguaje,*

10. $\mathcal{L}$ *es modelo de* $\ulcorner \models_{\mathcal{S}.\mathcal{S}} \varphi_{GCH} \urcorner \rightarrow \ulcorner \vdash_{CAL} \varphi_{GCH} \urcorner$

    Luego,

11. $\mathcal{L}$ *es modelo de* $\ulcorner \vdash_{CAL} \varphi_{GCH} \urcorner$

    Ahora es cuando se da el paso crucial:

12. $\mathcal{U}$ *es modelo de* $\ulcorner \vdash_{CAL} \varphi_{GCH} \urcorner$

13. CAL *es correcto (supuesto inicial)*

14. $ZFC \vdash \ulcorner$**CAL** *es correcto con* $\mathcal{S}.\mathcal{S}\urcorner$

    Por lo tanto,

15. $ZFC \vdash \ulcorner \vdash_{CAL} \varphi_{GCH} \urcorner \rightarrow \ulcorner \models_{\mathcal{S}.\mathcal{S}} \varphi_{GCH} \urcorner$

    Puesto que $\mathcal{U}$ *es un modelo de* ZFC *y aceptamos corrección en el metalenguaje obtenemos:*

16. $\mathcal{U}$ *es un modelo de* $\ulcorner \ulcorner \vdash_{CAL} \varphi_{GCH} \urcorner \rightarrow \ulcorner \models_{\mathcal{S}.\mathcal{S}} \varphi_{GCH} \urcorner$

    Luego,

17. $\mathcal{U}$ *es un modelo de* $\ulcorner \models_{\mathcal{S}.\mathcal{S}} \varphi_{GCH} \urcorner$

Por consiguiente, cuando nuestro universo sea $\mathcal{U}$,

18. $\models_{\mathcal{S}.\mathcal{S}} \varphi_{GCH}$

Pero $\mathcal{U}$ *no es un modelo de* GCH*, nosotros hemos partido de un modelo obtenido mediante forcing que demostraba el teorema de Cohen. Por consiguiente, siempre que* $\mathcal{U}$ *sea* "el" *universo matemático,*

19. $\not\models_{\mathcal{S}.\mathcal{S}} \varphi_{GCH}$
Esto constituye una contradicción.

■

Hemos utilizado los resultados conocidos de Cohen y Gödel y hemos generado una contradicción: Dependiendo del Universo matemático en el que nos movamos la hipótesis del continuo es verdadera o no, y la fórmula φ_{GCH} de segundo orden que la expresa es válida o no. Esto no constituye en sí misma una contradicción, pero si SOL tuviera un cálculo completo, dicha fórmula habría de ser derivable en un caso y no derivable en el otro. Pero esto es imposible porque el concepto de derivabilidad es tan simple que no debiera afectarle el cambio de universo matemático. De aquí nace, justamente, la contradicción que prueba la incompletud de SOL.

Es tal vez importante insistir en que la contradicción es que φ_{GCH} debe ser válida y no válida en el mismo universo matemático $\mathcal{U}$. Sabíamos desde el principio que mientras estuviéramos en $\mathcal{U}$ la fórmula φ_{GCH} de SOL no sería válida pero que en su parte constructible $\mathcal{L}$ sí que lo sería; esto no es ninguna contradicción. Esto no es más que una obviedad que viene siendo destacada desde antiguo: Henkin, Church, Quine, etc. son conscientes de que con la lógica de segundo orden traspasamos la frontera de la teoría de conjuntos. En los libros de texto de Enderton, Ebbinghaus-Flum-Thomas y en el artículo de van Benthem y Doets del Handbook of Mathematical Logic ya se apuntan algunos problemas de esta índole. Evidentemente, sin ser una contradicción, es un aviso de que algo falla con la semántica estándar. La contradicción de verdad llega cuando se supone la completud de SOL. Usándola podemos dar el salto desde validez a derivabilidad y entonces la conexión entre los modelos de Cohen y Gödel se consigue gracias a la propiedad de *persistencia* de la fórmula que expresa demostrabilidad.

Comentario 358 *Este procedimiento es fácilmente exportable; toda lógica capaz de expresar conceptos no absolutos, independientes de* ZFC*, tiene que ser incompleta pues la fórmula de esta lógica que expresa su validez no puede ser equivalente a la que expresa su derivabilidad.*

10.5.2. Conclusión

¿Cuál es la conclusión de todo esto?

Hay una lección que deberíamos aprender y que tiene varias lecturas:

1. La primera nos retrotrae a la imagen de la balanza —recordad el comentario 49— y nos dice: *"Es evidente que no puedes tener las dos cosas a la vez; esto es, poder expresivo y buenas propiedades lógicas"*. De hecho, sabemos por el teorema de Lindström que la lógica de primer orden es la lógica más potente, de una amplia familia, que verifica simultáneamente completud, compacidad y Löwenheim-Skolem.

2. Hay una lectura más liberal según la cual admitimos que hemos cometido diversos errores al definir la semántica estándar para SOL. En nuestras estructuras o sistemas estándar tomamos el conjunto de las partes del universo de individuos como universo de conjuntos y el de las partes del producto cartesiano del universo de individuos como universo de relaciones. Al hacerlo la noción de *subconjunto* es la de la metateoría de conjuntos —la estamos tratando como un concepto *"lógico"*, de la misma forma acrítica con la que se toma a la identidad en la lógica de primer orden, y por consiguiente es la del metalenguaje—. El problema es que la categoría de ser un subconjunto es muy poco descriptiva, muy laxa, y terminamos en una *lógica no absoluta*. Pero, bien mirado, la propia semántica estándar puede ser considerada como una especie de error.

3. Tras meditar sobre el argumento anterior concluimos que la incompletud de la lógica de segundo orden con semántica estándar nada tiene que ver con la naturaleza del razonamiento de segundo orden, sino con el modo en el que ha sido construido el *"modelo"* de razonamiento de segundo orden en esta semántica. Sin advertirlo, hemos ligado la metateoría de conjuntos ZFC a la semántica de segundo orden, que es nuestro lenguaje en estudio. Los efectos secundarios que se han producido en consecuencia no están relacionados con la naturaleza del fenómeno sino con el modo de construir el modelo[9].

4. Este tipo de consideraciones nos llevan a la necesidad de dar *versiones absolutas* de las lógicas que no lo sean. El pionero fue Henkin, que en 1949 dio una versión semejante para la lógica superior. Y es así como los resultados se invierten para obtener finalmente uno feliz[10]: *Podemos hacer que SOL sea una lógica completa modificando la semántica*. Esto es justamente lo que hacemos en la siguiente sección.

[9] Se podría establecer un paralelismo con el fenómeno que se produce en las ciencias experimentales cuando el observador incide sobre lo observado.

[10] Se me podría objetar que de fin feliz nada y que el precio de la completud es muy alto, pero *¿quién quiere hablar ahora de precios?*

10.6. Completud de *SOL* con semántica general

En [37] expongo con detalle una prueba de completud basada en la de Henkin de completud de la teoría de tipos. Aunque es una demostración de sobra conocida, pues es básicamente la misma que se usa en primer orden[11], voy a detenerme algo en los supuestos que la posibilitan.

Habíamos dicho —en la sección 2.4— que una lógica puede identificarse con el conjunto de sus fórmulas válidas, que al ser verdaderas en toda estructura, no pueden estar describiendo a ninguna en partículas, *describen a la lógica.* De manera que, si somos capaces de generarlas mediante un *cálculo deductivo*, habremos captado la esencia de la lógica, su *"perfume"* —en el sentido bárbaro y radical de la novela de Suskin—. A modo de inciso, tened en cuenta que la lógica —y muy en especial, la lógica de orden superior— es un paquete con un contenido, que es el que nos interesa —queremos llevarnos el gato al agua— y un continente o contenedor, prescindible. El continente lo proporciona la teoría de conjuntos, que actúa como base. (Pero, como diría Mikael Ende, *esto es otra historia, que debe ser contada en otro lugar.*)

Volvamos al cálculo. El proceso habitual en cualquier lógica es introducir primero su gramática mediante reglas de formación de fórmulas y a continuación su semántica, para pasar a definir un cálculo deductivo que genere como teoremas a las fórmulas válidas. Entonces contamos con dos procedimientos de selección de fórmulas: las reglas del cálculo deductivo, capaces de generar el conjunto de sus teoremas lógicos TEO, y la semántica, que nos ayuda a seleccionar a las fórmulas válidas VAL.

¿Coinciden estos conjuntos?

Cuando $TEO \subseteq VAL$ decimos que el cálculo es correcto, y cuando $VAL \subseteq TEO$ decimos que es completo —en sentido débil—. Desgraciadamente, para la lógica de segundo orden lo último nunca es cierto ya que hay fórmulas válidas que no pueden derivarse en ningún cálculo. Acabamos de demostrar que el conjunto de las fórmulas válidas en la semántica estándar es demasiado grande, inabarcable. Por consiguiente, con ella el conjunto de las fórmulas válidas en SOL nunca podrá ser generado mediante un cálculo deductivo finito (ver figura: 10.6).

Pero incluso sabiendo que no puede existir ninguno capaz de obtener como teoremas todas las fórmulas válidas —*i.e.*, las fórmulas de SOL verdaderas en todo modelo estándar—, podemos definir cálculos que sean correctos. Este hecho nos proporciona mayor libertad a la hora de definirlos, ya que sabemos que ninguno podrá ser completo. No obstante, en la lógica de segundo orden hay dos cálculos que tienen un cierto *"pedigree"*; llamémosles MSL —pues es esencialmente el de la lógica multivariada— y SOL. MSL es la extensión simple del de FOL, lo obtenemos al extender las reglas con cuantificadores para que den cuenta de las nuevas fórmulas, que son ahora fórmulas de segundo

[11] Curiosamente, la demostración de Henkin de la completud para la teoría de tipos precede su prueba de completud de la lógica de primer orden.

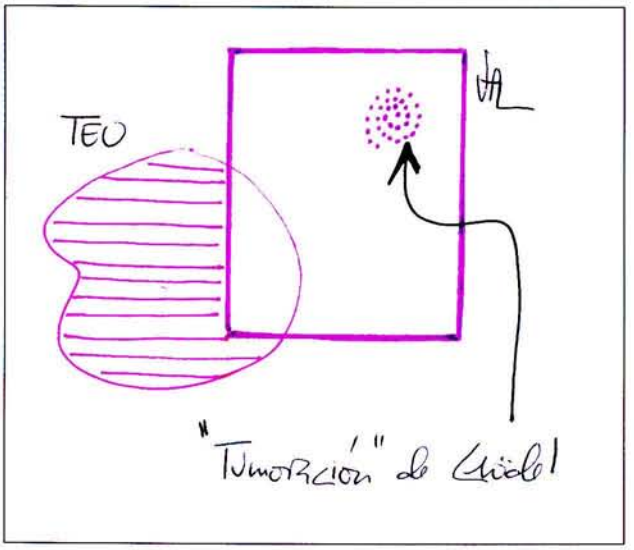


Figura 10.6: *Las fórmulas válidas están fuera de control*

orden y tienen variables predicativas cuantificadas. El cálculo *SOL* lo definimos añadiendo al de *MSL* el esquema de comprehensión,

$$\exists X^n \forall x_1 ... x_n (X^n x_1 ... x_n \leftrightarrow \varphi)$$

—con X^n no libre en φ—

Naturalmente ambos son incompletos, si tomamos como referencia la semántica estándar. Pero si dejamos abierto el concepto de *"conjunto de elementos de* $\mathbf{A}$*"*; es decir, si lo tomamos como un concepto a fijar para cada estructura, la situación cambia radicalmente. Esto fue justamente lo que hizo Henkin en 1950. La idea es muy sencilla: El conjunto de las fórmulas válidas es tan grande porque el concepto de estructura estándar es demasiado restrictivo. Si aceptamos como estructuras adecuadas para *SOL* a las estructuras no-estándar —*i.e.*, estructuras idénticas a las de la definión 353 pero en donde $\mathbf{A}_n \subseteq \wp(\mathbf{A}^n)$; es decir, no se da necesariamente la igualdad—, el conjunto de las fórmulas válidas se reduce. De hecho, obtenemos un primer resultado de completud para el cálculo *MSL* tomando esta semántica, que podemos denominar con Henkin *semántica de marcos.* (ver figura: 10.7)

No obstante, si la selección de las relaciones que vayamos a incluir en $\mathbf{A}_n$ es dejada completamente al arbitrio, el axioma de comprehensión podría fallar; es decir, algunos relaciones definibles mediante fórmulas del lenguaje podrían haberse quedado fuera. Para que el cálculo *SOL* sea correcto, necesitamos la semántica de *modelos generales* de Henkin; esto es, los universos de la estructura deben estar cerrados bajo definibilidad.

Definición 359 *(mediante la condición de corrección). Sea* $\mathcal{A}$ *un marco. De-*

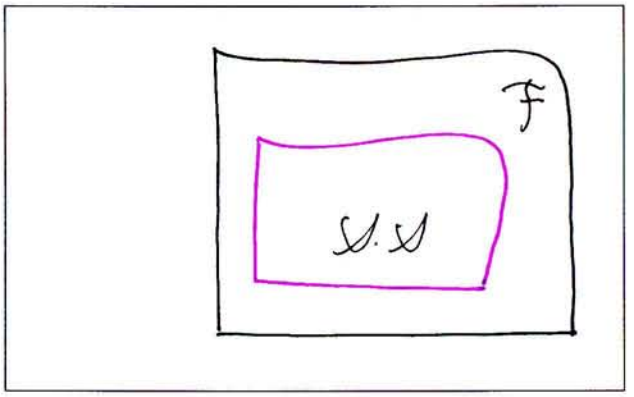


Figura 10.7: *Marcos y estructuras estándar*

cimos que $\mathcal{A}$ *es una* ***estructura general*** *syss es un modelo del conjunto*

$$\Delta = \{\exists X^n \forall x_1 ... x_n (X^n x_1 ... x_n \leftrightarrow \varphi) \mid \varphi \in FORM, \text{ con } X^n \text{ no libre en } \varphi\}$$

formado por todas las instancias del axioma de comprehension.

Definición 360 *(mediante la condición de clausura mediante definibilidad). Sea* $\mathcal{A}$ *un marco. Decimos que* $\mathcal{A}$ *es una* ***estructura general*** *syss todas las relaciones paramétricamente definibles en* $\mathcal{A}$ *con el lenguaje* L_2 *están en el universo correspondiente de la estructura; es decir*

$$A_n = \wp(A^n) \cap PARAM.DEF(\mathcal{A}, L_2)$$

Las dos definiciones son equivalentes, como muestro en [37], página 166. La situación ahora es la de este nuevo esquema (ver figura: 10.8):

Hay una demostración de la completud de *SOL* con semántica de modelos generales prácticamente idéntica a la de la lógica de primer orden univariada, ya expuesta en la sección 2.4.

Podemos también demostrar la completud de *SOL* bajo esta semántica, simplemente usando la completud de *MSL*[12]. Como veremos en la sección 13.3, la suerte es que la propiedad de ser una estructura general puede ser axiomatizada mediante un conjunto de fórmulas Δ que incluye básicamente los axiomas de comprehensión. Esto es, tenemos,

$$\vDash_{\mathfrak{G}.\mathfrak{S}} \varphi \Longleftrightarrow \Delta \vDash_{\mathfrak{F}} \varphi$$

(donde $\mathfrak{F}$ es la clase de los marcos y $\mathfrak{G}.\mathfrak{S}$ la de las estructuras generales)

En realidad, al cambiar la semántica de modelos estándar por la de modelos generales, *hemos cambiado de lógica.* El planteamiento que se hace en esta

[12] Es así como la demuestro en [37], en vez de *como* en la lógica de primer orden, *con* la lógica de primer orden multivariada.

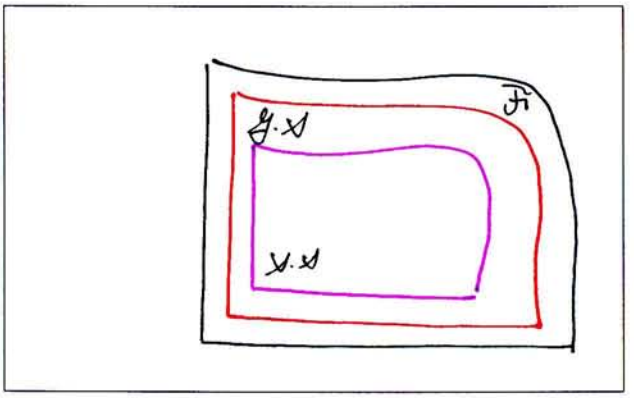


Figura 10.8: *Marcos, modelos generales y estándar*

prueba de completud de la lógica superior —completud respecto de modelos generales, naturalmente— es exportable. De ella aprendemos a modificar convenientemente la clase de modelos admitidos, variándola de forma que las fórmulas válidas en la nueva semántica coincidan con los teoremas lógicos. La verdad es que no se trata, como cabría pensar, de una prueba *"ad hoc"*, ya que la nueva semántica es incluso más natural y razonable que la estándar porque tomamos como *"conjunto de elementos de* $\mathbf{A}$*"* a los que son definibles en $\mathcal{A}$. Comparando las estructuras usadas en la semántica estándar con las de la no-estándar vemos que las primeras son homogéneas —la cuantificación sobre conjuntos y relaciones es un ardiz para acercar lo que sabemos a lo que podemos decir— y el concepto de subconjunto se toma sin más de la metateoría, de la teoría de conjuntos. En la semántica no-estándar sucede lo contrario. El resultado es que en la lógica de segundo orden estándar la balanza se inclina del lado del poder expresivo, en detrimento de las cualidades lógicas mientras que en la no-estándar se equilibra.

Resumen

Siempre que tenemos una clase de sistemas tenemos también el conjunto de las sentencias de SOL en ellas verdaderas. Y cada vez que tengamos un conjunto de sentencias nos podemos preguntar si existe un cálculo capaz de generarlas como teoremas. Tenemos $\mathfrak{S}.\mathfrak{S}$, la clase formada por los sistemas estándar y usándola definimos el conjunto $\models_{\mathfrak{S}.\mathfrak{S}}$ formado por las fórmulas válidas en ella. Sabemos que este conjunto $\models_{\mathfrak{S}.\mathfrak{S}}$ no puede ser generado mediante ningún cálculo, pues no es recursivamente enumerable. Pese a todo, se introducen dos cálculos deductivos, MSL y SOL que generan como teoremas lógicos los conjuntos

$$\vdash_{SOL} \text{ y } \vdash_{MSL}$$

pero si la semántica es la estándar se produce incompletud (ver figura: 10.9)

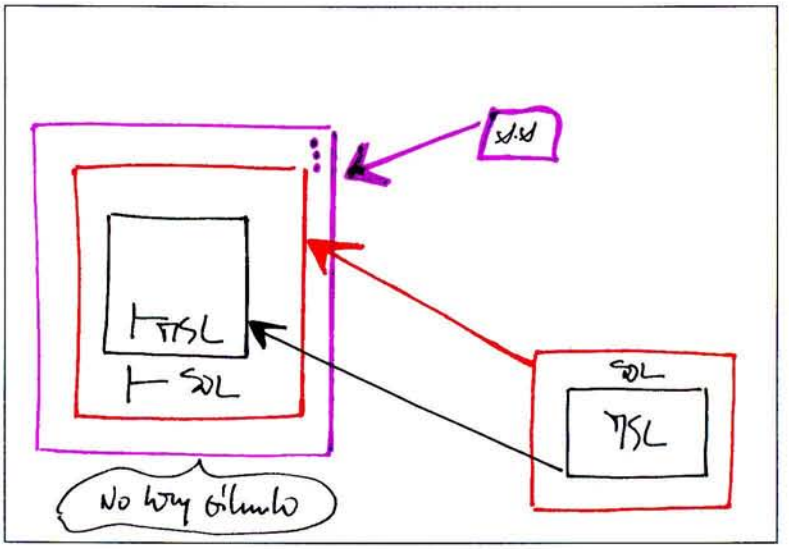


Figura 10.9: *Semántica estándar causa incompletud*

Puesto que $\vdash_{SOL}$ es un subconjunto propio de $\models_{\mathfrak{S}.\mathfrak{S}}$

$$\vdash_{SOL} \subset \models_{\mathfrak{S}.\mathfrak{S}}$$

para conseguir una semántica adecuada necesitamos ampliar la clase de estructuras de manera que se reduzca el conjunto de fórmulas en ellas válidas —cuanto mayor sea la clase, menor será el conjunto porque para ser válida en una clase más amplia necesitan pasar más *"controles de calidad"*—. Definimos tanto a los marcos $\mathfrak{F}$ como a los modelos generales $\mathfrak{G}.\mathfrak{S}$ que producen los conjuntos de fórmulas en ellos válidas $\models_{\mathfrak{F}}$ y $\models_{\mathfrak{G}.\mathfrak{S}}$ y sucede que coinciden exactamente con los conjunto de teoremas anteriores. Concretamente,

$$\models_{\mathfrak{F}} = \vdash_{MSL} \quad \text{y} \quad \models_{\mathfrak{G}.\mathfrak{S}} = \vdash_{SOL}$$

Es fácil comprobar que puesto que

$$\mathfrak{S}.\mathfrak{S} \subseteq \mathfrak{G}.\mathfrak{S} \subseteq \mathfrak{F}$$

tenemos

$$\models_{\mathfrak{F}} \subseteq \models_{\mathfrak{G}.\mathfrak{S}} \subseteq \models_{\mathfrak{S}.\mathfrak{S}}$$

Dejadme sintetizar con un diagrama (ver figura: 10.10):
Estas son las ventajas más importantes:

1. Conseguimos así un *teorema de completud*
2. Sirve para destacar no sólo a la lógica *MSL* sino también a *otras lógicas* entre *MSL* y *SOL*, obtenidas *debilitando comprehensión.*

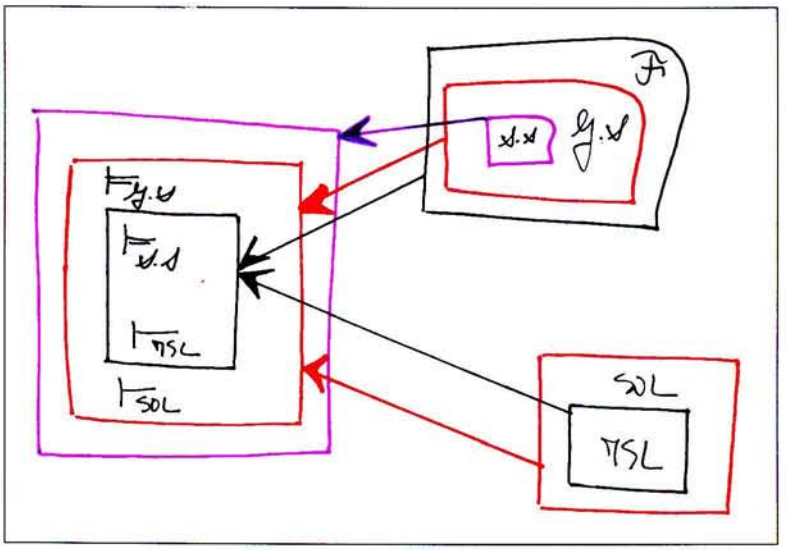


Figura 10.10: *Completud con semánticas nuevas*

Aquí radica el programa de investigación que comentaré posteriormente (capítulo 13), pero permitidme que os muestre donde se sitúan esas lógicas intermedias (ver figura: 10.11).

10.7. Modelos no estándar en primero y segundo orden

Comparación entre el axioma de inducción de segundo orden y el esquema de primer orden

Lo primero que quiero resaltar aparece en el propio título: la inducción en primer orden es un esquema; esto es, una colección infinita de axiomas de la forma

$$B\left(\frac{c}{x}\right) \wedge \forall x(B \rightarrow B\left(\frac{\sigma x}{x}\right)) \rightarrow \forall x\ B$$

mientras que en segundo orden es una única fórmula

$$\forall X(Xc \wedge \forall x(Xx \rightarrow X\sigma x) \rightarrow \forall x\ Xx)$$

Pese a ello, el poder expresivo del axioma de *SOL* es superior al de la colección infinita de *FOL*.

Ya a primera vista observamos que mientras que en primer orden sólo se aplica inducción a una clase numerable de conjuntos —puesto que sólo tenemos una colección numerable de fórmulas—, en *SOL* con semántica estándar

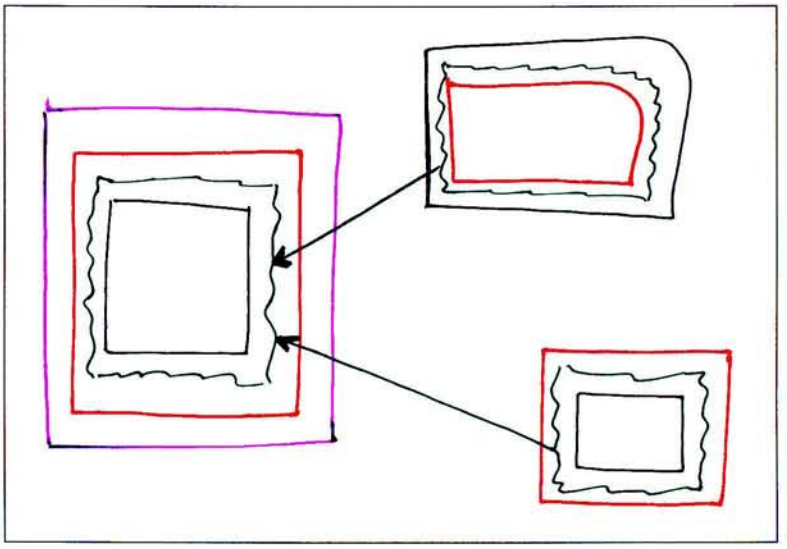


Figura 10.11: *Situación Lógicas intermedias*

la inducción se extiende a una colección supernumerable de ellos, la de los subconjuntos del universo $\mathbb{N}$. Pero la diferencia la marca no la cantidad, sino la calidad; no tendríamos modelos no isomorfos al de los naturales estándar en *FOL* si la cadena de los números estándar fuera definible en cualquier modelo de la aritmética de Peano, pero no es así. Me explicaré:

Ya vimos (el teorema 106) que construir un modelo no estándar de

$$\mathcal{N}_s = \langle \mathbb{N}, 0, s \rangle$$

o de

$$\mathcal{N} = \langle \mathbb{N}, 0, s, +, \cdot, \leq \rangle$$

resultaba sencillo en la lógica de primer orden aplicando el teorema de compacidad. Aquí no estándar significa no isomorfo al modelo pretendido. Para su demostración se extiende el lenguaje aritmético —que contenía al menos la constante individual para el cero c y el functor monario σ para la función del siguiente— con una nueva constante k y se toman en consideración las sentencias

$$k \neq c,\ k \neq \sigma c,\ k \neq \sigma\sigma c, \ldots.$$

que se añaden a las de la aritmética. Por compacidad se construye $\mathcal{M}$. Siendo $\mathcal{M}$ un modelo tanto de la aritmética de primer orden AP^1 como de las sentencias anteriores, el universo de la estructura contendrá números no estándar; es decir, números que no están en la serie de los siguientes del cero. Al conjunto de los números estándar le llamábamos

$$\mathbb{N}(\mathcal{M})$$

donde

$$\mathbb{N}(\mathcal{M}) = \{\mathcal{M}(c), \mathcal{M}(\sigma c), \mathcal{M}(\sigma\sigma c), ...\}$$

Es fácil ver que semejante conjunto no es definible en lógica de primer orden en la estructura $\mathcal{M}$. Como se comentó anteriormente, el esquema axiomático de inducción no tiene fuerza suficiente como para eliminar los números no estándar pudiéndose demostrar (es el teorema 77) que en cualquier modelo $\mathcal{A}$ de la aritmética de Peano AP^1 ser estándar equivale a la definibilidad en la estructura del conjunto de los números estándar $\mathbb{N}(\mathcal{A})$.

El argumento principal de la prueba es que si fuera definible mediante una fórmula B

$$\mathbb{N}(\mathcal{A}) = \{x \ / \ \mathcal{A}[x] \Vdash B(x)\}$$

también lo sería su complementario, mediante $\neg B(x)$

La fórmula

$$\exists x \neg B \rightarrow (\neg B\left(\frac{c}{x}\right) \vee \exists y(B\left(\frac{y}{x}\right) \wedge \neg B\left(\frac{\sigma y}{x}\right)))$$

equivale al axioma de inducción y por lo tanto es verdadera en $\mathcal{A}$. Pero su interpretación dice algo sorprendente

"Si existe un elemento en el complementario de los números estándar $x \notin \mathbb{N}(\mathcal{A})$ *entonces o bien* $\mathcal{A}(c) \notin \mathbb{N}(\mathcal{A})$ *—entre los números estándar no está el cero— o existe un* $y \in \mathbb{N}(\mathcal{A})$ *tal que su siguiente falta,* $\sigma^{\mathcal{A}}(y) \notin \mathbb{N}(\mathcal{A})$*."*

Así que de la existencia de un número no estándar se deriva una contradicción, asumiendo que $\mathbb{N}(\mathcal{A})$ fuera definible; por lo tanto no lo es.

Ahora se puede apreciar la diferencia entre la formulación de inducción en primero y segundo orden. En *FOL* no podemos aplicar inducción sobre el conjunto de los números estándar en un modelo que contuviera también números no estándar, porque sólo vale el axioma para conjuntos definibles. Sin embargo se puede demostrar la categoricidad de la aritmética de Peano de segundo orden[13].

Modelos no estándar de la aritmética

En el capítulo dedicado a teoría de modelos ya apuntamos que los modelos no estándar fueron descubiertos por Skolem en los años treinta, pero que entonces no se les prestó demasiada atención ya que eran vistos como contraejemplos patológicos. Skolem mismo los utilizó para intentar —sin éxito— desviar la atención del tratamiento formalista. A partir de 1949 reciben distinta consideración, siendo Henkin capaz de utilizarlos para demostrar la completud de la teoría de tipos usando una nueva semántica basada en modelos no estándar. Aunque el sentido de no estándar es aquí algo distinto, conviene que se vea que pese a todo están relacionados. Al final de su artículo *"Completeness in the theory of types"* [1950], Henkin construye un modelo no estándar por partida doble.

De hecho, como se ha visto en la sección anterior, con la semántica de modelos generales la lógica de segundo orden es completa en sentido fuerte y por tanto

[13] Le dedico todo el capítulo tercero de [37].

compacta. Por consiguiente, podemos utilizar un argumento similar al anteriormente esgrimido en lógica de primer orden y construir un modelo no estándar de la aritmética; entendiéndose como tal no isomorfo al clásico de los números naturales. Lo que veremos a continuación es que un modelo de AP^2 que contuviera números no estándar en el universo de individuos tendría que ser no estándar en el sentido que damos a ese término en *SOL*.

Veámoslo en detalle:

Sabemos cómo construir, utilizando compacidad, un modelo de la teoría de los números naturales que no sea isomorfo a

$$\langle \mathbb{N}, 0, s, +, \cdot \rangle$$

Por otra parte, en *SOL* una estructura no estándar, llamémosla $\mathcal{A}$, contiene un universo de individuos $\mathbf{A}$ y una familia de conjuntos $\langle \mathbf{A}_n \rangle_{n \geq 1}$ tales que cada $\mathbf{A}_n \subseteq \wp(\mathbf{A}^n)$, y $\mathbf{A}_m \neq \wp(\mathbf{A}^m)$ para al menos un $m \geq 1$. Esto es ser no estándar en términos de *SOL*.

Considerad ahora nuestro modelo no estándar de la teoría de los naturales $\mathcal{M}$ y construyamos una estructura de segundo orden sobre ella eligiendo una familia de subconjuntos $\langle \mathbf{M}_n \rangle_{n \geq 1}$ tales que cada $\mathbf{M}_n \subseteq \wp(\mathbf{M}^{\,n})$ para todo n. Llamemos $\mathcal{M}^*$ al resultado.

¿Es $\mathcal{M}^*$ un modelo de la aritmética de Peano de segundo orden?

Puesto que no he descrito una manera precisa de construir $\mathcal{M}^*$ la estructura podría no ser un modelo de AP^2; pero si lo fuera, necesariamente sería no estándar ya que

$$\mathbf{M}_1 \neq \wp(\mathbf{M})$$

La razón es ésta: Considerad el conjunto de los números estándar de $\mathcal{M}$; esto es, $\mathbb{N}(\mathcal{M})$. Si la estructura $\mathcal{M}^*$ fuera un modelo de AP^2, en particular lo sería del axioma de inducción. Imaginad que el conjunto de los números estándar $\mathbb{N}(\mathcal{M})$ estuviera en el universo de relaciones unarias de la estructura $\mathcal{M}^*$. Puesto que contiene el cero y el siguiente de cualquiera de sus elementos $\mathbf{M} = \mathbb{N}(\mathcal{M})$. Esto no es cierto, sabemos que en $\mathbf{M}$ había números no estándar.

¿Qué conclusión extraemos de todo esto?

Muy fácil

$$\mathbb{N}(\mathcal{M}) \notin \mathbf{M}_1 \text{ y por lo tanto } \mathbf{M}_1 \neq \wp(\mathbf{M})$$

—la estructura $\mathcal{M}^*$ no es estándar en el sentido de la lógica de segundo orden—

Espero no haberos confundido hablando ahora de modelos no estándar de la aritmética de Peano de segundo orden

¿No había dicho que en segundo orden esta aritmética es categórica?, ¿cómo podemos entonces tener un modelo no isomorfo a $\mathcal{N}$?

Bueno, como ya señaló Skolem en el 1929, la aritmética de Peano es categórica sólo cuando *conjunto* tiene el significado estándar; esto es, cuando lo tomamos de la metateoría: $\mathbf{A}_n = \wp(\mathbf{A}^n)$ para todo n. Pero Henkin, en 1949 arrojó una nueva luz sobre el tema y pudimos construir un modelo no estándar de AP^2 que usa semántica así mismo no estándar. Para ello debemos abandonar el punto de vista clásico y abrazar la semántica nueva que define para cada modelo sus propios universos de conjuntos y relaciones.

Cuando abrimos la puerta a las interpretaciones no estándar SOL pierde parte de su poder expresivo y la aritmética de Peano deja de ser categórica. Sigue siendo más potente que la de primer orden pues la consistencia de AP^1 se puede probar en AP^2, existiendo una fórmula $\varphi_{CON(PA^1)}$ que expresa la consistencia de la versión de primer orden de la aritmética de Peano.

$$AP^1 \not\models \varphi_{CON(PA^1)} \text{ en } FOL \text{ pero } AP^2 \models_{\mathcal{G}.\mathcal{S}} \varphi_{CON(PA^1)}$$

—en SOL con semántica general—

Como ya he dicho, la razón por la que la aritmética de Peano de primer orden no es estándar es que el conjunto de los números estándar no es definible en primer orden en una estructura que contuviera números no estándar, de manera que no podemos aplicar inducción sobre el conjunto que de verdad nos interesa.

