

Rodolfo Cristian Ertola *Biraben* / 53

Tese de Church: algumas questões histórico-conceituais



Dissertação de Mestrado apresentada ao
Departamento de Filosofia do Instituto de
Filosofia e Ciências Humanas da Universi-
dade Estadual de Campinas, sob a orienta-
ção do Prof. Dr. Carlos Alberto *Lungarzo* / 53

Este exemplar corresponde à redação final
da dissertação defendida e aprovada pela
Comissão Julgadora em 8/7/94.



UNICAMP

Julho de 1994



9410346

UNIDADE	BC
N.º OPERAÇÃO	T/UNICAMP
	053t
V.	17988
PROD.	286/94
C	☐
D	☒
PROD.	R\$ 11,00
DATA	06/05/94
N.º CPU	

CM-00062127-5

Agradecimentos

A Frank Thomas Sautter, pelas longas horas compartilhadas em discussões e suas correções de português.

Ao Dr. Carlos Gustavo González, pelos comentários efetuados e sua colaboração na edição deste trabalho.

Ao Dr. Göran Sundholm, pela generosa cooperação e pelos conselhos oportunos.

Aos professores Drs. Carlos Alberto Lungarzo, José Alexandre Durry Guerzoni, Michael Beaumont Wrigley e Walter Alexandre Carnielli, pelas críticas efetuadas.

A Rita de Cássia Oliveira, pelas correções de português.

Ao Dr. José Carlos Cifuentes Vásquez e a Denise Silva Vilela, pelos comentários realizados.

Ao pessoal da biblioteca do CLE, pela assistência nas buscas bibliográficas.

Ao CNPq, pela bolsa que possibilitou este trabalho.

Conteúdo

0	Introdução	iii
1	Histórico das funções recursivas	1
1.0	Introdução	1
1.1	Dedekind	5
1.2	Skolem	13
1.3	Ackermann	17
1.4	Gödel	22
1.5	Herbrand	28
2	Histórico da tese de Church	33
2.0	Church, Gödel e Kleene	33
2.1	Post	39
2.2	Turing	40
2.3	Debate entre Post, Church e Kleene	42
3	Questões históricas	49
3.0	Havia Gödel proposto a tese de Church?	49
3.1	Por que Gödel não aceitou em 1934 a tese de Church?	51
3.2	Por que Church não elegeu o seu próprio conceito de definibilidade lambda para enunciar a sua tese?	57
3.3	Qual foi o primeiro conceito formal a expressar a noção intuitiva de calculabilidade?	60
4	Questões conceituais	63
4.0	A tese de Church: uma conjectura?	63
4.1	O uso da tese de Church	64

4.2	As justificações de Church	67
4.3	O caráter absoluto da computabilidade	72
4.4	Prova de incompletude usando a tese de Church	76
4.5	A respeito de um argumento de Kalmár	83
5	Apêndice	89
6	Referências	91
7	Índice	99

Introdução

Nos textos de lógica é freqüente encontrarmos uma seção dedicada à tese de Church onde se tecem justificações conceituais. Entretanto, na prática matemática é usual tratar com definições, axiomas e teoremas, e efetuar demonstrações, mas tanto o termo “tese” como a presença de justificações conceituais não são freqüentes. Esta situação motivou, inicialmente, o nosso interesse por este tema. O interesse aumentou quando soubemos da atitude cautelosa de Gödel a respeito do assunto e da existência do trabalho de KALMÁR (1959), que apresenta argumentos contrários à plausibilidade da tese de Church.

É conhecida a relevância da tese de Church a respeito da existência de métodos de decisão. Tomando em consideração este fato, não é difícil justificar um trabalho sobre este tema. Em particular, nesta dissertação pretendemos ter feito as seguintes contribuições, esperando que as mesmas possam ser úteis de um ponto de vista didático:

- Advertência de um erro de van Heijenoort na interpretação de Gödel a respeito do conceito de expressabilidade (p. 27–28);

- Advertência de uma interpretação equivocada de Odifreddi sobre Gödel e Post (suposto platonismo) (p. 53–54);
- Resposta da questão do motivo pelo qual Church não usou a sua própria noção de definibilidade lambda para enunciar a sua tese (p. 58–59);
- Advertência da interpretação equivocada de Thomas a respeito do *status* da tese de Church (p. 63–64);
- Prova simples e direta do teorema de incompletude de Gödel usando a tese de Church, a qual esclarece a falácia de Kalmár comparando-a com o argumento correto “mais próximo” (p. 81–83);
- Interpretação da atitude de Church como “proposta ousada” (p. 89).

Nossa intenção inicial era efetuar um trabalho mais conceitual do que histórico. Porém, a análise dos textos consultados exigiu uma pesquisa histórica para poder compreender as idéias e atitudes envolvidas, por exemplo, um estudo do conceito de recursão. Desta forma, a nossa dissertação começa com um histórico do conceito de recursão até 1931 comentando as contribuições de Dedekind, Skolem, Ackermann, Gödel e Herbrand. A seguir apresentamos um histórico da tese de Church desde 1931 até 1936, iniciando com alguns comentários sobre um sistema de Church e concluindo com um debate a respeito do *status* da tese de Church. A divisão em questões históricas e conceituais dos dois capítulos seguintes é de certo modo arbitrária. No apêndice fornecemos um resumo histórico. Embora o nosso trabalho apresente inúmeras referências, gostaríamos de destacar as que foram nossa principal fonte de consulta. Para o textos originais dos autores estudados con-

sultamos as obras de DAVIS (1965), VAN HEIJENOORT (1967), GÖDEL (1986 e 1990). Para análises históricas foram muito úteis as obras de WEBB (1980), KLEENE (1981a), DAVIS (1982) e GANDY (1988).

Capítulo 1

Histórico das funções recursivas

1.0 Introdução

A história das funções recursivas até 1936 é normalmente dividida em dois períodos, antes e depois de Dedekind. Nesta introdução nos deteremos no primeiro período, ou seja, na “*Pré-história da recursão*”, apresentando previamente algumas observações conceituais e terminológicas. Nas seções posteriores examinaremos o segundo período, fornecendo as contribuições de Dedekind, Skolem, Ackermann, Gödel e Herbrand.

Os matemáticos estão familiarizados com provas por indução. A recursão nada mais é do que a *definição* por indução. A palavra *Rekursion* já ocorre em 1888 em CANTOR (1966, p. 434). Outro termo utilizado com uma acepção semelhante é *recorrente*, que ainda se usa hoje em dia (ver

e.g. SEDGEWICK, 1990). O termo *mathematical induction* provem de De Morgan em 1838 (ver CAJORI, 1991, p. 331) e *vollständige Induktion* em 1888 de DEDEKIND (1969, p. 354).

Embora seja natural pensar que a utilização de definições por indução é tão antiga quanto o uso de provas por indução, o que se verifica na história da matemática é que as provas por indução aparecem mais explicitamente do que as definições por indução.

O seguinte raciocínio do século XII (ver MANBER, 1989, p. 30) mostra que alguma forma de raciocínio indutivo parece ser tão antiga quanto o entendimento humano, em particular, anterior ao desenvolvimento matemático dos gregos. O raciocínio intenta provar que a expressão “3 dias antes do feriado” não deve ser interpretada incluindo o feriado. Caso base: “1 dia antes do feriado” não deve ser interpretada incluindo o feriado, porque teria o mesmo significado que a expressão “o feriado” e parece não ter sentido interpretar essas duas expressões como se fossem sinônimas. Em segundo lugar, “2 dias antes do feriado” não deve ser interpretada incluindo o feriado, porque teria o mesmo significado que “1 dia antes do feriado”, expressão que não inclui o feriado pelo que acabamos de ver, e também parece não ter sentido interpretar as orações “2 dias antes do feriado” e “1 dia antes do feriado” como se fossem sinônimas, uma vez que “1” e “2” têm significados diferentes. O caso de “3 dias antes do feriado” é similar e, portanto, fica provado que essa expressão não deve incluir o feriado. O argumento apresentado pode ser generalizado para expressões da forma “ n dias antes do feriado”, com n qualquer, pelo fato de que não tem sentido que “ n dias antes do feriado” e

" $n + 1$ dias antes do feriado" sejam sinônimas. Dado seu conteúdo elementar, o exemplo é útil para mostrar que os raciocínios indutivos têm uma origem muito antiga, tornando-se difícil o estabelecimento de sua data histórica.

Os historiadores da matemática consideram que o método de provas por indução está *implícito* na demonstração de Euclides de existência de infinitos primos (ver KLINE, 1972, p. 272 e CAJORI, 1991, p. 142). Lembremos que o matemático grego infere a existência de $n + 1$ primos a partir da existência de n primos. Isto, ademais do fato de que existe um primeiro número primo, é suficiente para concluir o teorema mencionado. Aparentemente, o primeiro matemático que fez um uso *explícito* da indução foi Franciscus Maurolycus (1494–1575), que usando este método provou que $1 + 3 + \dots + (2n + 1) = (n + 1)^2$ (para maiores esclarecimentos ver BUSSEY, 1917).

Na prova de Gottfried Leibniz (1646–1716) de que $2 + 2$ são 4 que aparece nos seus *Novos Ensaios* (LEIBNIZ, 1978, p. 394), obra que só foi publicada em 1765, faltava a fórmula $2 + (1 + 1) = (2 + 1) + 1$. Esta igualdade pode ser vista como um caso especial da lei de associatividade (ver FREGE, 1986, p. 18), mas também pode ser vista como um caso particular da equação

$$(A) \quad a + (b + 1) = (a + b) + 1,$$

que é justamente uma das equações utilizadas para definir indutivamente a função adição. Em 1860, o matemático alemão Hermann Grassmann (1809–1877) forneceu, aparentemente pela primeira vez, definições indutivas das operações aritméticas *básicas*, incluindo a equação (A):

“Wenn a und b beliebige Glieder der Grundreihe sind, so versteht man unter der Summe $a+b$ dasjenige Glied der Grundreihe, für welches die Formel

$$a + (b + e) = a + b + e$$

gilt.” (FREGE, 1986, p. 19)

O termo e que aparece na fórmula da citação refere-se ao número 1. Por sua vez, Gottlob Frege (1848–1925) criticou em 1884 a proposta de Grassmann de definir a adição por meio da equação (A):

“Gegen diese Erklärung lässt sich zweierlei einwenden. Zunächst wird die Summe durch sich selbst erklärt. Wenn man noch nicht weiss, was $a + b$ bedeuten soll, versteht man auch den Ausdruck $a + (b + e)$ nicht. Aber dieser Einwand lässt sich vielleicht dadurch beseitigen, dass man freilich im Widerspruch mit dem Wortlaute sagt, nicht die Summe, sondern die Addition solle erklärt werden. Dann würde immer noch eingewendet werden können, das $a + b$ ein leeres Zeichen wäre, wenn es kein Glied der Grundreihe oder deren mehre von der verlangten Art gäbe. Dass dies nicht statthabe, setzt Grassmann einfach voraus, ohne es zu beweisen, sodass die Strenge nur scheinbar ist.” (FREGE, 1986, p. 19)

Em primeiro lugar, vemos que Frege objeta que (A) é circular, pois, segundo ele, se se ignora o significado de $a+b$, não se pode conhecer o significado de $a + (b + 1)$. Esta objeção pode ser aplicada a qualquer definição indutiva

e, portanto, leva provavelmente mais em consideração a forma textual de Grassmann do que a idéia que ele pretende expressar. A nossa interpretação da primeira objeção de Frege é compartilhada por DUMMETT (1991, p. 56–57). A segunda objeção de Frege parece-nos mais pertinente. Ele afirma que o termo $a + b$, de acordo com uma prática matemática rigorosa, só deveria ser introduzido quando estiver provada a existência e unicidade de sua denotação. Porém, Grassmann não tinha fornecido tal prova.

Além destas objeções de Frege, é interessante observar que em (A) os termos $b + 1$ e $(a + b) + 1$ devem ser interpretados, respectivamente, como sucessor de b e sucessor de $a + b$, em símbolos, b' e $(a + b)'$, onde, usando terminologia atual, o sucessor é uma função inicial dada previamente às definições indutivas, neste caso, à definição da adição. Esta imprecisão ainda é frequente nos textos contemporâneos, onde o mesmo símbolo “+” se usa para indicar sucessor e adição.

1.1 Dedekind

Richard Dedekind (1831–1916) foi o primeiro matemático a fornecer uma justificação das definições por indução do tipo solicitado por Frege na sua segunda objeção a Grassmann. Esta justificação encontra-se na obra *Was sind und was sollen die Zahlen?* (DEDEKIND, 1969) publicada pela primeira vez em 1888, embora o próprio Dedekind afirme que as principais idéias já tinham sido elaboradas com bastante antecipação (ver DEDEKIND, 1969,

p. 335 e 336). As contribuições mais marcantes do mencionado trabalho são, em ordem de ocorrência, as seguintes:

- (D1) Definição de cadeia (*Kette*, em alemão);
- (D2) Definição de cadeia de um conjunto;
- (D3) Definição de conjunto infinito;
- (D4) Definição de conjunto simplesmente infinito;
- (D5) Teorema de indução completa;
- (D6) Teorema de definição indutiva;
- (D7) Isomorfismo entre conjuntos simplesmente infinitos.

Segue-se um comentário a respeito de cada um destes pontos com ênfase em (D6).

(D1): Seja φ uma função de S em S e K um conjunto incluído em S . Então, K é uma *cadeia* se $\varphi[K]$ (a imagem de K por φ) está incluída em K (Definição 37). Dedekind refere-se às cadeias de maneira absoluta, i.e. sem especificar a função φ envolvida, mas comenta que elas sempre dependem de uma função.

(D2): Seja φ uma função de S em S e A um conjunto incluído em S . A pode ser ou não ser uma cadeia. Porém, como S é uma cadeia, A sempre estará incluído numa cadeia. Desta forma, é natural pensar na menor cadeia na qual A está contido. Assim, define-se A_0 como a interseção de todas as

cadeias nas quais A está contido (Definição 44). Em particular resulta que se A é uma cadeia, então $A_0 = A$ (Teorema 51). Os conjuntos A_0 também dependem da função φ escolhida.

(D3): Um conjunto é *infinito* se existe uma função bijetiva dele em uma parte própria de si mesmo (Definição 64). Esta noção corresponde ao que atualmente denominamos *Dedekind-infinito* e implica a noção usual de conjunto infinito. A implicação recíproca cumpre-se com auxílio do axioma da escolha.

(D4): Um conjunto N é *simplesmente infinito* (Definição 71) se existe uma função φ de N e um objeto $1 \in N$ que satisfazem as seguintes condições:

- (a) $\varphi[N] \subseteq N$;
- (b) $N = \{1\}_0$;
- (c) $1 \notin \varphi[N]$;
- (d) φ é injetiva.

Dedekind consegue caracterizar a série dos números naturais com esta noção de conjuntos simplesmente infinitos.

Em uma carta datada de 27 de fevereiro de 1890 (VAN HEIJENORT, 1967, p. 98–103), que foi redescoberta por WANG (1957), Dedekind expôs as grandes dificuldades desta questão. Numa tentativa de expressar em linguagem de primeira ordem as condições que definem a série dos números

naturais, poder-se-ia chegar às seguintes fórmulas, onde “ ’ ” designa a função sucessor intuitiva:

$$(N1) \quad (\forall n)(\forall m)(n \neq m \rightarrow n' \neq m');$$

$$(N2) \quad (\forall n)(1 \neq n');$$

$$(N3) \quad (\forall n)[(\forall m)(n \neq m') \rightarrow n = 1];$$

$$(N4) \quad (\forall n)(n \neq n').$$

Dedekind observou que (N1)–(N4) não descartavam a existência de elementos que denominou, muito significativamente, de *intrusos*. Por exemplo, usando a notação ordinal, $\omega + (\omega^* + \omega)$ satisfaz (N1)–(N4). Dedekind conseguiu eliminar os intrusos com auxílio da condição (b), quer dizer, a condição de que N é idêntico à interseção de todas as cadeias que contêm o primeiro elemento, expressado em terminologia atual, a interseção de todos os conjuntos indutivos. É conhecido que esta condição não pode ser expressa em linguagem de primeira ordem.

(D5): Seja n um elemento de um conjunto simplesmente infinito N . Para provar que um teorema vale para todos os elementos de $\{n\}_0$ é suficiente provar que vale para n e que se vale para um elemento m qualquer de $\{n\}_0$, então vale para $\varphi(m)$, onde φ é uma função injetiva que existe por ser N um conjunto simplesmente infinito (Teorema 80). Dedekind prova este teorema como caso particular de uma formulação geral da indução completa para cadeias (Teorema 59). A partir deste ponto usaremos a notação “ n ” em lugar de “ $\varphi(n)$ ”.

