UNIVERSIDADE DE SÃO PAULO
FACULDADE DE FILOSOFIA, LETRAS E CIÊNCIAS HUMANAS
DEPARTAMENTO DE FILOSOFIA
PROGRAMA DE PÓS-GRADUAÇÃO EM FILOSOFIA

A SINTAXE LÓGICA DA LINGUAGEM DE RUDOLF CARNAP: UMA ANÁLISE DO PRINCÍPIO DE TOLERÂNCIA E DA NOÇÃO DE ANALITICIDADE

Tiago Tranjan

São Paulo
2005

UNIVERSIDADE DE SÃO PAULO
FACULDADE DE FILOSOFIA, LETRAS E CIÊNCIAS HUMANAS
DEPARTAMENTO DE FILOSOFIA
PROGRAMA DE PÓS-GRADUAÇÃO EM FILOSOFIA

A SINTAXE LÓGICA DA LINGUAGEM DE RUDOLF CARNAP: UMA ANÁLISE DO PRINCÍPIO DE TOLERÂNCIA E DA NOÇÃO DE ANALITICIDADE

Tiago Tranjan

Dissertação apresentada ao Programa de Pós-Graduação em Filosofia, do Departamento de Filosofia da Faculdade de Filosofia, Letras e Ciências Humanas da Universidade de São Paulo, para a obtenção do título de Mestre em Filosofia.

Orientador: Prof. Dr. João Vergílio Gallerani Cuter

São Paulo
2005

DEDICATÓRIA

Para minha vó Ruth, que sempre nos ensinou
a todos, e nos ensina, cada dia mais.

Para minha vó Linda (*in memoriam*),
que tanta falta faz.

AGRADECIMENTOS

A meus pais, Nilce e Ercilio, pelo amor, pela educação e por tudo o que me deram, mas sobretudo pelo que são.

À Silvia, cuja inteligência e generosidade sempre me fascinam.

À minha irmã Marina, que veio ao mundo para me fazer feliz.

Ao João Vergílio, que me recebeu de braços e mente aberta.

Ao Rui, pela amizade e pela boa vontade inenarrável.

Ao Rogério, que sempre esteve lá (e sempre estará), no último momento.

Ao Thomas, pelo prazer que tem em compartilhar seus conhecimentos.

À Maria Helena, à Geni e à Mariê, que me ajudaram desde o início.

À Fapesp, que possibilitou este trabalho.

RESUMO

A Sintaxe Lógica da Linguagem, escrita por Rudolf Carnap no início da década de 1930, contém alguns aspectos notáveis. Dois deles, em particular, atraem a atenção dos estudiosos até hoje: a idéia inovadora defendida por Carnap – expressa pelo Princípio de Tolerância – de que não haveria uma moldura lógica privilegiada dentro da qual descrever o mundo; e a definição de analiticidade desenvolvida por Carnap para captar a noção de "verdade matemática", em que aparecem, de forma pioneira, métodos hoje associados à moderna teoria semântica. O presente trabalho busca examinar essas duas idéias contidas na obra de Carnap, com o objetivo de avaliar-lhes o significado filosófico e de verificar as razões por que, finalmente, não conseguiram realizar o que delas esperava o próprio autor.

ABSTRACT

The Logical Syntax of Language, written by Rudolf Carnap in the beginning of the thirties of the 20th century, comprises some remarkable aspects. Two of them, in particular, draw the attention of scholars until today: the innovating idea defended by Carnap – expressed in the Principle of Tolerance – that there were no privileged logical frame within which to describe the world; and the definition of analyticity developed by Carnap as to capture the notion of "mathematical truth", in which appear, for the first time, methods today associated to the modern semantic theory. This works intends to examine those two ideas contained in the work of Carnap, with the purpose of assessing their philosophical meaning and verifying the reasons why, in the end, they could not accomplish what the author himself expected from them.

PALAVRAS-CHAVE / KEY WORDS

CARNAP, PRINCÍPIO DE TOLERÂNCIA, ANALITICIDADE, POSITIVISMO LÓGICO, CÍRCULO DE VIENA.

ÍNDICE

Introdução

I

Carnap escreveu *A Sintaxe Lógica da Linguagem* (abreviada neste trabalho como *SLL*) entre os anos de 1932 e 1933. A primeira edição da obra, em alemão, surgiu em 1934; a edição em inglês, revista e ampliada, ficou pronta dois anos mais tarde, em 1936, e foi publicada somente em 1937[1]. Para qualquer pessoa familiarizada com a história da Lógica no século XX, a simples indicação dessas datas basta para mostrar a situação ao mesmo tempo privilegiada e extremamente complexa em que se encontrava Carnap ao tentar desenvolver – como faz em *SLL* – um sistema filosófico baseado na análise lógica da linguagem.

Justamente no início da década de 1930, Gödel havia dado ao mundo seus dois célebres teoremas – o da *completude* e o da *incompletude*. O primeiro veio a público em uma conferência realizada em Königsberg a 7 de outubro de 1930; o segundo apareceu originalmente sob a forma de artigo, no volume 38 do *Monatsheft für Mathematik und Physik*, em 1931[2]. Já em 1933, foi a vez do lógico polonês Alfred Tarski publicar a primeira versão, ainda em polonês, de sua clássica obra *Der Wahrheitsbegriff in den formalisierten Sprachen*[3], com a qual deitou as bases para as modernas teorias semânticas. Esses resultados fundamentais de Gödel e Tarski demonstram bem o estado de verdadeira ebulição por que ainda passava a Lógica nessa época, mais de cinqüenta anos após a aparição dos trabalhos pioneiros de Frege. A obra de Carnap, por seu lado, encontra-se em estreito diálogo com ambos, e não pode ser compreendida sem referência a eles.

Em termos históricos, portanto, podemos dizer que a década de 1930 foi o segundo grande momento – após o período que vai da publicação por Frege do seu *Begriffsschrift* (1879) até a publicação, por Russell e Whitehead, dos *Principia Mathematica* (1910-1913) – em que métodos essencialmente novos apareceram para a Lógica. E métodos, dessa vez,

[1] Carnap conta que as 22 seções adicionais que completam a edição em inglês já haviam ficado prontas em dezembro de 1933, para a edição original em alemão. Infelizmente, porém, tiveram de ser cortadas por falta de espaço. (p. xi, *Prefácio à edição inglesa*.)
[2] Para uma agradável exposição histórica, ver Hintikka (2000).
[3] Esse é o título da versão em alemão, publicada em 1935 com importantes alterações e correções em relação ao texto anterior.

extremamente complicados do ponto de vista matemático, para os quais extrair um significado filosófico preciso parecia (e, de certa forma, ainda parece) tarefa bastante elusiva.

Até a década de 1930, os novos caminhos da Lógica (e da Filosofia que nela se inspirava), apesar de árduos e trabalhosos – e de exigirem um extenso esforço de análise conceitual, às vezes problemático e capaz de conduzir a reveses momentâneos – pareciam razoavelmente seguros, e permitiam mesmo o otimismo exagerado exibido por Russell em sua primeira fase. Essa situação, porém, logo viria a se modificar. Alguns resultados surpreendentes[4] exigiram que se colocasse todo o trabalho até então realizado sob nova perspectiva – uma perspectiva, vale dizer, razoavelmente incômoda para pensadores que haviam se acostumado a considerar a lógica formal como a base firme a partir da qual erguer grandiosos edifícios teóricos[5]. Tornavam-se agora necessários esclarecimentos filosóficos totalmente renovados, de um tipo bem diferente do que até então se poderia imaginar.

As três figuras de Gödel, Tarski e Carnap, nesse sentido, sobressaem como os pensadores mais significativos desse período da história da Lógica. Os dois primeiros como aqueles que impulsionaram a Lógica em novas e até então insuspeitas direções. Carnap, por sua vez, como o pensador que mais lucidamente – e com o maior conhecimento de causa – tentou extrair-lhe as conclusões filosóficas.

II

Nosso trabalho buscará examinar alguns dos principais aspectos do projeto filosófico exposto em *SLL*. Mais especificamente, concentraremos nossa atenção na tentativa feita por Carnap de tratar a lógica completamente dentro do plano *sintático*. Esse ponto de vista sintático, como se sabe, foi superado – e rapidamente superado – pela visão semântica elaborada por Tarski, até hoje amplamente prevalente. Mais ainda, o próprio

[4] A peça fundamental dessa reviravolta foi, certamente, o teorema da incompletude de Gödel.

[5] Em *Infinite Regress and the Foundations of Mathematics*, republicado em [Lakatos, 1978], Lakatos oferece uma visão extremamente interessante da passagem de Russell de uma filosofia otimista e confiante para a situação de "sofrimento intelectual" que o próprio Russell descreve em *My Philosophical Development* (1959).

Carnap não tardaria em rejeitar, ele próprio, as opiniões que havia defendido em *SLL*, para abraçar uma concepção abertamente semântica da lógica e da fundamentação da matemática[6].

Isso significou para *SLL* um destino que, de certa maneira, não faz jus ao rico material teórico desenvolvido na obra. Por um lado, todos reconhecem a maestria com que Carnap lida com diversas das questões mais difíceis da lógica simbólica de seu tempo; e todos reconhecem no livro, não apenas uma etapa importante do pensamento do autor, como um dos melhores exemplares do pensamento do Círculo de Viena e do positivismo lógico em geral. Por outro lado, é inegável a tendência generalizada de enxergá-lo como portador de um projeto equivocado, rapidamente reconhecido como tal e, por isso mesmo, abandonado.

Essa avaliação está, em grande medida, correta. De fato, procuraremos mostrar em nosso trabalho que a concepção sintática elaborada por Carnap em *SLL* não apenas não se sustenta, como não se sustenta de acordo com os próprios parâmetros internos fixados no livro. Não obstante, isso não encerra a questão, da maneira como talvez se esperasse. O livro merece uma atenção cuidadosa e renovada: As idéias e métodos que alí aparecem exibem um requinte poucas vezes igualado em textos de lógica; e a obra como um todo compõe uma tentativa surpreendentemente abrangente, rica e perspicaz de tratar os problemas-chave colocados até hoje pela lógica e pela fundamentação matemática[7].

Trata-se, com efeito, da última e mais grandiosa tentativa de formular o projeto logicista sob uma perspectiva aceitável, após o golpe que essa escola de pensamento havia recebido com a publicação dos resultados de Gödel. Em *SLL*, o projeto logicista original surge alargado e aprofundado, para dar conta justamente da nova situação teórica que se oferecia. Carnap desenvolve seu método sintático, em direções novas e insuspeitas, porque foi um dos primeiros a compreender a natureza dos novos problemas que precisavam ser enfrentados.

---

[6] Já em *Introduction to Semantics*, livro publicado em 1942, Carnap adota o ponto de vista semântico. Esse ponto de vista recebe seu tratamento mais completo na obra de Carnap, porém, com a publicação de *Meaning and Necessity*, em 1947.

[7] De certa maneira, pode-se dizer a respeito de *SLL* aquilo que Wordsworth disse a respeito do poeta William Blake: "Não há dúvida de que esse pobre homem era louco, mas há algo na loucura dele que me interessa mais do que a sanidade de Lord Byron e Sir Walter Scott". Há, de fato, no projeto sintático de *SLL* certa "loucura" que interessa muito mais do que a sanidade de outros produtos do positivismo lógico.

III

Sobressaem, no contexto do trabalho de Carnap, duas idéias fundamentais: o Princípio de Tolerância em lógica e a noção de analiticidade. O Princípio de Tolerância revela-se o princípio diretor de todo o projeto filosófico de *SLL*. Ele possui grande alcance teórico, não somente para a lógica, mas para a própria maneira de conceber o estudo lógico e a natureza dessa disciplina. Já a noção de analiticidade, que se ramifica em duas no livro, encontra-se no limiar entre a lógica propriamente dita – tal como concebida em *SLL* e de acordo com o Princípio de Tolerância – e os estudos matemáticos, como campo específico de conhecimento.

O nosso objetivo, nas páginas a seguir, consiste precisamente em mostrar como Carnap enfrenta os principais problemas surgidos, em seu tempo, para a lógica e para a fundamentação lógica da matemática. Trata-se de um enfrentamento consciente, articulado segundo um plano ao mesmo tempo abrangente e minucioso, cujo propósito estava em atacar de maneira sistemática todas as principais questões em aberto nessas áreas. Mais do que isso, iremos argumentar que Carnap consegue obter respostas satisfatórias (no sentido de corretas) para cada uma das intrincadas questões que aborda – ao menos quando isoladamente consideradas. O problema final de *SLL* reside então em que, apesar da correção das respostas individuais encontradas pelo autor, algumas dessas peças não podiam ser encaixadas, da maneira como ele imaginou que podiam, na arquitetura lógica geral da obra.

A peça fundamental a que estamos nos referindo, certamente, é dada pelo conceito de "analiticidade" desenvolvido para a linguagem II, por meio do qual Carnap tenta – e consegue – encontrar um "critério de validade" para a matemática clássica. A construção desse conceito utiliza métodos inovadores para a época, paralelos aos métodos desenvolvidos por Tarski. Ao contrário de Tarski, porém, Carnap não enxergou claramente a natureza semântica de sua construção, e acreditou que pudesse manter-se no campo estritamente sintático. Esse equívoco de Carnap – o mais danoso para a avaliação crítica e para o destino teórico de *SLL* – possui diversas raízes, que tentaremos identificar.