La razón por la que es categórica la aritmética de Peano de segundo orden con semántica estándar es porque podemos aplicar inducción sobre todos los conjuntos posibles y una estructura que contuviera números no estándar nunca podría ser modelo del axioma de inducción de segundo orden.

Cuando en segundo orden permitimos que halla estructuras con universos relacionales incompletos, la cuantificación sólo se aplica a los conjuntos y relaciones que están presentes en los universos de la estructura y pudiera muy bien suceder que el de relaciones unarias no contuviera al de los números estándar como uno de sus miembros. En los sistemas generales de Henkin lo que hacemos es tomar todos los conjuntos y relaciones que son definibles con parámetros mediante fórmulas de segundo orden. Bueno, al igual que sucedía en primer orden, aquí tampoco es definible el conjunto de los números estándar en un sistema cuyo universo contuviera números no estándar.

Bibliografía

[1] Ajtai, M. [1979]. *"Isomorphism and higher-order equivalence"*, **Annals of Mathematical Logic**, vol. 16, pp. 181-203.

[2] Andrews, P. B. [1972]. *"General models, descriptions and choice in type theory"*, **The Journal of Symbolic Logic**, vol. 37, n. 2, pp. 385-394.

[3] Andrews, P. B. [1972]. *"General models and extensionality"*, **The Journal of Symbolic Logic**, vol. 37, n. 2, pp. 395-397.

[4] Andrews, P.B., Miller, D., Cohen, R. [1984]. *"Automating higher-order logic"*, *Automated Theorem Proving: After 25 Ears* (Ed. W. W. Bledsoe, D. W. Loveland), Providence, American Mathematical Society, pp. 169-192.

[5] Apt, K. R., Marek, W. [1974]. *"Second order arithmetic and related topics"*, **Annals of Mathematical Logic**, vol. 6, pp. 177-229.

[6] Baldwin, J. [1985]. *"Definable second-order quantifiers"*, en Barwise, Feferman (1985).

[7] Baldwin, J., Shelah, S. [1985]. *"Second-order quantifiers and the complexity of theories"*, **Notre Dame Journal of Formal Logic**, vol. 26, n.3, p. 119 y ss.

[8] Barwise, J. [1972]. *"The Hanf number of second-order logic"*. **Journal of Symbolic Logic**, n. 37, 588-594.

[9] Benthem, J y Doets, K. [2001]. *"Higher-order logic"*. (en [22])

[10] van Benthem, J. [1995]. *"The sources of Complexity: Content versus wrapping"*, en [38].

[11] Boolos, G. [1975]. *"On second-order logic"*, **Journal of Philosophy**, vol. 72, 509-527.

[12] Boolos, G. [1995]. *"Frege's theorem and the Peano postulates"*, **The Bulletin of Symbolic Logic**, vol. 1, n. 3, pp. 317-326.

[13] Boudreaux, J. C. [1979], *"Defining general structures"*, **Notre Dame Journal of Formal Logic**, vol. XX, n. 3, pp. 465-488.

[14] Cocchiarella, N.B. [1974]. *"A new formulation of predicative second order logic"*, **Logique et Analyse**, vol. 65, 61-86.

[15] Cocchiarella, N.B. [1985]. *"Two λ-extensions of the theory of homogeneous simple types as a second order logic"*, **Notre Dame Journal of Formal Logic**, vol. 26, n. 4, 307-407.

[16] Church, A. [1940]. *"A formulation of the simple theory of types"*. **The Journal of Symbolic Logic.** vol. 5, pp. 56-68.

[17] Church, A. [1941]. *The calculi of lambda-conversion.* Annals of Mathematical Studies. num. 6. Princeton University Press. Princeton. USA.

[18] Church, A. [1956]. *Introduction to Mathematical Logic***.** Princeton University Press. Princeton. USA.

[19] Ebbinghaus, H. D., Flum, J. y Thomas, W. [1984]. *Mathematical Logic***.** Springer-Verlag. Berlin. Alemania.

[20] Enderton, H. B. [1972]. *A Mathematical Introduction to Logic***.** Academic Press. New York. USA.

[21] Frege, G. [1879]. *Begrgriffsschrift, eine der arithmetischen nachgebildete Formlesprache des reinen Denkens.* Traducción inglesa en van Heijenoort, J. ed. [1967]. pp.5-82.

[22] Gabbay, D y Guenthner, F. eds. [2001]. *Handbook of Philosophical Logic***.** vol.I-II. Kluwer Academic Press. Amsterdam. Holanda.

[23] Gabbay, D. y Hogger, C. J. [1994-2002]. *Handbook of Logic in Artificial Intelligence and Logic Programming.* Vols 1-5. Oxford University Press. Oxford. U.K.

[24] van Heijenoort, J. [1967]. *From Frege to Gödel: a source book in mathematical logic*, 1879-1931. Harvard University Press. Cambridge, Mass. USA.

[25] Henkin, L. [1949]. *"The completeness of the first order functional calculus"*. **JSL.** vol. 14, pp. 159-166.

[26] Henkin, L. [1950]. *"Completeness in the theory of types"*. **JSL.** vol. 15. pp. 81-91.

[27] Henkin, L. [1953]. *"Banishing the rule of substitution for functional variables"*. **JSL.** vol. 18, num. 3. pp. 201-208.

[28] Hilbert, D. y Ackermann, W. [1928]. *Grundzege der theoretischen Logik.* Traducción inglesa en Hilbert y Ackermann [1938]. *Principles of Mathematical Logic.* Chelsea. New York. USA.

[29] Kneale, W y Kneale, M. [1962]. *The Development of Logic***.** The Clarendon Press. Oxford. U.K.

[30] Kurucz, A, Manzano, M y Sain, I [1992]. *"How to Increase applicability of a mathematical concept of higher order logic to real higher order phenomena"*. **Logic @ Work**. Oxford University Press. Oxford. U.K.

[31] Kurucz, A, Manzano, M y Sain, I [1992]. *"The little mermaid"*

[32] Leivan, D. [1996]. *Higher Order Logic.* (en [23], pp 229-300)

[33] Manzano, M. [1980]. *Teoría de Tipos***.** Ediciones de la Universidad de Barcelona. Barcelona.

[34] Manzano, M. [1982]. *"Los sistemas generales"*. en

[35] Manzano, M. [1999]. *Model Theory.* Oxford University Press. Oxford. U.K. (traducción de: *Teoría de modelos.* AUT/126. Madrid: Alianza Editorial.)

[36] Manzano, M. [1993]. *"Introduction to Many-Sorted Logic"*. (en [39])

[37] Manzano, M. [1996]. *Extensions of First Order Logic***.** Cambridge Tracts in Theoretical Computer Science. Cambridge University Press. Cambridge. U.K.

[38] Marx, M., Pol∉s, L. (Eds.) [1995]. *Logic @ Work*, Oxford, Oxford University Press.

[39] Meinke, K y Tucker, J. eds. [1993]. *Many-Sorted Logic and its Applications.* John Wiley & Sons. Chichester. U.K.

[40] Mosterín, J. [1979]. *Un cálculo deductivo para la lógica de segundo orden.* Cuadernos Teorema 51. Valencia.

[41] Quintanilla, M. A. edts. [1983]. *Estudios de Lógica y Filosofía de la Ciencia***.** Serie: Manuales Universitarios. Ediciones Universidad de Salamanca. Salamanca.

[42] Russell, B. [1908]. *"Mathematical Logic as based in the theory of types"*. (en [24])

[43] Tarski, A. [1923] *"Sur le terme primitif de la logistique"*. **Fundamenta Mathematicae.** Vol. IV. pp. 196.

[44] Tarski, A. [1956]. *"The concept of truth in formalized languages"*. (en [45])

[45] Tarski, A. [1956]. *Logic, semantics, metamathematics***.** Papers from 1923 to 1938. Clarendon Press. Oxford.

Capítulo 11

Teoría de Tipos

11.1. Paradojas

Hacia el cambio del siglo XIX al XX, un número creciente de paradojas fueron apareciendo en la matemática, particularmente en la recién estrenada teoría de conjuntos. En un principio parecían afectar sólo a los conjuntos muy grandes por lo que se pensó que tal vez fuera la noción de infinito la que causaba problemas, pero cuando Russell descubrió su famosa paradoja, que minaba su noción fundamental de definición de conjunto, todos se convencieron de la necesidad de modificar substancialmente la teoría. Es de sobra conocida la vía de solución axiomática, representada por Zermelo y sus modificadores por un lado, y por Von Neuman y los suyos[1], por otro. Otra segunda vía, que no comentaré es la tomada por Brower y los intuicionistas[2]. Finalmente, Russell hizo borrón y cuenta nueva al diseñar su *teoría de tipos*. Mientras que en las teorías axiomáticas el procedimiento es el de *"parcheado"*, taponando con enunciados apropiados las vías de agua descubiertas, en la de tipos no pueden reproducirse porque el propio lenguaje impide su formulación.

Las paradojas conocidas son muchas[3], entre ellas voy a distinguir cuatro, muy sencillas. En 1926 Ramsey [62] las clasificó en dos grandes grupos:

1. Lógicas o matemáticas
2. Semánticas o epistemológicas

Esta división ha resultado ser extraordinariamente fructífera, permitiéndonos utilizar técnicas diferentes para evitar las de uno u otro grupo: para expresar las del primero sólo se precisan nociones matemáticas tales como el concepto de *clase*, o de *cardinalidad*, o el de *número*; para las del segundo hace falta mezclar insidiosamente lenguaje y metalenguaje usando conceptos como el de *verdad* o el de *definibilidad* en el lenguaje objeto.

[1] El capítulo 5 está enteramente dedicado a la axiomática de Zermelo-Fraenkel.
[2] Una buena exposición de ella la hace Dirk van Dalen en [21].
[3] Consultad la bibliografía de este capítulo y en especial el libro de Sainsbury [65].

La clasificación de Ramsey ayuda no sólo a entender las paradojas sino también a resolverlas mediante distintas técnicas. Las lógicas o matemáticas se evitan en la teoría simple de tipos estructuralmente—también se resuelven en las teorías axiomáticas de conjuntos, de manera algo artificiosa formulando una serie de axiomas específicos que las bloquean—, pero para eludir las semánticas precisamos de la teoría ramificada de tipos, o de la distinción de Tarski entre lenguaje y metalenguaje[4]. Es por ello por lo que al utilizar un lenguaje lógico respetamos la estipulación de no mezclar lenguaje y metalenguaje; así, si queremos expresar el concepto de verdad de una fórmula del lenguaje objeto utilizamos otro lenguaje que servirá de metalenguaje del primero.

Paradojas Lógicas

Entre ellas la de Russell y la de Cantor son las más conocidas. La de Russell cuestiona el mismísimo principio de definición de clases, mientras que la de Cantor se basa en un concepto más elaborado, pero tan básico como aquel, que es el de cardinalidad.

Paradoja de Russell[5]

En la teoría intuitiva, cantoriana, de conjuntos todas las propiedades permiten definir conjuntos, pero si tomamos como propiedad la de no pertenecerse a sí mismo, llegamos a una contradicción al analizar la clase

$$U = \{X/X \notin X\}$$

y preguntarnos

$$\textit{¿Es } U \textit{ un elemento de } U\textit{?}$$

Esta paradoja no puede producirse en la teoría de tipos simple porque la sucesión de signos $X \notin X$ no es una fórmula y el axioma de definición de clases no se aplica para ella. Nuestras fórmulas atómicas son de la forma

$$X^{\alpha} \in Y^{\beta} \text{ —o si se prefiere } Y^{\beta}X^{\alpha}\text{—, siendo } \beta = \langle 0, \alpha \rangle$$

donde la distinción de tipos es fundamental.

Lo explico mejor: Se admite actualmente que tanto la teoría de conjuntos como la de tipos representan dos alternativas distintas en lo que a la fundamentación de la matemática se refiere y las diferencias entre ellas se consideran simplemente notacionales, sin mayor relevancia. Nosotros vamos a pasar de la teoría de tipos a la de conjuntos en dos pasos, señalando el talón de Aquiles de la transformación y el punto en el que hay que poner el parche en la teoría de conjuntos.

[4] Recordad lo expuesto en la sección 1.4.1.
[5] La expuse con mayor detalle en la sección 5.1.2.

1. Admitimos que las propiedades definen conjuntos y que podemos identificarlas con las fórmulas que las definen. Además de esto, las relaciones pueden reducirse a conjuntos, puesto que los pares ordenados se asimilan a ciertos conjuntos y las n-tuplas son pares encajados unos en otros. Naturalmente, cuando expresamos que X^α está en el conjunto Y^β —donde $\beta = \langle 0, \alpha\rangle$— podemos escribir

$$Y^\beta X^\alpha \quad \text{o} \quad X^\alpha \in Y^\beta$$

Esto no es otra cosa que un cambio superfluo, notacional, ya que las dos fórmulas tienen el mismo alcance y significado. En este punto el axioma de definición de clases y relaciones toma la forma siguiente:

$$\exists Y^\beta \forall X^\alpha (X^\alpha \in Y^\beta \leftrightarrow \varphi)$$

—siendo $\beta = \langle 0, \alpha\rangle$— y en donde la distinción de tipos no es en modo alguno superficial.

2. Si la distinción de tipos no existiera y admitiéramos que nuestro universo sólo contiene conjuntos, el axioma de definición de conjuntos se expresaría mediante la fórmula

$$\exists Y \forall X (X \in Y \leftrightarrow \varphi)$$

Pero, sin distinción de tipos, podemos de nuevo tomar como fórmula φ a la fórmula $X \notin X$, obteniendo así

$$\exists Y \forall X (X \in Y \leftrightarrow X \notin X)$$

Por el principio de extensionalidad el conjunto definido mediante esta fórmula es único, llamémosle U. Entonces,

$$\forall X (X \in U \leftrightarrow X \notin X)$$

La contradicción se reproduce cuando intentamos determinar si este U es un miembro de sí mismo; entonces obtenemos

$$U \in U \leftrightarrow U \notin U$$

Para evitar este problema, en las teorías axiomáticas de conjuntos se adopta una de estas medidas: o bien se distingue entre clases y conjuntos o se tiene un axioma restringido de definición de nuevos conjuntos a partir de conjuntos previamente definidos[6].

Comentario 361 *Sin embargo, si somos fieles a la imagen intuitiva de la jerarquía de conjuntos, en donde los conjuntos se construyen por niveles, el lenguaje de teoría de tipos es más adecuado pues en él se respeta esa intuición; por el contrario, el de primer orden utilizado en la teoría de conjuntos axiomática es completamente plano. ¿Para representar un paisaje montañoso no es mejor un sistema de representación que incluya curvas de nivel?*

[6] Es nuestro axioma de separación, introducido en la sección 5.4.1.

Paradoja de Cantor

Consideremos la clase U de todas las clases, la clase universal. Se puede demostrar tanto que la cardinalidad de su conjunto potencia $\wp(U)$ es mayor que la cardinalidad de la clase universal como que es menor o igual que ella.

Sea U el conjunto de todos los conjuntos. Evidentemente su cardinal $|U|$ sobrepasa a cualquier otro cardinal; *i.e.* $|A| \preceq |U|$ para todo A. Así,

$$\exists X \forall Y (|Y| \preceq |X|) \tag{11.1}$$

Pero veamos qué sucede cuando consideramos la clase $\wp(U)$. Es fácil ver que la cardinalidad de una clase cualquiera es estrictamente menor que la de su potencia. La prueba, del propio Cantor, se basa en dos hechos:

1. Para cada A, hay una función inyectiva f de A en $\wp(A)$; concretamente, la función
$$f : A \longrightarrow \wp(A)$$
definida mediante $f(X) = \{X\}$

2. Pero ninguna función h de A en $\wp(A)$ puede ser exhaustiva. La razón es como sigue: Para la clase
$$X = \{x \in A / x \notin h(x)\}$$
que es ciertamente un elemento de $\wp(A)$, no hay ningún elemento z de A tal que $h(z) = X$. Para demostrarlo, suponed lo contrario
$$h(z) = X \text{ para un cierto } z \in A \tag{11.2}$$
Claramente,
$$x \in X \leftrightarrow x \notin h(x), \text{ para cualquier } x \in A$$
En particular, para este z obtenemos
$$z \in X \leftrightarrow z \notin h(z) \tag{11.3}$$
Usando las líneas (11.2) y (11.3) obtenemos:
$$z \in h(z) \leftrightarrow z \notin h(z)$$
Esto, obviamente, es una contradicción.

Hemos, pues, demostrado

$$\forall X \exists Y (|Y| \succ |X|)$$

que equivale a

$$\neg \exists X \forall Y (|Y| \preceq |X|)) \tag{11.4}$$

Los enunciados (11.1) y (11.4) se contradicen.

La paradoja de Cantor no puede reproducirse en teoría de tipos porque un conjunto y el de sus partes están en tipos diferentes. En teoría axiomática de conjuntos la clase universal no es un conjunto y sólo la clase de las partes de un conjunto es también un conjunto[7].

Paradojas semánticas

La más antigua que se conoce de esta clase es la de Epiménides, otra es la de Richard.

Paradoja del mentiroso

Ya comentamos[8] que Epiménides el cretense sostenía que todos los cretenses eran unos mentirosos y que todos los enunciados proferidos por cretenses son puras mentiras.

El problema surge al definir la verdad en el lenguaje objeto mediante la fórmula

$$\forall x(Verdad\,(x) \leftrightarrow x) \tag{11.5}$$

y analizar el enunciado autorreflexivo *"Estoy mintiendo"* bajo su propia luz.

En la teoría de tipos finitos o en cualquier otra teoría formalizada en la que se distinga lenguaje de metalenguaje la fórmula (11.5) con el sentido arriba mencionado no puede ser una fórmula del lenguaje objeto y no puede pertenecer a la teoría.

Paradoja de Richard

Está directamente inspirada en el procedimiento diagonal que Cantor utiliza en su celebrada demostración de que los números reales constituyen un conjunto no numerable; es decir, cuya cardinalidad excede a la de los números naturales.

Recordemos esta prueba de Cantor: Sin pérdida de generalidad nos concentramos en los números reales del intervalo $[01]$ y consideramos una enumeración cualquiera de estos decimales;

$$a_m = \sum_{n\geq 1}(a_{mn} \times 10^{-n})$$

donde a_{mn} es la n-ésima cifra del m-ésimo decimal de esta lista

$$\begin{aligned} a_1 &:= 0.a_{11}a_{12}\cdots a_{1n}\cdots \\ &\vdots \\ a_m &:= 0.a_{m1}a_{m2}\cdots a_{mn}\cdots \\ &\vdots \end{aligned}$$

[7] El axioma de las partes o potencia de un conjunto, que intrudujimos en la sección 5.4.2 dice que si algo es un conjunto, el de sus partes también, pero U no lo es.

La interpretación estándar de la teoría se hace sobre un universo constituído exclusivamente por conjuntos de manera que *existir* en este contexto es *ser un conjunto.*

[8] En la sección 1.4.1, dedicada a la distinción entre lenguaje y metalenguaje, se explica con algo de detalle.

Es fácil ver que ninguna enumeración de esta índole contiene a todos los decimales del intervalo [01], para lo cual especificaremos uno,

$$d = \sum_{n \geq 1} (d_n \times 10^{-n})$$

que no está en la enumeración anterior.

En efecto, sea

$$0.d_1\ d_2\ d_3 ... d_n ...$$

el decimal cuya n-ésima cifra d_n es $a_{nn} + 1$ cuando $a_{nn} \leq 9$ y es 0 cuando a_{nn} es 9. Es obvio que d es un número decimal entre 0 y 1. Para ver que no está en la enumeración demostraremos que es distinto de a_n, para cada n. Esto último es evidente pues al menos su cifra n-ésima d_n difiere de la cifra n-ésima de a_n que es a_{nn}.

La paradoja de Richard es como sigue: consideremos las fracciones decimales que se pueden definir mediante un número finito de palabras. Ordenemos este conjunto utilizando en primer lugar el número de palabras empleadas en la definición y aplicando a continuación un orden lexicográfico. Mediante el método diagonal definamos un decimal d que no aparece en la ordenación. Claramente d difiere de todos los de la serie, y puesto que puede ser descrito con un número finito de palabras, debiera haber sido incluido en ella.

Veamos cual es el vínculo que se establece entre las paradojas y su solución citando al propio Russell [63]:

> In all the above contradictions (which are merely selections from an indefinite number) there is a common characteristic, which we may describe as self-reference or reflexiveness. The remark of Epimenides must include itself in its own scope. If all classes, provided they are not members of themselves are members of U, this must also apply to $U; /.../$ Thus all our contradictions have in common the assumption of a totality such that, if it were legitimate, it would at once be enlarged by new members defined in terms of itself.

Para él había una diferencia fundamental entre los conjuntos por un lado y las propiedades por otro. Los conjuntos o clases son extensionales y su existencia la deben a sus miembros. En esta concepción los conjuntos existen con independencia de si pueden o no ser definidos y no hay entonces razón para evitar las definiciones circulares o no-predicativas. Por otra parte, los conceptos o propiedades existen sólo en la medida en que podamos definirlos, captarlos. Cuando las entidades que constituyen la teoría de conjuntos se conciben a la manera conceptualista, que sólo existen en la medida en que pueden ser definidos, la restricción a las definiciones predicativas es inevitable.

La teoría de tipos de Russell [1908] sobre la que se construyeron los *Principia Mathematicae* cumple algunos de estos requisitos. Analiza las paradojas en dos pasos. En el primero reformula la paradoja de teoría de conjuntos convirtiéndola en otra en la que las consideraciones acerca de la predicatividad son relevantes;

la cuantificación sobre conjuntos se desdibuja diciendo que son *modos de hablar* y lo único que queda es una teoría de propiedades. Los conceptos fundamentales de la teoría de propiedades son los de proposición y de función proposicional, que habían sido siempre conceptos lógicos, y en esta fase el análisis que realiza es más fino que el de Frege o el suyo de 1903.

Se considera hoy que la jerarquía de Frege, en realidad estipula una teoría de tipos para conceptos.

Comentario 362 *Ironías de la historia, en la teoría simple de tipos,cercana a la teoría de conjuntos de Zermelo, se evitan las paradojas lógicas y matemáticas. El único problema es que Frege, como Russell, tendría que haber explicado la teoría de conjuntos como un* modo de hablar *acerca de conceptos, y no estipular la existencia de los conjuntos como una categoría separada de entidades sin tipo. Habría tenido entonces sólo los problemas que Ramsey denomina epistemológicos.*

Por su parte Russell, en su teoría ramificada de tipos, evita todas las paradojas, y no solamente las matemáticas. Sin embargo, el lenguaje se torna innecesariamente complejo y para poder obtener ciertos resultados matemáticos se ve forzado a admitir el denominado axioma de reducibilidad, de dudosa aceptación.

Aunque se cree que parte de la motivación para desarrollar la teoría de tipos se debe al descubrimiento de las paradojas de la teoría intuitiva de conjuntos, está lejos de ser una construcción *ad hoc* carente de naturalidad y sentido común matemático. Citemos a Gandy [29]:

> The simple theory of types provides a straightforward, reasonably secure, foundation for the greater part of classical mathematics. That is why a number of authors (Carnap, Gödel, Tarski, Church, Turing) gave a precise formulation of it, and used it as a basis for metamathematical investigations. The theory is straightforward because it embodies two principles which (at least before the advent of modern abstract concepts) were part of the mathematician's normal code of practice. Namely that a variable always has a precisely delimited range, and that a distinction must always be made between a function and its arguments. In this sense one might claim that all good mathematicians had anticipated simple type theory. [Indeed Turing made this claim for primitive man]. The claim goes too far, but it does draw attention to the fact that the justification for the theory is not to be found by considering it as a formal device to avoid Russell's paradox.

11.2. Teoría simple de tipos de Church

En lo que sigue presentaré muy brevemente la teoría simple de tipos del artículo de Church de 1940, complementada en su parte semántica con los de

Henkin [36] y [38]. Está basada en la de Russell, pero distingue claramente entre sintaxis y semántica y entre lenguaje y metalenguaje, por lo que aúna las ventajas del tratamiento de Russell para evitar las paradojas lógicas y las del de Tarski, para evitar las semánticas. Si se admite para las fórmulas una semántica de modelos generales, conseguimos un teorema de completud; al precio, naturalmente, de la merma de capacidad expresiva.

Jerarquía de tipos

La teoría de tipos allí expuesta, que llamaré $\mathfrak{T}$ podría describirse así: Los tipos están estructurados en una jerarquía que cuenta con:

- $\mathcal{D}_1$ es un conjunto no vacío; el de individuos de la jerarquía.
- $\mathcal{D}_0$ es el dominio de valores de verdad (como estamos en lógica binaria, dichos valores se reducen a V y F)
- El resto de los dominios se construyen a partir de los tipos básicos: si $\mathcal{D}_\alpha$ y $\mathcal{D}_\beta$ han sido construidos ya, definimos $\mathcal{D}_{(\alpha\beta)}$ como el dominio formado por todas las funciones de $\mathcal{D}_\beta$ en $\mathcal{D}_\alpha$. Los conjuntos se representan mediante funciones características y así $\mathcal{D}_{(0\beta)}$ es el conjunto potencia de $\mathcal{D}_\beta$, formado por todos los subconjuntos de $\mathcal{D}_\beta$.

Para hablar acerca de esta jerarquía se introduce un lenguaje formal. El del artículo de Church [13] es, como dijimos, un lenguaje de tipos, que posee distintas clases de variables y se añade además el abstractor.

Alfabeto

Contamos con variables de diferentes tipos, con un superíndice que lo indica.

En este artículo de Church se introducen también las constantes $N_{(00)}$ y $A_{((00)0)}$ (para la negación y la conjunción) y $\Pi_{(0(0\alpha))}$ (nos permitirá expresar cuantificación). Finalmente $\iota_{(\alpha(0\alpha))}$ es un selector.

Expresiones

Hay dos formas fundamentales de formar expresiones:

- *Las simples*: Fórmulas o constantes solas son fórmulas cuyo tipo es el del índice.
- *Las compuestas* son de dos clases: (1) Con λ formamos funciones. Si X^β es una variable y M^α es una expresión, $(\lambda X^\beta M^\alpha)$ es una expresión de tipo $(\alpha\beta)$. (2) Las otras corresponden a la asignación de valor bajo una función: si $F^{(\alpha\beta)}$ y B^β son expresiones, $F^{(\alpha\beta)}B^\beta$ es una expresión de tipo α.

Interpretación

La interpretación de estas expresiones es como sigue:

- Las variables toman valores en los universos o dominios correspondientes. Las constantes tendrán valores decididos de antemano; por ejemplo, $N_{(00)}$ será negación y $A_{((00)0)}$ disyunción.
- La expresión $(\lambda X^{\beta} M^{\alpha})$ se interpretará como una función de $\mathcal{D}_{\beta}$ en $\mathcal{D}_{\alpha}$. (Por ejemplo, $(\lambda X^{\beta} X^{\beta})$ será la función identidad. Sin embargo, $(\lambda X^{\beta} \phi)$ se interpretará como la clase de los elementos de $\mathcal{D}_{\beta}$ que verifican la condición expresada por la fórmula ϕ.)
- Finalmente, la expresión $F^{(\alpha\beta)} B^{\beta}$ se interpreta como el valor de la función que interprete a $F^{(\alpha\beta)}$ para el valor que interprete a B^{β}.
- Por su parte, la función $\Pi_{(0(0\alpha))}$ asigna a cada subconjunto de $\mathcal{D}_{\alpha}$ —que será un elemento de $\mathcal{D}_{(0\alpha)}$— el valor F excepto cuando dicho subconjunto sea el propio $\mathcal{D}_{\alpha}$, la totalidad. Así, podemos pensar que este predicado de predicados afirma de ellos su universalidad. Se entenderá, pues, que $\Pi_{(0(0\alpha))}\,(\lambda X^{\alpha} \phi)$ exprese cuantificación, $\forall X^{\alpha} \phi$. Literalmente afirma: la clase de los individuos que verifican ϕ es universal, es todo el universo $\mathcal{D}_{\alpha}$. Hay un lenguaje de la teoría de tipos que a mí me gusta aún más, es el que tiene λ e igualdad como únicos signos primitivos. En él los conectores y los cuantificadores se definen usando sólo estos signos. Por ejemplo, $\forall X^{\alpha} \phi$ se define mediante esta fórmula[9]:

$$(\forall X^{\alpha} \phi) =_{Df} (\lambda X^{\alpha} X^{\alpha} = X^{\alpha})$$

- La interpretación de $\iota_{(\alpha(0\alpha))}$ es la de un selector que a cada elemento de $\mathcal{D}_{(0\alpha)}$ —es decir, a cada subconjunto de $\mathcal{D}_{\alpha}$— le asigna un elemento de $\mathcal{D}_{\alpha}$. Cuando la interpretación de la fórmula

  $$(\lambda X^{\beta} \phi)$$

  es una clase unitaria el selector asigna a dicha clase ese único elemento y así

  $$(\iota_{(\alpha(0\alpha))} (\lambda X^{\alpha} \phi))$$

  se abrevia como

  $$(\iota X^{\alpha} \phi)$$

  considerándose como un descriptor.

La diferencia de interpretación entre las constantes $N_{(00)}$, $A_{((00)0)}$ y $\Pi_{(0(0\alpha))}$ por un lado, y $\iota_{(\alpha(0\alpha))}$ por otro, es que en el primer caso ésta está fijada de antemano y de forma unívoca. En el segundo se dice que ha de ser cualquier función de elección del tipo adecuado, sin determinarse cual.

[9] Lo presentaré con detalle en la sección 11.5. Está basado en Henkin [37] y [38], y también en Andrews [6].

Comentario 363 *Parece ser que esta asimetría, esta indeterminación, está en la base de la investigación que llevó a Henkin a descubrir sus pruebas de completud. De manera que a ella debemos una pieza clave de toda lógica, la reina de las pruebas de los sistemas formales, pues su procedimiento, a diferencia del usado por Gödel para demostrar completud para la lógica de primer orden, es tan versátil que se adapta fácilmente a otras lógicas.*

Cálculo El cálculo deductivo[10] de dicho artículo de Church incluye seis reglas de inferencia y once axiomas. Las tres primeras reglas son reglas de sustitución, incluyendo las reglas de λ-conversión; entre otras, la siguiente

$$\frac{(\lambda X^{\beta} M^{\alpha}) N^{\beta}}{M^{\alpha}\left(\frac{N\beta}{X\beta}\right)}$$

(una aplicación de esta regla nos permitiría pasar de $(\lambda X^{\alpha} R X^{\alpha}) A$ a RA)
y su conversa.

Hay también una regla de *instanciación*, *modus ponens* y *generalización*.

Además de tener los axiomas proposicionales y los de cuantificadores, se incluye el siguiente *axioma para las descripciones*

$$(f^{(0\alpha)} X^{\alpha}) \to (\forall Y^{\alpha} (f^{(0\alpha)} Y^{\alpha} \to X^{\alpha} = Y^{\alpha}) \to f^{(0\alpha)} (\iota_{(\alpha(0\alpha))} f^{(0\alpha)}))$$

Se añade también *extensionalidad*, *elección* e *infinitud*, pues Church quería que sirviera como alternativa a la teoría de conjuntos, como sistema fundacional. El axioma de definición de clases y relaciones no es preciso añadirlo ya que se sigue de las reglas de conversión lambda.

$$(f^{(0\alpha)} X^{\alpha}) \to f^{(0\alpha)} (\iota_{(\alpha(0\alpha))} f^{(0\alpha)})) \tag{11.6}$$

es la elegante formulación del *axioma de elección* que en este lenguaje propone Church.

$$\forall X^{\beta} (f^{(\alpha\beta)} X^{\beta} = g^{(\alpha\beta)} X^{\beta}) \to f^{(\alpha\beta)} = g^{(\alpha\beta)}$$

es *extensionalidad*.

Henkin en [39] sostiene lo siguiente:

> Several features of the theory $\mathfrak{T}$ sketched above were interesting to me, but I was especially attracted by the neatness and shortness of the formula expressing the axiom of choice[11]. It seemed to me that

[10] En [52] y en [54] propongo otros cálculos para la teoría de tipos.
[11] La fórmula 11.6.

> the symbol $\iota_{\alpha(0\alpha)}$ was put into the formal languaje of $\mathfrak{T}$ originally to serve the function of the definite article *the,* as expressed in Axiom 9^{α} (el de las descripciones), and that its availability to provide such a succint formulation of the axiom of choice was a fortuitous circumstance that must have come to Church as an inspired afterthought.

Números naturales en $\mathfrak{T}$

En el artículo de Church se introducen como abreviaturas las constantes para cada número natural y se define a éstos, siguiendo a Russell y a Frege. Se prueba para ellos, como teoremas, los postulados de Peano.

En particular, el cero, el uno y el dos se definen así:

$0^{\alpha'} = \lambda f^{(\alpha\alpha)} \lambda X^{\alpha} X^{\alpha}$

$1^{\alpha'} = \lambda f^{(\alpha\alpha)} \lambda X^{\alpha} (f^{(\alpha\alpha)} X^{\alpha})$

$2^{\alpha'} = \lambda f^{(\alpha\alpha)} \lambda X^{\alpha} (f^{(\alpha\alpha)} (f^{(\alpha\alpha)} X^{\alpha}))$

La función del siguiente y el concepto de número natural también son definidos mediante expresiones con lambda. Incluye Church la definición de la función del predecesor de un número natural, uno de los logros de Kleene, que sirvió a ambos para calibrar la enorme potencia de la $\lambda-$definibilidad.

Incompletud

Cuando uno se da cuenta de que el cálculo no sólo es completo, sino incompletable —esto es, no se torna completo añadiéndole nuevos axiomas o reglas— la pregunta es si podemos separar de alguna forma a las fórmulas que son no sólo válidas, sino también deducibles en el cálculo de las que son sólo válidas. Cito a Henkin [38]:

> This broad, somehow, vague question, has several answers —among them (a), (b), (c) below—
>
> (a) It is known that there can be no decision procedure to determine automatically, in a finite number of steps, whether or not a given logically valid formula is provable.
>
> (b) We can describe certain sets of formulas of type 0 for which we can demonstrate that any formula of the set will be provable if, and only if, it is logically valid. Thus, we have completeness relative to each of these sets...*[Se refiere a la teoría de tipos proposicional y a las fórmulas de teoría de tipos que son de primer orden]*
>
> (c) Returning to the totality of logically valid formulas of L, we can pick out from among them those which are provable by introducing certain non-classical notions of validity. These are defined by enlarging the notion of model, a possibility engendered by observing that the language L may be used to refer to other kinds of systems than those we have described so far. There are two different directions in

which we can proceed, described below...*[Aquí describe dos métodos: el de los modelos generales y el de los booleanos].*

11.3. *HOL* en programación y en computación

La lógica de orden superior *HOL* incluye a la de segundo orden, a la de tercer orden y en general a la de cualquier orden finito, así como a la teoría simple de tipos expuesta en el apartado precedente.