(D6): (Teorema 126) Seja θ uma função de Ω em Ω , onde Ω é um conjunto qualquer, e seja ω um elemento qualquer de Ω . Então, existe uma e só uma função ψ de N em Ω , onde N é um conjunto simplesmente infinito qualquer, tal que:

$$(1) \quad \psi(1) = \omega ;$$

$$(2) \quad \psi(n') = \theta\psi(n).$$

Detalharemos a prova deste teorema. Dedekind fornece uma demonstração baseada no seu teorema 125, que presupõe as seguintes definições. Sejam n e m elementos quaisquer de um conjunto simplesmente infinito N . Diz-se que n é maior que m (em símbolos, $m < n$) se $\{n\} \subseteq \{m'\}_0$ (Definição 89). Seja n um elemento de um conjunto simplesmente infinito N . Então, denomina-se Z_n ao conjunto de elementos de N tais que não são maiores que n (Definição 98). Intuitivamente, no caso que N seja o conjunto de números naturais e $'$ a função sucessor, $Z_n = \{1, \dots, n\}$. Agora estamos em condições de apresentar o enunciado do teorema 125:

(T125) Seja θ uma função de Ω em Ω , onde Ω é um conjunto qualquer, e seja ω um elemento qualquer de Ω e $n \in N$, onde N é um conjunto simplesmente infinito qualquer. Então, existe uma e só uma função ψ_n de Z_n em Ω tal que :

$$(1) \quad \psi_n(1) = \omega ;$$

$$(2) \quad \psi_n(t') = \theta\psi_n(t), \text{ se } t < n.$$

Para provar (D6) é necessário usar também o seguinte lema, que pode ser demonstrado facilmente:

(L) Se $n \leq p$, então $\psi_p(n) = \psi_n(n)$,

onde ψ_p e ψ_n são as funções de (T125). Este lema significa que as funções ψ_i são compatíveis, quer dizer, coincidem na interseção dos domínios, pois um domínio está incluído no outro.

A prova de (D6) pode ser formulada da seguinte maneira.

1. Existência. Temos que provar que dada $\theta: \Omega \longrightarrow \Omega$ e $\omega \in \Omega$, existe uma função $\psi: N \longrightarrow \Omega$ tal que satisfaz:

$$(1) \quad \psi(1) = \omega;$$

$$(2) \quad \psi(n') = \theta\psi(n).$$

Para cada $n \in N$ existe um e somente um $\psi_n(n)$, onde ψ_n é a única função dada por (T125) para n . Então, definimos:

$$(3) \quad \psi(n) = \psi_n(n).$$

Temos que provar que a função ψ assim definida cumpre com (1) e (2). A função ψ cumpre com (1), pois $\psi(1) = \psi_1(1) = \omega$. A função ψ também cumpre com (2), pois $\psi(n') = \psi_{n'}(n') = \theta\psi_{n'}(n) = \theta\psi_n(n) = \theta\psi(n)$. A única igualdade não trivial é $\theta\psi_{n'}(n) = \theta\psi_n(n)$, que se cumpre por (L).

2. Unicidade. Sejam φ e ψ duas funções de N em Ω tais que (1) e (2)

se cumprem para ambas. Provaremos por indução completa que $\varphi(n) = \psi(n)$ para todo $n \in N$. Cumpre-se que $\varphi(1) = \omega = \psi(1)$. Além disto, $\varphi(n') = \theta\varphi(n) = \theta\psi(n) = \psi(n')$. A segunda igualdade cumpre-se pela hipótese indutiva.

Em relação a esta prova é interessante destacar três questões. Em primeiro lugar, (3) é um exemplo do procedimento de diagonalização, o qual será muito usado na posterior formalização do conceito de efetividade. Em segundo lugar, ao recorrer a (T125), esta prova de alguma maneira utiliza o conceito de recursão parcial, o qual só reaparecerá na história da recursão na prova do teorema de recursão transfinita de von Neumann de 1923 (ver VAN HEIJENOORT, 1967, p. 354) e no trabalho de KLEENE (1938) (ver GANDY, 1988, p. 89 para mais comentários históricos). Em terceiro lugar, Dedekind foi claramente consciente do fato de que para provar a existência de funções definidas indutivamente não é suficiente supor que o domínio da função seja uma cadeia; é necessário supor que o domínio da função seja um conjunto simplesmente infinito. Caso contrário, podem-se encontrar contra-exemplos (ver DEDEKIND, 1969, p. 373–374). Desta maneira, resulta uma diferença entre (D5) e (D6), pois (D5) pode ser formulado para cadeias, sem que estas sejam conjuntos simplesmente infinitos, por exemplo, da maneira feita pelo próprio Dedekind (Teorema 59 já mencionado). Esta questão pode ser estudada no trabalho de HENKIN (1960).

(D7): Todos os conjuntos simplesmente infinitos são similares entre si (Teorema 132). Dedekind enunciou este resultado com o conceito de similaridade, cuja definição é que dois conjuntos S e R são similares se existe

uma função φ de S tal que $\varphi[S] = R$ (Definição 32). Porém, ele não apenas provou a existência de uma função que cumpria este requisito; ele provou a existência de um isomorfismo. No comentário 134 do texto, Dedekind é perfeitamente consciente do significado deste isomorfismo, no sentido que todo teorema a respeito dos elementos de S pode ser traduzido, utilizando a função do isomorfismo, em um teorema a respeito de R .

Concluindo nossas observações a respeito da obra de Dedekind é interessante notar que, embora ela tenha sido útil para a posterior formalização do conceito de efetividade, o autor não estava preocupado com esta questão. Dedekind também não pensava que fosse necessário executar seu trabalho de fundamentação usando exclusivamente definições decidíveis no sentido de Kronecker, quer dizer, definições que fornecem procedimentos para decidir, dado um argumento qualquer, se ele cumpre ou não a condição da definição. Isto pode ser ilustrado pelas seguintes passagens, tomadas do começo do ensaio de Dedekind que estamos comentando, onde se examina a natureza da noção de conjunto (*sistema* na terminologia de Dedekind):

“Ein solches System S (...) ist als Gegenstand unseres Denkens ebenfalls ein Ding (...); es ist vollständig bestimmt, wenn von jedem Ding bestimmt ist, ob es Element von S ist oder nicht.” (DEDEKIND, 1969, p. 345)

Para esclarecer o sentido deste texto, Dedekind acrescenta a seguinte nota de rodapé:

“Auf welche Weise diese Bestimmtheit zustande kommt, und ob wir einen Weg kennen, um hierüber zu entscheiden, ist für alles Folgende gänzlich gleichgültig; die zu entwickelnden allgemeinen Gesetze hängen davon gar nicht ab, sie gelten unter allen Umständen. Ich erwähne dies ausdrücklich, weil Herr Kronecker vor kurzem (...) der freien Begriffsbildung in der Mathematik gewisse Beschränkungen hat auferlegen wollen, die nicht als berechtigt anerkenne;...” (DEDEKIND, 1969, p. 345)

Na próxima seção apresentaremos o trabalho de um lógico que estava interessado em fazer precisamente o que não preocupava a Dedekind, quer dizer, fornecer uma fundamentação da aritmética usando somente definições decidíveis, também denominadas *finitísticas*.

1.2 Skolem

Em 1919 Thoralf Skolem (1887–1963) escreveu um artigo, publicado em 1923, no qual afirma que se pode fundamentar uma parte considerável da matemática, em particular a aritmética, sem usar os quantificadores universal e existencial, ou seja, sem usar variáveis ligadas. Segundo Skolem, esta fundamentação pode ser efetuada usando “o modo recursivo de pensar”, o qual satisfaz os requisitos de Kronecker. Constatemos:

“... diese Arbeit ist aber eine konsequent finitistische; sie ist auf dem Prinzip KRONECKERS gebaut, daß eine mathematische Bestimmung

dann und nur dann eine wirkliche Bestimmung ist, wenn sie mit Hilfe endlich vieler Versuche zum Ziele führt." (SKOLEM, 1970, p. 188)

A estratégia recursiva de Skolem incluía definições recursivas primitivas de relações e funções aritméticas, e demonstrações mediante o princípio de indução.

Consideremos um exemplo simples para apreciar a técnica que Skolem utiliza repetidamente. É habitual a seguinte definição da relação de divisibilidade:

$$(D) \quad D(a, b) \text{ se } (\exists x) (a = bx).$$

Esta apresenta um inconveniente, pois não proporciona, *por si mesma*, um procedimento para decidir, dados dois números naturais quaisquer a e b , se a é ou não divisível por b . O melhor que fornece é um procedimento *semi-decidível*, a saber, o de testar, caso a caso, a equação $a = bx$ substituindo sucessivamente x por $1, 2, \dots$ até que a equação se cumpra. Se a é divisível por b , então esse procedimento finaliza. Em caso contrário, nunca tem fim. Em resumo, a observação de Skolem é que as definições habituais das relações e funções aritméticas básicas não proporcionam procedimentos decisórios, não satisfazendo as mencionadas exigências de Kronecker. Como resolver este inconveniente? Skolem afirma que basta, em primeiro lugar, considerar resultados já demonstrados. No caso da divisibilidade, consideremos o seguinte fato:

$$(R) \quad \text{Se } a = bx, \text{ então } x \leq a.$$

(R) permite substituir o “definiens” de (D), que tem a variável x ligada e sem limite, por uma fórmula com quantificador existencial limitado, dando lugar à seguinte definição:

$$(D') \quad D(a, b) \text{ see } (\exists x)_{1 \leq x \leq a} (a = bx).$$

Esta definição sim apresenta um procedimento decisório. Skolem dá mais um passo. Define indutivamente a relação ternária $D(a, b, c)$, a qual expressa que a é igual ao produto de b por um número entre 1 e c :

$$\Delta(a, b, 1) \text{ see } a = b;$$

$$\Delta(a, b, c + 1) \text{ see } \Delta(a, b, c) \text{ ou } a = b(c + 1).$$

A seguir, pode-se definir a divisibilidade do seguinte modo:

$$(D'') \quad D(a, b) \text{ see } \Delta(a, b, a).$$

Tem-se assim um procedimento decisório que é visivelmente recursivo.

É interessante acrescentar que Skolem pensava sempre ser possível passar de (D') a (D''):

“Da nun die SCHRÖDER'sche Aussagensumme in dieser Definition eine endliche ist, kann sie selbst wieder durch Rekursion definiert werden, und dadurch läßt sich zuletzt die Anwendung einer scheinbaren Veränderlichen ganz vermeiden.” (SKOLEM, 1970, p. 161)

Esta idéia equivale a enunciar sem demonstração um conhecido resultado da teoria das funções recursivas:

Se a relação $R(x_1, \dots, x_n, y)$ é recursiva, então a relação $(\exists y)_{y < z} R(x_1, \dots, x_n, y)$ também é recursiva.

Skolem pensava algo análogo a respeito do quantificador universal:

“Eben weil diese Aussagenprodukte endlich sind, können sie rekurrend definiert werden, wodurch die scheinbaren Veränderlichen völlig vermieden werden.” (SKOLEM, 1970, p. 173)

Estas observações de Skolem podem nos levar em direção à tese de Church. É óbvio que Skolem pensava que toda função recursiva (primitiva) era finitária. A respeito do recíproco, Webb sugere que

“It is quite possible that both Skolem and Hilbert believed in 1923 that *all* effective or finitary functions could be defined by primitive recursion.” (WEBB, 1980, p. 185)

Porém, não encontramos nenhum texto de Skolem que pudesse confirmar este ponto de vista, assim como Webb não pôde.

Finalmente, completamos a descrição do trabalho de Skolem. Além de definir recursivamente a divisibilidade, faz o mesmo com a soma, a relação de menor estrito, o produto, a diferença truncada, a divisão, o máximo divisor comum, o mínimo múltiplo comum, a propriedade de ser primo, o fatorial,

etc. Entre estas aparece uma definição duplamente recursiva, ou seja, recursiva relativa a duas variáveis. Trata-se da definição 17, onde estabelece-se o sentido da relação ternária $P(x, y, z)$, significando que x é o produto de y números primos todos menores ou iguais a z . Por meio destas definições, Skolem obtém teoremas elementares da teoria dos números, chegando a demonstrar a existência de infinitos números primos e o teorema fundamental da aritmética, a saber, que todo número maior do que 1 pode ser decomposto de maneira única como produto de fatores primos. Skolem também demonstra a equivalência entre o princípio de indução matemática e o princípio de indução por curso de valores. Analogamente, demonstra que é equivalente usar qualquer das definições indutivas correspondentes. Além disto, demonstra o princípio do número mínimo: se um número tem uma propriedade, então existe o menor número que tem esta propriedade. Finalmente, introduz o operador *Min* limitado, o qual, como é conhecido, é recursivo primitivo. Skolem o diferencia do operador *Min* sem limite.

1.3 Ackermann

Wilhelm Ackermann (1896–1962) contribuiu na teoria da recursão com o descobrimento de uma função efetivamente calculável que não é recursiva primitiva. Forneceu uma prova construtiva de existência de uma tal função. Porém, no artigo onde se apresenta este resultado, intitulado “Zum Hilbertschen Aufbau der reellen Zahlen”, a noção de calculabilidade efetiva não é mencionada. O título refere-se a uma construção das funções da teoria de números desenvolvida por David Hilbert (1862–1943) no trabalho intitulado

“Über das Unendliche”, onde, por outro lado, já se anuncia o mencionado resultado de Ackermann (ver VAN HEIJENOORT, 1967, p. 388).

Não incluímos Hilbert na história das funções recursivas, porque a mencionada construção seria a sua única contribuição pontual nesta área, mas ela é considerada obscura e, aparentemente, não foi muito desenvolvida fora de sua escola (ver GANDY, 1988, p. 100). Provavelmente, Hilbert pensava encontrar, dentro desta construção, uma função efetivamente calculável que resolvesse positivamente o problema de decisão do cálculo de predicados. Possivelmente, Hilbert acreditava que o décimo problema de sua lista de 1900 também pudesse ser resolvido positivamente e, em geral, que o ambiente matemático ainda não estava preparado para fornecer uma definição de calculabilidade efetiva (ver GANDY, 1988, p. 65). Isto explicaria que a sua escola não tenha se preocupado em obter uma definição do mencionado conceito, que é necessária só se se acredita em soluções negativas. Isto também explicaria o fato de que nenhum membro da escola de Hilbert tenha encontrado a solução do problema de decisão do cálculo de predicados (ver GANDY, 1988).

A seguir apresentamos um resumo do artigo de Ackermann. A mencionada construção hilbertiana das funções da teoria dos números se baseia na seguinte tipificação. As funções do tipo 1 são aquelas cujos argumentos e valores são números naturais; as do tipo 2 são aquelas cujos argumentos incluem pelo menos uma função de tipo 1 e nenhuma função de tipo maior que 1; e assim seguindo para funções de tipo n . As funções de tipo maior ou igual a 2 atualmente são denominadas “funcionais”. Resta acrescentar

que as funções de todos estes tipos são definidas pela aplicação das regras de substituição e recursão, a partir de um conjunto apropriado de funções iniciais. O seguinte é um exemplo da regra de recursão definindo uma função de tipo 2:

$$\rho_c(f(c), a, 0) = a;$$

$$\rho_c(f(c), a, n + 1) = f(\rho_c(f(c), a, n)).$$

A letra c subscrita liga a variável de f . Intuitivamente, $\rho_c(f(c), a, n)$ é a iteração n -ária de f sobre a , isto é:

$$\rho_c(f(c), a, n) = \underbrace{f(f(\dots f(a)\dots))}_{n \text{ vezes}}.$$

Chamamos a atenção sobre o fato de que a recursão está aplicada sobre um argumento numérico.

Dada esta tipificação, podem-se construir as funções com argumentos exclusivamente numéricos, as quais ficam classificadas em diferentes grupos. O primeiro grupo contém as funções que podem ser definidas pelo uso exclusivo de funções do tipo 1, ou seja, coincide exatamente com o conjunto das funções recursivas primitivas. Em geral, o n -ésimo grupo contém as funções que somente podem ser definidas se se inclui alguma função de tipo n , mas funções de tipo maior são prescindíveis.