O quadro final que emergirá de nossa análise de *SLL*, portanto, mostra um livro dentro do qual confluem todos os mais relevantes problemas da lógica e da fundamentação matemática contemporânea. Carnap soube percebê-los como poucos pensadores, e atacou-os com grande vigor teórico, desenvolvendo métodos e idéias substancialmente novos. Nós buscaremos estabelecer os pressupostos da abordagem sintática de Carnap e, por meio de um exame cuidadoso dos diversos momentos da obra, articulá-los segundo um projeto com a maior coerência interna possível. Essa busca nos permitirá alcançar algum ganho na compreensão teórica da obra.

Com efeito, o Princípio de Tolerância e o conceito de analiticidade desenvolvido por Carnap para a sua linguagem II relacionam-se de maneira complexa, aproximando-se da contradição. Por essa razão, a tendência generalizada, assumida com maior ou menor clareza por diferentes estudiosos, costuma ser a de tratá-los separadamente. Como conseqüência, ficam também separados – mais do que, na nossa opinião, deveriam – os aspectos da obra que lidam com a fundamentação da matemática e a visão filosófica e lógica geral proposta por ela. Nosso principal resultado reside em mostrar a articulação possível – e profunda – entre esses dois momentos. Essa articulação deve ser mantida sempre em vista, à luz do papel determinante que o Princípio de Tolerância assume em relação a todo o projeto da obra. O caminho que nós propomos para compreender essa relação, então, permitirá penetrar em alguns aspectos até hoje obscuros de *SLL*, como a tensão entre sintaxe e semântica, e entre lógica e matemática, que nela se revela.

Capítulo 1: Lógica, formalismo e sintaxe

I

Em *A Sintaxe Lógica da Linguagem* (*SLL*), Carnap propõe uma abordagem dita "sintática" para a filosofia. Isso significa que a questão central da obra pode ser resumida da seguinte maneira: Mostrar a possibilidade e a adequação de substituir todo o complexo emaranhado de problemas normalmente estudados em filosofia – muitos dos quais, segundo Carnap, sequer deveriam ser considerados como verdadeiros problemas – pela análise sintática de certas linguagens formais, devidamente caracterizadas. Logo na seção 2 do livro, ao explicar o percurso que pretende percorrer, Carnap expõe nos seguintes termos sua posição:

> *"We see, therefore, that whenever we investigate or judge a particular scientific theory from the logical standpoint, the results of this logical analysis must be formulated as* syntactical sentences *(...). The* logic of science *(...) is nothing else than the* syntax of the language of science. *This fact will be shown clearly in the concluding chapter of this book. The syntactical problems acquire a greater significance by virtue of the anti-metaphysical attitude represented by the Viena Circle. According to this view, the sentences of metaphysics are pseudo-sentences which on logical analysis are proved to be either empty phrases or phrases which violate the rules of syntax. Of the so-called philosophical problems, the only questions which have any meaning are those of the logic of science. To share this view is to* substitute logical syntax for philosophy*."* [8] (destaques do autor; grifos meus)

Carnap, portanto, deseja substituir a filosofia por algo a que chama "sintaxe lógica". Eis o seu projeto, condensado em algumas poucas palavras. Por trás desse enunciado, simples e direto como possa ser, vemos aparecer algumas das principais dificuldades que teremos de enfrentar para compreender a obra. A primeira delas, certamente, reside na

[8] *SLL*, págs. 7 e 8.

seguinte pergunta: O que é essa "sintaxe lógica", que deve substituir a filosofia? (De que tipo de estudo se trata?) Responder a essa questão será o objetivo, principalmente, do primeiro e do segundo capítulo do nosso trabalho.

Para entender o que é a sintaxe lógica elaborada por Carnap, devemos nos concentrar no exame de um conceito fundamental para o esquema de *SLL*: o conceito de "linguagem formal". O estudo sintático consiste precisamente no estudo dos mecanismos puramente formais que determinam e caracterizam certas linguagens – as chamadas linguagens formais. Substituir a filosofia pela sintaxe lógica é substituir a filosofia pelo estudo sistemático e abrangente dessas linguagens formais, assim como de sua relação com a ciência e a atividade científica.

Assim, dedicaremos o restante do presente capítulo à análise do conceito de "linguagem formal", tal como elaborado por Carnap. Antes, porém, de iniciar esse estudo, acreditamos ser útil chamar a atenção para um importante aspecto do projeto filosófico desenvolvido em *SLL*. Trata-se de um fato que, embora revelado mesmo pelo exame de uma passagem inicial como a citada acima, acaba muitas vezes por ser perdido de vista ao longo da leitura da obra – circunstância esta que pode causar alguma confusão ou até má vontade com relação às posições nela defendidas.

A idéia de substituir a filosofia, em toda sua amplitude, pela simples análise sintática, é inegavelmente estranha. Em certa medida, e com o auxílio de alguma perspectiva histórica, não podemos nos furtar à impressão de que o projeto de Carnap parece fadado ao fracasso desde o início, devido à própria ambição com que é concebido. Trata-se de uma redução grande demais: Substituir toda a filosofia, área ilimitada do pensamento humano, pelo mero estudo de estruturas formais. Esse projeto, contudo, irá nos parecer menos enigmático se tivermos em mente – também desde o início – que ele é baseado não em uma, mas em *duas* reduções distintas. E que essas duas reduções possuem significados bem diferentes.

De fato, se examinarmos com atenção o trecho reproduzido acima, veremos que o projeto de *SLL* apresenta-se por meio de um duplo movimento. Em primeiro lugar, Carnap acredita que todas as questões relativas à lógica da ciência possam ser reduzidas a questões sintáticas, vale dizer, questões relativas à sintaxe das linguagens utilizadas pela ciência.

Essa primeira redução, portanto, seria uma redução da lógica à sintaxe da linguagem. Sua formulação pode ser expressa assim: *Lógica é sintaxe*.

Em segundo lugar, porém, aparece uma segunda proposta de redução. Carnap acredita que a filosofia como um todo possa (e deva) ser reduzida ao estudo da lógica da ciência. Dito de outra maneira: Uma vez expurgada de tudo o que é vazio e carente de significado, a filosofia ficaria resumida à lógica da ciência. A formulação dessa segunda redução, portanto, pode ser expressa assim: *Filosofia é lógica* (da ciência). Como, pela primeira redução, lógica é sintaxe, chega-se à visão defendida no livro, de que a filosofia deva ser substituída pela sintaxe lógica.

É importante manter em vista, desde agora, esse duplo movimento no qual se baseia a abordagem de Carnap, pois essas duas alegações essencialmente distintas – por um lado, de que lógica é sintaxe, e por outro, de que a (verdadeira) filosofia é lógica da ciência – irão desempenhar papéis diversos no esquema da obra. Na passagem acima, representativa a esse respeito, sublinhamos três frases que indicam justamente esse roteiro. "*The* logic of science *(...) is nothing else than the* syntax of the language of science" traduz a primeira redução; "*Of the so-called philosophical problems, the only questions which have any meaning are those of the logic of science*", a segunda; a conclusão vem logo a seguir, na afirmação de que "*To share this view is to* substitute logical syntax for philosophy".

A distinção entre esses dois motivos é reafirmada na parte final de *SLL* (Parte V: "Philosophy and Syntax"), em que Carnap procura esclarecer a visão de filosofia que deveria emergir da consideração dos métodos expostos ao longo da obra. As duas seções iniciais dessa parte V são dedicadas justamente a defender, cada uma por sua vez, as duas reduções de que falamos. A primeira delas (seção 72) intitula-se "Philosophy Replaced by the Logic of Science", e trata – previsivelmente – da necessidade de excluir da filosofia toda uma série de problemas que, ultrapassando os limites da lógica científica, ultrapassam também os limites de qualquer discussão válida ou significativa. Carnap escreve:

> *"According to this view, then, once philosophy is purified of all unscientific elements, only the logic of science remains."* [9]

[9] *SLL*, pág. 279.

Já a seção seguinte (seção 73) intitula-se "The Logic of Science is the Syntax of the Language of Science". Ela se ocupa, como é igualmente claro, da outra proposta de redução defendida em *SLL*, e busca esclarecer (mais uma vez, após muitas ao longo do livro) que todas as questões da lógica podem e devem ser tratadas no plano exclusivamente formal da sintaxe:

> *"In what follows we shall examine the nature of the questions of the logic of science in the wide sense, (...), and we shall show that these questions are questions of syntax."* [10]

Em nosso trabalho, não nos preocuparemos em discutir o significado dessa segunda redução. Embora ela corresponda a uma crença bastante arraigada por parte de Carnap, e forneça provavelmente um impulso importante para suas idéias, o sistema de *SLL* não é, de nenhuma maneira, dependente dela. Carnap reconhece abertamente esse fato, na seguinte passagem:

> *"The view that, as soon as claims to scientific qualifications are made, all that remains of philosophy is the logic of science, cannot be established here and will not be assumed in what follows. (...) For anyone that share with us the antimetaphysical standpoint it will thereby be shown that all philosophical problems which have any meaning belong to syntax. The following investigations concerning the logic of science as syntax are not, however, dependent upon an adherence to this view; those who do not subscribe to it can formulate our results simply as a statement that the problems of that part of philosophy which is neither metaphysical nor concerned with values and norms are syntactical."* [11] (grifos meus)

Nós nos concentraremos, portanto, na seguinte redução: a visão de que o estudo lógico pode ser resumido ao estudo da sintaxe da linguagem. É aí que reside a visão peculiar da obra, bem como seus maiores méritos. Na verdade, é para estabelecer esse fato que Carnap dedica todos os seus esforços, e que desenvolve todo o arsenal técnico presente

[10] *SLL*, pág. 281.
[11] *SLL*, pág. 280.

no livro. É para mostrar que a lógica pode ser vista como uma ciência puramente formal que Carnap desenvolve em novas direções o conceito de "linguagem formal", do qual deveremos nos ocupar a partir de agora. Passemos, portanto, ao seu exame pormenorizado.

II

Carnap designa as linguagens formais, freqüentemente, pelo termo "cálculo" (ou ainda "cálculo formal"[12]). Na já mencionada seção 2 de *SLL*, cujo título é justamente "Languages as Calculi", ele fornece a seguinte definição:

> *"By a* calculus *is understood a system of conventions or rules of the following kind. These rules are concerned with elements – the so-called symbols – about the nature and relations of which no more is assumed than that they are distributed in various classes. Any finite series of symbols is an* expression *of the calculus in question.*
>
> *The rules determine, in the first place, the conditions under which an expression can be said to belong to a certain category of expressions; and, in the second place, under what conditions the transformation of one or more expressions in to another or others may be allowed. (...)"* [13] (destaques do autor)

A definição acima incorpora o método então corrente de expor sistemas de lógica formal, consagrado desde a publicação por Russell e Whitehead dos *Principia Mathematica*. Os dois tipos de regra a que Carnap faz referência nada mais são do que as regras de formação e de transformação de uma linguagem. Até aí, nenhuma novidade. O aspecto que nos interessa, porém, deve ser buscado no caráter formal que Carnap atribui aos cálculos assim definidos. Em que ele consiste? Posto de outra maneira: Carnap fornece uma definição para certas estruturas que denomina de "cálculos formais"; por outro lado, já sabemos que a essência mesma de seu sistema reside na tentativa de mostrar que a lógica pode ser completamente tratada no plano sintático, ou seja, de maneira puramente formal.

[12] Seguindo Carnap, usaremos indiferentemente os termos "linguagem formal", "cálculo formal" e "cálculo".
[13] *SLL*, pág. 4.

Precisamos esclarecer, portanto, em que consiste exatamente a formalidade das estruturas assim definidas.

Um cálculo, segundo a definição acima, é um conjunto de regras referidas a símbolos. Contudo, o cálculo ocupa-se dos símbolos – seus elementos básicos – apenas no que diz respeito à sua possibilidade de formar seqüências finitas (as chamadas "expressões"). Dentro de um cálculo formal, o único objeto de estudo são as possíveis combinações lineares em que seus símbolos podem se organizar. Em relação a esses símbolos, porém, Carnap observa:

> *"The term 'symbol' in what follows will have the same meaning as the word 'character'. It will not be assumed that such a symbol possesses a meaning, or that it designates anything."* [14] (grifo meu)

A construção de uma linguagem formal inicia-se pela escolha de um conjunto de símbolos que lhe servirão de base. Mas que tipo de coisas são esses símbolos? A resposta de Carnap é clara: Quaisquer coisas podem ser utilizadas como símbolos, de marcas de tinta sobre um papel a objetos do mundo físico[15]. A escolha é absolutamente arbitrária. Isso ocorre porque os símbolos, em si, são absolutamente mudos. Como diz Carnap, nada mais deve ser assumido em relação a eles além do fato de estarem distribuídos em várias classes.

