UNIVERSIDAD NACIONAL AUTÓNOMA
DE MÉXICO

FACULTAD DE CIENCIAS

Un resultado de completud para progresiones transfinitas de sistemas axiomáticos recursivamente enumerables.

T E S I S

QUE PARA OBTENER EL TÍTULO DE:

Matemático

PRESENTA:

Antonio Nakid Cordero

TUTOR

Dr. Carlos Torres Alcaraz

Ciudad Universitaria, CD. MX., Agosto 2018

HOJA DE DATOS DEL JURADO

1. DATOS DEL ALUMNO
   Nakid
   Cordero
   Antonio
   56055841
   Universidad Nacional Autónoma de México
   Facultad de Ciencias
   Matemáticas
   413005845

2. DATOS DEL TUTOR
   Dr
   Carlos
   Torres
   Alcaraz

3. DATOS DEL SINODAL 1
   Dr
   David
   Meza
   Alcántara

4. DATOS DEL SINODAL 2
   M en C
   Fernando Javier
   Nuñez
   Rosales

5. DATOS DEL SINODAL 3
   Dr
   Favio Ezequiel
   Miranda
   Perea

6. DATOS DEL SINODAL 4
   M en C
   Mariana
   Martínez
   González

7. DATOS DEL TRABAJO ESCRITO
   Un resultado de completud para progresiones transfinitas de sistemas axiomáticos recursivamente enumerables
   73 p
   2018

A mi papá, por quien pienso en el infinito.

Agradecimientos

Primero que nada, a la persona más importante en mi vida, mi mamá. Quien más ha contribuido, no sólo al desarrollo y termino este trabajo, sino a toda mi vida y mi formación. Ha sido, a lo largo de muchos años, el cimiento a partir del que he crecido, el hombro en el que me he apoyado, el consejo que me ha guiado y el modelo de lo que me gustaría llegar a ser. El lenguaje no basta para decir lo mucho que te agradezco.

A mis hermanas. Lia, siempre has sido un ejemplo de lo que el trabajo duro y la disciplina puede lograr. Muchas gracias por todo lo que he aprendido de ti, ya sea directamente o a través de tu ejemplo. Ceci, de ti he aprendido que si realmente quieres algo, hay que hacer de todo para lograrlo. Gracias por ser una buena amiga durante tantos años.

Al doctor Carlos Torres, por sus cursos de lógica, por presentarme las ideas que más han ocupado mi mente los últimos años, por cambiar mi perspectiva de las matemáticas, por las conversaciones tan estímulantes y por las anécdotas tan entretenidas. Pero, más que nada, por contagiarme su pasión por los fundamentos de las matemáticas.

Al doctor David Meza, por sus atinados consejos, su buen humor, su sinceridad y todo su apoyo. A la profesora Mariana Martínez, por introducirme a la lógica matemática y a la filosofía de las matemáticas, y por su genuino interés en este trabajo. Sus comentarios ayudaron a incrementar significativamente la calidad de esta tesis. Al doctor Favio Miranda, por la precisión de sus comentarios y compartir su conocimiento y sus ideas para mejorar este trabajo. A la profesora Guadalupe Lucio, por la oportunidad de dar clase y por ser siempre la voz de la experiencia.

A Narda, por compartir conmigo el gis y la sangre, y ser un apoyo tan valioso en ambos roles.

A Javier, por el papel fundamental que ha tenido en mi formación como matemático. Por ser la persona que más matemáticas me ha enseñado, por darme trabajo haciendo una de mis cosas favoritas y, más que nada, por ser un gran amigo.

A toda la banda del prome, por retrasar considerablemente la finalización de este proyecto. Llenaría más de una página nombrarlos a todos (y probablemente olvidaría a alguno), pero ustedes saben quiénes son. Un agradecimiento especial a Andy, por las sesiones de trabajo que incluyeron más chelas que trabajo y a César, por todas las ideas que hemos discutido, y por ser un ejemplo tan genuino de la amistad.

A Ana y Karina, por la variedad de anécdotas inolvidables, por acompañarme en mis mejores momentos y soportarme en los peores.

A mis alumnos, de quienes he aprendido tanto. Especialmente a Marcela, Roxana, Diana y Elias por haberse convertido en grandes amigos.

Finalmente, a la Facultad de Ciencias y a la UNAM, por ampliar los límites de mi lenguaje, de mi mundo y de mi pensamiento.

INTRODUCCIÓN

Los teoremas de incompletud de Gödel marcan un punto de quiebre fundamental en el estudio de la lógica matemática. El descubrimiento de que una teoría tan natural como la aritmética de Peano es incompleta y –más aún– que es imposible obtener una prueba finitista de su consistencia obliga a cambiar la perspectiva que tenemos en cuanto a los fundamentos de las matemáticas. Un primer paso, dejando a un lado los métodos completamente constructivos y explorando los límites del fénomeno de incompletud, lo dio Alan Turing en su tesis doctoral realizada en Princeton entre 1936 y 1938, bajo la tutela de Alonzo Church.

En su trabajo *Systems of logic based on ordinals* [28], Turing propone construir una progresión de teorías cada vez más completas que se extiende a lo largo de la sucesión ordinal transfinita. La descripción de este proceso la realiza de forma magistral en las primeras líneas de su trabajo:

> *El bien conocido teorema de Gödel muestra que cualquier sistema de lógica es en un cierto sentido incompleto, pero al mismo tiempo indica un medio por el cual, a partir de un sistema de lógica* L*, un sistema más completo* L' *puede ser obtenido. Repitiendo este proceso obtenemos una sucesión* L, $\mathsf{L}_1 = \mathsf{L}'$, $\mathsf{L}_2 = \mathsf{L}_1'$, $\mathsf{L}_3 = \mathsf{L}_2'$, … *de lógicas cada vez más completas. Una lógica* L_ω *puede ser construida en la que los teoremas derivables sean exactamente la totalidad de los teoremas derivables con la ayuda de los sistemas* L, L_1, L_2, … *Podemos construir así* $\mathsf{L}_{2\omega}$ *relacionada con* L_ω *de la misma forma que* L_ω *está relacionada a* L*. Proceder de esta manera nos permite asociar un sistema de lógica a cualquier ordinal constructivo dado.*

Partiendo de una descripción efectiva de un sistema axiomático $\mathcal{A}$, uno puede describir recursivamente la sucesión de sistemas axiomáticos $\langle \mathcal{A}_n \rangle_{n\in\omega}$ donde $\mathcal{A}_0 = \mathcal{A}$ y $\mathcal{A}_{n+1}$ consta de los axiomas de $\mathcal{A}_n$ junto con su enunciado de consistencia $\mathrm{Con}_{\mathcal{A}_n}$. Después, es posible construir un nuevo sistema axiomático $\mathcal{A}_\omega$ considerando la unión de todos los sistemas $\mathcal{A}_n$ con $n \in \omega$. Este nuevo sistema quedará descrito de forma efectiva, lo que permite construir los sistemas $\mathcal{A}_{\omega+1}$, $\mathcal{A}_{\omega+2}$, … Continuando el proceso de forma análoga a lo largo de la sucesión ordinal transfinita. Si trabajamos con un lenguaje numerable y queremos mantener la efectividad del proceso en la medida de lo posible, debemos restringir nuestra sucesión a los ordinales recursivos. La herramienta más efectiva para asociar a cada ordinal recursivo α un sistema axiomático $\mathcal{A}_\alpha$ resultan ser los sistemas de notación ordinal.

Del trabajo de Church y Kleene en [1] obtenemos el sistema de notación ordinal $\mathcal{O}$ que utiliza números naturales como nombres para ordinales. Entre sus propiedades más significativas se encuentra la maximalidad, es decir, que cualquier ordinal que reciba un nombre mediante un sistema de notación, lo recibe mediante $\mathcal{O}$. El primer ordinal que no tiene una notación en $\mathcal{O}$ es conocido como el ordinal de Church-Kleene y está denotado por ω_1^{CK}.

El teorema más importante en [28] es el siguiente resultado de completud parcial: Si partimos de la aritmética de Peano y realizamos la construcción descrita anteriormente, cualquier fórmula de primer orden que sólo contenga cuantificadores universales y sea verdadera en el modelo estándar será un teorema de $\bigcup_{d\in\mathcal{O}} \mathcal{A}_d$. Este resultado es impresionante y desconcertante al principio. Considerando que incluso la hipótesis de Riemann puede ser expresada mediante una fórmula aritmética con sólo cuantificadores universales [3, pág. 335], el teorema de completud de Turing parece sugerir un camino para resolver uno de los problemas matemáticos más relevantes de la actualidad. Sin embargo, la utilidad práctica de este resultado depende por completo de poder contestar la pregunta: Dado $d \in \mathbb{N}$, ¿$d \in \mathcal{O}$? Esta pregunta resulta ser sumamente difícil de contestar, pues no existe una fórmula de primer orden que describa[1] a $\mathcal{O}$. Por otro lado, Turing esperaba probar un resultado de completud más fuerte que incluyera fórmulas de la forma $\forall\exists$, esto amplificaría el alcance de su resultado para incluir a problemas como la conjetura de los primos gemelos, pero no lo consiguió.

A pesar de la originalidad y variedad de ideas estímulantes, la tesis doctoral de Turing se encuentra entre sus trabajos menos conocidos. En [9], Feferman conjetura que una de las razones de la baja recepción del trabajo es el uso del cálculo-λ como modelo preferido de computabilidad. Pasaron 20 años de su introducción antes de que reviviera el estudio de las ideas propuestas por Turing y se resolviera la conjetura planteada sobre la completud para fórmulas de la forma $\forall\exists$. En su artículo *Transfinite progressions of axiomatic theories* [7], Feferman rebautizó las lógicas ordinales de Turing como *progresiones transfinitas de sistemas axiomáticos*, modernizó su tratamiento traduciendo la construcción a la teoría de las funciones recursivas y resolvió negativamente la conjetura de Turing.

Actualmente, la literatura dirigida al estudio de las progresiones recursivas se limita a los trabajos ya mencionados de Turing y Feferman, y a textos más avanzados como [12] de Torkel Franzén. Sin embargo, existe un vacío en textos dirigidos a ser un primer contacto con las ideas planteadas originalmente por Turing, con

[1]Los detalles de exactamente qué significa esto esto, así como una prueba, pueden encontrarse en [23].

un tratamiento moderno y poniendo las construcciones teóricas en contraste con las motivaciones históricas y filosóficas que las originaron. La tesis doctoral de Turing, aunque brillante en contenido, padece de imprecisiones en sus definiciones y tratamiento. Por ejemplo, la problemática de asignar enunciados de consistencia a teorías arbitrarias de manera uniforme y su naturaleza intencional no había sido investigada, por lo que Turing dejó ese detalle pendiente. Por otro lado, el artículo de Feferman destaca por su precisión y generalidad; sin embargo, su interés está centrado en responder problemas abiertos y tratar con precisión las ideas introducidas por Turing, no en presentarlas a un público ajeno a ellas.

El objetivo de este trabajo es servir de introducción al estudio de las progresiones recursivas de sistemas axiomáticos con una prueba detallada del teorema de completud de Turing. Seguiremos el enfoque propuesto por Feferman en [7], pero nos limitaremos a estudiar las progresiones definidas mediante la adición de enunciados de consistencia. Estas son las más interesantes para nuestro objetivo por la naturalidad en su planteamiento y porque generalizar a partir de ellas a otro tipo de progresiones es un proceso relativamente sencillo. Además, dedicaremos un capítulo al análisis de la intenciones y motivaciones filosóficas de Turing en [28], así como de la relación de ese trabajo con otros de sus artículos más importantes.

En el capítulo 1 presentaremos los preliminares de teoría de la recursión y teoría de la demostración necesarios para el desarrollo del trabajo, así como la aritmetización de la metamatemática. También, daremos un método para asignar enunciados de consistencia a teorías arbitrarias de forma recursiva, partiendo de descripciones de un conjunto de axiomas en lugar del conjunto de axiomas propiamente. También, discutiremos la diferencia entre definiciones intencionales y extensionales. En partícular, veremos que utilizar definiciones que no reflejan la intención original pero son numéricamente correctas puede derivar en resultados sumamente extraños.

En el capítulo 2 estudiaremos los sistemas de notación ordinal. Específicamente, nos enfocaremos en la definición del sistema $\mathcal{O}$ de Kleene y la prueba de sus propiedades básicas, pues este será la médula espinal de la construcción de las progresiones recursivas de sistemas axiomáticos.

En el capítulo 3 daremos la definición formal de progresión de consistencia de sistemas axiomáticos y probaremos el teorema de completud de Turing. También, daremos una breve descripción de las limitaciones y generalizaciones de este enfoque.

Finalmente, en el capítulo 4 daremos el contexto necesario para dar sentido a los aspectos técnicos del trabajo mediante el planteamiento de las motivaciones filosóficas que llevaron a Turing a su tesis doctoral. Más aún, revisaremos el desarrollo de muchas de sus ideas sobre la inteligencia, las máquinas y la computabilidad a lo largo de varios de sus artículos, con el objetivo de entender el lugar de sus lógicas ordinales en el contexto de la totalidad de sus trabajo.

Concluimos esta introducción con una palabra de advertencia: los primeros tres capítulos de este trabajo pueden ser vistos de la misma forma que una hilera de fichas de domino, donde cada una debe ser colocada con cuidado y precisión. Sólo cuando todas las piezas han sido acomodadas de forma correcta podemos disfrutar de un bello –aunque efímero– resultado. Si el lector siente –en algún punto de la lectura– que se ha perdido entre detalles y tecnicismos, debería dirigirse al último capítulo que cumple la función de ser un mapa de las ideas que se esconden en los aspectos matemáticos y que provee de luz a nuestras intenciones.

Índice general

1 PRELIMINARES

1.1. Preliminares matemáticos

A lo largo de este trabajo, cualquier variable para la cual no se especifique el conjunto sobre el que varía, será considerada como variando sobre el conjunto de los números naturales $\omega = \{0,1,2,3...\}$. Utilizaremos conceptos básicos de teoría de conjuntos y aritmética con sus significados usuales como los de función, relación y las operaciones básicas de conjuntos y números naturales. Además supondremos familiaridad con el concepto de *número ordinal* que puede ser revisado en [17], a la clase de los números ordinales la llamaremos OR. También, consideraremos conocida la teoría básica de funciones recursivas primitivas y de funciones recursivas en general. Las referencias utilizadas en este aspecto serán los libros de Kleene [19] y Rogers [23]. El mismo libro de Kleene, junto con los libros de Enderton [5] y Shoenfield [24] servirán en la consulta de los aspectos lógico-matemáticos de la discusión. Más aún, cualquier definición omitida en este trabajo puede ser buscada en [19]. Utilizaremos el símbolo $\rightleftharpoons$ para hacer asociaciones informales entre conceptos, para introducir abreviaturas y para asignar interpretaciones cotidianas a expresiones de un lenguaje formal, este símbolo no debe ser entendido como parte de una afirmación matemática formal, sino como un elemento informal para la exposición intuitiva de las ideas.

Algunas funciones recursivas primitivas de especial interés serán la función $\mathcal{P}$ que enumera los números primos respetando el orden usual de los naturales, escribiremos[1] p_k en vez de $\mathcal{P}(k)$; la función $(m)_k$ donde, para cualesquiera $m, k > 0$, $(m)_k$ representa al único $n \in \omega$ tal que $p_k^n \mid m$ y $p_k^{n+1} \nmid m$ y la función ℓ donde $\ell(m)$ denota el mayor $n \in \omega$ tal que $p_n \mid m$ cuando $m > 0$ y $\ell(1) = 0$. Diremos que un número natural m *representa una secuencia* si y sólo si $m \neq 0$ y si $p_k \mid m$, entonces $p_j \mid m$ para todo $j < m$. Esta última definición cobra sentido cuando observamos que para todo número m que representa una secuencia, podemos asociarle una su-

[1] $p_1 = 2, p_2 = 3, p_3 = 5, ...$ Podemos definir $p_0 = 1$ para obtener la totalidad de la función.

cesión[2] de números naturales distintos de cero $(m)_1, ..., (m)_{\ell(m)}$. El conjunto de los números naturales que representan alguna secuencia es recursivo primitivo.

Con respecto a las funciones recursivas, adoptaremos la *tesis de Church-Turing* a lo largo de este trabajo. Supondremos que tenemos una enumeración efectiva de las funciones recursivas y denotaremos por φ_y^k a la y-ésima función recursiva de aridad k respecto a esta enumeración. Llamaremos a y el *índice* de la función φ_y^k y nuestra enumeración es tal que a partir del índice podemos obtener –de forma efectiva– la función φ_y^k. El superíndice lo ignoraremos cuando la aridad sea uno, o cuando no haya confusión sobre ésta en el contexto. Es importante notar que esta enumeración no es inyectiva. Es decir, cada función tiene no sólo más de un índice, sino una infinidad de ellos[3]. Si φ es una función recursiva diremos que $\varphi(x)$ converge cuando x está en el dominio de φ y que diverge si no. También, será importante tener en consideración la función recursiva primitiva S_n^{m+1} que cumple:

$$\varphi_{S_n^{m+1}(z, y_0, ..., y_{m-1})}(x_0, \ldots, x_{n-1}) = \varphi_z(y_0, \ldots, y_{m-1}, x_0, \ldots, x_{n-1})$$

Dos teoremas que ocupan un lugar sumamente importante en el desarrollo de este trabajo y en la teoría de la recursión, son el teorema de la forma normal de Kleene y el teorema de la recursión, también establecido por Kleene. A continuación enunciamos estos teoremas, incluyendo dos versiones del teorema de la recursión que serán de utilidad más adelante. Las demostraciones pueden encontrarse en [23, Caps. 1,11].

Teorema 1.1 (Forma normal de Kleene)**.** *Para todo $k \in \omega$, existen dos funciones recursivas primitivas f y g, de aridades 1 y $k+2$ respectivamente, tales que, para toda $z \in \omega$,*

$$\varphi_z^k(x_0, ..., x_{k-1}) = f\left(\mu y \left[g\left(z, x_0, ..., x_{k-1}, y\right) = 1\right]\right)$$

Donde μ es el operador de minimalización usual.

Teorema 1.2 (De la Recursión I)**.** *Si f es una función recursiva, entonces hay $y \in \omega$ tal que*

$$\varphi_y = \varphi_{f(y)}$$

Teorema 1.3 (De la Recursión II)**.** *Sea f una función recursiva de aridad $k+1$. Entonces hay una función recursiva g_f de aridad k tal que para cualesquiera $x_1, ..., x_k$ se cumple:*

$$\varphi_{g_f(x_1, ..., x_k)} = \varphi_{f(g_f(x_1, ..., x_k), x_1, ..., x_k)}$$

[2]Si $\ell(m) = 0$, la sucesión será vacía.

[3]La construcción de esta numeración usualmente se hace para el conjunto de instrucciones asociadas a cada máquina de Turing, sin embargo, muchas máquinas de Turing computan exactamente la misma función. Más detalles al respecto pueden encontrarse en [23].

1.2. Preliminares metamatemáticos

En esta sección estableceremos los elementos básicos sobre lenguajes de primer orden, teoría de la demostración y la aritmética de Peano que serán necesarios a lo largo de este trabajo.

1.2.1. Lenguajes de primer orden

A lo largo de este trabajo, nos fijaremos únicamente en teorías axiomáticas formalizadas en un lenguaje de primer orden. Sin embargo, en aras de la simplicidad, los lenguajes que ocuparemos se basaran en un alfabeto conformado por números y las expresiones se construiran a partir de operaciones aritméticas. Esto nos permitirá identificar cada fórmula del lenguaje con lo que usualmente se considera su número de Gödel, simplificando así el proceso de aritmetización.

Como trabajaremos exclusivamente con teorías de primer orden, sólo necesitaremos utilizar los siguientes símbolos (números):

- $x_n = 18n + 5$ para todo $n \in \omega$.
- $c_n = 18n + 7$ para todo $n \in \omega$.
- $f_n^m = 18\left[\frac{(n+m)(n+m+1)}{2} + n\right] + 9$ para todo $n \in \omega$ y todo $m \in \omega \setminus \{0\}$.
- $r_n^m = 18\left[\frac{(n+m)(n+m+1)}{2} + n\right] + 11$ para todo $n \in \omega$ y todo $m \in \omega \setminus \{0\}$.

Donde x_n representa la $n+1$-ésima variable, c_n la $n+1$-ésima constante, f_n^m la $n+1$-ésima letra funcional de aridad m y r_n^m la $n+1$-ésima letra relacional de aridad m. Es importante notar que no hay letras funcionales ni relacionales de aridad cero, es decir, $m > 0$. Sin embargo, en ciertas ocasiones nos referiremos a las constantes como letras funcionales de aridad cero, pues esto permite una exposición más completa. Llamaremos $\mathcal{VAR}$, $\mathscr{C}$, $\mathscr{F}$ y $\mathscr{R}$ a los conjuntos de variables, constantes, letras funcionales y letras relacionales relacionales, respectivamente.

Así, como los símbolos de nuestro lenguaje son números, utilizaremos operaciones recursivas para construir las expresiones. Este enfoque nos permite omitir la inclusión a nuestro lenguaje de símbolos especiales para los operadores lógicos y signos ortográficos, sin embargo, es útil pensar que estos símbolos están asociados con ciertos número naturales de la misma forma que en el proceso usual de asignación de números de Gödel. Para ser específicos, podemos establecer la siguiente correspondencia:

- $\neg \rightleftharpoons 1$
- $\rightarrow \rightleftharpoons 3$
- $\forall \rightleftharpoons 9$
- $\approx \rightleftharpoons 11$
- $(\rightleftharpoons 13$
- $) \rightleftharpoons 15$
- $, \rightleftharpoons 17$

De esta forma, podemos definir las operaciones $\approx : \omega \times \omega \to \omega$, $\neg : \omega \to \omega$, $\to : \omega \times \omega \to \omega$, $\forall x_m : \omega \to \omega$, $F_n^{m+1} : \omega^{m+1} \to \omega$, $R_n^m : \omega^m \to \omega$ para cualesquiera $n, m \in \omega$ con $m > 0$ de manera que las expresiones resultantes coincidan con los números de Gödel de sus equivalentes en el tratamiento usual. Es decir, en vez de que una expresión sea una cadena de símbolos, cada expresión será un número cuya descomposición en números primos representa una secuencia y de tal forma que los exponentes de los números primos formen la cadena de símbolos. Por ejemplo, el número de Gödel de la expresión $(x_0 \approx x_0)$ sería $2^{13} \cdot 3^5 \cdot 5^{11} \cdot 7^5 \cdot 11^{15}$. Podemos recuperar esto y ahorrarnos el paso de asignar números de Gödel construyendo las expresiones mediante las siguientes operaciones recursivas. Sean $n, m \in \omega$ con $m > 0$ y $a, b, a_1, a_2, ..., a_m \in \omega$

- $\approx (a, b) = 2^{13} \cdot 3^a \cdot 5^{11} \cdot 7^b \cdot 11^{15}$
- $\neg(a) = 2^{13} \cdot 3^1 \cdot 5^a \cdot 7^{15}$
- $\to (a, b) = 2^{13} \cdot 3^a \cdot 5^3 \cdot 7^b \cdot 11^{15}$
- $\forall x_n(a) = 2^{13} \cdot 3^9 \cdot 5^{x_n} \cdot 7^a \cdot 11^{15}$
- $F_n^{m+1}(a_1, \ldots, a_m, a) = p_1^{f_n^m} \cdot p_2^{13} \cdot p_3^{a_1} \cdot p_4^{17} \cdot ... \cdot p_{2m}^{17} \cdot p_{2m+1}^{a_m} \cdot p_{2(m+1)}^{15} \cdot p_{2(m+1)+1}^{11} \cdot p_{2(m+2)}^{a}$
- $R_n^m(a_1, \ldots, a_m) = p_1^{r_n^m} \cdot p_2^{13} \cdot p_3^{a_1} \cdot p_4^{17} \cdot ... \cdot p_{2m}^{17} \cdot p_{2m+1}^{a_m} \cdot p_{2(m+1)}^{15}$

En adelante, escribiremos $(a \approx b)$, $(\neg a)$, $(a \to b)$ y $\forall x_n a$ en vez de $\approx (a, b)$, $\neg(a)$, $\to (a, b)$ y $\forall x_n(a)$, respectivamente.