Me voy a detener sólo en dos aspectos:

1. Los tipos de datos de orden superior
2. La naturaleza computacional de la deducción natural

11.3.1. Tipos de datos de orden superior

Las funciones de orden superior —esto es, las funciones de funciones— constituyen la base de los lenguajes de programación de orden superior. Marcó el punto de inflexión, todavía en los años 60 del siglo XX, el lenguaje *Algol* y los que de él se derivaron, tales como el *ADA* y el *Algol 68*; en todos ellos se usa la estructura de tipos de dicha lógica de forma similar a como aparece en el cálculo lambda con tipos —usan sólo tipos finitos en *Algol 60* y *ADA*, infinitos en *Algol 68*.

Una vez que se ha integrado explícitamente en el lenguaje la estructura de tipos, es natural aplicar también las operacioens de λ—abstracción sobre ellos. Nos servimos de los tipos tanto para estructurar los datos como para fijar su comportamiento funcional. Otro lenguaje de programación inspirado en el cálculo lambda es el *LISP*[12].

El denominado *polimorfismo* es de nuevo una forma de abstracción sobre tipos propia del lenguaje de programación *ML*. En este caso se utiliza un cálculo lambda con abstracción de segundo orden.

¿Qué es lo que caracteriza semejante abstracción?

En la teoría de tipos simple no tenemos algo tan básico como la función global de identidad, para ello haría falta un abstractor de tipos[13].

11.3.2. La naturaleza computacional de la deducción natural

Podemos remontarnos a 1924, a los inicios de la lógica combinatoria, cuando Schönfinkel observó que las inferencias básicas de la lógica son de naturaleza funcional; es decir, que son funciones de fórmulas en fórmulas. Esta analogía fue posteriormente elaborada por Curry en el 1958 y redescubierta por Howard en 1980.

[12] Extremo que, contra toda evidencia, niega su creador.

[13] En la sección 11.4 nos ocupamos extensamente de este problema.

Vamos a indicar brevemente de qué se trata; el cálculo deductivo que consideraremos será el de deducción natural, sus reglas pueden ser interpretadas como funciones que toman fórmulas como argumentos y devuelven una fórmula como valor. Estas funciones entre fórmulas lo serán del cálculo lambda: las fórmulas del cálculo de deducción natural son los tipos básicos del cálculo lambda —esta analogía es conocida bajo este epígrafe: *"proposiciones como tipos"*—, las reglas del cálculo de deducción natural son o bien $\lambda-$términos u otras expresiones del $\lambda-$cálculo.

Vamos a presentar brevemente el cálculo de deducción natural para examinarlo a la luz de la semántica de Heyting; es decir, considerando a las fórmulas como secuencias de instrucciones capaces de determinar su *sentido* y no como el resultado de efectuar dichas operaciones —esto es, su *referencia* o *denotación*—. La dicotomía sentido/referencia opera a nivel profundo en lógica.

Mientras que la *referencia* o *denotación*, la *verdad*, la *semántica* y las *operaciones algebraicas* están en la base de la concepción de Tarski de la verdad, en la de Heyting se asocia el *sentido* con la idea de *sintaxis* y de *prueba*. La corriente mayoritaria, de tradición algebraica, abandona toda reflexión sobre el sentido y acata el punto de vista extensional —opuesto a *intensional*— que es estático, denotativo. Sin embrago, el punto de vista intensional es dinámico. En vez de preguntar *¿En qué condiciones es verdadera la fórmula* A*?* nos preguntamos, *¿qué es una prueba de* A*?*

Por ejemplo, una prueba de $A \wedge B$ es un par $\langle p, q\rangle$ que consta de una prueba p de A y de una prueba q de B.

Cálculo de deducción natural

La notación

$$\begin{array}{c}\vdots\\ A\end{array}$$

la usamos para indicar una deducción de A; esto es, que acaba en A. La deducción se escribirá como un árbol finito cuyas hojas están etiquetadas con sentencias. Las sentencias pueden aparecer en dos estados: *vivas* y *muertas*. En general estarán vivas en una deducción y sólo están muertas cuando dejan de funcionar activamente en la prueba.

El caso típico es el de la introducción del condicional

$$\dfrac{\begin{array}{c}[A]\\ \vdots\\ B\end{array}}{A \to B}\ I \to$$

(Descargar una sentencia es *matarla*)

Las reglas de este cálculo las introdujimos en la sección 4.4, por lo que no las repito ahora, consultadlas allí.

Todas las reglas preservan todas las hipótesis, salvo $I \rightarrow$ ya que desaparece (se *descarga* o se *mata*) $[A]$.

Interpretación de las reglas de deducción natural como $\lambda-$términos

Una fórmula A se entenderá como el conjunto de sus posibles deducciones; en vez de decir *"δ prueba A"* diremos $\delta \in A$.

Las reglas del cálculo de deducción natural se verán ahora como un modo de construir funciones; una deducción de A a partir de las hipótesis $B_1, ..., B_n$ puede expresarse como una función $t\,[z_1....z_n]$ que asocia a los elementos $b_i \in B_i$ un resultado $t\,[b_1....b_n] \in A$.

1. *Hipótesis:* Una deducción que consta de una hipótesis sola A se representa mediante una expresión x, donde x es una variable para A

2. *Introducción de la conjunción*: Si se ha obtenido una deducción mediante $I\wedge$ a partir de otras, que corresponden a los términos

   $$u\,[x_1...x_n] \quad \text{y} \quad v\,[x_1...x_n]$$

   asociamos a la conjunción el par

   $$\langle u\,[x_1...x_n]\,,\ v\,[x_1...x_n]\rangle$$

   ya que la prueba de una conjunción es un par. (Hacemos que las variables de las que dependen u y v sean las mismas.)

3. *Eliminación de la conjunción:* Si una deducción acaba con semejante eliminación $E1\wedge$ y el término $t\,[x_1...x_n]$ estaba asociado con la subprueba inmediatamente precedente, asociamos a la prueba final

   $$\pi^1 t\,[x_1...x_n]$$

   La razón es que puesto que la demostración de una conjunción es un par, asignamos a esta prueba de eliminación su proyección primera o segunda, dependiendo de qué miembro de la conjunción se tome.

4. *Introducción del condicional:* Si una deducción acaba con $I \rightarrow$ asociamos a nuestra deducción la función $t\,[x_1...x_n]$ que manda cada argumento $a \in A$ a $v[a\ x_1...x_n]$. Esto se expresa mediante el término

   $$\lambda x\ v\,[x\ x_1...x_n]$$

   De esta manera la demostración de un condicional es una función del cálculo lambda.

5. *Eliminación del condicional:* El caso de una deducción que termina en $E \rightarrow$ es tratado considerando las dos funciones $u\,[z - z_n]$ y $t\,[z - z_n]$

asociadas a las dos subpruebas inmediatamente anteriores —de A y de $A \to B$— y asignando

$$t\,[x_1...x_n]\,(u\,[x_1...x_n])$$

al resultado final. La razón es que t es una función de A en B y por lo tanto $t\,(u)$ es un elemento de B.

Ecuaciones fundamentales

$\pi^1\,\langle u, v\rangle = u \qquad \pi^2\,\langle u, v\rangle = v \qquad \langle \pi^1 t, \pi^2 t\rangle = t$

$(\lambda x\ v)\,u = v\left(\frac{u}{x}\right) \qquad \lambda x\ tx = t$

11.4. La Identidad

11.4.1. Identidad en *FOL.*

En la lógica de primer orden se suele introducir entre los signos lógicos el de igualdad, que se estipula ha de interpretarse como identidad. Es decir, la identidad es un concepto lógico, primitivo.

En el lenguaje formal, a partir de términos cualesquiera τ y t formamos la expresión

$$\tau = t$$

que será verdadera en una interpretación $\Im$ cuando sus denotaciones coincidan; es decir,

$$\Im(\tau = t) = V \text{ syss } \Im(\tau) = \Im(t)$$

El cálculo deductivo nos debería permitir deducir las leyes habituales de: reflexividad, simetría, transitividad, sustitución de iguales, etc. Para ello se añaden algunas reglas al cálculo deductivo; por ejemplo, nosotros tomamos las reglas de reflexividad de la igualdad y de sustitución de iguales como reglas primitivas de inferencia:

$$\textbf{RI} \quad \frac{}{\dashv \tau = \tau}$$

$$\textbf{SI} \quad \frac{\Omega \dashv \varphi\left(\frac{\tau}{x}\right)}{\Omega,\ \tau = t \dashv \varphi\left(\frac{t}{x}\right)}$$

y las restantes leyes mencionadas se obtienen como teoremas.

El que la interpretación del signo de igualdad sea la genuina relación de identidad —es decir, la identidad genérica, la que mantiene un objeto consigo mismo, pero sólo consigo mismo— no se lo debemos a la capacidad expresiva de la lógica de primer orden; sencillamente es un concepto primitivo y se toma así del metalenguaje.

Pero,

¿Se puede definir la identidad en FOL sin igualdad?

¿Hay una φ *que la defina?*; es decir, tal que para cada $\Im = \langle \mathcal{A}, H \rangle$

$$\Im(\varphi) = \{\langle \mathbf{x}, \mathbf{y} \rangle \in \mathbf{A} \times \mathbf{A} \mid \mathbf{x} = \mathbf{y}\}$$

La respuesta es que nó. Sin embargo, en los casos en los que el lenguaje sólo tenga un número finito de relatores; por ejemplo, R (monario) y T (binario), la fórmula

$$(Rx \leftrightarrow Ry) \wedge \forall z(Txz \leftrightarrow Tyz) \wedge \forall z(Tzx \leftrightarrow Tzy) \tag{11.7}$$

expresa que x e y no se pueden distinguir mediante fórmulas del lenguaje de primer orden. Esta *"definición"* obedece las leyes habituales de la igualdad, pero hay modelos en donde la relación que define esta fórmula no es la de identidad genérica.

Ejemplo 364 *En el modelo* $\Im = \langle \mathcal{A}, H \rangle$ *donde* $\mathcal{A} = \left\langle \{1,2,3\}, T^{\mathcal{A}}, R^{\mathcal{A}} \right\rangle$, *siendo* $T^{\mathcal{A}} = \emptyset$ *y* $R^{\mathcal{A}} = \{1,2,3\}$ *la relación definida por la fórmula (11.7) no es la identidad.*

Comentario 365 *Aunque sea tan sólo una relación similar a la de identidad y requiera un alfabeto con un número finito de relatores, es la mejor aproximación que tenemos en primer orden al* ***Principio de Leibniz de los Indiscernibles*** *—según el cual dos individuos son iguales si comparten todos sus atributos—.*

11.4.2. Identidad en *HOL*

Aquí veremos cómo se define la identidad usando otros conceptos lógicos.

Identidad para individuos en *HOL*

En cualquier lógica de orden superior, como la cuantificación se extiende a variables predicativas, el principio de Leibniz se formula así:

$$\forall X(Xx \leftrightarrow Xy) \tag{11.8}$$

y la igualdad para individuos se introduce mediante la siguiente definición formal:

$$\forall xy(x = y \leftrightarrow_{Df} \forall X(Xx \leftrightarrow Xy))$$

No necesitamos entonces reglas primitivas para la igualdad, pues

$$\mathbf{RI} \quad \frac{}{\dashv \tau = \tau}$$

$$\mathbf{SI} \quad \frac{\Omega \dashv \varphi\left(\frac{\tau}{x}\right)}{\Omega,\ \tau = t \dashv \varphi\left(\frac{t}{x}\right)}$$

son ambas reglas derivadas de inferencia.

No es ésta la única forma de definir la identidad, pues también puede usarse

$$\forall Y(\forall z Yzz \rightarrow Yxy) \tag{11.9}$$

—Dos individuos son iguales cuando están en todas las relaciones reflexivas.—

Estas dos fórmulas son sintácticamente equivalentes en cualquier cálculo deductivo de orden superior, por ejemplo en $\lambda - C_2$. Éste es uno de los que utilizo para la lógica de segundo orden en [54].

$$\vdash_{\lambda - C_2} \forall xy(\forall X(Xx \leftrightarrow Xy) \leftrightarrow \forall Y(\forall zYzz \rightarrow Yxy))$$

Aunque la igualdad definida de cualquiera de las formas propuestas se comporta adecuadamente, la pregunta es,
¿Definen a la identidad prototípica estas fórmulas?

La respuesta es que la relación definida mediante la fórmula 11.8 es la *"genuina"* en cualquier *Estructura Estándar* de orden superior. De hecho, podemos usar la fórmula

$$\forall X(Xx \rightarrow Xy)$$

La razón es que las dos fórmulas son equivalentes en la clase formada por todas las estructuras estándar:

$$\models_{\mathfrak{S}.\mathfrak{S}} \forall xy(\forall X(Xx \rightarrow Xy) \leftrightarrow \forall X(Xx \leftrightarrow Xy))$$

(pues toda estructura estándar contiene a todos los conjuntos unitarios en el dominio de conjuntos)

También la relación definida mediante (11.9) es la *"genuina"* en cualquier estructura estándar de segundo orden, pues la menor relación reflexiva es precisamente la de identidad y ésta la tenemos en el dominio de las relaciones binarias de toda estructura estándar. Todo lo dicho se hace extensivo a cualquier otra lógica de orden superior.

Marcos y modelos generales

HOL con semántica estándar tiene un extraordinario poder expresivo pero son escasas sus propiedades lógicas[14]: falla completud, tanto en sentido débil como fuerte; falla compacidad y falla Löweheim-Skolem. Sin embargo, la aritmética puede caracterizarse hasta isomorfía y algunas propiedades del universo matemático cifradas en axiomas tales como el de infinitud, elección o las hipótesis del continuo, pueden expresarse en *HOL*.

Para recuperar algunas propiedades lógicas propias de *FOL* se puede modificar la semántica: se introduce semántica no-estándar, en particular los *Marcos y Modelos Generales*. Al ampliar la clase de modelos el conjunto de fórmulas válidas se reducirá. Esta es la situación con las definiciones de igualdad propuestas; la fórmula

$$\forall X(Xx \leftrightarrow Xy)$$

es equivalente a

$$\forall Y(\forall zYzz \rightarrow Yxy)$$

[14] Lo vimos para *SOL* en el capítulo 10, pero vale igualmente para cualquier otra lógica de orden superior.

en sentido estándar, incluso en sentido general, pero no en marcos. Tampoco definen a la identidad en todo marco.

¿Qué pasaría si estando en HOL (sin igualdad primitiva) cambiásemos la semántica y adoptásemos la de marcos o la de modelos generales?

La respuesta es que volveríamos a una situación propia de la *FOL* (sin = primitiva): se puede concebir un marco $\mathcal{A}$ y una interpretación $\Im = \langle \mathcal{A}, H \rangle$ tal que

$$\Im(x = y) = V \text{ pero } \Im(x) \neq \Im(y)$$

Ejemplo 366 *Tomad una estructura cualquiera* $\mathcal{A}$ *con* $\mathbf{A} = \{1, 2, 3\}$ *y donde* $\mathbf{A}_1 = \{\emptyset, \mathbf{A}\}$)

¿Qué se puede hacer para remediarlo?

Hay dos vías de solución:

- Tomar la igualdad como signo lógico, primitivo y actuar como en *FOL* con igualdad.
  1. Se añade el signo $=$ al lenguaje formal (signo lógico)
  2. Como denotación tomamos del metalenguaje la identidad prototípica. Haciéndolo de esta forma, la expresión
     $$\tau = t$$
     será verdadera en un modelo-marco si sus denotaciones coinciden
     $$\Im(\tau = t) = V \text{ syss } \Im(\tau) = \Im(t)$$
  3. Añadir (RI) y (SI) como reglas primitivas de inferencia
- Eliminar los marcos problemáticos

Identidad para conjuntos y relaciones

Tradicionalmente en lógica de segundo orden no se suele incluir la igualdad para conjuntos y relaciones entre los signos. Pero, *¿ Se puede definir la igualdad para conjuntos y relaciones?*.

En caso afirmativo, *¿Podemos usar el principio de Leibniz?*

Para usar el principio de Leibniz necesitaríamos lógica de tercer orden:

$$\forall X^{0(01)}(X^{0(01)}Z^{(01)} \leftrightarrow X^{0(01)}Y^{(01)})$$

Lo que se puede hacer es usar el *Principio de Extensionalidad* para introducir el signo de igualdad entre relaciones:

$$\forall X^n Y^n (X^n = Y^n \leftrightarrow \forall x_1 ... x_n (X^n x_1 ... x_n \leftrightarrow Y^n x_1 ... x_n) \tag{11.10}$$

Haciendo esto, la fórmula $\Pi^n = \Psi^n$ debe entenderse como

$$\forall x_1 ... x_n (\Pi^n x_1 ... x_n \leftrightarrow \Psi^n x_1 ... x_n) \tag{11.11}$$

Quisiera destacar los puntos siguientes:

1. La fórmula (11.11) establece una relación de equivalencia entre los conjuntos y relaciones de un modelo dado, pero no siempre se trata de la identidad prototípica.

2. Sin embargo, al introducir la igualdad mediante la fórmula (11.10) ésta deja de funcionar como principio de extensionalidad, pasando a ser una mera definición formal del signo de igualdad.

3. Como normalmente tenemos extensionalidad en el metalenguaje (no consideramos más que modelos extensionales), el que se pierda la capacidad de postularlo axiomáticamente no es grave.

4. Además, la construcción de un modelo extensional a partir de uno no extensional es fácil: al ser de equivalencia la relación definida por (11.11), se construye el modelo cociente.

Identidad global

En la teoría de tipos simple no tenemos algo tan básico como la función global de identidad: para cada tipo τ tenemos la identidad de dicho tipo $I_\tau := \lambda x^\tau\, x^\tau$. Si usamos una variable $\widehat{t}$ para tipo, la función identidad sería una generalización de $\lambda x^{\widehat{t}}\, x^{\widehat{t}}$. Pero esto no es una expresión bien formada del formalismo; $\widehat{t}$ no es un tipo. Para conseguir que la identidad sea una expresión lo que se hace es utilizar Λ, un abstractor de tipos; la identidad polimorfa es definida como

$$I :=_{Df} \Lambda\widehat{t}\, \lambda x^{\widehat{t}} x^{\widehat{t}}$$

La colección de tipos se enriquece así con un $\forall$ y $\Lambda\widehat{t}\, \lambda x^{\widehat{t}} x^{\widehat{t}}$ es de tipo $\forall\widehat{t}\langle\widehat{t},\widehat{t}\rangle$. Una expresión cuyo tipo es $\forall\widehat{t}\,\sigma$ puede aplicarse a una de tipo τ y origina una de tipo $\sigma\left(\frac{\tau}{\widehat{t}}\right)$

Por ejemplo, la expresión

$$\Lambda\widehat{t}\, \lambda x^{\widehat{t}} x^{\widehat{t}} \tau \tag{11.12}$$

es de tipo $\langle\tau,\tau\rangle$

Si se usa una regla de reducción β para tipos, de la expresión $(\Lambda t\ E)\,\tau$ se pasa a $E\left(\frac{\tau}{t}\right)$. En nuestro caso (11.12) se reduce a $\lambda x^\tau x^\tau$; es decir a I_τ.

¿Qué opción propongo?

En lógica de segundo orden, cuando se quiera usar modelos no estándar propongo tomar el genérico de identidad en todos los casos. En [54] presento al efecto los lenguajes *SOL* y *λ-SOL*.

1. En el lenguaje *SOL* la igualdad, tanto para individuos, como para conjuntos y relaciones, se toma como signo lógico, primitivo. Siendo así el *Axioma de Extensionalidad*

$$\textbf{Ext}\ \frac{}{\dashv \forall X^n Y^n (X^n = Y^n \leftrightarrow \forall x_1 ... x_n (X^n x_1 ... x_n \leftrightarrow Y^n x_1 ... x_n)}$$

que se incluye como regla sin premisas, surte el efecto deseado; esto es, excluye modelos no extensionales.
Al combinar su acción con la del *Esquema Axiomático de Comprehension*

$$\mathbf{EC}\ \frac{}{\dashv \exists X^n \forall x_1 ... x_n (X^n x_1 ... x_n \leftrightarrow \varphi)}$$

que se incluye también como regla sin premisas, nos permite demostrar

$$\vdash_{C_2} \exists ¡ X^n \forall x_1 ... x_n (X^n x_1 ... x_n \leftrightarrow \varphi)$$

¿Qué significado y alcance tiene este teorema?.
Siempre que tengamos extensionalidad y comprehensión —con igualdad como signo primitivo— podemos darle un nombre a la relación definida por cada fórmula φ —módulo cierta sucesión de variables—, puesto que es única. Esto quiere decir, que se puede extender de forma *"conservativa"* el lenguaje mediante un operador de abstracción λ formando así predicados de la forma

$$\lambda x_1 ... x_n \varphi$$

cuya interpretación es:

$$\Im(\lambda x_1 ... x_n \varphi) = \left\{ \langle \mathbf{x}_1, ..., \mathbf{x}_n \rangle \mid \Im^{\mathbf{x}_1 ... \mathbf{x}_n}_{x_1 ... x_n}(\varphi) = V \right\}$$

Al escribirlo en notación lambda, el axioma de comprehensión queda así:

$$\forall z_1 ... z_n (\lambda x_1 ... x_n \ \varphi \ z_1 ... z_n \leftrightarrow \varphi)$$

2. El lenguaje λ-*SOL*, contiene igualdad y abstracción como signos lógicos, primitivos, permitiéndonos formar predicados a partir de fórmulas de la manera mencionada antes. Su cálculo deductivo contiene las reglas de *Introducción del Abstractor en el Consecuente*

$$\mathbf{IAC}\ \frac{\Omega \dashv \varphi \left(\frac{\tau_1 ... \tau_n}{x_1 ... x_n} \right)}{\Omega \dashv \lambda x_1 ... x_n \ \varphi \ \tau_1 ... \tau_n}$$

y de *Introducción del Abstractor en el Antecedente*

$$\mathbf{IAA}\ \frac{\Omega \ \ \varphi \left(\frac{\tau_1 ... \tau_n}{x_1 ... x_n} \right) \dashv \psi}{\Omega \ \ \lambda x_1 ... x_n \ \varphi \ \tau_1 ... \tau_n \dashv \psi}$$

En este cálculo se puede demostrar como teorema el axioma de comprehensión, no precisándose como regla primitiva

$$\vdash_{\lambda - C_2} \exists X^n \forall x_1 ... x_n (X^n x_1 ... x_n \leftrightarrow \varphi)$$

Otro teorema de nuestro cálculo $\lambda - C_2$ nos va a servir de introducción al siguiente tema que quisiera, aunque sea muy brevemente, tratar aquí. El teorema es éste:

$$\vdash_{\lambda - C_2} \forall x \varphi \leftrightarrow \lambda x \varphi = \lambda x \ x = x$$

(En un lenguaje de segundo orden que contiene como primitivos λ e igualdad, la cuantificación universal equivale a la igualdad de los predicados $\lambda x \varphi$ y $\lambda x\, x = x$.)

11.5. Teoría de tipos ecuacional

Hemos planteado el tema de la identidad en el contexto de la lógica de primer orden y en el de la de segundo orden: principalmente nos preocupaba el saber si podíamos definirla en función de otros signos lógicos o si en vez de conseguir el *"genérico"* teníamos que conformarnos con *"similares"* o *"equivalentes"*[15]

En esta última parte el problema que planteo es el inverso y también recibe respuesta afirmativa en lógica superior.

¿Se pueden definir, con sólo identidad y abstracción como primitivos, los restantes signos lógicos?

La idea de reducir otros conceptos al de identidad es muy antigua y fue tratada con algún resultado por Tarski, quien resolvió lo concerniente a los conectores, Ramsey, que planteó el tema en su conjunto, y Quine, quien introdujo los cuantificadores. Fue finalmente completamente resuelto por Henkin y mejorado por Andrews.

Las definiciones de todos estos conceptos (conectores y cuantificadores) son posibles en un lenguaje de orden superior; por ejemplo, el de la teoría de tipos finitos que incluye variables proposicionales para todos los órdenes finitos —esto es, incluye el 0 y todos los que se obtienen del 0—. De hecho, la teoría de tipos proposicional es una subteoría importante de esta presentación de teoría de tipos y se puede estudiar separadamente. El cálculo que introduciremos para la teoría de tipos ecuacional, cuando trabaja exclusivamente con tipos proposicionales, genera todas y solas las fórmulas válidas de la lógica de tipos proposicionales, como mostró Henkin en 1963.

Comentario 367 *Este cálculo es de una belleza y naturalidad extraordinarias y parece estar concebido para prestar apoyo a la tesis de Wittgenstein sobre fundamentación de las matemáticas, según la cual toda la matemática está hecha de ecuaciones, no de tautologías.*

Definiciones básicas

El lenguaje de tipos ecuacional que usaremos sólo contiene: Variables para todos los tipos finitos —con un superíndice que lo indica—; igualdad para todos los tipos; abstractor λ.

1. Bicondicional: Para fórmulas φ y ψ, la expresión

$$\varphi \leftrightarrow \psi$$

se usará en vez de

$$((=^{\langle\langle 00\rangle 0\rangle}\ \varphi)\psi)$$

[15] Véase [57] y [58].

abreviado, $\varphi = \psi$

2. Lo verdadero: $\top^0$. Su definición abreviada es
$$\lambda X^0 X^0 = \lambda X^0 X^0$$

3. Lo falso: $\bot^0$. Su definición abreviada es
$$\lambda X^0 X^0 = \lambda X^0 \top$$

4. Negación: $\neg^{\langle 00\rangle}$. Su definición abreviada es
$$\lambda X^0 (\bot = X^0)$$

5. Predicado de existencia: $\Sigma^{\langle 0\langle 0\alpha\rangle\rangle}$. Su definición abreviada es
$$\lambda X^{\langle 0\alpha\rangle} \lambda X^\alpha (X^{\langle 0\alpha\rangle} X^\alpha) \neq \lambda X^\alpha \bot$$

6. Cuantificador existencial: $\exists X^\alpha \varphi$. Su definición abreviada es
$$\lambda X^\alpha \varphi \neq \lambda X^\alpha \bot$$

7. Disyuntor: $\vee^{\langle 0\langle 00\rangle\rangle}$. Su definición abreviada es
$$\lambda X^0 (\lambda Y^0 \exists X^{\langle 00\rangle} ((X^{\langle 00\rangle} \neg X^0 = X^{\langle 00\rangle} \top) \neq \neg Y^0))$$

Explicación

La semántica empleada es estándar, pero se puede también usar la de marcos o la de modelos generales.

Sea $\Im = \langle \mathcal{D}, M\rangle$ un modelo. Es fácil comprobar que con las siete definiciones anteriores:

1. $\Im(\varphi \leftrightarrow \psi) = V$ syss $\Im(\varphi) = \Im(\psi)$ —Funciona como el bicondicional—
2. $\Im(\top) = V$ —Es un nombre para lo verdadero—
3. $\Im(\bot) = F$ —Es un nombre para lo falso—
4. $\Im(\neg)$ es una función de tipo $\langle 00\rangle$ cuyo valor es V syss $\Im(X^0) = F$. —Es la función veritativa de negación—
5. $\Im(\Sigma)$ es una función que asigna a los conjuntos de tipo $\langle 0\alpha\rangle$ el valor V syss el conjunto no es vacío. —Es un predicado de predicados, afirma del predicado que *"no predica en el desierto"*, que bajo él cae cuanto menos un objeto—
6. La cuantificación existencial $\exists X^\alpha \varphi$ podría reducirse a lo anterior, $\Sigma(\lambda X^\alpha \varphi)$. Hemos optado por definirlo directamente.
7. Finalmente, incluso la disyunción puede definirse usando identidad y lambda; la fórmula se interpreta como la función que asigna F syss
$$\text{tanto } \Im(X^0) = F \text{ como } \Im(Y^0) = F$$

Un cálculo para *ETT*

Cálculo axiomático: Sólo contiene cuatro axiomas y una regla de reemplazamiento.

Axiomas

(Ax.1) $(X^{\langle 00\rangle}\bot \wedge X^{\langle 00\rangle}\top) = \forall X^0(X^{\langle 00\rangle}X^0)$ —Hay sólo dos valores de verdad—

(Ax.2) $(X^\alpha = Y^\alpha) \to (Z^{\langle 0\alpha\rangle}X^\alpha = Z^{\langle 0\alpha\rangle}Y^\alpha)$ —Sustitución de iguales—

(Ax.3) $\forall X^\beta(X^{\langle\alpha\beta\rangle}X^\beta = Y^{\langle\alpha\beta\rangle}X^\beta) = (X^{\langle\alpha\beta\rangle} = Y^{\langle\alpha\beta\rangle})$ —Extensionalidad—

(Ax.4) $(\lambda X^\beta B^\alpha)A^\beta = B^\alpha\left(\frac{A^\beta}{X^\beta}\right)$ —Conversión lambda—

(Ax.5) $t^{\langle 1\langle 01\rangle\rangle}(\lambda X^1(X^1 = Y^1)) = Y^1$ —Descripción—

Regla de inferencia: R

Reemplazamiento A partir de φ y de $A^\alpha = B^\alpha$ inferir cualquier fórmula ψ que se obtenga de φ al reemplazar en φ cualquier parte de la forma A^α por B^α.

11.6. Apéndice: La piedra filosofal

En lo que sigue intentaré hacer plausible mi tesis de que el gran descubrimiento de Church fue, sin lugar a dudas, el del cálculo lambda, y que el resto de sus contribuciones, así como las de algunos de sus discípulos se deben a este acierto inicial[16]. El hilo argumental será éste:

(1) El cálculo lambda

En las primeras décadas de este siglo seguramente se consideraba imprescindible para un lógico que se preciara el presentar un sistema lógico propio. Church creó el cálculo lambda tomando como base el concepto de función y distinguiendo claramente entre el valor de una función para un argumento $F(x)$ y la propia función $\lambda xF(x)$. Para ilustrarlo pongamos uno de los ejemplos que el propio Church usa[17]:

> If we say, $(x^2 + x)$ *is greater than 1,000* we make a statement which depends on x and actually has no meaning unless x is determined as some particular natural number. On the other hand, if we say (x^2+x) *is a primitive recursion function*, we make a definite statement whose

[16] Esta idea la desarrollo en [55] y en [56].
[17] Véase [14]. (Página 6 de la edición de 1951.)

> meaning in no way depends on a determination of the variable x (so that in this case x plays the rôle of an apparent, or bound variable). The difference between the two cases is that in the first case the expression (x^2+x) serves as an ambiguous, or variable, denotation of a natural number, while in the second case it serves as the denotation of a particular function. We shall hereafter distinguish by using (x^2+x) when we intend an ambiguous denotation of a natural number, but $(\lambda x(x^2+x))$ as the denotation of the corresponding function —and likewise in other cases—.

El elegir el concepto de función como base fue ya un gran acierto, pero el distinguir en el lenguaje los valores de la función de la función misma, lo fue aún mayor. Si a esto añadimos la característica de que la formalización permite que una función anide a otras, el lenguaje está preparado para expresar la recursión. De hecho, a diferencia de los sistemas axiomáticos fundacionales, la existencia de los objetos matemáticos (en nuestro caso, funciones) no se postula, sino que son generados por las propias reglas de formación de expresiones.

Este lenguaje es tan compacto y tan adecuado para expresar funcionalidad que se convirtió en lenguaje de programación[18].

El cálculo lambda, aunque llevaba circulando varios años, no aparece en prensa hasta principios de los cuarenta[19]. A comienzos de los años treinta Church ya sabía que tanto la función del siguiente como la suma eran $\lambda-$definibles, pero fue su alumno Kleene quien fue gradualmente descubriendo lo amplia que era la clase de las funciones definibles mediante lambda y estos resultados constituyeron parte de su tesis. Este es el origen de la teoría de la recursión. Sin embargo, los resultados de la investigación les llevaron más lejos de lo esperado. Veamos lo que dice Martin Davis [23] al respecto:

> Church published a pair of substantial papers on the system he developed and set his students Stephen C. Kleene and J. Barkley Rosser to work on it. Their work was extremely effective, if not exactly what one dreams of having one's graduate students accomplis for one: Kleene and Rosser proved that Church's system was inconsistent!

Pese a ese tropiezo inicial, pronto se vio que la definibilidad mediante lambda equivalía a la recursividad y a la computabilidad mediante máquinas de Turing, proporcionando así un argumento de gran fuerza a favor de la denominada tesis de Church; es decir, la de tomarlas como una definición posible de computabilidad efectiva o algoritmo. Además, usando la mencionada tesis, Church demostró la indecidibilidad de la lógica de primer orden.

[18] Se atribuye la inspiración para crear el lenguaje LISP, desarrollado por John Mc Carthy, al cálculo lambda.

[19] El pequeño volumen [14] es interesante. Una presentación moderna se encuentra en los libros de Hindley [42] y [**?**] y en el de Barendregt [7].

Veremos más adelante cómo el cálculo lambda sirvió también de base a la Teoría de tipos, produciendo una presentación muy elegante de la misma. Y no sólo eso, dio origen a la prueba de completud de Henkin, la que seguimos actualmente en cualquier lógica, no sólo en la teoría de tipos con lambda.

(2) Entscheidungsproblem

Uno de los temas que más interesaba en aquel momento (estamos en los años treinta) era el de la decidibilidad. Hilbert y Ackerman [41] lo expresan así:

> El Entscheidungsproblem se resuelve cuando uno conoce un procedimiento que le permite decidir, mediante un número finito de operaciones, acerca de la validez, respectivamente la satisfacibilidad, de una expresión lógica.

Este problema estaba en el centro de los intereses de la escuela de Hilbert, quienes por aquel entonces esperaban una solución positiva del famoso problema décimo. Pronto, sin embargo, se pasó a esperar una respuesta negativa pues ya se veía que en caso contrario traería consecuencias poco plausibles. En particular, tras la demostración de Gödel de su teorema de incompletud, el panorama y las expectativas eran completamente diferentes[20].

Como comenté anteriormente, mostrar que un conjunto es decidible no requiere más que señalar el algoritmo que lo genera. Y puesto que la noción intuitiva basta para identificar a un procedimiento como efectivo, puede hacerse sin una definición matemática de computabilidad. Sin embargo, para demostrar que no es decidible no basta con tener una noción intuitiva, se necesita una definición en regla.

O sea, que del deseo de resolver problemas de indecidibilidad se derivaba la necesidad imperiosa de contar con una definición matemática de computabilidad. Y en particular, el resolver el famoso problema décimo de Hilbert pudo muy bien servir de acicate en Church para ponerse a la tarea de proporcionar tal definición[21]. Puesto que las investigaciones de Kleene y las suyas propias habían mostrado el enorme poder expresivo del lenguaje introducido por Church, parecía razonable pensar que una tal definición podía ser precisamente la $\lambda-$definibilidad.