Essa arbitrariedade na escolha dos símbolos que compõem uma linguagem é extremamente relevante, na medida em que indica os pressupostos simples que permitem a Carnap caracterizar como "formal" sua abordagem. A construção de uma linguagem, nesse sentido, depende exclusivamente de uma série de operações não-problemáticas estabelecidas com referência à noção de "símbolo", e cuja possibilidade Carnap toma, em princípio, como plenamente garantidas.

Em primeiro lugar, há a capacidade de reconhecer símbolos. Para Carnap, o reconhecimento de símbolos arbitrários – justamente porque são arbitrários – é tarefa que,

---

[14] *SLL*, pág. 5.

[15] Na página 6, por exemplo, Carnap escreve:

> *"Further, it is equally unimportant from the syntactical point of view, that, for instance, the symbol 'and' should be specifically a thing consisting of printers' ink. If we agreed always to place a match upon the paper instead of that particular symbol, the formal structure of the language would remain unchanged."* – *SLL*, pág. 6.

por definição, não oferece problemas. Trata-se de uma atividade não-empírica que figura na base da própria racionalidade do seu sistema. O fato de duas ocorrências distintas de um mesmo símbolo serem ao mesmo tempo diferentes (não são a mesma ocorrência) e iguais (são o mesmo símbolo) não gera nenhum problema no seu esquema sintático: Pelo contrário, a capacidade de realizar essa operação extremamente básica, ou seja, de instituir símbolos arbitrários e reconhecê-los (em suas diversas ocorrências) como tais, faz parte do mínimo de racionalidade necessário ao esquema formal de *SLL*. Carnap refere-se a esse estado de coisas quando escreve:

> *"The two symbols 'a' and 'a' occur at different places on this page. They are therefore* different *symbols (not the* same *symbol); but they are* equal *(not* unequal*). The syntactical rules of a language must not only determine what things are to be used as symbols, but also under what circumstances these symbols are to be regarded as syntactically equal."* [16] (destaques do autor)

Não apenas a operação de reconhecer símbolos arbitrários, mas também outras operações semelhantes gozam desse mesmo *status* não-problemático no sistema de *SLL*. Assim, temos a possibilidade de agrupar os símbolos em classes (como vimos, a única coisa que se pode assumir em relação aos símbolos é a sua distribuição em classes, tais como "variáveis", "símbolos lógicos" etc.). Essas classes são igualmente arbitrárias, o que faz da separação de símbolos arbitrários em classes arbitrárias uma operação tão não-problemática como o reconhecimento de símbolos.

Por outro lado, há a possibilidade de colocar os símbolos em seqüências lineares finitas. Aqui, obviamente, o verbo "colocar" não deve ser entendido no sentido literal, referido a uma operação física. Não é necessário nem mesmo que os símbolos sejam efetivamente escritos um atrás do outro[17]. O que é importante é a capacidade de considerar, como tais, seqüências formadas pelos símbolos de uma linguagem. Assim, não é necessário que a seqüência "ab" esteja efetivamente escrita, aqui nesta página ou em qualquer outro

[16] *SLL*, pág. 15.
[17] Essa caracterização aplica-se, mais precisamente, àquilo que Carnap denomina "sintaxe pura", por oposição a "sintaxe descritiva".

lugar, para que se possa falar a respeito da seqüência de dois símbolos composta pelo símbolo "a" seguido do símbolo "b". Símbolos podem ser postos em seqüência, e essa possibilidade essencialmente abstrata fornece a própria essência de uma linguagem formal.

Finalmente, o sistema sintático de Carnap tem como pressuposto a possibilidade de comparar duas ou mais seqüências lineares e finitas de símbolos. Se é possível reconhecer e falar a respeito de seqüências de símbolos, é igualmente possível falar de certas relações entre essas seqüências – desde que essas relações refiram-se exclusivamente à ordenação dos símbolos em seqüência. Trata-se do tipo de operação envolvida na constatação de que a seqüência "ab" aparece como "parte" da seqüência "abc" – ou, mais precisamente, de que a seqüência "abc" é formada a partir da seqüência "ab" pela justaposição final do símbolo "c".

Nada disso, de fato, parece muito problemático. É esse conjunto de operações simples que permitem a Carnap caracterizar como formais as linguagens que se constroem a partir delas. Uma linguagem será formal se sua construção fizer referência exclusivamente à possibilidade de reconhecer símbolos, agrupá-los em classes e construir e comparar seqüências lineares finitas desses símbolos. Os dois tipos de regras que caracterizam uma linguagem – as regras de formação e de transformação – apresentam-se como regras formais porque dependem somente dessas operações "formais". Ainda na seção 2, Carnap exemplifica-as da seguinte maneira:

> *"The two kinds of rules are those which we have previously called the rules of formation and the rules of transformation – namely, the syntactical rules in the narrower sense (e.g. 'An expression of this language is called a sentence when it consists, in such and such a way, of <u>symbols of such and such a kind</u>, occurring <u>in such and such an order</u>'), and the so-called logical laws of deduction (e.g. 'If a sentence is composed of <u>symbols combined in such and such a way</u>, and if another is composed of <u>symbols combined in such and such another way</u>, then the second can be deduced from the first')."* [18] (grifos meus)

[18] *SLL*, pág. 4.

A sintaxe formal, portanto, é o estudo de certo conjunto de símbolos exclusivamente em sua capacidade de se combinar em seqüências lineares. Na parte IV de *SLL*, em que Carnap oferece um tratamento técnico abrangente para aquilo que denomina de “sintaxe geral”, esse fato é exposto de maneira bastante clara:

> *“By a language we mean here any sort of calculus, that is to say, a system of formation and transformation rules concerning what are called* expressions*, i.e. finite, ordered series of elements of any kind, namely, what are called* symbols *(...). In pure syntax, only syntactical properties of expressions, in other words, those that are dependent <u>only upon the kind and order of the symbols</u> of the expressions, are dealt with.”* [19] (destaques do autor; grifo meu)

Eis por que Carnap, em outra passagem, chama a sintaxe formal de análise combinatorial:

> *“Pure syntax is thus wholly analytic, and is nothing more than* combinatorial analysis*, or, in other words, the* geometry *of finite, discrete, serial structures of a particular kind.”* [20] (destaques do autor)

Essas duas caracterizações da sintaxe pura – como análise combinatorial e como tipo específico de geometria – são bastante significativas. A análise combinatorial é uma disciplina matemática que se ocupa, precisamente, das possibilidades de combinar elementos, e que se distingue pela abstração que faz de quaisquer outras características desses elementos que não a sua participação em conjuntos e séries. Por outro lado, também a geometria é uma ciência que trata das relações ordenadas que podem aparecer entre certos elementos (pontos de um “espaço”) e suas possíveis combinações (como “retas”, “planos” e “curvas”).

Nos dois casos, porém, a comparação sugerida por Carnap é importante não somente por reafirmar o caráter formal da sintaxe como estudo de certas relações lineares combinatoriais, mas também por caracterizá-lo como um estudo “analítico”. A definição de

[19] *SLL*, pág. 168.
[20] *SLL*, pág. 7.

"analiticidade", e suas relações com a idéia de "verdade", constitui um dos aspectos mais intrincados da obra de Carnap, no qual somente aos poucos poderemos ir penetrando. Igualmente, o *status* sintático e possivelmente analítico da matemática (assim como da geometria, na sua qualidade de disciplina matemática) e, mais especificamente, a posição da aritmética para o sistema de *SLL*, representam problemas consideráveis para a interpretação das idéias contidas no livro, de que trataremos em capítulo adequado.

Por enquanto, podemos observar que o método de aritmetização de Gödel, ferramenta utilizada de maneira essencial em diversos momentos de *SLL* e a respeito do qual tornaremos a falar, baseia-se no princípio de colocar os símbolos de uma linguagem em correspondência com certos números naturais. Para Carnap, o que esse método fornece é justamente uma análise combinatorial adequada, que lhe permitirá extrair importantes conclusões sintáticas. Os números naturais possuem certa estrutura combinatorial amplamente conhecida que, por meio da associação entre números e símbolos, pode ser transposta para o estudo sintático. E o estudo sintático, como procuramos estabelecer, resume-se ao estudo combinatorial das possíveis seqüências lineares em que se pode arranjar um conjunto arbitrário de símbolos.

III

É importante fixar, nesse ponto, precisamente a quê se opõe a visão sintática da lógica defendida por Carnap. De fato, Carnap insiste em que a lógica deva ser tratada exclusivamente no plano formal, entendido da maneira como explicamos. Ao adotar essa postura, ele acredita estar se contrapondo a outra maneira de encarar o estudo lógico, corrente em sua época. Qual seria ela?

Ainda na seção 1 de *SLL* – que, ao lado da seção 2, compõe a introdução do livro – Carnap oferece uma descrição dessa situação, que reproduzimos na íntegra para melhor poder comentá-la:

> *"The prevalent opinion is that syntax and logic, in spite of some points of contact between them, are fundamentally theories of a very different type. The syntax of a language is supposed to lay down rules according to*

which the linguistic structures (e.g. the sentences) are to be built up from the elements (such as words or parts of words). The chief task of logic, on the other hand, is supposed to be that of formulating rules according to which judgments may be inferred from other judgments; in other words, according to which conclusions may be drawn from premises.

But the development of logic during the past ten years has shown clearly that it can only be studied with any degree of accuracy when it is based, not on judgments (thoughts, or the content of thoughts), but rather on linguistic expressions, of which sentences are the most important, because only for them is it possible to lay down sharply defined rules. And actually, in practice, every logician since Aristotle, in laying down rules, has dealt mainly with sentences. But even those modern logicians who agree with us in our opinion that logic is concerned with sentences, are yet for the most part convinced that logic is equally concerned with the relations of meaning between sentences. They consider that, in contrast with the rules of syntax, the rules of logic are non-formal. (...)" [21]

Há, no trecho citado acima, diversos aspectos dignos de nota. O mais importante deles é a veemência com que Carnap se opõe à idéia de que a lógica deva basear-se em relações de "sentido" ou de "significado". "Sentido" e "significado", para ele, são entidades (ou supostas entidades) que não possuem um papel a desempenhar no estudo lógico. Precisamos esclarecer, portanto, como uma tal postura se insere no esquema formal de *SLL*.

O objetivo central de Carnap, como vimos, é mostrar que a lógica esgota-se – pode ser completamente tratada – no plano sintático. O plano sintático, por sua vez, é aquele plano formal que, na análise de expressões e sentenças, faz referência unicamente à ordem linear dos símbolos de que tais expressões e sentenças são compostas. Em suma: no plano sintático-formal da análise lingüística, nenhum elemento deve aparecer que vá além da consideração imediata das expressões como seqüências de símbolos. Em particular, não há espaço para dois tipos de investigação normalmente tidos como inerentes ou necessários ao estudo da lógica.

[21] *SLL*, pág. 1.

Por um lado, Carnap rejeita, como parte da lógica, investigações baseadas no "significado" do símbolo. Conforme já observamos, o símbolo, do ponto de vista adotado em *SLL*, é completamente mudo, ou seja, é um elemento arbitrário que, para desempenhar sua função, exige unicamente a possibilidade de ser instituído e reconhecido como tal. Considerado em si mesmo, o símbolo não aponta para nada: "*It will not be assumed that such a symbol possesses a meaning, or that it designates anything*"[22]. Por outro lado, o mesmo princípio aplica-se ao estudo das sentenças (e das expressões em geral). Elas devem ser encaradas exclusivamente como seqüências de símbolos. Assim como os símbolos, também elas são mudas: não possuem nenhum "sentido" que necessite ser previamente analisado para determinar-lhes as características lógicas ou as relações lógicas em que podem entrar com outras sentenças. Igualmente, não apontam para nenhuma esfera de "significados" aos quais seja necessário recorrer para possibilitar o exame dessas características e relações lógicas.

Na passagem acima, Carnap inicia sua crítica à visão tradicional da lógica pela constatação de que, para atingir resultados mais seguros, esta disciplina deve dedicar-se à análise de sentenças, ou seja, expressões lingüísticas para as quais é possível propor regras bem definidas. Essas expressões lingüísticas – seqüências de símbolos – opõem-se aqui a "juízos", como entidades mentais de *status* incerto. Na verdade, não há nenhum mal intrínseco em supor que haja algum tipo de isomorfismo entre sentenças e juízos, vale dizer, em supor que a toda sentença corresponda um pensamento e que a todo pensamento (ou ao menos a todo pensamento assertivo, com um conteúdo lógico bem determinado) corresponda uma sentença. Carnap, porém, deseja livrar-se justamente desse tipo de discussão: expressões lingüísticas, como seqüências de símbolos arbitrários, comportam um exame direto e inequívoco, sem as ambigüidades que inevitavelmente prejudicam o estudo da lógica quando esta busca se referir a entidades subjetivas tais como "juízos".