Para os propósitos de Hilbert era imprescindível que nenhum grupo fosse vazio. Em 1925 Ackermann já havia encontrado uma função que pertence ao segundo grupo. Este resultado foi publicado em 1928, e é fácil

generalizá-lo para um grupo qualquer. A função de Ackermann se constrói a partir da definição da seguinte função φ de três argumentos:

$$\varphi(a, b, 0) = a + b;$$

$$\varphi(a, b, n + 1) = \rho_c(\varphi(a, c, n), \alpha(a, n), b);$$

onde ρ é a função de tipo 2 definida acima e $\alpha(a, n)$ é a seguinte função de tipo 1:

$$\alpha(a, n) = \begin{cases} 0 & \text{se } n = 0; \\ 1 & \text{se } n = 1; \\ a & \text{caso contrário.} \end{cases}$$

Para compreender a função φ , observemos que

$$\varphi(a, b, 0) = a + b;$$

$$\varphi(a, b, 1) = ab;$$

$$\varphi(a, b, 2) = a^b;$$

$$\varphi(a, b, 3) = a^{a^{a^{\cdot^{\cdot^{\cdot^a}}}}} \left. \vphantom{a^{a^{a^{\cdot^{\cdot^{\cdot^a}}}}} } \right\} b \text{ vezes.}$$

Intuitivamente podemos ver que, para cada n , a função de duas variáveis $\varphi(a, b, n)$ é uma função recursiva primitiva. O que ocorre se tomamos a função $A(a) = \varphi(a, a, a)$ definida a partir de $\varphi(a, b, n)$ por substituição? Para ter uma idéia da velocidade de crescimento de A , basta levar em conta que, embora os três primeiros valores sejam simplesmente 0, 1 e 4, o valor de A para 3 já é 27^{27} . É razoável pensar que dada uma função recursiva

primitiva f qualquer, se tomamos um n_0 suficientemente grande, para todo $n > n_0$ ocorrerá que $f(n) < A(n)$, ou seja, A cresce mais rápido que qualquer função recursiva. Esta é a intuição que Ackermann formalizou em 1925 para demonstrar que A é uma função que pertence ao grupo 2. Ackermann chamou a atenção sobre o fato de que ele não admitia recursões sobre mais de uma variável. Caso contrário, a função A pertenceria ao grupo 1, pois se poderia definir φ sem usar funções de tipo 2, da seguinte maneira:

$$\varphi(a, b, 0) = a + b;$$

$$\varphi(a, 0, n + 1) = \alpha(a, n);$$

$$\varphi(a, b + 1, n + 1) = \varphi(a, \varphi(a, b, n + 1), n).$$

Nesta formulação a terceira equação apresenta uma recursão dupla.

É interessante acrescentar que em 1935 Rózsa Péter usou outro método para chegar à mesma conclusão que Ackermann. O procedimento foi o seguinte: fazer uma enumeração efetiva das funções recursivas primitivas unárias e depois diagonalizar, i.e. definir $D(x) = f_x(x) + 1$, onde f_x é a x -ésima função recursiva primitiva unária. É claro que D não pode pertencer à lista das funções recursivas primitivas unárias. Porém, D é efetiva. Portanto, D é uma função efetiva não primitiva. (ver PÉTER, 1935, p. 49–53)

1.4 Gödel

Gödel desenvolveu a teoria das funções recursivas com o fim de obter seus conhecidos resultados de incompletude de 1931. Efetuou as seguintes contribuições:

- (G1) Definição das funções e relações recursivas primitivas;
- (G2) Prova de propriedades elementares das funções recursivas;
- (G3) Prova de que algumas funções e relações elementares da teoria de números são recursivas primitivas;
- (G4) Prova de que as funções e relações utilizadas para aritmetizar os sistemas formais são recursivas primitivas;
- (G5) Formulação do conceito de *entscheidungsdefinitheit*;
- (G6) Prova de que toda relação recursiva primitiva é numeralmente expressável.

Segue-se um comentário a respeito de cada um destes pontos.

(G1): A definição das funções e relações recursivas primitivas é efetuada sem fazer referência a nenhum formalismo: uma função f de teoria de números é recursiva primitiva se existe uma sucessão finita de funções de teoria de números $f_1, f_2, \dots, f_n$ que termina com f e tal que cada uma das f_j da sucessão está definida recursivamente a partir de duas funções precedentes, ou resulta da substituição dos argumentos de uma das funções preceden-

tes por algumas das funções precedentes, ou é uma função constante ou a função sucessor. Previamente, Gödel define a regra de recursão da maneira conhecida e permite explicitamente que nem todas as variáveis do “lado esquerdo” apareçam do “lado direito”, tanto na regra de substituição como na regra de recursão. Esta liberalidade explica porque não necessita das funções identidade $U_j^n(x_1, \dots, x_n) = x_j$ (com $0 < j \leq n$) que as vezes se usam para definir as funções recursivas. A definição de Gödel coincide exatamente, salvo questões estilísticas, com as definições que se usam atualmente.

(G2): Gödel prova que se R e S são relações recursivas, então também o são as relações $\neg R$, $R \vee S$ e $R \& S$. Em segundo lugar, se as funções $f(x_1, \dots, x_n)$ e $g(x_1, \dots, x_n)$ são recursivas, então também o é a relação $f(x_1, \dots, x_n) = g(x_1, \dots, x_n)$. Em terceiro lugar, se a função $f(x_1, \dots, x_n)$ e a relação $R(y, x_1, \dots, x_n)$ são recursivas primitivas, então também o são as relações

$$(\exists y)[y \leq f(x_1, \dots, x_n) \& R(y, x_1, \dots, x_n)]$$

e

$$(\forall y)[y \leq f(x_1, \dots, x_n) \rightarrow R(y, x_1, \dots, x_n)]$$

e a função

$$\mu y[y \leq f(x_1, \dots, x_n) \& R(y, x_1, \dots, x_n)].$$

Estas propriedades elementares das funções recursivas também têm ficado presentes nas exposições habituais da teoria da recursão.

(G3): Depois de observar que é fácil verificar que as funções $x + y$, xy e x^y e as relações $x < y$ e $x = y$ são recursivas, Gödel prova que outras funções e relações elementares de teoria de números, como a função fatorial e a relação de divisibilidade, também são recursivas primitivas.

(G4): Prova do caráter recursivo primitivo de quarenta funções de teoria de números. Neste caso, trata-se de funções e relações que podem ser formuladas em termos de uma aritmetização do sistema formal, a qual foi desenvolvida por Gödel com vistas a seus resultados sobre incompletude e consistência.

(G5): O adjetivo *entscheidungsdefinit*, que poderia ser traduzido como *definido na sua decidibilidade*, foi traduzido por Gödel em 1965 ao inglês por *decidable*. Trata-se da noção conhecida atualmente como relação *numeralmente expressável* de KLEENE (1952, p. 195) ou de relação *expressável* de MENDELSON (1979, p. 134). Pode-se formular da seguinte maneira:

Uma relação $R(x_1, \dots, x_n)$ de teoria de números é *numeralmente expressável no sistema formal S* se existe uma fórmula $A(x_1, \dots, x_n)$ (cujas variáveis livres são exclusivamente as variáveis distintas $x_1, \dots, x_n$) tal que para números naturais quaisquer $k_1, \dots, k_n$:

- (a) se $R(k_1, \dots, k_n)$, então $\vdash_S A(\bar{k}_1, \dots, \bar{k}_n)$;
- (b) se $\neg R(k_1, \dots, k_n)$, então $\vdash_S \neg A(\bar{k}_1, \dots, \bar{k}_n)$.

(G6): A proposição de que toda relação recursiva primitiva é numeralmente expressável no sistema formal definido por Gödel pode ser provada

por indução exigindo somente destreza técnica, considerável no caso da regra de recursão. Também é o caso de um resultado que tem sobrevivido na literatura *standard* (ver KLEENE, 1952, p. 297 e MENDELSON, 1979, p. 151). Se Gödel tivesse disposto em 1931 de seu conceito de recursão geral de 1934, também teria podido provar a proposição inversa, quer dizer, que toda relação numeralmente expressável é recursiva geral.

É interessante indicar que na prova original do resultado indicado em (G6), Gödel aplicou a indução sobre o que denominou *grad* (*Stufe*, em alemão) de uma função recursiva, quer dizer:

“Die Länge der kürzesten Reihe von φ_i , welche zu einer rekursiven Funktion φ gehört, ...” (GÖDEL, 1986, p. 158)

Esta noção de grau não é construtiva, como foi observado por Kleene (ver GÖDEL, 1986, p. 131–132). Portanto, seu uso invalidaria as pretensões construtivas de Gödel. Porém, é possível realizar induções com o fim de obter o resultado indicado, sem necessidade de baseá-las no conceito de grau. Para provar, por exemplo, que todas as funções recursivas primitivas são numeralmente representáveis, basta provar que todas as funções iniciais são representáveis e que as operações de substituição e recursão preservam a representabilidade, quer dizer, que se as funções operandos às quais estas operações se aplicam são numeralmente representáveis, então também o são as funções valores destas operações. Esta é a argumentação mais usual nestes casos, usada também por Gödel no seu trabalho “On Undecidable Propositions of Formal Mathematical Systems” de 1934 sobre a mesma questão, argumentação que

não usa o conceito de grau de uma função recursiva (ver GÖDEL, 1986, p. 358-359).

Também é útil efetuar alguns comentários sobre a nota introdutória do trabalho de Gödel de 1931 publicada por van Heijenoort. Considere-se a apresentação deste autor sobre o resultado indicado em (G6):

"The proof that every primitive recursive number-theoretic predicate is numeralwise representable in P . That is, the predicate holds of some given numbers if and only if a definite formula of P is provable whenever its free variables are replaced by the symbols that represent these numbers in P ." (VAN HEIJENOORT, 1967, p. 593)

Esta passagem contém certa dose de imprecisão. Em primeiro lugar, uma simples questão terminológica, é mais apropriado usar o termo *expressável* em lugar de *representável*, pois na literatura é comum usar *representável* para funções e *expressável* para predicados. Em segundo lugar, a definição dos predicados numeralmente expressáveis em um sistema formal P está exposta com pouco cuidado. Na realidade, de acordo com o texto de Gödel, um predicado é numeralmente expressável se existe uma fórmula F de P tal que 1) se o predicado vale para certos números, então pode-se provar em P a fórmula que resulta da substituição das variáveis livres de F pelos símbolos que representam os números dados e 2) se o predicado não vale para certos números, então pode-se provar em P a negação da fórmula mencionada no caso anterior. Isto coincide com a definição de relações *numeralmente*

expressáveis fornecida em (G5). Se se entende literalmente a definição de van Heijenoort pode-se confundí-la com a seguinte noção:

Uma relação $R(x_1, \dots, x_n)$ de teoria de números é *resolúvel* no sistema formal S se existe uma fórmula $A(x_1, \dots, x_n)$ (cujas variáveis livres são exclusivamente as variáveis distintas $x_1, \dots, x_n$) tal que para números naturais quaisquer $k_1, \dots, k_n$:

- (a) $\vdash_S A(\bar{k}_1, \dots, \bar{k}_n)$ ou $\vdash_S \neg A(\bar{k}_1, \dots, \bar{k}_n)$;
- (b) $R(k_1, \dots, k_n)$ se e só se $\vdash_S A(\bar{k}_1, \dots, \bar{k}_n)$.

É fácil provar que resolubilidade implica expressabilidade numeral. Pode-se também facilmente provar a afirmação inversa se se supõe que S é consistente (ver KLEENE, 1952, p. 296 para uma prova desta equivalência).

Para poder entender bem o próximo comentário, é bom ter em conta que Gödel introduz o conceito de predicado numeralmente expressável sem usar a expressão *numeralmente expressável* nem nenhuma outra. Um pouco mais adiante em relação à citação anterior de van Heijenoort, ele menciona alguns comentários de Gödel, um dos quais é apresentado da seguinte maneira:

“He [Gödel] introduces predicates that are *entscheidungsdefinit* (in the translation below these are called *decidable* predicates, at the author’s suggestion).” (VAN HEIJENOORT, 1967, p. 593)

A impressão de qualquer leitor atento é que van Heijenoort pensa que os predicados que aqui chama *decidíveis* não têm relação, a princípio, com os predicados que havia chamado *numeralmente expressáveis* e que comentamos no parágrafo anterior. Nada mais distante da verdade. Os predicados *entscheidungsdefinit* são exatamente os predicados que acabamos de chamar *numeralmente expressáveis*. Van Heijenoort devia ter dito, em todo caso, simplesmente que Gödel introduz o nome *entscheidungsdefinit* para os predicados que já havia apresentado previamente sem dar-lhes nenhum nome, pois Gödel não usou a denominação *expressabilidade numeral*, a qual aparentemente provém de KLEENE (1952, p. 195).

Acreditamos que estes comentários podem ser úteis para que qualquer leitor da nota introdutória de van Heijenoort atente para os seguintes pontos:

- (a) não confunda os conceitos de expressabilidade numeral e resolubilidade;
- (b) não seja conduzido a pensar que os conceitos de expressabilidade numeral e entscheidungsdefinitheit (decidibilidade) não são um e o mesmo.

1.5 Herbrand

Na obra de Jacques Herbrand (1908–1931) existem três alusões às noções de calculabilidade efetiva e função recursiva, todas elas do ano de 1931. A primeira alusão ocorre em uma nota não assinada sobre a sua própria tese de doutorado. Em uma descrição do que chama de *intuicionismo em sua forma extrema*, o autor inclui a seguinte passagem:

- (H1) “toutes les fonctions introduites devront être effectivement calculables pour toutes les valeurs de leurs arguments, par des operations décrites entièrement d’avance;” (HERBRAND, 1968, p.210).

Esta passagem, onde segundo GANDY (1988, p. 73) ocorre pela primeira vez o termo *efetivamente calculável*, aparentemente só foi publicada no ano de 1968 e, conforme veremos mais adiante, provavelmente não foi conhecida por Gödel.

A segunda alusão foi feita em uma carta dirigida a Gödel, extraviada posteriormente, a qual continha uma sugestão para definir a noção de função recursiva. Segundo Gödel, a definição proposta por Herbrand foi a seguinte:

- (H2) “One may attempt to define this notion as follows: If φ denotes an unknown function, and $\psi_1, \dots, \psi_k$ are known functions, and if the ψ 's and φ are substituted in one another in the most general fashions and certain pairs of the resulting expressions are equated, then, if the resulting set of functional equations has one and only one solution for φ , φ is a recursive function.” (GÖDEL, 1986, p. 368).

Gödel, em uma carta dirigida a van Heijenoort datada de 23 de abril de 1963, interpretou que a existência e a unicidade de φ da qual se fala em (H2) deveria ser obtida *construtivamente*, embora Herbrand não tenha se referido ao assunto (ver VAN HEIJENOORT, 1967, p. 619). Pode-se notar que em (H2) falta a menção das regras ou “operações descritas inteiramente em avanço” conforme expressado em (H1). Gödel obteve a sua definição de recursão geral

justamente acrescentando a (H2) duas restrições, a primeira das quais introduz duas regras, que em inglês denominam-se *substitution* e *replacement*. Por causa desta ausência, Gödel julgou que não era apropriado considerar Herbrand como o introdutor da noção de função recursiva (VAN HEIJENOORT, 1982, p. 74). É interessante acrescentar que KALMÁR (1955) encontrou uma função recursiva no sentido de (H2), porém não recursiva geral no sentido de Gödel.

A terceira alusão encontra-se no seu último artigo, dedicado à prova da consistência de um sistema formal de aritmética elaborado por ele mesmo, o qual inclui, além dos axiomas de Peano, o seguinte grupo de axiomas (hipóteses na linguagem de Herbrand):

- (H3) “On pourra aussi introduire un nombre quelconque de fonctions $f_1x_1x_2\dots x_{n_i}$ avec des hypotheses telles que
- a) *Elles ne contiennent pas de variables apparentes;*
 - b) *Considérées intuitionnistiquement, elles permettent de faire effectivement le calcul de $f_ix_1x_2\dots x_{n_i}$ pour tout système particulier de nombres; et l'on puisse démontrer intuitionnistiquement que l'on obtient un résultat bien déterminé*” (HERBRAND, 1968, p. 226–227).

Herbrand acrescenta a seguinte nota de rodapé para esclarecer o significado da expressão “considerados intuicionisticamente” que aparece no começo de b):

(H4) “Cette expression signifie: traduites en langage ordinaire, considérées comme une propriété des entiers, et non comme un pur symbole.”
(HERBRAND, 1968, p. 227)

Da mesma maneira que em (H2), em (H3) Herbrand não fala de “operações descritas inteiramente de antemão”. Em 1934 Gödel afirmou que (H2) e (H3) são trivialmente equivalentes na matemática intuicionista. Além disto, ele diz não saber se (H3) é ou não é equivalente à noção de recursividade geral (GÖDEL, 1986, p. 368). Nesta última questão Gödel não esclarece se está falando da matemática clássica ou da intuicionista. No ano de 1964 a situação não tinha se modificado (VAN HEIJENOORT, 1982, p. 74). Por outro lado, em uma carta a van Heijenoort datada de 14 de agosto de 1964, Gödel interpreta (H4) como se desconhecesse (H1):

“Herbrand, de son côté, exclut explicitement la spécification de règles formelles de calcul par la locution ‘considérées intuitionnistiquement’.”
(VAN HEIJENOORT, 1982, p. 74)

Parece que não se pode interpretar Herbrand como se ele quisesse excluir regras formais, dado que em (H1) ele pede explicitamente “operações descritas inteiramente de antemão”. Assim pensa van Heijenoort quando apresenta a seguinte pergunta:

“... Herbrand n’avait-il pas écrit quelques mois plus tôt que les calculs devaient se faire ‘par des opérations décrites entièrement d’avance’? Il n’avait pas donné une liste de ces opérations, mais peut-on dire que

la note 5 [H4] exclut catégoriquement la possibilité d'une telle liste?"

(VAN HEIJENOORT, 1982, p. 76)

Uma explicação da omissão de regras explícitas em (H3) por parte de Herbrand pode ser atribuída ao fato de que (H3) pertence a um sistema formal de aritmética cuja consistência Herbrand queria provar, ao qual, portanto, não deveriam poder se aplicar os resultados de Gödel sobre consistência de 1931, já conhecidos por Herbrand no momento da publicação do texto ao qual pertencem (H3) e (H4).