Observación 1.4. Los operadores lógicos $\vee$, $\wedge$, $\longleftrightarrow$ y $\exists x_k$ pueden definirse como operaciones aritméticas a partir de los otros de forma usual.

Definición 1.5. Decimos que τ es un *tipo de semejanza* o *signatura* si y sólo si $\tau \subseteq \mathscr{F} \cup \mathscr{C} \cup \mathscr{R}$.

Definición 1.6. Sea τ un tipo de semejanza. Definimos el *conjunto de términos del tipo* τ, $\mathcal{T}rm_\tau$, el *conjunto de fórmulas átomicas del tipo* τ, $\mathcal{A}tm_\tau$, y el *conjunto de fórmulas del tipo* τ, $\mathscr{L}_\tau$, de la siguiente forma:

(I) $\mathcal{T}rm_\tau$ es el $\subset$-menor subconjunto de ω que cumple:

a) Contiene a $VAR \cup \mathscr{C}$.

b) Es cerrado bajo F_n^m, para cualesquiera $n, m \in \omega$, con $m > 0$.

(II)

$$\begin{aligned} \mathcal{A}tm_\tau = & \left\{R_n^m(t_1, ..., t_m) | t_1, ..., t_m \in \mathcal{T}rm_\tau, n \in \omega, m > 0\right\} \\ & \cup \left\{(t_1 \approx t_2) | t_1, t_2 \in \mathcal{T}rm_\tau\right\} \end{aligned}$$

(III) $\mathscr{L}_\tau$ es el $\subset$-menor conjunto de ω que cumple:

a) Contiene a $\mathcal{A}tm_\tau$ como subconjunto.

b) Es cerrado bajo $\neg$, $\rightarrow$ y $\forall x_k$ para toda $k \in \omega$.

Es importante notar que, debido a nuestra construcción de $\mathcal{T}rm_\tau$, $\mathcal{A}tm_\tau$ y $\mathscr{L}_\tau$, estos conjuntos son recursivos y cada término y fórmula del tipo τ se escribe de forma única. Si $\theta \in \mathscr{L}_\tau$, en ocasiones escribiremos $\theta(x_k)$ ó $\theta(x_{i_1}, ..., x_{i_n})$ significando exactamente lo mismo que θ, sin que esto implique relación alguna entre θ y las variables. Dicho esto, utilizaremos esta notación para resaltar la presencia de ciertas variables en una fórmula.

De esta forma se puede continuar el desarrollo de la sintaxis de nuestros lenguajes de primer orden estrictamente dentro de la aritmética y definir las nociones de *variable libre en una fórmula*, de *presencia de una variable en un término*, el *conjunto* $VL(\theta)$ *de variables libres en una fórmula* θ, el *conjunto* $VP(t)$ *de variables que figuran en un término* t y el *conjunto* $\mathscr{L}_\tau^n$ *de fórmulas del tipo* τ *con* n *variables libres*. También, es posible definir la *substitución simultánea de variables por términos en una fórmula o término del tipo* τ de tal forma que si $t_0, ..., t_n \in \mathcal{T}rm_\tau$ y $\theta \in \mathscr{L}_\tau \cup \mathcal{T}rm_\tau$, $\theta(t_1, ...t_n)$ representa el resultado de sustituir en θ de forma simultánea x_0 por $t_0, ..., x_n$ por t_n. Esta operación siempre podrá ser realizada, renombrando variables acotadas si es necesario para evitar el problema de captura por cuantificadores de variables libres.

1.2.2. Aspectos de teoría de la demostración

A continuación desarrollaremos algunos conceptos básicos de teoría de la demostración necesarios para el presente trabajo, empezando por la definición de sistema axiomático. En adelante, sea τ un tipo de semejanza.

Definición 1.7. Decimos que $\mathcal{A} = \langle \tau, A \rangle$ es un *sistema axiomático*[4] si y sólo si $A \subseteq \mathscr{L}_\tau^0$.

[4] La definición usual de sistema axiomático requiere también de la determinación de un conjunto de reglas de inferencia. En nuestro caso, las únicas reglas de inferencia en todos los sistemas axiomáticos serán modus ponens y generalización sobre variables, por lo que no es necesario incluirlas en este momento.

Si $\mathcal{A} = \langle \tau, A \rangle$ es un sistema axiomático y $\Gamma \subseteq \mathscr{L}^0_\tau$, denotaremos por $\mathcal{A} + \Gamma$ al sistema axiomático $\langle \tau, A \cup \Gamma \rangle$

En general, es importante diferenciar entre los axiomas lógicos y los axiomas específicos de cada teoría, mientras que el conjunto A en la definición anterior servirá para representar los últimos, el conjunto Λ_τ que definiremos a continuación ocupará el papel de los primeros.

Definición 1.8. Definimos el *conjunto de axiomas lógicos del tipo* τ, Λ_τ, como el conjunto de las fórmulas φ del tipo τ que cumplen con alguna de las siguientes afirmaciones:

(I) $\varphi = (\theta \to (\psi \to \theta))$

(II) $\varphi = ((\theta \to (\psi \to \zeta)) \to ((\theta \to \psi) \to (\theta \to \zeta)))$

(III) $\varphi = ((\neg\theta \to \neg\psi) \to ((\neg\theta \to \psi) \to \theta))$

(IV) $\varphi = \forall x_k (\theta (x_k) \to \theta(t))$

(V) $\varphi = \big(\forall x_j (\theta \to \psi)\big) \to \big(\theta \to \forall x_j (\psi)\big)$

Con $\theta, \psi, \zeta \in \mathscr{L}_\tau$, $t \in \mathcal{T}rm_\tau$, $k, j \in \omega$ y , en (V), $x_j \notin VL(\theta)$.

Definición 1.9. Decimos que $\xi \in \omega$ es una *prueba de* θ *en un sistema axiomático* $\mathcal{A} = \langle \tau, A \rangle$ si y sólo si se cumplen los siguientes:

- ξ es una secuencia.
- $\theta = (\xi)_{\ell(\xi)}$, es decir, la última fórmula en la sucesión representada por ξ es θ.
- Para toda $i < \ell(\xi)$, $(\xi)_i \in \mathscr{L}_\tau$ y ocurre alguno de los siguientes casos:
  - $(\xi)_i \in \Lambda_\tau$
  - $(\xi)_i \in A$
  - Hay $j, k < i$ tales que $(\xi)_k = (\xi)_j \to (\xi)_i$
  - Hay $j < i$ y $m \in \omega$ tal que $(\xi)_i = \forall x_m \Big((\xi)_j\Big)$

Si ξ es una prueba de θ en $\mathcal{A}$, escribiremos $\mathtt{Pba}_{\mathcal{A}}(\xi, \theta)$.

Llamaremos $\mathtt{Teo}_{\mathcal{A}}$ al *conjunto de teoremas de* $\mathcal{A}$, es decir, al conjunto de fórmulas $\theta \in \mathscr{L}_\tau$ para las cuales hay una prueba en $\mathcal{A}$. Escribiremos $\vdash_{\mathcal{A}} \theta$ en vez de $\theta \in \mathtt{Teo}_{\mathcal{A}}$.

Observación 1.10. La relación binaria $\mathtt{Pba}_{\mathcal{A}}$ es recursiva siempre que A lo es y tanto $\mathtt{Pba}_{\mathcal{A}}$ como $\mathtt{Teo}_{\mathcal{A}}$ son recursivamente enumerables cuando A lo es [20, Cap. 10].

Definición 1.11. Sean $\theta, \zeta \in \mathscr{L}_\tau$. Decimos que θ es *lógicamente equivalente* a ζ, en símbolos $\theta \equiv \zeta$, si y sólo si $\vdash_{\mathcal{A}} (\theta \leftrightarrow \zeta)$ para todo sistema axiomático $\mathcal{A}$.

Esta definición coincide con la noción usual de equivalencia lógica, por lo que define una relación de equivalencia sobre $\mathscr{L}_\tau$.

Teniendo ya definida una noción de prueba, podemos continuar con las definiciones usuales de teoría de la demostración:

Definición 1.12. Sean $\mathcal{A} = \langle \tau, A \rangle$, $\mathcal{B} = \langle \rho, B \rangle$ sistemas axiomáticos.

- Decimos que $\mathcal{A}$ es *cerrado bajo deducción* si y sólo si $\mathrm{Teo}_{\mathcal{A}} \subseteq A$.
- Decimos que $\mathcal{A}$ es *consistente* si y sólo si hay $\theta \in \mathscr{L}_\tau$ tal que $\nvdash_{\mathcal{A}} \theta$.
- Decimos que $\mathcal{A}$ es *completo* si y sólo si para toda $\theta \in \mathscr{L}^0_\tau$, $\vdash_{\mathcal{A}} \theta$ ó $\vdash_{\mathcal{A}} (\neg\theta)$.
- Decimos que $\mathcal{A}$ es *finitamente axiomatizable*, *recursivamente axiomatizable* o *recursivamente enumerable* si hay $\mathcal{C} = \langle \tau, C \rangle$ tal que $\mathrm{Teo}_{\mathcal{A}} = \mathrm{Teo}_{\mathcal{C}}$ y C es finito, recursivo o recursivamente enumerable; respectivamente.
- Decimos que $\mathcal{B} = \langle \rho, B \rangle$ es una ρ-*extensión* de $\mathcal{A} = \langle \tau, A \rangle$ y que $\mathcal{A}$ es un τ-*subsistema* de $\mathcal{B}$, en símbolos $\mathcal{A} \subseteq \mathcal{B}$, si y sólo si $\tau \subseteq \rho$ y $\mathrm{Teo}_{\mathcal{A}} \subseteq \mathrm{Teo}_{\mathcal{B}}$. Cuando $\tau = \rho$ diremos únicamente extensión o subsistema.
- Definimos $\mathcal{A} \cup \mathcal{B} = \langle \tau \cup \rho, A \cup B \rangle$

Ejemplo 1.12.1. El sistema axiomático $\mathcal{A} = \langle \varnothing, \{(x_0 \approx x_0), \neg(x_0 \approx x_0)\} \rangle$ es inconsistente, completo y finitamente axiomatizable. Para ver esto, sólo hace falta probar que cualquier fórmula es derivable a partir de una contradicción.

Ejemplo 1.12.2. Sea $\tau = \{c_0\}$. El sistema axiomático $\mathcal{B} = \langle \tau, \{(c_0 \approx c_0) \wedge \neg(c_0 \approx c_0)\} \rangle$ es una τ-extensión del sistema axiomático $\mathcal{A}$ del ejemplo anterior. Sin embargo, el sistema axiomático $\mathcal{C} = \langle \tau, \{\forall x_0 (x_0 \approx x_0)\} \rangle$ no lo es.

A continuación puede desarrollarse de forma usual la teoría alrededor del concepto de prueba. Aquí sólo enunciamos algunos de los resultados más importantes.

Teorema 1.13 (De la Deducción). *Sean* $\theta, \varphi \in \mathscr{L}_\tau$, *entonces*

$$\vdash_{\mathcal{A}+\varphi} \theta \quad \textit{si y sólo si} \quad \vdash_{\mathcal{A}} (\varphi \to \theta)$$

Teorema 1.14 (de Finitud). *Para toda* $\theta \in \mathscr{L}_\tau$,

$$\vdash_{\mathcal{A}} \theta \quad \textit{si y sólo si hay } \mathcal{B} \textit{ un subsistema finito de } \mathcal{A} \textit{ tal que } \vdash_{\mathcal{B}} \theta$$

Lema 1.15 (de Lindenbaum). *Si* $\mathcal{A} = \langle \tau, A \rangle$ *es consistente, entonces existe un sistema axiomático completo y consistente* $\mathcal{B} = \langle \tau, B \rangle$ *tal que* $A \subseteq B$.

1.2.3. La aritmética de Peano

Sea $\mathrm{AP} = \left\{c_0, f_0^1, f_0^2, f_1^2\right\}$. AP es el tipo de semejanza que nos provee de los símbolos necesarios para hablar de aritmética dentro de nuestros lenguajes de primer orden. Diremos que una fórmula $\theta \in \mathscr{L}_{\mathrm{AP}}^0$ es *verdadera* si es verdadera bajo la definición de Tarski en el modelo estándar[5]. Llamaremos *sistema aritmético* a cualquier sistema axiomático cuyo tipo de semejanza tenga contenido a AP. Algunas convencioness que utilizaremos serán sa, $a+b$ y $a \cdot b$ en vez de $f_0^1(a)$, $f_0^2(a,b)$ y $f_1^2(a,b)$, respectivamente. Más aún, introduciremos los símbolos $\not\approx$, $\leqslant$ y $<$ como sigue:

- $a \not\approx b \rightleftharpoons \neg(a \approx b)$
- $a \leqslant b \rightleftharpoons \exists x_k(a + x_k \approx b)$, con x_k la primer variable que no aparece en a ni b.
- $a < b \rightleftharpoons (a \leqslant b) \wedge (a \not\approx b)$

Definición 1.16. Definimos recursivamente el *numeral*[6]de un número natural n, denotado $\overline{n}$, de la siguiente forma:

(I) $\overline{0} = c_0$

(II) $\overline{n+1} = s\overline{n}$

El sistema aritmético que nos será de mayor interés a lo largo de este trabajo será $\mathscr{AP} = \langle \mathrm{AP}, \mathrm{P}\rangle$ usualmente conocido como *aritmética de Peano*. Aquí P denota al conjunto $\{ap_1, ap_2, \ldots, ap_6\} \cup \{ind_n : n \in \omega\}$ donde,

- $ap_1 = \forall x \forall y\,(sx \approx sy \rightarrow x \approx y)$
- $ap_2 = \forall x\left(sx \not\approx \overline{0}\right)$
- $ap_3 = \forall x\left(x + \overline{0} \approx x\right)$
- $ap_4 = \forall x \forall y\,(x + sy \approx s\,(x+y))$
- $ap_5 = \forall x\left(x \cdot \overline{0} = \overline{0}\right)$
- $ap_6 = \forall x \forall y\,(x \cdot sy \approx (x \cdot y) + x)$
- $ind_n = \left(\theta\left(\overline{0}\right) \wedge \forall x\,(\theta\,(x) \rightarrow \theta\,(sx))\right) \rightarrow \forall x\,(\theta\,(x))$ donde θ es la n-ésima fórmula en una enumeración[7] efectiva dada de $\mathscr{L}_{\mathrm{AP}}$.

[5]La discusión sobre exactamente qué es el modelo estándar escapa de los objetivos de este trabajo, pero los detalles de su definición y los antecedentes modelo-teóricos necesarios pueden encontrarse en [5] .

[6]Recordemos que $c_0 \in \omega$, esto significa que utilizaremos números como los nombres dentro del sistema formal para otros números. En particular, $\overline{0} = 25$.

[7]Como $\mathscr{L}_{\mathrm{AP}}$ es recursivo primitivo [20] y numerable , hay una función inyectiva y recursiva primitiva $f : \omega \rightarrow \mathscr{L}_{\mathrm{AP}}$. Así, $\theta = f(n)$.

Un resultado sumamente importante es que cualquier función y relación recursiva puede expresarse mediante una fórmula de $\mathscr{L}_{AP}$. Aclararemos esta noción mediante la siguiente definición.

Definición 1.17. Sean $\mathcal{A}$ un sistema aritmético con tipo de semejanza τ, $R \subseteq \omega^{n+1}$ y $\theta \in \mathscr{L}_{\tau}^{n+1}$ tal que $VL(\theta) = \{x_0, x_1, ..., x_n\}$.

(I) Decimos que θ *representa* a R en $\mathcal{A}$ si y sólo si para cualesquiera $a_0, ..., a_n \in \omega$
$$(a_0, a_1, ..., a_n) \in R \iff \vdash_{\mathcal{A}} \theta(\overline{a_0}, \overline{a_1}, ..., \overline{a_n})$$

(II) Decimos que θ *representa fuertemente* a R en $\mathcal{A}$ si y sólo si θ representa a R en $\mathcal{A}$ y $\neg\theta$ representa a $\omega^{n+1} \setminus R$ en $\mathcal{A}$

Diremos que R es *representable (fuertemente)* en $\mathcal{A}$ si hay una fórmula que lo representa (fuertemente) en $\mathcal{A}$. Cada fórmula $\theta \in \mathscr{L}_\tau$ con $VL(\theta) = \{x_0, ..., x_n\}$ es llamada una *representación* de $\left\{(a_0, ..., a_n) \in \omega^{n+1} | \vdash_{\mathcal{A}} \theta(\overline{a_0}, ..., \overline{a_n})\right\}$.

Observación 1.18. A cada función f de aridad $n+1$ le podemos asociar de forma natural un subconjunto de ω^{n+2}. Así, podemos extender la definición anterior diciendo que una fórmula θ con $VL(\theta) = \{x_1, ..., x_{n+1}\}$ *representa* a f si la representa como relación, es decir, si para cualesquiera $a_0, ..., a_n, a \in \omega$
$$f(a_0, ..., a_n) = a \text{ si y sólo si } \vdash_{\mathcal{A}} \theta(\overline{a_0}, ..., \overline{a_n}, \overline{a})$$
Si θ es de la forma $\varphi(x_0, ..., x_n) = x_{n+1}$, entonces θ representa fuertemente a f en $\mathcal{A}$.

La forma más sencilla de obtener fórmulas que representen funciones recursivas primitivas es haciéndolo en ciertas extensiones de $\mathcal{AP}$ que llamaremos *extensiones recursivas primitivas*. Para construir estas extensiones, expandiremos el tipo de semejanza con nuevos símbolos funcionales y añadiremos nuevos axiomas adecuados. La siguiente definición hace explícito este procedimiento:

Definición 1.19. Diremos que $\mathcal{AP}' = \langle \tau, A \rangle$ es una extensión recursiva primitiva de $\mathcal{AP}$ si y sólo si hay $g_0^{n_0+1}, ..., g_m^{n_m+1} \in \mathscr{F}$ y $\theta_1, ..., \theta_m$ tales que:

(I) Para todo $i \leqslant m$, hay $r, t \in \mathcal{T}rm_{\tau_i}$, con $\tau_i = AP \cup \{g_0, ..., g_i\}$, tales que $VP(r) \subseteq \{x_0, ... x_{n_i-1}\}$, $VP(t) \subseteq \{x_0, ... x_{n_i+1}\}$ y
$$\theta_i = \forall x_0 ... \forall x_{n_i-1} \left(g_i^{n_i+1}\left(x_0, ..., x_{n_i-1}, \overline{0}\right) \approx r(x_0, ..., x_{n_i-1}) \right.$$
$$\left. \wedge \forall x_{n_i}\, g_i^{n_i+1}(x_0, ..., x_{n_i-1}, sx_{n_i}) \approx t\left(x_0, ..., x_{n_i}, g_i^{n_i+1}(x_0, ..., x_{n_i})\right) \right)$$

(II)

$$\begin{aligned}\tau &= \tau_m \quad \text{y}\\ A &= P \cup \{\theta_0, ..., \theta_m\}\\ &\qquad \cup \left\{ \left(\psi\left(\overline{0}\right) \wedge \forall x \left(\psi\left(x\right) \rightarrow \psi\left(sx\right)\right)\right) \rightarrow \forall x \psi\left(x\right) | \psi \in \mathscr{L}_\tau \right\}\end{aligned}$$

El uso de los términos r, t en la descripción de θ_i permiten expresar, en un paso dentro del sistema, la construcción de nuevas funciones a partir de otras dadas mediante composición, substitución y permutación de variables.

Ejemplo 1.19.1. Para aclarar el significado de la definición anterior, construiremos una extensión recursiva primitiva $\mathscr{AP}'$ de $\mathscr{AP}$ en la que podamos representar, de forma inmediata, la función exponencial $exp : \omega \times \omega \rightarrow \omega$. Escribiremos n^m en lugar de $exp(n, m)$. La exponencial se define por recursión de la siguiente forma:

(I) $n^0 = 1$ para todo $n \in \omega$.

(II) $n^{m+1} = n \cdot n^m$.

Necesitaremos una nueva letra funcional g_0^2 de aridad 2 para representar a la función exponencial en $\mathscr{AP}'$ y el axioma θ_0 que nos dirá, dentro del sistema, como computar sus valores. Es decir, formalizará la definición recursiva anterior.

$$\theta_0 = \forall x_0 \ \left(\left(g_0^2\left(x_0,\overline{0}\right) \approx \overline{1}\right) \wedge \forall x_1 \ \left(g_0^2\left(x_0, sx_1\right) \approx \left(x_0 \cdot g_0^2\left(x_0, x_1\right)\right)\right)\right)$$

De esta forma, si definimos

$$A' = P \cup \{\theta_0\} \cup \left\{ \left(\psi\left(\overline{0}\right) \wedge \forall x \left(\psi\left(x\right) \rightarrow \psi\left(sx\right)\right)\right) \rightarrow \forall x \psi\left(x\right) | \psi \in \mathscr{L}_\tau \right\}$$

el sistema axiomático $\mathscr{AP}' = \left\langle AP \cup \{g_0^2\}, A' \right\rangle$ es una extensión recursiva primitiva de $\mathscr{AP}$. Los términos r y t que aparecen en la definición anterior corresponden –respectivamente– a los términos $\overline{1}$ y $\left(x_0 \cdot g_0^2(x_0, x_1)\right)$ en este caso particular. Así, para cualesquiera $a, b \in \omega$, se cumple

$$\vdash_{\mathscr{AP}'} g_0^2(\overline{a}, \overline{b}) \approx \overline{c} \quad \text{si y sólo si} \quad a^b = c$$

Gracias a la definición anterior, podemos obtener el siguiente resultado sobre representabilidad de funciones y relaciones recursivas primitivas. La prueba puede encontrarse en [19].