Posto em outras palavras: Apenas expressões lingüísticas admitem uma análise objetiva e clara (sintática). Eliminar a referência a "pensamentos" ou "juízos", portanto, em prol da consideração direta de expressões lingüísticas plenamente determinadas, é o primeiro passo que Carnap julga necessário dar para livrar o estudo lógico de sérias

[22] *SLL*, pág. 5; ver nota 6 acima.

ambigüidades que impediam seu desenvolvimento segundo linhas claras e científicas. Contudo, Carnap avalia que isso só não é suficiente. Ele vê claramente que, mesmo entre os lógicos que aceitam essa visão "lingüística" de sua disciplina, uma outra confusão costuma estabelecer-se. Na verdade, ele constata que essencialmente a mesma confusão, relativa a "pensamentos" e "juízos", parece reintroduzir-se na análise lógica de sentenças, ainda que de maneira disfarçada.

O primeiro passo para perceber esse fato está na observação feita por Carnap de que, de certo modo, o estudo lógico de sentenças, e não de pensamentos, sempre foi a regra, desde Aristóteles ("*And actually, in practice, every logician since Aristotle, in laying down rules, has dealt mainly with sentences*"). O que ele quer dizer com isso? Carnap parece estar simplesmente constatando o fato óbvio de que qualquer resultado lógico que se deseje comunicar ou fixar – regras lógicas, silogismos, etc. – precisa ser formulado dentro de uma linguagem. Ao ser formulado dentro de uma linguagem, porém, o resultado lógico adquire uma expressão lingüística: juízos, por exemplo, são traduzidos por sentenças, e as relações entre juízos aparecem como relações entre sentenças. Por isso Carnap fala que, "na prática", as regras lógicas têm lidado essencialmente com sentenças. A expressão de uma regra lógica, ainda que esta regra lide supostamente com pensamentos ou juízos (ou com seu conteúdo), necessita estar referida à formulação lingüística desses pensamentos e juízos, e pode ser *vista* como uma regra acerca dessas formulações lingüísticas.

Portanto, é sempre possível considerar que, ao estabelecer regras lógicas acerca de entidades subjetivas e aparentemente intangíveis como juízos e pensamentos, qualquer lógico (desde Aristóteles) esteja na verdade propondo regras acerca das sentenças que são sua expressão lingüística. Contrariamente, porém – e é aqui que reside o foco das preocupações de Carnap –, é de se temer que, mesmo ao propor regras objetivas a respeito de entidades lingüísticas tangíveis, o lógico esteja tentando captar regras acerca de supostos conteúdos impalpáveis que lhes estariam subjacentes. O perigo, então, reside para Carnap nessa tendência inversa e igualmente difundida, da qual buscará se livrar. Ele percebe que, mesmo entre os lógicos que aceitam como seu campo de trabalho o estudo exclusivo de sentenças (e não de pensamentos ou juízos), o elemento não-formal torna a aparecer: Pois, apesar de lidar com expressões lingüísticas objetivamente analisáveis, eles parecem julgar que as relações lógicas entre tais expressões dependem de um conteúdo qualquer que transcende o plano objetivo de sua organização como seqüências de símbolos; que dependem, em outras palavras, de supostos "significados" e "sentidos", que nada mais seriam do que a reintrodução dos conteúdos subjetivos associados a "juízos" e "pensamentos".

Na oposição a essa visão tradicional, vemos reafirmar-se em toda sua força a abordagem formal de *SLL*. Sentenças são plenamente caracterizáveis como seqüências de símbolos. Até aí, não parece haver grande controvérsia. Segundo o próprio Carnap observa, a maioria dos estudiosos parece disposta a reconhecer esse fato, e aceitar que a sintaxe (ciência formal) trata "da forma como estruturas lingüísticas (por exemplo sentenças) são construídas a partir de seus elementos (como palavras e partes de palavras)"[23]. No entanto, Carnap deseja dar um passo além – e, vale dizer, um passo bastante grande. Para ele, não apenas as sentenças ficam plenamente caracterizadas como seqüências de símbolos, como também as relações lógicas entre sentenças deveriam ser estabelecidas sem a necessidade de recorrer a nenhuma outra ordem de considerações que não essa estrutura simbólica.

Se uma sentença pode ser completamente (univocamente) descrita pela referência aos símbolos de que é composta e à ordem em que esses símbolos aparecem, então as relações lógicas entre sentenças, e as características lógicas de sentenças, também poderiam (e, na opinião de Carnap, deveriam) ser exclusivamente estudadas por referência a essa estrutura linear de símbolos. É por isso que, na continuação da passagem citada acima, Carnap escreve:

> *"(...) In the following pages, in opposition to this standpoint, the view that logic, too, is concerned with the* formal *treatment of sentences will be presented and developed. We shall see that the logical characteristics of sentences (...) and the logical relations between them (...) are solely dependent upon the syntactical structure of the sentences. In this way, logic will become a part of syntax, provided that the latter is conceived in*

[23] Ver citação acima (nota 13).

> *a sufficiently wide sense and formulated with exactitude."* [24] (destaque do autor)

E não seria demais, agora que já lhe esclarecemos as razões, insistir na caracterização negativa que Carnap faz das teorias formais, com a rejeição de conceitos como "sentido" e "significado". A seguinte passagem é especialmente digna de atenção, na medida em que tais aspectos negativos estão diretamente antepostos à caracterização positiva do âmbito sintático e formal, de que tratamos na seção anterior:

> *"A theory, a rule, a definition, or the like is to be called* formal *when no reference is made in it either to the meaning of the symbols (for example, the words) or to the sense of the expressions (e.g. the sentences), but simply and solely to the kinds and order of the symbols from which the expressions are constructed."* [25] (destaque do autor)

IV

Mesmo após oferecer uma caracterização positiva (seção II acima) e outra negativa (seção III acima) da abordagem sintática, devemos inspecionar ainda mais de perto a idéia, característica de *SLL*, de que a lógica é puramente formal, ou seja, de que a lógica da linguagem *é* a sintaxe da linguagem. Já escrevemos que, segundo a argumentação de Carnap, se uma expressão lingüística fica plenamente caracterizada pela seqüência de símbolos de que se compõe, então todas as suas características e relações lógicas devem ser determinadas igualmente (e exclusivamente) por referência a essa seqüência de símbolos. Em outras palavras: todas as características e relações *lógicas* de expressões lingüísticas ficam determinadas no plano formal. Mas será que poderiam existir, com relação a uma linguagem, outras características e relações passíveis de estudo? Certamente que sim. A esse respeito, Carnap observa:

> *"When we maintain that logical syntax treats language as a calculus, we do not mean by that statement that language is nothing more than a*

[24] *SLL*, págs. 1 e 2.
[25] *SLL*, pág. 1.

> *calculus. (...) In addition, any particular language has, apart from that aspect* [aspecto formal]*, others which may be investigated by other methods. (...)"* [26] (colchete meu)

Por exemplo, uma linguagem pode ser estudada com relação aos sentimentos que suas sentenças despertam no ouvinte, em relação ao significado de seus símbolos, em relação à sua utilização como instrumento de comunicação, e assim por diante. O que significa, então, dizer que todos os aspectos *lógicos* da linguagem ficam determinados exclusivamente no plano formal e sintático?

Na verdade, essa posição defendida por Carnap equivale a uma delimitação do campo lógico, cujo propósito e significado exato teremos de examinar mais a fundo ao longo deste trabalho. Por enquanto, devemos compreender que a abordagem sintática proposta por Carnap *para a lógica* equivale à proposição da seguinte identidade: Tudo o que eu desejo chamar de "lógico" pode ser esgotado pelo exame sintático (formal) das estruturas lingüísticas; tudo o que pode ser estabelecido pelo exame sintático (formal) das estruturas lingüísticas eu estou disposto a chamar de "lógico".

É isso o que está por trás da idéia de considerar como mudos – do ponto de vista da *lógica* – os símbolos de uma linguagem. Símbolos, na verdade, nem sempre são mudos. Pelo contrário: na hora de utilizar uma linguagem como linguagem científica, por exemplo, será necessário associar alguns de seus símbolos (símbolos descritivos) a certos elementos (objetos, eventos, processos etc.) do mundo empírico. Nesse tipo de situação prática, nasce toda uma complexa rede de relações extra-sintáticas: da linguagem com aqueles que a utilizam, bem como da linguagem com o domínio de objetos para a descrição do qual ela foi construída. O método científico terá de lidar com todos esses aspectos. Nada disso tem relevância, porém, para o estudo formal (vale dizer, lógico). Esse estudo busca revelar a estrutura interna de uma linguagem, e analisar da melhor maneira possível as relações entre suas sentenças, termos e expressões, com o objetivo justamente de fornecer uma ferramenta útil e precisa para a ciência.

É nesse sentido que o estudo sintático opta – e é necessário esclarecer que se trata de uma opção – por desconsiderar o "significado" dos símbolos, o que quer que esses

[26] *SLL*, pág. 5.

“significados” possam ser (idéias, sensações ou outros), para se ater ao próprio símbolo como tal. O que importa, do ponto de vista formal, são as possibilidades de combinação desses símbolos, e as regras que possam ser formuladas com referência a essas possibilidades e a essas combinações.

Um modo interessante de ver essa situação é fornecido pelo próprio Carnap, ao comentar a respeito do possível isomorfismo entre linguagens. Mesmo dispondo de símbolos distintos – e já que os símbolos são mudos –, duas linguagens podem ter exatamente a mesma estrutura sintática (e portanto a mesma estrutura lógica). Isso ocorrerá, é claro, desde que esses símbolos distintos se articulem de acordo com as mesmas regras sintáticas (regras equivalentes). Eis como Carnap descreve a situação:

> *“We have already said that syntax is concerned solely with the formal properties of expressions. We shall now make this assertion more explicit. Assume that two languages (*Sprachen*), S1 and S2, use different symbols, but in such a way that a one-to-one correspondence may be established between the symbols of S1 and those of S2 so that any syntactical rule about S1 becomes a syntactical rule about S2 if, instead of relating it to the symbols of S1, we relate it to the correlative symbols of S2; and conversely. Then, although the two languages are not alike, they have the same* formal structure *(we call them isomorphic languages), and syntax is concerned solely with the structure of languages in this sense.”* [27] (destaques do autor)

Vemos assim, sob nova perspectiva, o que significa para Carnap dizer que a lógica é uma ciência formal. Trata-se de uma escolha, cujo propósito e motivação ainda teremos de examinar. As linguagens possuem diversos aspectos, sob os quais poderíamos considerá-las e estudá-las. O estudo sintático, no qual Carnap propõe resumir a lógica, trata de possíveis estruturas de símbolos, nas quais o importante é a estrutura, e não os símbolos. Como já observamos, uma linguagem formal é um conjunto de regras referidas a símbolos; o essencial nessa definição, porém, são as regras, e a maneira como elas permitem a articulação de símbolos mudos. A natureza específica do símbolo não é relevante, assim

[27] *SLL*, pág. 6.

como não são relevantes supostos "significados" de símbolos ou expressões. Somente a estrutura combinatorial desse conjunto de símbolos em expressões lineares, descrita pelas regras sintáticas, é importante.

V

Devemos analisar agora, ainda que parcialmente, um último aspecto extremamente importante da proposta sintática de Carnap. Ele diz respeito à principal novidade trazida por *SLL* na maneira de conceber o estudo lógico, e poderíamos mesmo dizer que se trata do aspecto fundamental da obra: Pois será somente a consideração desse aspecto, em todos os seus desdobramentos, que permitirá ver o projeto de *SLL* como portador de uma idéia realmente original: O Princípio de Tolerância em lógica.

Para entender o ponto essencial da questão que iremos examinar, necessitamos perguntar: A abordagem formal de Carnap, tal como a procuramos descrever, é realmente uma novidade no estudo lógico? Carnap insiste em que seu tratamento puramente formal da lógica é algo novo; que a maioria dos lógicos, até então, não aceitava a idéia de que a lógica fosse uma ciência sintática. Para esses lógicos tradicionais, segundo Carnap, a sintaxe poderia fornecer no máximo uma descrição do processo de formação de sentenças a partir de símbolos, ao passo que o estudo lógico, ao se ocupar da relação de conseqüência e dedução entre sentenças, escaparia necessariamente ao plano formal[28]. Mas será realmente verdade que a lógica, até então, não era tratada de maneira formal?

Ainda na introdução da obra, antes de desenvolver o complexo arsenal de conceitos e resultados que lhe permitirão uma análise mais aprofundada de todos os problemas envolvidos em seu projeto, Carnap tenta justificar ou motivar, por meio de exemplo, o tratamento formal que pretende adotar também para as regras lógicas de dedução ou inferência[29] (regras de transformação), e não só para as regras de formação. Eis o que ele diz:

[28] Ver as citações associadas às notas 13 e 16.
[29] Carnap usa, freqüentemente, o termo "inferência" no lugar do termo "dedução". O importante a observar é que os dois termos são absolutamente intercambiáveis.