Em resumo, as contribuições de Herbrand parecem se reduzir a (H2), que contém uma idéia que foi usada por Gödel para o seu posterior conceito de recursão geral, um dos primeiros conceitos formais para a noção intuitiva de efetividade.

Este processo histórico de formação do conceito de recursão geral comentado neste capítulo culminou em 1934 com o mencionado conceito de recursão geral. Detemos o nosso relato sobre a história da recursão no ano de 1931, porque começaremos, no próximo capítulo, a partir desta data um histórico da tese de Church.

Capítulo 2

Histórico da tese de Church

2.0 Church, Gödel e Kleene

No começo da década de 30, o lógico norte-americano Alonzo Church (1902–) estava interessado em desenvolver um programa de fundamentação da matemática baseado no conceito de função e que não incluía estratégias como a teoria de tipos de *Principia Mathematica*. Trabalhou em conjunto com seus doutorandos Stephen Kleene (1909–) e John Rosser (1907–). Quando os resultados de Gödel de 1931 foram conhecidos, Church passou a se preocupar com a consistência de seu sistema. Algum tempo depois, Kleene e Rosser provaram a sua inconsistência (ver KLEENE, 1981a, p. 52 para alguns comentários sobre este assunto). Porém, resultou possível separar um subsistema que não possuía o mencionado inconveniente: o cálculo de conversão lambda, que consistia em poucas regras, e no qual os números naturais podiam ser representados. Neste contexto podem se definir as funções

de N^n em N que se denominam de λ -definíveis. Trata-se, intuitivamente, das funções cujos valores podem ser calculados com o formalismo da notação lambda.

No início de 1934, Gödel chegou em Princeton convidado para proferir um curso a respeito de seus resultados de 1931. O teorema IX do trabalho de GÖDEL (1986, p. 186) tornava quase certa uma solução negativa ao problema de decisão do cálculo de predicados enunciado e investigado pela escola de Hilbert. Para provar que existe um método de decisão, basta encontrar um. Mas para provar que não existe um tal método, é preciso ter uma definição rigorosa da noção de método de decisão (a rigor, pelo menos uma condição necessária de tal noção). Como vimos na seção 1.3, a comunidade matemática não tinha se preocupado em encontrar uma definição de calculabilidade efetiva. Church, consciente, graças ao trabalho de Kleene, da potência do seu conceito de função λ -definível, teve a idéia, pela primeira vez na história da matemática, de identificar o conceito intuitivo de função efetivamente calculável com um conceito formal, a saber, o conceito de função λ -definível. Comunicou oralmente sua idéia a Gödel. Este considerou a sugestão inteiramente insatisfatória, como o testemunha a seguinte passagem de Church sobre Gödel:

“My proposal that lambda-definability be taken as a definition of [effective calculability] he regarded as thoroughly unsatisfactory.” (DAVIS, 1982, p. 9)

Aproximadamente na mesma época, Gödel expôs, no curso mencionado, a noção de função recursiva geral, baseado na sugestão de Herbrand que citamos no capítulo anterior. Porém, ficará claro mais adiante que Gödel não pretendia estar formalizando o conceito intuitivo de calculabilidade efetiva, numa atitude cautelosa em contraste com a de Church.

Kleene, ao contrário de Gödel, tornou-se partidário da idéia de Church, depois de não ter conseguido refutá-la pelo potente método de diagonalização:

“When Church proposed this thesis, I sat down to disprove it by diagonalizing out of the class of the λ -definable functions. But, quickly realizing that the diagonalization cannot be done effectively, I became overnight a supporter of the thesis.” (KLEENE, 1981a, p. 59)

Antes de prosseguir com a história da tese de Church é conveniente esclarecer, a respeito da relação entre os trabalhos de Church e Kleene sobre a definibilidade lambda, a seguinte confusão de Webb:

“It was Kleene who undertook the detailed study of number theory in the λ -system, with particular attention to those functions of the integers which were ‘formally definable’ in it. Church had, in fact, already suspected that these ‘ λ -definable’ functions might provide a good approximation of effectiveness.” (WEBB, 1980, p. 212)

Em KLEENE (1987) nega-se a afirmação citada, se se interpretar que Church suspeitava que as funções λ -definíveis caracterizariam a efetividade *quando* Kleene começou a estudar o sistema lambda. Kleene acrescenta que

“In fact, Church came to this suspicion from contemplating Kleene’s results.” (KLEENE, 1987, p. 47)

Para justificar sua afirmação, Kleene indica que, por exemplo, quando em janeiro ou fevereiro de 1932 conseguiu, depois de muito esforço, provar que a função predecessor era λ -definível, Church já havia descartado esta possibilidade; de fato, o leitor poderá verificar que qualquer fórmula que λ -defina a função predecessor é bastante complicada (ver KLEENE, 1981a, p. 56–57 ou CHURCH, 1985, p. 31).

Após o curso proferido em Princeton, Gödel retornou à Áustria. Church e sua equipe continuaram trabalhando sobre o tema em Princeton. Finalmente, em 19 de abril de 1935 Church publicou um resumo onde anunciava um trabalho que demonstraria a existência de problemas indecidíveis. Para isto necessitava de uma definição da noção de calculabilidade efetiva, pelo qual Church incluiu no referido *abstract* a seguinte passagem:

“... it is maintained that the notion of an effectively calculable function of positive integers should be identified with that of a recursive function...” (CHURCH, 1935, p. 333)

De acordo com o anunciado, em março de 1936 Church publicou um artigo que continha uma *definição* do conceito de calculabilidade efetiva:

“We now define the notion, already discussed, of an *effectively calculable* function of positive integers by identifying it with the notion of a recursive function of positive integers (or of a λ -definable function of positive integers).” (DAVIS, 1965, p. 100)

A razão da frase entre parênteses é que nesse momento Church e Kleene já tinham demonstrado a equivalência entre os conceitos de definibilidade lambda, elaborado por Church, e de recursão geral de Gödel. A demonstração de que toda função λ -definível é recursiva geral foi feita simultânea e independentemente por Church e Kleene antes de abril de 1935 (ver DAVIS, 1965, p. 99), enquanto que o enunciado recíproco foi demonstrado por Kleene em junho de 1935 e publicado em KLEENE (1936). A prova do enunciado recíproco é mais complexa e originou o teorema da forma normal. Nesta prova Kleene introduz outro conceito formal equivalente aos conceitos de definibilidade lambda e recursão geral: o conceito de μ -recursão. De acordo com esta noção, uma função é μ -recursiva se é possível construí-la a partir de um conjunto de funções iniciais por meio das regras de substituição, recursão e μ -recursão, a última das quais se define do seguinte modo:

$$f(x_1, \dots, x_n) = \mu y [g(x_1, \dots, x_n, y) = 0]$$

onde (a) $g(x_1, \dots, x_n, y)$ é uma função de teoria de números tal que dados $x_1, \dots, x_n$ existe pelo menos um y tal que $g(x_1, \dots, x_n, y) = 0$ e (b) a expressão $\mu y R(x_1, \dots, x_n, y)$ denota o menor número y , se existe, tal que $R(x_1, \dots, x_n, y)$. Para evitar confusões terminológicas deve-se ter em conta

que este conceito de μ -recursão às vezes denomina-se simplesmente *recursão*, por exemplo, em MENDELSON (1979, p. 138).

O conceito de μ -recursão se define, como visto, sem necessidade de envolver nenhum sistema formal, da mesma maneira que o conceito mais restrito de função recursiva primitiva. Webb ressaltou este caráter da μ -recursão:

"Thus, unlike Gödel's general recursive functions, the μ -recursive functions are not defined by reference to any *formalism* for expressing equations whose calculations proceed by fixed rules governing the use of that formalism." (WEBB, 1980, p. 206–207)

Retornando a Church, é necessário acrescentar que ele forneceu duas justificações para a definição de calculabilidade efetiva apresentada. Mas, antes de expô-las, emitiu a seguinte advertência:

"This definition is thought to be justified by the considerations which follow, so far as positive justification can ever be obtained for the selection of a formal definition to correspond to an intuitive notion."
(DAVIS, 1965, p. 100)

Comentaremos as justificações de Church na seção 4.3.

2.1 Post

Em outubro de 1936 já se conheciam três conceitos equivalentes relacionados com a calculabilidade efetiva: a λ -definibilidade, a recursão geral e a μ -recursão. No mesmo mês foi recebido para publicação um artigo do lógico norte-americano Emil Post (1897–1954). Este trabalho apresentava um quarto conceito que provar-se-ia equivalente aos anteriores, mas do qual o autor pensava que tinha maior “fidelidade psicológica” que o conceito de Gödel:

“The writer expects the present formulation to turn out to be logically equivalent to recursiveness in the sense of the Gödel-Church development. Its purpose, however, is not only to present a system of a certain logical potency but also, in its restricted field, of psychological fidelity.”

(DAVIS, 1965, p. 291)

O conceito de Post baseia-se no seguinte: 1) uma fita infinita nos dois sentidos, dividida em quadrados que podem estar vazios ou conter uma marca, dos quais seleciona-se um como ponto inicial; 2) um conjunto de ordens baseadas nas seguintes operações: a) marcar o quadrado atual; b) apagar o quadrado atual; c) mover-se para o quadrado à direita, d) mover-se para o quadrado à esquerda; e) determinar se o quadrado atual está ou não marcado. Não fornecemos mais detalhes sobre as idéias de Post, pois são muito similares com as que apresentamos na próxima seção.

2.2 Turing

Também em 1936, mas independentemente dos trabalhos de Church, Kleene e Post, Alan Mathison Turing (1912–1954) publicou um trabalho onde apresentava outra noção equivalente à de recursão geral. Trata-se da noção de computabilidade, a qual requer certas máquinas teóricas que desde então se conhecem como máquinas de Turing. Em realidade, no referido trabalho não se fala de funções efetivamente calculáveis, senão de números (reais) computáveis, que Turing define assim:

“A number is computable if it differs by an integer from the number computed by a circle free machine.” (DAVIS, 1965, p. 119)

Este não é o lugar apropriado para extender-se sobre o significado do conceito de máquina livre de círculos. Só nos interessa que se registre o fato de que Turing, em uma atitude similar à de Church, simplesmente fornece uma definição.

As noções de recursividade e definibilidade lambda já tinham história em 1936. No primeiro caso, quase cinquenta anos; no segundo, cinco anos. Em contraposição, a noção de computabilidade de Turing é simultânea à tese de Church. Nasceu a partir de uma *análise* do conceito intuitivo de calculabilidade efetiva, exatamente o que faltou nos outros dois casos.

Antes de expor a referida análise, Turing emite a seguinte advertência:

"All arguments which can be given are bound to be, fundamentally, appeals to intuition, and for this reason rather unsatisfactory mathematically." (DAVIS, 1965, p. 135)

É interessante indicar que na seção 2.0 vimos que Church efetuou uma advertência similar, antes de apresentar as justificações de sua "definição" de calculabilidade efetiva.

Turing analisa o conceito de calculabilidade efetiva na situação de um computador *humano* que realiza cálculos. (O uso do itálico deve-se a querermos enfatizar algo não usual: chamar a uma pessoa de computador.) Os elementos que considera relevantes são os estados mentais do computador, e o meio (por exemplo, papel) onde escreve símbolos. Turing considera ser possível supor, sem perda de generalidade, que: a) o meio é uma fita unidimensional potencialmente infinita e dividida em quadrados; b) o número de símbolos a ser impresso é finito; c) o número de quadrados observados está limitado; d) o número de estados mentais do computador é finito; e) as operações do computador constituem uma ordem discreta, e f) as operações são a troca de um símbolo de um dos quadrados observados junto com uma possível troca do estado mental, e a troca de um dos quadrados observados por outro (o qual está no máximo a uma certa distância fixa dos quadrados previamente observados) junto com uma possível troca do estado mental.

Turing fornece argumentos para justificar que não existe perda de generalidade ao efetuar os supostos mencionados. A respeito de a), indica ser possível supor que o meio é um papel quadriculado bidimensional, o qual se

pode reduzir ao nível unidimensional. De fato, Turing também está supondo que o meio é discreto. A respeito de b), afirma que se o número de símbolos fosse infinito, então haveria símbolos que difeririam tão pouco como fosse desejado, o que os faria indistinguíveis. Ele elabora um argumento topológico para demonstrar a referida afirmação. Turing não faz nenhuma afirmação para justificar que c) não implica perda de generalidade, mas parece óbvio que um observador humano não poderia observar um número arbitrariamente grande de quadrados. De qualquer maneira, o observador pode realizar outras observações, se desejar observar outros quadrados. A respeito de d), o comentário de Turing é similar ao da suposição b): se o número de estados mentais fosse infinito, então haveria estados mentais que difeririam entre si tão pouco como fosse desejado, o que os faria indistinguíveis. Turing não afirma e), mas isto encontra-se implícito na sua argumentação. Finalmente, a respeito de f), parece difícil encontrar operações que não possam ser descritas em termos das indicadas.

2.3 Debate entre Post, Church e Kleene

Post, em contraste com Church, se opôs a usar o termo “definição” no caso do conceito de calculabilidade efetiva. Para que se possa entender bem a posição de Post, o citamos detalhadamente:

“The writer expects the present formulation to turn out to be logically equivalent to recursiveness in the sense of the Gödel-Church development. Its purpose, however, is not only to present a system of a certain

logical potency but also, in its restricted field, of psychological fidelity. In the latter sense wider and wider formulations are contemplated. On the other hand, our aim will be to show that all such are logically reducible to formulation 1. We offer this conclusion at the present moment as a *working hypothesis*. And to our mind such is Church's identification of effective calculability with recursiveness.* Out of this hypothesis, and because of its apparent contradiction to all mathematical development starting with Cantor's proof of the non-enumerability of the points of a line, independently flows a Gödel-Church development. The success of the above program would, for us, change this hypothesis not so much to a definition or to an axiom but to a *natural law*.

Actually the work already done by Church and others carries the identification considerably beyond the working hypothesis stage. But to mask this identification under a definition hides the fact that a fundamental discovery in the limitations of the mathematicizing power of Homo Sapiens has been made and blinds us to the need of its continual verification." (DAVIS, 1965, p. 291)

O nosso ponto de vista é que Post interpretou mal a Church. Quando Post fala de Church como se este estivesse ocultando um fato importante, parece-nos que Post pensava que Church tinha fornecido uma *definição nominal*, quer dizer, uma simples abreviação. Porém, isto não é compatível com o fato de que Church tenha apresentado varias justificações para a sua definição logo em seguida à sua formulação. Considere-se o seguinte comentário de Church acerca de sua definição de função efetivamente calculável, citada acima:



"This definition is thought to be justified by the considerations which follow, so far as positive justification can ever be obtained for the selection of a formal definition to correspond to an intuitive notion."
(CHURCH, 1936, p. 100)

A presença de tais justificações indica que Church tinha tentado fornecer uma *explicação* no sentido de CARNAP (1956, p. 7-8), quer dizer, uma correspondência entre um termo formal e outro informal ou intuitivo. Esta distinção entre abreviações e explicações, que provem de Frege, o qual usa a terminologia de definições construtivas e analíticas (ver FREGE, 1969, p. 227), é importante de um ponto de vista filosófico. Concluimos que a crítica de Post a Church é injustificada, porque Church usou o termo "definição" no sentido de explicação e não no sentido de abreviação. No entanto, no texto de Church ocorre uma imprecisão que poderia ter motivado uma interpretação errônea: quando na última citação Church fala de "formal definition" na verdade deveria ter dito "formal notion".

Qual foi a reação de Church a respeito da comentada objeção de Post? Não esteve de acordo com ela e continuou usando o termo "definição" (ver CHURCH, 1937, p. 43 e 1938, p. 227). Mas Kleene cedeu parcialmente, alguns anos depois, à terminologia de Post. Depois de indicar que todas as funções recursivas gerais são "subjativa" ou "intuitivamente" efetivamente calculáveis, Kleene acrescenta o seguinte:

"Conversely, as a heuristic principle, such functions (predicates) as have been recognized as being effectively calculable (effectively decida-

ble), and for which the question has been investigated, have turned out always to be general recursive, or, in the intensional language, equivalent to general recursive functions (general recursive predicates). This heuristic fact, as well as certain reflections on the nature of symbolic algorithmic processes, led Church to state the following thesis. The same thesis is implicit in Turing's description of computing machines.