Teorema 1.20 (Representación de funciones y relaciones recursivas primitivas)**.**

(I) *Si* f *es una función recursiva primitiva de aridad* $n+1$*, entonces hay una* τ*-extensión recursiva primitiva* $\mathscr{AP}'$ *de* $\mathscr{AP}$ *y un término* t *del tipo* τ *con* $\mathsf{VL}(t) = \{x_0, ..., x_n\}$ *tales que* $t(x_0, ..., x_n) \approx x_{n+1}$ *representa a* f *en* $\mathscr{AP}'$ *(si* $\mathscr{AP}'$ *es consistente). Más aún, cualquier fórmula de la forma* $t(x_0, ..., x_n) \approx x_{n+1}$ *representa a una función recursiva primitiva en* $\mathscr{AP}'$*.*

(II) *Si* R *es una relación recursiva primitiva de aridad* $n+1$*, entonces hay una* τ*-extensión recursiva primitiva* $\mathscr{AP}'$ *de* $\mathscr{AP}$ *y un término* t *del tipo* τ *con* $\mathsf{VL}(t) = \{x_0, ..., x_n\}$ *tales que* $t(x_0, ..., x_n) \approx \overline{0}$ *representa fuertemente a* φ *en* $\mathscr{AP}'$ *(si* $\mathscr{AP}'$ *es consistente). Más aún, cualquier fórmula de la forma* $t(x_0, ..., x_n) \approx \overline{0}$ *representa fuertemente a una relación recursiva primitiva en* $\mathscr{AP}'$*.*

Las representaciones de funciones y relaciones recursivas primitivas también pueden obtenerse directamente en $\mathscr{AP}$ utilizando la llamada *función* β presentada por Gödel. El método se encuentra descrito con todo detalle por Kleene en [19] por lo que aquí no profundizaremos en la técnica ni presentaremos las demostraciones. Sin embargo, utilizaremos lo siguiente:

Definición 1.21. Para cada τ-extensión recursiva primitiva $\mathscr{AP}'$ de $\mathscr{AP}$ y cualquier fórmula $\theta \in \mathscr{L}_\tau$, hay asociada una fórmula $\theta^{(\mathscr{AP}')} \in \mathscr{L}_{\mathrm{AP}}$ con exactamente las mismas variables libres tal que:

- $(\neg\theta)^{(\mathscr{AP}')} = \neg\left(\theta^{(\mathscr{AP}')}\right)$
- $(\theta \to \zeta)^{(\mathscr{AP}')} = \theta^{(\mathscr{AP}')} \to \zeta^{(\mathscr{AP}')}$
- $(\forall x_k \theta)^{(\mathscr{AP}')} = \forall x_k \left(\theta^{(\mathscr{AP}')}\right)$

Para cualesquiera $\theta, \zeta \in \mathscr{L}_\tau$ y $k \in \omega$. También se cumple que, para toda $\theta \in \mathscr{L}_{\mathrm{AP}}$, $\theta^{(\mathscr{AP}')} = \theta$.

Más aún, tenemos el siguiente resultado.

Teorema 1.22. *Sea* $\theta \in \mathscr{L}_\tau$*. Tenemos que:*

- $\vdash_{\mathscr{AP}'} \theta \leftrightarrow \theta^{(\mathscr{AP}')}$
- *Si* $\theta \in \mathscr{L}_\tau^0$*, entonces*
$$\vdash_{\mathscr{AP}'} \theta \text{ si y sólo si } \vdash_{\mathscr{AP}} \theta^{(\mathscr{AP}')}$$
- $\mathscr{AP}'$ *es consistente si y sólo si* $\mathscr{AP}$ *es consistente.*

Ahora que ya somos capaces de expresar todas las funciones y relaciones recursivas primitivas en la aritmética, definiremos la *jerarquía aritmética*, que jugará un papel sumamente importante en los resultados principales de este trabajo.

Definición 1.23. Sea $\theta \in \mathscr{L}_{AP}$.

(I) Decimos que $\theta \in \Delta_0 = \Pi_0 = \Sigma_0$ si y sólo si hay $\mathcal{AP}'$ una τ-extensión recursiva primitiva de $\mathcal{AP}$ y $t \in \mathcal{T}rm_\tau$ tales que

$$\theta \equiv \left(t \approx \overline{0}\right)^{(\mathcal{AP}')}$$

(II) Decimos que $\theta \in \Sigma_{n+1}$ si y sólo si hay $\zeta \in \Pi_n$ y $v_1, ..., v_m \in \mathcal{VAR}$ tales que

$$\theta \equiv \exists v_1 \ldots \exists v_m \, (\zeta)$$

(III) Decimos que $\theta \in \Pi_{n+1}$ si y sólo si hay $\zeta \in \Sigma_n$ y $v_1, ..., v_m \in \mathcal{VAR}$ tales que

$$\theta \equiv \forall v_1 \ldots \forall v_m \, (\zeta)$$

Observación 1.24. En ocasiones, se define el conjunto Δ_0 como el conjunto de fórmulas de $\mathscr{L}_{AP}$ que sólo contienen cuantificadores acotados. Debido a que las fórmulas del tipo $\left(t \approx \overline{0}\right)^{(\mathcal{AP}')}$ pueden contener cuantificadores existenciales no acotados, nuestro conjunto Δ_0 difiere un poco de la definición usual. Sin embargo, el resto de los conjuntos de la jerarquía permanecen idénticos. Para probarlo, basta observar que Σ_1 es el mismo conjunto en ambas definiciones y que $\Pi_1 = \{\varphi : \neg\varphi \in \Sigma_1\}$.

Dado que cualquier fórmula es equivalente a otra en forma normal prenex [5] (es decir, escrita como una cadena de cuantificadores seguida de una fórmula sin cuantificadores), cualquier fórmula aparece en la jerarquía. Más aún, la prueba de la afirmación anterior revela que la combinación booleana de fórmulas en el mismo estrato de la jerarquía, vuelve a pertenecer al mismo estrato. Por ejemplo, si $\varphi, \psi \in \Sigma_1$, entonces $\neg\varphi, \varphi \to \psi, \varphi \vee \psi, \varphi \wedge \psi \in \Sigma_1$

Un breve análisis del método de Gödel usado en la constucción de las fórmulas θ del tipo $\left(t \approx \overline{0}\right)^{(\mathcal{AP}')}$ permite notar que t y un mínimo $\mathcal{AP}'$ quedan determinados de forma única y pueden obtenerse a partir de θ. Esto nos lleva a concluir que Δ_0 es un conjunto recursivo primitivo, por lo que todos los conjuntos de nuestra jerarquía aritmética lo son.

Antes de presentar el resultado principal concerniente a funciones recursivas y sus representaciones, introduciremos una nueva definición.

Definición 1.25. Decimos que un sistema aritmético $\mathcal{A} = \langle \tau, A \rangle$ es ω-*consistente* si y sólo si para toda $\theta \in \mathscr{L}_\tau$ con $VL(\theta) = \{x\}$ no es el caso que $\vdash_\mathcal{A} \exists x \, \theta(x)$ y $\vdash_\mathcal{A} \neg\theta(\overline{n})$ para toda $n \in \omega$. Diremos que $\mathcal{A}$ es *débilmente* ω-*consistente* si sólo hacemos esa suposición para fórmulas $\varphi \in \Delta_0$.

Teorema 1.26.

(I) *Una relación* $R \subseteq \omega^n$ *es recursiva primitiva si y sólo si hay* $\theta \in \Delta_0$ *tal que* θ *representa fuertemente a* R *en* $\mathcal{AP}$*.*

(II) *Una relación* $R \subseteq \omega^n$ *es recursivamente enumerable si y sólo si hay* $\theta \in \Sigma_1$ *tal que* θ *representa a* R *en* $\mathcal{AP}$*.*

(III) *Los dos puntos anteriores siguen valiendo si sustituimos* $\mathcal{AP}$ *por cualquier extensión* $\mathcal{A}$ *consistente y recursivamente axiomatizable.*

Observación 1.27. En el teorema anterior, la misma fórmula que representa fuertemente a una relación recursiva primitiva en $\mathcal{AP}$ lo hace en $\mathcal{A}$. Sin embargo, en el caso de fórmulas en Σ_1 no es evidente que se extiende más que a las extensiones débilmente ω-consistentes de $\mathcal{AP}$. Aunque sí es posible debilitar la hipótesis y sólo pedir consistencia para $\mathcal{A}$ [4], la misma fórmula en Σ_1 representará diferentes relaciones en $\mathcal{A}$ y en $\mathcal{AP}$ a menos que $\mathcal{A}$ sea débilmente ω-consistente. Gracias al teorema de forma normal para funciones recursivas de Kleene obtenemos lo siguiente.

Teorema 1.28. *Una relación es recursiva si y sólo si es fuertemente representable en* $\mathcal{AP}$*. Más aún, esto se mantiene si sustituimos* $\mathcal{AP}$ *por cualquier extensión recursivamente axiomatizable y consistente.*

Aunque los últimos teoremas sólo están enunciados para relaciones, debido a la observación 1.18 se pueden traducir inmediatamente para funciones.

Finalmente, tenemos el siguiente resultado de completud:

Teorema 1.29 (Teorema de Σ_1-completud)**.** *Sea* $\theta \in \mathscr{L}_{AP}$ *tal que* $\theta \in \Sigma_1$*. Si* θ *es verdadera, entonces* $\vdash_{\mathcal{AP}} \theta$*.*

Este teorema surge a partir de pruebas finitistas de la consistencia de la aritmética de Robinson[8] $\mathcal{Q}$ como las que pueden encontrarse en [24] y [19].

1.3. Aritmetización

El principal interés de este trabajo es estudiar las consecuencias de construir extensiones de un sistema aritmético $\mathcal{A}$ de alguna forma específica, principalmente mediante la adición de enunciados de consistencia. Teniendo esto en cuenta, de ahora en adelante consideraremos un sistema aritmético $\mathcal{A} = \langle \tau, A \rangle$ con τ un tipo de semejanza finito y $P \subseteq A$. Más aún, pediremos que A sea un conjunto recursivamente enumerable y acceso a un concepto de *validez* para los enunciados del tipo τ que cumpla lo siguiente:

- Para toda $\theta \in \mathscr{L}^0_{AP}$, θ es válida si y sólo si es verdadera.

[8] El sistema $\mathcal{Q}$ corresponde a $\mathcal{AP}$ sin el esquema de inducción.

- Para toda $\theta \in \mathscr{L}_\tau$, si $VL(\theta) = \{x\}$, entonces $\forall x\, \theta(x)$ es válida si y sólo si $\theta(\overline{n})$ es válida para todo $n \in \omega$.

- Si $\theta \in A$, entonces θ es válida.

Los criterios que llevan a un matemático a juzgar un conjunto de axiomas como digno de ser utilizado son muy variados y difícilmente formalizables. En buena parte, esto se debe a que, en la práctica matemática cotidiana, la axiomatización es un proceso que llega mucho después de que las teorías se han establecido y los teoremas importantes se han probado. Una posibilidad para el tratamiento formal de esta situación es pedir que el conjunto de axiomas sea consistente. De esta forma, cuando menos, podemos asegurarnos que tiene algún sentido hablar de los teoremas del sistema axiomático, pues al menos una fórmula no es demostrable. A lo largo de este trabajo adoptaremos otro enfoque, pensaremos que cualquier sistema axiomático que utilizemos ha sido juzgado válido bajo algún criterio y que esa validez satisface, cuando menos, las tres condiciones enunciadas anteriormente. Dejaremos para el capítulo 3 la discusión de por qué este enfoque es el más adecuado para nuestros objetivos y de sus ventajas sobre pedir únicamente consistencia a nuestros sistemas axiomáticos.

Como las fórmulas de nuestros lenguajes son, en realidad, números; el estudio de la sintaxis de sistemas de tipo τ se reduce al estudio de ciertas funciones y relaciones recursivas primitivas. Esto quiere decir que la aritmetización de la metateoría de estos sistemas puede llevarse a cabo en una extensión recursiva primitiva de $\mathscr{AP}$, a la cual llamaremos $\mathcal{S}$. Para facilitar la notación, utilizaremos el símbolo $*$ para referirnos a las fórmulas y términos de $\mathcal{S}$ mediante nuestra notación matemática usual. Esto quiere decir que cada vez que encontremos el símbolo $*$ acompañando una expresión, estamos haciendo referencia al término o fórmula que representa dentro del sístema $\mathcal{S}$ a la expresión usual. Así, dadas $u, v, w \in VAR$ podemos construir los siguientes términos y fórmulas en $\mathcal{S}$:

- $u^{*v} \rightleftharpoons$ "u elevado a la potencia v"
- $pr_{*u} \rightleftharpoons$ "el u-ésimo primo"
- $Var_*(u) \rightleftharpoons$ "u es una variable"
- $vr_{*u} \rightleftharpoons$ "la u-ésima variable"
- $(u)_{*v} \rightleftharpoons$ "el exponente de pr_{*v} en u"
- $Sec_*(u) \rightleftharpoons$ "u representa una secuencia"
- $L_*(u) \rightleftharpoons$ "la longitud de la secuencia representada por u"
- $Num_*(u) \rightleftharpoons$ "el u-ésimo numeral"
- $\neg_* u$
- $u \rightarrow_* v$
- $u \leftrightarrow_* v$
- $u \vee_* v$

- $u \wedge_* v$
- $\exists_* u\,(v)$
- $\forall_* u\,(v)$
- $\mathtt{Trm}_{*\tau}(u) \rightleftharpoons$ "$u \in \mathcal{T}rm_\tau$"
- $\mathtt{Atm}_{*\tau}(u) \rightleftharpoons$ "$u \in \mathcal{A}tm_\tau$"
- $\mathtt{Form}_{*\tau}(u) \rightleftharpoons$ "$u \in \mathscr{L}_\tau$"
- $\mathtt{Vl}_*(u,v) \rightleftharpoons$ "$u \in \mathrm{VL}(v)$"
- $\mathtt{Enun}_*(u) \rightleftharpoons$ "$u \in \mathscr{L}^0_\tau$"
- $\mathtt{Sub}^u_v(w) \rightleftharpoons$ "la substitución de v por u en w"
- $\mathtt{SbS}(u,v,w) \rightleftharpoons$ "la substitución simultánea de cada término de la secuencia u por el respectivo término de v en w"
- $\mathtt{Ax}_{*\tau}(u) \rightleftharpoons$ "$u \in \Lambda_\tau$"
- $\mathtt{Ind}_*(u) \rightleftharpoons$"el u-ésimo axioma del esquema de inducción. Es decir, $\mathtt{ind}_u$"

Cada una de las fórmulas enlistadas se construye directamente de un término en $\mathcal{S}$. Por ejemplo, hay un término t del tipo τ tal que $\mathtt{Ax}_{*\tau}(u)$ es una abreviatura para $t(u) = \overline{0}$. También escribiremos $\overline{u}_*$ para $\mathtt{Num}_*(u)$. Para cargar más información en la notación, si suponemos que $\mathtt{L}_*(t) = \overline{n}$, escribiremos

$$\mathtt{Sb}_* \begin{pmatrix} (u)_{*\overline{1}} & \dots & (u)_{*\overline{n}} \\ \\ (t)_{*\overline{1}} & \dots & (t)_{*\overline{n}} \end{pmatrix} (w)$$

en vez de $\mathtt{SbS}(u,v,w)$ y, más aún, $w\left((t)_{*\overline{1}},\dots,(t)_{*\overline{n}}\right)$ en vez de:

$$\mathtt{Sb}_* \begin{pmatrix} vr_{*\overline{1}} & \dots & vr_{*\overline{n}} \\ \\ (t)_{*\overline{1}} & \dots & (t)_{*\overline{n}} \end{pmatrix} (w)$$

El objetivo de estas convenciones sólo es el de imitar lo más posible la notación usual dentro del sistema $\mathcal{S}$. Por ejemplo, si $\theta \in \mathscr{L}_\tau$ y $\mathrm{VL}(\theta) = \{x_0\}$, entonces $\overline{\theta}(\overline{u}_*)$ denota la substitución de la variable x_0 por el numeral de u en θ. Así, $\vdash_{\mathcal{S}} \overline{\theta}(\overline{n}_*) \approx \overline{\theta(\overline{n})}$.

Estas fórmulas y términos nos permiten describir las operaciones lógicas usuales dentro de la aritmética. Sin embargo, las complicaciones comienzan al intentar describir el sistema $\mathcal{A}$ dentro de $\mathcal{S}$.

Dada una fórmula ξ que represente al conjunto de axiomas de $\mathcal{A}$ en $\mathcal{S}$, podemos construir una nueva fórmula $\mathtt{Pba}_\xi(u,v)$ que represente la relación "u es una prueba de v en $\mathcal{A}$" . A partir de ésta, se definen los enunciados $\mathtt{Teo}_\xi(u)$ y $\mathtt{Con}_\xi$ como $\exists x\, \mathtt{Pba}_\xi\,(x,u)$ y $\exists x\, \neg \mathtt{Teo}_\xi(x)$, respectivamente. Sin embargo, distintas representaciones de los axiomas de $\mathcal{A}$ en $\mathcal{S}$ pueden no ser equivalentes, lo que deriva en enunciados de consistencia no equivalentes. Un ejemplo un tanto radical de este problema lo presenta Feferman en [8] a través del siguiente teorema:

Teorema 1.30 (Feferman)**.** *Existe una fórmula* $\pi \in \mathscr{L}_{\mathrm{AP}}$ *que representa fuertemente a* P *en* $\mathcal{AP}$ *tal que*

$$\vdash_{\mathcal{AP}} \mathsf{Con}_{\pi}$$

A primera vista, este resultado parece contradecir el segundo teorema de incompletud; sin embargo, ese no es el caso. La clave en la demostración se encuentra en la libertad que hay en la construcción de la fórmula π. La técnica de Feferman consiste en construir una fórmula $\pi(x)$ que puede interpretarse como ''x está en el conjunto P y $\mathsf{P} \cap \{0, \ldots, x\}$ es consistente''. Como P es consistente, la fórmula π representa fuertemente a P en $\mathcal{AP}$ y permite probar –dentro de $\mathcal{AP}$– la fórmula Con_{π}. Este resultado no contradice el segundo teorema de incompletud, pues otras propiedades sobre derivabilidad no pueden obtenerse para la fórmula Pba_{π} como lo son las tres propiedades de derivabilidad establecidas por Hilbert y Bernays.

El resultado de Feferman señala la importancia de un método específico para trabajar con enunciados de consistencia de teorías con infinitos axiomas. En las teorías que utilizamos usualmente (la aritmética, por ejemplo) tenemos formulaciones canónicas de los axiomas, lo que nos brinda un enunciado canónico de consistencia. Por otro lado, cuando trabajamos con sistemas axiomáticos arbitrarios no tenemos dicha formulación canónica. Afirmaciones como el primer teorema de incompletud de Gödel no generan ningún problema, podemos enunciarlo y probarlo utilizando cualquier fórmula que represente al conjunto de axiomas del sistema $\mathcal{A}$. A propiedades de este tipo les llamamos *extensionales*. En contraste, hay situaciones en las que una fórmula describe de forma numéricamente correcta un conjunto de axiomas (es decir, lo representa) pero falla en expresar de forma adecuada las nociones que se trabajan. Rescatando el ejemplo presentado por el teorema anterior, la fórmula π representa fuertemente al conjunto P; sin embargo, no refleja de forma adecuada la noción intuitiva de ''ser un elemento del conjunto de axiomas''. Llamamos *intensionales* a las propiedades, situaciones o afirmaciones que requieren fórmulas que además de representar de forma numéricamente correcta cierto conjunto o relación, también expresen de forma adecuada las definiciones involucradas.

Afortunadamente, no todo son malas noticias. En [8], Feferman probó que si nos limitamos a utilizar fórmulas Σ_1 como descripciones de conjuntos de axiomas, se puede demostrar el segundo teorema de incompletud con bastante generalidad. Por otro lado, enfocarnos en las descripciones de los axiomas en vez de los axiomas mismos permite definir los enunciados de consistencia para cada descripción mediante un método uniforme, evitando así la necesidad de elegir uno caso por caso. Cabe mencionar que incluso con la restricción a fórmulas Σ_1 podemos obtener descripciones que no sean intencionalmente correctas, esto jugará un papel significativo en la demostración del teorema de completud de Turing.

Para aterrizar estas ideas, utilizaremos las siguientes definiciones:

Definición 1.31. Decimos que una fórmula aritmética Σ_1 con una variable libre $\xi(x)$ describe a un conjunto de axiomas A si y sólo si

$$\xi(\bar{\phi}) \text{ es verdadera si y sólo si } \phi \in A$$

Definición 1.32. Si $\xi(x)$ es una fórmula aritmética y Σ_1 con una variable libre, definimos la descripción de ξ, denotada por $\widehat{\xi}$, como:

$$\widehat{\xi} = \{n \in \omega : \ \vdash_{AP} \xi(\bar{n})\}$$

Las dos definiciones anteriores juegan papeles íntimamente relacionados que cobran sentido al examinarse con el siguiente enfoque: Si partimos de un sistema axiomático recursivamente enumerable $\mathcal{A}\langle\tau, A\rangle$, podemos construir una fórmula $\xi \in \Sigma_1$ que describa al conjunto de axiomas A. Esta fórmula nos permite hablar sobre el sistema axiomático $\mathcal{A}$ dentro de $\mathcal{AP}$ y –por extensión– de $\mathcal{S}$. Sin embargo, como mencionamos anteriormente, hay muchas fórmulas que describen al conjunto A y no son equivalentes, por lo que en ocasiones será más conveniente partir de la fórmula ξ y no del sistema axiomático $\mathcal{A}$. Así, cuando tenemos una fórmula $\xi \in \Sigma_1$ (sin importar exactamente como la obtuvimos) podemos asociarle un único conjunto $\widehat{\xi}$ recursivamente enumerable al cual describe. El conjunto $\widehat{\xi}$ no corresponde necesariamente a un conjunto de axiomas, sin embargo, sólo nos preocuparemos por los casos en que esto sí ocurra. De esta forma, podemos reconstruir un sistema axiomático[9] a partir de la fórmula ξ.