> *"For the moment we will leave aside the question of the formal deficiencies of the word-language, and, by the consideration of examples, proceed to convince ourselves that rules of formation and transformation are of like nature, and that both permit of being formally apprehended. For instance, given an appropriate rule, it can be proved that the word-series 'Pirots karulize elatically' is a sentence, provided 'pirots' is known to be a substantive (in the plural), 'karulize' a verb (in the third person plural), and 'elatically' an adverb; all of which, in a well-constructed language – as, for example, in Esperanto – could be gathered from the form of the words alone. The meaning of the words is quite inessential for the purpose, and need not be known. Further, given an appropriate rule, the sentence 'A karulizes elatically' can be deduced from the original sentence and the sentence 'A is a pirot' – again provided that the type to which the individual words belong is known. Here also, neither the meaning of the words nor the sense of the three sentences need be known."* [30]

Nessa passagem, Carnap parece estar exagerando um pouco a novidade de sua abordagem, dita formal e sintática. De fato, todo lógico desde Aristóteles (para parodiar o próprio Carnap[31]), ao propor regras lógicas de dedução, havia tentado formulá-las com referência justamente ao aspecto formal das proposições envolvidas (sejam estas proposições consideradas como sentenças, como juízos ou qualquer outra coisa; aqui, estamos usando o termo "proposição" de maneira essencialmente neutra). Uma regra pode instituir-se como regra lógica precisamente porque faz abstração das entidades concretas presentes em qualquer de seus casos particulares, para expor uma estrutura geral e abstrata de dedução que depende apenas da forma das proposições envolvidas. É dessa maneira que surge o clássico exemplo: "Todo homem é mortal"; "Sócrates é homem"; portanto, "Sócrates é mortal". Esse silogismo famoso é apenas a concretização, para certo caso particular, de uma estrutura formal – "Todo A é B"; "C é A"; portanto, "C é B" – que não depende do significado dos termos envolvidos.

[30] *SLL*, pág. 3.
[31] Ver citação associada à nota 13.

Vemos assim que, também nesse exemplo mais do que tradicional, a essência do procedimento lógico-dedutivo é buscada nos aspectos formais das sentenças. Além disso, nenhuma referência é feita a algo que fosse o "significado" dos termos, sendo a abstração desses significados a própria condição que permite estabelecer o resultado oferecido como um resultado propriamente lógico. Portanto, devemos concluir que também essa abordagem tradicional pode ser dita – e com grande justiça – formal. Mas, então, qual seria a verdadeira novidade aportada pelas idéias de Carnap? Por que esse autor insiste em caracterizar a lógica, *contra* as opiniões correntes, como uma ciência *puramente* formal, a ser exaurida exclusivamente no plano sintático?

Nesse mesmo sentido, se considerarmos as tendências mais modenas em lógica no tempo de Carnap, vemos a seguinte situação: A lógica simbólica, nascida com os trabalhos de Frege na segunda metade do século XIX, e consagrada com a publicação dos *Principia Mathematica* por Russell e Whitehead cerca de 20 anos antes do aparecimento de *SLL*, propunha igualmente o estudo lógico como o estudo de linguagens simbólicas formais, baseadas em conjuntos de regras bem definidas, referidas exclusivamente às sentenças e expressões como seqüências ordenadas de símbolos. Em outras palavras: A lógica moderna, dita também lógica simbólica, adotava exatamente o mesmo procedimento (formal) proposto como Carnap; e Carnap, de fato, fará em *SLL* amplo uso de resultados anteriores, obtidos dentro do quadro de uma lógica simbólica formal essencialmente igual à dos sistemas que ele mesmo irá desenvolver. Voltamos, portanto, à indagação: O que exatamente Carnap propõe, que pode ser considerado de fundamentalmente novo?

Essa questão não poderia ser mais relevante para a compreensão do sistema de *SLL*, pois ela somente permite chegar ao fundo daquilo que é a "sintaxe lógica da linguagem" defendida por Carnap. A distinção essencial que temos de verificar, aqui, ocorre entre o método puramente formal, que Carnap deseja desenvolver, e aqueles métodos que, embora formais, baseavam-se porém em considerações não-formais. Para esclarecer essa idéia, devemos considerar novamente a maneira como se apresenta o estudo tradicional da lógica, assim como o estudo moderno da lógica simbólica, iniciado no século XIX.

Em um como em outro caso, há de fato uma abstração aos elementos formais das proposições. Mas em que consiste essa abstração? Consiste em que as relações lógicas entre proposições são *reveladas* pelos aspectos formais dessas proposições. O fato de uma proposição ter a forma "C é B" permite ver, independentemente dos termos concretos que se substituam por "C" e "B", que ela é conseqüência de sentenças que tenham a forma "Todo A é B" e "C é A", independentemente do termo concreto que se substitua por "A". A natureza dessas relações, por outro lado, sempre permaneceu controversa dentro da lógica. Ou por outra: Ao longo dos séculos, diferentes pensadores entenderam de diferentes maneiras – e buscaram justificar de diferentes maneiras – esse tipo de relações formais encontradas entre proposições.

Posto de outra maneira ainda: Certas *formas* proposicionais apresentam certas relações lógicas com outras *formas* proposicionais. Esse fato, pode-se dizer, nenhum lógico contestaria, e reside aí o aspecto inegavelmente formal da lógica – da lógica de Aristóteles bem como da de Russell ou de Carnap. No entanto, diante da indagação pelos elementos constitutivos dessas relações lógicas, o estudo tradicional da lógica recorria a aspectos não-formais da questão. Diante da indagação, "Por que tal forma proposicional está em tal relação lógica com tais outras formas proposicionais?", a lógica julgava-se obrigada a buscar explicações que ultrapassassem o campo sintático. E essas explicações, para além do campo sintático, socorriam-se freqüentemente de conceitos mais ou menos vagos – melhor ou pior articulados, segundo um ou outro autor – como os de "significado", "sentido" etc.

É precisamente para essa circunstância que Carnap deseja chamar a atenção, ao escrever o seguinte trecho (já citado anteriormente):

> *"But even those modern logicians who agree with us in our opinion that logic is concerned with sentences, are yet for the most part convinced that logic is equally concerned with the relations of meaning between sentences. They consider that, in contrast with the rules of syntax, the rules of logic are non-formal. (...)"* [32]

O mais correto (e certamente mais claro), da parte de Carnap, talvez fosse atenuar um pouco essa passagem. Talvez o melhor fosse dizer, dos outros lógicos, não que eles costumassem considerar as regras da lógica como não-formais, afirmação que, como vimos,

[32] *SLL*, pág. 1.

não é precisa; mas sim que eles buscavam compreender, ou justificar, ou ainda apoiar essas regras, que eram de fato regras formais, em algum tipo de consideração não-formal. Carnap, porém, estava plenamente certo na seguinte observação fundamental: Lógicos de todas as vertentes, e de maneira mais ou menos explícita, mostravam-se convencidos de que a ciência lógica deveria lidar com relações de significado entre sentenças – e "significado", aí, é uma noção que certamente ultrapassa o âmbito formal.

É nesse sentido que podemos compreender a insistência de Carnap a respeito do aspecto puramente formal da lógica, insistência essa que se oferece – já o observamos – como negação de um suposto apoio da lógica em outras ordens não-formais de consideração, em especial em esferas do "sentido" e do "significado". Retomamos, pela clareza, a seguinte citação, igualmente já apresentada:

> *"A theory, a rule, a definition, or the like is to be called* formal *when no reference is made in it either to the meaning of the symbols (for example, the words) or to the sense of the expressions (e.g. the sentences), but simply and solely to the kinds and order of the symbols from which the expressions are constructed."* [33] (destaque do autor)

Essa posição defendida por Carnap, de que a lógica não se baseia – em nenhum aspecto, em nenhum momento e sob nenhuma ordem de considerações – em elementos não-formais, é o que confere novidade ao pensamento de *SLL*. Para Carnap, a lógica é um estudo puramente sintático, que trata das relações formais entre seqüências de símbolos, e que institui essas relações de maneira totalmente arbitrária. Não há questão de justificar essas relações com base em aspectos extra-sintáticos. A lógica, no sentido mais absoluto da cópula, *é* sintaxe. Ela é o estudo abstrato dessas relações sintáticas – de quaisquer relações sintáticas que possam ser expressas com referência à estrutura combinatorial de símbolos.

Eis por que Carnap irá formular o seu Princípio de Tolerância: Porque, sendo a lógica absolutamente formal, não há razão para preferir um sistema de regras sintáticas a outro sistema de regras sintáticas, ambos formulados de forma igualmente clara e precisa. Todos os diferentes sistemas lógicos (sintáticos) merecem ser estudados: somente assim

[33] *SLL*, pág. 1.

eles terão sua estrutura corretamente analisada e poderão se tornar, conforme o caso, ferramentas úteis às diversas esferas do conhecimento científico, sem que haja contudo razão para supor uma prioridade necessária de um em relação a outro. Neste ponto, contudo, devemos reconhecer que o exame mais amplo e aprofundado dessa questão, introduzida pelo Princípio de Tolerância, ultrapassa os propósitos do presente capítulo. Adiemo-lo, portanto, até o Capítulo 3 deste trabalho.

Capítulo 2: Métodos infinitos em sintaxe

I

No capítulo anterior, observamos que o projeto formulado por Carnap em *SLL* baseia-se na idéia de substituir a filosofia pela análise sintática de linguagens formais. O livro, nesse sentido, divide-se em dois tipos de discussão. Por um lado, Carnap tenta mostrar a adequação de sua abordagem: Expõe as razões que o levaram a optar pelo ponto de vista sintático-formal, as vantagens que esse método oferece e sua fecundidade para o avanço da ciência.

Esses temas são tratados de maneira sistemática na introdução (seções 1 e 2) e, principalmente, na parte V de *SLL*, intitulada "Philosophy and Syntax" (seções 72 a 86). Há ainda, espalhadas ao longo do texto, diversas passagens nas quais questões desse tipo são debatidas. Isso acontece sempre que Carnap vê ocasião para examinar, em relação a determinado assunto específico, as conseqüências da visão filosófica defendida na obra.

Por outro lado, Carnap busca desenvolver, na prática, os principais aspectos do método formal por ele concebido. Não lhe interessa ficar apenas em um programa abstrato de pesquisa. Ele deseja mostrar, concretamente, de que maneira a abordagem formal pode ser utilizada para resolver problemas tradicionais em filosofia e em matemática. Parte substancial da obra, assim, é dedicada precisamente a esta tarefa: a discussão técnica, ao mesmo tempo minuciosa e abrangente, do método sintático.

Um brevíssimo sumário desse percurso pode ser inserido aqui, apenas para dar ao leitor uma idéia preliminar a respeito de como as diferentes etapas de desenvolvimento técnico encadeiam-se no livro. O melhor é indicar a própria divisão realizada pelo autor: Na parte I de *SLL* ("The Definite Language I", seções 3 a 17), Carnap constrói uma linguagem formal específica, denominada "linguagem I", dotada de certas características finitistas. Já na parte III do livro ("The Indefinite Language II", seções 26 a 40), Carnap constrói uma segunda linguagem formal, a "linguagem II", que não se atém às limitações finitistas impostas à linguagem I. Na parte II do livro ("The Formal Construction of the Syntax of Language I", seções 18 a 25), situada entre a construção de uma e outra linguagem, Carnap explica o método da numeração de Gödel, utilizado para formular aritmeticamente a sintaxe da linguagem I e, posteriormente (já no final da parte IV), para obter importantes resultados teóricos[34].

O que mais nos interessa neste momento, porém, é a parte IV de *SLL*, intitulada "General Syntax" (seções 41 a 71e). É nela que podemos identificar a espinha dorsal da discussão técnica proposta por Carnap deseja oferecer. A esse respeito, algumas observações são necessárias.

Considerado o projeto da obra como um todo, a posição de destaque da parte IV não surpreende. O objetivo central de *SLL* reside em mostrar que toda discussão filosófica realmente relevante (fecunda) pode e deve ser tratada como uma discussão relativa à estrutura interna das linguagens formais usadas para formular a ciência. Desse ponto de vista, não poderia haver tarefa mais importante para a filosofia (para a lógica) do que criar um quadro geral de análise dentro do qual a estrutura formal das linguagens pudesse ser devidamente desvendada, e por meio do qual os principais conceitos sintáticos pudessem ser identificados e estudados. É precisamente essa a tarefa que Carnap buscará avançar na parte IV de *SLL*.

A chamada "sintaxe geral", portanto, constitui a tentativa realizada por Carnap de esboçar um conjunto de definições sintáticas (conceitos sintáticos) que obedeçam aos seguintes requisitos. Em primeiro lugar, como já vimos no Capítulo 1 deste trabalho, essas definições devem ser exclusivamente formais, ou seja, devem referir-se unicamente às expressões da linguagem na sua qualidade de seqüências lineares de certos símbolos, sem fazer apelo a nenhuma instância que possa ser dita "transcendente".