THESES I. *Every effectively calculable function (effective decidable predicate) is general recursive.*

Since a precise mathematical definition of the term effectively calculable (effective decidable) has been wanting, we can take this thesis, together with the principle already accepted to which it is converse, as a definition of it for the purpose of developing a mathematical theory about the term. To the extent that we have already an intuitive notion of effective calculability (effective decidability), the thesis has the character of an hypothesis—a point emphasized by Post and by Church. If we consider the thesis and its converse as definition, then the hypothesis is an hypothesis about the application of the mathematical theory developed from the definition. For the acceptance of the hypothesis, there are, as we have suggested, quite compelling grounds.” (DAVIS, 1965, p. 274)

Este texto merece as seguintes observações. Em primeiro lugar, nele ocorre pela primeira vez a expressão “tese”. (A expressão completa “tese de Church” ocorre pela primeira vez em KLEENE (1952).) Em segundo lugar, é claro que Kleene tem em mente a pequena polêmica entre Church e Post que já comentamos. A análise do último parágrafo da citação parece propor duas

alternativas a respeito do caráter da tese de Church. Felizmente, outro trecho de Kleene confirma esta interpretação:

“The thesis may be considered a hypothesis about the intuitive notion of effective calculability, or a mathematical definition of effective calculability; in the latter case, the evidence is required to give the theory based on the definition the intended significance.” (KLEENE, 1952, p. 318–319)

Nossa leitura deste trecho é que a primeira opção de Kleene corresponde à explicação que mencionamos em nossa crítica a Post, enquanto que a opção que Kleene chama de definição matemática corresponde ao que chamamos de definição nominal ou abreviação. Em qualquer caso, o que Church chamou de definição não é o que Kleene está chamando de definição matemática, mas corresponde ao que Kleene chama de hipótese. Ademais, Kleene refere-se a uma definição (no sentido de abreviação) do termo “efetivamente calculável” usando o termo “recursivo geral”. Esta opção parece-nos inútil, uma vez que não se está abreviando nada; simplesmente ocorre uma substituição de um termo por outro. Isto está fora da prática matemática habitual.

Daqui em diante quando nos referirmos à tese de Church, estaremos pensando no seguinte enunciado:

(TC) Uma função é efetivamente calculável se e somente se é recursiva geral.

Esta formulação coincide totalmente com a usada originariamente em 1935 e requer alguns esclarecimentos. Em primeiro lugar, usamos o termo “função” como abreviação de “função de N^n em N ”. Em segundo lugar, a noção de calculabilidade efetiva é de caráter intuitivo e refere-se a um procedimento que, dados n números naturais como entrada, termina depois de um número finito de passos proporcionando um número natural como saída. Em terceiro lugar, fica claro que (TC) tenta conectar um conceito intuitivo com outro de natureza formal.

Capítulo 3

Questões históricas

3.0 Havia Gödel proposto a tese de Church?

Vimos que Church enunciou a sua tese usando o conceito de função recursiva geral que Gödel tinha apresentado em 1934 durante o curso ministrado em Princeton. Neste curso Gödel também forneceu o seguinte comentário a respeito da constatação de que as funções recursivas primitivas podem ser computadas mediante um procedimento finito:

“The converse seems to be true, if besides recursion ... recursions of other forms (e.g., with respect to two variables simultaneously) are admitted. This cannot be proved, since the notion of finite computation is not defined, but it serves as a heuristic principle.” (GÖDEL, 1986, p. 348)

No mesmo trabalho, Gödel dá uma solução ao problema do significado da expressão “toda função recursiva”. Esta situação origina a pergunta que intitula esta seção.

Na primeira versão da introdução ao trabalho de Gödel de 1934 Davis sugeriu que Gödel já tinha alcançado a tese de Church no ano mencionado (ver DAVIS, 1982, p. 8). Mas, a resposta de Gödel foi categórica:

“... it is *not true* that footnote 3 is a statement of Church’s Thesis. The conjecture stated there only refers to the equivalence of “finite (computation) procedure” and “recursive procedure.” However, I was, at the time of these lectures, not at all convinced that my concept of recursion comprises all possible recursions; and in fact the equivalence between my definition and Kleene’s ... is not quite trivial.” (DAVIS, 1982, p. 8)

A nota de rodapé mencionada por Gödel é a primeira citação desta seção. Nela está claro que o conceito de procedimento finito de computação ainda não estava definido. Agora, considerando a citação de 1982, descobrimos que também não estava definido o outro conceito, ou seja, o conceito de procedimento recursivo. Gödel não estava seguro que o seu conceito de recursão geral incluísse todas as recursões possíveis. Na nota de rodapé Gödel tinha falado de heurística no seguinte sentido: se se encontrar uma formalização das recursões mais gerais, então ter-se-á uma versão formal da noção intuitiva de procedimento finito de computação (calculabilidade efetiva na linguagem de Church). A nota de rodapé sugere uma equivalência. Mas, não se trata de

uma equivalência entre um conceito formal e outro intuitivo, como ocorre na tese de Church. Trata-se de uma equivalência entre *dois* conceitos intuitivos, ou seja, os conceitos de procedimento finito de computação e de recursão mais geral, o último dos quais deve ser distinguido do conceito formal de recursão geral de Gödel. Esta interpretação já foi feita em WEBB (1980, p. 188–9). Por tudo que foi colocado, a pergunta que intitula esta seção tem resposta negativa.

3.1 Por que Gödel não aceitou em 1934 a tese de Church?

Além de constatarmos que em fevereiro ou março de 1934 Gödel não aceitava (TC) em termos da definibilidade lambda, podemos observar que ele tinha uma proposta própria, de acordo com o seguinte testemunho de Church sobre Gödel:

“My proposal that lambda-definability be taken as a definition of [effective calculability] he regarded as thoroughly unsatisfactory. I replied that if he would propose any definition of effective calculability which seemed even partially satisfactory I would undertake to prove that it was included in lambda-definability. His only idea at that time was that it might be possible, in terms of effective calculability as an undefined notion, to state a set of axioms which would embody the generally accepted properties of this notion, and to do something on that basis.” (DAVIS, 1982, p. 9)

A idéia de Gödel, apresentada por Church de forma um pouco resumida, era obter um conjunto de axiomas a partir de uma *análise* do conceito de calculabilidade efetiva, de maneira similar à obtenção dos axiomas para os números naturais de Dedekind a partir da análise do conceito intuitivo correspondente. (Para confirmar esta interpretação do texto de Church pode-se ver WANG, 1991, p. 200.) Porém, Gödel não alcançou o desenvolvimento do referido projeto, pois as suas exigências foram satisfeitas quando conheceu o trabalho de Turing:

“Turing’s work gives an analysis of the concept of ‘mechanical procedure’ (alias ‘algorithm’ or ‘computation procedure’ or ‘finite combinatorial procedure’). This concept is shown to be equivalent with that of a ‘Turing machine’.” (DAVIS, 1965, p. 72)

Pelo que vimos na seção 2.2, é claro que Turing não efetuou nenhuma axiomatização. Entretanto, é indubitável que, usando a linguagem de Church, Turing fez uma análise das “propriedades geralmente aceitas da noção de efetividade”, a qual desembocou na noção de computabilidade, que o mesmo Turing demonstrou ser equivalente à noção de definibilidade lambda (ver TURING, 1937).

É interessante notar que o ponto de vista de Post era similar ao de Gödel. Já nos anos vinte, embora sem nada publicar, Post ocupou-se da noção geral de indecidibilidade. Para que os seus resultados possuíssem a significação que ele desejava, era necessário que estabelecesse o caráter universal de sua noção de conjunto de seqüências arbitrariamente geradas.

Esse estabelecimento era, na realidade, a versão de Post da tese de Church. Podemos constatar que as exigências de Post eram similares às de Gödel:

“Establishing this universality is not a matter of mathematical proof, but of psychological analysis of the mental processes involved in combinatory mathematical processes.” (DAVIS, 1965, p. 418)

“But for full generality a complete analysis would have to be made of all the possible ways in which the human mind could set up finite processes for generating sequences.” (DAVIS, 1965, p. 408)

Independentemente de que Post aceitasse a análise de Turing, parece claro que esta é semelhante à análise pedida por Post.

Como explicar as exigências de Gödel e Post? Odifreddi analisa esta questão. Segundo ele, as atitudes dos lógicos mencionados resultariam de uma espécie de platonismo:

“[Post] proposed the identification of the notions of a set of strings effectively generable on one side, and generable by canonical systems on the other. [This is] indirectly equivalent to Church’s Thesis (for partial functions). However, Post had a platonist philosophical view, and saw his proposal as something that had to be proved somehow, by a kind of ‘psychological analysis of the mental processes involved in combinatorial mathematical processes’.” (ODIFREDDI, 1989, p. 105)

É sabido que Gödel se reconhecia como platônico (ver WANG, 1991, p. 112). Porém, isso não significa que todos os seus atos devam ser explicados invocando a sua postura filosófica. Além disso, nem Gödel nem Post mencionam o tema do platonismo quando manifestam suas opiniões acerca de temas vinculados com a tese de Church. Por outro lado, Kleene e Davis, que se ocuparam destas questões históricas detalhadamente, também não mencionam em nenhum momento a relevância do platonismo para as mesmas.

Dado não aceitarmos a explicação de Odifreddi, devemos proporcionar uma outra. Acreditamos que o nosso problema pode ser resolvido considerando um episódio da história das funções recursivas. Na nossa resposta anterior vimos que em 1934 Gödel pensava que estava bem encaminhado com o conceito de recursão, embora duvidasse da abrangência de seu conceito de recursão geral. É claro que Gödel não duvidava que todas as funções recursivas gerais fossem efetivamente calculáveis. Esta é uma comprovação óbvia que para o caso das funções primitivas já está presente, por exemplo, em 1888:

“Zur praktischen Berechnung bieten sich Rekursionsformeln dar.”

(CANTOR, 1966, p. 434)

É a afirmação recíproca que provoca dúvidas. Antes de Ackermann, podia se pensar que todas as funções efetivamente calculáveis eram recursivas primitivas, como vimos que Webb o sugere a respeito de Skolem e Hilbert. Entretanto, Ackermann encontrou um contra-exemplo. Posteriormente, quando

Gödel conseguiu generalizar a definição de recursão, como podia estar seguro de que não apareceria um segundo Ackermann?

O caso da definibilidade lambda é similar. Adiante veremos que, aparentemente, também Church duvidava da generalidade do seu conceito de definibilidade lambda. O fato de que todas as funções efetivamente calculáveis submetidas a teste resultassem ser lambda definíveis, não é evidência suficiente para pensar que toda função efetivamente calculável é lambda definível. Tanto neste caso como no caso da recursão, a opção de efetuar uma demonstração está ausente, pois o conceito de efetividade é intuitivo.

Qual é a diferença do trabalho de Turing? A sua análise estabeleceu uma *conexão* entre o conceito intuitivo de efetividade e um conceito formal (computabilidade). É interessante indicar que, além de Gödel, tanto Church como Kleene deram valor à contribuição de Turing:

“For rendering the identification with effective calculability the most plausible—indeed, I believe compelling—Turing computability has the advantage of aiming directly at the goal...” (KLEENE, 1981a, p. 61)

“Perhaps the most convincing form in which this definition of an effective process can be put results from the adoption of an idea of Turing.” (CHURCH, 1938, p. 227)

Por outro lado, Kleene também denunciou o aspecto negativo—no sentido que estamos comentando nesta seção—da noção de definibilidade lambda:

“Who would have guessed that this formulation, generated as I have described to clarify the notation for functions, has implicit in it the notion (not known in mathematics in 1931 in a precise version) of all functions on the positive integers (or on the natural numbers) for which there are algorithms?” (KLEENE, 1981a, p. 54)

“... λ -definability is not intrinsically persuasive (the thesis using it was supported not by the concept itself but rather by results established about it) ...” (KLEENE, 1981b, p. 49)

Estas citações juntamente com o conhecimento da gênese do conceito de definibilidade lambda, são suficientes para concluir que a noção de definibilidade lambda não foi gerada a partir de uma análise do conceito intuitivo de calculabilidade efetiva. Simplesmente, “resultou” que as funções lambda definíveis incluíram as efetivamente calculáveis.

Portanto, Gödel não aceitou (TC) em 1934, porque ninguém havia fornecido uma análise (no sentido fornecido por Turing da noção intuitiva de calculabilidade efetiva, análise que justificaria a identificação deste conceito com algum conceito formal. Isto não parece ter nenhum vínculo com pontos de vista platônicos.

3.2 Por que Church não elegeu o seu próprio conceito de definibilidade lambda para enunciar a sua tese?

Esta pergunta resulta do fato de que as noções de definibilidade lambda e recursão geral são equivalentes. Vimos que Church propôs a Gödel definir o conceito de calculabilidade efetiva mediante a noção de definibilidade lambda, durante conversações em fevereiro de 1934, de acordo com documentos citados em DAVIS (1982). Gödel julgou que a proposta era totalmente insatisfatória. Aproximadamente na mesma época, provavelmente um pouco depois, Gödel também não concordava com uma definição mediante a noção de recursividade geral. Durante o ano 1934, Church e Kleene demonstraram que as funções lambda definíveis são recursivas. Church anunciou a sua “tese” no dia 19 de abril de 1935, usando o conceito de recursividade geral de Gödel, em lugar do seu próprio conceito de definibilidade lambda. Considere-se a seguinte passagem do *abstract* de Church:

“And it is maintained that the notion of an effectively calculable function of positive integers should be identified with that of a recursive function, since other plausible definitions of effective calculability turn out to yield notions which are either equivalent to or weaker than recursiveness.” (CHURCH, 1935, p. 333)

Quais eram essas outras definições plausíveis de calculabilidade efetiva? Parece indiscutível que uma delas tinha que usar a noção de definibilidade

lambda. Isto significa que em abril de 1935 Church duvidava que a referida noção fosse tão geral como o conceito de recursividade geral? É possível. Como dissemos, até esse momento só tinha sido demonstrado que todas as funções lambda definíveis são recursivas. Só em junho de 1935 Kleene demonstrou a afirmação recíproca.

Embora o trabalho de Church tenha sido publicado só no ano de 1936, Church manteve a posição do *abstract*, dando preferência à recursividade:

"We now define the notion, already discussed, of an *effectively calculable* function of positive integers by identifying it with the notion of a recursive function of positive integers (or of a λ -definable function of positive integers)." (DAVIS, 1965, p. 100)

Como explicar a atitude de Church, agora que já estava estabelecida a equivalência entre os dois referidos conceitos? O seguinte comentário de Kleene talvez a explique:

"I myself, perhaps unduly influenced by rather chilly receptions from audiences around 1933-35 to disquisitions on λ -definability, choose, after general recursiveness had appeared, to put my work in that format." (KLEENE, 1981a, p. 62)

Church deve ter experimentado recepções similares quando apresentava a noção de definibilidade lambda. Em contraste, a alternativa da recursão,

além de excluir simbolismos desconhecidos, já tinha uma tradição de quase 50 anos.

Outra explicação da atitude de Church poderia ser dada pelo fato de que as justificações que comentaremos na seção 4.2 talvez fossem mais complicadas usando a noção de definibilidade lambda. O seguinte texto de Kleene, enquanto sugere que a noção de recursão geral é mais intrinsecamente persuasiva que o conceito de definibilidade lambda, documentaria esta interpretação:

“Turing’s computability is intrinsically persuasive in the sense that the ideas embodied in it directly support the thesis that the functions encompassed are all for which there are algorithms; lambda definability is not intrinsically persuasive (the thesis using it was supported not by the concept itself but rather by results established about it) and general recursiveness scarcely so (its author Gödel being at that time not at all persuaded).” (KLEENE, 1981b, p. 49)

As vantagens da definibilidade lambda se reduziriam ao cálculo.

A seguir apresentamos um resumo da resposta para a pergunta que intitula esta seção. Quando Church escreveu o *abstract* do trabalho do ano 1936, tinha dúvidas a respeito da generalidade do conceito de definibilidade lambda. Quando efetuou a redação final, embora já conhecesse a equivalência entre ambos os conceitos, manteve a preferência pela recursividade, porque foi melhor recebida no ambiente matemático ou porque era mais facilmente justificável.

3.3 Qual foi o primeiro conceito formal a expressar a noção intuitiva de calculabilidade?

Entendemos esta pergunta independentemente do fato de que o conceito formal correspondente tenha sido *proposto* como versão do conceito intuitivo.

Os seguintes comentários de Kleene indicam que o conceito de definibilidade lambda foi o primeiro conceito formal com a propriedade mencionada:

“The notion of “ λ -definability” was *the first* of what are now accepted as equivalent exact mathematical descriptions of the class of the functions for which algorithms exist.” (KLEENE, 1981a, p. 52) (nossa ênfase)

“Who would have guessed that this formulation, generated as I have described to clarify the notation for functions, has implicit in it the notion (*not known in mathematics in 1931 in a precise version*) of all functions on the positive integers (or on the natural numbers) for which there are algorithms?” (KLEENE, 1981a, p. 54) (nossa ênfase)

Por outro lado, Turing, tinha outra opinião:

“Such a definition was first given by Gödel at Princeton in 1934 ...” (DAVIS, 1965, p. 160)

No entanto, se interpretarmos nossa questão da maneira indicada no primeiro parágrafo desta seção, tanto Kleene como Turing estão equivocados. O artigo de Gödel de 1931 sobre a incompletude da aritmética elementar incluía um conceito formal que corresponde ao conceito intuitivo de calculabilidade efetiva. Trata-se da noção de relações *entscheidungsdefinit*, mais conhecidas na bibliografia como relações expressáveis em uma teoria formal, que foram definidas na seção 1.4. No artigo mencionado, Gödel provou que as funções recursivas primitivas são expressáveis (Teorema V). Na verdade, pode ser provado de forma mais ampla que uma relação é recursiva geral se e somente se é expressável, o que não foi feito por Gödel, pois ele não precisava desta ampliação para as suas provas de incompletude. Gödel não estava interessado em formalizar o conceito de calculabilidade efetiva quando elaborou a noção de relação expressável. Porém, é interessante comentar que ele usou uma expressão que continha a idéia de decisão, ou seja, o verbo “entscheiden”, em lugar de usar algum termo correspondente às idéias de expressabilidade ou representabilidade. Igualmente, em 1965 sugeriu que fosse usada a palavra “decidable” na tradução de seu trabalho.