Ejemplo 1.32.1. El conjunto de axiomas de la aritmética de Peano es descrito por la fórmula

$$\begin{aligned} ap(x) = &(x \approx \overline{ap_1}) \vee (x \approx \overline{ap_2}) \vee \cdots \vee (x \approx \overline{ap_6}) \\ &\vee \left(\exists y \left(x \approx \overline{Ind_*(y)}\right)\right) \end{aligned}$$

Así, $\widehat{ap} = P$. A la fórmula $ap(x)$ la llamamos la *descripción estándar* de la aritmética de Peano.

Ahora sí nos encontramos en condiciones de dar explícitamente las fórmulas que representan las nociones de prueba, teorema y consistencia de un sistema axiomático en $\mathcal{S}$

Definición 1.33. Sea ξ una descripción de los axiomas de $\mathcal{A}$, definimos[10]:

- $Pba_\xi(u,v) = Sec_*(u) \wedge (L_*(u) \not\approx \bar{0}) \wedge \forall w(w < L_*(u) \rightarrow Form_{*\tau}((u)_{*w})$
  $\wedge (Ax_{*\tau}((u)_{*w}) \vee \xi((u)_{*w}) \vee \exists z \exists t((z < w) \wedge (t < w)$
  $\wedge ((u)_{*z} \approx (u)_{*t} \rightarrow_* (u)_{*w}))) \wedge (v \approx (u)_{*L_*(u)})$

[9]En cuanto al tipo de semejanza, basta conformarlo por las constantes, letras funcionales y letras relacionales que figuren en alguna fórmula del conjunto $\widehat{\xi}$

- $\mathtt{Teo}_\xi(\nu) = \exists u \mathtt{Pba}_\xi(u,\nu)$
- $\mathtt{Con}_\xi = \exists u(\mathtt{Form}_{*\tau}(u) \wedge \neg \mathtt{Teo}_\xi(u))$

Observación 1.34. Estas fórmulas son traducciones directas de las definiciones de prueba, teorema y consistencia en 1.9 y 1.12.

En vista de la construcción de los enunciados $\mathtt{Pba}_\xi$, $\mathtt{Teo}_\xi$ y $\mathtt{Con}_\xi$ a partir de la fórmula ξ, tenemos funciones recursivas primitivas f, g y h tales que $\varphi(\xi) = \mathtt{Pba}_\xi$, $\psi(\xi) = \mathtt{Teo}_\xi$ y $\zeta(\xi) = \mathtt{Con}_\xi$. Así, podemos añadir representaciones de estas funciones a $\mathcal{S}$. Es decir, para cualquier fórmula $\xi \in \mathscr{L}_{AP}$, $\mathtt{Pba}_*(\overline{\xi})$, $\mathtt{Teo}_*(\overline{\xi})$ y $\mathtt{Con}_*(\overline{\xi})$ son términos de $\mathcal{S}$ que representan a $\mathtt{Pba}_\xi$, $\mathtt{Teo}_\xi$ y $\mathtt{Con}_\xi$, respectivamente.

En ocasiones necesitaremos trabajar con fórmulas ξ que tengan más de una variable libre, estas variables libres extra pueden ser pensadas como parámetros. Por ejemplo, podemos imaginar una fórmula $\xi(x,y)$ que enumere a una sucesión de conjuntos de axiomas, donde para cada valor n que tome la variable x, la fórmula $\xi(\overline{n},y)$ describe al n-ésimo conjunto de la sucesión. De esta forma, podemos utilizar una misma fórmula como descripción para toda una sucesión de sistemas axiomáticos. Las construcciones de los enunciados de prueba, teorema y consistencia pueden replicarse en este caso. Para enfatizar la distinción entre las variables, escribiremos $\mathtt{Pba}_{\xi(x,)}(u,\nu)$, $\mathtt{Teo}_{\xi(x,)}(u)$, $\mathtt{Con}_{\xi(x,)}$. De igual manera, las funciones f, g y h del párrafo anterior pueden extenderse a estas fórmulas y, por ejemplo, $h(\xi(\overline{n},y)) = \mathtt{Con}_{\xi(\overline{n},)}$. Dicho de otra forma, si tenemos un conjunto recursivamente enumerable $\{\xi_n : n \in \omega\}$ donde $\xi_k \in \Sigma_1$ para todo $k \in \omega$, podemos construir una fórmula $\xi(x,y) \in \Sigma_1$ tal que $\xi(\overline{n},y)$ es lógicamente equivalente con $\xi_n(y)$.

Ya que hemos establecido los puntos importantes sobre cómo la aritmetización de las metamatemáticas puede llevarse a cabo dentro del sistema $\mathcal{S}$, concluiremos el tema con algunos resultados relacionados con los teoremas de incompletud de Gödel que serán importantes en los siguientes capítulos, las pruebas omitidas pueden encontrarse en [8]. A continuación, sea ξ una fórmula de $\mathcal{S}$ con $\mathrm{VL}(\xi) = \{x\}$.

Teorema 1.35. *Si $\xi \in \Sigma_1$, entonces hay $\theta, \phi \in \Sigma_1$ tales que*

$$\vdash_{\mathcal{AP}} \mathtt{Pba}_\xi \leftrightarrow \theta$$

$$\vdash_{\mathcal{AP}} \mathtt{Teo}_\xi \leftrightarrow \phi$$

Teorema 1.36. *Si $\theta \in \Sigma_1$ y $\mathrm{VL}(\theta) \subseteq \{x_0, \ldots, x_n\}$, entonces*

$$\vdash_{\mathcal{AP}} \theta(x_0, \ldots, x_n) \to \mathtt{Teo}_{ap}(\overline{\theta}(\overline{x}_0, \ldots, \overline{x}_n))$$

[10]Haremos una excepción en el uso de nuestra notación $*$ con las fórmulas $\mathtt{Pba}_\xi$, $\mathtt{Teo}_\xi$ y $\mathtt{Con}_\xi$, en las cuales lo omitiremos. Dado que estas fórmulas siempre serán usadas junto a una fórmula ξ, no hay riesgo de confundirlas con las nociones metateóricas de prueba, teorema y consistencia.

Al enunciar el teorema de Σ_1-completud 1.29 mencionamos su origen en las pruebas finitistas de consistencia de la aritmética de Robinson $\mathcal{Q}$. Esto es importante porque la naturaleza finitista de las pruebas nos permite formalizarlas dentro de $\mathcal{AP}$, una descripción de este proceso puede encontrarse en [24]. A partir de la aritmetización de estas pruebas obtenemos el siguiente resultado:

Teorema 1.37. *Sea* $\theta \in \Sigma_1$ *con* $\mathsf{VL}(\theta) \subseteq \{x_0, \ldots, x_{n-1}\}$*. Entonces*

$$\vdash_{\mathcal{AP}} \mathsf{Teo}_{ap}\left(\overline{\theta}\left(\overline{x}_0, \ldots, \overline{x}_{n-1}\right)\right) \to \theta\left(x_0, \ldots, x_{n-1}\right)$$

Teorema 1.38 (Lema diagonal)**.** *Para toda* $\theta \in \mathscr{L}_{AP}$ *con* $\mathsf{VL}(\theta) = \{x_0, \ldots, x_n\}$*, podemos asociar (efectivamente) una fórmula* $\phi \in \mathscr{L}_{AP}$ *tal que*
$\mathsf{VL}(\phi) = \{x_0, \ldots, x_{n-1}\}$ *y*

$$\vdash_{\mathcal{AP}} \phi(x_0, \ldots, x_{n-1}) \leftrightarrow \theta(x_0, \ldots, x_{n-1}, \overline{\phi})$$

Más aún, si $\theta \in \Sigma_1$*, podemos pedir que* $\phi \in \Sigma_1$*.*

Gracias al lema diagonal, podemos asociarle a una descripción ξ un enunciado de Gödel g_ξ que cumple:

$$\vdash_{\mathcal{AP}} g_\xi \leftrightarrow \neg\mathsf{Teo}(\overline{g_\xi})$$

Teorema 1.39 (Teoremas de incompletud de Gödel)**.** *Si* $\mathcal{A} = \langle \tau, A \rangle$ *es un sistema axiomático consistente y recursivamente enumerable que extiende a* $\mathcal{AP}$ *y* $\widehat{\xi} = A$ *entonces:*

(I) $\nvdash_{\mathcal{A}} g_\xi$ *y, si* $\mathcal{A}$ *es* $\omega-$*consistente,* $\nvdash_{\mathcal{A}} \neg g_\xi$*.*

(II) Si $\xi \in \Sigma_1$*, entonces* $\vdash_{\mathcal{AP}} \mathsf{Con}_\xi \leftrightarrow g_\xi$*. Por lo tanto,* $\nvdash_{\mathcal{A}} \mathsf{Con}_\xi$*.*

Teorema 1.40 (Teoremas de incompletud de Gödel para descripciones con parámetros)**.** *Sean* $\xi(x,y) \in \Sigma_1$ *y* $A_d = \widehat{\xi(\overline{d},y)}$ *para cada* d*. Escribiremos* $\mathsf{Pba}(x,u,v) = \mathsf{Pba}_{\xi(x,)}(u,v)$*,* $\mathsf{Teo}(x,u) = \mathsf{Teo}_{\xi(x,)}(u)$ *y* $\mathsf{Con}(x) = \mathsf{Con}_{\xi(x,)}$*. Así, podemos construir una fórmula* $g(y)$ *tal que:*

(I) $\vdash_{\mathcal{AP}} g(x) \leftrightarrow \neg\mathsf{Teo}(x, g(\overline{x}_*))$

(II) $\vdash_{\mathcal{AP}} g(x) \leftrightarrow \mathsf{Con}(x)$

Otro aspecto importante a tratar es cómo podemos llevar a cabo la aritmetización de la teoría de las funciones recursivas parciales en $\mathcal{S}$. Para lograr esto, formalizaremos el sistema ecuacional de Kleene y haremos uso de la lista presentada junto con él en [19] de funciones recursivas primitivas acompañadas por las ecuaciones que las definen. Supondremos que a cada una de esas le corresponde un símbolo funcional y los axiomas apropiados[11] en $\mathcal{S}$. Esto incluye un símbolo funcional g_n

[11] La forma exacta de estos axiomas puede verse en la definición 1.19.

de aridad $n+2$ para cada uno de los enunciados T_n de Kleene que necesitaremos. Los enunciados de T_n permiten representar las funciones recursivas en sistemas aritméticos. En resumidas cuentas, el predicado $T_n(e, i_1, \ldots, i_n, x)$ es verdadero si y sólo si x codifica el cómputo completo de una máquina con índice e sobre las entradas $i_1, \ldots, i_n$. La relación definida por T_n es recursiva primitiva, por lo que podemos asociarle la función U de Kleene que, cuando $T_n(e, i_1, \ldots, i_n, x)$ es verdadero, $U(x)$ devuelve el valor computado por la máquina con índice e con las entradadas $i_1, \ldots, i_n$. Los detalles en cuanto a las definiciones y pruebas pertinentes, pueden encontrarse en [19]. Escribiremos como T_{*n} a la fórmula $g_n(v_0, \ldots, v_{n+1}) \approx \overline{0}$. Más aún, un símbolo funcional U_* corresponde a la función U de Kleene.

Sean $r, t_0, \ldots, t_{n-1}, s$ términos de $\mathcal{S}$. Escribiremos $\{r\}(t_0, \ldots, t_{n-1}) \approx_* s$ para abreviar la fórmula $\exists u\, (T_{*n}(r, t_0, \ldots, t_{n-1}, u) \wedge U_*(u) \approx s)$ donde la variable u no aparece en ninguno de los términos. Utilizando esta abreviación, sólo es necesario traducir los resultados sobre deducciones en el cálculo ecuacional de Kleene a una forma aritmética y arreglar apropiadamente sus pruebas. Nos limitaremos a presentar dos versiones del teorema de la recursión formalizadas en $\mathcal{S}$ pues es la herramienta que necesitaremos para nuestros objetivos.

Teorema 1.41. *Sea* f *una función recursiva de aridad* $k+1$ *y* g_f *de aridad* k *tal que para cualesquiera* $v_1, \ldots, v_k$ *se cumple:*

$$\varphi_{g_f(v_1,\ldots,v_k)} = \varphi_{f(g_f(v_1,\ldots,v_k),v_1,\ldots,v_k)}$$

Entonces

$$\vdash_{\mathcal{S}} \{\overline{g}_f\}(t_1, \ldots, t_k) \approx_* r \leftrightarrow \left\{\overline{f}\right\}(\overline{g}_f, t_1, \ldots, t_k) \approx_* r$$

A partir de este, obtenemos:

Teorema 1.42. *Sea* $\theta \in \Sigma_1$ *con* $\mathsf{VL}(\theta) = \{x_0, \ldots, x_{n+1}\}$ *tal que*

$$\vdash_{\mathcal{S}} \theta(r, t_1, \ldots, t_n, u) \wedge \theta(r, t_1, \ldots, t_n, v) \to u \approx v$$

Entonces podemos encontrar (efectivamente) un número $e \in \omega$ *tal que*

$$\vdash_{\mathcal{S}} \{\overline{e}\}(t_1, \ldots, t_n) \approx_* u \leftrightarrow \theta(\overline{e}, t_1, \ldots, t_n, u)$$

En este punto nos detendremos en nuestra labor de aritmetización en $\mathcal{S}$. Este trabajo lo terminaremos una vez que hayamos desarrollado la teoría básica de los sistemas de notación ordinal en el siguiente capítulo.

2 El sistema $\mathcal{O}$ de Kleene

2.1. Sistemas de Notación Ordinal

En este capítulo desarrollaremos la herramienta necesaria para trabajar con números ordinales dentro de la aritmética. Esta herramienta son los sistemas de notación ordinal, los cuales nos permiten nombrar segmentos iniciales de la sucesión ordinal utilizando números naturales. Esta asignación de nombres (números naturales) es tal que al conocer el nombre correspondiente a un ordinal dado, podemos obtener de forma efectiva nombres para todos los ordinales previos. Podemos formalizar esta idea mediante la siguiente definición:

Definición 2.1. Un *sistema de notación ordinal* $\mathcal{M}$ es una quinteta ordenada $\mathcal{M} = \langle \mathcal{D}_\mathcal{M}, \nu_\mathcal{M}, k_\mathcal{M}, p_\mathcal{M}, q_\mathcal{M} \rangle$ tal que:

(I) $\nu_\mathcal{M} : \mathcal{D}_\mathcal{M} \subseteq \omega \to \mathrm{OR}$

(II) $k_\mathcal{M}$ es una función recursiva parcial tal que:

$$k_\mathcal{M}(x) = \begin{cases} 0 & \text{si } \nu_\mathcal{M}(x) = 0 \\ 1 & \text{si } \nu_\mathcal{M}(x) \text{ es un sucesor} \\ 2 & \text{si } \nu_\mathcal{M}(x) \text{ es límite} \end{cases}$$

(III) $p_\mathcal{M}$ es una función recursiva parcial tal que si $\nu_\mathcal{M}(x)$ es sucesor, entonces $p_\mathcal{M}(x)$ converge y
$$\nu_\mathcal{M}(x) = \nu_\mathcal{M}(p_\mathcal{M}(x)) + 1$$

(IV) $q_\mathcal{M}$ es una función recursiva parcial tal que si $\nu_\mathcal{M}(x)$ es límite, entonces $q_\mathcal{M}(x)$ converge, $\varphi_{q_\mathcal{M}(x)}$ es total y $\left\{\nu_\mathcal{M}(\varphi_{q_\mathcal{M}(x)}(n))\right\}_{n=0}^{\infty}$ es una sucesión creciente que tiene a $\nu_\mathcal{M}(x)$ como límite.

Decimos que $\mathcal{D}_\mathcal{M}$ es el conjunto de notaciones de $\mathcal{M}$ y si $\nu_\mathcal{M}(n) = \alpha$, decimos que n es un nombre para α.

Observemos que, por inducción, es inmediato que si un sistema $\mathcal{M}$ asigna un nombre a un ordinal α, entonces le asigna uno a todos los ordinales anteriores a α. Dicho de otra forma, la imagen de $\nu_{\mathcal{M}}$ es un segmento inicial de la sucesión ordinal. A continuación presentamos un ejemplo sumamente sencillo, para ayudar a la intuición y después presentar el ejemplo que utilizaremos a lo largo de este trabajo: el sistema $\mathcal{O}$ de Kleene.

Ejemplo 2.1.1. Sea $\mathcal{M} = \langle \omega, \nu_{\mathcal{M}}, k_{\mathcal{M}}, p_{\mathcal{M}}, q_{\mathcal{M}} \rangle$, donde

$$\nu_{\mathcal{M}}(n) = n \text{ para todo } n \in \omega$$

$$k_{\mathcal{M}}(n) = \begin{cases} 1 & \text{si } n > 0 \\ 0 & \text{si } n = 0 \end{cases}$$

$$p_{\mathcal{M}}(n) = \begin{cases} n-1 & \text{si } n > 0 \\ \text{diverge} & \text{si } n = 0 \end{cases}$$

$$q_{\mathcal{M}}(n) = 0 \text{ para todo } n \in \omega$$

Este sistema $\mathcal{M}$ es sumamente sencillo, sólo asigna un nombre a los ordinales finitos y, convenientemente, cada número natural es su propio nombre. Aún cuando este es un ejemplo trivial, sirve para mejorar la intuición en el uso de las funciones $k_{\mathcal{M}}$, $p_{\mathcal{M}}$ y $q_{\mathcal{M}}$. Aquí, la función $k_{\mathcal{M}}$ sólo tiene el trabajo de distinguir entre 0 y cualquier otro ordinal, pues no se nombra a ningún ordinal límite. La función $p_{\mathcal{M}}$ asigna, a cada número natural $n > 0$, su antecesor, es decir, $n-1$. En este caso, como 0 no es un ordinal sucesor, podríamos elegir definir $p_{\mathcal{M}}(0)$ de cualquier manera o, como lo hicimos arriba, dejarlo sin definir. De la misma forma, como ningún ordinal límite es nombrado por el sistema $\mathcal{M}$, cualquier función recursiva servirá como $q_{\mathcal{M}}$.

Presentaremos algunas definiciones importantes en la teoría de sistemas de notación ordinal antes de continuar con la presentación del sistema $\mathcal{O}$ de Kleene.

Definición 2.2. Un ordinal α es:

(I) *recursivo* si y sólo si hay $A \subseteq \omega$, $R \subseteq \omega \times \omega$ y $f : A \to \alpha$ tales que:

   a) $\langle A, R \rangle$ es un buen orden,

   b) R es recursiva y

   c) $\langle A, R \rangle \overset{f}{\cong} \langle \alpha, \in \rangle$.

   Un ordinal recursivo es, en resumidas cuentas, aquel con un tipo de orden recursivo. Esto sólo significa que la relación de pertenencia es computable cuando la restringimos a elementos de dicho ordinal.

(II) *construible* si y sólo si hay un sistema de notación ordinal que le asigna un nombre.

Uno de los objetivos principales de este capítulo es estudiar con más precisión a los ordinales construibles. Para esto, utilizaremos las siguientes definiciones:

Definición 2.3. Un sistema de notación ordinal $\mathcal{M}$ es:

(I) *Maximal* si y sólo si le asigna un nombre a cada ordinal construible.

(II) *Universal* si y sólo si para cualquier sistema de notación $\mathcal{N}$ hay una función recursiva parcial $\varphi : \mathcal{D}_{\mathcal{N}} \to \mathcal{D}_{\mathcal{M}}$ tal que si $x \in \mathcal{D}_{\mathcal{N}}$, entonces $\nu_{\mathcal{N}}(x) \leqslant \nu_{\mathcal{M}}(\varphi(x))$

Observación 2.4. Todo sistema universal es maximal. Si $\mathcal{M}$ es universal y $\mathcal{N}$ le asigna el nombre $a \in \omega$ a un ordinal α, entonces hay una función $\varphi : \mathcal{D}_{\mathcal{N}} \to \mathcal{D}_{\mathcal{M}}$ tal que $\nu_{\mathcal{N}}(a) \leqslant \nu_{\mathcal{M}}(\varphi(a))$. De esta forma, $\mathcal{M}$ le asigna el nombre $\varphi(a)$ a un ordinal $\beta \geqslant \alpha$. Así, como la imagen de $\nu_{\mathcal{M}}$ es un segmento inicial de OR, $\mathcal{M}$ le asigna un nombre a α.

A continuación definiremos un sistema de notación ordinal conocido como $\mathcal{O}$ de Kleene, el cual será vital para nuestros objetivos en este trabajo:

Definición 2.5 ($\mathcal{O}$ de Kleene)**.** Definimos simultaneamente por recursión $\nu_{\mathcal{O}} : \mathcal{O} \subseteq \omega \to OR$ y un orden parcial para $\mathcal{O}$ el cual denotaremos $<_{\mathcal{O}}$. El conjunto $\mathcal{O}$ queda definido implícitamente como el dominio de $\nu_{\mathcal{O}}$:

(I) $\nu_{\mathcal{O}}(1) = 0$

(II) Sea γ un ordinal y supongamos que para cualquier otro ordinal α tal que $\alpha < \gamma$, hay $n \in \omega$ que cumple $\nu_{\mathcal{O}}(n) = \alpha$, supongamos también que $<_{\mathcal{O}}$ está definido para las notaciones de los ordinales anteriores a γ:

a) Si $\gamma = \beta + 1$ para algún ordinal β, entonces para todo $n \in \omega$ tal que $\nu_{\mathcal{O}}(n) = \beta$, definimos $\nu_{\mathcal{O}}(2^n) = \gamma$. Más aún, para todo $k \in \omega$, si $k = n$ ó $k <_{\mathcal{O}} n$, agregamos el par $(k, 2^n)$ a $<_{\mathcal{O}}$.