Por lo que al *Entscheidungsproblem* se refiere, Church demostró que para la lógica de primer orden no es posible encontrar un algoritmo que determine si una fórmula es o no válida. El artículo en el que Church prueba este teorema [12] apareció en 1936 y está basado en un artículo suyo del mismo año [11] y en la gödelización [22].

[20] Gandy [30] expone maravillosamente cual era la situación.

[21] Sabemos que fue tambien ésta la razón que movió a Turing a definir el concepto de computable mediante una máquina abstracta; sus famosas *máquinas*.

[22] También se dice que el teorema de incompletud, caso de no haber sido probado por Gödel, podria haberse derivado de los resultados de Church; veamos lo que dice Shoenfield [66] en

(3) Computabilidad: recursividad

La Teoría de la recursión clásica abarca el estudio de las funciones definidas sobre los naturales, pero la teoría de la recursión actual, al ir desarrollando su potencial propio y sus métodos específicos, ha alcanzado un notable desarrollo abstracto y aplicaciones insospechadas. Los orígenes de la teoría clásica pueden hallarse en Dedekind, cuando en 1888 introduce el estudio de las funciones definibles sobre el conjunto de los números naturales usando ecuaciones y, recurrentemente, la inducción sobre los naturales que él había formulado y precisado. De ahí viene, justamente, el que se adoptara el nombre de Teoría de la recursión.

Por lo que respecta a su estadio presente, cuyo radio de acción cubre la totalidad de las funciones efectivamente computables, los orígenes hay que buscarlos en el grupo de Princeton; empezó con Church, pero si hay que atribuirle un padre, éste es Kleene. El fue quien la impulsó, definió y acotó. Suyos son los teoremas de la forma normal y el de la recursión[23].

En cuanto a la definición misma, comentamos con anterioridad que circulaban varias versiones de este concepto, aunque, como veremos en el siguiente apartado, había cierta resistencia a aceptarlas como definición. Varios de estos conceptos aparecieron en los años 30 para caracterizar nociones que en un principio parecían diferentes: la primera era la caracterización de Gödel de las funciones definidas mediante recursión, la segunda era la noción de función definible con el operador λ, que Church y Kleene introdujeron, y la tercera, era la de función computable mediante una máquina abstracta, las máquinas de Turing. Pronto se demostró que los tres nociones definían las mismas funciones. Si bien estaba claro que todas las funciones definidas mediante cualquiera de los procesos anteriores era efectivamente computable, este resultado seguía sin servir para demostrar que un conjunto es indecidible; necesitamos la conversa. Por su propia naturaleza esta afirmación no puede ser demostrada, y si no se considera adecuado que sea una definición, hay que proporcionarle otro status.

(4) Tesis de Church

Otro de los grandes méritos suyos es el haber formulado la tesis de que el concepto matemático preciso de recursión, tal y como lo había definido Turing, se correspondía exactamente con el concepto intuitivo de computabilidad efectiva.

Shoenfield [66] dice que la tesis de Church surgió de manera casual, yo diría algo muy parecido, que surgió de forma natural. Habiendo introducido su sistema lambda y estando interesado en el alcance de las funciones $\lambda-$definibles, se

una nota a pie de página al respecto:

"Kleene has pointed out that if Gödel had not already proved his Incompleteness Theorem, Church might well have derived it from his results. The reader interested in such a derivation of the Incompleteness Theorem can consult my book Mathematical Logic. Of course, this takes nothing away from Gödel's supreme achievement in discovering the Incompleteness Theorem".

[23] Davis [22] es una antología de los artículos básicos de este campo. En Kleene [51] se hace un estudio histórico de los inicios.

fue dando cuenta gradualmente (como dije, en parte gracias a su alumno Kleene) de que dicha clase era muy, pero que muy extensa. Estaba claro que las funciones $\lambda-$definibles eran computables y parecía razonable pensar que todas las computables eran $\lambda-$definibles. Es decir, que la noción intuitiva de computabilidad (o efectividad) se correspondía exactamente con la de $\lambda-$definibilidad. En un principio la formuló como definición y luego Kleene la cambió al status de tesis, con el que la conocemos actualmente. En su primera formulación computabilidad efectiva fue identificada con $\lambda-$definibilidad, públicamente fue anunciada como recursividad y más tarde fue generalmente aceptada como computabilidad mediante máquinas de Turing. Las vicisitudes de la tesis pueden verse bien en Davis [23].

Aunque por tratarse de una tesis no se puede demostrar matemáticamente, hay razones poderosas para aceptarla:

(1) La primera es una razón basada en la experiencia acumulada: todos los algoritmos conocidos son recursivos. Gandy [30] añade al respecto:

> Kleene's dissertation shows, as we might now put it, the power of the $\lambda-$calculus as a high-level programming language. A $\lambda-$term can take other $\lambda-$terms (subroutines), instead of numbers, as inputs; and, if taken in the right order, successive conversions do correspond to a natural way of implementing the program wich the $\lambda-$term represents. Of course Kleene did not think in these terms, but this way of putting it helps one to appreciate the intuitions which led Church and Kleene to believe in the truth of the thesis.

(2) La segunda está basada en la equivalencia entre los tres conceptos introducidos para establecer la noción de algoritmo: resulta tranquilizador que desde planteamientos tan distintos se lleguen a definir exactamente las mismas funciones.

(3) La tercera, que fue la que sin duda convenció a Gödel, está relacionada con el análisis que Turing hizo del concepto de computabilidad; no se interesa tanto por la naturaleza de las funciones computables como por el propio proceso de computación. Al comienzo de su trabajo Turing afirma:

> The real question at issue is *What are the possible processes which can be carried out in computing a number?*

Las funciones se construyen a partir de funciones elementalísimas de las que no cabe duda de su carácter algorítmico mediante procesos que tampoco plantean dudas al respecto.

(4) De hecho, el argumento anterior es muy similar al dado por Church, que expone Gandy, también en [30] así:

> *The Step-by step Argument.* This is Church's chief argument. He considers the evaluation of a value fm of a function either by the application of an algoritm, or by the derivation of $fm = n$ from axioms about f in some formal system. In each case the evaluation

> proceeds in a series of steps. If each step is recursive then f will be recursive. As evidence for the premise he points out that Gödel had shown that the steps in a proof in Gödel system P are primitive recursive.

(5) Finalmente, la razón que convenció a Kleene: la del fracaso del argumento diagonal[24]. Llamamos diagonalización al proceso que mimetiza el que Cantor introdujo para demostrar que $\wp(\omega)$ no es biyectable con ω y que permite señalar conjuntos (en el caso mencionado, $Y = \{x \mid x \notin f(x)\}$) que no han sido incluídos en una enumeración, pretendidamente completa de un determinado dominio.

Ahora se podía demostrar que un conjunto no era decidible pues bastaba con mostrar que no era recursiva la función que cifraba la pertenencia a él. Esto lo hizo Church en 1936 para un problema del cálculo lambda. En el mismo año, apoyándose en los resultados de Gödel, Church demostró que si fuera decidible la lógica de primer orden, también lo sería el sistema del artículo anterior.

(5) El cálculo lambda y la teoría de tipos

En primer lugar, el cálculo λ de Church permite indagar la naturaleza de las operaciones matemáticas y hemos visto que se consiguen resultados de gran importancia asociados a la definibilidad mediante λ. En segundo lugar, y de eso nos ocupamos ahora, el cálculo lambda —que introdujimos en las sección 11.2— sirve también para hacer una magnífica presentación de la teoría de tipos.

Posiblemente para solucionar el problema de inconsistencia del cálculo lambda sin tipos, Church formuló la teoría de tipos que Russell y Withehead habían introducido en los Principia en su potente y peculiar lenguaje lambda y para él definió un cálculo con algunas de las reglas de la conversión lambda. El resultado fue una maravilla, pues a la capacidad formalizadora de las expresiones con lambda se añadía la naturalidad de la representación con tipos. Aquí viene a cuento el comentario 361.

Como mencioné con anterioridad, Church [13] propone un lenguaje λ con tipos simples en el que las denominadas paradojas lógicas o matemáticas, como la de Russell, o la de Cantor, desaparecen porque la propia sintaxis del lenguaje no permite que se formen las construcciones autorreflexivas causantes de las contradicciones. Las llamadas paradojas semánticas, tales como la del mentiroso, se evitan al adoptar la semántica de Tarski, como resultado de distinguir claramente lenguaje objeto y metalenguaje. Haciendo esto no precisa la complicación adicional de establecer órdenes, tal y como hace Russell en su teoría ramificada.

Pasados muchos años Church publica un artículo [15] en el que se compara la solución de Russell con la de Tarski.

[24] En el artículo [1] Enrique Alonso y yo tratamos de desentrañar el misterio de porqué el fallo del proceso de diagonalización sobre la clase de las funciones recursivas conevció a Kleene, tan fervientemente, de la tesis de Church.

Otra versión del cálculo lambda es la que publica Church [16] en los cincuenta. En ella se toma la tesis de Frege, conforme a la cual para entender un lenguaje no nos basta con conocer la estructura del lenguaje simbólico y el universo de objetos a los que el primero se refiere; hay que interponer una tercera instancia de entes abstractos; a saber, los sentidos o conceptos. Church en el artículo citado desarrolla una teoría matemática del sentido, en la acepción arriba mencionada, y de su relación con los objetos del universo. Para ello utiliza el lenguaje de tipos del artículo de 1940 y le añade una nueva jerarquía de tipos para los sentidos.

(6) Completud de Henkin

Cuando se introduce un lenguaje formal, con un cálculo deductivo asociado, y una semántica, es una cuestión obligada la de indagar acerca de la adecuación de estos dos procedimientos de selección de fórmulas. Gödel lo había resuelto positivamente para la lógica de primer orden y negativamente para cualquier sistema lógico capaz de contener la aritmética.

El cálculo lambda para la teoría de tipos, con la semántica habitual sobre una jerarquía estándar de tipos era capaz de expresar la aritmética y por consiguiente no podía ser otra cosa que incompleto.

No obstante, Henkin demostró que si se interpretan las fórmulas de una manera menos rígida, aceptando otras jerarquías de tipos que no tengan necesariamente que contener a todas las funciones, sino sólo a las definibles, se prueba fácilmente que toda consecuencia de un conjunto de hipótesis es demostrable en el cálculo. Las fórmulas válidas en esta nueva semántica, llamada general, se reducen hasta coincidir con las generadas por las reglas del cálculo.

Como todo el mundo sabe, la prueba de completud de Henkin consiste en demostrar que todo conjunto consistente tiene un modelo. Para hacerlo se extiende el conjunto consistente a uno máximamente consistente y ejemplificado y se construye el modelo que las fórmulas de este conjunto máximamente consistente están describiendo; pues estos conjuntos precisamente se caracterizan por ser una descripción pormenorizada de un modelo.

Por supuesto, no entraré aquí en el detalle de este teorema de Henkin[25], lo único que me interesa señalar es en qué modo su descubrimiento está relacionado con el cálculo lambda.

Según cuenta su creador, estaba tratando de representarse mentalmente las funciones que pueden ser nombradas mediante expresiones con λ . Claramente, si tenemos un universo de individuos numerable, el de conjuntos de individuos será supernumerable, pero puesto que el lenguaje es sólo numerable, los conjuntos que pueden ser nombrados es sólo numerable. El *clik* definitivo se produjo, según parece, cuando se dio cuenta de que al intentar representarse a los objetos de esta jerarquía de tipos, para eliminar repeticiones, estaba usando las reglas de conversión lambda y que por lo tanto implicaba a la sintaxis y al cálculo. Para

[25] Lo hice *extensamente* para la lógica de segundo orden en el capítulo precedente.

identificar a los objetos nombrados mediante M^α y N^α usaba al cálculo como criterio; en especial que, $\vdash M^\alpha = N^\alpha$. Finalmente, la prueba se cerró al darse cuenta de que para que el universo de los objetos nombrados mediante proposiciones (el de los valores de verdad) se redujera a sólo dos habría de ampliar los axiomas hasta que constituyeran un conjunto máximamente consistente, de suerte que las clases de equivalencia inducidas por la relación $\vdash M^\alpha = N^\alpha$ se redujeran a dos.

Extraeré una serie de frases claves del bonito e ilustrativo relato que hace Henkin [39] de su propio descubrimiento.

> I decided to try to see just which objects of the hierarchy of types did have names in $\mathfrak{T}$.
>
> ...
>
> As I struggled to see the action of functions more clearly in this way, I was struck by the realization that I have used $\lambda-$conversion, one of the formal rules of inference in Church's deductive system for the language of the theory $\mathfrak{T}$. All my efforts had been directed toward *interpretations* of the formal language, and now my attention was suddenly drawn to the fact that these were related to the *formal deductive system* for that language.
>
> ...
>
> As soon as I observed this, it occurred to me that if we were to add further cwffs of type 0 to the list of formal axioms, this would have the effect of reducing the number of elementes in $\mathcal{D}'_0$ and that ultimately, by taking a maximal consistent set of axioms, the number of elements in $\mathcal{D}'_0$ would be two.

Bibliografía

[1] Alonso, E y Manzano, M. [2002]. *"Kleene's homework"*. **History and Philosophy of Logic,** en prensa.

[2] Andrews, P. B. [1963]. *"A reduction of the axioms for the theory of propositional types"*. **The Journal of Symbolic Logic.** vol. 52. pp. 345-350.

[3] Andrews, P. B. [1965] *A Transfinite Type Theory with Type Variables.* Amsterdam: North Holland.

[4] Andrews, P.B. [1972]. *"General Models, Descriptions, and Choice in Type Theory"*. **The Journal of Symbolic Logic.** vol. 37. num. 2. pp. 385-394.

[5] Andrews, P.B. [1972]. *"General Models and Extensionality"*. **The Journal of Symbolic Logic.** vol. 37. num. 2. pp. 394-397.

[6] Andrews, P. B. [1986]. *An Introduction to Mathematical Logic and Type Theory: To Truth through Proof.* Orlando: Academic Press, Inc.

[7] Barendregt, H. P. [1981]. *The Lambda Calculus, its Syntax and Semantics.* Amsterdam: North Holland Publishing Company.

[8] Beth, E.W. [1965]. *Las Paradojas de la Lógica.* Ed. Castellana de Juan Manuel Lorente. Cuadernos Teorema 4. Valencia, 1978.

[9] Benthem, J& Doets, K. [1983]. *"Higher-order logic"*. en [31]

[10] Church, A. [1928]. *"On the law of excluded middle"*. Bulletin of the American Mathematical Society. vol. 34. pp. 75-78.

[11] Church, A. [1936]. *"An unsolvable problem of elementary number theory"*. American journal of mathematics. vol. 58

[12] Church, A. [1936]. *"A note on the Entscheidungsproblem"*. The Journal of Symbolic Logic. vol. 1, Number 1. Marzo de 1936. pp 40-41

[13] Church, A. [1940]. *"A formulation of the simple theory of types"*. The Journal of Symbolic Logic. vol. 5, pp. 56-68.

[14] Church, A. [1941]. *The calculi of lambda-conversion.* (Annals of Mathematical Studies. num. 6). Princeton: Princeton University Press. USA.

[15] Church, A. [1976]. *"Comparison of Russell's resolution of the semantical antinomies with that of Tarski"*. **The Journal of Symbolic Logic.** vol. 41. pp. 747-760.

[16] Church, A. [1951]. *"A formulation of the logic of sense and denotation"*. **Structure, Methods and Meaning**, Essays in Honor of Henry M. Sheffer, New York. pp. 3-24. (Revisado en NOUS: vol. 7 (1973), pp. 24-33; vol. 8 (1974), pp. 135-156; and vol. 27 (1993), pp. 141-157.)

[17] Church, A. [1995]. "*A theory of the meaning of names"*. **The heritage of Kazimierz Ajdukiewicz.** Rodopi

[18] Church, A. y Kleene, S. C. [1936]. *"Formal definitions in the theory of ordinal numbers"*. **Fundamenta Mathematica.** vol. 28. pp. 11-21.

[19] Church, A. y Quine, W. V. [1952]. *"Some theorems on definability and decidability"*. **The Journal of Symbolic Logic.** vol. 17. núm. 3..

[20] Copi, I. [1971]. *The Theory of Logical Types*. London: Routledge & Kega Paul.

[21] van Dalen, D. [2001]. *"Intuitionistic Logic"*. en [31]

[22] Davis, M. [1965]. *The Undecidable.* Rave Press. Nueva York.

[23] Davis, M. [1982]. *"Why Gödel Didn't Have Church's Thesis"*. **Information and Control,** 54, pp. 3-24.

[24] Davis, M. [1988]. *Mathematical Logic and the origins of modern computers.* (en Herken [40]).

[25] Davis, M. [1995]. *"American Logic in the 1920s"*. **Bulletin of Symbolic Logic**, vol. 1, Number 3, Sept. 1995.

[26] Feferman, S. [1988]. *"Turing in the Land of O(z)"* (en Herken [40]).

[27] Frege, G. [1879]. *"Begrgriffsschrift, eine der arithmetischen nachgebildete Formlesprache des reinen Denkens"*. Traducción al inglés en van Heijenoort, J. ed. [1967]. pp.5-82.

[28] Gandy, R. O. [1956]. *"On the axiom of extensionality"*. **The Journal of Symbolic Logic.** vol. 21. pp. 36-48.

[29] Gandy, R. O. [1977]. *"The simple theory of types"*. en [32]

[30] Gandy, R. [1988] . "The Confluence of Ideas in 1936". (en Herken [40])

[31] Gabbay, D y Guenthner, F. eds. [2001]. *Handbook of Philosophical Logic*. vol.I-IV. Kluwer Academic Press. Amsterdam. Holanda.

[32] Gandy, R.O. and Hyland eds. [1977]. *Logic Colloquium 76.* Amsterdam: North Holland.

[33] Gandy, R. (1994): "The Confluence of Ideas in 1936", en The Universal Turing Machine. A Half-Century Survey. Springer. Viena, Nueva York, 1994.

[34] van Heijenoort, J. [1967]. *From Frege to Gödel: a source book in mathematical logic,* 1879-1931. Cambridge, Mass: Harvard University Press.

[35] Henkin, L. [1949]. *"The completeness of the first order functional calculus".* **The Journal of Symbolic Logic.** vol. 14, pp. 159-166.

[36] Henkin, L. [1950]. *"Completeness in the theory of types".***The Journal of Symbolic Logic.** vol. 15. pp. 81-91.

[37] Henkin, L. [1963]. *"A theory of propositional types".* **Fundamenta mathematicae**. vol. 52. pp. 323-344.

[38] Henkin, L. [1975]. *"Identity as a logical primitive".* **Philosophia**. vol. 5. pp. 31-45.

[39] Henkin, L. [1996]. "*The discovery of my completeness proofs"*, Dedicated to my teacher, Alonzo Church, in his 91^{st} year, **Bulletin of Symbolic Logic**, vol. 2, Number 2, June 1996. (presentado el 24 de Agosto de 1993 en el XIX International Congress of History of Science, Zaragoza, Spain).

[40] Herken, R. ed. [1988]. *The Universal Turing Machine: A Half-Century Survey.* Oxford: Oxford University Press.

[41] Hilbert, D. and Ackermann, W. [1928]. *Grundzege der theoretischen Logik.* Traducido en: Hilbert & Ackermann [1938]. *Principles of Mathematical Logic.* New York: Chelsea.

[42] Hindley, R. y Seldin, J. [1986]. *Introduction to combinators and* $\lambda-$*.calculus.* Cambridge: Cambridge University Press.

[43] Hindley, R. [1997]. *Basic Simple Type Theory.* Cambridge Tracts in Theoretical Computer Science. Cambridge: Cambridge University Press.

[44] Hodges, A. [1983]. *Alan Turing: The Enigma.* Publicado por Burnett Books Ltd, la edición que yo uso es de 1992, de Vintage. Londres.—

[45] Horn, B. K P. y Winston, P. H. [1981]. **LISP.** Reading: Addison-Wesley Publishing Company.

[46] Kleene, S. [1935]. *"A theory of positive integers in formal logic".* **American journal of mathematics.** vol. 57. pp. 153-173, 210-244.

[47] Kleene, S. [1936a]. *"General recursive functions on natural numbers".* **Mathematische Annalen**, vol. 112 (1936), pp. 727-742.(en [22])

[48] Kleene, S. [1936b]. *"*$\lambda-$*definability and recursiveness".* **Duke Mathematical Journal.** vol. 2. pp. 340-353.

[49] Kleene, S. [1936c]. *"A note on recursive functions"*. **Bulletin of the American Mathematical Society.** vol. 42. pp. 544-546.

[50] Kleene, S. [1967]. *Mathematical Logic.* Wiley.

[51] Kleene, S. [1981]. "Origins of recursive function theory". **Annals of History of Computing.** vol. 3. pp. 52-65.

[52] Manzano, M. [1980]. *Teoría de Tipos.* Barcelona: Ediciones de la Universidad de Barcelona.

[53] Manzano, M. [1982]. *"Los sistemas generales"*. en **Estudios de Lógica y Filosofía de la Ciencia.** Serie: Manuales Universitarios. Salamanca: Ediciones Universidad de Salamanca.

[54] Manzano, M. [1996]. *Extensions of First Order Logic***.** Cambridge Tracts in Theoretical Computer Science. Cambridge: Cambridge University Press.

[55] Manzano, M. [1997]. *"Alonzo Church: Hil Life, His Work and Some of His Miracles"*. **History and Philosophy of Logic,** vol 18, pp 211-232.

[56] Manzano, M. [1999]. *Lambda "Lambda Abstractor was Philosopher´s Stone for Alonzo Church"*. **Proccedings of the 11**th International Congress of Logic, Methodology and Philosophy of Sciene. Cracoww, Poland, página 435.

[57] Manzano, M. 2000. *"Genéricos, Similares y Equivalentes de Identidad"*. en **Logic, Language and Information**. Angel Nepomuceno et alts eds. Kronos. ISBN: 84-85101-57-X

[58] *"La Identidad y la Igualdad en la Lógica de Orden Superior"*. en **El conocimeinto y el desarrollo en el siglo XXI.** León Olivé et alts eds.

[59] Priest, G. [1979]. *"The Logic of Paradox",* Journal of Philosophical Logic, vol.8, 1979, pp. 219-241.

[60] Quine, W. [1937]. *"Logic based on inclusion and abstraction"*. **The Journal of Symbolic Logic.** vol. 2 pp. 145-152.

[61] - [1976]. *The ways of paradoxes and other essays.* Cambridge. Massachussets: Harvard University Press.

[62] Ramsey, F. [1926]. *The foundations of mathematics.* Proc. London Math Soc. Ser 2. 338-384.

[63] Russell, B. [1908]. *"Mathematical Logic as based in the theory of types"*. in van Heijenoort, J. ed. [1967]

[64] Russell, B. y Whitehead, A. [1910-13]. *Principia Mathematica.* vol. 1-3. Cambridge: Cambridge University Press.

[65] Sainsbury, R. [1988]. *Paradoxes.* Cambridge: Cambridge University Press.

[66] Shoenfield, J. R. [1995]. *"The mathematical work of S. C. Kleene".* **Bulletin of Symbolic Logic**. vol. 1. Núm 1. pp 9- 43.

[67] Tarski, A. [1923] *"Sur le terme primitif de la logistique".* **Fundamenta Mathematicae.** Vol. IV. pp. 196.

[68] Tarski, A. [1956]. *"The concept of truth in formalized languages".* en Tarski [1956].

[69] Tarski, A. [1956]. *Logic, semantics, metamathematics.* Artículos desde 1923 hasta 1938. Oxford: Clarendon Press.

[70] Turing, A. [1936]. *"On computable numbers with an application to the ensheidungsproblem".* **Proceedings of the London Mathematical Society**, vol. 42, 1936-37. pp. 230-265. también vol. 43, 1937, pp. 544-546.

[71] Turing, A. [1937]. *"Computability and $\lambda-$definability".* **The Journal of Symbolic Logic**, vol. 2, pp. 153-163.

Parte III

Tratamientos unificadores

Capítulo 12

EL DEBATE

La comparación entre lógicas requiere una cierta caracterización de las mismas y un posicionamiento respecto a la siguiente cuestión:

¿Es la lógica clásica un sistema lógico universal, adecuado para ser utilizado en todas las áreas en las que se aplica la lógica?

Los argumentos a favor y en contra del uso de la clásica frente a otras lógicas no clásicas son frecuentes en la historia de esta disciplina; e incluso el trazar fronteras y desterrar a una gran parte de los sistemas que extienden o se oponen a la de primer orden porque no son *lógica* fue durante años cosa corriente. La diversidad de lógicas existentes y la flexibilidad y amplitud de miras de los teóricos de la lógica actuales hacen que hoy no se conciba una postura semejante. Es interesante a este respecto la lectura del libro que en 1994 editó Gabbay [15], en el que destacados especialistas se enfrentan a *la pregunta*.

12.1. ¿Qué es un sistema lógico?

Obsérvese que ya no nos interrogamos *'¿qué es "la" lógica?'* sino que utilizamos el artículo indefinido.

¿Debemos empezar por definirlo?

El concepto que se tome debería considerar, al menos en principio, la multiplicidad de lógicas que se usan en aplicaciones en *I.A* y en software, así como las empleadas en filosofía, matemáticas y lingüística.

Como todos sabemos, el estudio de la lógica se remonta a los filósofos griegos[1] —aunque no se denominó así hasta cinco siglos más tarde— que la concebían como el campo de investigación en el que se proporcionan los patrones o estrategias —*tópoi*— que pueden desarrollarse en el curso de una buena argumentación competitiva. Puesto que la mejor y más honesta manera de triunfar en una controversia es proporcionar una argumentación correcta, el interés práctico conduce al teórico, investigándose la inferencia válida. En los *Tópicos,* que se ocupan del razonamiento dialéctico, se recogen consideraciones no sólo lógicas,

[1] Aristóteles [322 a.C]. *Organon.* (Recopilación de sus enseñanzas, hecha a su muerte)

sino también psicológicas y lingüísticas. Este razonamiento se caracteriza por el uso de *entimemas* —esto es, silogismos cuyas premisas son sólo probables— a diferencia del razonamiento *apodíctico* que toma premisas verdaderas y ciertas. En las *Categorías* se tipifican los predicados y los términos, creando así una ontología que se debate constantemente entre *las palabras y las cosas.*

En la página 4 expusimos nuestro punto de vista según el cual la lógica es el soporte de la creación, distribución y asimilación de la información y en la página 10 indicamos que es el fundamento teórico de la informática. En el comentario 1 ya avanzamos que durante el siglo XX la lógica se ha extendido y diversificado, abarcando casi todos lo contextos gobernados por reglas, y de ello hemos tenido constancia en los capítulos precedentes.

12.1.1. Lógica dinámica

Hay un interés creciente por la estructura lógica de las acciones cognitivas que subyacen al razonamiento humano: La lógica tradicional se ocupaba del resultado de dichas acciones; esto es, de las proposiciones, los argumentos y las pruebas. Ahora también se estudian los mecanismos de la trasmisión de la información. A este respecto el libro de Johan van Benthem [8] que investiga la lógica de la transmisión de la información resulta especialmente fecundo.

En los años setenta se creó la lógica dinámica de programas [14]. Vimos en el capítulo 9, dedicado a ella[2], que extendía a la modal siendo no sólo multimodal, sino que permitía que las modalidades se combinasen para formar nuevas. Se inspiraba en ella haciendo que cada programa se comportase como una relación de accesibilidad sobre un universo de estados —que aquí representan a los estados de un ordenador; esto es, a sus registros de memoria—; la intuición es que un programa es un objeto dinámico capaz de transportar al ordenador de un estado a otro, de manera tal que refleje la forma en que como resultado de la ejecución de un programa cambia el valor de verdad de los enunciados a ellos asociados. Mediante composición, unión o clausura reflexiva y transitiva se generan programas nuevos. Vimos en la sección 8.4.1 que los axiomas de cada lógica modal tratan de captar las propiedades de la relación de accesibilidad que les es propia —esto es, reflexiva en T, transitiva en 4, etc.— mientras que los de la lógica de programas expresan las propiedades de las operaciones entre programas.

Inspirada en la lógica dinámica de programas, nace la nueva lógica dinámica; aquí el paso de un estado a otro se produce mediante obtención de información. Los estados cognitivos están ordenados mediante inclusión $\subseteq$ de contenido informativo y entre las operaciones fundamentales se encuentran

1. *actualización de datos: up*(P)

$$\lambda xy\,(x \subseteq y \wedge Py)$$

[2] La mejor presentación es la de Harel [22] y mi libro favorito el de Goldblatt [20].

2. *actualización minimal de datos:* $\mu - up(P)$

$$\lambda xy\,(x \subseteq y \wedge Py \wedge \neg\exists z(x \subset z \subset y \wedge Pz))$$

3. *descargar datos:* $down(P)$

$$\lambda xy\,(x \supseteq y \wedge Py)$$

4. *posibilidad:* $poss(P)$

$$\lambda x\,(\exists y(x \subseteq y \wedge Py))$$

¿Hemos de incluir a esta nueva lógica dinámica?

¿No tendrá una arquitectura excesivamente compleja, difícil de vertebrar mediante una definición?

¿No corremos el riesgo de tener que dar una respuesta tan abstracta a la pregunta sobre la naturaleza de los sistemas lógicos que deje de ser operativa?

12.1.2. Lógica y teoría de juegos

Pero además, la argumentación puede entenderse como un *juego* cuyas reglas determinan la comunicación civilizada entre los individuos y la toma de decisiones. Los juegos formaron parte de la lógica en la antigüedad. La idea intuitiva —presente en la *Retórica* de Aristóteles— es que la validez recoge las estrategias ganadoras en el debate. En lógica se dio más importancia a la teoría de la prueba y a la consecuencia semántica, siendo justamente en este último contexto donde se utilizaron juegos para construir y comparar modelos [23]. Sin embargo en los últimos años han aparecido estudios sobre juegos lógicos [13], relacionándolos con la teoría de juegos de la que se sirve la economía, y se está desarrollando una teoría general de la teoría de juegos [3].

¿Incluimos a la lógica de la teoría de juegos aquí?

12.1.3. Lógica con diagramas

Centrándonos en el razonamiento clásico de las matemáticas, habida cuenta de lo útiles que resultan los diagramas para la resolución de problemas varios, podemos investigar las reglas del razonamiento visual.

¿Hay una lógica visual, diagramática?

Ya Euler, Venn y Peirce advirtieron la importancia que tienen los diagramas en las pruebas matemáticas. Su valor didáctico no se puso nunca en duda: los profesores de lógica somos conscientes de lo útiles que son, por ejemplo, los diagramas de Venn en teoría de conjuntos y en lógica de predicados monarios; o de la diferencia en cuanto al aprendizaje se refiere entre los cálculos de árboles y los lineales. Algunos lógicos y matemáticos —entre ellos, Barwise y Etchemendy— han estado trabajando en un modelo de inferencia que sea capaz de incorporar

[3] Me consta que Johan van Benthem está escribiendo un libro sobre este tema.

una explicación del razonamiento válido a partir de representaciones no lingüísticas., creando así una lógica heterogénea en la cual se incluyen sistemas de representación diversos [7]. En muchos casos los razonamientos diagramáticos no actúan como meras comparsas pues sustituyen con ventaja a los lingüísticos, probándose para ellos teoremas de completud y corrección. Ello requiere que su presentación como cálculo deductivo sea matemáticamente impecable, con el rigor, precisión y falta de ambigüedad que exigimos a cualquier cálculo lógico. Es interesante a este respecto el libro que en 1996 escribieron Barwise y Allwein [6].

¿Hemos de incluir el razonamiento heterogéneo?, ¿El exclusivamente diagramático?

12.1.4. Programa de investigación

Si nos limitamos al estudio de los sistemas lógicos existentes, nuestra respuesta sobre la naturaleza de la lógica se torna contingente, haciéndose dependiente de los avatares históricos que produjeron como fruto los sistemas lógicos que tenemos hoy. Y puesto que la creación de otros nuevos no cesa, debemos realizar un corte temporal y nuestra respuesta será perecedera; pero ya sabemos que el dilema *diacronía/sincronía* se plantea en todos los ámbitos y que el historicismo no fue nunca popular entre los lógicos. Una posibilidad es la siguiente: En vez de estudiar los sistemas lógicos existentes para determinar donde reside la *"logicidad"*, podemos estudiarlos procurando desentrañar qué se intentaba resolver con ellos[4]; confiando en que se halle en las preguntas planteadas, que no en las respuestas. Si se encontrase una *invariante lógica*, común a los casos en estudio, tal vez pudiéramos avanzar una respuesta: proponer unos *universales lógicos*. La solución obtenida por esta vía tendría la ventaja de abarcar más posibilidades y de señalar los huecos existentes; esto es, las necesidades que pudieran ser cubiertas mediante nuevos sistemas lógicos. Lo que estoy proponiendo es embarcarse en un programa de investigación que incluye el razonamiento diagramático así como otros posibles razonamientos heterogéneos y de la acción[5].

Puesto que el proyecto de investigación está aún produciendo resultados nuevos, en lo que sigue dejamos en suspense este trabajo que incluye una definición conceptual de la lógica y pasamos a estudiar la variedad de lo que hay.

12.2. La divergencia entre sistemas lógicos

Por lo que respecta a la comparación de la lógica clásica con otras, lo que hago a continuación es plantearlo en función de los sistemas lógicos existentes,

[4] Esta es la solución propuesta en [6]

[5] Este proyecto está funcionando ya, *Summa Logicae en el siglo XXI*, véase a este respecto su biblioteca digital en

http://logicae.usal.es

tomando como referencia la que es hoy comúnmente aceptada [1], [16] y [17], de los que en los capítulos precedentes ofrecimos una muestra significativa.