Em segundo lugar, os conceitos sintáticos definidos devem ser suficientemente abrangentes para serem aplicáveis a qualquer linguagem formal que se possa conceber. A esse respeito, Carnap faz a seguinte observação, no início da seção 46 [35]:

> *"In this section we shall attempt to construct a* syntax for languages in general*, that is to say, a system of definitions of syntactical terms which*

[34] Diversos termos e idéias mencionados neste parágrafo, sem que contudo ficasse esclarecido seu conteúdo, serão retomados ao longo do trabalho.
[35] Situada na parte IV, a seção 46 é aquela na qual, após algumas considerações introdutórias – seções 41 a 45 – a sintaxe geral começa a ser exposta de maneira sistemática.

are so comprehensive as to be applicable to any language whatsoever.” [36] (destaque do autor)

Finalmente, esses conceitos devem ser, além de suficientemente abrangentes para garantir sua aplicabilidade a “qualquer linguagem que seja”, profundos a ponto de captar as características estruturais realmente relevantes de cada linguagem. Das três exigências, talvez a segunda e a terceira possam parecer contraditórias entre si. Na verdade, a situação é exatamente a inversa. As três estão intimamente ligadas umas às outras; e somente é possível compreender uma em relação com as outras duas. A estratégia adotada em *SLL*, nesse sentido, articula-se de maneira bastante conseqüente, que vale a pena examinar.

Carnap fornece uma definição geral – a qual já examinamos no capítulo anterior – para certo conjunto de estruturas simbólicas chamadas de “cálculos formais”. Essa definição geral, ao determinar qual a classe de estruturas relevantes, determina também a maneira – dita “sintática” – como elas devem ser encaradas (como o conjunto das combinações lineares possíveis a partir de certos símbolos separados em classes). Em outras palavras, a definição geral de cálculo formal *delimita* o âmbito válido de análise proposto por Carnap. Qualquer conceito, para ser admissível, necessita estar referido exclusivamente aos aspectos sintático-formais de um cálculo. Essa é, justamente, a primeira exigência que mencionamos acima: os conceitos da sintaxe geral (afirmação que neste momento já deveria soar um tanto quanto óbvia) devem ser formais.

Por outro lado, como todos os cálculos são estruturas de um mesmo tipo comum (indicado, justamente, pela definição geral de cálculo formal), muitos dos conceitos aplicáveis a um cálculo serão aplicáveis aos outros, e possivelmente a todos os outros. Basta, nesse último caso, que o conceito faça uso apenas das características gerais presentes na definição dessa classe de estruturas. Reside aí a possibilidade da segunda exigência mencionada. Isso não significa, obviamente, que não se possam formular, de maneira estritamente formal, conceitos específicos a uma linguagem, ou a um conjunto de linguagens. Por exemplo: podemos definir um conceito que se aplique às expressões de determinada linguagem se, e somente se, a expressão iniciar-se pelo símbolo “a” da

[36] *SLL*, pág. 167.

linguagem; embora esse conceito seja claramente formal, ele só faz sentido para linguagens que contém "a" entre seus símbolos.

Chegamos, assim, à terceira exigência. Certos conceitos, como vimos no exemplo acima, podem dizer respeito apenas à estrutura de uma linguagem específica (e podem mesmo ser bastante úteis para a descrição dessa linguagem). Não obstante, justamente os conceitos mais importantes serão aqueles cuja aplicabilidade é mais geral. Eles permitem obter uma análise, não apenas de uma linguagem e suas características, mas da própria classe de estruturas que são as linguagens formais. Em outras palavras: eles permitem analisar os aspectos formais que, aplicáveis a todos os cálculos desse tipo, os unificam como classe de estruturas aptas a desempenhar seu papel essencial como veículos de construção de teorias científicas.

Essa discussão ficará mais clara ao apresentarmos, neste e no próximo capítulo, os dois conceitos sintáticos mais importantes introduzidos por Carnap, e que obedecem justamente a essas três exigências: o conceito de "analiticidade" (que aparece ao lado de outros importantes conceitos, a ele estreitamente relacionados, como "validade", "sinteticidade" etc.) e o conceito de "conseqüência direta". O primeiro conjunto de conceitos (analiticidade e afins) irá se revelar fundamental para os propósitos do nosso trabalho. Sua análise, no entanto, terá de ficar reservada para o próximo capítulo. É necessário compreender e examinar, antes de qualquer outra coisa, o conceito de "conseqüência direta". É por meio desse conceito que Carnap busca fixar a própria essência do método formal. Dedicamos o restante do capítulo, portanto, a ele.

II

Para Carnap, o conceito de "conseqüência direta" é aquele no qual se expressam todas as regras formais de uma linguagem. No Capítulo 1 (ver seção II), verificamos como as linguagens formais se definem: Como *sistemas de regras* referidas a seqüências finitas (ditas "expressões") de símbolos, em relação aos quais tudo o que se pode supor é que estejam separados em classes. O conceito de "conseqüência direta", portanto, ao indicar a maneira como podem ser formuladas as regras formais, revela-se o mais característico do método formal. Para Carnap, são precisamente os limites estabelecidos para a formulação desse conceito – limites que, como veremos logo à frente, ele propõe estender de maneira significativa – que determinam o alcance de toda a abordagem sintática.

O melhor caminho para expor a nova concepção do método formal proposta por Carnap (baseada agora no conceito de "conseqüência direta") é por meio da comparação entre o seu método e os métodos tradicionais, utilizados até então na exposição de sistemas de lógica simbólica. Isso é possível porque, conforme observamos também no Capítulo 1 (ver seção V), a formulação de sistemas formais de lógica, por meio de regras sintáticas, não constitui uma inovação de Carnap. Pelo contrário: Carnap aproveita, com grande conhecimento de causa, todo o vasto repertório técnico legado por lógicos e matemáticos anteriores, que desde meados do século XIX dedicavam-se à tarefa de reorganizar essas disciplinas[37]. A comparação nos permitirá ver, com maior nitidez, em que sentido Carnap busca estender a concepção formal, para que essa possa servir aos propósitos que ele julga necessários.

Tradicionalmente, uma linguagem formal compõe-se de regras de dois tipos. Em primeiro lugar, regras de formação, cuja função está em indicar quais expressões da linguagem devem ser consideradas como "sentenças"[38]. Em segundo lugar, regras de transformação, que determinam quais sentenças do cálculo são conseqüência de quais

[37] Podemos destacar, entre os lógicos cujo trabalho havia sido essencial para a formação de Carnap, os nomes fundamentais de Russell e Hilbert e, em alguma medida, também aqueles de Brouwer, Ramsey, Chwistek e da escola polonesa (Lesniewski, Lukasiewicz etc.). A importância desses pensadores, sobretudo dos dois primeiros, fica evidente pelo número de referência que Carnap inclui a eles em *SLL*. O caso de Wittgenstein é bastante peculiar: sua influência sobre Carnap é suficientemente conhecida e debatida; não diz respeito, porém, às técnicas de lógica formal propriamente ditas, na medida em Carnap não assimilou nenhuma de suas propostas mais características (a respeito, por exemplo, da utilização de variáveis, do sinal de identidade etc.). É importante mencionar ainda, como ficará evidente no decorrer desta exposição, os nomes de Gödel e Tarski, contemporâneos com quem Carnap pôde debater grande parte de suas idéias. A relação entre o projeto articulado por Carnap em *SLL* e o trabalho desses dois gigantes da lógica e da matemática constituirá, mais à frente, tema específico de nossa análise.

[38] A importância da noção de "sentença", para uma linguagem formal qualquer, reside no fato de que somente as sentenças são capazes de expressar afirmações acerca do domínio de objetos para cuja descrição a linguagem foi desenvolvida. Em outras palavras: As sentenças são, do ponto de vista *descritivo*, as unidades básicas de uma linguagem, pois somente por meio delas é possível descrever um domínio de objetos. Essas questões, no entanto, não precisam nos preocupar agora. Do ponto de vista de *SLL*, elas dizem respeito à utilização de uma linguagem formal como instrumento científico. Uma tal discussão, portanto, ultrapassaria os aspectos puramente formais da linguagem, que devem – segundo o esquema fundamental de *SLL* – ser estudados de maneira independente (embora tal estudo, uma vez realizado, mostre-se útil justamente pela possibilidade que oferece de um tratamento mais rigoroso para as questões científicas, por meio da utilização, na descrição de domínios científicos, de linguagens rigorosamente construídas e analisadas).
O que importa, por enquanto, é o fato de que, tanto nos sistemas tradicionais como no de Carnap, uma linguagem deve possuir uma classe rigorosamente definida de expressões – as chamadas "sentenças" –, e que essa definição deve ser fornecida por meio de regras sintáticas, vale dizer, formais.

outras sentenças. As regras de transformação, portanto, condensam os processos dedutivos de uma linguagem, por meio dos quais é possível passar de um conjunto de sentenças (premissas) a uma outra sentença (conclusão). Finalmente, além das regras de formação e de transformação, aparecem também as chamadas "sentenças iniciais", conjunto de sentenças que, colocadas na base do sistema dedutivo, desempenham o papel de postulados a partir dos quais aplicar as regras de transformação. Essa era a abordagem usual dos diversos tratamentos formais que, até a época de publicação de *SLL*, haviam sido desenvolvidos para a lógica.

Tudo isso Carnap assimila dos sistemas anteriores. E é a partir justamente desse esquema que ele desenvolve o conceito de "conseqüência direta". Ao construir sua sintaxe geral, Carnap pretende fornecer o tratamento mais amplo e elegante possível para essas idéias, para poder determinar-lhes o exato alcance teórico. A primeira coisa que ele faz, assim, é unificar – sob um único conceito denominado "conseqüência direta" – tanto as regras de formação como as regras de transformação de uma linguagem (bem como a idéia de sentença inicial, que se apresenta para ele como um caso especial de conseqüência direta do conjunto vazio de sentenças). Eis como a questão é exposta no livro:

> *"We will assume the* definition of 'direct consequence' *to be stated in the following form: 'A1 is called a direct consequence of K1 in S if: (1) A1 and every expression of K1 has one of the following forms: ... ; and (2) A1 and K1 fulfill one of the following conditions: ...'. The definition thus contains under (1) the formation rules and under (2) the transformation rules of S."* [39] (destaques do autor)

A idéia de Carnap, na verdade, é simples: Como tanto as regras de formação como as regras de transformação devem ser formuladas – e já eram formuladas nos sistemas tradicionais – de maneira sintática, elas podem perfeitamente ser tratadas como subespécies de um único conceito, o conceito de "conseqüência direta", que condensa toda a estrutura formal da linguagem. Até aí, portanto, deve-se reconhecer que Carnap não oferece nada de essencialmente novo. Seu tratamento pode ser mais elegante, mais conciso e unificado, mas

[39] *SLL*, p. 169.

não acrescenta nenhuma idéia relevante aos métodos formais tais como eram estudados até então[40].

A verdadeira novidade que Carnap introduz, no seio do seu conceito de "conseqüência direta", para o estudo de sistemas formais – uma novidade a avaliação de cujo alcance e significado irá se revelar de importância capital para a compreensão de todo o projeto de *SLL* – reside na admissão (e no estudo sistemático) de certa possibilidade sintática até então desconsiderada. Estamos falando da idéia, pioneiramente vislumbrada por Carnap, de aceitar regras de transformação *indefinidas* na construção de cálculos formais. Mas o que são, exatamente, regras de transformação indefinidas?

As regras de transformação, como ficou dito, dizem respeito à estrutura dedutiva de uma linguagem. Elas estabelecem as relações de dedução consideradas como válidas entre as sentenças da linguagem. Sua formulação básica, em sistemas formais como o de *SLL*, assume o seguinte aspecto: A partir de certo conjunto K_1 de sentenças com determinada forma sintática, permite-se a obtenção (dedução) de outra sentença C_1, com determinada outra forma sintática (é exatamente esse mecanismo que Carnap procura expressar por meio do item "2" na passagem citada acima). No que segue, chamaremos esse conjunto K_1 de sentenças – a partir das quais a transformação (dedução, inferência) é feita – de "conjunto-base" da transformação.

Podemos agora esclarecer a novidade sugerida por Carnap. Em todos os sistemas tradicionais de lógica, o conjunto-base da transformação era visto como necessariamente finito. Em outras palavras: Todas as regras de transformação normalmente formuladas em lógica costumavam basear-se em conjuntos finitos de premissas (uma ou duas, nos casos típicos). Como é sempre possível examinar a forma sintática de um conjunto finito de sentenças, para compará-lo com a forma sintática da sentença a ser deduzida e verificar se a regra de transformação pode ou não ser aplicada, esse procedimento é chamado de "definido". É definido justamente na medida em que, para qualquer caso concreto, deve necessariamente chegar a uma decisão: ou a regra pode ser aplicada, ou não pode; ou uma

[40] Mais ainda, deve-se reparar que, apesar do tratamento unificado, Carnap mantém sempre em vista a distinção entre os dois tipos de regra, conforme ele próprio deixa claro no trecho citado acima. Na verdade, os dois tipos de regra – de formação e de transformação – desempenham funções bastante distintas na construção de qualquer sistema formal.

sentença é conseqüência de certas outras sentenças, ou não é. Não há margem para nenhuma indefinição.

Regras indefinidas de transformação, por outro lado, utilizam conjuntos-base infinitos de sentenças. Permite-se, no caso de tais regras, que uma sentença seja deduzida a partir de um número infinito de premissas. É fácil perceber, porém, que tipo de problema ameaça surgir: Contrariamente ao que acontecia na situação anterior, a inspeção de um número infinito de sentenças não é tarefa que possa ser diretamente realizada. Segue daí que não há nenhuma razão para supor que, uma vez formulada uma regra desse tipo, seja possível determinar sua aplicação (ou sua não-aplicação) em todos os casos concretos. É essa situação que, ainda neste capítulo, teremos de analisar detalhadamente.