Observación 2.6. Para todo $n \in \omega$, si $n \in \mathcal{O}$, entonces $2^n \in \mathcal{O}$

b) Si γ es un ordinal límite, entonces para cualquier $y \in \omega$ tal que $\{\varphi_y(n)\}_{n=0}^{\infty}$ son nombres para una sucesión creciente de ordinales que convergen a γ tales que

$$\forall i \forall j [i < j \implies (\varphi_y(i), \varphi_y(j)) \in <_{\mathcal{O}}]$$

definimos:

$$\nu_{\mathcal{O}}(3 \cdot 5^y) = \gamma$$

y los pares $(z, 3 \cdot 5^y)$ son agregados a $<_{\mathcal{O}}$ para todo $z \in \omega$ para el cual exista $n \in \omega$ que cumpla que $z <_{\mathcal{O}} \varphi_y(n)$

Así, quedan definidas las funciones parciales recursivas $k_{\mathcal{O}}$, $p_{\mathcal{O}}$, $q_{\mathcal{O}}$ de la siguiente forma:

$$k_{\mathcal{O}}(x) = \begin{cases} 0 & \text{si } x = 0 \\ 1 & \text{si } x = 2^y \quad \text{para algún } y \in \omega \\ 2 & \text{si } x = 3 \cdot 5^y \quad \text{para algún } y \in \omega \end{cases}$$

$$p_{\mathcal{O}}(x) = \begin{cases} y & \text{si } x = 2^y \quad \text{para algún } y \in \omega \\ \text{diverge} & \text{si } x \neq 2^y \quad \text{para todo } y \in \omega \end{cases}$$

$$q_{\mathcal{O}}(x) = \begin{cases} y & \text{si } x = 3 \cdot 5^y \quad \text{para algún } y \in \omega \\ \text{diverge} & \text{si } x \neq 3 \cdot 5^y \quad \text{para todo } y \in \omega \end{cases}$$

Utilizaremos indistintamente $\mathcal{O}$ para denotar al dominio de $\nu_{\mathcal{O}}$ y al sistema de notación ordinal $\langle \mathcal{O}, \nu_{\mathcal{O}}, k_{\mathcal{O}}, p_{\mathcal{O}}, q_{\mathcal{O}} \rangle$. También, denotaremos a $\nu_{\mathcal{O}}(x)$ por $|x|_{\mathcal{O}}$ ó por $x_{\mathcal{O}}$ y a los pares $(x, y) \in <_{\mathcal{O}}$ por $x <_{\mathcal{O}} y$.

Es importante notar que como existen una infinidad de sucesiones crecientes que convergen a un ordinal límite dado, el orden parcial $<_{\mathcal{O}}$ determina un árbol infinito en el cual ramificaciones consecutivas ocurren en ordinales límite consecutivos (figura 2.1).

Entre las propiedades más importantes del sistema $\mathcal{O}$ se encuentra el siguiente resultado por Kleene:

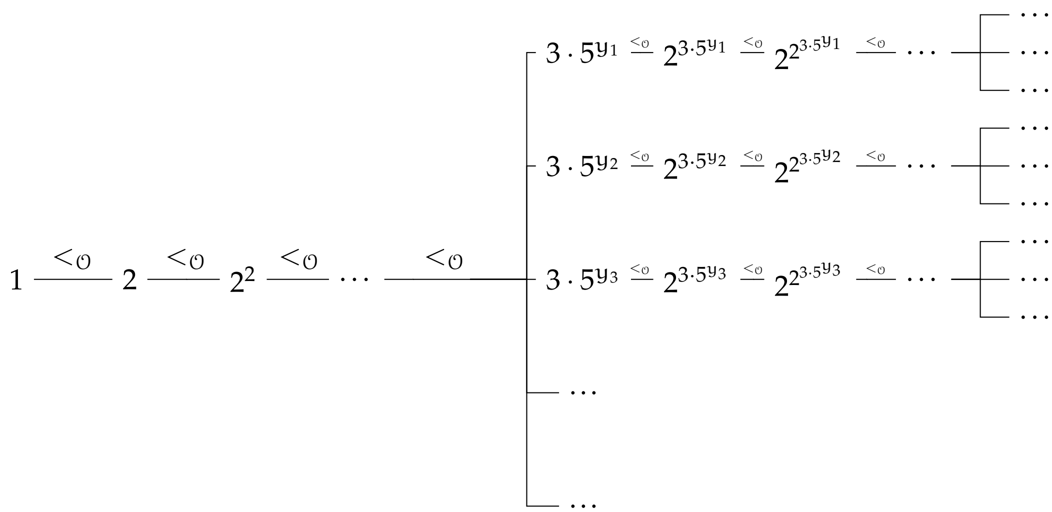

Figura 2.1: Ramificaciones de $<_{\mathcal{O}}$

Teorema 2.7. *Existe una función recursiva en dos variables $+_{\mathcal{O}}$ tal que para cualesquiera $x, y \in \mathcal{O}$:*

(I) $x +_{\mathcal{O}} y \in \mathcal{O}$

(II) $|x +_{\mathcal{O}} y|_{\mathcal{O}} = |x|_{\mathcal{O}} + |y|_{\mathcal{O}}$

(III) Si $y \neq 1$, entonces $x <_{\mathcal{O}} x +_{\mathcal{O}} y$

Demostración. Sean $a, b \in \omega$. Definimos la siguiente función recursiva parcial:

$$\psi(x) = \begin{cases} b, & \text{si } x = 1 \\ 2^{\varphi_a(u)}, & \text{si } x = 2^u \\ 3 \cdot 5^{y'}, & \text{si } x = 3 \cdot 5^y \\ 0, & \text{en otro caso} \end{cases}$$

Con y' un índice para $\varphi_a \circ \varphi_y$.

ψ es recursiva total y su índice depende recursivamente[1] de a, b. Así, existe una función recursiva parcial en dos variables f tal que $\psi = \varphi_{f(a,b)}$. Del teorema de la recursión (1.3) se sigue que hay una función recursiva θ tal que:

$$\varphi_{\theta(x)} = \varphi_{f(\theta(x),x)}$$

Así, definimos $x +_{\mathcal{O}} y = \varphi_{\theta(x)}(y)$. Veamos que posee las propiedades deseadas. Sea $x_0 \in \mathcal{O}$.

[1] Esto quiere decir que, para cada par de números $a, b \in \omega$, tenemos una función ψ distinta. Sin embargo, a partir de a y b podemos obtener de forma efectiva el índice su respectiva función ψ.

(I) De lo anterior, tenemos que

$$x_0 +_{\mathcal{O}} y = \varphi_{\theta(x_0)}(y) \quad \text{para todo } y \in \mathcal{O}$$

Pero

$$\varphi_{\theta(x_0)}(y) = \begin{cases} x_0, & \text{si } y = 1 \\ 2^{\varphi_{\theta(x_0)}(u)}, & \text{si } y = 2^u \\ 3 \cdot 5^{\nu'}, & \text{si } y = 3 \cdot 5^{\nu} \\ 0 & \text{en otro caso} \end{cases}$$

Con ν' un índice para $\varphi_{\theta(x_0)} \circ \varphi_\nu$. Probaremos que si $y \in \mathcal{O}$, entonces $x_0 +_{\mathcal{O}} y \in \mathcal{O}$ por inducción sobre y.

a) Si $y = 1$, entonces $x_0 +_{\mathcal{O}} y = x_0 \in \mathcal{O}$.

b) Si $y = 2^{y'}$ para algún $y' \in \mathcal{O}$, entonces:

$$x_0 +_{\mathcal{O}} y = 2^{\varphi_{\theta(x_0)}(y')} = 2^{x_0 +_{\mathcal{O}} y'} \in \mathcal{O}$$

c) Si $y = 3 \cdot 5^\nu$ con $\{\varphi_\nu(n)\}_{n=0}^\infty$ notaciones para una sucesión creciente de ordinales que convergen a $|y|_{\mathcal{O}}$, entonces:

$$x_0 +_{\mathcal{O}} y = 3 \cdot 5^{\nu'}$$

Con ν' un índice para $\varphi_{\theta(x_0)} \circ \varphi_\nu$. Pero

$$\varphi_{\theta(x_0)}(\varphi_\nu(n)) = x_0 +_{\mathcal{O}} \varphi_\nu(n) \in \mathcal{O} \quad \text{para todo } n \in \omega$$

Por lo que $3 \cdot 5^{\nu'} \in \mathcal{O}$.

De todo esto concluimos que $\forall x, y \in \mathcal{O}\, (x +_{\mathcal{O}} y \in \mathcal{O})$

(II) Tenemos que

$$|x_0 +_{\mathcal{O}} y|_{\mathcal{O}} = |\varphi_{\theta(x_0)}(y)|_{\mathcal{O}}$$

Probaremos por inducción sobre y que $|x_0 +_{\mathcal{O}} y|_{\mathcal{O}} = |x_0|_{\mathcal{O}} + |y|_{\mathcal{O}}$

a) Si $y = 1$, entonces:

$$|x_0 +_{\mathcal{O}} y|_{\mathcal{O}} = |x_0|_{\mathcal{O}} = |x_0|_{\mathcal{O}} + 0 = |x_0|_{\mathcal{O}} + |y|_{\mathcal{O}}$$

b) Si $y = 2^{y'}$ para algún $y' \in \mathcal{O}$, entonces:

$$\begin{aligned} |x_0 +_{\mathcal{O}} y|_{\mathcal{O}} &= |2^{x_0 +_{\mathcal{O}} y'}|_{\mathcal{O}} = |x_0 +_{\mathcal{O}} y'|_{\mathcal{O}} + 1 \\ &= |x_0|_{\mathcal{O}} + |y'|_{\mathcal{O}} + 1 = |x_0|_{\mathcal{O}} + |y|_{\mathcal{O}} \end{aligned}$$

c) Si $y = 3 \cdot 5^{\nu}$ con $\{\varphi_{\nu}(n)\}_{n=0}^{\infty}$ notaciones para una sucesión creciente de ordinales que convergen a $|y|_{\mathcal{O}}$, entonces:

$$|x_0 +_{\mathcal{O}} y|_{\mathcal{O}} = |3 \cdot 5^{\nu'}|_{\mathcal{O}}$$

Con ν' un índice para $\varphi_{\theta(x_0)} \circ \varphi_{\nu}$. Pero

$$|\varphi_{\theta(x_0)}(\varphi_{\nu}(n))|_{\mathcal{O}} = |x_0 +_{\mathcal{O}} \varphi_{\nu}(n)|_{\mathcal{O}} = |x_0|_{\mathcal{O}} + |\varphi_{\nu}(n)|_{\mathcal{O}} \quad \text{para todo } n \in \omega$$

Así,

$$|3 \cdot 5^{\nu'}|_{\mathcal{O}} = |x_0|_{\mathcal{O}} + |y|_{\mathcal{O}}$$

De todo esto concluimos que $\forall x, y \in \mathcal{O}\,(|x +_{\mathcal{O}} y|_{\mathcal{O}} = |x|_{\mathcal{O}} + |y|_{\mathcal{O}})$

(III) Veremos que si $y \neq 1$, entonces $x_0 <_{\mathcal{O}} x_0 +_{\mathcal{O}} y$ aplicando inducción sobre y

a) Si $y = 2^{y'}$ para algún $y' \in \mathcal{O}$, entonces:

$$x_0 +_{\mathcal{O}} y = 2^{x_0 +_{\mathcal{O}} y'} \quad \text{y por hipótesis de inducción} \quad x_0 \leqslant_{\mathcal{O}} x_0 +_{\mathcal{O}} y'$$

Pero, de la definición de $<_{\mathcal{O}}$, tenemos:

$$x_0 +_{\mathcal{O}} y' <_{\mathcal{O}} x_0 +_{\mathcal{O}} y$$

Así,

$$x_0 \leqslant_{\mathcal{O}} x_0 +_{\mathcal{O}} y' <_{\mathcal{O}} x_0 +_{\mathcal{O}} y$$

b) Si $y = 3 \cdot 5^{\nu}$ con $\{\varphi_{\nu}(n)\}_{n=0}^{\infty}$ notaciones para una sucesión creciente de ordinales que convergen a $|y|_{\mathcal{O}}$, entonces:

$$x_0 +_{\mathcal{O}} y = 3 \cdot 5^{\nu'}$$

Con ν' un índice para $\varphi_{\theta(x_0)} \circ \varphi_{\nu}$. Pero

$$x_0 <_{\mathcal{O}} x_0 +_{\mathcal{O}} \varphi_{\nu}(n) \quad \text{para todo } n \in \omega \setminus \{0\}$$

Además, por la construcción de $<_{\mathcal{O}}$ tenemos que

$$\varphi_{\nu}(n) <_{\mathcal{O}} y \quad \text{para todo } n \in \omega$$

Por lo tanto,

$$x_0 <_{\mathcal{O}} x_0 +_{\mathcal{O}} y$$

$\square$

De la misma forma en que $+_{\mathcal{O}}$ es una función recursiva primitiva relacionada con $\mathcal{O}$, podemos obtener un resultado más general con condiciones para la existencia de este tipo de funciones:

Teorema 2.8 (Feferman). *Sean ψ_1, ψ_2 y ψ_3 funciones primitivas recursivas de aridad $k, k+2$ y $k+2$, respectivamente. Entonces podemos encontar funciones primitivas recursivas ϕ, ψ de aridad $k+1$ tales que, para cualesquiera $a_1, \ldots a_k$:*

- *Si $d = 1$ ó $d \neq 2^{d_1}$ ó $d \neq 3 \cdot 5^{d_2}$, entonces*

$$\phi(d, a_1, \ldots, a_k) = \psi_1(a_1, \ldots, a_k)$$

- *Si $d \neq 0$, entonces*

$$\phi(2^d, a_1, \ldots, a_k) = \psi_2(d, a_1, \ldots, a_k, \phi(d, a_1, \ldots, a_k))$$

- $\phi(3 \cdot 5^d, a_1, \ldots, a_k) = \psi_3(d, a_1, \ldots, a_k, \psi(d, a_1, \ldots, a_k))$*, donde*

$$\varphi_{\psi(d, a_1, \ldots, a_k)}(n_{\mathcal{O}}) = \phi(\varphi_d(n_{\mathcal{O}}), a_1, \ldots, a_k)$$

Si $\phi = \varphi_e$, entonces $\psi(d, a_1, \ldots, a_k) = S_1^{k+3}(s, e, d, a_1, \ldots, a_k)$ para un s adecuado[2].

Demostración. Podemos realizar la prueba sin tomar en cuenta los parámetros $a_1, \ldots a_k$ ya que su inclusión no modifica la demostración. Así, $k = 0$ y ψ_1 es una constante, llamémosla r. La estrategia que seguiremos para encontrar la función ϕ será utilizar el teorema de la recursión para obtener su índice; para esto, definiremos una relación recursiva adecuada. Antes, consideremos $s \in \omega$ tal que

$$\varphi_s(x, y, z) = \varphi_x(\varphi_y(z))$$

Definimos la relación recursiva $R \subseteq \omega \times \omega \times \omega$ como:

$$\begin{aligned}(z, d, w) \in R \quad \iff \quad & \big(d = 1 \vee d \neq 2^{d_1} \vee d \neq 3 \cdot 5^{d_2}\big) \wedge (w = r) \\ & \vee \big(d = 2^{d_1} \wedge d_1 \neq 0 \wedge w = \psi_2(d_1, \varphi_z(d_1))\big) \\ & \vee \big(d = 3 \cdot 5^{d_2} \wedge w = \varphi_3(d_2, S_1^3(s, z, d_2))\big)\end{aligned}$$

Por el teorema de la recursión hay $e \in \omega$ tal que

$$\varphi_e(d) = w \quad \iff \quad (e, d, w) \in R$$

Supongamos que φ_e está definida para toda $d < d_0$.

[2]Recordemos del capítulo anterior que la función S_n^{m+1} es una función recursiva primitiva que cumple:

$$\varphi_{S_n^{m+1}(z, y_0, \ldots, y_{m-1})}(x_0, \ldots, x_{n-1}) = \varphi_z(y_0, \ldots, y_{m-1}, x_0, \ldots, x_{n-1})$$

(I) Si $d_0 = 1$ ó $d_0 \neq 2^{d_1}$ ó $d_0 \neq 3 \cdot 5^{d_2}$, entonces $\varphi_e(d_0) = r$.

(II) Si $d_0 = 2^{d_1}$ para algún $d_1 > 0$, entonces $\varphi_e(d_0) = \psi_2(d_1, \varphi_e(d_1))$

(III) Si $d_0 = 3 \cdot 5^{d_2}$ para algún $d_2 \in \omega$, entonces $\varphi_e(d_0) = \psi_3(d_2, S^3_1(s, e, d_2)))$

Por lo tanto, φ_e está bien definida y es primitiva recursiva. Así, $\phi = \varphi_e$ y $\psi(d) = S^3_1(s, e, d)$ cumplen con lo deseado. □

Las razones por las que el sistema $\mathcal{O}$ es perfecto para nuestros propósitos van más allá de poder definir funciones recursivas que respeten su estructura. Probablemente, la propiedad más importante de $\mathcal{O}$ es su universalidad. Esto significa que $\mathcal{O}$ extiende, en cuanto a los ordinales nombrados, a cualquier sistema de notación ordinal.

Teorema 2.9 (Kleene)**.** *$\mathcal{O}$ es un sistema universal de notación.*

Demostración. Sea $\mathcal{M}$ un sistema de notación ordinal. Sea $n \in \omega$. Definimos $\psi : D_{\mathcal{M}} \to \mathcal{O}$ de la siguiente forma:

$$\psi(x) = \begin{cases} 1 & \text{si } k_{\mathcal{M}}(x) = 0 \\ 2^{\varphi_n(p_{\mathcal{M}}(x))} & \text{si } k_{\mathcal{M}}(x) = 1 \\ 3 \cdot 5^y & \text{si } k_{\mathcal{M}}(x) = 2 \text{ y } q_{\mathcal{M}}(x) \text{ es convergente} \\ diverge & \text{en otro caso} \end{cases}$$

Con y un índice para ξ, y ξ definida por:

$$\xi(0) = \varphi_n \circ \varphi_{q_{\mathcal{M}}(x)}(0)$$

$$\xi(m+1) = \xi(m) +_{\mathcal{O}} \varphi_n \circ \varphi_{q_{\mathcal{M}}(x)}(m+1)$$

Como ψ es recursiva parcial y podemos obtener su índice de forma efectiva para cada $n \in \omega$, existe f una función recursiva tal que $\psi = \varphi_{f(n)}$. Por el teorema de la recursión (1.2), hay una función recursiva parcial que cumple que $\varphi_k = \varphi_{f(k)}$. Veamos que dicha función exhibe la universalidad del sistema $\mathcal{O}$ por reducción al absurdo.

Supongamos que la condición de universalidad no se cumple para el sistema $\mathcal{M}$, es decir, que hay un primer ordinal α tal que φ_k no está definido en α ó $\nu_{\mathcal{M}}(x) = \alpha$ y $|\varphi_k(x)|_{\mathcal{O}} < \alpha$

Como el caso $\alpha = 0$ es inmediato, solo demostraremos los casos en que α es sucesor y límite.

(I) Si $\alpha = \beta + 1$ para algún ordinal β y $\nu_{\mathcal{M}}(x) = \alpha$, entonces:

$$|\varphi_k(x))|_{\mathcal{O}} = |2^{\varphi_k(p_{\mathcal{M}}(x))}|_{\mathcal{O}} = |\varphi_k(p_{\mathcal{M}}(x))|_{\mathcal{O}} + 1$$

Sin embargo,
$$\beta = \nu_{\mathcal{M}}(p_{\mathcal{M}}(x)) \leqslant |\varphi_k(p_{\mathcal{M}}(x))|_{\mathcal{O}}$$
Así,
$$\alpha \leqslant |\varphi_k(p_{\mathcal{M}}(x))|_{\mathcal{O}} + 1 = |2^{\varphi_k(p_{\mathcal{M}}(x))}|_{\mathcal{O}} < \alpha$$
Lo cual es una contradicción.

(II) Si α es un ordinal límite, $\nu_{\mathcal{M}}(x) = \alpha$ y y es como arriba, entonces:
$$|\varphi_k(x)|_{\mathcal{O}} = |3 \cdot 5^y|_{\mathcal{O}} = \lim\{|\xi(n)|_{\mathcal{O}}\}_{n=0}^{\infty}$$
Pero
$$\nu_{\mathcal{M}}(\varphi_{q_{\mathcal{M}}(x)}(n)) \leqslant |\varphi_k(\varphi_{q_{\mathcal{M}}(x)}(n)|_{\mathcal{O}} \quad \text{para todo } n \in \omega$$
Además tenemos:
$$\varphi_k(\varphi_{q_{\mathcal{M}}(x)}(n) <_{\mathcal{O}} \xi(n) <_{\mathcal{O}} \xi(n+1) \quad \text{para todo } n > 2$$
Entonces
$$\nu_{\mathcal{M}}(\varphi_{q_{\mathcal{M}}(x)}(n)) < |\xi(n)|_{\mathcal{O}} < |\varphi_k(x)|_{\mathcal{O}} \quad \text{para todo } n > 2$$
De lo anterior, podemos concluir el siguiente absurdo:
$$\alpha \leqslant |\varphi_k(x)|_{\mathcal{O}} < \alpha$$

Con esto queda demostrada la universalidad del sistema $\mathcal{O}$. □

Corolario 2.10. *$\mathcal{O}$ es un sistema maximal de notación ordinal.*

De la maximalidad de $\mathcal{O}$ se sigue que hay ordinales numerables no construibles. Esto se debe a que los ordinales nombrados por $\mathcal{O}$ forman un segmento inicial numerable de OR.

Definición 2.11. Definimos el *ordinal de Church-Kleene*, al cual denotaremos como ω_1^{CK}, como el primer ordinal no construible. Es decir, el primer ordinal al cual $\mathcal{O}$ no le asigna un nombre.

Corolario 2.12. *ω_1^{CK} es numerable.*

Demostración. Como los ordinales a los que el sistema $\mathcal{O}$ les asigna una notación forman un segmento inicial numerable de ordinales, el primer ordinal que no aparece ahí debe ser numerable. Como $\mathcal{O}$ es maximal, ese ordinal es ω_1^{CK}. □

Terminamos esta sección enunciando un resultado sumamente importante en la teoría de los sistemas de notación ordinal. La demostración puede encontarse en [23].

Teorema 2.13 (Markwald, Spector)**.** *Un ordinal es construible si y sólo si es recursivo.*

2.2. Aritmetización de $\mathcal{O}$

Ahora que hemos desarrollado la teoría de los sistemas de notación ordinal y hemos presentado las propiedades básicas de $\mathcal{O}$, procederemos a formalizarlo parcialmente dentro del sistema $\mathcal{S}$. Para ello, denotaremos por $x_{*\mathcal{O}}$ a un término de $\mathcal{S}$ que cumple:

$$\vdash_{\mathcal{S}} \overline{0}_{*\mathcal{O}} \approx \overline{1} \tag{2.1}$$

$$\vdash_{\mathcal{S}} \overline{sn}_{*\mathcal{O}} \approx \overline{2}^{*\, n_{*\mathcal{O}}} \tag{2.2}$$

El objetivo de esto es poder hablar dentro de $\mathcal{M}$ de los nombres que da $\mathcal{O}$ a los ordinales finitos.