12.2.1. Definición de sistema lógico

La génesis de una lógica puede ser ésta: del universo matemático $\mathcal{V}$ se seleccionan ciertas estructuras matemáticas o modelos $\mathcal{A}$ de una cierta clase $\mathfrak{M}$, los que queremos estudiar. Para ello se introduce un lenguaje adecuado L y se define el concepto de verdad de una fórmula φ en un modelo, $\mathcal{A} \Vdash \varphi$. Con cada uno se asocia de forma natural el conjunto de todas las fórmulas verdaderas en él $Th(\mathcal{A})$, que tomadas en su conjunto describen con precisión a $\mathcal{A}$; son su exhaustiva caracterización con los recursos que brinda L. Todas las teorías de la clase de estructuras en estudio tienen un núcleo común, el de las fórmulas válidas en todas ellas —VAL— y que, consecuentemente, no describen a ninguna estructura en particular, describen a la lógica misma, *son* la lógica en estudio. Por consiguiente, podemos decir que la lógica es este conjunto *VAL,* que no es otra cosa que la gran intersección de todas las fórmulas en $Th(\mathcal{A})$, para cada $\mathcal{A} \in \mathfrak{M}$.

$$\bigcap \{Th(\mathcal{A}) \mid \mathcal{A} \in \mathfrak{M}\}$$

Frecuentemente, junto a la noción de verdad en un modelo se introduce el de consecuencia semántica de una fórmula a partir de un conjunto de ellas: usamos $\Gamma \vDash_{\mathfrak{M}} \varphi$ para indicar que todos los modelos $\mathcal{A} \in \mathfrak{M}$ de Γ lo son de φ. Esto es,

$$\forall \mathcal{A}(\mathcal{A} \in \mathfrak{M} \Rightarrow (\forall \gamma(\gamma \in \Gamma \Rightarrow \mathcal{A} \Vdash \gamma) \Rightarrow \mathcal{A} \Vdash \varphi))$$

El concepto semántico de consecuencia puede ser poco operativo, por lo que el paso siguiente suele ser el de intentar reemplazarlo por una réplica sintáctica, computable; esto es, mediante las reglas de un cálculo deductivo diseñadas para generar VAL. Decimos que φ es un teorema lógico —y escribimos $\vdash \varphi$— cuando dicha fórmula se ha conseguido siguiendo las reglas del cálculo. Por supuesto, el conjunto TEO de todos los teoremas lógicos y el conjunto VAL de las fórmulas válidas deben coincidir para que el objetivo de equivalencia entre sintaxis y semántica se haya cubierto, como hemos visto con detalle para varios sistemas lógicos.

Se ha descrito una lógica con una fuerte componente semántica, al menos en su génesis. Sin embargo, hay casos en los que el procedimiento se invierte y lo que tenemos en primera instancia es un cálculo deductivo: el concepto de consecuencia es sintáctico y la semántica se añade, si acaso, al final. Para permitir un acercamiento neutro al tema, en donde no primen los aspectos sintácticos sobre los semánticos —ni a la inversa— se puede definir un sistema lógico como un conjunto de fórmulas cerrado bajo consecuencia y fijar con precisión este concepto.

Como punto de partida Gabbay [15] propone trabajar con una definición provisional, para la que es preciso concretar:

1. Un lenguaje formal L incluyendo un alfabeto y una definición de fórmula

2. La relación de consecuencia se establece entre conjuntos finitos de fórmulas, exigiéndole que satisfaga los requisitos de reflexividad, monotonía y corte. Puesto que la relación de consecuencia puede alcanzarse tanto sintácticamente como semánticamente —esto es, como deduciblidad $\vdash$ y como $\models$ — en una definición no comprometida con ninguno de estos planteamientos se habla de consecuencia como relación binaria entre conjuntos de fórmulas $\Delta \mid\sim \Gamma$ [6].

3. Dicha relación puede definirse de varias maneras: se pueden listar los pares $\langle \Delta, \Gamma \rangle$ para los que $\Delta \mid\sim \Gamma$ se verifica, o incluso se pueden proporcionar los pares de conjuntos de fórmulas $\langle \Delta, \Gamma \rangle$ a un programa de un ordenador y esperar la respuesta.

4. Una semántica para L es una interpretación de las fórmulas de L en una familia de estructuras matemáticas que nos permita entender la relación de consecuencia $\mid\sim$ haciéndola depender de los modelos en los que que sean verdaderas. Se puede definir un *modelo* como una función que a cada fórmula le asigna un valor en el conjunto $\{F, V\}$. Es decir,

$$Modelo : FORM(L) \longrightarrow \{F, V\}$$

Una *semántica*, S, es un conjunto de modelos, junto a la definición de $\Delta \mid\sim \Gamma$ mediante la condición siguiente

$$\forall s \in S\, [\forall X \in \Delta(s(X) = V) \rightarrow \exists Y \in \Gamma(s(Y) = V)]$$

5. Hay casos en los que para generar la relación $\mid\sim$ contamos con procedimientos algorítmicos, que pueden ser procedimientos de decisión o demostradores automáticos de teoremas.

6. El concepto de sistema o cálculo deductivo no está bien delimitado; tenemos ejemplos interesantes en sistemas de Gentzen, de Hilbert, tableaux, etc[7]. Como primera aproximación podemos definir un sistema deductivo como un procedimiento algorítmico para generar $\mid\sim$ que usa reglas de la forma

$$\frac{\Delta_1 \mid\sim \Gamma_1; ...;\ \Delta_n \mid\sim \Gamma_n}{\Delta \mid\sim \Gamma}$$

y axiomas de la forma

$$\frac{\varnothing}{\Delta \mid\sim \Gamma}$$

[6] Tarski define la consecuencia como una operación establecida sobre el conjunto de las partes, o conjunto potencia del de las fórmulas y la caracteriza mediante una serie de axiomas. Sintéticamente, un sistema lógico es definido como aquel conjunto de sentencias que es idéntico al conjunto de sus consecuencias.

Axioma 1. $card(S) \leq \aleph_0$
Axioma 2. Si $X \subseteq S$, entonces $X \subseteq Cn(X)$
Axioma 3. Si $X \subseteq S$, entonces $Cn(Cn(X)) = Cn(X)$
Axioma 4. Si $X \subseteq S$, entonces $Cn(X) = \sum Cn(Y)_{Y \subseteq X}$ para $card(X) \prec \aleph_0$
Axioma 5. Existe una sentencia $x \in S$ tal que $Cn(\{x\}) = S$

[7] Como vimos en el capítulo 4.

Los axiomas son los elementos iniciales del conjunto de pares $\langle \Delta, \Gamma \rangle$ que constituyen la relación de consecuencia, $\mid\!\sim$. Un cálculo deductivo $S_{\mid\sim}$ no es otra cosa que un modo de generar $\mid\!\sim$.

12.2.2. Propiedades matemáticas de $\mid\!\sim$

Qué propiedades matemáticas poseen dichas lógicas?

La relación clásica de consecuencia y la de los sistemas que la extienden cumple:

1. $\Delta \mid\!\sim A$, si $A \in \Delta$ *(reflexividad)*
2. $\Delta \mid\!\sim A \Longrightarrow \Delta, \Delta\prime \mid\!\sim A$ *(monotonía)*
3. $\Delta \mid\!\sim B$; $\Delta, B \mid\!\sim A \Longrightarrow \Delta \mid\!\sim A$ *(corte)*

Por su parte, las lógicas no monotónicas violan esa ley y las subestructurales pueden carecer de la de corte. Además, muchas tienen una versión más estricta de reflexividad, $A \mid\!\sim A$. Estas propiedades constituyen un filtro matemático adecuado para juzgar y comparar lógicas[8]. Se tratará de una comparación de naturaleza estructural, pero interesante.

12.2.3. Equilibrio entre propiedades

Deberíamos comparar el uso de la lógica clásica con la no-clásica en sus aplicaciones a la *I.A* y a otras áreas. Puesto que cada lógica intenta modelizar un determinado ámbito, cabe preguntarse hasta qué punto lo ha conseguido. Destacan por una lado, la *naturalidad* del lenguaje, su adecuación como vehículo de expresión de las características de la parcela de realidad que está siendo estudiada, hasta qué punto los razonamientos se clarifican al ser formalizados. Por otro lado está la *eficiencia* computacional, si tenemos buenos cálculos, incluso si tenemos probadores automáticos que los implementen, automaticen y suplan. Lo he repetido frecuentemente, una lógica es como una balanza: en un platillo se coloca el cálculo deductivo y se evalúan las *características computacionales*; en el otro, su *capacidad expresiva*.

¿Dónde está el fiel de la lógica que estudiamos?

Comentario 368 *Deberíamos comparar el uso de la lógica clásica con la no-clásica en sus aplicaciones a la* I.A *y a otras áreas. Sabemos que la lógica de primer orden es la más equilibrada de una gran familia de lógicas, que en la proposicional prima lo computacional y en la de orden superior la capacidad expresiva.*

[8] Hay quienes han querido situar las barricadas para defender las fronteras de la lógica en el cumplimiento de las tres propiedades, considerando que se podría admitir que hay otras relaciones interesantes entre proposiciones, pero que *no son lógica*.

12.2.4. Recursos matemáticos utilizados

Al definir un sistema lógico particular necesitamos utilizar una serie de recursos matemáticos, tanto al definir su cálculo deductivo, como al trazar su semántica. *¿Qué supuestos conceptuales hemos tenido que asumir?* Hay diferencias entre el uso de la lógica clásica y el de las no-clásicas. Si la primera pretende ser un sistema universal, ello debería notarse a este nivel: se tendría que asumir lo mínimo. Pondré un ejemplo: en la lógica de segundo orden estándar se puede expresar la hipótesis generalizada del continuo y el axioma de elección; ello nos obliga a decidir en qué entorno matemático nos movemos, o lo que es lo mismo, qué extensión de ZF estamos usando como metalenguaje. De manera que en esta lógica se asume la teoría de conjuntos a un nivel no intuitivo ni trivial [9].

12.2.5. Traducción

Para poder relacionar a las diferentes lógicas, lo natural es construir puentes entre ellas; es decir, establecer correspondencias, traducciones, morfismos. Ciertamente toda deducción lógica puede ser simulada mediante una máquina de Turing.

¿Es suficiente?

El concepto de traducción debe también ser clarificado. Mediante gödelización todas las lógicas pueden serlo, usando como lenguaje de llegada el de la aritmética; otra posibilidad es emplear teoría de conjuntos no sólo como semántica subyacente, sino como lenguaje de llegada; también sabemos que todo conjunto recursivamente enumerable puede generarse mediante cláusulas de Horn.

¿Nos basta con esta traducción?

Lo ideal es que junto a la traducción de las fórmulas se produzca una conversión bidireccional de las estructuras de la lógica en estudio a las de la lógica usada como referencia y que haya una correspondencia entre todos los componentes de los sistemas lógicos en estudio. Como veremos con detalle en el próximo capítulo lo que propongo es convertir las estructuras en multivariadas, añadiendo universos en donde estén explícitamente representados todas las categorías de objetos —conjuntos, relaciones, funciones, etc— definibles en el lenguaje y tener el lenguaje y cálculo multivariado para realizar los cómputos.

12.2.6. Marco unificador

La elección de la lógica clásica como lógica unificadora está directamente relacionada con el éxito obtenido en la traducción de otras lógicas a ella; es decir, en su capacidad de servir como marco común[10].

¿Qué es un marco unificador?

[9] En la sección 10.5 utilizamos esta peculiaridad de la lógica de segundo orden estándar para demostrar su incompletud.

[10] Véase a este respecto [9], la teoría de la traducción propuesta en este trabajo es la que propongo en [29]

Se podría usar la lógica clásica simplemente como metalenguaje formalizado e incluso implementar en el propio lenguaje objeto las características del metalenguaje. *¿Es esto deseable?*

Hemos usado el término de *"lógica clásica"*, *¿incluye a la heterogénea o multivariada?, ¿incluye a la de orden superior?*

Mi objetivo es mostrar que la lógica clásica, si como tal incluimos a la multivariada o heterogénea, puede usarse como sistema universal, especialmente desde el punto de vista de la teoría de la prueba. Sin embargo, puesto que hay otros planteamientos globalizadores, expondremos dos de ellos para apreciar la conmensurabilidad de paradigmas; concretamente, las lógicas generales de Meseguer —en el capítulo 15— y los sistemas deductivos etiquetados de Gabbay —en el capítulo 16—.

No obstante, la existencia de un marco general en el que podamos representar los casos particulares no significa, necesariamente, que el mejor modo de tratar dichos casos sea inmersos en el general; simplemente da idea de la universalidad del tratamiento.

Comentario 369 *Pretendemos una unificación del* "lenguaje máquina" *y realizar en él todos los cómputos que en otro caso se harían en una multiplicidad de sistemas, y arrastrar las metapropiedades de la lógica multivariada a los sistemas lógicos en estudio. Sin embargo, mantenemos los diversos lenguajes como* "lenguaje de usuario" *por ser más naturales y adaptarse mejor a sus respectivos ámbitos de estudio.*

Bibliografía

[1] Abramsky, S, Gabbay, D. y Maibaum, T. [1992-2000]. *Handbook of Logic in Computer Science.* vol 1 a 6. OUP. Oxford. U.K.

[2] Anderson, A. R. y Belnap, N.D. [1975]. *Entailment: the Logic of Relevance and Necessity.* Vol.I. Princeton University Press.

[3] Anderson, A. R. y Belnap, N.D. [1992]. *Entailment: the Logic of Relevance and Necessity.* Vol.II. Princeton University Press.

[4] Aristóteles. *Organon.* Edición, traducción y notas de Miguel Candel Sanmartín. Biblioteca Clásica Gredos. Madrid 1982, 1988.

[5] Aristóteles. *Categorías. De Interpretatione.* Edición de Alfonso García Suárez, Luis. M. Valdés Villanueva, y Julian Velarde Lombraña. Tecnos. Madrid, 1999.

[6] Allwein, G.y Barwise, J. [1996]. *Logical Reasoning with Diagrams.* Oxford University Press. New York. USA.

[7] Barwise, J y Etchemendy, J. [1994]. *Hyperproof.* CSLI. Stanford. USA.

[8] J. van Benthem. [1996]. *Exploring Logical Dynamics.* CSLI. Stanford. USA.

[9] van Benthem, Johan. [2001]. *Correspondence theory.* (en [17])

[10] Boole, G. [1847]. *The Matematical Analysis of Logic.* Londres, Cambridge. Ed. castellana El Análisis Matemático de la Lógica, Cátedra. Madrid, 1979.

[11] Boolos, G. [1979]. *The Unprovability of Consistency.* Cambridge University Press. Cambridge.

[12] Dalen, D. van [2001]. *"Intuitionistic Logic",* en [17]. Vol. III: Alternatives to Classical Logic.

[13] Dismarch, H. [2000]. *Knowledge games.* ILLC Dissertation Series. Groningen. Holanda.

[14] Fischer, M y Ladner, R. [1979]. *"Propositional modal logic of programs".* Proc. 9^{th} ACM Symp. Theory of Comput. 18. 194-211

[15] D. Gabbay [1994] *What is a Logical System?* Oxford University Press. Oxford U.K.

[16] Gabbay, D. Hogger, G. y Robinson, J. [1993-1998]. *Handbook of Logic in Artificial Intelligence and Logic Programming.* vol 1 a 6. OUP.

[17] Gabbay, D y Guenthner, F. [2001]. *Handbook of Philosophical Logic* 2^{nd} edition. Kluwer Academic Publishers. Dordrecht. Holanda.

[18] Gardies, J.L. [1979]. *Lógica del Tiempo.* Ed. castellana de J. Ordóñez y P. Martínez-Freire. Paraninfo. Madrid, 1979.

[19] Gerbrandy, J. y otros eds. *JFAK: Essays dedicated to Johan van Benthem on the occasion of his* 50^{th} *birthday.* Amsterdam University Press. Amsterdam. Holanda.

[20] Goldblatt, R. [1992]. *Logics of Time and Computation*, Center for the Study of Language and Information. Stanford. USA

[21] Haack, S. [1974]. *Lógica Divergente.* Paraninfo. Madrid, 1979.

[22] Harel, D. [2001]. *"Dynamic Logic"* en [17].

[23] Hodges, W. [1984]. *Building models by games.* Cambridge University Press. Cambridge. U.K.

[24] Hughes, G.E. y Cresswell, M.J. [1968]. *Introducción a la lógica modal.* Ed. castellana de Esperanza Guisan Seijas. Tecnos. Madrid, 1973.

[25] Hughes, G.E. y Cresswell, M.J. [1984]. *A Companion to Modal Logic.* Methuen. Londres.

[26] Huertas, A y Manzano, M. [1999]. *"A fashionable partial and heterogeneous mirror for modality"* en [19].

[27] Langholm, T. [1988]. *Partiality, truth and persistence.* CSLI. Stanford.

[28] Levesque, H. y Pirri, F. [1999]. *Logical Foundations for Cognitive Agents.* Springer. Berlin.

[29] Manzano, M. [1996]. *Extensions of first order logic*. Number 19 in Cambridge Tracts in Theoretical Computer Science. Cambridge University Press. Cambridge. U.K.

[30] Manzano, M. [1998]. *"The little mermaid"* en [31].

[31] Martínez, C y otros eds. [1998]. *Truth in Perspective.* Ashgate. Aldershot. U.K.

[32] J. Meseguer. *General logics.* In H.-D. E. et al., editor, Logic Colloquium'87, pages 275–329. North-Holland, 1989.

[33] Priest, G., Routley, R. y Norman, J. [1986]. *Paraconsistents Logics*, Philosophia Verlag.

[34] Trillas, E. [1980]. *Conjuntos Borrosos.* Vicens Universidad. Barcelona.

[35] Trillas, E., Alsina, C. y Terricabras J.M. [1995]. *Introducción a la Lógica Borrosa.* Ariel Matemática. Barcelona.

[36] Urquhart, A. [2001]. *"Many-valued Logics"*, en [17].

Capítulo 13

Lógica Multivariada es Unificadora

13.1. Introducción

Este capítulo será una sucinta exposición de la segunda parte de mi libro [19]. En *Extensions of first order logic* se estudian varias extensiones de la lógica de primer orden que tienen aplicación en filosofía, informática, matemáticas, lingüística e inteligencia artificial. En él se introducen con cierto detalle los siguientes sistemas lógicos (ver figura 13.1): lógica de segundo orden (*SOL*), tres lenguajes para la teoría de tipos, lógicas modales proposicionales (*PML*) y lógica modal de primer orden-*S5*, lógica dinámica proposicional (*PDL*) y lógica multivariada o heterogénea (*MSL*).

Una de las características del libro es que en él se administra una sustanciosa dosis de lo que Johan van Benthem denominaría *perspectiva lógica* (ver figura 13.2).

El primer objetivo es, sencillamente, el de presentar al lector una colección de lógicas —es relativamente detallada en el caso de la lógica de orden superior y de la multivariada, y como meros ejemplos de aplicación de la maquinaria de traducción propuesta para las modales y dinámica— ; el segundo objetivo es desarrollar la tesis de que la mayoría de los sistemas *"razonables"* de lógica se traducen de forma natural a la lógica multivariada. Esta tesis se mantiene a lo largo de todo él, pero se desarrolla abierta y explícitamente en el último capítulo, en el que todas las lógicas consideradas se ponen en correspondencia directa con la lógica *MSL*.

En el presente libro he decidido reducir la exposición al caso general y a un sólo ejemplo, el de la lógica modal proposicional.

Me cito a mí misma[1]:

[1] Lo digo en el prefacio de [19].

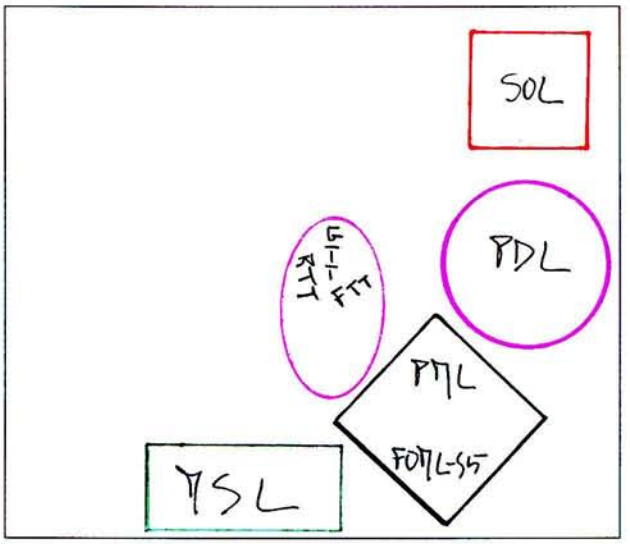


Figura 13.1: *Lógicas estudiadas*

The book is also connected to my own intellectual and personal biography, not only in the obvious sense, in terms of the time it took me to write it, but because the subject has been around me (back and forth) for many years. The subject of my Master's thesis was completeness for second order logic and my PhD thesis was on second order logic as well. Both were presented in the Department of Logic in the University of Barcelona. It was decisive for this book that I spent the academic year 1977-1978 as a Fulbright Scholar in Berkeley, and that Leon Henkin guided me as my advisor. There I learnt many of the things that directly or indirectly, I hope, will show in these pages, including an intellectual appreciation of the beauty of teaching and the value of effort put into pedagogical issues. I have always thanked Leon Henkin for introducing me, with his enormous gifts for teaching, in a non-traumatic style, to his own wise overview of metamathematics and algebra. The subject presented was the usual one in graduate courses, but the presentation and insights were a challenge. The daily handwritten handouts in the unmistakable violet color of the Vietnamese copier... unforgettable!

When I first learnt about modal logic, the idea of translating it into first order many-sorted logic appeared immediately. Right from the start for me it was directly connected with what had been done in higher order logic. I have always had the feeling that this was just putting another piece of the puzzle[2] in the right place. I soon

[2] Como muestra la figura 13.3.

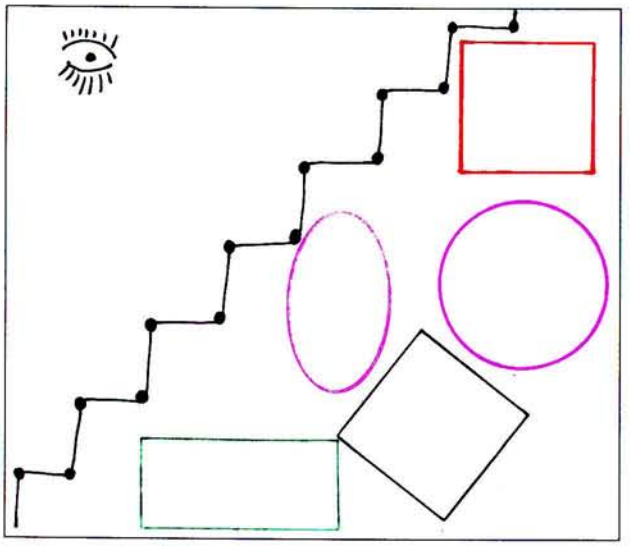


Figura 13.2: *Perspectiva lógica*

discovered that this idea had already generated a whole industry and I became very happy afterwards when reading Johan van Benthem's survey and book. Wonderful, hats off! I applied the same treatment to dynamic logic and wrote a paper on this. I was in Leeds at that time, at the Centre for Theoretical Computer Science. Somehow this book grew from that paper and that visit.

13.2. ¿Por qué la lógica multivariada?

¿Por qué MSL?, se podría preguntar.

Vimos en el capítulo 7 que *MSL* es una lógica muy flexible, resultando ser la elección obvia cuando se pretende tratar con más de un tipo de objetos. Pero además de ser la lógica *natural* para estos casos, es también una lógica *eficaz* ya que su teoría de la prueba está bien desarrollada —posee un cálculo no sólo correcto, sino también completo. Debido a su versatilidad, *MSL* puede ser usada como *marco unificador* en el que situar a otras lógicas.

Como ya comenté anteriormente, la proliferación actual de lógicas usadas en filosofía, informática, lingüística, matemáticas e inteligencia artificial hacen urgente y muy necesaria la reducción operativa de todas ellas. El objetivo es triple:

1. Usar en todos los casos un único cálculo deductivo y, a ser posible, un único demostrador de teoremas

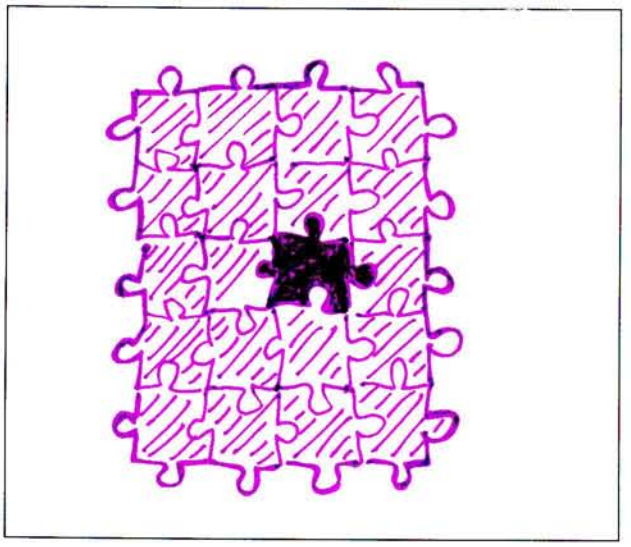


Figura 13.3: *La pieza del puzzle*

2. Arrastrar algunas de las metapropiedades desde MSL hasta otras lógicas en estudio

3. Comparar entre sí a las diferentes lógicas, mediante la comparación entre las teorías multivariadas que las representen.

Desde mi punto de vista, este planteamiento es tan obvio e intuitivo que sólo precisa de un poco de sentido común para entender la construcción. Además, si se modifican los ingredientes básicos, la receta puede adaptarse para cocinar diferentes platos[3].

Es difícil rastrear el origen y desarrollo de este enfoque, ya que cada lógica no clásica contó con su *"doble"* clásica casi desde su nacimiento. Sin embargo, debo atribuir la mayoría de las ideas que yo uso en este planteamiento a dos artículos de Henkin que menciono en las referencias: *"Completeness in the theory of types"*, de 1950, y *"Banishing the rule of substitution for functional variables"*, de 1953.

Sin embargo, no quisiera inducir a error. No hay en su mencionado artículo de 1950 traducciones de fórmulas, ni tan siquiera aparece explícitamente el lenguaje y el cálculo multivariado. En conexión con la lógica de orden superior, este cálculo se usa, que yo sepa, por primera vez en el artículo de 1953. En él Henkin propone el axioma de definición de clases y relaciones (*comprehensión*) como alternativa a la regla de sustitución usada en los cálculos anteriormente

[3] Así lo planteamos en [17].

propuestos para la lógica superior. Si del nuevo se elimina el axioma de comprehensión, obtenemos uno multivariado; es decir, el cálculo *MSL*. Otra de las ventajas del que Henkin propone en 1953 es que permite aislar a otros cálculos situados entre el de *MSL* y los de orden superior. El procedimiento es sencillo: debilitando comprehensión. En el último capítulo de mi libro *Extensions of First-order Logic* esta idea se usa tanto en el estudio de la lógica dinámica como en el de la lógica modal.

El Teorema de Completud en la Lógica de Orden Superior

En la sección 10.6 vimos que las ventajas de la nueva semántica no-estándar son evidentes ya que:

1. Conseguimos así un *teorema de completud*
2. Sirve para destacar no sólo a la lógica *MSL* sino también a *otras lógicas* entre *MSL* y *SOL*, obtenidas *debilitando comprehensión*.

Debilitar el axioma de comprehensión

Aquí radica el programa de investigación que quiero comentar a continuación, pero permitidme que os recuerde donde se sitúan esas lógicas intermedias[4].

Esta otra idea, ahora tomada del artículo de 1953, también resulta interesante: Si debilitamos el axioma de comprehensión —por ejemplo, lo restringimos a fórmulas de primer orden, o a traducciones de fórmulas dinámicas o modales, o a cualquier otro conjunto recursivo— obtenemos cálculos entre *MSL* y *SOL*. Y resulta fácil encontrarles su semántica correspondiente. Naturalmente, la clase de estructuras que les corresponde estará situada entre $\mathfrak{F}$ y $\mathfrak{G}.\mathfrak{S}$. La nueva lógica, llamémosla *XL*, será también completa. El motivo es que esta clase de modelos es axiomatizable. Esta idea se explotará tanto en *PML* como en *PDL*, aunque con distinto alcance.

13.3. Reducción de otras lógicas a la multivariada

Lo que haré ahora es explicar brevemente la idea general del programa de reducción. Dicho programa se desarrolla en tres niveles, o etapas (ver figura 13.4); en el primero nos detenemos en la mera representación de la verdad en la lógica en estudio como verdad de su traducción en estructuras multivariadas, módulo una cierta teoría cuya definición constituye una parte importante del programa de reducción; en el segundo la equivalencia se extiende al concepto de consecuencia a partir de conjuntos cualesquiera de hipótesis; en el tercero se demuestra la equivalencia del cálculo de la lógica en estudio con la teoría multivariada que la simula.

[4] Consultad la figura de la página 278.

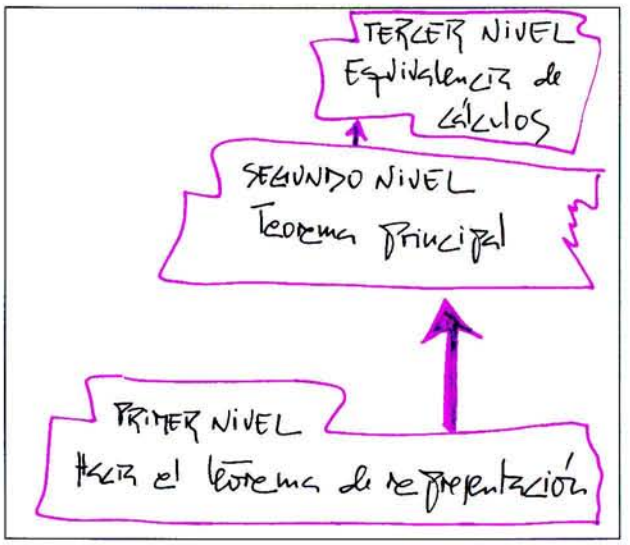


Figura 13.4: *Niveles del proceso de reducción*

13.3.1. Primer nivel: Teoremas de representación

El plan general es como sigue:

La signatura de la lógica en estudio —llamémosla *XL*— se transforma en una signatura heterogénea, las expresiones de la lógica *XL* se traducen a *MSL* y las estructuras propias de *XL* se convierten en estructuras heterogéneas. Por consiguiente, necesitamos definir una función recursiva *TRANS* para la traducción y una conversión directa de estructuras, $CONV_1$.

Una posibilidad es añadir a las estructuras heterogéneas universos que contengan todas las categorías de objetos matemáticos de los que queramos —y podamos— hablar en la lógica *XL*. Por consiguiente, en $CONV_1(\mathcal{A})$ añadiremos universos que contengan los conjuntos y relaciones definibles en las estructuras originales, $\mathcal{A}$, mediante los constructos de *XL*. A consecuencia de este cambio, parece que estemos tomando estructuras propias para *SOL* en vez de para *MSL*. Pero nosotros ya sabemos —lo aprendimos en la prueba de completud de Henkin— cómo evitar el uso explícito de *SOL*, cambiando a *MSL*: nuestros universos serán multivariados o heterogéneos, pero añadiremos relaciones de pertenecia y el axioma de extensionalidad.

El primer objetivo es definir $CONV_1$ y *TRANS* de forma que podamos demostrar

Lema 370 $\mathcal{A} \Vdash \varphi \Longleftrightarrow \mathrm{CONV}_1(\mathcal{A}) \Vdash \overline{\forall} TRANS(\varphi)$

Con la conversión directa de estructuras queremos obtener como resultado la equivalencia entre la validez de una fórmula cualquiera φ en las estructuras

originales de la lógica en estudio —llamémoslas simplemente $\mathfrak{S}\mathfrak{T}(XL)$—, y la validez de la clausura universal de su traducción a lógica multivariada

$$\overline{\forall} TRANS(\varphi)$$

en la clase $\mathfrak{S}^*$ de las estructuras obtenidas por conversión

$$\text{donde } \mathfrak{S}^* = CONV_1(\mathfrak{S}\mathfrak{T}(XL))$$

Esto es,

Teorema 371 $\models_{\mathfrak{S}\mathfrak{T}(\mathrm{XL})} \varphi \Longleftrightarrow \models_{\mathfrak{S}^*} \overline{\forall} TRANS(\varphi)$

Naturalmente, la primera pregunta que uno debe hacerse es si $\mathfrak{S}^*$ puede reemplazarse por la clase de modelos de un conjunto de fórmulas heterogéneas a determinar, Δ. Por consiguiente, la clave para las definiciones de *TRANS* y de $CONV_1$, es el poder simplificar la demostración de la equivalencia semántica, y en este sentido la relevancia de los resultados que pudieran derivarse depende de si se puede o no axiomatizar $\mathfrak{S}^*$. En algunos casos, bastará con axiomatizar una clase algo más amplia, que incluya a $\mathfrak{S}^*$.

Resumiendo, nuestro primer objetivo es demostrar un teorema de representación; es decir, un teorema de esta forma

Teorema 372 *(De representación)* $\models_{\mathfrak{S}\mathfrak{T}(XL)} \varphi \Longleftrightarrow \Delta \models_{\mathfrak{S}\mathfrak{T}(MSL)} \overline{\forall} TRANS(\varphi)$.

A partir de este teorema se demuestra fácilmente el *teorema de enumerabilidad para la lógica XL*. Por lo tanto, sabemos que podremos diseñar un cálculo para *XL*, pero también sabemos que en *MSL* podemos simular ese cálculo e incluso, si lo hubiera, *usar el demostrador de teoremas de MSL*.

También se demuestra

Teorema 373 $TRANS(\Pi) \cup \Delta \models_{\mathfrak{S}\mathfrak{T}(MSL)} TRANS(\varphi) \Longrightarrow \Pi \models_{\mathfrak{S}\mathfrak{T}(XL)} \varphi$

Así que, aunque la reducción se hubiera detenido en este nivel, los resultados desde un punto de vista práctico serían impresionantes (ver figura 13.5).

13.3.2. Segundo nivel: El Teorema Principal

Una vez demostrado el teorema de representación la pregunta inmediata es,

¿Se puede hacer algo más?

¿Es el teorema de representación el objetivo máximo?