III

A formulação de regras de transformação indefinidas[41] – bem como a tentativa de estudá-las sistematicamente – representa uma inovação significativa introduzida por Carnap[42]. Trata-se de proposta aparentemente estranha para uma abordagem, como a de *SLL*, que se propõe formal. Nesta seção, tentaremos indicar (embora tal tema só possa ser plenamente desenvolvido nos capítulos finais deste trabalho) a situação teórica que motivou essa inovação.

---

[41] Na seção precedente, vimos que Carnap defende a necessidade de formular regras indefinidas de *transformação*. E com relação às regras de formação? Elas poderiam ser indefinidas? Embora, de acordo com o Princípio de Tolerância, Carnap não considere proibido (ou impossível) construir linguagens dotadas de regras de formação indefinidas, ele considera que uma tal abordagem resultaria muito pouco prática. Como iremos mostrar, Carnap possuía razões específicas para introduzir as regras indefinidas de transformação. O mesmo, porém, não ocorria com as regras de formação, para as quais um tratamento estritamente definido parecia suficiente e adequado. Eis como ele expõe a questão:

> *"In connection with the use of indefinite syntactical terms in the construction of a particular language, we must above all differentiate the formation and the transformation rules. The task of the formation rules is the construction of the definition of 'sentence'. (...) Usually the rules are so qualified that not only the terms 'elementary sentence' and 'sentence-forming operation' but also the term 'sentence' is definite. In this case it can always be decided whether a particular expression is a sentence or not. Although the adoption of an indefinite term 'sentence' is not inadmissible, it would in most cases be inexpedient."* – *SLL*, pág. 166.

[42] O próprio Carnap observa que alguns outros autores, como Hilbert e Herbrand, já haviam esboçado algumas regras indefinidas para lidar com certas situações específicas dentro de seus sistemas de lógica. Não obstante, tais tentativas pareciam ainda muito tímidas, quer pelo escopo limitado de sua aplicação, quer pela falta de uma análise sistemática e abrangente do significado de novo método (ver *SLL*, pág. 173).

Por um lado, a idéia de utilizar regras de transformação indefinidas é estranha porque parece trazer, para dentro do estudo de sistemas de lógica simbólica, uma insegurança e uma incerteza em tudo contrárias ao método formal, e às razões que haviam motivado o interesse por esse tipo de método. De fato, não seria mesmo errado dizer que uma das principais vantagens que haviam estimulado o grande desenvolvimento da lógica formal naquele início de século XX residia justamente na circunstância de que ela podia ser formulada de maneira inequívoca e rigorosa, desprovida de quaisquer ambigüidades quanto à aplicação das regras.

Por que, então, Carnap mostra-se disposto a alterar essa situação de certeza e segurança? Por que ele sugere a utilização de regras de transformação indefinidas? Há uma razão importante para esse movimento, que precisamos agora verificar. A maneira mais fácil de fazer isso, embora não a única, é por meio da análise de uma situação matemática de grande interesse teórico e prático. Foi a consideração dessa situação matemática, além do mais – bem como de alguns resultados obtidos a respeito dela –, que deu a Carnap a certeza de que o caminho mais adequado para tratar o método sintático passava pela consideração de regras indefinidas. Vejamos, portanto, qual é ela.

Suponhamos que, em determinada linguagem S, apareça um predicado numérico P. Esse predicado P é dito numérico porque admite, como argumento, somente números ou variáveis numéricas. Ele expressa, nesse sentido, uma propriedade que pode ser atribuída (ou não) a um número, e somente a um número. "P(5)", por exemplo, é a sentença que indica que o número 5 possui a propriedade P; e "$\forall x P(x)$", em que x é uma variável numérica, é a sentença que indica que todos os números possuem a propriedade P.

Suponhamos agora que, nessa mesma linguagem S, seja possível obter, como teoremas (sentenças demonstráveis dentro do sistema), o conjunto K_P de sentenças: K_P={P(0), P(1), P(2), ... , P(n), ...}. Esse conjunto é infinito, pois possui todas as sentenças da forma "P(n)", em que "n" deve ser substituído por um número natural qualquer. Como K_P inclui *todas* as sentenças com essa forma, parece razoável supor que a sentença universal "$\forall x P(x)$" (a qual afirma justamente que *todas* as sentenças da forma "P(n)" são teoremas) possa ser vista como conseqüência lógica do conjunto K_P, e que deva aparecer, portanto, como teorema da linguagem S. Dito de outra maneira: Uma linguagem S na qual, embora seja possível demonstrar todas as sentenças da forma “P(n)”, não é possível demonstrar a correspondente sentença universal “$\forall x P(x)$”, não parece uma linguagem completa; não parece, por assim dizer, uma linguagem adequada para formular a matemática.

Cabe, porém, perguntar: Haveria alguma razão para supor que esse tipo de anomalia pudesse se instalar dentro de sistemas tão finamente organizados como os cálculos formais? Existiriam casos em que essa situação pudesse realmente ocorrer? Surpreendentemente, sim. Até a década de 1930, matemáticos e lógicos em geral sequer haviam considerado muito seriamente essa possibilidade[43]. No entanto, os surpreendentes resultados de Gödel – de 1930 e sobretudo de 1931[44] – mostraram justamente esse fato notável: que, em qualquer linguagem formal baseada somente em regras sintáticas definidas, existem propriedades numéricas para as quais, embora seja possível demonstrar todos os casos individuais, não é possível demonstrar a sentença universal correspondente. Tratava-se de um resultado, portanto, de conseqüências profundas para a lógica moderna e, particularmente, para toda a escola logicista de pensamento. Gödel havia demonstrado, não somente que neste ou naquele sistema formal uma tal situação indesejável *poderia* acontecer, mas que em *todo e qualquer* sistema formal de certo tipo – e especificamente em todos os sistemas formais “regulares”, baseados em regras de transformação definidas – essa situação *efetivamente* acontecia.

Eis, então, a razão que Carnap possuía para sugerir o estudo sistemático de regras indefinidas de transformação. Ao propor, por exemplo, uma regra sintática desse tipo para sua linguagem I – primeiro momento de *SLL* em que regras indefinidas de dedução são mencionadas, ainda no contexto de uma linguagem particular, e não no contexto geral da parte IV –, ele escreve[45]:

[43] Por trás dessa posição de lógicos e matemáticos, havia a crença (ainda que dificilmente formulada de maneira explícita) de que métodos definidos, tais como a “indução finita”, deveriam ser suficientes para dar conta de todas as situações do tipo indicado.
[44] Ver introdução.
[45] Na citação abaixo, alteramos a notação lógica utilizada por Carnap. Ao longo de *SLL*, Carnap introduz certos símbolos sintáticos góticos que, embora tornem sua exposição extremamente clara e precisa, demandariam excessivo espaço para serem explicados. É importante assinalar que as alterações realizadas no trecho abaixo resumem-se a questões de notação, que foi adaptada para uma forma mais facilmente reconhecível, sem prejuízo do sentido do texto; adicionamos também – via notas especiais indicadas por asteriscos – alguns esclarecimentos específicos, nos casos em que esses pareceram necessários.

"The case may arise where, for a particular Pr_l, say P1, every sentence of the form P1(n) is demonstrable, but not the universal sentence $\forall x P1(x)$. We shall encounter a Pr of this kind later on (§36). Although every individual case is inferable, there is no possibility of inferring the sentence $\forall x P1(x)$."* [46]

Carnap refere-se aqui, já sabemos, aos resultados de Gödel. E podemos agora formar um quadro mais amplo do deslocamento teórico que ele tentará articular. De certa maneira, o teorema da incompletude de Gödel havia acabado com o sonho logicista – do qual os *Principia Mathematica* é a manifestação mais grandiosa e acabada – de que sistemas lógicos formais definidos pudessem fornecer um tratamento completo (e consistente) para toda a aritmética e matemática clássica, por meio de regras de dedução finitas e bem determinadas. Diante do quadro de perplexidade geral que se instalou, Carnap acreditou ver a saída natural e honrosa para a proposta logicista que ele buscava reformular e aprofundar: Considerar seriamente o estudo de sistemas formais dotados de regras indefinidas de transformação[47].

IV

Na seção anterior, vimos algumas razões que levaram Carnap a propor a extensão do método formal para abarcar, não somente regras definidas, mas também regras indefinidas de transformação. Devemos investigar agora os argumentos que Carnap tem em mãos para justificar a validade, dentro de um esquema que se propõe sintático e formal,

---

* A notação "Pr", aqui, indica um predicado qualquer; o índice "l", subscrito, indica que o predicado é lógico (um predicado lógico, basicamente, é um predicado na definição do qual aparecem exclusivamente símbolos lógicos e matemáticos; em outras palavras, trata-se de um predicado que não depende de nenhum termo descritivo.)

[46] *SLL*, pág. 37.

[47] A situação teórica, na verdade, é mais complexa do que podemos expor neste momento. Em capítulos futuros, teremos de fundamentar detalhadamente nossa posição, analisando a proposta sintática de Carnap à luz não apenas dos resultados de Gödel (tal como originalmente expostos pelo lógico alemão), mas também de outros resultados análogos que não tardaram em aparecer, e em especial dos resultados de Tarski. Por enquanto, desejamos apenas indicar as razões imediatas – sem ainda justificar plenamente nossa afirmação – que Carnap possuía para introduzir o estudo de regras de transformação indefinidas.

dessa nova abordagem. Adiantamos desde logo, porém, que não encontraremos aí uma tarefa fácil, na medida em que Carnap é relativamente obscuro com relação a esse ponto.

De fato, não existe em *SLL* nenhuma discussão específica a respeito da admissibilidade de regras indefinidas de transformação. A melhor indicação que Carnap chega a oferecer, em relação a esse tema, encontra-se na seção 43 do livro, intitulada "On the Admissibility of Indefinite Terms". É importante deixar claro, porém, que os argumentos expostos por Carnap dizem respeito à admissibilidade de *termos indefinidos*, e não de *regras de transformação indefinidas*. Há entre esses casos, como veremos, semelhanças bastante relevantes. Contudo, não está claro até que ponto as duas situações possam ser tratadas da mesma maneira, tal como Carnap parece desejar. Para examinar essa questão, precisamos verificar, antes de qualquer outra coisa, como se definem os termos indefinidos. Eis o que Carnap escreve, no início da seção 43:

> *"We have called a defined symbol of language II* definite *when no unrestricted operator occurs in the chain of its definitions; otherwise,* indefinite *(§15)."* [48] (destaques do autor)

Para que um termo seja caracterizado como "indefinido", portanto, deve haver a ocorrência, em sua cadeia de definição, de ao menos um operador irrestrito[49]. Trata-se daquele tipo de operador (existencial ou universal) cuja quantificação abrange todos os infinitos termos numéricos de uma linguagem aritmética como II. Um termo indefinido, nesse sentido, é aquele em cuja definição aparece uma referência a "todos" os infinitos números naturais, ou à existência de um número natural qualquer não-especificado (e que pode ser, portanto, qualquer um entre "todos" os infinitos números naturais). O ponto

[48] *SLL*, pág. 160.

[49] Essa definição para "termo indefinido" é uma definição parcial, fornecida especificamente para a linguagem II. Em particular, trata-se de uma definição diferente daquela que Carnap irá formular mais à frente no livro, dentro do contexto da sintaxe geral. Mais ainda: a definição formulada na sintaxe geral revela-se incompatível com a definição apresentada aqui. Esse é um dos maiores problemas para quem quer que deseje realizar uma exposição das idéias contidas em *SLL*: diversos termos são redefinidos ao longo da obra, para assumir contornos bastante peculiares, distintos em todo caso da maneira como eram e são normalmente empregados em lógica formal. Da parte de Carnap, isso corresponde a um movimento bastante consciente. Para o leitor de um trabalho como o nosso, porém, essa circunstância exige redobrada atenção. Neste momento, importa reter o seguinte: Carnap fornece aqui uma definição de "termo indefinido" que, estando de acordo com a visão tradicional a respeito do tema, irá permitir-lhe examinar algumas opiniões correntes a seu respeito, com o objetivo de refutá-las. Para nós, essa discussão torna-se importante porque, justamente por meio dela, Carnap busca mostrar a validade da utilização de métodos e conceitos "infinitistas" em sintaxe.

importante a observar, aqui, é que a aplicação (ou não) de um tal termo, em certo caso concreto, deveria supor – ao menos em princípio – a verificação de um número infinito de outras sentenças.

Um exemplo típico – e de grande interesse prático – pode ser fornecido com relação ao teorema de Fermat (que, à época de Carnap, ainda não havia sido matematicamente demonstrado). Assim, para ilustrar essa questão, podemos definir (em uma linguagem aritmética básica qualquer) certo termo que denominaremos de "Fermatiano", da seguinte maneira: O número n é Fermatiano se e só se, para todo número natural a, e para todo número natural b, e para todo número natural c, vale que (a^n)+(b^n)≠(c^n) (na notação utilizada aqui, o símbolo "^" indica a operação de exponenciação). Ou, em notação simbólica: Fermatiano(n) ≡ ∀a∀b∀c [(a^n)+(b^n)≠(c^n)].