La versión formalizada del teorema 2.8 (para el caso $k = 0$) queda de la siguiente manera:

Teorema 2.14. *Sean $r \in \omega$ y t_1, t_2 términos de $\mathcal{S}$ con $\mathrm{VL}(t_1) = \mathrm{VL}(t_2) = \{u, v\}$. Entonces podemos construir de forma efectiva términos t, s de $\mathcal{S}$ con $\mathrm{VL}(t) = \mathrm{VL}(s) = \{u\}$ tales que:*

(I) $\vdash_{\mathcal{S}} \left(z \approx \overline{1} \vee z \not\approx \overline{2}^{*z_1} \vee z \not\approx \overline{3} \cdot \overline{5}^{*z_2}\right) \to t(z) \approx \overline{r}$

(II) $\vdash_{\mathcal{S}} z \not\approx \overline{0} \to t\left(\overline{2}^{*z}\right) \approx t_1(z, t(z))$

(III) a) $\vdash_{\mathcal{S}} t\left(\overline{3} \cdot \overline{5}^{*z}\right) \approx s(z, t_2(z))$

b) $\vdash_{\mathcal{S}} (\{s(z)\}(x_{*\mathcal{O}}) \approx_* w) \leftrightarrow \exists v (\{z\}(x_{*\mathcal{O}}) \approx_* v \wedge t(v) \approx w)$

Observación 2.15. Como $\mathcal{S}$ es una extensión primitiva recursiva de $\mathcal{AP}$, los términos t_1, t_2, t y s representan funciones recursivas primitivas en $\mathcal{S}$.

Finalmente hemos concluido la construcción del sistema $\mathcal{S}$. Aunque no hemos presentado explicítamente sus axiomas y términos, estos podrían reconstruirse a partir de la definición de extensión recursiva primitiva de $\mathcal{AP}$. Más aún, si $\tau_{\mathcal{S}}$ es el tipo de semejanza de $\mathcal{S}$ podemos suponer al trabajar con un sistema aritmético $\mathcal{A} = \langle \tau, \mathrm{Ax} \rangle$ que $\tau \cap \tau_{\mathcal{S}} = \mathrm{AP}$. A partir de esto obtenemos el siguiente resultado[3]:

Teorema 2.16. *Si $\mathcal{A} = \langle \tau, \mathrm{Ax} \rangle$ es un sistema aritmético y $\theta \in \mathscr{L}_{\tau_{\mathcal{S}}}$ cumple que $\vdash_{\mathcal{A} \cup \mathcal{S}} \theta$, entonces $\vdash_{\mathcal{A}} \theta^{(\mathcal{S})}$. En particular, si $\theta \in \mathscr{L}_{\mathrm{AP}}$ tenemos que $\vdash_{\mathcal{A}} \theta$.*

La prueba puede realizarse fácilmente después de extender de forma natural la definición de $\theta^{(S)}$ para $\theta \in \mathscr{L}_{\tau \cup \tau_{\mathcal{S}}}$.

Finalmente contamos con todos los ingredientes necesarios para trabajar con sucesiones de sistemas aritméticos recursivamente enumerables.

[3]Recordemos que si $\theta \in \mathscr{L}_{\tau}$, la fórmula $\theta^{(\mathcal{S})}$ expresa lo mismo que θ, pero utilizando únicamente el lenguaje de la aritmética de Peano. Es decir, $\theta^{(\mathcal{S})} \in \mathscr{L}_{\mathrm{AP}}$ y $\vdash_{\mathcal{S}} \theta \leftrightarrow \theta^{(\mathcal{S})}$. Ver definición 1.21.

3 Progresiones recursivas de teorías axiomáticas

3.1. Construcción de las progresiones

Consideremos $\mathcal{A} = \langle \tau, A \rangle$ un sistema axiomático consistente, recursivamente enumerable y que contenga a la aritmética de Peano. Del segundo teorema de incompletud de Gödel, sabemos que dado un enunciado $\mathtt{Con}_{\mathcal{A}}$ que exprese la consistencia de $\mathcal{A}$ ocurre que $\nvdash_{\mathcal{A}} \mathtt{Con}_{\mathcal{A}}$. De este resultado, surge naturalmente la idea de estudiar sucesiones de sistemas axiomáticos[1] $\{\mathcal{A}_\alpha = \langle \tau, A_\alpha \rangle\}_{\alpha<\gamma}$ donde γ es un ordinal que cumplan las siguientes condiciones:

(I) $\mathcal{A}_0 = \mathcal{A}$

(II) $\mathcal{A}_{\alpha+1} = \mathcal{A}_\alpha + \{\mathtt{Con}_{\mathcal{A}_\alpha}\}$

(III) $\mathcal{A}_\alpha = \bigcup_{\beta<\alpha} \mathcal{A}_\beta \quad$ si α es límite.

Este tipo de sucesiones son interesantes –en primer lugar– por ser la forma más natural para intentar evadir el fenómeno de incompletud. Por otro lado, tienen la siguiente propiedad: dado un conjunto inicial de axiomas, la validez[2] de cada elemento de la sucesión se sigue de la validez de los elementos anteriores, por lo que la validez de toda la sucesión dependerá únicamente de la elección de axiomas iniciales. No basta aceptar la consistencia de un sistema axiomático para justificar agregar como axioma su enunciado de consistencia ya que los teoremas de Gödel nos muestran que hay conjuntos de axiomas consistentes que prueban su propia inconsistencia. Por ejemplo, $\mathcal{AP} + \{\neg\mathtt{Con}_{\mathcal{AP}}\}$ es consistente y $\vdash_{\mathcal{AP}+\{\neg\mathtt{Con}_{\mathcal{AP}}\}} \neg\mathtt{Con}_{\mathcal{AP}}$.

[1]Podemos pensar que todos los sistemas axiomáticos comparten el mismo tipo de semejanza pues, de lo contrario, podríamos realizar la unión de todos ellos.

[2]Recordemos que, al comienzo de la sección 1.3, establecimos a qué nos referimos por validez.

Es necesario aceptar una propiedad más fuerte: $\mathcal{AP}$ es válido, es decir, los axiomas de Peano son verdaderos. Esto ocurre en general, si sabemos que $\mathcal{A}$ es válido, entonces $\mathcal{A}+\{\mathrm{Con}_{\mathcal{A}}\}$ también lo será[3].

Al comenzar a considerar sucesiones $\{\mathcal{A}_\alpha\}_{\alpha<\gamma}$ como la descrita anteriormente, surgen dos preguntas de gran importancia que trataremos de contestar en este capítulo:

- ¿Para cuáles ordinales γ tiene sentido definir la sucesión?
- ¿Qué podemos decir sobre el conjunto de teoremas de $\bigcup\limits_{\alpha<\gamma} \mathcal{A}_\alpha$?

En general, podemos encontrar muchos enunciados no equivalentes que describan la consistencia de una teoría dada, siendo el mejor ejemplo de esto el encontrado en 1.30. Esto significa que al trabajar con teorías arbitrarias, no hay un método uniforme para definir los enunciados de consistencia de todas las teorías en -por ejemplo- una sucesión transfinita de sistemas axiomáticos. Para evadir esta complicación, trabajaremos con descripciones de los axiomas de una teoría y los enunciados de prueba y consistencia que definimos anteriormente. Las descripciones de los axiomas junto con sus respectivos enunciados de consistencia y las otras fórmulas que utilizaremos, las pensaremos formalizadas dentro de la estructura $\mathcal{S}$ definida en la sección de aritmetización.

Ahora podemos estudiar una sucesión transfinita de sistemas axiomáticos $\{\mathcal{A}_\alpha\}_{\alpha<\gamma}$ mediante sucesiones de fórmulas[4] $\xi_\alpha \in \Sigma_1$ con $\alpha < \gamma$, para γ un ordinal fijo; sólo hace falta pedir que $\widehat{\xi_\alpha} = A_\alpha$, para todo $\alpha < \gamma$.

Ahora sí, la condición 2 puede ser reformulada como:

(II) $\mathcal{A}_{\alpha+1} = \mathcal{A}_\alpha + \{\mathrm{Con}_{\xi_\alpha}\}$

Si hemos construido una fórmula ξ_α, siempre podemos construir la siguiente fórmula de la sucesión $\xi_{\alpha+1}$ de la siguiente forma:

$$\xi_{\alpha+1}(x) = \xi_\alpha(x) \vee (x \approx \overline{\mathrm{Con}_{\xi_\alpha}})$$

Así, las sucesiones $\{\xi_\alpha\}_{\alpha<\gamma}$ que nos interesaran serán aquellas para las que γ sea un ordinal límite. Adicionalmente, nos interesa que el conjunto de axiomas de $\mathcal{A}_\alpha$ este contenido propiamente en el de $\mathcal{A}_{\alpha+1}$, por lo que basta fijarnos en γ numerable (ya que $\mathscr{L}_\tau$ es numerable).

[3]Para verificar esto, basta observar que $\neg\mathrm{Con}_{\mathcal{A}} \in \Pi_1$ y revisar las condiciones que impusimos a nuestra noción de validez.

[4]En el capítulo 1 establecimos que lo más apropiado es utilizar fórmulas Σ_1 para representar conjuntos de axiomas pues esto nos permite probar el segundo teorema de incompletud con bastante generalidad. Además, cualquier fórmula Σ_1 puede pensarse como representando un conjunto de axiomas. Específicamente, si $\xi \in \Sigma_1$, entonces ξ representa al conjunto $\widehat{\xi} = \{n \in \omega : \vdash_{\mathcal{AP}} \xi(\bar{n})\}$

El último obstáculo que falta enfrentar antes de poder responder la primera de las preguntas presentadas arriba es cómo definir ξ_α cuando α es un ordinal límite (suponiendo que ya definimos ξ_β, para todo $\beta < \gamma$). El analísis presentado a continuación debería servir como justificación al enfoque que adoptaremos:

Si α es un ordinal límite, entonces hay una sucesión creciente de ordinales $\{\beta_n\}_{n\in\omega}$ que converge a α.

Así, queremos que

$$\mathcal{A}_\alpha = \bigcup_{n\in\omega} \mathcal{A}_{\beta_n}$$

Es decir,

$$\psi \in A_\alpha \text{ si y sólo si } \xi_{\beta_n}(\overline{\psi}) \text{ es verdadero para algún } n \in \omega.$$

Ahora, si pudieramos obtener la sucesión $\{\beta_n\}_{n\in\omega}$ de forma efectiva, es decir, si tuvieramos una función recursiva φ_z tal que, para todo $n \in \omega$ $\varphi_z(n) = \beta_n$, podríamos obtener una fórmula $\xi_\alpha \in \Sigma_1$ que describiera los axiomas de $\mathcal{A}_\alpha$ de la siguiente manera:

$$\xi_\alpha^{(z)}(x) = \exists m \exists n[(\{z\}(m) \approx n) \wedge (\mathsf{Teo}_{ap}(\overline{n}(\overline{x}_*))]^{(\mathcal{S})}$$

Observación 3.1. Dado que la definición de ξ_α depende de la sucesión de ordinales que converge a α, tendremos una infinidad de fórmulas para cada ordinal límite.

De lo anterior, concluimos que los sistemas de notación ordinal desarrollados previamente son la herramienta perfecta para trabajar con sucesiones recursivas de ordinales que convergen a un ordinal límite. Más aún, utilizaremos el sistema $\mathcal{O}$ como estándar debido a su maximalidad entre los sistemas de notación y a la simpleza de su definición.

Finalmente nos encontramos en condiciones de dar una definición formal de progresión recursiva de sistemas axiomáticos:

Definición 3.2 (Progresión de consistencia de sistemas axiomáticos recursivamente enumerables). $\{\xi_d | d \in \mathcal{O}\}$ es una progresión de consistencia de sistemas axiomáticos si y sólo si:

- Para cada $d \in \mathcal{O}$, $\xi_d \in \Sigma_1$.
- ξ_1 describe un conjunto de axiomas recursivamente enumerable que extiende a la aritmética.
- Si $d \in \mathcal{O}$, entonces $\xi_{2^d}(x) = \xi_d(x) \vee (x \approx \overline{\mathsf{Con}}_{\xi_d})$
- Si $3 \cdot 5^d \in \mathcal{O}$, entonces $\xi_{3.5^d}(x) = \exists m \exists n[(\{d\}(\overline{m}) \approx \overline{n}) \wedge (\mathsf{Teo}_{ap}(\overline{n}(\overline{x}_*))]$

  Podemos pensar también la progresión como $\{\mathcal{A}_d | d \in \mathcal{O}\}$ donde $\mathcal{A}_d = \widehat{\xi_d}$.

Observación 3.3. Otra forma de ver la definición anterior, es mediante una función $f : \omega \to \omega$ tal que $f(d) = \xi_d$ para todo $d \in \mathcal{O}$. Esta función es recursiva parcial.

Definición 3.4. Si $\{\xi_d \mid d \in \mathcal{O}\}$ es una progresión de consistencia, llamaremos Ω_{ξ_1} a $\bigcup_{d \in \mathcal{O}} \widehat{\xi_d}$. La fórmula ξ_1 es la más importante, pues a partir de esta queda determinada toda la progresión.

Será importante que las propiedades de una progresión de consistencia sean verificables dentro de la aritmética. Afortunadamente, podemos construir una fórmula $\xi \in \Sigma_1$ con $VL(\xi) = \{x_0, x_1\}$ (para simplificar la notación, escribiremos $x_0 = x$ y $x_1 = y$) que cumpla las siguientes condiciones:

$$\vdash_{\mathcal{S}} \left((x \approx \overline{1}) \vee \left(x \not\approx \overline{2}^{*(x)_{*\overline{1}}} \right) \vee \left(x \not\approx \overline{3} \cdot \overline{5}^{*(x)_{*\overline{3}}} \right) \right) \to (\xi_1(y) \leftrightarrow \xi(x,y)) \tag{3.1}$$

$$\vdash_{\mathcal{S}} x \neq \overline{0} \to \left(\xi \left(\overline{2}^{*x}, y \right) \leftrightarrow \left(\xi(x,y) \vee y \approx \mathsf{Con}_* \left(\overline{\xi} \left(\overline{x}_*, \nu r_{*\overline{1}} \right) \right) \right) \right) \tag{3.2}$$

$$\vdash_{\mathcal{S}} \xi \left(\overline{3} \cdot \overline{5}^{*x}, y \right) \leftrightarrow \xi_1(y) \vee \exists u \exists \nu \left((\{x\}(u_{*\mathcal{O}}) \approx_* \nu) \wedge \xi(\nu, y) \right) \tag{3.3}$$

Estas tres condiciones aseguran que la fórmula ξ verifica, dentro de el sistema $\mathcal{S}$, la construcción de la progresión $\{\xi_d \mid d \in \mathcal{O}\}$. En su mayor parte, estas condiciones son la formalización en $\mathcal{S}$ de la definición 3.2 de progresión de consistencia.

Observación 3.5. Podemos entender mejor el significado de (3.2) si pensamos que para todo $d \in \mathcal{O}$, $\xi_d(y) = \xi(\overline{d}, y)$. Así, tenemos que

$$\vdash_{\mathcal{S}} \overline{\xi} \left(\overline{x}_*, \nu r_{*\overline{1}} \right) \approx \overline{\xi \left(\overline{d}, y \right)}$$

Y también

$$\vdash_{\mathcal{S}} \overline{\xi \left(\overline{d}, y \right)} \approx \overline{\xi_d(y)}$$

Por lo cual tenemos

$$\vdash_{\mathcal{S}} x \approx \mathsf{Con}_* \left(\overline{\xi} \left(\overline{x}_*, \nu r_{*\overline{1}} \right) \right) \leftrightarrow x \approx \overline{\mathsf{Con}_{\xi_d}}$$

Para construir dicha fórmula ξ partiremos de la siguiente fórmula:

$$\psi(x,y,z)=\Bigg[\Big(\big(x\approx\overline{1}\big)\vee\Big(x\not\approx\overline{2}^{*(x)_{*\overline{1}}}\Big)\vee\Big(x\not\approx\overline{3}\cdot\overline{5}^{*(x)_{*\overline{3}}}\Big)\wedge\xi_1(y)\Big)$$

$$\vee\Bigg[\Big(x\approx\overline{2}^{*(x)_{*\overline{1}}}\Big)\wedge\big((x)_{*\overline{1}}\not\approx\overline{0}\big)\wedge$$

$$\Big(\mathrm{Teo}_{ap}\Big(z\Big(\overline{(x)_{*\overline{1}}}_{*},\overline{y}_*\Big)\Big)\vee\Big(y\approx\mathrm{Con}_*\Big(z\Big(\overline{(x)_{*\overline{1}}}_{*},\nu r_{*\overline{1}}\Big)\Big)\Big)\Big)\Bigg]$$

$$\vee\Bigg[\Big(x\approx\overline{3}\cdot\overline{5}^{*(x)_{*\overline{3}}}\Big)\wedge$$

$$\big(\xi_1(y)\vee\exists u\exists v\,\big(\big(\{(x)_{*\overline{3}}\}(u_{*\mathcal{O}})\approx_* v\big)\wedge\mathrm{Teo}_{ap}\,(z\,(\overline{v}_*,\overline{y}_*))\big)\big)\Bigg]\Bigg]$$

Esta fórmula ψ es el esqueleto de lo que queremos que la fórmula ξ cumpla, el punto clave es el uso de la variable z, que será sustituida por $\overline{\psi}$ con ayuda del lema diagonal.

Observación 3.6. $\psi\in\Sigma_1$, pues es combinación booleana de fórmulas en Σ_1.

Por el lema diagonal 1.38 podemos encontrar una fórmula $\xi\in\Sigma_1$ tal que:

$$\vdash_{\mathcal{S}}\psi(x,y,\overline{\psi})\leftrightarrow\xi(x,y)$$

Más aún, por 1.36 y 1.37 obtenemos:

$$\vdash_{\mathcal{S}}\xi(x,y)\leftrightarrow\mathrm{Teo}_{ap}\left(\overline{\xi}\,(\overline{x}_*,\overline{y}_*)\right)$$

Así, tenemos

$$\vdash_{\mathcal{S}}\xi(x,y)\leftrightarrow\Bigg[\Big(\big(x\approx\overline{1}\big)\vee\Big(x\not\approx\overline{2}^{*(x)_{*\overline{1}}}\Big)\vee\Big(x\not\approx\overline{3}\cdot\overline{5}^{*(x)_{*\overline{3}}}\Big)\wedge\xi_1(y)\Big)$$

$$\vee\Bigg[\Big(x\approx\overline{2}^{*(x)_{*\overline{1}}}\Big)\wedge\big((x)_{*\overline{1}}\not\approx\overline{0}\big)\wedge$$

$$\Big(\xi\Big(\overline{(x)_{*\overline{1}}}_{*},\overline{y}_*\Big)\vee\Big(y\approx\mathrm{Con}_*\Big(\overline{\xi}\Big(\overline{(x)_{*\overline{1}}}_{*},\nu r_{*\overline{1}}\Big)\Big)\Big)\Big)\Bigg]$$

$$\vee\Bigg[\Big(x\approx\overline{3}\cdot\overline{5}^{*(x)_{*\overline{3}}}\Big)\wedge$$

$$\big(\xi_1(y)\vee\exists u\exists v\,\big(\big(\{(x)_{*\overline{3}}\}(u_{*\mathcal{O}})\approx_* v\big)\wedge\xi\,(v,y)\big)\big)\Bigg]\Bigg]$$

La función ξ describe por completo a la progresión $\{\xi_d\mid d\in\mathcal{O}\}$ dentro de $\mathcal{S}$. El primer renglón corresponde al sistema inicial de la progresión, es decir, a ξ_1.

El segundo y tercer renglón describen la adición de enunciados de consistencia en los pasos de progresión de la forma 2^d. Finalmente, el cuarto y quinto renglón describe la unión de todos los pasos anteriores en los niveles de la forma $3 \cdot 5^d$ en la progresión. Llamaremos a ξ la *enumeración verificable* de $\{\xi_d | d \in \mathcal{O}\}$, pues su función primordial es la de verificar las propiedades de la progresión $\{\xi_d \mid d \in \mathcal{O}\}$ dentro del sistema $\mathcal{S}$, lo cual será de vital importancia en la siguiente sección. Es importante notar que las condiciones (3.1), (3.2) y (3.3) se siguen inmediatamente de la construcción de ξ.

Teorema 3.7. *Si $\{\xi_d | d \in \mathcal{O}\}$ es una progresión de consistencia y ξ su enumeración verificable, entonces:*

(I) $\vdash_{\mathcal{S}} \xi_1(y) \to \xi(x, y)$

(II)

$$\vdash_{\mathcal{S}} \exists u \exists v\,(\{x\}(u_{*\mathcal{O}}) \approx_* v) \\ \to \left(\xi\left(\overline{3} \cdot \overline{5}^{*x}, y\right) \leftrightarrow (\exists u \exists v\,(\{x\}(u_{*\mathcal{O}}) \approx_* v \wedge \xi(v, y)))\right)$$

Demostración. La prueba de (I) se hace por inducción y es inmediata a partir de las propiedades (3.1), (3.2) y (3.3). (II) se prueba directo de (I) y (3.3). □

Una enumeración verificable es una descripción de axiomas con parámetros como definimos anteriormente. Más aún, captura toda la información de su progresión de consistencia. Así, si tenemos $\{\xi_d | d \in \mathcal{O}\}$ una progresión de consistencia y ξ su enumeración verificable se cumple que $\xi(\overline{d}, x) \equiv \xi_d(x)$.

El propósito de definir las progresiones de consistencia es utilizarlas para facilitar el estudio de las sucesiones transfinitas de sistemas axiomáticos. El siguiente corolario nos brinda el primer resultado sobre este tipo de sucesiones:

Corolario 3.8. *Sea $\mathcal{A} = \langle \tau, A \rangle$ un sistema axiomático consistente, recursivamente enumerable y que extiende a la aritmética, entonces existe una sucesión transfinita de fórmulas $\{\xi_\alpha\}_{\alpha < \omega_1^{CK}}$ y una de sistemas axiomáticos consistentes y recursivamente enumerables $\mathcal{A}_\alpha = \langle \tau, A_\alpha \rangle$ con $A_\alpha = \widehat{\xi_\alpha}$ tal que:*

(I) $\mathcal{A}_0 = \mathcal{A}$

(II) $\mathcal{A}_{\alpha+1} = \mathcal{A}_\alpha + \{\mathrm{Con}_{\xi_\alpha}\}$ *para todo* $\alpha < \omega_1^{CK}$

(III) $\mathcal{A}_\alpha = \bigcup_{\beta < \alpha} \mathcal{A}_\beta$ *para todo* $\alpha < \omega_1^{CK}$ *ordinal límite*

3.2. Teorema de Completud de Turing

Ahora que hemos podido dar una definición satisfactoria para las progresiones de consistencia, tenemos los elementos necesarios para probar el notable teorema de completud de Turing.