Cuando la lógica en estudio *XL* posee un concepto de consecuencia, podemos intentar demostrar lo que yo llamo el *teorema principal*; esto es, que consecuencia en *XL* equivale a consecuencia de las traducciones módulo la teoría Δ. Concretamente,

Teorema 374 *(Principal)* $\Pi \models_{\mathfrak{S}\mathfrak{T}(XL)} \varphi \Longleftrightarrow TRANS(\Pi) \cup \Delta \models_{\mathfrak{S}\mathfrak{T}(MSL)} TRANS(\varphi)$

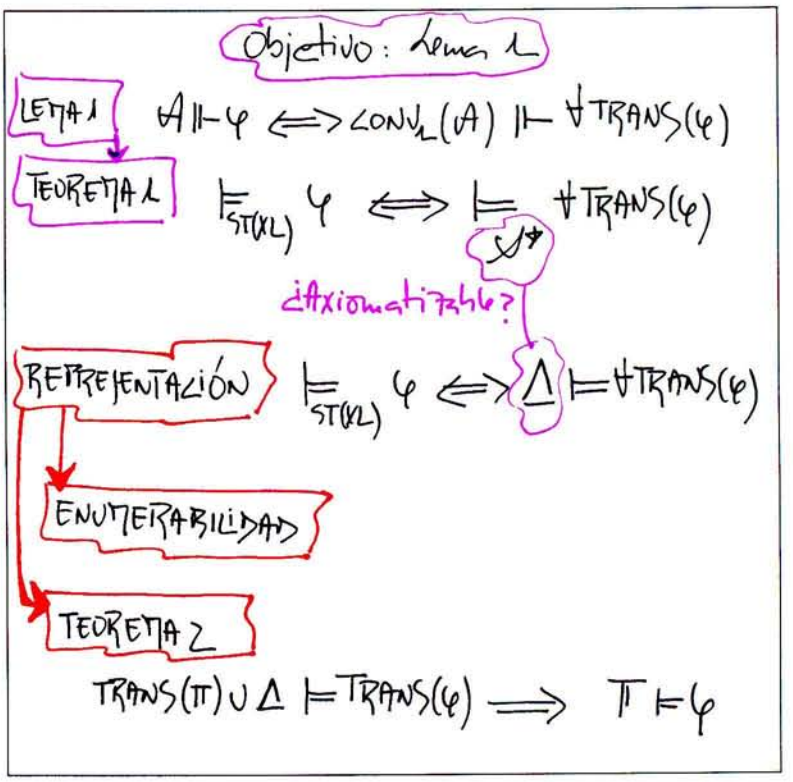


Figura 13.5: *Primer nivel*

A partir del teorema principal se obtienen *compacidad y Löwenheim-Skolem para XL*, usando las correspondientes propiedades de *MSL*. Para demostrar el teorema principal hace falta definir una conversión inversa de estructuras; nuestro objetivo al definirlas es demostrar que si partimos de un modelo de Δ, digamos, $\mathcal{B}$, y de una fórmula modal φ, $\mathcal{B}$ es modelo de la clausura universal de la traducción de φ, $\overline{\forall}TRANS(\varphi)$, si y sólo si la conversión inversa de $\mathcal{B}$ lo es de φ. Esto es,

Lema 375 $\mathrm{CONV}_2(\mathcal{B}) \Vdash \varphi \Longleftrightarrow \mathcal{B} \Vdash \overline{\forall}TRANS(\varphi)$

Resumiendo, éste sería el resultado (ver figura 13.6)

13.3.3. Tercer nivel: Equivalencia de los cálculos

Cuando la lógica en estudio XL posee también un cálculo deductivo, podemos intentar utilizar la maquinaria de la correspondencia ya desarrollada para demostrar, si es posible, corrección y completud para XL.

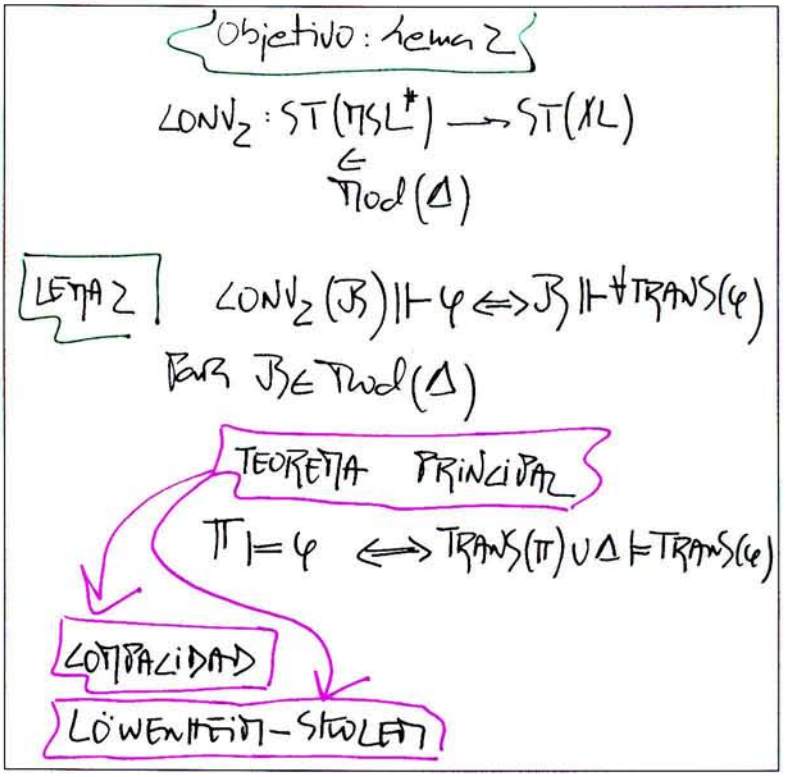


Figura 13.6: *Segundo nivel*

Puesto que contamos ya con el teorema principal y con completud y corrección para *MSL*, nos gustaría completar el cuadro demostrando la doble flecha en la línea de abajo

$$\begin{array}{c} TRANS(\Pi) \cup \Delta \models_{\mathfrak{ST}(MSL)} TRANS(\varphi) \Longleftrightarrow \Pi \models_{\mathfrak{ST}(XL)} \varphi \\ \Updownarrow \\ TRANS(\Pi) \cup \Delta \vdash TRANS(\varphi) \Longleftrightarrow \Pi \vdash \varphi \end{array}$$

La flecha hacia la izquierda es normalmente de demostración muy sencilla ya que podemos añadir a Δ las condiciones requeridas, como axiomas. En algunos casos resulta útil poder usar la construcción del modelo canónico $\mathcal{A}_{CAN}$ para poder demostrar

Lema 376 $\mathrm{CONV}_1(\mathcal{A}_{CAN}) \Vdash \overline{\forall} TRANS(\varphi) \Longrightarrow \vdash_{XL} \varphi$

13.4. Lógicas modales *K* y *S4*

En esta sección lo que haremos es analizar el caso de la lógica modal y muy especialmente la minimal, K y la lógica *S4*. La traducción de fórmulas será la estándar, la conversión de estructuras consiste en añadir un universo en donde situar los conjuntos de mundos significativos y definiremos una teoría heterogénea que representa y equivale a la modal, tanto en el caso de K como de *S4*.

13.4.1. El teorema de representación en PML

En este caso se reducirá *PML* a $MSL^{\Box}$. Este lenguaje multivariado contiene los símbolos siguientes: $\bot, \neg, \rightarrow$, los relatores monarios $P_0, P_1, P_2, ...$ el relator binario S, un signo de pertenecia y dos de igualdad; para individuos y relaciones. Hay variables de dos tipos, para individuos y para conjuntos. Usaremos la conocida traducción estándar, que expresa en primer orden las condiciones de verdad de las fórmulas modales. En particular,

- $TRANS^{\Box}(\bot)[u] = u \neq u$
- $TRANS^{\Box}(p)[u] = Pu$
- $TRANS^{\Box}(\neg\varphi)[u] = \neg TRANS^{\Box}(\varphi)[u]$
- $TRANS^{\Box}(\varphi \rightarrow \psi)[u] = (TRANS^{\Box}(\varphi)[u] \rightarrow TRANS^{\Box}(\psi)[u])$
- $TRANS^{\Box}(\Box\varphi)[u] = \forall v(Ruv \rightarrow TRANS^{\Box}(\varphi)[v])$

Lo que hacemos es formular en lógica clásica las condiciones semánticas de verdad de las fórmulas modales. Prescindimos de nuestras lógicas *de-moda-modales* y nos ponemos los anteojos clásicos (ver figura 13.7).

Es fácil ver que las traducciones de los axiomas K y $Df\Diamond$ indican propiedades obvias de la cuantificación y que por lo tanto estas fórmulas pueden demostrarse en *MSL* como teoremas lógicos.

Ejercicio 377 $\vdash_{MSL} \forall u TRANS^{\Box}(K)[u]$

Ejercicio 378 $\vdash_{MSL} \forall u TRANS^{\Box}(Df_{\Diamond})[u]$

Por otra parte, las traducciones tanto del axioma T como del axioma 4 no son fórmulas válidas en *MSL* pues necesitan que se verifiquen ciertas hipótesis.

Ejercicio 379 $(???) \vdash_{MSL} \forall u TRANS^{\Box}(T)[u]$

Ejercicio 380 $(???) \vdash_{MSL} \forall u TRANS^{\Box}(4)[u]$

¿Cuáles?, nos preguntamos.

Las estructuras multivariadas que usaremos se obtienen a partir de las estructuras modales al añadir universos que contengan conjuntos de estados, o

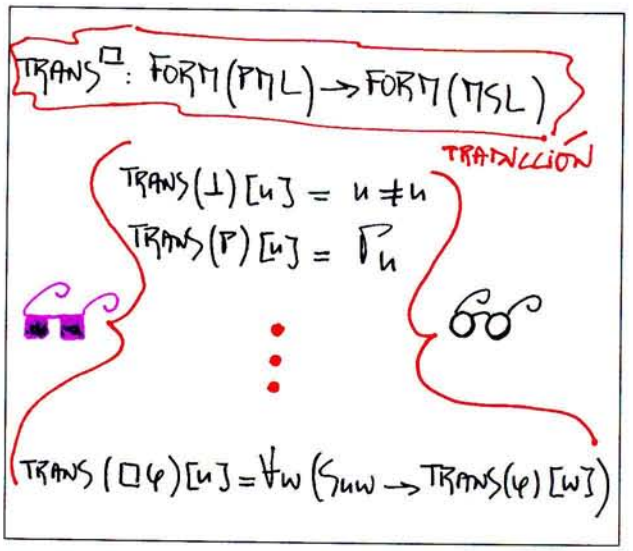


Figura 13.7: *Gafas*

mundos. La intuición aquí es que queremos tener explícitamente representados todos los conjuntos que pueden ser definidos modalmente —esto es, mediante fórmulas modales en estructuras de Kripke—.

Dada una estructura modal $\mathcal{A}$ construimos un marco al añadir un universo de subconjuntos de $\mathbf{W}$, $\mathbf{W}'$. Una clase muy especial de marcos son los que denomino *modelos generales construidos sobre estructuras modales.* Para construir un modelo general de esta clase lo que hacemos en incluir en $\mathbf{W}'$ el conjunto DEF de los subconjuntos de $\mathbf{W}$ definibles algebraicamente —esto es, $DEF \subseteq \mathbf{W}'$ y está cerrado bajo ciertas operaciones—; añadiremos los conjuntos unitarios y el de los puntos accesibles a partir de uno dado. DEF contiene el conjunto vacío $\emptyset$, todo $\mathbf{W}$ y las interpretaciones de las proposiciones atómicas. DEF está cerrado bajo operaciones booleanas; esto es,

$$\forall T(T, S \in DEF \Rightarrow T \cup S, \mathbf{W} - T \in DEF)$$

y bajo la siguiente operación

$$\forall T(T \in DEF \Rightarrow Dom(R \cap (\mathbf{W} \times T)) \in DEF)$$

Fuera de DEF, pero en $\mathbf{W}'$, tenemos los unitarios de todos los elementos de $\mathbf{W}$. La pertenencia a $\mathbf{W}'$ también obedece esta regla

$$\forall w(w \in \mathbf{W} \Rightarrow Rec(R \cap (\{w\} \times \mathbf{W})) \in \mathbf{W}')$$

Llamemos $\mathfrak{F}$ a la clase de todos los marcos construidos sobre estructuras modales y $\mathfrak{G}$ a la clase de todas las estructuras generales contruidas sobre estructuras modales.

Se puede demostrar que $\mathfrak{G}$ está contenido en la clase de marcos construidos sobre estructuras modales cuyo universo $\mathbf{W}'$ contiene todos los conjuntos definibles mediante fórmulas modales en sus propias estructuras modales. Así mismo, se puede demostrar que una fórmula modal φ define en una estructura modal $\mathcal{A}$ el mismo conjunto que su traducción define en cualquier estructura general construida sobre $\mathcal{A}$; esto es, en $\mathcal{A}\mathfrak{G}$. Es decir,

Lema 381 $\mathcal{A}\mathfrak{G}(\lambda u TRANS(\varphi)[u]) = \mathcal{A}(\varphi)$

Finalmente, demostramos que la validez de una fórmula φ en *PML* equivale a la validez de la clausura universal de su traducción en la clase $\mathfrak{G}$. Es decir,

Teorema 382 $\vDash \varphi$ *en* PML $\Longleftrightarrow \vDash_{\mathfrak{G}} \forall u TRANS(\varphi)[u]$

Ahora la pregunta es, *¿podemos axiomatizar* $\mathfrak{G}$*?*

La respuesta en este caso es afirmativa, ya que definimos la teoría *MODO* cuyos axiomas son los esquemas de comprehensión para traducciones de fórmulas modales, para fórmulas atómicas con igualdad y para átomos que usan el relator binario S —que representa a la relación de accesibilidad—. Esto es,

$$\forall \exists X \forall u(\varepsilon u X \leftrightarrow \varphi) \text{ donde } \varphi \in TRANS(PML) \cup I \cup \Sigma$$

I contiene las fórmulas de la forma $u = v$, y Σ contiene todas las fórmulas de la forma Suv.

Añadiremos extensionalidad, porque queremos simular la lógica de segundo orden en la multivariada.

Se demuestra lo siguiente:

Teorema 383 $Mod(\text{MODO}) = \mathfrak{G}$

Usando estos resultados obtenemos un *teorema de representación* para la lógica minimal, K.

Teorema 384 *(De representación, para* K*)*

$$\vDash \varphi \textit{ en } \text{PML} \Longleftrightarrow \text{MODO} \vDash_{\mathfrak{F}} \forall u TRANS(\varphi)[u] \textit{ en } \text{MSL}$$

A partir del teorema de representación se demuestra *enumerabilidad* para la lógica K de *PML*. Además, caso de existir un probador de teoremas para *MSL*, podría ser usado para demostrar teoremas modales.

Ya que tenemos un teorema de representación para K, buscamos uno para la lógica *S4*. Llamemos *MODO(S4)* a la teoría heterogénea obtenida al añadir a *MODO* las condiciones abstractas de segundo orden correspondientes a las fórmulas modales T y 4 —llamémoslas *MS(T)* y *MS(4)*—. Sea $\mathcal{D}$ la clase de las estructuras de Kripke reflexivas y transitivas. El teorema de representación de *S4* tiene esta forma

Teorema 385 *(De representación, para S4)* $\vDash_{\mathcal{D}} \varphi$ *en* PML $\Longleftrightarrow$MODO(S4)$\vDash_{\mathfrak{F}}$ $\forall u TRANS(\varphi)[u]$ *en* MSL

Este resultado es de fácil obtención porque en el cálculo multivariado se puede demostrar que las formulaciones habituales de reflexividad y transitividad equivalen a *MS(T)* y *MS(4)* módulo la teoría *MODO*.

Ejercicio 386 *(Resultado crucial) En el cálculo multivariado,*

$$\text{MODO} \vdash \text{MS(T)} \leftrightarrow \text{Reflexividad}$$
$$\text{MODO} \vdash \text{MS(4)} \leftrightarrow \text{Transitividad}$$

Comentario 387 *Esta situación es mejor que la que encontramos en PML con la semántica de modelos. Allí una estructura de Kripke reflexiva es un modelo de T, una transitiva lo es de 4, etc, pero hay modelos irreflexivos de T y modelos intransitivos de 4. Por lo tanto, si pensamos que los axiomas modales tratan de definir la relación de accesibilidad que les es propia, esto puede ser considerado como un cierto fracaso. Verdad es que se puede tomar la conocida como semántica de marcos para PML. La alternativa que yo propongo es que vayamos al entorno de las estructuras generales aquí definidas, puesto que nos ofrece una caracterización de las propiedades semánticas de la relación de accesibilidad sin perder el cálculo.*

13.4.2. El teorema principal en PML

Puesto que ya tenemos el teorema de representación y en lógica modal contamos con el concepto de consecuencia, intentamos conseguir el teorema principal. En *PML* la conversión inversa es la función inversa de $CONV_1$ (borramos el universo $\mathbf{W}'$) y así obtenemos fácilmente el *teorema principal* tanto para K como para S4. Esto es,

Teorema 388 *(Principal, K)* $\Pi \vDash \varphi \Longleftrightarrow TRANS(\Pi) \cup MODO \vDash TRANS(\varphi)$

Teorema 389 *(Principal, S4)* $\Pi \vDash_{\mathcal{D}} \varphi \Longleftrightarrow TRANS(\Pi) \cup MODO(S4) \vDash TRANS(\varphi)$

(siendo $\mathcal{D}$ la clase de estructuras de Kripke reflexivas y transitivas.

Como dije antes, así conseguimos gratis los teoremas de *compacidad* y de *Löwenheim-Skolem* para las lógicas *K* y *S4*. La demostración se basa en el resultado crucial antes mencionado en el ejercicio 386.

13.4.3. Los cálculos modales K y S4

Queremos demostrar que los cálculos modales para *K* y *S4* tienen un correlato en *MSL*. Es decir, que las teorías heterogéneas que las representan, *MODO* y *MODO(S4)*, son también equivalentes sintácticamente a *K* y *S4*. Es decir,

$$TRANS(\Pi) \cup MODO \vdash TRANS(\varphi) \Longleftrightarrow \Pi \vdash_K \varphi$$

y

$$TRANS(\Pi) \cup \mathit{MODO(S4)} \vdash TRANS(\varphi) \Longleftrightarrow \Pi \vdash_{S4} \varphi$$

Por lo que a la lógica K se refiere la flecha hacia la izquierda se obtiene fácilmente porque las traducciones tanto de K como de $Df\Diamond$ expresan propiedades obvias de la cuantificación. La regla de necesariedad se preserva bajo traducción. Por lo que respecta a la lógica *S4*, usando *MS(T)* y *MS(4)* y comprehensión obtenemos la traducción deseada de cada ocurrencia de T o de 4.

En lógica modal, dada una lógica cualquiera, digamos B, hay un modelo canónico, $\mathcal{B}_B$, cuyo universo W contiene todos los conjuntos $B-$máximamente consistentes y cuya relación de accesibilidad se define así

$$\{\langle x, y\rangle |\, \{\varphi | \Box\varphi \in x\} \subseteq y\}$$

A partir de este modelo, mediante conversión directa, obtenemos el modelo canónico general, $\mathcal{B}_B\mathfrak{G}$.

En el caso de la lógica K esta estructura no es tan sólo un modelo de *MODO*, sino que también podemos demostrar que la traducción de una fórmula modal φ es verdadera en un punto del modelo canónico si y sólo si la fórmula pertenece a dicho punto,

$$\mathcal{B}_K\mathfrak{G}[t] \vDash TRANS(\varphi)[u] \Longleftrightarrow \varphi \in t$$

Ahora podemos demostrar que

$$\mathcal{B}_K\mathfrak{G} \vDash \forall u TRANS(\varphi)[u] \Longrightarrow \vdash_K \varphi$$

Concluimos finalmente que

$$\mathit{MODO} \vdash_{MSL} \forall u TRANS(\varphi)[u] \Longleftrightarrow \vdash_K \varphi$$

Usando este resultado llegamos a la completud y corrección de la lógica minimal de *PML*, K. No hemos hecho una demostración directa, la hemos traído de *MSL*.

Para la lógica *S4* se obtiene un resultado similar.

$$\mathit{MODO(S4)} \vdash_{MSL} \forall u TRANS(\varphi)[u] \Longleftrightarrow \vdash_{S4} \varphi$$

Ahora, los teoremas de *completud* y *corrección* se demuestran con facilidad.

Comentario 390 *Si alguien me preguntara que qué hemos obtenido, la respuesta sería que ahora tenemos a una serie de lógicas preciosamente enmarcadas (ver figura: 13.8).*

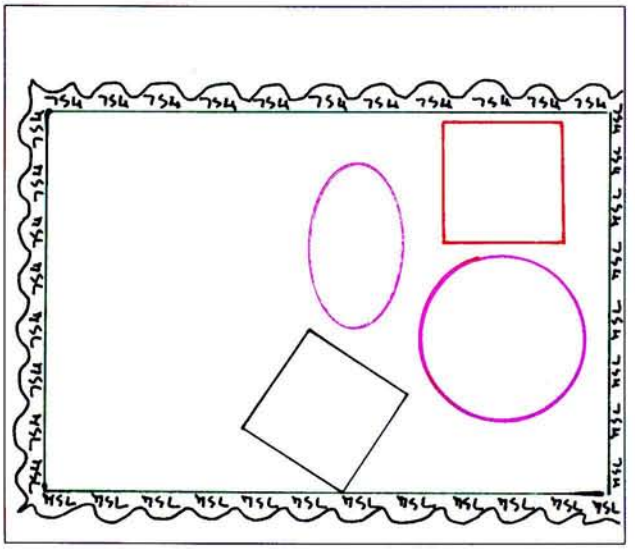


Figura 13.8: *Marco multivariado*

Bibliografía

[1] Abramsky, S, Gabbay, D. y Maibaum, T. [1992-2000]. *Handbook of Logic in Computer Science.* vol 1 a 6. OUP. Oxford. U.K.

[2] Allwein, G.y Barwise, J. [1996]. *Logical Reasoning with Diagrams.* Oxford University Press. New York. USA.

[3] Barwise, J y Etchemendy, J. [1994]. *Hyperproof.* CSLI. Stanford. USA.

[4] van Benthem, J. [1975]. *"A note on modal formulae and relational properties"*, **The Journal of Symbolic Logic**, vol. 40, n. 1, pp. 55-58.

[5] van Benthem, J. [1977]. *"Modal logic as second order logic"*, Mathematisch Institut, University of Amsterdam.

[6] van Benthem, J. [1983]. *Modal Logic and Classical Logic*, Naples, Bibliopolis.

[7] van Benthem, J. [2001]. *Correspondence theory.* (en [12])

[8] Clavel, M. y Manzano, M. [1997]. *"Mapas de Instituciones: de la Lógica Modal S4 a la lógica multivariada"* en

[9] Estany, A. y otros eds. [1997]. *Actas del II Congreso de la Sociedad de Lógica, Metodología y Filosofía de la Ciencia.* Servei de Publicaciones de la UAB. Barcelona. España.

[10] D. Gabbay [1994] *What is a Logical System?* Oxford University Press. Oxford U.K.

[11] Gabbay, D. Hogger, G. y Robinson, J. [1993-1998]. *Handbook of Logic in Artificial Intelligence and Logic Programming.* vol 1 a 6. OUP.

[12] Gabbay, D y Guenthner, F. [2001]. *Handbook of Philosophical Logic* 2^{nd} edition. Kluwer Academic Publishers. Dordrecht. Holanda.

[13] Gerbrandy, J. y otros eds. *JFAK: Essays dedicated to Johan van Benthem on the occasion of his* 50^{th} *birthday.* Amsterdam University Press. Amsterdam. Holanda.

[14] Henkin, L. [1950]. *"Completeness in the theory of types"*. **JSL** vol. 15, pp. 81-91.

[15] Henkin, L. [1953]. *"Banishing the rule of substitution for functional variables"*. **JSL** vol.18, num. 3 pp. 201-208.

[16] Huertas, A. [1994]. *Modal Logic and Non-Classical Logic,* tesis doctoral de la Universidad de Barcelona.

[17] Huertas, A y Manzano, M. [1995]. *"Partial and Heterogeneous Logic: Cooking up your Logic"* en 10^{th} International Congress of Logic, Methodology and Philosophy of Science. Florencia.

[18] Huertas, A y Manzano, M. [1999]. *"A fashionable partial and heterogeneous mirror for modality"* en [4].

[19] Manzano, M. [1996]. *Extensions of first order logic*. Number 19 in Cambridge Tracts in Theoretical Computer Science. Cambridge University Press. Cambridge. U.K.

Capítulo 14

Lógica Multivariada y Parcial

14.1. Introducción

14.1.1. Planteamiento general: *"Cooking up your logic"*

Una situación que se presenta con cierta frecuencia es la siguiente: tenemos un sistema lógico no totalmente desarrollado o aceptado —al que llamaremos aquí la *Prelógica XL*— que se ha definido de manera puramente sintáctica —esto es, como un conjunto de fórmulas— pero su semántica tal vez no esté fijada, o contamos tan sólo con una clase reducida de interpretaciones o modelos posibles. Se propone entonces la construcción de una lógica multivariada y posiblemente también parcial *PHL* como procedimiento para estudiar y mejorar la lógica *XL*. La nueva servirá de lógica subyacente sobre la que definiremos, mediante un proceso de traducción, la teoría que representa a la lógica *XL*.

La nueva lógica parcial y multivariada será lo suficientemente potente como para axiomatizar en ella las características semánticas de *XL*; contaremos además con un cálculo deductivo tipo Gentzen para demostrarlas.

Para construir la lógica marco *PHL* empezamos elaborando un cuestionario para *XL*, cuyos items dependen de la propia lógica en estudio, pero que están basados en estas dos preguntas:

1. Los modelos o interpretaciones estándar —o las simplemente buscadas, deseadas o pretendidas— *¿utilizan objetos o conceptos?* En el primer caso, *¿cuántos tipos o variedades de objetos tenemos?*

2. *¿Es un sistema bivariado o hay fórmulas que no reciben valor verdadero ni falso?*

El caso que desarrolla con detalle Antonia Huertas [9] —el de la lógica modal de primer orden— responde afirmativamente a ambas preguntas, así que es

preciso construir una lógica parcial y multivariada: hay que definir su semántica y su sintaxis de manera que las propiedades de *XL* que queremos estudiar y expresar se puedan formular explícitamente en *PHL.*

En primer lugar hay que encontrar un *conjunto de conectivas funcionalmente completo* para la nueva lógica multivariada y parcial, al que exigimos que sea además *significativo.* Esta búsqueda requiere un estudio detallado bidireccional en el que intervienen los factores siguientes: la naturaleza de la parcialidad que queremos caracterizar; la adecuación del concepto de consecuencia semántica de la nueva lógica *PHL* (local o global, fuerte o débil, que conserve reglas clásicas, etc.); y si queremos tener cuantificadores clásicos o nó. Otro asunto verdaderamente complicado cuando tratamos con valores de verdad no clásicos es el de la introducción de la igualdad.

En segundo lugar presentamos un *cálculo deductivo tipo Gentzen* para *PHL*, definiendo reglas de introducción y eliminación de conectivas y reglas para los cuantificadores.

Comentario 391 *Es preferible este tipo de cálculo porque, al tener la propiedad de la subfórmula, permite ir hacia atrás en las deducciones y acoplar el cálculo de* ***PHL*** *al de la nueva lógica que resultará al final de todo el proceso; esto es, la lógica* ***PXL*** *de la que hablaremos luego.*

Construido el marco *PHL,* inspirado en el cuestionario inicial, lo que hacemos es traducir la lógica en estudio, *XL*, a esta nueva lógica subyacente; para ello se traducen su signatura y sus fórmulas y se convierten las estructuras de la lógica *XL* en estructuras propias de *PHL.* Obtendremos así una teoría Δ escrita en lógica multivariada y parcial que representa —y rivaliza— con la original *XL.*

Estas son las tres primeras etapas del proceso (ver figura: 14.1)

La novedad del planteamiento es que ahora, mediante un proceso de retrotraducción vamos a definir una nueva lógica parcial *PXL* que será la que sustituya a la original *XL.* En este proceso de retrotraducción debemos separar claramente las características parciales de las multivariadas. El resultado final es una lógica segura, matemáticamente mejor presentada y definida, una alternativa clara a *XL* (ver figura:14.2).

Comentario 392 *Mediante este procedimiento en [10] puede comprobarse cómo la modalidad de primer orden y la nueva parcialidad se encuentran de manera natural en una lógica modal y parcial. En este caso no se buscaba una reducción a la clásica, sino una nueva presentación de la lógica modal de primer orden, un* espejo[1] *que nos devolviera una imagen mejorada. Caso de querer reducirla ahora, nos bastaría con usar los resultados de reducción de la lógica parcial.*

14.1.2. Modalidad y Parcialidad

Como el título sugiere este estudio se centra en la interrelación de las nociones de *parcialidad* y *modalidad.* La lógica parcial nació en los años veinte y fructificó

[1] Como explicamos con detalle en [10].

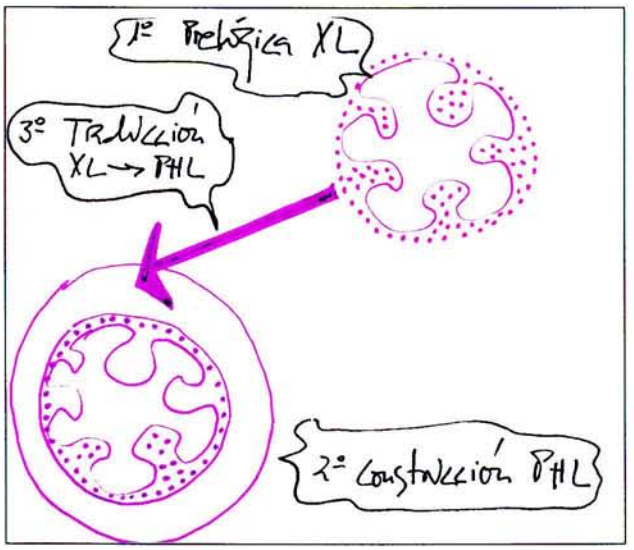


Figura 14.1: *Las tres primeras etapas*

entre los años treinta y finales de los cincuenta, en lo que se dio en llamar *lógicas multivaloradas* o *polivalentes*. Después cayeron en cierto olvido hasta que en los años noventa volvieron a despertar interés por su aplicación en la representación del conocimiento humano y en la inteligencia artificial. Por otro lado la lógica modal, que podemos considerar afianzada en los años cuarenta con la semántica de Kripke, ha seguido interesando hasta nuestros días, cabe destacar su aplicación a la representación del conocimiento. En la historia de la lógica modal, sin embargo, la versión proposicional y la de predicados han tenido comportamientos muy diferentes. En este trabajo se explica cómo la modalidad de primer orden y la nueva parcialidad se encuentran de manera natural en una lógica modal y parcial.

14.2. Lógica Modal de Predicados

Lenguaje Modal

Dos de los trabajos más importantes en lógica modal de predicados son Hughes & Cresswell [12] y van Benthem [1]. En ellos, y prácticamente en todos los artículos sobre el tema, la lógica modal de predicados se obtiene añadiendo el operador monádico $\Box$ (o L), con la interpretación modal usual de necesidad, al lenguaje estándar de lógica de predicados, exactamente de la misma forma que se añade al lenguaje estándar proposicional para obtener la lógica modal proposicional.

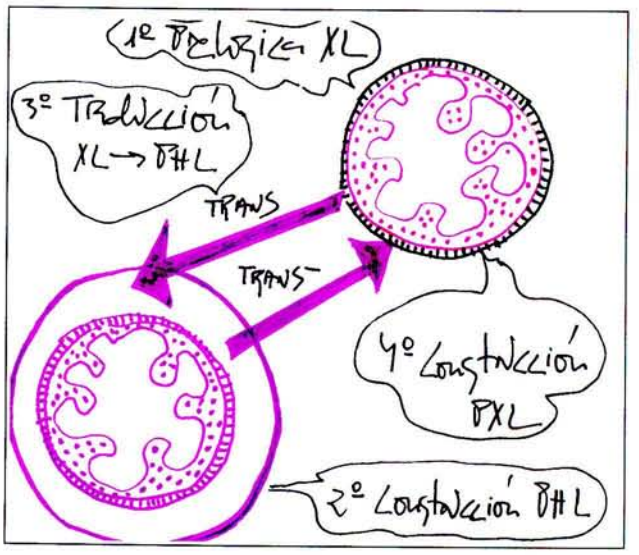


Figura 14.2: *Broche final*

Semántica Modal de Predicados

La historia de la lógica modal de predicados nos enseña que no se puede encontrar un sistema semántico satisfactorio, pero también nos dice que si no consideramos la igualdad lógica, la propuesta más exitosa para una lógica modal de predicados es el sistema de huecos de valor de verdad de Hughes & Cresswell, donde las fórmulas que hablan de individuos inexistentes en el dominio de un mundo posible carecen de valor de verdad clásico —verdadero o falso—; esto es, hay huecos de valor de verdad. Los dos problemas de este sistema son la imposibilidad de definir la identidad y la existencia de demasiadas fórmulas fuera del alcance de la semántica. El problema de la igualdad no tiene solución como los propios autores explican, pero, respecto al otro problema, hay una posibilidad de evitar las fórmulas *'indefinidas'* o los *'huecos de valor de verdad'* (ver figura: 14.3), aceptando que la ausencia de valor de verdad debe ser un valor de verdad él mismo —un tercer valor no clásico—. Para encontrar este nuevo valor debemos investigar la lógica parcial adecuada a los huecos de valor de verdad de Hughes & Cresswell.

Interpretación Modal

Definiremos las estructuras modales y las interpretaciones basadas en Hughes & Cresswell [12], $\mathbf{M}$ el conjunto de los *mundos*, y $\mathbf{R}$ la *relación de accesibilidad entre mundos*. Una *estructura modal* $\mathcal{A}$ contiene a $\mathbf{A}$ el *universo de individuos*, $\mathbf{A_m}$ el *dominio de individuos de* $\mathbf{m}$ para cada mundo $\mathbf{m} \in \mathbf{M}$ y la interpretación de los símbolos de operación. Una *asignación modal*

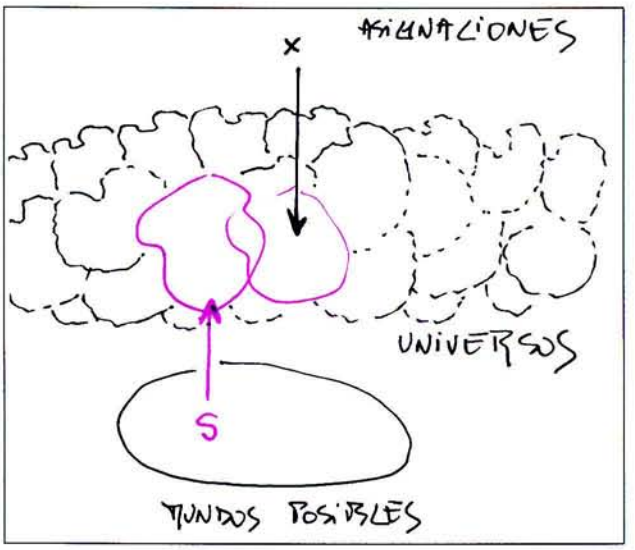


Figura 14.3: *Huecos de valor de verdad*

de variables sobre $\mathcal{A}$ es una función

$$H : \mathcal{V} \longrightarrow \mathbf{A}$$

Una *interpretación* sobre la estructura $\mathcal{A}$ *en el mundo* $\mathbf{m} \in \mathbf{M}$ es una tripla ordenada

$$\Im_{\mathbf{m}} = \langle \mathcal{A}, H, m \rangle$$

donde H es una asignación modal de variables. La *denotación* de una expresión ϵ con la interpretación $\Im_{\mathbf{m}}$; esto es, $\Im_{\mathbf{m}}(\epsilon)$, se define por recursión de la manera esperada, basándonos en Hughes & Cresswell.