Em resumo, podemos responder à pergunta que intitula esta seção dizendo que o primeiro conceito formal que correspondeu à noção de calculabilidade efetiva foi o conceito de relação expressável de Gödel de 1931. Tanto Enderton (BARWISE, 1977, p. 534) como DAVIS (1982, p. 12) chamam atenção para este fato.

Capítulo 4

Questões conceituais

4.0 A tese de Church: uma conjectura?

Na seção 2.3 vimos que para o esclarecimento de (TC) foram utilizados os termos “definição” e “hipótese de trabalho”. Além disto, tem-se o caso de Thomas, o qual considera (TC) como muito próxima de uma conjectura:

“The basic piece of ‘heuristic evidence’, the lack of a known counterexample, provides no very convincing support for CT. Consider that evidence of this sort has never been thought convincing in mathematics. One could argue in the same fashion for the truth of, say, the Goldbach or the Fermat conjecture, but surely no one regards either of those conjectures as having a known truth-value. Indeed, if a mathematician even entertained mere suspicions as to the truth value of either of these conjectures, those suspicions would be founded on

intuition (perhaps arising from a study of suggestive results in number theory) but certainly not on the cornucopia of confirming cases available to him. Now, if we are to allow 'heuristic evidence' as convincing in the matter of the decision about CT, there ought to be some justification for treating *this* question as somehow very different from other mathematical questions, different in a way that licenses our taking serious account of 'heuristic' considerations. Such a justification does not seem to be forthcoming." (THOMAS, 1973, p. 56)

Em nossa opinião Thomas comete um erro grave ao equiparar (TC) com as conjecturas de Goldbach, Fermat ou, eventualmente, qualquer outra conjectura matemática. A tese de Church difere de uma conjectura matemática no seguinte sentido: contém um conceito intuitivo, o conceito de efetividade. Por outro lado, todas as conjecturas matemáticas são enunciados completamente formalizáveis, dos quais pode-se esperar uma demonstração ou uma refutação no futuro. Em contraste, não se pode esperar uma legítima prova de (TC), a menos que se troque o conceito intuitivo de efetividade por um conceito formal, modificando o caráter de (TC). Ademais, vimos nas seções 2.0 e 2.2 que nem Church nem Turing pretenderam que as suas argumentações a favor de (TC) fossem provas matemáticas no sentido estrito.

4.1 O uso da tese de Church

Já consideramos as *discussões* dos lógicos a respeito do caráter de (TC). Uma outra forma de abordar o caráter de (TC) é examinar a sua *utilização*.

Neste enfoque, o primeiro fato observável é que os lógicos utilizam somente a seguinte metade de (TC):

(TC1) Toda função efetivamente calculável é recursiva geral.

Esta utilização não pode ser interpretada no sentido de abreviação, uma vez que as abreviações têm sentido só no caso de serem enunciados bicondicionais. Portanto, (TC1) deve ser interpretada como hipótese de trabalho no sentido comentado na seção 2.3.

O segundo fato observável é que alguns autores utilizam (TC1) para concluir recursividade a partir de efetividade, enquanto outros utilizam-na para concluir não-efetividade a partir de não-recursividade, ou seja, utilizam o enunciado contra-recíproco. A primeira forma de trabalho pode ser exemplificada pelo seguinte método, que o autor denomina “prova pela tese de Church”:

“Give an informal (though rigorous) proof that the given informal algorithm is indeed an algorithm that serves to compute f . Then appeal to Church’s thesis and conclude immediately that f is URM-computable.” (CUTLAND, 1992, p. 68)

A noção de função URM-computável é equivalente à noção de recursão geral. A segunda forma de trabalho pode ser exemplificada pela seguinte passagem:

“... if a function is shown to be non-recursive, then by Church’s Thesis we conclude that it is non-algorithmic—and this is the only known way to reach that conclusion.” (BELL, MACHOVER, 1977, p. 259)

É claro que em ambas as passagens o verbo “concluir” não é utilizado no sentido de “derivar formalmente”. A diferença entre as duas formas de trabalho relaciona-se com uma maior ênfase no desenvolvimento da teoria intuitiva ou informal no primeiro caso (o exemplo de Cutland) e uma maior ênfase no desenvolvimento da teoria formal no segundo caso (o exemplo de Bell e Machover). Esta diferença de ênfase explica o fato que estamos indicando neste parágrafo, uma vez que toda teoria sobre a computabilidade necessita trabalhar com elementos formais e informais, mas se trabalhamos principalmente no campo informal, precisamos de (TC1) para transitar em direção à teoria formal; por outro lado, se trabalhamos principalmente na teoria formal, precisamos do contra-recíproco de (TC1) para transitar em direção à teoria informal. A diferença que estamos comentando não exclui a possibilidade de alguns autores utilizarem as duas formas, (TC1) e sua contra-recíproca. É o caso de Monk, o qual, ademais, utiliza (TC1) em uma forma fraca (MONK, 1976, p. 46).

Poder-se-ia pensar que a distinção entre (TC1) e seu contra-recíproco é irrelevante, uma vez que ambos os enunciados, dito informalmente, são logicamente equivalentes. Porém, é interessante observar que alguns autores têm criticado o uso de (TC1) sem estender a crítica também para o seu contra-recíproco. É o caso, por exemplo, de Epstein e Carnielli:

“To invoke Church’s thesis when ‘the proof is left to the reader’ is meant amounts to giving a fancy name to a routine piece of mathematics while at the same time denigrating the actual mathematics.” (EPSTEIN, CARNIELLI, 1989, p. 229)

Também é o caso de Manin, embora por outras razões:

“... there is a certain danger in this practice. It is possible that the habit of increasingly using informal arguments delayed the discovery of such a fundamental fact as the result that recursively enumerable sets and Diophantine sets coincide.” (MANIN, 1977, p. 184)

4.2 As justificações de Church

Nesta seção examinaremos duas provas que Church fornece para justificar sua tese (ver DAVIS, 1965, p. 100–102). Em primeiro lugar, prova-se que se uma função é tal que existe um algoritmo para calcular seus valores, então é recursiva geral. Em segundo lugar, prova-se que se uma função é calculável em um sistema de lógica simbólica, então é recursiva geral. Depois de apresentar estas provas, veremos que elas ensejam duas interpretações.

Examinemos, em primeiro lugar, a prova que usa a noção de algoritmo. Para simplificar, consideremos uma função F de um único argumento. Suponhamos que F é tal que existe um algoritmo para calcular seus valores. Para Church, isto significa que existe um procedimento tal

que, dado um argumento n , pode-se obter uma sucessão finita de expressões $E_{n_1}, E_{n_2}, \dots, E_{n_{r_n}}$ tal que a) E_{n_1} pode ser calculado efetivamente a partir de n e b) dados n e E_{n_i} , para $i < j+1$, pode-se saber efetivamente se $j = r_n$, em cujo caso se pode encontrar efetivamente $F(n)$; caso contrário pode-se calcular efetivamente $E_{n_{j+1}}$. A seguir, Church efetua uma numeração de Gödel das expressões e sucessões finitas de expressões. Depois, define as funções G e H da seguinte maneira, onde na definição de G a notação e_{n_i} indica o número de Gödel da expressão E_{n_i} :

$$G(n, x) = \begin{cases} e_{n_{k+1}} & \text{se } x \text{ é o número de uma subsequência} \\ & \text{inicial } E_{n_1}, E_{n_2}, \dots, E_{n_k} \text{ da sequência} \\ & \text{finita gerada pelo algoritmo para } F \\ & \text{com argumento } n \text{ e } k \neq r_n; \\ 10 & \text{se } x \text{ é o número da sequência finita ge-} \\ & \text{rada pelo algoritmo para } F \text{ com argu-} \\ & \text{mento } n; \\ 1 & \text{se } x \text{ não é número de nenhuma sub-} \\ & \text{sequência inicial } E_{n_1}, E_{n_2}, \dots, E_{n_k} \text{ da} \\ & \text{sequência finita gerada pelo algoritmo} \\ & \text{para } F \text{ com argumento } n. \end{cases}$$

$$H(n, x) = \begin{cases} G(n, x) & \text{se } G(n, x) \neq 10; \\ F(n) & \text{se } G(n, x) = 10. \end{cases}$$

Finalmente, Church expressa que se a calculabilidade efetiva de G e H implica que elas são recursivas (usando (TC)), então pode-se derivar a recursividade de F .

Em segundo lugar, examinemos a prova que usa a noção de função calculável em um sistema de lógica simbólica. Suponhamos que estamos tra-

balhando com um sistema que contem o símbolo de identidade, o símbolo $()$ para aplicar uma função de um inteiro positivo a seu argumento e as expressões $1, 2, 3, \dots$ para representar os inteiros positivos. Os teoremas do sistema são, do modo costumeiro, os axiomas do sistema e as expressões que resultam da aplicação das regras de inferência a estes axiomas. Church não especifica os axiomas e as regras de inferência do sistema ao qual está-se referindo, porém supõe que as regras de inferência são operações efetivamente calculáveis, que os conjuntos de axiomas e de regras de inferência são efetivamente numeráveis e que a relação entre os inteiros positivos e as expressões que os representam é efetivamente decidível. A seguir, Church interpreta que isto significa, dada uma numeração de Gödel das expressões do sistema, que as regras de inferência são operações *recursivas*, que os conjuntos de axiomas e de regras de inferência são *recursivamente* numeráveis e que a relação entre os inteiros positivos e as expressões que os representam é *recursiva*. (Isto, de acordo com nossa opinião, equivale a usar (TC).) A seguir, Church define as funções que chama de *calculáveis*: Seja F uma função de um inteiro positivo, F é calculável em um sistema se e somente se existe uma expressão f no sistema tal que $f(u) = v$ é teorema se e somente se $F(m) = n$, onde u e v são as expressões que denotam a m e n . Usando o fato de que o conjunto de teoremas do sistema é recursivamente numerável, pode-se provar que toda função calculável, no sentido que acabamos de definir, é recursiva.

Havendo exposto estas duas provas, Church expressa-se da seguinte maneira:

“Thus it is shown that no more general definition of effective calculability than that proposed above can be obtained by either of two methods which naturally suggest themselves ...” (CHURCH, 1936, p. 102)

A interpretação mais imediata deste texto é que Church pensa que as duas noções introduzidas, quer dizer, a de função tal que existe um algoritmo para calcular seus valores e função calculável em um sistema de lógica simbólica, são dois novos conceitos *formais* equivalentes aos já conhecidos de recursão geral e definibilidade lambda. Porém, esta interpretação parece estar equivocada, uma vez que em ambas as provas utilizam-se a noção informal de efetividade e (TC) para concluir recursividade das funções envolvidas. Além disto, Church, antes de introduzir as provas apresentadas, adverte:

“This definition is thought to be justified by the considerations which follow, so far as positive justification can ever be obtained for the selection of a formal definition to correspond to an intuitive notion.”
(CHURCH, 1936, p. 100)

Nesta citação “definition” refere-se a (TC). Portanto, as noções introduzidas nas provas que estamos examinando nesta seção não são totalmente formais.

Porém, estas noções guardam certa semelhança com noções formais equivalentes à de recursão geral. Em primeiro lugar, a noção de função tal que existe um algoritmo para calcular seus valores é parecida com a de função computável por uma máquina de Turing ou por um algoritmo de Markov.

Em particular, uma sucessão finita de expressões corresponde à noção de computação de uma máquina de Turing ou de um algoritmo de Markov.

Em segundo lugar, o conceito de função calculável coincide exatamente com a noção de função *precisável* (reckonable) de KLEENE (1952, p. 295). Trata-se do equivalente funcional do conceito de relação *resolúvel*, definido em 1.4:

Uma função $f(x_1, \dots, x_n)$ de teoria de números é *precisável* no sistema formal S se existe uma fórmula $A(x_1, \dots, x_{n+1})$ (cujas variáveis livres são exclusivamente as variáveis distintas $x_1, \dots, x_{n+1}$) tal que para números naturais quaisquer $k_1, \dots, k_{n+1}$:

$$f(k_1, \dots, k_n) = k_{n+1} \text{ sse } \vdash_S A(\bar{k}_1, \dots, \bar{k}_{n+1}).$$

A diferença entre a segunda justificação de Church e uma prova qualquer de que as funções precisáveis são recursivas reside unicamente em que Church não diz exatamente quais são os axiomas e as regras de inferência de seu sistema, o que permitiria provar recursividade sem necessidade de supor (TC).

Uma segunda interpretação das justificações de Church é a que corresponde a KLEENE (1952, p. 323). Parece-nos a interpretação correta. De acordo com ela, Church simplesmente teria apresentado dois métodos de definir funções de maneira efetiva a partir de outras funções. Neste caso, a argumentação de Church se interpreta como estabelecendo que estes métodos são recursivos, no sentido de produzir funções recursivas a partir de funções recursivas. Tratar-se-ia do tipo de evidência mais simples de (TC).

O fato de que o *status* das justificações de Church não é totalmente claro, tem dado lugar a objeções de circularidade em relação aos argumentos apresentados. É o caso de THOMAS (1973, p. 61), que interpreta que as duas provas que estamos comentando são intentos de provar (TC) usando (TC). Por outro lado, como pode-se interpretar a partir de textos citados no capítulo 2, nem Gödel nem Post ficaram satisfeitos com as justificações de Church.

A prova que usa a noção de algoritmo foi recuperada por outros lógicos, entre eles SHOENFIELD (1967, p. 120) e MONK (1976, p. 46), que a utilizam como uma espécie de demonstração intuitiva de (TC). Evita-se a objeção de circularidade argumentando que as funções G e H são relativamente simples e, portanto, a aplicação de (TC) a elas não tem o mesmo peso do que a aplicação de (TC) a todas as funções efetivamente calculáveis.

4.3 O caráter absoluto da computabilidade

Em 1936 Gödel publicou um resumo de uma conferência realizada na Universidade de Viena em 19 de junho do ano anterior. O resumo foi intitulado "Über die Länge der Beweisen" (Sobre a longitude das provas). Gödel referiu-se à importância do caráter *absoluto* da noção de computabilidade no seguinte comentário, o qual foi agregado ao final do resumo quando se estavam corrigindo as provas de impressão:

“Es läßt sich übrigens zeigen, daß eine Funktion, die in einem der Systeme S_i oder auch in einem System transfiniten Stufe berechenbar ist, schon in S_1 berechenbar ist, so daß also der Begriff ‘berechenbar’ in gewissem Sinne ‘absolut’ ist, während fast alle sonst bekannten metamathematischen Begriffe (z. B. beweisbar, definierbar etc.) sehr wesentlich vom zu Grunde gelegten System abhängen.” (GÖDEL, 1986, p. 398)

O autor refere-se a uma sucessão de sistemas formais. Excetuando detalhes difíceis de decidir devido a algumas imprecisões de Gödel, o sistema S_1 coincide com uma aritmética de Peano (Gödel o chama de aritmética clássica), e dado S_j , S_{j+1} define-se como a extensão de S_j por meio de variáveis e quantificadores de tipo $j + 1$, onde as variáveis de tipo 1 são as variáveis habituais dos números naturais, as variáveis de tipo 2 são variáveis de classes de números naturais, etc. Por outro lado, de acordo com a definição de Gödel, uma função $f(x)$ é *computável* em um sistema formal S se para cada numeral m existe um numeral n tal que $f(m) = n$ é demonstrável em S . Trata-se de uma variante de sua noção de *representability* de 1934. O estilo de Gödel é um pouco descuidado, uma vez que na definição de computabilidade que acabamos de expor não se distingue entre as funções e as expressões funcionais formais que representam as funções no sistema formal.

Pode-se usar a sucessão S_j para contrastar resultados sobre demonstrabilidade e computabilidade. Por um lado, cumpre-se que existem proposições de S_j não demonstráveis em S_j , mas que têm prova em S_{j+1} . Além disso, o resultado principal do trabalho expressa que existem fórmulas que

podem ser provadas tanto em S_j como em S_{j+1} , mas cujas provas mais curtas em S_{j+1} são muito mais curtas que as respectivas menores provas em S_j . Por outro lado, contrastando com estes resultados, cumpre-se que a computabilidade em qualquer dos sistemas S_j é independente do j , quer dizer, uma função é computável em S_j se e somente se é computável em S_k , onde j e k são números naturais quaisquer ou ainda números transfinitos. É neste sentido que Gödel fala do carácter absoluto da computabilidade. É interessante acrescentar que na apresentação do resultado citado sobre a longitude das provas Gödel expressa-se como se a computabilidade dependesse de cada sistema S_j :

“Zu jeder in S_i berechenbaren Funktion φ gibt es unendlich viele Formeln $f \dots$ ” (GÖDEL, 1986, p. 396)

Isto reforça a opinião de que Gödel só foi consciente do carácter absoluto da computabilidade, no momento que as provas de impressão de seu trabalho estavam sendo feitas.