Teorema 3.9 (Turing)**.** *Sea* $\{\xi_d | d \in \mathcal{O}\}$ *una progresión de consistencia y* ξ *su enumeración verificable. Si* $\zeta \in \Pi_1$ *es verdadera, entonces hay* $d \in \mathcal{O}$ *tal que* $|d|_{\mathcal{O}} = \omega + 1$ *y* $\vdash_{\xi_d} \zeta$*.*

Antes de pasar a la demostración, es importante reflexionar un momento sobre el significado de este notorio teorema. Si pensamos en términos de sucesiones de sistemas axiomáticos, tenemos que la fórmula ζ es demostrable en el sistema $\mathcal{A}_{\omega+1}$, es decir, ζ es demostrable en el sistema que afirma la consistencia de $\mathcal{A}_\omega$, donde a su vez $\mathcal{A}_\omega$ cuenta con una infinidad de enunciados de consistencia como axiomas. Aquí, la pregunta significativa es ¿cuál es la relación entre los enunciados de consistencia y la fórmula ζ? La respuesta: no demasiada. La verdadera clave del teorema es que el paso ω nos permite dar una descripción nueva de los axiomas. Esto se hara evidente una vez que tengamos la demostración.

Demostración. Sea $\{\xi_d | d \in \mathcal{O}\}$ una progresión de consistencia y $\theta \in \Delta_0$ tal que $\zeta = \forall x\theta$. Definimos la función recursiva ψ de la siguiente forma:

$$\psi(n) = \begin{cases} n_{\mathcal{O}} & \text{si para todo } x \leqslant n,\ \theta(x) \text{ es verdadera.} \\ 2^{3\cdot 5^y} & \text{si hay } x \leqslant n \text{ tal que } \neg\theta(x) \text{ es verdadera.} \end{cases}$$

Por el teorema de la recursión, sabemos que existe $e \in \omega$ tal que:

$$\varphi_e(n) = \begin{cases} n_{\mathcal{O}} & \text{si para todo } x \leqslant n,\ \theta(x) \text{ es verdadera.} \\ 2^{3\cdot 5^e} & \text{si hay } x \leqslant n \text{ tal que } \neg\theta(x) \text{ es verdadera.} \end{cases}$$

Sea $a = 3 \cdot 5^e$.

Como $\forall x\theta$ es verdadera, a es una notación en $\mathcal{O}$ para ω y $|a|_{\mathcal{O}} = \omega$.

Sin embargo, supongamos que no sabemos que $\forall x\theta$ es verdadera. Supongamos que hay $n \in \omega$ tal que $\neg\theta(n)$ es verdadero y sea m el menor natural tal que $\neg\theta(m)$ es verdadero. Así, tenemos que

$$\mathcal{A}_a = \bigcup_{n\in\omega} \mathcal{A}_{\varphi_e(n)}$$

Por lo tanto,

$$\mathcal{A}_a = \mathcal{A}_0 \cup \mathcal{A}_1 \cup \cdots \cup \mathcal{A}_{m-1} \cup \mathcal{A}_{2^a}$$

Pero $\mathcal{A}_{2^a} = \mathcal{A}_a + \{Con_{\xi_a}\}$, por lo que $\vdash_{\mathcal{A}_a} Con_{\xi_a}$. Esto implica que $\mathcal{A}_a$ es inconsistente.

Para resumir, tenemos que si existe $n \in \omega$ tal que $\neg\theta(n)$, entonces $\vdash_{A_a} \mathsf{Con}_{\xi_a}$. Este argumento puede formalizarse mediante la fórmula

$$\exists x \neg\theta(x) \to \mathsf{Teo}_{\xi_a}(\overline{\mathsf{Con}_{\xi_a}}) \tag{3.4}$$

La demostración del teorema está dada a partir de la derivación de (3.4) dentro de $\mathcal{AP}$, como veremos a continuación. Para cada $\theta \in \Delta_0$ podemos encontrar (usando 1.42) $e \in \omega$ tal que

$$\vdash_{\mathcal{S}} \{\overline{e}\}(x) \approx_* u \leftrightarrow \exists v\,(x \approx v_{\mathcal{O}}) \wedge \Big[(\forall y\,(y \leqslant v \to \theta(y)) \wedge u \approx v_{\mathcal{O}}) \\ \vee \left(\exists y\,(y \leqslant v \wedge \neg\theta(y)) \wedge u \approx \overline{2}^{*\overline{3} \cdot \overline{5}^{*\overline{e}}} \right) \Big]$$

Es decir, podemos definir la misma función que antes dentro de $\mathcal{AP}$. De nuevo, siempre que θ sea verdadera se cumplirá que $a = 3 \cdot 5^e \in \mathcal{O}$ y $|a|_{\mathcal{O}} = \omega$. Ahora, por (3.3) tenemos que

$$\vdash_{\mathcal{S}} \exists u \neg\theta(u) \to \forall x \left(\xi\left(\overline{2}^{*\overline{a}}, x\right) \to \xi(\overline{a}, x) \right) \tag{3.5}$$

El enunciado de Gödel g_{ξ_a} es el mismo que $g(\overline{a})$ como se definió en 1.40. Así,

$$\vdash_{\mathcal{AP}} g(\overline{a}) \leftrightarrow \neg\mathsf{Teo}\left(\overline{a}, \overline{g(\overline{a})}\right) \tag{3.6}$$

Por 1.40 y (3.2) tenemos que

$$\vdash_{\mathcal{S}} \mathsf{Teo}\left(\overline{2}^{*\overline{a}}, \overline{g(\overline{a})}\right) \tag{3.7}$$

Por lo tanto, de (3.5) y (3.7) se sigue que

$$\vdash_{\mathcal{AP}} \exists x \neg\theta(x) \to \mathsf{Teo}_{\xi_a}(\overline{g_{\xi_a}}) \tag{3.8}$$

Esto quiere decir, por la propiedad diagonal de g_{ξ_a}, que:

$$\vdash_{\mathcal{AP}} \exists x \neg\theta(x) \to \neg g_{\xi_a} \tag{3.9}$$

Por contrapositiva, esto es equivalente a

$$\vdash_{\mathcal{AP}} g_{\xi_a} \to \forall x \theta(x) \tag{3.10}$$

Pero $\vdash_{\mathcal{A}_{2^a}} g_{\xi_a}$, por lo tanto

$$\vdash_{\mathcal{A}_{2^a}} \forall x \theta(x) \;\; \text{y} \;\; |2^a|_{\mathcal{O}} = \omega + 1$$

□

Corolario 3.10. *Ω_{ap} es completa para fórmulas Π_1. Es decir, para toda $\zeta \in \Pi_1$ verdadera, se tiene que $\vdash_{\Omega_{ap}} \zeta$.*[5]

Dos puntos importantes sobre el teorema de Turing surgen de analizar la prueba. El primero, que el conocimiento de la verdad de ζ es fundamental para asegurar que $3 \cdot 5^e \in \mathcal{O}$. En la siguiente sección hablaremos de las repercusiones de esto en la utilidad de las progresiones y posibles variaciones al respecto. Por otro lado, se vuelve claro que la adición de una infinidad de axiomas de consistencia no juega un papel relevante en la prueba. En realidad, como Torkel Franzén señala con gran precisión en [11], lo que permite obtener este resultado es la descripción no estándar de los axiomas que damos en el paso ω. De nuevo, el uso de una definición numéricamente correcta pero intencionalmente errónea le da sentido a un resultado inicialmente desconcertante. Para ser específicos, la función φ_e utilizada en la prueba sirve para obtener un nuevo nombre para ω en $\mathcal{O}$. Este nombre resulta ser numéricamente correcto pues $\{\varphi_e(n)\}_{n=0}^{\infty}$ son nombres para una sucesión creciente de ordinales que convergen a ω, pero es intencionalmente incorrecto ya que el índice e contiene información sobre la verdad de la fórmula θ que no debería cargar un nombre de ω en $\mathcal{O}$. El siguiente corolario muestra de foma más clara cómo el uso de definiciones intencionalmente incorrectas influyen en este tipo de resultado. En este caso, en lugar de obtener un nombre de ω que carga información extra, incluiremos esta información en la descripción de los axiomas.

Corolario 3.11 (El Truco de Turing)**.** *Sea $\mathcal{A}$ un sistema axiomático consistente y recursivamente enumerable que extiende a la aritmética y $\zeta \in \Pi_1$ verdadera. Entonces hay $\xi \in \Sigma_1$ una descripción de los axiomas de $\mathcal{A}$ tal que:*

$$\vdash_{\mathcal{AP}} \mathrm{Con}_{\xi} \to \zeta$$

Demostración. Sean $\theta \in \Delta_0$ tal que $\zeta = \forall x \theta$ es verdadera y $\xi_0 \in \Sigma_1$ cualquier descripción de los axiomas de $\mathcal{A}$ tal que $\mathrm{VL}(\xi_0) = x_0$. Definimos ξ de la siguiente forma:

$$\xi = \xi_0 \vee \left(\exists x \neg \theta \wedge x_0 \approx \overline{\overline{0} \approx \overline{1}} \right)$$

[5] Recordemos que ap es la descripción estándar de los axiomas de $\mathcal{AP}$. Ver ejemplo 1.32.1.

Así,

$$\vdash_{\mathcal{AP}} \exists x \neg\theta(x) \to \mathrm{Teo}_{\xi}(\overline{g_{\xi}})$$

Por lo que

$$\vdash_{\mathcal{AP}} \exists x \neg\theta(x) \to \neg g_{\xi}$$

Aplicando contrapositiva obtenemos

$$\vdash_{\mathcal{AP}} g_{\xi} \to \forall x\theta(x)$$

Por lo tanto

$$\vdash_{\mathcal{AP}} \mathrm{Con}_{\xi} \to \forall x\theta(x)$$

□

Así, siempre que tengamos que $\forall x\theta$ es verdadera, ξ será una descripción de los axiomas de $\mathcal{A}$ y $\forall x\theta$ será un teorema en $\mathcal{A} + \{\mathrm{Con}_{\mathcal{A}}\}$. Este corolario esclarece el rol que tienen las definiciones intencionalmente incorrectas y evita la posible confusión en cuanto al papel que juegan los enunciados de consistencia.

3.3. Limitaciones y generalizaciones

3.3.1. Progresiones autónomas

Como mencionamos, la prueba del teorema de Turing requiere determinar si un número natural d es el nombre de un ordinal, es decir, si $d \in \mathcal{O}$. En general, esta es una pregunta muy difícil de contestar[6]lo cual elimina la potencial utilidad de este resultado para producir pruebas. Un enfoque que evade estas dificultades consiste en utilizar *progresiones autónomas de sistemas axiomáticos* en las cuales aparte de las condiciones en 3.2 pedimos poder probar que $d \in \mathcal{O}$ en un sistema de la sucesión $A_{d'}$ con $d' <_{\mathcal{O}} d$. Los detalles de cómo podría definirse tal progresión se escapan de los objetivos de este trabajo; sin embargo, Feferman proporciona en [7] la siguiente analogía que captura las ideas centrales alrededor de las progresiones autónomas:

Podemos imaginar a un matemático trabajando con distintos sistemas axiomáticos $\mathcal{A}_d$ de tal forma que le asignamos una progresión recursiva dada por ξ. Así, para cada $d \in \mathcal{O}$ el matemático puede encontrar la fórmula ξ_d que describe los axiomas de $\mathcal{A}_d$. Más aún, el conocimiento de que $d \in \mathcal{O}$ garantizá la validez de los axiomas descritos por ξ_d y de todos los elementos posteriores de la progresión. Sin embargo, el matemático podría encontrar imposible determinar, para un d_0 determinado, si $d_0 \in \mathcal{O}$ por más que avance en la progresión. Es decir, podría verse obligado a apelar a un oráculo.

Esta analogía pone en evidencia la absoluta incapacidad de las progresiones recursivas de sistemas axiomáticos en cuanto a su aplicabilidad a la actividad matemática cotidiana, para resultar de utilidad sería necesario que el matemático pudiera obtener por si mismo la información de que $d \in \mathcal{O}$ en algún punto anterior de la progresión a ξ_d.

Por otro lado, una progresión autonóma necesariamente producirá un conjunto de teoremas recursivamente enumerable, por lo que será susceptible a los teoremas de incompletud de Gödel de forma que un resultado de completud como el obtenido por Turing sería imposible.

3.3.2. Principios de reflexión

En [7], Feferman propone construir progresiones de sistemas axiomáticos no sólo a partir de enunciados de consistencia, sino con la adición iterada de distintas fórmulas. Es decir, si partimos de un sistema axiomático $\mathcal{A} = \langle \tau, A \rangle$ consistente, recursivamente enumerable y que contenga a la aritmética, podemos construir una progresión de sistemas axiomáticos $\{\mathcal{A}_\alpha\}_{\alpha<\omega_1^{CK}}$ con sus respectivas descripciones de axiomas $\{\xi_\alpha\}_{\alpha<\omega_1^{CK}}$ de tal forma que se cumplan las siguientes condiciones:

(I) $\mathcal{A}_0 = \mathcal{A}$

(II) $\mathcal{A}_{\alpha+1} = \mathcal{A}_\alpha + \Gamma$ con $\Gamma \subseteq \mathscr{L}_\tau^0$

(III) $\mathcal{A}_\alpha = \bigcup_{\beta<\alpha} \mathcal{A}_\beta$ si α es límite.

Exactamente qué conjuntos Γ son adecuados para esta construcción es un tanto difícil de determinar. Feferman argumenta que los axiomas de $\mathcal{A}_{\alpha+1}$ deben "expresar una cierta confianza" en el sistema $\mathcal{A}_\alpha$. Intuitivamente, esta confianza se puede asociar a la inclinación de un matemático para aceptar un cierto conjunto de axiomas y trabajar con ellos. Por supuesto, esto no nos brinda un criterio general y, mucho menos, uno formal. Al comienzo de la sección 1.3, presentamos una serie de criterios necesarios para un concepto de *validez*. Es precisamente un concepto que cumpla estos criterios lo que permite determinar si un conjunto Γ es apropiado para la construcción de una progresión de sistemas axiomáticos. Para poder realizar una construcción similar a la realizada en este trabajo y rescatar resultados como el teorema de Turing, basta pedir que la validez de los axiomas de $\mathcal{A}_{\alpha+1}$ se siga de la validez de los axiomas de $\mathcal{A}_\alpha$; a los procesos para pasar de $\mathcal{A}_\alpha$ a $\mathcal{A}_{\alpha+1}$ que cumplen esta propiedad, Feferman les llamó *principios de reflexión*. Algunos ejemplos de principios de reflexión son:

[6] Se requieren ω_1^{CK} saltos de Turing a partir del grado de Turing de los conjuntos computables para llegar al grado de Turing de $\mathcal{O}$. Esto contrasta con los ω saltos que son necesarios para llegar al grado de Turing del conjunto de enunciados de AP verdaderos. Más información en [23].

(I) Agregar a los axiomas de $\mathcal{A}_\alpha$ el siguente conjunto:

$\Gamma = \left\{\mathsf{Teo}_{\xi_\alpha}(\overline{\theta}) \to \theta \mid \theta \in \mathscr{L}^0_\tau\right\}$.

(II) Agregar a los axiomas de $\mathcal{A}_\alpha$ el conjunto:

$\Gamma = \left\{\theta \in \mathscr{L}^0_\tau \mid \vdash_{\mathcal{A}_\alpha} \theta\right\}$.

(III) Agregar a los axiomas de $\mathcal{A}_\alpha$ el conjunto:

$\Gamma = \left\{\forall x \mathsf{Teo}_{\xi_\alpha}\left(\overline{\theta}\left(\overline{x}_*\right)\right) \to \forall x \theta(x) \mid \theta \in \mathscr{L}_\tau \text{ y } \mathsf{VL}(\theta) \subseteq \{x\}\right\}$.

(IV) Agregar a los axiomas de $\mathcal{A}_\alpha$ el siguiente conjunto:

$\Gamma = \left\{\forall x \theta(x) \mid \vdash_{\mathcal{A}_\alpha} \forall x \mathsf{Teo}_{\xi_\alpha}\left(\overline{\theta}\left(\overline{x}_*\right)\right)\right\}$ con θ como en el inciso anterior.

Turing conjeturó que un resultado de completud para fórmulas Π_2 similiar al presentado anteriormente podía obtenerse al trabajar con progresiones basadas en el principio de reflexión (I). Sin embargo, no consiguió probarlo. En [7], Feferman mostró que esta conjetura era falsa.

Por otro lado, (III) es especialmente interesante pues los teoremas de una progresión basada en él coinciden con los teoremas del primer miembro de la sucesión de sistemas axiomáticos cuando le agregamos la *regla* ω[7] [7]. Así, en una progresión de este tipo pueden probarse todos los enunciados verdaderos de la aritmética.

Otras limitaciones y distintas propiedades interesantes de las progresiones recursivas surgen de explorar lo que ocurre cuando en lugar de todo $\mathcal{O}$, construimos las progresiones a partir de cadenas en $\mathcal{O}$. Por ejemplo, una progresión $\{\mathcal{A}_d \mid d \in \mathcal{O}\}$ es llamada *invariante* si y sólo si $\mathsf{Teo}_{\mathcal{A}_d} = \mathsf{Teo}_{\mathcal{A}_{d'}}$, siempre que $|d|_\mathcal{O} = |d'|_\mathcal{O}$. Otro de los resultados más significativos de Turing fue que, para progresiones bastante generales, no puede ocurrir que sean completas e invariantes al mismo tiempo. Otras propiedades de las progresiones relacionadas con los caminos de $\mathcal{O}$ se exploran en [10].

[7] La regla ω establece que si un sistema aritmético $\mathcal{A}$ cumple que $\vdash_\mathcal{A} \theta(\bar{n})$ para todo $n \in \omega$, entonces $\vdash_\mathcal{A} \forall x \theta(x)$. Es fácil probar mediante inducción sobre la construcción de fórmulas que cualquier fórmula aritmética verdadera es derivable a partir de $\mathcal{AP}$ adicionado con la regla ω.

4 El propósito de Turing

Al igual que la mayoría del trabajo realizado en lógica matemática durante la primera mitad del siglo XX, las progresiones transfinitas de sistemas axiomáticos (o lógicas ordinales, como las llamó Turing) nacieron del interés en resolver cuestiones relativas a la filosofía y los fundamentos de las matemáticas. En este caso, algunas de las preguntas que motivaron la investigación fueron: ¿qué tan necesaria es la intuición en las matemáticas?, ¿cuál es el alcance del fenómeno de incompletud descubierto por Gödel?, ¿es posible reconciliar métodos constructivos y no constructivos para obtener resultados de utilidad? y ¿cómo podría funcionar una máquina capaz de producir demostraciones?. A lo largo de este capítulo exploraremos con más detalle estas preguntas, así como su lugar en el trabajo de Turing, con la esperanza de que esto provea de un mejor contexto a los capítulos anteriores.

4.1. Intuición e ingenio

A principios del siglo XX surge el *programa de Hilbert*, una propuesta para resolver diversos problemas relativos a los fundamentos de las matemáticas presentada por el matemático alemán David Hilbert. El programa propone encontrar un conjunto finito de axiomas a partir de los cuales se puedan derivar, mediante una serie de reglas mecánicas, cualquier proposición verdadera. Además, el sistema habrá de acompañarse de una prueba de su consistencia basada exclusivamente en métodos finitistas[1].

[1]Hilbert no es del todo claro en cuanto a qué se refiere por finitismo. En [14], da pistas sobre lo que es, para él, el enfoque finitista:

> *Es necesario entonces, si es que hemos de tener a nuestra disposición deducciones e inferencias lógicas confiables, que los objetos sean suceptibles de una visión global completa de sus partes y que su presencia, sus diferencias mutuas, su ordenación, su sucesión o su concatenación acompañe a los objetos, al mismo tiempo, como algo dado de manera inmediata en la intuición, como algo irreducible a cualquier otra cosa, como algo que no requiere ya ninguna reducción.*

El éxito del programa significaría que no es necesario aceptar la existencia de objetos matemáticos como los conjuntos transfinitos, pues toda la matemática podría ser reducida a la manipulación mecánica de símbolos. La noción de verdad podría ser formalmente definida (pues sería equivalente a derivabilidad en el sistema) y así quedaría resuelto en gran medida el problemas de los fundamentos de las matemáticas.

Sin embargo, los teoremas de incompletud de Gödel probaron que era imposible cumplir con las condiciones del programa, al menos en su forma más estricta: Para cualquier sistema axiomático con un conjunto de axiomas decidible, consistente y capaz de expresar a la aritmética recursiva, es posible construir un enunciado verdadero pero que el sistema no es capaz de probar. Más aún, uno de estos enunciados indecidibles afirma la consistencia del propio sistema, por lo que las pruebas de consistencia finitistas para teorías más fuertes[2] que la aritmética se vuelven imposibles. Estos resultados parecen sugerir que la intuición no puede eliminarse por completo de las matemáticas, pues siempre habrán afirmaciones que sólo mediante intuición podremos reconocer como verdaderas.

Turing afirma en [28] que determinar la verdad de una proposición matemática conlleva dos procesos: *intuición* e *ingenio*. En sus palabras,

> *La actividad de la intuición consiste en hacer juicios espontáneos que no son el resultado de vías de razonamiento conscientes. Estos juicios son a menudo, pero de ninguna manera invariablemente, correctos.*
>
> *El ejercicio del ingenio en las matemáticas consiste en ayudar a la intuición mediante el arreglo apropiado de proposiciones y, tal vez, de figuras geométricas y dibujos. La intención es que cuando se utilizan buenos arreglos, la validez de los pasos intuitivos necesarios no pueden ser puestos en duda seriamente.*

Bajo esta interpretación, Turing describe los sistemas formales como un mecanismo para sistematizar el uso del ingenio y la intuición en las demostraciones matemáticas. Así, Turing plantea que la carga puesta en la intuición es aligerada por la elección de ciertos axiomas y reglas de inferencia que siempre producen resultados intuitivamente válidos, mientras que el ingenio toma una forma muy específica: determinar la sucesión de aplicaciones de reglas de inferencia más eficiente (en cuanto a longitud de la prueba) para probar un teorema particular. En este sentido, el programa de Hilbert pretende eliminar la necesidad de la intuición

---

En este trabajo adoptaremos la postura planteada por Tait en [26] donde, analizando la postura de Hilbert, defiende que "todos los modos de razonamiento finitista son primitivos recursivos."

[2] En general, podemos pensar que una teoría es más fuerte que la aritmética si la aritmética es interpretable en ella en el sentido de Tarski. Excactamente qué significa esto puede encontrarse en [5].

por completo, encontrando un conjunto de axiomas y reglas de inferencia a partir de los cuales sea posible producir cualquier teorema matemático.