Una *interpretación* sobre la estructura $\mathcal{A}$ es una familia de interpretaciones para cada mundo

$$\Im = \{\Im_{\mathbf{m}}\}_{\mathbf{m} \in \mathbf{M}}$$

El conjunto de *valores de verdad* es

$$\mathbf{A}_0 = \{T,\ F,\ N\}$$

donde T y F son los valores *verdadero* y *falso* de la lógica clásica y N es el valor *nulo* de nuestra lógica parcial que introduciremos en la siguiente sección.

14.3. Lógica Multivariada y Parcial

Conectivas y Cuantificación Parciales

El sistema de conectivas adecuado se basa en el sistema trivalorado de Bočvar's[2]. Este sistema no es funcionalmente completo y es necesario añadir las nuevas conectivas parciales V, *v*erificación y ξ *desfalsificación* —ver Huertas [9]. En esa misma referencia esta la cuantificación adecuada a nuestro sistema parcial.

Lenguaje Multivariado y Parcial

El lenguaje lo forman el alfabeto y las expresiones

$$L = ALF(L) \cup EXPR(L)$$

con

$$ALF(L) = OPER.SIM \cup \mathcal{V} \cup \{\forall\}$$

En particular, las conectivas las incluimos en este conjunto

$$\{\neg, \rightarrow, V, \xi, R, Q, =\} \subseteq OPER.SIM$$

El conjunto de *variables*

$$\mathcal{V} = \mathcal{V}_1 \cup \mathcal{V}_2$$

está partido, siendo $\mathcal{V}_i$ el de variables de variedad (o de género) i

Semántica Heterogénea Parcial

Una *estructura heterogénea parcial* $\mathcal{A}$ se define de manera que coincide con una estructura modal añadiendo la interpretación de los nuevos símbolos parciales. Las *asignaciones* de variables H son multivariadas; es decir, respetan las variedades.

Una *interpretación* es un par

$$\Im = \langle \mathcal{A}, H \rangle$$

La *denotación* de una expresión ϵ por $\Im$ es la adecuada a nuestra lógica parcial, teniendo en cuenta que en la interpretación del cuantificador debe tomarse el universo correspondiente a cada tipo de variable.

Definición 393 *Una interpretación* $\Im$ *es un* ***modelo fuerte*** *de una fórmula* φ *—y escribimos* $\Im \Vdash_s \varphi$*— syss*

$$\Im(\varphi) = T$$

$\Im$ *es un* ***modelo débil*** *de* φ *—y escribimos* $\Im \Vdash_w \varphi$*— syss*

$$\Im(\varphi) \neq F$$

[2] Ver Haak [5]

Decimos que φ *es* x***-satisfacible*** $(x \in \{s, w\})$ *syss existe una interpretación* $\Im$ *que es un* x***-modelo*** *de* φ

Definición 394 *Una fórmula* φ *es* ***xy-consecuencia*** *lógica* $(x, y \in \{s, w\})$ *de un conjunto de fórmulas* Γ *—y escribimos* $\Gamma \models_{xy} \varphi$ *— syss toda interpretación* $\Im$ *que sea* x-modelo *de* Γ *es* y*-modelo de* φ*.*

A partir de la comparación de las cuatro consecuencias (ss, sw, ws, ww) se obtienen los mejores resultados para la *consecuencia fuerte-débil* $(\models_{sw})$ y la satisfacibilidad fuerte (*s-satisfiable*). Y por tanto son estas las que tomamos.

14.4. Diseño de la Lógica Modal Parcial

Traducción en PHL

La función de traducción $TRAN$ de cada objeto de la lógica Modal de Predicados en *PHL* es la siguiente:

1. Un lenguaje modal se traduce en uno *PHL* añadiendo los símbolos R, Q, V, ξ más un conjunto de variables de variedad 2 y eliminando el operador modal de necesidad.
2. En cuanto a las expresiones modales, los términos se traducen en ellos mismos como *PHL* términos y la traducción de las fórmulas se define por recursión.

Por su parte, una estructura modal se convierte en una *PHL* de manera obvia, añadiendo sólo la interpretación de las nuevas conectivas V y ξ.

Se puede definir un conjunto de fórmulas de *PHL,* Π tal que si $\mathcal{B}$ es un estructura multivariada y parcial se cumpla:,

$$\text{si } \mathcal{B} \Vdash_{sw} \Pi \text{ entonces } TRANS^{-1}(\mathcal{B}) \text{ es una estructura modal}$$

Lógica Modal Parcial PML

Redefinimos la Lógica Modal de predicados como una lógica parcial y la llamaremos lógica *Modal Parcial PML*, de manera que podamos obtener el siguiente teorema de equivalencia :

$$\Gamma \models \varphi \text{ en } PML \text{ syss } \overline{\forall}\Gamma \cup \Pi \models_{sw} \forall u TRAN\varphi[u] \text{ en } PHL$$

La *L*ógica Modal Parcial *PML* se construye a partir de la lógica modal de predicados original como sigue:

1. Un lenguaje para PML se define añadiendo el conjunto $\{V, \xi, =\}$ a $ALF(L)$
2. La semántica de PML se define añadiendo las condiciones relativas a los nuevas conectivas parciales. Validez en PML es la fuerte y consecuencia en *PML* es la fuerte-débil.

Una síntesis del proceso, puede verse en la figura 14.4.

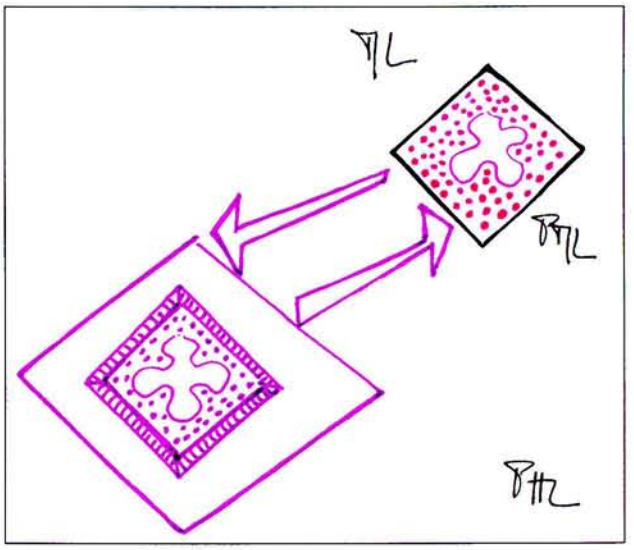


Figura 14.4: *Síntesis del proceso*

14.5. Implementaciones informáticas

Los siguientes proyectos de informática se hicieron bajo mi dirección, están disponibles en

$$http://aracne.usal.es$$

en donde se explican con detenimiento. En todos los casos se trata de traductores automáticos entre lenguajes formales.

1. Iván Marcos. *Traductor de Lógicas: Modal a Multivariada.* Marzo 1999
2. María Iglesias. *Traductor de Lógicas: Dinámica a Multivariada.* Febrero 2000
3. Raquel Caño. *Traductor de Lógicas: Modal de Primer Orden a Multivariada y Parcial.* Septiembre 2001
4. José Escuadra. *Traductor de Lógicas: Multivariada a Primer Orden sin variedades.* Junio 1999

Bibliografía

[1] van Benthem, J. [1983]. *Modal Logic and Classical Logic,* Bibliopolis, Napoli.

[2] van Benthem, J. [1986]. *"Partiality and nonmonotonicity in classical logic"*, Logique et Analyse 114, n^{o}29, pp.225-247.

[3] Blamey, S. [1986]. *"Partial Logic"*, en Handbook of Philosophical Logic. Vol. III, D. Gabbay y F. Guenthner (eds.). Reidel. Dordrecht, 1986.

[4] Gerbrandy, J. y otros eds. *JFAK: Essays dedicated to Johan van Benthem on the occasion of his 50th birthday.* Amsterdam University Press. Amsterdam. Holanda.

[5] Haak, S [1974]. *Deviant Logic,* Cambridge University Press. cambridge. U.K. Paraninfo. Madrid, 1979.

[6] Huertas, A. [1993]. *"El tercer valor de verdad en la lógica modal de predicados". Actas I Congreso de la Sociedad de Lógica, Metodología y Filosofía de la Ciencia.* Departamento de Reprografía de la UNED Madrid, pp. 80-83. Madrid.

[7] Huertas, A. [1997]. *"Parcialidad y Modalidad". Actas II Congreso de la Sociedad de Lógica, Metodología y Filosofía de la Ciencia en España.* Servei de Publicacions de la Universidad Autónoma de Barcelona, pp. 482-485. Barcelona.

[8] Huertas, A. *"Partial and Heterogeneous Logic for Modal Logic"*. En *Partial, Epistemic and Dynamic Logic* . Kluwer Academic Publishers.

[9] Huertas, A. [1994]. *Modal Logic and Non-Classical Logic,* tesis doctoral de la Universidad de Barcelona.

[10] Huertas, A y Manzano, M. [1999]. *"A fashionable partial and heterogeneous mirror for modality"* en [4].

[11] Huertas, A. y Manzano, M. [1991]. *"Many-sorted logic as a unifying framework". 9th International Congress of Logic, Metodology and Philosophy of Science.* (Logic Colloquium'91). Uppsala University. Uppsala. Resumen en: *Journal of Symbolic Logic.* Vol.58,n.2. p.755. USA, junio de 1993.

[12] Hughes, G. y Cresswell, M. [1968]. *An Introduction to Modal Logic,* Methuen, London. Ed. castellana de Esperanza Guisan Seijas. Tecnos. Madrid, 1973.

[13] Hughes, G.E. y Cresswell, M.J. [1984]. *A Companion to Modal Logic.* Methuen. Londres.

[14] Langholm, T. [1988]. *Partiality, truth and persistence.* CSLI. Stanford.

[15] Urquhart, A. [1986]. *"Many-valued Logics"*, en Handbook of Philosophical Logic. Vol. III, D. Gabbay y F. Guenthner (eds.). Reidel. Dordrecht, 1986.

Capítulo 15

Lógicas Generales

15.1. Introducción

La teoría de las lógicas generales [7] es un estudio axiomático de los diferentes elementos de una lógica y de las relaciones entre diferentes lógicas; en particular, su relación de consecuencia sintáctica —integrada en el denominado *sistema implicativo*—, su cálculo y su aparato semántico —este último, se estudia bajo la noción de *institución*.

La idea más interesante, extraída de la teoría de categorías, es que las propiedades más fecundas de un sistema lógico o de una estructura matemática no residen en ellos mismos, sino en su interrelación con sistemas o estructuras similares. Por ejemplo, se estudia mejor la teoría de grupos partiendo del hecho de que ellos y sus homomorfismos constituyen una *categoría*. Este punto de vista se adopta también en la teoría de las lógicas generales, en donde se considera que la consecuencia sintáctica, el cálculo deductivo y las denominadas *instituciones* son estructuras semejantes a los grupos del ejemplo y se establecen entre ellas correspondencias (*mapas*) capaces de preservar su estructura. Así que el punto fuerte de estas lógicas generales reside en definir adecuadamente las correspondencias entre ellas de tal manera que los *mapas* entre sistemas implicativos preserven la relación de consecuencia sintáctica, los establecidos entre los cálculos dejen fija lo anterior y las propias pruebas y los que se definen entre instituciones conserven la relación semántica de satisfacibilidad.

En consonancia con esta *filosofía*, una lógica marco $\mathfrak{F}$ será aquella capaz de representar a muchas otras y ello se concreta en el establecimiento de mapas de las lógicas en estudio a $\mathfrak{F}$.

En el capítulo 13 propuse una correspondencia entre lógicas basada en una traducción de fórmulas y una conversión de estructuras, pues al utilizar la lógica multivariada como marco en donde situar las lógicas en estudio se pueden economizar recursos; esto es, con frecuencia se pueden arrastrar resultados desde la multivariada a la lógica en estudio. Uno de los casos allí tratados es el de la lógica modal *S4*.

Así que tanto en la teoría de las lógicas generales, como en la teoría de la correspondencia —esto es, la corriente traduccionista a la que me ascribo— se persigue una reducción operativa. El trabajo que publiqué con Manuel Clavel [3] supone un primer esfuerzo hacia la construcción de un puente entre sendos planteamientos, que permita la comunicación de resultados ya establecidos en ambas direcciones. En la sección 15.2 se recogen las nociones relevantes para nuestro trabajo de la teoría de lógicas generales; en la 15.3 se define el aparato semántico de la lógica modal *S4* como una institución; por último, en el apartado 15.4 se define el *mapa* de instituciones entre *S4* y la lógica multivariada (o heterogénea), y proponemos una ampliación de las condiciones para establecer un *mapa* de instituciones tal que permita obtener un teorema de representación —lo hacemos en 15.4.1—.

15.2. Lógicas Generales

Estos son los componentes básicos de las lógicas generales:

Sintaxis

Para presentar un lenguaje se proporciona su signatura Σ y las reglas de formación de fórmulas. Para cada lógica hay una categoría *Sign* de sus posibles signaturas y un functor *sen* que a cada signatura Σ le asigna el conjunto de sus fórmulas, $sen(\Sigma)$

Sistemas implicativos

Dada una signatura Σ en *Sign* la derivabilidad de una sentencia $\alpha \in sen(\Sigma)$ a partir de un conjunto $\Gamma \subseteq sen(\Sigma)$ es una relación binaria que normalmente indicamos así:

$$\Gamma \vdash \alpha$$

y que se establece siempre que a partir de los axiomas de Γ se siga α usando las reglas de la lógica en estudio —se dice que Γ *implica* α—

Formalmente se define un sistema implicativo como una tripla ordenada

$$\mathcal{E} = \langle Sign, sen, \vdash \rangle$$

La relación $\vdash$ cumple reflexividad, monotonía y transitividad (corte) además de la propiedad de $\vdash$ −traducibilidad; esto es,

Si

$$\Gamma \vdash_{\Sigma} \alpha$$

entonces para cada $H : \Sigma \longrightarrow \Sigma'$ en *Sign* se cumple

$$sen(H)(\Gamma) \vdash_{\Sigma'} sen(H)(\alpha)$$

Es decir, la derivabilidad se conserva aunque cambie el lenguaje[1].

[1] A esto se le suele llamar principio de sustitución.

Instituciones

Una institución es una cuádrupla

$$\Im = \langle Sign,\ sen,\ Mod,\ \Vdash \rangle$$

tal que $Sign$ es una categoría cuyos objetos llamamos signaturas, sen es un functor que asocia a cada signatura un conjunto de sentencias, Mod es un functor que asocia a cada signatura una clase de modelos y $\Vdash$ es una función que asocia a cada signatura Σ una relación binaria que llamamos satisfacción.

La condición de satisfacción exige que para cada traducción sintáctica entre signaturas H un modelo de la segunda signatura satisface la traducción de una fórmula si y sólo si la traducción del modelo satisface la fórmula original. El functor Mod es contravariante, funciona hacia atrás.

Consecuencia

Sea Γ un conjunto de $\Sigma-$sentencias. Entonces, se define $Mod(\Sigma, \Gamma)$ como el subconjunto de $Mod(\Sigma)$ determinado por todos los modelos $\mathcal{M} \in Mod(\Sigma)$ tales que satisfacen todas las sentencias de Γ; esto es, $\mathcal{M} \Vdash_\Sigma \varphi$ para cada $\varphi \in \Gamma$. Podemos ahora introducir una relación entre conjuntos de sentencias y sentencias, que denotamos $\models$ del siguiente modo:

$$\Gamma \models_\Sigma \varphi \quad \text{syss} \quad \mathcal{M} \Vdash_\Sigma \varphi \ \text{ para cada } \ \mathcal{M} \in Mod(\Sigma, \Gamma)$$

Mapa de Instituciones

Sean las instituciones

$$\Im = \langle Sign, sen, Mod, \Vdash \rangle \quad \text{y} \quad \Im' = \langle Sign', sen', Mod', \Vdash' \rangle$$

Entonces, un mapa de instituciones (Φ, α, β) consta de una *transformación natural* α un functor $\alpha-sensible$ Φ entre las categorías de teorías de $\Im$ e $\Im'$ y una transformación natural β.

A continuación definimos con un poco más de detalle algunos de estos componentes básicos.

Definición 395 *Una* ***institución*** *—véase [4] y [5]— es una cuádrupla*

$$\Im = \langle Sign,\ sen,\ Mod,\ \Vdash \rangle$$

tal que $Sign$ *es una categoría cuyos objetos llamamos signaturas,*

$$sen : Sign \longrightarrow Set$$

es un functor que asocia a cada signatura un conjunto de sentencias,

$$Mod : Sign \longrightarrow Set^{op}$$

es un functor que asocia a cada signatura una clase de modelos, y $\Vdash$ *es una función que asocia a cada signatura* Σ *una relación binaria*

$$\Vdash_\Sigma \subseteq Mod(\Sigma) \times sen(\Sigma)$$

que llamamos satisfacción, tal que para todo

$$\mathcal{M}' \in Mod(\Sigma')$$

y para cada

$$H : \Sigma \longrightarrow \Sigma' \quad y \quad \varphi \in sen(\Sigma)$$

la siguiente propiedad, que llamamos ***Condición de Satisfacción****, se cumple:*

$$Mod(H)(\mathcal{M}') \Vdash_\Sigma \varphi \quad syss \quad \mathcal{M}'_{\Sigma'} \Vdash sen(H)(\varphi).$$

Definición 396 *Sea* Γ *un conjunto de* $\Sigma-$*sentencias. Entonces, definimos* $Mod(\Sigma, \Gamma)$ *como el subconjunto de* $Mod(\Sigma)$ *determinado por todos los modelos* $\mathcal{M} \in Mod(\Sigma)$ *tales que satisfacen todas las sentencias de* Γ*, esto es,* $\mathcal{M} \Vdash_\Sigma \varphi$ *para cada* $\varphi \in \Gamma$*. Podemos ahora definir una relación entre conjuntos de sentencias y sentencias —que denotamos* $\models$ *— del siguiente modo:*

$$\Gamma \models_\Sigma \varphi \quad syss \quad \mathcal{M} \Vdash_\Sigma \varphi \quad para\ cada \quad \mathcal{M} \in Mod(\Sigma, \Gamma)$$

Definición 397 *Sean las instituciones*

$$\Im = \langle Sign, sen, Mod, \Vdash \rangle \quad y \quad \Im' = \langle Sign', sen', Mod', \Vdash' \rangle$$

Entonces, un mapa de instituciones

$$(\Phi, \alpha, \beta) : \Im \longrightarrow \Im'$$

consta de una transformación natural

$$\alpha : sen \Longrightarrow sen' \circ \Phi$$

un functor $\alpha-$*sensible*

$$\Phi : Th \longrightarrow Th'$$

entre las categorías de teorías de $\Im$ *e* $\Im'$*, y una transformación natural*

$$\beta : Mod' \circ \Phi^{op} \Longrightarrow Mod$$

tal que para cada

$$\Sigma \in Sign, \varphi \in sen(\Sigma) \quad y \quad \mathcal{M}' \in Mod'(\Phi(\Sigma, \emptyset))$$

se cumple la siguiente propiedad:

$$\mathcal{M}'_{sign'(\Phi(\Sigma,\emptyset))} \Vdash \alpha(\varphi) \quad syss \quad \beta_{(\Sigma,\emptyset)}(\mathcal{M}') \Vdash_\Sigma \varphi$$

15.3. La Lógica Modal Proposicional S4

Dada la similitud de objetivos entre este planteamiento y el de mi libro [6], parecía natural establecer vínculos. En [3] se realiza un primer esfuerzo para construir un puente entre sendos planteamientos, que permita la comunicación de resultados ya establecidos en ambas orillas. Lo que hicimos en ese artículo fue centrarnos en una de las lógicas que habían sido satisfactoriamente traducidas a la multivariada, el sistema *S4* de Lewis.

En primer lugar, definimos el aparato semántico de la lógica modal *S4* como una institución

$$\langle Sign_{PM}, Mod_{PM}, sen_{PM}, \Vdash \rangle$$

A tal efecto se define la categoría de signaturas modales proposicionales, que denotamos $Sign_{PM}$ y que tiene signaturas modales proposicionales como objetos y morfismos de signaturas modales proposicionales como morfismos; la composición y la identidad en $Sign_{PM}$ se expresan como composición de funciones y la función identidad, respectivamente.

¿Cuales serán los modelos propios de esa lógica?

Un $\Sigma-$modelo de *S4* es básicamente un modelo de Kripke cuya relación de accesibilidad es reflexiva y transitiva.

Se requieren también los functores

$$Mod_{PM} : Sign_{PM} \longrightarrow Set^{op}$$

y

$$sen_{PM} : Sign_{PM} \longrightarrow Set$$

Para la introducción del concepto de verdad en un modelo de Kripke seguimos el procedimiento habitual, lo que nos lleva a establecer, por inducción sobre φ, que para todo morfismo $H : \Sigma \longrightarrow \Sigma'$ y $\Sigma'-$modelo $\mathcal{M}'$,

$$Mod_{PM}(\phi)(\mathcal{M}') \Vdash_{\Sigma} \varphi \text{ syss } \mathcal{M}' \Vdash_{\Sigma'} sen_{PM}(\phi)(\varphi)$$

Por lo tanto así se presenta a la lógica *S4* como una institución.

A continuación expongo algunos de los detalles de esta definición.

Una *signatura modal proposicional* Σ es la unión de dos conjuntos disjuntos de signos que llamamos proposiciones atómicas

$$P = \{P_n \,|n \geq 0\}$$

y el signo para lo falso; $\{\bot\}$.

Un *morfismo de signaturas modales proposicionales* H, desde una signatura Σ a una signatura Σ' es una función

$$f : \Sigma \longrightarrow \Sigma'$$

tal que $f(\bot) = \bot$.

Definición 398 *Sea* Σ *una signatura modal proposicional. Entonces, un* Σ*-modelo de S4 consiste en: (1) un conjunto no vacío* $\mathbf{W}$*, llamado el universo de estados o mundos; (2) una relación binaria* $\mathbf{R}$*,* $\mathbf{R} \subseteq \mathbf{W} \times \mathbf{W}$*, que es reflexiva y transitiva; y (3) una función*

$$V : \Sigma \longrightarrow \wp(\mathbf{W})$$

que asigna un conjunto de estados a cada proposición atómica y tal que $V(\bot) = \emptyset$

El functor

$$Mod_{PM} : Sign_{PM} \longrightarrow Set^{op}$$

envía cada signatura Σ a la clase $Mod(\Sigma)$ de todos los $\Sigma-$modelos, y cada morfismo de signaturas

$$H : \Sigma \longrightarrow \Sigma'$$

a la función

$$Mod_{PM}(H) : Mod(\Sigma') \longrightarrow Mod(\Sigma)$$

que envía cada $\Sigma'-$modelo

$$\langle \mathbf{W}', \mathbf{R}', V' \rangle$$

al $\Sigma-$modelo

$$\langle \mathbf{W}, \mathbf{R}, V \rangle$$

tal que

$$\mathbf{W} = \mathbf{W}', \mathbf{R} = \mathbf{R}' \text{ y } V = V'; H$$

Una $\Sigma-$fórmula φ es un elemento del conjunto *"carrier"* del álgebra libre $sen_{PM}(\Sigma)$ que tiene como generadores los elementos de Σ y cuya signatura está constituida por los operadores prefijos unarios —esto es, $\neg$, $\Diamond$ y $\Box$ — y por los operador infijos binarios —esto es, $\vee$, $\wedge$, $\rightarrow$ y $\leftrightarrow$ —.

El functor

$$sen_{PM} : Sign_{PM} \longrightarrow Set$$

envía cada signatura Σ al conjunto $sen_{PM}(\Sigma)$ de todas las $\Sigma-$fórmulas, y cada morfismo de signaturas

$$H : \Sigma \longrightarrow \Sigma'$$

a la función

$$sen_{PM}(H) : sen_{PM}(\Sigma) \longrightarrow sen_{PM}(\Sigma')$$

inducida por H.

Sea φ una $\Sigma-$fórmula y $\mathcal{M} = \langle \mathbf{W}, \mathbf{R}, V \rangle$ un $\Sigma-$modelo. Entonces, $\mathcal{M}(\varphi)$ es el conjunto de estados o mundos en los que φ es verdadero, y se define como en la sección 8.2.2.

Un $\Sigma-$modelo

$$\mathcal{M} = \langle \mathbf{W}, \mathbf{R}, V \rangle$$

satisface una $\Sigma-$fórmula φ si y sólo si $\mathcal{M}(\varphi) = \mathbf{W}$. En este caso, escribimos $\mathcal{M} \Vdash_{\Sigma} \varphi$.

Por inducción estructural sobre φ, se puede demostrar que para todo morfismo $H : \Sigma \longrightarrow \Sigma'$ y $\Sigma'-$modelo $\mathcal{M}'$,

$$Mod_{PM}(H)(\mathcal{M}') \Vdash_{\Sigma} \varphi \text{ syss } \mathcal{M}' \Vdash_{\Sigma'} sen_{PM}(H)(\varphi)$$

Por lo tanto,

$$\langle Sign_{PM}, Mod_{PM}, sen_{PM}, \Vdash \rangle$$

define la lógica $S4$ como una institución.

15.4. El Mapa de Instituciones $PMS4 \longrightarrow FOEQ$

Se define un *mapa* de instituciones entre la lógica proposicional modal *S4* y la lógica multivariada. La función

$$\alpha(\varphi, u) = \forall u TRANS(\varphi, u)$$

—para la función $TRANS$ se usa la de la sección 13.4.1—
define una transformación natural.

Se precisa así mismo de un functor Δ que envía:

1. cada teoría modal proposicional (Σ, Γ) a una teoría multivariada de primer orden integrada por el conjunto $MODO(S4)_{\Sigma}$ y las traducciones de las fórmulas de la teoría (Σ, Γ)

2. cada morfismo de teorías modales proposicionales a un morfismo de teorías multivariadas de primer orden

Si añadimos a lo anterior la conversión de estructuras, $CONV_2$, el resultado es un *mapa* de *instituciones.*

Estos son algunos de los detalles del *mapa* de instituciones entre la lógica proposicional modal $S4$ y la lógica multivariada[2].

El functor

$$\Delta^{\Diamond} : Sign_{PM} \longrightarrow Sign_{FoEq}$$

envía:

- cada signatura modal proposicional Σ a la signatura heterogénea

  $$\Omega = \langle S, \Upsilon, \Pi \rangle$$

  tal que $S = \{0, 1\}$, Υ es una familia de conjuntos vacíos y Π es la familia formada por los conjuntos

  $$\Pi_1 = \Sigma - \{\bot\}, \quad \Pi_{10} = \{\varepsilon\}, \quad \Pi_{11} = \{S, \equiv_1\}, \quad \Pi_{00} = \{\equiv_0\}$$

[2] Para la definición de la institución $FOEQ$, ver Ejemplo 67 en [4].

- cada morfismo de signaturas modales proposicionales $H : \Sigma \longrightarrow \Sigma'$, al morfismo
$$\Delta^{\Diamond}(H) = (H_1, H_2, H_3) : \Delta^{\Diamond}(\Sigma) \longrightarrow \Delta^{\Diamond}(\Sigma')$$
tal que H_1 y H_2 son las funciones identidad y $H_3 = H$ para todos los predicados en Π_1 y la función identidad para el resto de conjuntos de predicados.

La función
$$\alpha : sen_{PM}(\Sigma) \times \{u\} \longrightarrow sen_{FoEq}(\Delta^{\Diamond}(\Sigma))$$
tal que para toda $\varphi \in sen_{PM}(\Sigma)$ y u una variable individual de la lógica multivariada,
$$\alpha(\varphi, u) = \forall u TRANS(\varphi, u)$$
define una transformación natural
$$\alpha : sen_{PM} \Longrightarrow sen_{FoEq} \circ \Delta^{\Diamond}$$
El functor
$$\Delta : Th_{PM} \longrightarrow Th_{FoEq}$$
envía:

- cada teoría modal proposicional (Σ, Γ) a la teoría heterogénea de primer orden
$$(\Delta^{\Diamond}(\Sigma), MODO(S4)_{\Sigma} \cup \alpha(\Gamma))$$
—Para la definición del conjunto $MODO(S4)_{\Sigma}$ ver la página 350, o consultad [6]—
- cada morfismo de teorías modales proposicionales
$$H : (\Sigma, \Gamma) \longrightarrow (\Sigma', \Gamma')$$
al morfismo de teorías heterogéneas de primer orden
$$\Delta(H) = \Delta^{\Diamond}(H) : \Delta(\Sigma, \Gamma) \longrightarrow \Delta(\Sigma', \Gamma')$$

La familia de funciones $CONV_2$, tal que para cada $(\Sigma, \Gamma) \in |Th_{PM}|$,
$$CONV_{2_{(\Sigma,\Gamma)}} : Mod_{FoEq}(\Delta(\Sigma, \Gamma)) \longrightarrow Mod_{PM}(\Sigma, \Gamma)$$
envía cada $\Delta(\Sigma, \Gamma)-$modelo $\langle A, \mu, \nu \rangle$ a un (Σ, Γ)-modelo $\langle \mathbf{W}, \mathbf{R}, V \rangle$, tal que $\mathbf{W} = A_1$, $\mathbf{R} = \nu(S)$, y $V(P) = \nu(P)$, es una transformación natural.

(Lema 4.5.2 en [6].) Para todo
$$\Sigma \in Sign_{\mathbf{PM}}, \varphi \in sen(\Sigma) \text{ y } \mathcal{M} \in Mod_{\mathbf{FoEq}}(\Delta(\Sigma, \emptyset))$$
se cumple que:
$$\mathcal{M} \Vdash_{\Delta^{\Diamond}(\Sigma)} \alpha(\varphi) \text{ syss } CONV_{2_{(\Sigma,\emptyset)}}(\mathcal{M}) \Vdash_{\Sigma} \varphi$$
Por tanto, $(\Delta, \alpha, CONV_2)$ define un mapa de instituciones: $PMS4 \longrightarrow FOEQ$.

15.4.1. El Teorema de Representación para $PMS4$

En el capítulo 13 presento un plan general para definir correspondencias entre lógicas que tienen como objetivo arrastrar ciertas propiedades desde la lógica marco a la lógica objeto de estudio; por ejemplo, enumerabilidad, compacidad y Löwenheim-Skolem. El siguiente lema relaciona *mapas de instituciones* con *correspondencia entre lógicas.*

Lema 399 *Sean*

$$\Im = \langle Sign, sen, Mod, \Vdash \rangle$$

y

$$\Im' = \langle Sign, sen', Mod', \Vdash' \rangle$$

instituciones, y sea $(\Phi, \alpha, \beta) : \Im \longrightarrow \Im'$ *un mapa de instituciones tal que para cada* $\Sigma \in Sign$, $\beta_{(\Sigma,\emptyset)}$ *es una función epiyectiva. Entonces, para cada* $\Sigma \in Sign$, $\varphi \in sen(\Sigma)$, *se cumple que: (Teorema de representación)*

$$\emptyset' \models_{sign'(\Phi(\Sigma,\emptyset))} \alpha_\Sigma(\varphi) \quad syss \quad \models_\Sigma \varphi$$

donde $\emptyset'$ *es el conjunto de axiomas de la teoría* $\Phi(\Sigma, \emptyset)$.

De hecho,

$$(\Delta, \alpha, CONV_2) : PMS4 \longrightarrow FOEQ$$

define una correspondencia en el nivel del teorema de representación.

Comentario 400 *Vimos [13] que hay tres niveles de* correspondencia *entre lógicas: (1) teorema de representación, (2) teorema principal, y (3) equivalencia de cálculos. El lema demostrado en [3] para* mapas de instituciones *establece el nivel de correspondencia entre lógicas del teorema de representación. Es de esperar que estos resultados puedan mejorarse en investigaciones futuras.*

Bibliografía

[1] Estany, A. y otros eds. [1997]. *Actas del II Congreso de la Sociedad de Lógica, Metodología y Filosofía de la Ciencia.* Serveis de Publicacions de la UAB. Barcelona.

[2] Clarke, E. and Kozen, D. editores. [1984]. *Proceedings of the Logics of Programming Workshop*, volume 164 of Lecture Notes in Computer Science, pages 221–256. Springer-Verlag. Berlín. Alemania.

[3] Clavel, M. y Manzano, M. [1997]. *"Mapas de Instituciones: de la Lógica Modal S4 a la lógica multivariada"* en [1].

[4] Goguen, J. y Burstall, R. [1984]. *Introducing institutions.* En [2].

[5] Goguen, J. y Burstall, R. [1992]. *"Institutions: Abstract model theory for specification and programming".* Journal of the ACM, 39(1):95–146, .

[6] Manzano, M. [1996]. *Extensions of first order logic*. Number 19 in Cambridge Tracts in Theoretical Computer Science. Cambridge University Press. Cambridge. U.K.

[7] Meseguer, J. [1989]. *"General logics"*. En *Logic Colloquium'87*, páginas 275–329. North-Holland. Amsterdam. Holanda.

Capítulo 16

Sistemas Deductivos Etiquetados

16.1. Introducción

Gabbay se propone demostrar que la lógica clásica puede usarse como sistema universal, al menos desde el punto de vista de la teoría de la prueba: existe un algoritmo que traduce *casi* cualquier lógica a lógica clásica.

Considera que la única forma de responder de forma convincente a la cuestión sobre la universalidad de la lógica clásica es investigar cuidadosamente cómo actúa y cuales pudieran ser los procedimientos de traducción. Su conclusión es que aunque sea ésta —incluyendo claramente en ella a la multivariada, y planteándose si también hacerlo con la de segundo orden— matemáticamente adecuada, queda por resolver si es *"natural"* y si los mecanismos de razonamiento se adaptan bien a los campos respectivos. Su consejo es usar lógicas especializadas para mantenerse próximo a las aplicaciones, pero sacrificarlas desde el punto de vista computacional y usar la maquinaria clásica para el *"trabajo pesado"*. Hay también un compromiso con el razonamiento híbrido, que resulta de la combinación de lógicas.

Pese al conservadurismo que podría sugerir esta declaración de principios, lo que hace es construir un marco lógico más flexible y próximo a las áreas de aplicación, investigando primero las necesidades que dichas aplicaciones suscitan. Olvidando el tema de la comparación entre lógica clásica y no clásica, desarrolla una lógica nueva, que según él aúna las ventajas de la lógica clásica y de la no-clásica.

A partir de una serie de *"experiencias de laboratorio"*, desarrolla un sistema mejor. Este sistema se traduce a lógica clásica y su traducción preserva, hasta cierto punto la *naturalidad.* Gabbay considera que para poder presentar a la lógica clásica dentro de una pluralidad de lógicas necesitamos una noción de *qué es la lógica* muy amplia, completamente general. Para ello propone el sistema de los *LDS* y muestra cómo unifica y sirve de marco para otras lógicas,

especialmente cómo se resuelve para el caso del *cálculo Lambeck* y para la *lógica temporal.*

Su idea más revolucionaria es que la diferencia entre sistemas reside en el metalenguaje: la mayor parte de las lógicas están basadas en *modus ponens*, las reglas de cuantificadores son las mismas; así que se distinguen en ciertos aspectos de su teoría de la prueba o de su semántica que se establecen desde fuera. Si se encontrase una lógica en la que las peculiaridades metalingüísticas se cifrasen en el lenguaje objeto —para ser recogidos en un cálculo— podríamos razonablemente esperar que ésta fuese un *marco general.* El que aparentemente mejor resuelve los problemas planteados es el de los sistemas deductivos etiquetados, *LDS.*

Para apoyar su argumento se propone hacer dos cosas:

1. Mostrar que *casi* cualquier sistema lógico puede presentarse de forma natural como *LDS.*

2. *LDS* se traduce a lógica clásica de forma *natural.*

¿Es de calidad la traducción?