Em 1946 Gödel voltou a se referir à questão do carácter absoluto da computabilidade:

“Tarski has stressed in his lecture (and I think justly) the great importance of the concept of general recursiveness (or Turing’s computability). It seems to me that this importance is largely due to the fact that with this concept one has for the first time succeeded in giving

an absolute definition of an interesting epistemological notion, i.e., one not depending on the formalism chosen.” (GÖDEL, 1990, p. 150)

Em 1965 acrescentou o seguinte comentário:

“To be more precise: a function of integers is computable in any formal system containing arithmetic, if and only if it is computable in arithmetic, where a function f is called computable in S if there is in S a computable term representing f .” (GÖDEL, 1990, p. 150)

Poder-se-ia interpretar que nestas citações, ao falar do caráter absoluto do conceito de computabilidade, Gödel tinha em mente a equivalência entre as noções de representabilidade, definibilidade lambda, recursão geral, μ -recursão e computabilidade por uma máquina de Turing. Esta é, por exemplo, a interpretação de ZIMBARG SOBRINHO (1987, p. 12).

Uma segunda interpretação é que Gödel estaria pensando no conceito de μ -recursão de Kleene. Por quê? Os conceitos de representabilidade, definibilidade lambda e recursão geral definem-se sempre relativamente a um sistema formal fixado. A μ -recursão, por outro lado, como vimos na seção 2.0, é um conceito que em sua própria definição prescinde de qualquer referência a um sistema formal. Neste sentido, é semelhante ao conceito de recursão primitiva. Gödel estava bem consciente de que para definir as funções recursivas primitivas não é preciso fazer referência a nenhum sistema formal:

“Wir schalten nun eine Zwischenbetrachtung ein, die mit dem formalen System P vorderhand *nichts zu tun hat*, und geben zunächst folgende

Definition: Eine zahlentheoretische Funktion ...” (GÖDEL, 1986, p. 156) (nossa ênfase)

Porém, é possível que Gödel estivesse pensando de outra maneira nas citações onde fala sobre o conceito absoluto de computabilidade.

A terceira interpretação é que Gödel estaria simplesmente reforçando suas idéias citadas de 1936, quer dizer, ressaltando a ausência, no caso da computabilidade, da relatividade a um sistema formal, evitando-se a estratificação que se produz nos casos da definibilidade e da demonstrabilidade. É esta a interpretação que Parsons fornece ao texto de Gödel (ver GÖDEL, 1990, p. 145)

Devido ao estilo extremamente cauteloso e sucinto dos textos de Gödel, é difícil decidir por uma interpretação. A primeira é improvável e a segunda teria merecido um texto mais específico. Portanto, talvez a terceira seja a interpretação correta.

4.4 Prova de incompletude usando a tese de Church

KLEENE (1987) fornece uma prova do teorema de incompletude de Gödel usando a tese de Church. A sua prova, de caráter não construtivo, é apresentada a seguir.

Seja S uma teoria formal de números como a descrita em MENDELSON (1979). Ela cumpre as seguintes condições:

(REP) Todas as funções recursivas primitivas são numeralmente representáveis em S .

(Ω CON) S é ω -consistente, i. e. se para todo número natural n se cumpre que $\vdash_S A(\bar{n})$, então $\not\vdash_S (\exists x)\neg A(x)$ (onde $\bar{n}$ é o numeral que representa a n em S).

Vamos provar que se S é ω -consistente, então é incompleta. Suponhamos que S é ω -consistente. Pelo absurdo, suponhamos que S é completa, quer dizer:

(COM) Para toda fórmula fechada A se cumpre que $\vdash_S A$ ou que $\vdash_S \neg A$.

Nestas suposições, vamos construir uma função efetiva não recursiva, em contradição com (TC).

Lembremos a enumeração efetiva com repetições das funções recursivas parciais unárias:

$$f_z(x) = \begin{cases} U(\mu y T_1(z, x, y)) & \text{se } (\exists y) T_1(z, x, y); \\ \text{não definida} & \text{se } \neg(\exists y) T_1(z, x, y). \end{cases}$$

Nesta enumeração T_1 é a relação ternária recursiva primitiva definida em KLEENE (1952, p. 281) e U a função recursiva primitiva definida em KLEENE (1952, p. 278). Consideremos a seguinte modificação desta enumeração:

$$g_z(x) = \begin{cases} U(\mu y T_1(z, x, y)) & \text{se } (\exists y) T_1(z, x, y); \\ 0 & \text{se } \neg(\exists y) T_1(z, x, y). \end{cases}$$

Em primeiro lugar, é claro que, para cada z , g_z está bem definida, i.e. para cada x existe um e só um valor $g_z(x)$. Em segundo lugar, é claro que trata-se de uma sucessão efetiva de funções, pois para cada z podemos encontrar efetivamente a função g_z . Em terceiro lugar, provaremos que a sucessão g_z inclui todas as funções recursivas gerais unárias. Seja h uma função recursiva geral unária. Pelo teorema da forma normal de KLEENE (1952, p. 288) existe um número natural e tal que:

$$(1) \quad (x)(\exists y) T_1(e, x, y);$$

$$(2) \quad (x)[h(x) = U(\mu y T_1(e, x, y))].$$

Se provamos que $h = g_e$, então ficara provado que toda função recursiva geral unária ocorre na sucessão. Para isso, basta provar que $h(x) = g_e(x)$ para todo x e N . Fixemos x . Por (2) se cumpre que $h(x) = U(\mu y T_1(e, x, y))$. Por (1) e a definição de g_z temos que $g_e(x) = U(\mu y T_1(e, x, y))$. Portanto, $h(x) = g_e(x)$. Em quarto lugar, provaremos que segundo nossas suposições (REP), (Ω CON) e (COM), todas as funções g_z são efetivamente calculáveis. Fixemos z e obtenhamos $\bar{z}$. Consideremos o seguinte algoritmo para a função g_z :

0. Ler x
1. Calcular $\bar{x}$
2. Calcular $NUM = \text{número da fórmula } \neg(\exists y)T_1(\bar{z}, \bar{x}, y)$
3. $n = 0$
4. Se $T_1(z, x, n)$, então $g_z(x) = U(n)$. Parar
5. Se n é número de uma prova, então se o número da última fórmula desta prova é NUM , então $g_z(x) = 0$. Parar
6. $n = n + 1$
7. Retornar ao passo 4

Provaremos que este algoritmo calcula os valores da função g_z . Fixemos um número natural arbitrário x . Suponhamos que $(\exists y)T_1(z, x, y)$, em primeiro lugar. Pela definição da relação T_1 , isto significa que existe um *único* y tal que $T_1(z, x, y)$. Chamemo-lo m . Então, a condição do passo 4 se cumpre para $n = m$. Portanto, neste caso o algoritmo fornece $g_z(x) = U(m)$. Mas $m = \mu y T_1(z, x, y)$. Portanto, $g_z(x) = U(\mu y (T_1(z, x, y)))$.

Resta provar que para todo $n < m$ não se cumpre a condição do passo 5. Na verdade, podemos provar que esta condição não se cumpre com nenhum n , pois pode-se provar, segundo nosso suposto, i.e. $(\exists y)T_1(z, x, y)$, que a fórmula $\neg(\exists y)T_1(\bar{z}, \bar{x}, y)$ não é provável em S . Como T_1 é numeralmente representável, a partir de (REP) e do fato que $T_1(z, x, m)$ segue-se que $\vdash_S T_1(\bar{z}, \bar{x}, \bar{m})$. Então, segue-se que $\vdash_S (\exists y)T_1(\bar{z}, \bar{x}, y)$. Então, como S

é consistente, dado que se cumpre (ΩCON) , temos que $\not\vdash_S \neg(\exists y)T_1(\bar{z}, \bar{x}, y)$. Então o número da fórmula $\neg(\exists y)T_1(\bar{z}, \bar{x}, y)$ nunca ocorrerá como número da última fórmula de uma prova, e, portanto, nunca se cumprirá a condição do passo 5.

Até agora foi provado que se $(\exists y)T_1(z, x, y)$, então o algoritmo apresentado fornece $g_z(x)$. Resta provar que também o fornece no caso contrário. Suponhamos, então, que $\neg(\exists y)T_1(z, x, y)$. Temos que provar que o algoritmo pára cumprindo as condições do passo 5 para algum valor de n . É evidente que nunca vai se cumprir a condição de 4. Se provamos que $\vdash_S \neg(\exists y)T_1(\bar{z}, \bar{x}, y)$, então, existirá um m tal que é o menor número que é número de uma prova desta fórmula, e se cumprirá a condição do passo 5 para m . Neste caso o algoritmo fornecerá 0, que coincide com o valor de $g_z(x)$. Portanto, neste caso, basta provar que $\vdash_S \neg(\exists y)T_1(\bar{z}, \bar{x}, y)$.

Da suposição que $\neg(\exists y)T_1(z, x, y)$ segue-se $(y)\neg T_1(z, x, y)$. Então, para todo $n \in N$, $\neg T_1(z, x, n)$. Por (REP), para todo $n \in N$, $\vdash_S \neg T_1(\bar{z}, \bar{x}, \bar{n})$. Por (ΩCON) segue-se $\not\vdash_S (\exists y)\neg\neg T_1(\bar{z}, \bar{x}, y)$. Então, $\not\vdash_S (\exists y)T_1(\bar{z}, \bar{x}, y)$. Então, pela suposição (COM), temos que $\vdash_S \neg(\exists y)T_1(\bar{z}, \bar{x}, y)$, como queríamos provar. Portanto, foi provado que o algoritmo calcula os valores da função g_z , e, portanto, g_z é efetivamente calculável para todo z .

Finalizando nossa argumentação, definimos $D(x) = g_x(x) + 1$. É claro que D é uma função unária efetiva que não pode estar na sucessão. Segue-se que D não é recursiva pelo que vimos em terceiro lugar. Então, D é uma função efetiva não recursiva, em contradição com (TC). Portanto, S é incompleta.

O texto de KLEENE (1987) contém essencialmente a mesma prova, mas existe um pequeno erro que indicamos agora. Kleene define a sucessão de funções $\bar{\varphi}_z(x)$, que corresponde às funções g_z da nossa versão, da seguinte maneira (ver KLEENE, 1987, p. 490):

$$\bar{\varphi}_z(x) = \begin{cases} U(y) & \text{se } T_1(z, x, y) \\ 0 & \text{se } \vdash_F (y) \neg T_1(\bar{z}, \bar{x}, y). \end{cases}$$

Mas as expressões $U(y)$ e $T_1(z, x, y)$ não têm sentido, pois as únicas variáveis livres podem ser z ou x . Kleene deveria ter escrito algo parecido com o seguinte:

$$\bar{\varphi}_z(x) = \begin{cases} U(\mu y T_1(z, x, y)) & \text{se } (\exists y) T_1(z, x, y); \\ 0 & \text{se } \vdash_F (y) \neg T_1(\bar{z}, \bar{x}, y). \end{cases}$$

A seguir, apresentamos uma prova de incompletude que é, aparentemente, mais direta e simples que a de Kleene, embora mantendo o caráter não construtivo. Em lugar de supor (COM), provaremos incompletude diretamente usando o fato muito conhecido de que na teoria S já mencionada o conjunto de teoremas é recursivamente enumerável (RNUM). Seja a seguinte função:

$$f(x) = \begin{cases} \mu y T_1(x, x, y) & \text{se } (\exists y) T_1(x, x, y); \\ 0 & \text{se } \neg(\exists y) T_1(x, x, y). \end{cases}$$

onde T_1 é a relação primitiva recursiva já comentada. Desde que Kleene o provou em 1936, sabe-se que f não é recursiva geral (ver DAVIS, 1965, p. 251). Portanto, usando (TC) segue-se que f não é efetivamente calculável. Consideremos agora o seguinte algoritmo:

0. Ler x
1. Calcular $\bar{x}$
2. $n = 0$
3. Se $T_1(x, x, n)$, então escrever n . Parar
4. Se o n -ésimo teorema é $\neg(\exists y)T_1(\bar{x}, \bar{x}, y)$, então escrever 0.
Parar
5. $n = n + 1$
6. Retornar ao passo 3

Como f não é efetivamente calculável, não existe algoritmo que calcula seus valores para argumentos arbitrários. O algoritmo apresentado calcula os valores de f para todo x tal que $(\exists y)T_1(x, x, y)$. No caso que $\neg(\exists y)T_1(x, x, y)$ a condição do passo 3 não se cumpre nunca. Como o algoritmo tem que falhar para algum x , para algum x o algoritmo não pode cumprir a condição do passo 4 para nenhum n , i.e. para nenhum n se cumpre que o n -ésimo teorema (na enumeração recursiva disponível em S) é a fórmula $\neg(\exists y)T_1(\bar{x}, \bar{x}, y)$. Como (por RNUM) na enumeração aparecem *todos* os teoremas de S , temos que $\not\vdash_S \neg(\exists y)T_1(\bar{x}, \bar{x}, y)$. Então, existe um número natural x tal que $\neg(\exists y)T_1(x, x, y)$ e tal que $\not\vdash_S \neg(\exists y)T_1(\bar{x}, \bar{x}, y)$. Segue-se que $(\exists y)T_1(\bar{x}, \bar{x}, y)$ é uma fórmula indecidível, pois $\not\vdash_S (\exists y)T_1(\bar{x}, \bar{x}, y)$ por (REP) e (Ω CON), e já temos que $\not\vdash_S \neg(\exists y)T_1(\bar{x}, \bar{x}, y)$. Portanto, temos provado a incompletude de S .

Sem dúvida nossa prova também não constrói uma fórmula indecidível, mas parece-nos mais direta que a de Kleene por não supor (COM) pelo absurdo; por outro lado parece-nos mais simples pelo fato de usar (RNUM) em lugar da enumerabilidade recursiva das funções parciais. Além disso, tem a vantagem de ser muito semelhante com uma argumentação enganosa apresentada por KALMÁR (1959) “contra a plausibilidade” de (TC). Esta semelhança será útil na análise do trabalho de Kalmár, o qual será apresentado na próxima seção.

Consideramos pertinente comentar que as provas de incompletude que apresentamos só parecem ter interesse teórico no estudo da relação entre a tese de Church e o teorema de incompletude de Gödel, pois não parecem ser mais simples que a prova original de Gödel de 1931, e, ademais, tem a desvantagem de não serem construtivas, e de usar mais uma hipótese, i.e. (TC).

4.5 A respeito de um argumento de Kalmár

Num colóquio sobre construtivismo em matemática realizado em Amsterdã no ano de 1957, o lógico húngaro László Kalmár apresentou um trabalho intitulado “An Argument against the Plausibility of Church’s Thesis” publicado em 1959. Depois de apresentar alguns esclarecimentos, Kalmár expressa que não considera (TC) como uma proposição que pode ser provada ou da qual pode-se encontrar um contra-exemplo. Kalmár considera que sua argumentação é de caráter pré-matemático. Além disso, ele diz que

não apresentará uma instância de função efetivamente calculável que não é recursiva geral. De fato ele apresenta exemplos de funções que não são recursivas gerais, a partir das quais chega-se a consequências “implausíveis” se for suposto que não são efetivamente calculáveis. Vejamos estes exemplos:

$$\psi(x) = \begin{cases} \mu y[\varphi(x, y) = 0] & \text{se } (\exists y)[\varphi(x, y) = 0]; \\ 0 & \text{se } \neg(\exists y)[\varphi(x, y) = 0]. \end{cases}$$

onde é suposto que φ é uma função recursiva geral tal que ψ não é recursiva geral. Kleene em 1936 forneceu exemplos de funções φ que cumprem esta condição, por exemplo, usando a relação $T_1(x, x, y)$ conforme fizemos na seção anterior. Nestas suposições, Kalmár pretende derivar as seguintes consequências implausíveis. Suponhamos um número natural p tal que $(\exists y)[\varphi(p, y) = 0]$. Então é óbvio que existe um algoritmo para calcular o valor $\psi(p)$:

0. $n = 0$
1. Se $\varphi(p, n) = 0$, então escrever n . Parar
2. $n = n + 1$
3. Retornar ao passo 1

Suponhamos a seguir um número natural p tal que a fórmula $\neg(\exists y)[\varphi(\tilde{p}, y) = 0]$ “pode-se provar por meios arbitrariamente corretos”. Kalmár pretende que também neste caso existe um algoritmo para calcular $\psi(p)$:

“... prove that no natural number y with $\varphi(p, y) = 0$ exists, which requires in any case but a finite number of steps, and gives immediately the value $\psi(p) = 0$.” (KALMÁR, 1959, p. 74)

Esta citação está mal redigida e nos parece muito obscura. Se existe uma prova por “meios arbitrariamente corretos” de que $\neg(\exists y)[\varphi(p, y) = 0]$, para p fixo, então se cumpre que $\neg(\exists y)[\varphi(p, y) = 0]$, e, portanto, o algoritmo, neste caso, simplesmente deve fornecer o valor 0, para coincidir com $\psi(p)$. MENDELSON (1963, p. 203) apresenta o algoritmo da mesma maneira, mas PÉTER (1967, p. 224) recai na confusão de Kalmár.