La distinción entre intuición e ingenio se hace más clara en la práctica. Trabajando en un sistema formal y aceptando como válidos sus axiomas y reglas de inferencia, la carga intuitiva de cada paso de una derivación formal es practicamente nula. Sin embargo, elegir cuál ruta seguir para derivar un teorema particular a partir de estos axiomas y reglas de inferencia puede requerir de una gran cantidad de ingenio.[3]

Considerando que si un conjunto de axiomas es recursivamente enumerable y las reglas de inferencia son recursivas, el conjunto de teoremas que es posible derivar a partir de ellos es recursivamente enumerable; el ingenio se convierte –al menos en teoría– en un elemento despreciable. Idealmente, uno podría, si contara con un suministro ilimitado de lápiz y papel, revisar todas las posibles derivaciones de un sistema formal hasta llegar a la buscada. Lo único que sería necesario (a causa de los teoremas de Gödel) es intuición para determinar si la proposición buscada es derivable a partir de esos axiomas. En otras palabras, podemos imaginar tener una intuición infalible; ésta nos permitiría determinar inequívocamente si una fórmula es derivable o no a partir de cierto conjunto de axiomas. Una vez que sabemos que cierta fórmula es en efecto demostrable a partir de los axiomas, encontrar la prueba será exclusivamente una cuestión de paciencia.

Es aquí donde encuentran su lugar las progresiones transfinitas de sistemas axiomáticos: ya no se trata de eliminar la intuición por completo, sino de limitar y unificar el uso de ésta. En lugar de intentar suprimir el uso de la intuición, lo hacen explícito. Los juicios intuitivos son exactamente decidir si un número natural determinado es o no el nombre de algún ordinal en $\mathcal{O}$, mientras que todos los demás pasos son recursivos.

Aunque Turing voltea hacia los métodos no constructivos de una forma muy ajena a las intenciones del programa de Hilbert, su trabajo está encaminado en la misma dirección: explorar la naturaleza de la demostración matemática. En una carta a Max Newmann, Turing presenta muchas de las ideas detrás de su tesis doctoral:

> *En las pruebas hay, en realidad, una enorme cantidad de trabajo duro, una cierta cantidad de ingenio, mientras que en la mayoría de los casos los "métodos de prueba" son muy conocidos ¿No podemos hacer más claro cuándo hay que hacer el trabajo duro, en qué forma contribuye el ingenio y qué métodos de prueba se usan? De hecho, ¿no podemos expresar brevemente cuál es el estatus*

[3] Esto se confirma en la actualidad con el uso de programas como COQ que asisten en la mecanización de demostraciones formales, pero requieren del ingenio humano en la toma de decisiones sobre los métodos de prueba. De forma similar a como una calculadora reduce la posibilidad del error humano en el cálculo aritmético básico, COQ hace lo mismo en la obtención de derivaciones lógicas formales. El lector interesado en COQ, puede encontrar más información en `https://coq.inria.fr/`.

de cada prueba? Los ordinales fueron pensados para dar notaciones precisas del estatus de pruebas.[4]

4.2. Completud e Incompletud

Cuando un matemático decide trabajar con un determinado conjunto de axiomas –los axiomas de Peano, por ejemplo– está aceptando implícitamente su validez. Así, ese mismo matemático no debería poner objeción en aceptar la validez de los axiomas de Peano junto con una proposición que afirma la consistencia de ellos. Este razonamiento nos otorga el camino a las progresiones transfinitas de sistemas axiomáticos partiendo del segundo teorema de incompletud de Gödel. El paso restante es aceptar la validez de la unión de toda una sucesión, donde cada elemento de la sucesión es el resultado de añadir el enunciado de consistencia a los axiomas del elemento anterior, lo cual debería ser inmediato si ya se ha aceptado la validez de cada elemento de la sucesión.

La pregunta inmediata es si es posible, una vez que nos hemos extendido sobre la sucesión ordinal transfinita, escapar de un resultado similar al de Gödel. Dado que los enunciados indecidibles construidos en la demostración de Gödel pertenecen al estrato Π_1 de la jerarquía aritmética, esto puede traducirse en el siguiente cuestionamiento: ¿Es la progresión de consistencia de la aritmética de Peano Ω_{ap} completa para enunciados Π_1?

El teorema de completud de Turing sirve como "seguro" ante el fenómeno de incompletud descubierto por Gödel. En [7], Feferman hace el siguiente comentario en esta dirección:

> *Un teorema general de incompletud para progresiones recursivas(...) habría sido una prueba dramática de la vasta extensión del fenómeno de incompletud. Sin embargo, la situación ha resultado no ser esa.*

Aunque a primera vista este teorema podría parecer un resultado abrumador, un análisis de la complejidad lógica de $\mathcal{O}$ revela que no es posible describirlo con ningún enunciado de la jerarquía aritmética, es decir, se requeriría una fórmula con una infinidad de cuantificadores universales y existenciales alternados para describir la propiedad de ser elemento de $\mathcal{O}$. Más aún, Turing describe el resultado como "completamente inútil para obtener pruebas". La justificación de esta afirmación se revela cuando consideramos, por ejemplo, la hipótesis de Riemann. De ser verdadera, puede probarse en Ω_{ap}[5], sin embargo, la verdad de la hipótesis de Riemann

[4] La carta puede encontrarse reproducida en [31].

en realidad se estaría asumiendo al tomar un número natural a como nombre de un ordinal en $\mathcal{O}$.

4.3. Máquinas e Inteligencia

Al observar la totalidad del trabajo publicado de Turing, se vuelven claras muchas de las cuestiones que capturaron su interés a lo largo de su vida. De la variedad de temas que exploró, destacan sus ideas sobre máquinas, inteligencia y formas de razonamiento no sólo por su genialidad, sino por su impacto sobre la historia. A continuación analizaremos el papel que juegan las lógicas ordinales en el desarrollo de estas ideas, partiendo de su trabajo en computabilidad, hasta llegar a sus ideas sobre máquinas inteligentes.

4.3.1. Computabilidad y el *Entscheidungsproblem*

En 1936, Turing publica su artículo *On computable numbers, with an application to the Entscheidungsproblem* [27] donde propone las máquinas de Turing (llamadas por el ''a-máquinas'') como formalizaciones de la noción intuitiva de proceso efectivo. La motivación para dar una definición precisa de computabilidad surge de su estudio del *Entscheidungsproblem* (problema de la decisión, en español), un problema planteado por Hilbert y Ackermann en [15] que pregunta la existencia de un método efectivo para determinar si una fórmula es demostrable, a partir de un cierto conjunto de axiomas, dentro del cálculo de predicados o bien, lo que es equivalente, determinar si tal fórmula es válida. Turing creía que dicho método no podía existir, por lo que necesitaba de una definición formal de método efectivo para poder dar una prueba matemática de la imposibilidad de dicho método. En el mismo artículo incluye su solución al problema de la decisión.

En su definición de máquina, Turing captura lo que él considera que hace una persona mientras calcula. Es importante recordar que en 1936 no se tenían los métodos de cómputo que se tienen ahora (en buena parte gracias a Turing) y que el oficio de *calculador* era realizado por personas que computaban operaciones con lápiz y papel. Turing no creía que la totalidad de los procesos mentales pudieran ser reducidos a procedimientos mecánicos, pero en su análisis sólo se preocupaba por el proceso de calcular. Su trabajo lo llevó a concluir que su definición aplicaba a cualquier procedimiento efectivo.

Esta definición destaca entre las distintas propuestas de fomalización de computabilidad de la época (como el cálculo λ y las funciones recursivas de Gödel-Herbrand)

[5]Como consecuencia del trabajo de Davis, Matiyasevic, Robinson y Putnam alrededor del décimo problema de Hilbert, la hipótesis de Riemann se probó equivalente a un enunciado Π_1. Ver [3].

por dos razones: la claridad con la que su definición corresponde con la noción intuitiva de proceso mecánico y su aplicabilidad al mundo real. Church reconoce esto en su revisión del artículo de Turing donde afirma, al referirse a la definición de Turing, que ”la identificación con la efectividad en el sentido ordinario (no definido explícitamente) es evidente inmediatamente”[2]. Respecto a su aplicabilidad, Turing afirmaba que las computadoras digitales como el motor de computación automática (ACE, por sus siglas en inglés) eran ”versiones prácticas de la máquina universal[6]”[30]. Su argumento podría repetirse para afirmar que cualquier computadora moderna es una versión práctica de la máquina universal, con sus respectivas ventajas y limitaciones físicas.

Acompañando su resolución del *Entscheidungsproblem*, Turing especula sobre una forma de superar estas limitaciones. Específicamente, comenta cómo utilizando un proceso no uniforme, podría –hasta donde sabemos– obtenerse una sucesión no computable δ:

> *Es (hasta donde sabemos) posible que cualquier cantidad asignada de cifras de δ pueda ser calculada, pero no por un proceso uniforme. Cuando suficientes dígitos de δ han sido calculados, un método esencialmente nuevo es necesario para obtener nuevas cifras [27].*

Por ejemplo, aunque no podemos esperar encontrar una máquina que determine correctamente en todos los casos si una fórmula es o no demostrable a partir de un conjunto específico de axiomas, sí podemos encontrar –a partir de una fórmula y un conjunto de axiomas– una máquina que decida la pregunta para este caso particular. Así, utilizando distintas máquinas podríamos aproximar tanto como quisieramos una solución del problema de la decisión. La no computabilidad de este proceso se encuentra en que la elección de la siguiente máquina a utilizar no puede realizarse de forma uniforme. Esta idea de proceso no uniforme aparece de nuevo en el corazón de sus lógicas ordinales.

4.3.2. Lógicas ordinales y procedimientos no uniformes

Para Turing, su solución al *Entscheidungsproblem* y el segundo teorema de Gödel son resultados de cáracter similar: ambos marcan límites claros en los procesos de deducción mediante técnicas computables. Al igual que en el caso del problema de la decisión, Turing propone utilizar procesos no uniformes para explorar los límites de la incompletud. Esto lo expone en una carta a su colega Max Neumann de 1940:

[6]La máquina universal que plantea Turing resulta el precursor teórico de la computadora moderna, pues es una máquina de Turing capaz de simular el comportamiento de cualquier otra. Esto ocurre de forma similar a como las computadoras de hoy en día soportan una gran variedad de programas y aplicaciones.

Uno imagina distintas máquinas que permiten distintos tipos de pruebas, y eligiendo una maquina adecuada, uno puede aproximar ''verdad" con ''demostrabilidad" mejor que con una menos adecuada, y en cierto sentido uno puede aproximarla tan bien como lo desee.[7]

La idea es que las sucesiones transfinitas de sistemas formales pueden ser vistas como una sucesión de máquinas que prueban teoremas, donde en cada paso se elige una máquina más adecuada y se añaden axiomas y métodos de demostración. Aunque no hay un proceso uniforme –y, por lo tanto, una máquina– capaz de determinar la sucesión de máquinas adecuada, el proceso se asemeja al de algunos casos en la evolución histórica de las matemáticas, donde se crean nuevas teorías para resolver problemas que no han podido resolverse con las herramientas que existían previamente. Por ejemplo, la demostración del último teorema de Fermat planteado en el siglo XVII requirió del desarrollo de áreas como la geometía algebraica para por fin ser probado en 1995 por Andrew Wiles.

4.3.3. La inteligencia artificial y la objeción matemática

Al terminar la segunda guerra mundial, Turing concentró sus esfuerzos en la realización de una "versión práctica"de su máquina universal. Trabajó en el Laboratorio Nacional de Física (Londres) en el diseño del motor de computación automática. A partir de 1947, gran parte de su trabajo se enfocó en el área de inteligencia artificial. En su famoso artículo de 1950, *Computing machinery and intelligence* [29], Turing propone un test para determinar si una computadora podía considerarse inteligente. Éste consistiría en una conversación en lenguaje natural entre la máquina y una persona a través de una interfaz de texto, durante la cual la máquina trataría de engañar a la persona y hacerle creer que, en realidad, la máquina también es una persona.

En ese mismo artículo, Turing debate nueve objeciones comunes contra la inteligencia artificial. Entre estas, es interesante considerar la que Turing llama "objeción matemática a la inteligencia artificial", pues tanto en su planteamiento como en la respuesta de Turing surgen muchas de las ideas presentes en este trabajo.

La objeción matemática consiste en rechazar la posibilidad de una máquina inteligente con base en resultados matemáticos limitativos como los teoremas de incompletud de Gödel y la propia solución de Turing al problema de la decisión, bajo la idea de que las personas no somos susceptibles a dichas limitaciones. En [29], Turing refuta esta postura señalando que hasta ahora no se ha dado una prueba de que el intelecto humano no sufra limitaciones similares y que "nosotros también damos la respuesta incorrecta demasiado seguido como para justificar estar

[7] Carta reproducida en [31].

muy satisfechos ante tal evidencia de falibilidad por parte de las máquinas". El argumento de Turing se centra en que, aunque puede existir una persona capaz de constestar correctamente preguntas que una máquina dada no podría responder, a su vez habrá alguna máquina capaz de contestar preguntas que esa persona no podría, repitiéndose el argumento *ad infinitum*.

En el artículo de 1950 Turing restringe su argumento contra la objeción matemática, al señalar que los humanos podrían sufrir las mismas limitaciones que las máquinas. Sin embargo, en su plática de 1947 sobre la ACE otorga otro argumento:

> *Se ha mostrado, por ejemplo, que en ciertos sistemas lógicos no puede haber una máquina que distinga las fórmulas demostrables en el sistema de las no demostrables, i.e. no hay un test aplicable por una máquina que pueda dividir las proposiciones con certeza en estas dos clases. Así, si una máquina se construye con este propósito debe, en algunos casos, ser incapaz de responder. Por otro lado, si un matemático es confrontado con este problema, buscaría y encontraría nuevos métodos de prueba, de tal forma que eventualmente llegaría a una decisión respecto a cualquier fórmula dada. Este sería el argumento. En su contra yo diría que se les debe otorgar juego limpio a la máquina. En lugar de que algunas veces no dé una respuesta, podría arreglarse que ocasionalmente dé respuestas erróneas. Pero el matemático humano cometería de la misma manera deslices al intentar técnicas nuevas. Es fácil para nosotros ignorar estos deslices y darle otra oportunidad, pero la máquina probablemente no recibiría misericordia. En otras palabras, si se espera que una máquina sea infalible, no puede también ser inteligente. Hay muchos teoremas matemáticos que dicen casi exactamente esto. Pero estos teoremas no dicen nada sobre cuánta inteligencia puede ser mostrada por una máquina si no se pretende que sea infalible. Para continuar mi petición de ''juego limpio para las máquinas" cuando evaluamos su coeficiente intelectual, un matemático humano siempre ha llevado un entrenamiento extensivo. Este entrenamiento puede ser considerado similar a poner tablas de instrucciones en una máquina. Por lo tanto, uno no debe esperar que una máquina realize la mayor parte del trabajo al construir sus propias tablas de instrucciones. Ningún hombre añade mucho al cuerpo del conocimiento, ¿por qué deberíamos esperar más de una máquina? Dicho de otra forma, debe permitírsele a la máquina contacto con humanos para que pueda adaptarse a sus estándares. El juego de ajedrez puede ser adecuado para este propósito, pues los movimientos del oponente de la máquina otorgarán automáticamente este contacto*[8]*[30].*

[8]Esta cita es especialmente interesante si observamos cómo se ha desarrollado la inteligencia artificial en la actualidad. No hay mejor ejemplo que AlphaGo, la inteligencia artificial desarrollada por

Uno puede rastrear muchos de los rasgos de este argumento en sus ideas sobre lógicas ordinales. Específicamente, la idea de que el matemático humano, al enfrentarse con un problema, es capaz de añadir métodos de prueba en la forma de nuevos axiomas que, en principio, le permitirían resolver cualquier problema particular. Más aún, permitir a una máquina ser falible nos otorga la posibilidad a superar las restricciones inherentes de seguir un método uniforme de la misma forma que desprendernos de los métodos completamente finitistas con las lógicas ordinales nos permiten superar las restricciones de los teoremas de incompletud.

4.4. Comentario final

Alan Turing fue, principalmente, un pensador. Como tal, no sólo examinó a máxima profundidad las ideas que despertaron su intéres, sino que revolucionó la forma en que entendemos hoy en día la inteligencia y las máquinas. Sin embargo, el rasgo que lo separa de otras grandes mentes en la historia es la amplitud con que analizó tales ideas y la habilidad para hallar mejores respuestas.

Sería incorrecto calificar a Turing como un teórico, aún cuando mucho de su trabajo apuntaba a responder preguntas de carácter filosófico, pues el no se veía a sí mismo como tal. Evidencia de esto puede encontrarse en su participación en la ruptura de los códigos alemanes durante la segunda guerra mundial y, de forma aún más prominente, en su participación en el diseño del ACE. Más aún, el carácter polifacético de Turing queda en evidencia en el hecho de que su nombre tiene la misma probabilidad de ser mencionado en discusiones sobre filosofía de la mente, diseño de computadoras, lógica matemática, morfogénesis, criptografía, historia de la programación e inteligencia artificial. En ello no se equivocó Jack Copeland cuando, en [31], afirmó que para Turing no había distinción entre lo puro y lo aplicado. Más bien, se trataba de un pensador universal inmerso en problemas procedentes de muy distintos frentes, desde la lógica y la teoría de grupos, hasta la biología, desde la filosofía hasta los componentes de un circuito eléctrico. En verdad, un espíritu muy alejado del estrecho enfoque que podemos hallar en la investigación y la educación hoy en día.

DeepMind en 2015 que derrotó a un jugador profesional de Go en una partida igualada por primera vez en la historia. La relevancia de las ideas de Turing se hace evidente en que el diseño de AlphaGo es muy similar, al menos en los rasgos generales, al descrito en esta cita. Para más detalle sobre el funcionamiento de AlphaGo, ver [25].

Bibliografía

[1] A. Church y S. C. Kleene, «Formal definitions in the theory of ordinal numbers.», English, *Fundam. Math.*, vol. 28, págs. 11-21, 1937, ISSN: 0016-2736; 1730-6329/e.

[2] A. Church, «Review of *On computable numbers, with an application to the Entscheidungsproblem* by A. M. Turing», *The Journal of Symbolic Logic*, vol. 2, n.º 1, págs. 42-43, 1937, ISSN: 00224812. dirección: `http://www.jstor.org/stable/2268810`.

[3] M. Davis, Y. Matijasevic y J. Robinson, «Hilbert's tenth problem: Diophantine equations; positive aspects of a negative solution», en *Mathematical developments arising from Hilbert problems*, Proceedings of symposia in pure mathematics, F. E. Browder, ed., vol. 28, American Mathematical Society, 1976, págs. 323-378.

[4] A. Ehrenfeucht y S. Feferman, «Representability of recursively enumerable sets in formal theories», *Archiv für mathematische Logik und Grundlagenforschung*, vol. 5, n.º 1-2, págs. 37-41, 1960.

[5] H. Enderton, *A mathematical introduction to logic*. Academic press, 2001.

[6] S. Feferman, «Formal consistency proofs and interpretability of theories», Tesis doct., University of California, Berkeley, 1957.

[7] ——, «Transfinite Recursive Progressions of Axiomatic Theories», *The Journal of Symbolic Logic*, vol. 27, n.º 3, págs. 259-316, 1962, ISSN: 00224812. dirección: `http://www.jstor.org/stable/2964649`.

[8] ——, «Arithmetization of Metamathematics in a General Setting», *Journal of Symbolic Logic*, vol. 31, n.º 2, págs. 269-270, 1966.

[9] ——, «Turing's thesis», *Notices of the AMS*, vol. 53, n.º 10, pág. 2, 2006.

[10] S. Feferman y C. Spector, «Incompleteness along paths in progressions of theories 1», *The Journal of Symbolic Logic*, vol. 27, n.º 4, págs. 383-390, 1962.

[11] T. Franzén, *Inexhaustibility: A Non-Exhaustive Treatment*, A. K. P. Press, ed. Association for Symbolic Logic, 2004.

[12] ——, «Transfinite progressions: a second look at completeness», *Bulletin of Symbolic Logic*, vol. 10, n.º 3, págs. 367-389, 2004.

[13] K. Gödel, «Über formal unentscheidbare Sätze der Principia Mathematica und verwandter Systeme I», *Monatshefte für Mathematik und Physik*, vol. 38, n.º 1, págs. 173-198, 1931.

[14] D. Hilbert, «Acerca del infinito», *HILBERT, D. Fundamentos de las matemáticas. México, DF: UNAM*, págs. 83-121, 1993.

[15] D. Hilbert y W. Ackermann, *Grundzüge der theoretischen Logik (Principles of Mathematical Logic)*. Springer, Berlin, 1928.

[16] A. Hodges, *Alan Turing: the enigma*. Random House, 2012.

[17] K. Hrbacek y T. Jech, *Introduction to Set Theory, Revised and Expanded*. Crc Press, 1999.

[18] S. C. Kleene, «On notations for ordinal numbers», *The Journal of Symbolic Logic*, 1938.

[19] ——, *Introduction to metamathematics*. North-Holland Pub. Co., 1964.

[20] J. D. Monk, *Mathematical logic*, P. R. Halmos, ed. Springer-Verlag, 2012, vol. 37.

[21] A. Nakid Cordero, «Un resultado de completud para progresiones transfinitas de sistemas axiomáticos recursivamente enumerables», Tesis de Licenciatura, Universidad Nacional Autónoma de México, 2018.

[22] G. Piccinini, «Alan Turing and the mathematical objection», *Minds and Machines*, vol. 13, n.º 1, págs. 23-48, 2003.

[23] H. Rogers, *Theory of recursive functions and effective computability*, Mcgraw-Hill, ed. MIT Press, 1967.

[24] J. Shoenfield, *Mathematical Logic*, H. Rogers, ed. Adisson-Wesley Publishing Company, 1967.

[25] D. Silver, A. Huang, C. J. Maddison, A. Guez, L. Sifre, G. Van Den Driessche, J. Schrittwieser, I. Antonoglou, V. Panneershelvam, M. Lanctot y col., «Mastering the game of Go with deep neural networks and tree search», *Nature*, vol. 529, n.º 7587, págs. 484-489, 2016.

[26] W. W. Tait, «Finitism», *The Journal of Philosophy*, vol. 78, n.º 9, págs. 524-546, 1981, ISSN: 0022362X. dirección: `http://www.jstor.org/stable/2026089`.

[27] A. M. Turing, «On computable numbers, with an application to the Entscheidungsproblem», *Proceedings of the London mathematical society*, vol. 2, n.º 1, págs. 230-265, 1937.

[28] ——, «Systems of logic based on ordinals», *Proceedings of the London Mathematical Society*, vol. 2, n.º 1, págs. 161-228, 1939.

[29] ——, «Computing Machinery and Intelligence», *Mind*, vol. 59, n.º 236, págs. 433-460, 1950, ISSN: 00264423, 14602113. dirección: http://www.jstor.org/stable/2251299.

[30] ——, «Lecture to the london mathematical society on 20 february 1947», *MD COMPUTING*, vol. 12, págs. 390-390, 1995.

[31] ——, *The Essential Turing: Seminal Writings in Computing, Logic, Philosophy, Artificial Intelligence, and Artificial Life plus The Secrets of Enigma*, B. J. Copeland, ed. Oxford University Press, 2004.