Una posible objeción es la siguiente, la propia lógica clásica se traduce tanto a lógica intuicionista, como a lógica lineal; incluso a modal. ¿Por qué preferimos la clásica?

La fuerza del argumento descansa en los resultados siguientes:

- *Casi* cualquier lógica puede traducirse a lógica clásica
- Algunas lógicas poseen requisitos funcionales que reciben mejor tratamiento al ser traducidas
- Algunos de estos requisitos sólo se explicitan al ser traducidos

Resumen 401 *La motivación de Gabbay al crear los* ***Sistemas Deductivos Etiquetados (LDS)*** *era la de encontrar un concepto de lógica que permitiera el paso de un sistema lógico a otro de forma natural y siguiera patrones predefinidos de variación. Se trataría de establecer un marco en el que los aspectos computacionales de la lógica fueran lo relevante, así como de dar cuenta de las diferentes técnicas de la tradición lógica de forma simple y global.*

16.2. ¿Qué es un Sistema Lógico?

Gabbay empieza proponiendo una respuesta simple, que se va concretando, enriqueciendo y detallando. Para comenzar, un sistema lógico es una relación de consecuencia entre conjuntos de fórmulas. Por ello podemos entender toda relación entre conjuntos de fórmulas de la forma $\Delta \mid\!\sim \Gamma$ que se defina matemáticamente y que satisfaga los requisitos de reflexividad, monotonía y corte. Sus pares se pueden generar semánticamente, de manera conjuntista o mediante un algoritmo.

Para esto último hay distintas opciones: podemos generar primero el conjunto de los teoremas lógicos $\{A \mid \varnothing \mathrel{|\!\sim} A\}$ mediante un procedimiento axiomático (por ejemplo, en el cálculo de Hilbert) y a partir de ellos el de los pares de la relación $\mathrel{|\!\sim}$. También podemos generar directamente los pares de la relación; es decir, todos los de la forma

$$\{\langle \Delta, \Gamma\rangle \mid \Delta \mathrel{|\!\sim} \Gamma\}$$

mediante un cálculo tipo Gentzen. O utilizar cualquier otro procedimiento de prueba.

Como punto de partida propone una definición provisional, para la que concreta los apartados siguientes[1]: (1) Un lenguaje formal L, (2) La relación de consecuencia $\Delta \mathrel{|\!\sim} \Gamma$ y (2) Una semántica para L.

Un sistema deductivo $S_{\mathrel{|\!\sim}}$ no es otra cosa que un modo de generar $\mathrel{|\!\sim}$.

Puesto que Gabbay concede especial importancia al procedimiento deductivo, una lógica no viene determinada simplemente por la relación $\mathrel{|\!\sim}$ sino que se precisa entender también su peculiar modo de obtención de los pares de la relación de consecuencia; así podríamos definir una lógica como un par: $\langle \mathrel{|\!\sim}, S_{\mathrel{|\!\sim}}\rangle$

El paso siguiente se da al observar que el razonamiento humano no es estático, que agrega continuamente nuevos argumentos y que tampoco es monotónico. Ello le lleva a elaborar su primera idea de un sistema deductivo etiquetado, concebido como marco en el que desarrollar lógicas. Se separa claramente del planteamiento tradicional cuando permite que los argumentos se sumen; esto se puede hacer en los LDS. Ahora se podría definir

$$\textit{Lógica} = LDS + \textit{ciertos mecanismos}$$

Sistema Lógico como Relación de Consecuencia

Para definir un sistema lógico se introduce su lenguaje L; es decir, su alfabeto, sus reglas de formación de fórmulas y como resultado de ello, el conjunto de sus fórmulas. A continuación se pasa a definir la relación de consecuencia de esa lógica, usando la notación

$$\Delta \mathrel{|\!\sim_L} Q$$

para indicar que "Q *se sigue de* Δ *en la lógica* L"

De acuerdo con lo anteriormente planteado, la relación de consecuencia cumple:

1. $\Delta \mathrel{|\!\sim} Q$, si $Q \in \Delta$ —reflexividad—
2. $\Delta \mathrel{|\!\sim} Q \Longrightarrow \Delta, \Delta\prime \mathrel{|\!\sim} Q$ —monotonía—
3. $\Delta \mathrel{|\!\sim} A$; $\Delta, A \mathrel{|\!\sim} Q \Longrightarrow \Delta \mathrel{|\!\sim} Q$ —corte—

[1] Se expuso con algo más de detalle en la página 329.

La relación de consecuencia se puede definir explícitamente —es decir, mediante un sistema algorítmico, $S_{\shortmid\sim}$— o implícitamente —enunciando como postulados las propiedades que debe cumplir $\shortmid\sim$—

Si pensamos que Δ es una base de datos y Q una pregunta, entonces la reflexividad se interpreta diciendo que se obtiene respuesta a toda pregunta cuya solución esté en la base de datos. Monotonía indica acumulación de información y la regla del corte indica que se pueden usar lemas intermedios.

Tomando a Q como conjunto, lo anterior se generaliza así:

1. $\Delta \shortmid\sim \Gamma$, si $\Delta \cap \Gamma \neq \varnothing$ —reflexividad—
2. $\Delta \shortmid\sim \Gamma \Longrightarrow \Delta, \Delta\prime \shortmid\sim \Gamma$ —monotonía—
3. $\Delta \shortmid\sim A \ ; \ \Delta, A \shortmid\sim \Gamma \Longrightarrow \Delta \shortmid\sim \Gamma$ —corte—

Puesto que estas nociones son monotónicas y la *I.A* abandona este principio, muchos de los sistemas sólo coinciden en reflexividad restringida; es decir, en la regla $A \shortmid\sim A$. También, en ocasiones, se restringe la monotonicidad

$$\frac{\Delta \shortmid\sim A \ ; \ \Delta \shortmid\sim B}{\Delta, A \shortmid\sim B}$$

Sistemas Lógicos como *LDS*

Una lógica se había definido como un par $\langle \shortmid\sim, S_{\shortmid\sim} \rangle$ donde $\shortmid\sim$ es una relación de consecuencia estructurada —pudiera no ser monotónica, pero cumple alguna versión restringida de identidad y de corte— sobre un lenguaje L; $S_{\shortmid\sim}$ es ahora un *LDS*.

Un sistema *LDS* es una tripla ordenada, $\langle L, A, M \rangle$, donde L es un lenguaje lógico, A es un álgebra de etiquetas y M es la disciplina de etiquetado de fórmulas de la lógica. Contiene dos tipos de reglas: las de deducción y las de distribución y propagación de etiquetas.

Ejemplo 402 *L : lenguaje de primer orden, lógica clásica. A : etiquetas generadas con las constantes y variables del lenguaje usando functores (por ejemplo, $t_1(y)$, $t_2(x, y)$, etc). Las fórmulas y las etiquetas comparten variables. Se forma con ambas las* ***unidades declarativas*** *que son de la forma $t_1(y) : \exists x A(x, y)$, $t_1(a) : \exists x A(x, a)$*

16.3. Lenguaje de *LDS*

La definición precisa de estos sistemas deductivos etiquetados es como sigue:

Definición 403 *Partimos de un lenguaje de primer orden, cuyos signos peculiares constituyen la familia $\mathrm{A} = \langle A, R_1, ..., R_k, f_1, ..., f_m \rangle$ donde A es el conjunto de términos del álgebra, que incluye variables y constantes; cada R_i es un relator $i-$ario y cada f_j es un functor $j-$ario. Los elementos de A son*

etiquetas atómicas, con los functores se generan más etiquetas y los relatores les otorgan una cierta estructura.
Un ***diagrama*** *de etiquetas es un conjunto M que contiene los elementos generados a partir de A usando functores y las fórmulas atómicas $\pm R(t_1, ..., t_k)$, donde cada $t_i \in M$ y el relator R es del álgebra*
Sea L un lenguaje de primer orden que contiene las conectivas $\#_1, ...\#_n$ de diferentes ariedades, con cuantificadores y con los mismos términos atómicos que el álgebra.

Definimos los conceptos de *unidad declarativa, base de datos* y *etiqueta* así:

Definición 404 ***Etiquetas****: Atómicas, $t \in A$. De forma general todas las que se forman recursivamente a partir de A usando los functores $f_1, ..., f_m$.*
Fórmulas: *las fórmulas de L*
Unidad declarativa: *es un par de la forma $t : B$ donde t es una etiqueta y B es una fórmula.*
Base de datos: *Son unidades declarativas o de la forma $\langle a,\ M,\ \mathbf{f}\rangle$ donde M es un diagrama finito; $a \in M$ es una etiqueta singularizada y ; $\mathbf{f}$ asocia a cada etiqueta t de M o bien una base de datos o un conjunto finito de fórmulas. (Esta cláusula es recursiva. Una base de datos simple se obtiene siempre que la función $\mathbf{f}$ asocie a cada etiqueta un conjunto finito de fórmulas.)*

Una vez definido *LDS* se pasa a justificar su adecuación como *"lingua franca"*, lo que hemos venido llamando *marco unificador*.

¿Qué es un marco unificador? Como ya dije, hay reducciones que no reciben tal consideración. Ciertamente toda lógica puede ser simulada mediante una máquina de Turing. ¿Vale esto como marco unificador? En nuestro caso, ¿usamos las etiquetas para que emulen máquinas de Turing? La respuesta es que aunque se trate de características del metalenguaje, las etiquetas no actúan en ese nivel, lo hacen directamente sobre el lenguaje objeto. Además, hay varias restricciones sobre el uso de *LDS*. En especial:

1. Sólo se usan reglas de inferencia clásicas: *modus ponens* y una versión del *teorema de la deducción.*

2. La distribución de etiquetas obedece reglas fijadas para toda lógica, pero aunque el formato sea fijo, permite variaciones de detalle que hacen que incluso la regla de *modus ponens* se restrinja para ciertas lógicas.

3. Las variaciones entre lógicas que se producen en el metalenguaje aquí se expresan en el lenguaje objeto, en el mecanismo de etiquetado.

Resumen 405 *Un sistema* ***LDS*** *es una tripla ordenada,*

$$\langle L, A, M\rangle$$

donde L es un lenguaje lógico, A es un álgebra de etiquetas y M es la disciplina de etiquetado de fórmulas de la lógica. Contiene dos tipos de reglas: las

de deducción y las de distribución y propagación de etiquetas. ***LDS*** *supone también el concepto de* ***base de datos****. Las unidades declarativas están formadas por fórmulas y etiquetas; estas últimas aportan información extra sobre la fórmula que no está codificado en ella. De hecho, las etiquetas funcionan como un sistema de control de las manipulaciones simbólicas a las que sometemos a las fórmulas en un proceso deductivo; guardan memoria de lo acontecido en el mismo. Las fórmulas etiquetadas aparecen en bases de datos que, a su vez, pueden estar etiquetadas, lo que permite que puedan intervenir en el proceso deductivo. Las bases de datos pueden representar varias cosas: por ejemplo, un supuesto transitorio utilizado en una inferencia del cálculo de deducción natural para lógica clásica, o los mundos de la lógica modal o los estados de una máquina en lógica lineal, etc. El uso reglado de las etiquetas y lo que en ellas hayamos cifrado distingue a una lógica de otra.*

16.4. Demostraciones en *LDS*

En *LDS* se sustituye el concepto de prueba tradicional

$$B_1, ..., B_n \vdash C$$

por el de prueba de fórmulas con etiquetas

$$t_1 : B_1, ..., t_n : B_n \vdash s : C$$

Mientras que en la lógica tradicional las reglas de deducción se definen para fórmulas, en *LDS* se toman en consideración tanto ellas como sus etiquetas. Tenemos reglas para manipular las etiquetas, de forma que es ahí, en el álgebra de las etiquetas en donde residen las características del metalenguaje responsables de las diferencias entre lógicas; las reglas de las fórmulas se reflejan en el lenguaje objeto.

Ahora caracterizamos a un sistema lógico como a un par $\langle \mid\!\sim,\ LDS_{\mid\sim}\rangle$ donde $\mid\!\sim$ es una relación de consecuencia entre Δ, una base de datos etiquetada y unidades declarativas $t : A$ y $LDS_{\mid\sim}$ es un sistema algorítmico para calcular la relación anterior.

Retorno a la lógica original

Antes hablábamos de que un sistema lógico es un par $\langle \mid\!\sim,\ S_{\mid\sim}\rangle$ donde $\mid\!\sim$ es una relación de consecuencia que verifica Identidad y corte *quirúrgico* y $S_{\mid\sim}$ un algoritmo para calcularlo. Ahora se ha reelaborado el concepto, se han introducido los sistemas deductivos etiquetados, ¿Cómo rescatamos el concepto antiguo $\langle \mid\!\sim,\ S_{\mid\sim}\rangle$ a partir del nuevo $\langle \mid\!\sim,\ LDS_{\mid\sim}\rangle$? Es decir, debemos ahora añadir a nuestro sistema *LDS* la posibilidad de eliminar las etiquetas; a este proceso le denominamos *compresión.*

Ejemplo 406 *(Lógica lineal y de la relevancia). Considerad un lenguaje proposicional que sólo contiene el signo* $\rightarrow$ *Como regla tenemos* modus ponens MP *y*

deducción TD. *Tenemos un conjunto de etiquetas:* $\{a_1, a_2, ...\}$. *Desde el punto de vista de un probador de teoremas, la regla de modus ponens no deja de ser del lenguaje objeto. Por lo tanto, una demostración de*

$$\vdash (B \to A) \to ((A \to B) \to (A \to B))$$

podría hacerse de esta forma:
supongamos que $a_1 : B \to A$ *y demuestra que* $((A \to B) \to (A \to B))$. *Asume que* $a_2 : A \to B$ *y demuestra que* $A \to B$. *Asume que* $a_3 : A$ *y demuestra que* B. *El problema se plantea así:*

Supuestos			
1.-	$a_1 : B \to A$		
2.-	$a_2 : A \to B$		
3.-	$a_3 : A$		
Deducción			
4.-	$a_2a_3 : B$	*MP*	*2 y 3*
5.-	$a_1a_2a_3 : A$	*MP*	*4 y 1*
6.-	$a_2a_1a_2a_3 : B$	*MP*	*5 y 2*
7.-	$a_2a_1a_2 : A \to B$	*TD*	*3 y 6*
8.-	$a_2a_1 : (A \to B) \to (A \to B)$	*TD*	*2 y 7*
9.-	$a_2 : (B \to A) \to ((A \to B) \to (A \to B))$	*TD*	*8 y 1*

16.5. Semántica

No se puede concluir sin mencionar al menos la semántica. El punto de vista tradicional, tanto para la clásica como para la intuicionista o la modal, es introducir el concepto de clase de modelos y el de evaluación de las fórmulas. Es decir, tenemos una clase $\mathfrak{M}$ de modelos y definimos $m \Vdash A$ para cada modelo $m \in \mathfrak{M}$ (validez de A en $\mathfrak{M}$). Si no entramos en el detalle de los modelos ni de cómo se evalúan las fórmulas, lo único que podemos decir es que con cada m se asocia una función que a cada fórmula le asigna un valor en el conjunto $\{F, V\}$. Decimos que una lógica $\mid\sim$ es completa respecto de una clase $\mathfrak{M}$ de modelos si se verifica lo siguiente:

$$A \mid\sim B \text{ syss para cada } m \in \mathfrak{M} : (m \Vdash A \Longrightarrow m \Vdash B)$$

La semántica de Gabbay, por el contrario, se incluye en los propios mecanismos de *LDS* y pasa a formar parte de la sintaxis. Esto se puede hacer de forma simple en el caso de la lógica modal, donde las etiquetas denotan mundos posibles y las reglas reflejan la semántica de las fórmulas. Esto lo vimos claramente en el cálculo modal de tableaux en la sección 8.8, por estar más pegado a la semántica de las fórmulas.

¿Qué se necesita para establecer de manera meramente sintáctica las características de la semántica? ¿Qué reemplazará a los modelos, a las interpretaciones, al propio concepto de completud?

Definición 407 ***Semántica sintáctica****. Sea* $\mid\sim$ *una relación de consecuencia y sea* $\mathfrak{K}$ *una clase de relaciones de consecuencia, no necesariamente en el*

mismo lenguaje. Para cada $\mid\!\sim^* \in \mathfrak{K}$ *sea* $k_{\mid\sim^*}$ *una interpretación de* $\mid\!\sim$ *en* $\mid\!\sim^*$ *. Esto incluye un* ***mapa*** *que traduce el lenguaje propio de* $\mid\!\sim$ *al de* $\mid\!\sim^*$ *que además es un homomorfismo; es decir se cumple*

$$A \mid\!\sim B \text{ implica } A^* \mid\!\sim^* B^* \text{ (aquí } A^* \text{ es } k_{\mid\sim^*}(A) \text{ y lo mismo } B^*)$$

Decimos que $\mid\!\sim$ *es completa para* $\langle \mathfrak{K}, k\rangle$ *syss para cada* $\mid\!\sim^* \in \mathfrak{K}$, $A^* \mid\!\sim^* B^*$

Ejemplo 408 *Las siguientes pueden considerarse interpretaciones semánticas en el sentido de Gabbay*

1. *La interpretación de Boolos y Solovay de la lógica modal* G *(incluye el axioma de Löb) donde* $\Box A$ *significa "*A *es demostrable en la aritmética"*
2. *La interpretación de la lógica modal en la clásica*

Para justificar y motivar su semántica Gabbay presenta a la tradicional de esta forma.

Definición 409 *Sea* L *un lenguaje proposicional; por ejemplo, modal con* $\Box$ *o intuicionista con* $\rightarrow$ *.*

1. *Un modelo adecuado a* L *es una función* $\mathbf{s}$ *que asigna un valor en* $\{F, V\}$ *a cada fórmula de* L
2. *Una semántica* $\mathfrak{S}$ *es una clase de modelos*
3. *Sea* Δ *un conjunto de fórmulas y* A *una fórmula. Decimos que* $\Delta \vDash_{\mathfrak{S}} A$ *syss para cada* $\mathbf{s} \in \mathfrak{S}$ *: si* $\mathbf{s}(B) = V$ *para cada* $B \in \Delta$ *entonces* $\mathbf{s}(A) = V$

Gabbay afirma que este concepto de semántica es suficiente para caracterizar cualquier relación de consecuencia $\mid\!\sim$ entre conjuntos de fórmulas Δ (puede ser $\varnothing$) y cualquier fórmula A que satisfaga reflexividad, monotonicidad y corte. Por consiguiente, para cada $\mid\!\sim$ existe un $\mathfrak{S}$ tal que $\mid\!\sim$ equivale a $\vDash_{\mathfrak{S}}$.

A la semántica $\mathfrak{S}$ se le puede proporcionar más estructura, siempre en función de sus conectivas. Por ejemplo, se puede usar una relación binaria definida así:

$$\mathbf{t} \leq \mathbf{s} \text{ syss para cada } A : \mathbf{t}(A) \leq \mathbf{s}(A)$$

Se pueden definir otras relaciones sobre $\mathfrak{S}$; por ejemplo, para la lógica modal definimos:

$$\langle \mathbf{t}, \mathbf{s}\rangle \in R \text{ syss para cada } \Box A \text{ de } L : \mathbf{t}(\Box \mathbf{A}) = V \Longrightarrow \mathbf{s}(\Box A) = V$$

O se pueden postular conexiones entre valores,

$$t(\Box A) = V \text{ syss } \forall \mathbf{s}(\langle \mathbf{t}, \mathbf{s}\rangle \in R \Longrightarrow \mathbf{s}(A) = V)$$

Para la lógica intuicionista se introduce la siguiente condición:

$$\mathbf{t}(A \rightarrow B) = V \text{ syss } \forall \mathbf{s}(\mathbf{t} \leq \mathbf{s} \;\&\; \mathbf{s}(A) = V \Longrightarrow \mathbf{s}(B) = V)$$

La naturaleza de lo que está sucediendo aquí se explica muy bien usando la traducción a la lógica clásica, que veremos en la siguiente sección.

Conclusión 410 *Finaliza Gabbay su presentación de los* ***LDS*** *comparándola con la lógica clásica: el camino entre ambas es de ida y vuelta.*

- *Todo sistema* ***LDS*** *se puede traducir a la lógica clásica bivariada. Una de las variedades lo será para el álgebra de etiquetas. La traducción de* $t : A$ *es* $A^*(t)$*. Este es exactamente el procedimiento que seguimos en la traducción de la lógica modal. De esta forma el apoyo de la tesis de la universalidad de la lógica clásica viene dado por la generalidad de* ***LDS*** *y su papel unificador; no se necesita, según Gabbay, una interpretación semántica de la traducción. La elección del álgebra de etiquetas para representar a la lógica fue aquí lo verdaderamente determinante.*

- *Es interesante señalar que a su vez la lógica clásica puede presentarse como un* ***LDS*** *de forma no trivial. Tomad un predicado cualquiera de la lógica clásica; por ejemplo,* $Q(x, y, z)$ *e imaginad una teoría clásica* T *para él. Puede que al examinar los modelos en sentido clásico de* T *encontremos ciertas regularidades, algunos elementos destacados que nos permitan describir mejor* Q*. Nos puede ayudar escribir* $x : Q^*(y, z)$ *en vez de* $Q(x, y, z)$ *y establecer así semánticamente esta estructura. Este proceso es el inverso del de traducción y lo que interesa es hacer hincapié en las propiedades semánticas de* x*. Las* $x-$*etiquetas se manipularán separadamente, siguiendo los dictados de la teoría de modelos y el resultado podría ser una nueva teoría de la demostración para* T *más explícita. Esto significaría una reformulación de* T *desde* ***LDS****.*

Termina diciendo que podemos prescindir de la semántica entendida en los términos de la teoría de modelos. Según su criterio, puede ser sustituida por una correcta traducción. Dados dos sistemas L_1 y L_2 decimos que L_2 puede ser usado como semántica para L_1 si el último puede ser traducido correctamente a L_2. De esta forma lo que correspondería a un teorema de completud sería el teorema de adecuación de la traducción (nuestro teorema 13.4.1)

Semánticas de esta guisa se han dado en la historia de la lógica; la ya mencionada interpretación de Boolos de la lógica modal G en la aritmética de Peano y todas las *semánticas operacionales* que se usan en informática teórica.

Bibliografía

[1] Gabbay, D. [1996]. *Labelled Deductive Systems; principles and applications.* Vol 1: Introduction. Oxford Logic Guides, Vol. 33, Oxford University Press, Oxford. U.K.

[2] Gabbay, D. *Labelled Deductive Systems; principles and applications.* Vol 2: Further Developments. Oxford University Press. In preparation.

[3] Gabbay, D. [1986] *"What is Negation in a System?".* En F. R. Drake and J. K. Truss, editors, Logic Colloquium '86, Elsevier Science Publishers (North Holland)

[4] Gabbay, D. [1993]. *"General Theory of Structured Consequence Relations".* En P. Schroeder-Heister and K. Dosen, editors, *Substructural Logics*, Studies in Logic and Computation Series, Oxford University Press. Oxford. U.K.

[5] Gabbay, D. [1993]. *"Labelled Deductive Systems and Situation Theory".* En P. Aczel, D. Israel, Y. Katagin, and S. Peters, editors, S*ituation Theory and Applications*, Vol. 3, CSLI.

[6] Gabbay, D. [1994]. *"What is a Logical System?"* En D. Gabbay, editor, *What is a Logical System?*, Oxford University Press. Oxford. U.K.

[7] Gabbay, D. [1994]. *"Classical vs. Non-classical Logic".* En D. Gabbay, C. J. Hogger, J. A. Robinson, and J. Siekmann, editores, Handbook of Logic in Artificial Intelligence and Logic Programming, Vol. 2, Oxford University Press. Oxford. U.K.

Capítulo 17

Recapitulemos

17.1. Fundamentos (*Lógica*)

Hemos estudiado la lógica clásica en la primera parte de este libro; esto es, en los cinco primeros capítulos. Allí vimos los fundamentos de la disciplina tal y como se desarrolló bajo la presión de las necesidades de la matemática, con el énfasis puesto en la justificación del razonamiento atemporal y estático propio de este área del saber. Se sentaron las bases del resto de las ramas de la *Summa Logicae*, tanto de los desarrollos clásicos, como de las diversas alternativas y extensiones, que comparten cuanto menos el espíritu de rigor y precisión[1] que se fraguó en ese ámbito. Introdujimos la lógica matemática en sus diversas ramas de *teoría de modelos*, *de la recursión* y *de la prueba*, así como la *teoría de conjuntos*. Por supuesto, se pretendía presentar los resultados de manera crítica, comentada, como corresponde al espíritu filosófico de nuestra Área de conocimiento[2], pero hemos aprovechado el espacio para hacer hincapié tanto en los aspectos que pudieran interesar a un filósofo de la matemática, como a un informático, o a un estudioso de la Inteligencia Artificial. Por ejemplo, en el capítulo 5 dedicado a la teoría de conjuntos no nos hemos contentado con enumerar sus axiomas y algunas de sus consecuencias, sino que hemos seguido el ejercicio abductivo, cuasi detectivesco de Devlin y los hemos justificado por servir de andamiaje imprescindible en la construcción de la jerarquía de Zermelo, que es a mi juicio la imagen más satisfactoria intuitivamente del universo matemático.

Los cálculos deductivos que hemos desarrollado con más detalle son los que mejor se *implementan* y que por consiguiente despiertan un interés mayor en la comunidad informática; a saber, los de tableaux y de resolución, así como el de cláusulas de Horn. Para servir a los intereses de la *I.A.* se ha puesto énfasis en el planteamiento de los razonamientos expresados en lenguaje natural

[1] Mi tesis es que comparten algo más, que encierran a la *logicidad*, según veremos al final de este capítulo.

[2] *Lógica y Filosofía de la Ciencia.*

y en la búsqueda de soluciones en un árbol formado sólo de hipótesis. A su vez el concepto de computabilidad se ha estudiado ligado al concepto intuitivo de *procedimiento efectivo*.

17.2. Sistemas Lógicos (*Lógicas*)

En la segunda parte hemos seleccionado de entre la enorme variedad de sistemas lógicos existentes a seis, cinco de ellos extienden a la lógica clásica y el de cláusulas de Horn es una subteoría, tratándose en todos los casos de lógicas que tienen aplicaciones en filosofía, informática, matemáticas, lingüística e inteligencia artificial.

Hemos presentado con cierto detalle la lógica de segundo orden y la teoría de tipos —en los capítulos 10 y 11— que interesan al filósofo porque explicitan la tensión existente entre lo que sabemos y lo que podemos probar, y lo sensible que es la lógica superior a los supuestos ontológicos de la teoría de conjuntos subyacente. La interdependencia de la capacidad expresiva respecto de la deductiva quedó gráficamente fijada con el teorema de incompletud de la lógica estándar de segundo orden, o con la recuperación de la completud al admitir modelos no estándar. Viendo que el razonamiento de orden superior queda plasmado con naturalidad en la semántica de modelos no estándar, se fortalece la intuición de que en *SOL* con semántica estándar muchos de los resultados a ella atribuidos nada tienen que ver con la naturaleza del razonamiento efectuado, sino con su abigarrado envoltorio de teoría de conjuntos. La inspiración de la conversión de otras lógicas a la multivariada parte de la prueba de completud de la lógica superior con modelos generales y así lo expliqué en su momento, en la página 343.

Dejadme citarme a mí misma[3]:

> Further motivation for using general models may be found in van Benthem's recent essay *"The sources of complexity"*, where the author considers that with general semantics
>
> *"...we achieve a moral rarity, being a combination of philosophical virtue with computational advantage,..."*
>
> In fact, when considering the arguments used in the second chapter of this book, one can argue that the standard semantics is not logically adequate in the sense that it does not allow all logically possible interpretations of second order formulas as models because of the argument posed by Németi in the following form:
>
> *"We have to be placed in a set-theoretical universe, even assuming that there could be more than one such. Nevertheless, in the set theoretical universe you choose to be in, the GCH is either true or false.*

[3] En la página XVI del prefacio de mi libro *Extensions of First Order Logic*.

Assume it is true. Then, in every standard model for SOL the SOL formula φ expressing this hypothesis is true and so φ is valid. But since GCH is not derivable from ZFC, the result suggest that an interpretation $\Im$ such that $\Im \not\Vdash \varphi$ can not be excluded as 'logically impossible'. So, at least one $\Im$ with $\Im \Vdash \neg\varphi$ is a logical possibility (by Paul Cohen's classical result). But such a model is not allowed in the standard semantics. So, we feel that the standard semantics does not include all logically possible worlds as models (we have to think about formulas, like GCH, which are both expressible in second order logic and independent from Zermelo-Fraenkel set theory). This argument is reinforced by the fact that there is an inexhaustible supply of independent formulas like GCH. In Henkin's general semantics many possibilities are restored as possible models; for instance, models with or without the GCH."

As you will see, the general model strategy is also used in modal logic and dynamic logic. Both logics are faithfully represented by many-sorted theories with a comprehension schema restricted to a definable subclass of many-sorted formulas.

17.3. Tratamientos unificadores (*Logicidad*)

Intentaré sintetizar mi punto de vista. Para poder comparar sistemas lógicos es preciso saber qué son, ello nos conduce a una disyuntiva que se concreta en dos programas de investigación, posiblemente convergentes:

1. Ver qué problemas intentaban resolver con la esperanza de que la *"logicidad"* se oculte allí[4]; Este proyecto se retroalimenta ya que conlleva, además de la reflexión sobre el concepto de lógica, la investigación de aquellas áreas de estudio emergentes del análisis que estén necesitadas de lógicas que las comprendan y expresen de forma natural, relevante, explicitando mediante reglas de la lógica sus raíces empíricas, pero exigiendo al producto no sólo coherencia interna —consistencia— sino todo el rigor metodológico y la precisión matemática al que la lógica nos tiene acostumbrados. Por su propia naturaleza, este ambicioso proyecto se puede poner en marcha, pero no efectuar su balance ahora, por lo que en el presente trabajo no entra en consideración.

2. Pasando por alto el hacer previamente la definición comprensiva del concepto, usamos como criterio demarcador el meramente extensional, —obtenido de la colección de *Handbooks*— y tomamos una definición conceptual provisional, como herramienta de trabajo. Analizamos qué posibilidades tenemos para comparar lógicas, considerando de interés los siguientes criterios:

[4] Bajando la escalera, junto al *Aleph* de Borges.

a) Propiedades matemáticas de la relación de consecuencia

b) Equilibrio entre potencia expresiva y deductiva

c) Recursos matemáticos precisos para su desarrollo

d) Traducibilidad a un marco común

Así traducidas la comparación será sencillamente entre teorías de una misma lógica. El propio concepto de traducción y de lógica marco tiene que ser esclarecido; hemos optado por una traducción que permita que haya una correspondencia entre todos los componentes de las lógicas —modelos y cálculo— y hemos elegido a la lógica multivariada como marco.

Reducción de otras lógicas a la multivariada

El planteamiento está inspirado en la prueba de Henkin de completud de la teoría de tipos. Convertimos los modelos de la lógica en estudio *XL* en modelos extendidos en cuyos universos aparecen explícitamente todos los constructos definibles con las expresiones de *XL*. Para axiomatizar esta nueva clase de estructuras usamos la lógica multivariada, de la misma forma que en la prueba de completud mencionada. El *espíritu* de la correspondencia que se establece es *semántico*.

El programa se desarrolla a tres niveles, o etapas; en el primero nos detenemos en la mera representación de la verdad en la lógica en estudio como verdad de su traducción en estructuras multivariadas, módulo una cierta teoría cuya definición constituye una parte importante del programa de reducción; en el segundo la equivalencia se extiende al concepto de consecuencia a partir de conjuntos cualesquiera de hipótesis; en el tercero se demuestra la equivalencia del cálculo de la lógica en estudio con la teoría multivariada que la simula. Pese a su inspiración semántica, ya desde el primer nivel —incluso si la lógica *XL* careciera de cálculo—, podemos usar el cálculo multivariado para demostrar las fórmulas válidas de la lógica en estudio. La razón es que a partir del *teorema de representación* se demuestra fácilmente el *teorema de enumerabilidad* para la lógica *XL*. Por lo tanto, sabemos que podremos diseñar un cálculo para *XL*, pero también sabemos que en *MSL* podemos simular ese cálculo e incluso usar un *demostrador* de teoremas de *MSL*.

Así que, aunque la reducción se hubiera detenido en este nivel, los resultados desde un punto de vista práctico serían impresionantes.

Cuando la correspondencia supera también los niveles segundo y tercero obtenemos como subproducto los *teoremas de compacidad*, *Löwenheim-Skolem* y *completud* para la lógica *XL*.

En suma: Pretendemos una unificación del *"lenguaje máquina"* y realizar en él todos los cómputos que en otro caso se harían en una multiplicidad de sistemas, y arrastrar las metapropiedades de la lógica multivariada a los sistemas lógicos en estudio. Sin embargo, mantenemos los diversos lenguajes como *"lenguaje de usuario"* por ser más naturales y adaptarse mejor a sus respectivos ámbitos de estudio.

Otros planteamientos unificadores

Hemos estudiado otros planteamientos unificadores; a saber, el de las lógicas generales —en el capítulo 15— y el de los sistemas deductivos etiquetados —en el capítulo 16.

1. El análisis llevado a término en el primer caso es *supraestructural*, lo que quiere decir que se sitúa a un nivel de abstracción muy elevado. No nos extraña: la lógica subyacente es aquí la *teoría de la categorías*, que ocupa un nivel de abstracción superior al de la *teoría de modelos* que, a su vez, olvida la naturaleza de los objetos de la *"realidad matemática"* inmediata −números naturales, reales, etc.− y estudia las relaciones entre ellos. Una conceptualización de este tipo es matemáticamente muy poderosa, pero necesariamente más alejada del significado de las fórmulas.

2. En el segundo caso la maquinaria que se pone en funcionamiento es fundamentalmente sintáctica y a ella revierte la semántica. Se trata igualmente de una herramienta muy poderosa y de aplicabilidad en sistemas extraordinariamente diversos.

No deja de ser curioso que tanto cuando se utilizan en la traducción criterios *semánticos*, como cuando son *supraestructurales* o *sintácticos* el resultado obtenido sea el mismo.

¿No será que la "logicidad" equivale a la traducibilidad a lógica clásica?