A partir da existência deste algoritmo, Kalmár pretende extrair a seguinte conclusão, usando a suposição de que, por (TC), ψ não é efetivamente calculável: existe um número natural p tal que $\neg(\exists y)[\varphi(p, y) = 0]$ e este fato não pode ser provado por “meios arbitrariamente corretos”. Kalmár considera que esta consequência de (TC) é muito implausível. Não percebemos a maneira de extrair tal conclusão sem usar a hipótese da enumerabilidade efetiva dos teoremas obtidos por “meios arbitrariamente corretos”, de modo similar à uma prova de incompletude apresentada na seção anterior.

Kalmár prossegue na sua argumentação raciocinando como se tivesse encontrado uma sentença absolutamente indecidível. Porém, ele simplesmente provou a *existência* de um número natural p tal que a sentença $(\exists y)[\varphi(\bar{p}, y) = 0]$ é indecidível. MENDELSON (1963, p. 203–204) e EPSSTEIN, CARNIELLI (1989, p. 234) também fazem esta crítica.

Kalmár obtém outras conclusões que não nos parece importante apresentar, dado que dependem das críticas que acabamos de mencionar.

A argumentação de Kalmár já foi criticada em 1957, previamente ao colóquio, por Kleene, de acordo com os comentários deste autor publicados só 30 anos depois:

“I will recall that I was present at the Amsterdam Colloquium of 1957 when my good friend László Kalmár presented his argument against the plausibility of Church’s thesis [...]; I immediately concluded, as fast as I heard it, that he had not given an effective procedure for deciding as to the truth or falsity of $(x) \neg T_1(n, n, x)$. He would not be able to tell me in advance in a finite communication (no matter how long we both should live) what set of atomic rules would completely govern the concrete steps in his search for proofs by ‘arbitrary correct means’ of $(x) \neg T_1(n, n, x)$. I refrained from embarrassing him at the Colloquium by asking him for them on the spot.” (KLEENE, 1987, p. 494–495)

Como se percebe neste texto, Kleene também não entendeu qual era o algoritmo de Kalmár para calcular $\psi(p)$ no caso que p fosse um número natural tal que a fórmula $\neg(\exists y)[\varphi(p, y) = 0]$ é provável por “meios arbitrariamente corretos”.

A argumentação de Kalmár foi posteriormente criticada por HOERING (1962) e MENDELSON (1963), os quais fizeram observações similares às nossas. Porém, em 1967, Péter, uma colega húngara de Kalmár, que

também era especialista na teoria de recursão, a quem nos referimos na seção 1.3, mantinha o argumento de Kalmár (ver PÉTER, 1967, p. 224). Posteriormente, em 1976, Péter ainda insistia nesta posição chegando inclusive a afirmar que a noção de efetividade não era formalizável:

“I myself agree with Kalmár’s conviction that effective computability is one of those notions the definition of which can never be considered complete in the course of the development of mathematics.” (PÉTER, 1981, p. 142)

Além do mais, ela também esperava que quando aparecessem contra-exemplos de (TC), os computadores pudessem ser modificados para computá-los. Em *broken english*:

“Let us hope, provided a counter-example to Church’s thesis is made known, then, one hopes, the technological means will develop to modify computers to enable them to compute such functions.” (PÉTER, 1981, p. 142)

Péter faleceu em 1977. Em 1978, foi publicado um livro do físico e epistemólogo Imre Lakatos (1922–1974), onde ele interpreta a argumentação de Kalmár como um exemplo de refutação em matemática (ver LAKATOS, 1987, p. 42). Esta interpretação de Lakatos parece ser inapropriada em dois sentidos: em primeiro lugar, a argumentação de Kalmár é insuficiente para refutar a plausibilidade de (TC); em segundo lugar, é discutível considerar

que (TC) forme parte da matemática da mesma maneira que qualquer teorema; é usual, conforme o mesmo Kalmár, considerar (TC) como de caráter pré-matemático. Na bibliografia posterior a 1987 não encontramos nenhuma defesa do ponto de vista de Kalmár. Podemos, portanto, identificar o que chamaríamos de *falácia húngara*, dado que somente membros da comunidade científica húngara a defendem.

É interessante acrescentar que o argumento de Kalmár foi previsto por Gödel já em 1929. A sua dissertação doutoral deste ano intitulada “Über die Vollständigkeit des Logikkalküls” e o trabalho de 1931 sobre incompletude contém breves esclarecimentos sobre esta questão (ver GÖDEL, 1986, p. 62–64 e 186, WEBB, 1980, p. 189–190 e WEBB, 1983, p. 327–328).

Por tudo o que foi colocado podemos concluir que o argumento de Kalmár não cumpre requisitos de clareza mínima e contém pelo menos o erro de supor a enumerabilidade efetiva de maneira oculta.

Apêndice

Hilbert e os membros de sua escola acreditavam que o *Entscheidungsproblem* e o décimo problema teriam soluções positivas. Além disso, muitos matemáticos também acreditavam que o desenvolvimento matemático alcançado na década dos anos vinte ainda não era suficiente para uma formalização do conceito de calculabilidade efetiva. Quando em 1931 Gödel alcançou seus resultados de incompletude, também inesperados para Hilbert e sua escola, obteve um corolário que significava que seria muito pouco provável que o *Entscheidungsproblem* tivesse solução positiva. Então ficou claro para os lógicos que o único caminho seria formalizar o conceito de calculabilidade efetiva. A rigor, bastava uma condição necessária de calculabilidade efetiva, mas isto era tão difícil como obter uma condição necessária e suficiente.

Nenhum membro da escola de Hilbert nem Gödel alcançaram conscientemente uma tal condição necessária. Quando Church observou, graças aos trabalhos de Kleene, que a definibilidade lambda constituía um formalismo muito poderoso, arriscou uma “proposta audaciosa”, i.e. (TC), e apresentou funções não efetivamente calculáveis e uma prova negativa do *Entscheidungsproblem*. As suas justificações não convenceram nem a Gödel nem a Post.

Este último apresentou suas críticas por escrito, dizendo que Church simplesmente tinha alcançado uma hipótese de trabalho. Porém, no outro lado do Atlântico, e de forma independente, Turing propôs uma análise do conceito de calculabilidade efetiva, chegando a uma formalização que provou ser equivalente às noções de definibilidade lambda e recursão geral. Também deu uma prova negativa do *Entscheidungsproblem*. A partir de então, considera-se que o conceito de máquina de Turing é o conceito formal “mais próximo” do conceito intuitivo de calculabilidade efetiva.

Referências

BARWISE, Jon (Ed.) *Handbook of Mathematical Logic*. Amsterdam: North-Holland, 1977.

BELL, John, MACHOVER, Moshé. *A Course in Mathematical Logic*. Amsterdam: North-Holland, 1977.

BUSSEY, W. The Origin of Mathematical Induction. *American Mathematical Monthly*, v. 24, p. 199–207, 1917.

CAJORI, Florian. *A History of Mathematics*. New York: Chelsea Publishing Company, 1991.

CANTOR, Georg. *Gesammenlte Abhandlungen mathematischen und philosophischen Inhalts*. Editado por Ernst Zermelo. Hildesheim: Georg Olms, 1966.

CARNAP, Rudolf. *Meaning and Necessity*. Chicago: The University of Chicago Press, 1956.

CHURCH, Alonzo. An unsolvable problem of elementary number theory, Preliminary Report (Abstract 205). *Bulletin of the American Mathematical Society*, v. 41, p. 332–333, 1935.

_____. Review de Post. *The Journal of Symbolic Logic*, v. 2, p. 43, 1937.

_____. The constructive second number class. *Bulletin of the American Mathematical Society*, v. 44, p. 224–232, 1938.

_____. *The Calculi of Lambda-Conversion*. Princeton: Princeton University Press, 1985. (Reimpressão da edição de 1941).

CUTLAND, Nigel. *Computability*. Cambridge: Cambridge University Press, 1992.

DAVIS, Martin. *The Undecidable*. New York: Raven Press, 1965.

_____. Why Gödel didn't have Church's thesis. *Information and Control*, v. 54, p. 3–24, 1982.

DEDEKIND, Richard. Was sind und was sollen die Zahlen? In: _____. *Gesammelte mathematische Werke*. New York: Chelsea Publishing Company, 1969. v. 3, p. 335–391.

DUMMETT, Michael. *Frege: Philosophy of Mathematics*. Cambridge: Harvard University Press, 1991.

EPSTEIN, Richard, CARNIELLI, Walter. *Computability, Computable Functions, Logic, and the Foundations of Mathematics*. Pacific Grove: Wadsworth, 1989.

FREGE, Gottlob. *Die Grundlagen der Arithmetik*. Hamburg: Felix Meiner Verlag, 1986.

_____. Logik in der Mathematik. In: _____. *Nachgelassene Schriften*. Editado por Hans Hermes e outros. Hamburg: Felix Meiner Verlag, 1967. p. 219–270.

GANDY, Robin. The Confluence of Ideas in 1936. In: HERKEN, Rolf (Ed.) *The Universal Turing Machine: A Half Century Survey*. New York: Oxford University Press, 1988. p. 55–111.

GÖDEL, Kurt. *Collected Works*. Editado por Solomon Feferman e outros. Oxford: Oxford University Press, 1986–1990. 2v.

HENKIN, Leon. On Mathematical Induction. *American Mathematical Monthly*, v. 66, p. 323–338, 1960.

HERBRAND, Jacques. *Écrits Logiques*. Paris: Presses Universitaires de France, 1968.

KALMÁR, László. Über ein Problem, betreffend die Definition des Begriffes der allgemein-rekursiven Funktion. *Zeitschrift für mathematischen Logik und Grundlagen der Mathematik*, v. 1, p. 93–96, 1955.

KALMÁR, László. An Argument Against the Plausibility of Church's Thesis. In: HEYTING, Arend (Ed.) *Constructivity in Mathematics: Proceedings of the Colloquium held at Amsterdam, 1957*. Amsterdam: North-Holland, 1959. p. 72–80.

KLEENE, Stephen. λ -definability and recursiveness. *Duke Mathematical Journal*, v. 2, p. 340–353, 1936.

_____. On notation for Ordinal Numbers. *The Journal of Symbolic Logic*, v. 3, p. 150–155, 1938.

_____. *Introduction to Metamathematics*. New York: D. Van Nostrand, 1952.

_____. Origins of recursive function theory. *Annals of the History of Computing*, v. 3, p. 52–67, 1981a.

_____. The theory of recursive functions approaching its centennial. *Bulletin of the American Mathematical Society*, v. 5, p. 43–61, 1981b.

_____. Reflections on Church's Thesis. *Notre Dame Journal of Formal Logic*, v. 28, p. 490–498, 1987.

KLEIN, Morris. *Mathematical Thought from Ancient to Modern Times*. New York: Oxford University Press, 1972.

LAKATOS, Imre. *Mathematics, Science and Epistemology*. Cambridge: Cambridge University Press, 1987. (Philosophical Papers, v. 2)

LEIBNIZ, Gottfried. *Die Philosophischen Schriften von Gottfried Wilhelm Leibniz*. Editado por C. J. Gerhardt. Hildesheim: Georg Olms Verlag, v. 5, 1978.

MANBER, Udi. *Introduction to Algorithms*. Reading: Addison-Wesley Publishing Company, 1989.

MANIN, Yu. *A Course in Mathematical Logic*. New York: Springer-Verlag, 1977.

MENDELSON, Elliott. On some Recent Criticism of Church's Thesis. *Notre Dame Journal of Formal Logic*, v. 4, p. 201-205, 1963.

_____. *Introduction to Mathematical Logic*. New York: Van Nostrand, 1979.

MONK, James Donald. *Mathematical Logic*. New York: Springer, 1976.

ODIFREDDI, Piergiorgio. *Classical Recursion Theory*. Amsterdam: North-Holland, 1989.

PÉTER, Rózsa. Konstruktion nichtrekursiver Funktionen. *Mathematische Annalen*, v. 111, p. 42-60, 1935.

_____. *Recursive Functions*. New York: Academic Press, 1967.

_____. *Recursive Functions in Computer Theory*. Chichester: Ellis Horwood Limited, 1981.

SEDGEWICK, Robert. *Algorithms*. Reading: Addison Wesley Publishing Company, 1989.

SHOENFIELD, Joseph R. *Mathematical Logic*. Reading: Addison Wesley Publishing Company, 1967.

SKOLEM, Thoralf. *Selected Works in Logic*. Editado por Jens Erik Fenstad. Oslo: Universitetsforlaget, 1970.

THOMAS, William J. Doubts About Some Standard Arguments for Church's Thesis. In: BOGDEN, Radu, NIINILUOTO, Ilkka (Eds.) *Logic, Language, and Probability*. Dordrecht: D. Reidel, 1973. p. 55-65.

TURING, Alan. Computability and λ -definability. *The Journal of Symbolic Logic*, v. 2, p. 153-163, 1937.

VAN HELJENOORT, Jean. *From Frege to Gödel*. Cambridge: Harvard University Press, 1967.

_____. L'œuvre logique de Jacques Herbrand et son contexte historique. In: STERN, J. (Ed.) *Proceedings of the Herbrand Symposium. Logic Colloquium '81*. Amsterdam: North-Holland, 1982. p. 57-85.

WANG, Hao. The Axiomatization of Arithmetic. *The Journal of Symbolic Logic*, v. 22, p. 145-158, 1957.

_____. *From Mathematics to Philosophy*. New York: Humanities Press, 1974.

WANG, Hao. *Reflections on Kurt Gödel*. Cambridge: The MIT Press, 1991.

WEBB, Judson C. *Mechanism, Mentalism and Metamathematics*. Dordrecht: D. Reidel, 1980.

_____. Gödel's Theorems and Church's Thesis: A Prologue to Mechanism. In: COHEN, Robert, WARTOFSKY, Marx (Eds.) *Language, Logic, and Method*. Dordrecht: D. Reidel, 1983. p. 309–353. (Boston Studies in the Philosophy of Science, v. 31).

ZIMBARG SOBRINHO, Jacob. Aspectos da Tese de Church-Turing. *Matemática Universitária*, n. 6, p. 1–23, 1987.

Índice

- Ackermann, iv, 1, 17–21, 54, 55
- Barwise, 61
- Bell, 66
- Bussey, 3
- Cajori, 2, 3
- Cantor, 1, 43, 54
- Carnap, 44
- Carnielli, 66, 67, 85
- Church, iii, iv, 16, 32–46, 49–55,
57–59, 63–72, 76, 83, 86, 87,
89, 90
- Cutland, 65, 66
- Davis, v, 34, 37–41, 43, 45, 50–54,
57, 58, 60, 61, 67, 81
- Dedekind, iv, 1, 2, 5–9, 11–13, 52
- Dummett, 5
- Enderton, 61
- Epstein, 66, 67, 85
- Euclides, 3
- Fermat, 63, 64
- Frege, 3–5, 44
- Gödel, iii–v, 1, 22–39, 42, 43, 49–
57, 59–61, 68, 69, 72–76, 83,
88, 89
- Gandy, v, 11, 18, 29
- Goldbach, 63, 64
- Grassmann, 3–5
- Henkin, 11
- Herbrand, iv, 1, 28–32, 35
- Hilbert, 16–19, 34, 54, 89
- Hoering, 86
- Kalmár, iii, iv, 30, 83–88
- Kleene, v, 11, 24, 25, 27, 28, 33–
37, 40, 42, 44–46, 50, 54–
61, 71, 75, 76, 78, 81, 83,
84, 86, 89
- Kline, 3
- Kronecker, 12–14

Lakatos, 87	van Heijenoort, iii, v, 7, 11, 18, 26–
Leibniz, 3	32
Machover, 66	von Neumann, 11
Manber, 2	Wang, 7, 52, 54
Manin, 67	Webb, v, 16, 35, 38, 51, 54, 88
Markov, 70, 71	Zimbarg Sobrinho, 75
Maurolycus, 3	
Mendelson, 24, 25, 38, 77, 85, 86	
Monk, 66, 72	
Odifreddi, iv, 53, 54	
Péter, 21, 85–87	
Parsons, 76	
Peano, 30, 73	
Post, iv, 39, 40, 42–46, 52–54, 72,	
89	
Rosser, 33	
Schröder, 15	
Sedgewick, 2	
Shoenfield, 72	
Skolem, iv, 1, 13–17, 54	
Thomas, iv, 63, 64, 72	
Turing, 40–42, 45, 52, 53, 55, 56,	
59–61, 64, 70, 71, 74, 75, 90	