Vemos assim que a definição do termo "Fermatiano", que é um predicado numérico, estabelece que ele só pode ser aplicado a um número natural n quando certa fórmula envolvendo n – a saber, a fórmula "(a^n)+(b^n)≠(c^n)" – for válida para todos os números naturais a, b e c. A questão essencial reside na circunstância de que, para saber se o termo assim definido aplica-se (ou não) a um número n qualquer, é necessário, *em princípio*, verificar uma série infinita de fatos matemáticos: mais precisamente, é preciso verificar todas as infinitas fórmulas do tipo (a^n)+(b^n)≠(c^n), em que os símbolos "a", "b" e "c" são substituídos por números naturais[50].

---

[50] O exemplo escolhido talvez pareça mais complicado do que o necessário. Afinal, também a definição de um predicado simples como "número primo" pode envolver, em princípio, uma quantificação irrestrita. Isso ocorre, por exemplo, na seguinte definição usual (definiremos aqui o predicado indefinido "primo_1"): "Um número natural x é primo_1 se e somente se, para **todo número natural n,** diferente de x e de 1, não for verdade que x pode ser dividido por n".
Acontece que essa definição, na qual aparece uma quantificação irrestrita ("para todo número natural n") pode ser trivialmente substituída por uma definição equivalente, na qual a quantificação é restrita (definiremos agora um segundo predicado, definido, "primo_2"): "Um número natural x é primo_2 se e somente se, para **todo número natural n menor ou igual a x**, diferente de x e de 1, não for verdade que x pode ser dividido por n".
A quantificação é restrita porque, como se vê, faz referência não a "todo número natural n", mas somente a "todo número natural n menor ou igual a x".
O próprio Carnap comenta a respeito dessa situação. Segundo ele, no caso desses dois predicados distintos – primo_1 e primo_2 – é fácil demonstrar a igualdade "primo_1= primo_2" (ou seja, "primo_1(x) se e somente se primo_2(x)"). O que acontece é que se conhece, para o termo indefinido "primo_1", um sinônimo definido: o termo "primo_2". Nesse caso, torna-se de certa maneira fácil saber se o predicado indefinido "primo_1" aplica-se ou não a um número, em qualquer caso concreto.
O mesmo não acontece, porém, em relação ao predicado "Fermatiano", para o qual não parece haver nenhum sinônimo definido. É por essa razão que, no corpo do texto, nós preferimos definir um predicado desse tipo,

Vemos aí, portanto, a semelhança que irá surgir, para Carnap, entre a admissão de termos indefinidos em uma linguagem (os quais, a despeito de oposição de algumas vozes[51], eram comuns em quase todos os sistemas lógicos) e a admissão de regras indefinidas em uma linguagem (uma inovação própria sua). Termos indefinidos, para terem sua aplicação determinada em um caso concreto, envolvem – ao menos em princípio – a verificação de um número infinito de sentenças. Já as regras indefinidas de transformação são formuladas com referência a um conjunto infinito de premissas (sentenças), cuja inspeção seria necessária para determinar sua aplicação.

Carnap parece acreditar que as duas questões devam ser colocadas em pé de igualdade, embora não ofereça razões explícitas para tal suposição. Essa sua posição fica clara na passagem que oferecemos a seguir. Ela faz parte da seção 48 ("c-Terms") da parte IV, em que Carnap inicia a exposição dos métodos indefinidos na sintaxe geral:

> *"*In almost all known systems, only definite rules of transformation are stated*, that is to say, only d-rules. But we have already seen that it is possible to use also* indefinite syntactical terms *(§ 45).*" [52] (destaques do autor; grifos meus)

Vemos aqui como Carnap trata paralelamente a questão das "*regras* definidas de transformação" e a possibilidade de "*termos* sintáticos indefinidos"[53]. Necessitamos verificar, assim, como Carnap procura justificar a admissibilidade de termos indefinidos, para avaliar se seus argumentos podem, de fato, ser transpostos para o caso de regras indefinidas. Abaixo, oferecemos uma passagem completa em que Carnap realiza essa discussão, para poder examinar todos os detalhes importantes[54]:

> *"The lack of a method of resolution for indefinite terms has induced many logicians to reject these terms altogether, as meaningless (e.g. Poincaré,*

---

cuja aplicação a qualquer caso concreto realmente depende, em princípio, da verificação de um número infinito de sentenças.

[51] Carnap cita Poincaré, Brouwer, Wittgenstein e Kaufmann – ver *SLL*, pág. 161.

[52] *SLL*, pág. 172.

[53] A referência que aparece no trecho citado, como se vê, é à seção 45. Essa seção, porém, é quase um complemento da seção 43, na qual se encontra toda a discussão relevante.

[54] Na citação abaixo, alteramos novamente a notação com símbolos góticos usada por Carnap. A esse respeito, ver nota 45 acima. (Também aqui, nos casos necessários, introduziremos notas especiais indicadas por asteriscos.)

Brouwer, Wittgenstein, and Kaufmann). Let us consider as examples two indefinite $^{1}Pr_{l}$*, 'P_1' and 'P_2' (in II, for example), which, by means of a definite* $^{2}Pr_{l}$**, 'Q', may be defined in the following manner:*

$$P_1(x) \equiv (\exists y)\,(Q(x,y)) \qquad \text{(I)}$$

$$P_2(x) \equiv (\forall y)\,(Q(x,y)) \qquad \text{(II)}$$

The logicians referred argue roughly as follows: the question whether, for instance, '$P_1(5)$' (or '$P_2(5)$') is true or not, is meaningless, inasmuch as we know of no method by which the answer may be sought, and the meaning of a term consists solely in the method of determination of its applicability or non-applicability. To this it may be replied: it is true that we know of no method of searching for the answer, but we do know what form the discovery of the answer would take – that is to say, we know under what conditions we should say that the answer had been found. This would be the case, for example, if we discovered a proof of which the last sentence was '$P_1(5)$'; and the question whether a given series of sentences is a proof of this kind or not is a definite question. Thus there exists the possibility of the discovery of an answer*, and there appears to be no cogent reason for rejecting the question."* [55] (destaque do autor)

Carnap, portanto, parece justificar a admissão de termos indefinidos pela *possibilidade*, em princípio existente, de decidir com relação à sua aplicação nos diversos casos concretos (mesmo quando não se conheça um método uniforme que possa sempre conduzir a essa decisão). De maneira mais essencial ainda, ele se refere à capacidade de *reconhecer* uma resposta como tal, ou seja, de saber sob que condições uma resposta teria sido encontrada[56].

---

[*] O símbolo "Pr" indica um predicado numérico qualquer; a letra "l", subscrita à direita, indica que se trata de um predicado lógico; já o número "1", sobrescrito à esquerda, indica que o predicado possui apenas um único argumento numérico (predicado *unário*).

[**] O número "2", sobrescrito à direita, indica um predicado com dois argumentos numéricos (predicado *binário*).

[55] *SLL*, pág. 161.

[56] Na próxima seção, teremos ocasião de inspecionar essas questões de maneira mais aprofundada.

Esses argumentos são, possivelmente, excelentes; e não devemos esquecer que essa visão de Carnap, a respeito dos termos indefinidos, foi aquela que, historicamente, acabou prevalecendo (embora até hoje sejam discutidas diferentes formulações ditas "intuicionistas" ou "finitistas" para a lógica formal). Para a questão que estamos discutindo, contudo, isso ainda é muito pouco. Precisamos estabelecer em que medida, e de que maneira, essa posição de Carnap poderia ser traduzida em uma posição acerca da admissibilidade de regras indefinidas de transformação, dentro de um sistema que se deseje dizer "formal". Trata-se de uma investigação que realizaremos, na próxima seção, com referência aos pressupostos que o próprio Carnap coloca na base do método formal.

V

O método formal, conforme observamos no Capítulo 1 deste trabalho, baseia-se na possibilidade, tida como não-problemática por Carnap, de algumas operações simples, executadas sobre símbolos arbitrários. São elas: o reconhecimento de símbolos e seu agrupamento em diferentes classes; a possibilidade de formar seqüências a partir de um conjunto de símbolos; e a possibilidade de comparar diferentes seqüências de símbolos. Uma regra de transformação indefinida, por sua vez, envolve um número infinito de sentenças, situação essa que parece problemática. Precisamos verificar exatamente, portanto, em que ponto residiria o problema.

Em relação ao conjunto de símbolos que compõem uma linguagem, não há nenhuma diferença entre o caso de linguagens com regras indefinidas de transformação ou sem esse tipo de regras. O conjunto de símbolos, tanto em um como em outro caso, pode ser finito ou infinito, e nada disso terá a menor relevância para a situação de que estamos tratando.

Também não é difícil ver que o problema proposto pelas regras de transformação indefinidas não diz respeito à possibilidade de formar seqüências a partir dos símbolos de uma linguagem. Consideradas individualmente, cada uma das infinitas sentenças que pertencem ao conjunto-base de uma regra de transformação indefinida continua sendo, simplesmente, uma seqüência de símbolos. O fato de elas figurarem em número infinito na regra não modifica sua natureza individual. Cada uma delas é uma seqüência de símbolos trivialmente aceita como parte do cálculo; e nem se coloca em dúvida que possam existir infinitas seqüências formadas com certos símbolos. Nada mais está em jogo aí do que um fato combinatorial simples: sob certas condições, um conjunto de símbolos pode combinar-se de infinitas maneiras.

O problema, portanto, só poderia residir na necessidade – implícita em regras de transformação indefinidas – de comparar infinitas seqüências de símbolos. Uma regra sintática de inferência, ao indicar os casos em que se pode passar de um certo conjunto de sentenças (premissas) a outra sentença (conclusão), supõe a comparação entre essas diferentes seqüências de símbolos que são as sentenças envolvidas. Aplicar uma regra formal de inferência, nesse sentido, significa verificar a forma sintática do conjunto de premissas para cotejá-la com a forma sintática da conclusão. No caso de regras indefinidas, as premissas são em número infinito. Mas será que essa circunstância constitui, de fato, um problema?

Para entender corretamente essa questão, precisamos verificar a maneira como um conjunto-base contendo infinitas sentenças pode figurar em uma regra indefinida de transformação. Mais precisamente, é necessário chamar a atenção para o fato de que esse conjunto-base necessita, para fazer parte de uma regra sintática (e, portanto, de uma linguagem), ser descrito. Isso fica claro se atentarmos, mais uma vez, para a passagem em que Carnap indica como devem ser formuladas as regras de conseqüência direta (essa passagem já foi citada, anteriormente, na seção II acima):

> *"We will assume the* definition of 'direct consequence' *to be stated in the following form: 'A1 is called a direct consequence of K1 in S if: (1) A1 and every expression of K1 has one of the following forms: ... ; and* <u>*(2) A1 and K1 fulfil one of the following conditions: ...'*</u>*. The definition thus contains under (1) the formations rules and under (2) the transformation rules of S."* [57] (destaques do autor; grifo meu)

A menção, no trecho sublinhado acima, a certas *condições*, a que o conjunto-base da transformação deve obedecer, pode parecer um pouco vaga demais. Não obstante, uma

[57] *SLL*, p. 169.

vez indicadas, como elementos constituintes da regra, essas condições, fica perfeitamente descrito certo conjunto (o conjunto-base) de sentenças: mais precisamente, aquele conjunto de sentenças que satisfazem à condição sintática formulada. Isso significa que, dada qualquer sentença, é possível saber se ela faz parte (ou não) de determinado conjunto-base, necessário à aplicação de uma regra específica.

O fato de essas sentenças aparecerem, eventualmente, em número infinito, acaba sendo, desse ponto de vista, menos problemático do que em princípio parecia. Por trás dessa afirmação que fazemos, reside a constatação de que não é essencialmente problemático descrever, de maneira inequívoca, um número infinito de formas sintáticas. Basta, para tanto, capturar uma regularidade qualquer na organização simbólica de certo conjunto de expressões. O caso mais típico irá aparecer quando o conjunto for composto por todas as expressões que apresentem determinada forma sintática, em que uma das posições sintáticas possa ser variavelmente ocupada por qualquer símbolo de uma classe infinita de símbolos (ou por qualquer sub-expressão de uma classe infinita de sub-expressões).

Pertence a esse caso típico o exemplo matemático que orientou nossa exposição na seção III acima. Ali, fizemos referência ao conjunto infinito de sentenças $K_P=\{P(0), P(1), P(2), \ldots, P(n), \ldots\}$. Esse conjunto, embora infinito, está perfeitamente caracterizado do ponto de vista sintático. Aplicar uma regra de transformação que utilize esse conjunto como conjunto-base, do ponto de vista da comparação de seqüências de símbolos, é tão claro como aplicar uma regra de transformação definida, por exemplo, uma regra que permita a dedução da sentença "A e B" a partir do conjunto {A,B} de sentenças. Nos dois casos, sabemos exatamente quando uma e outra regra podem ser aplicadas para a obtenção da conclusão. No caso definido, quando as duas sentenças A e B estiverem presentes (entre os teoremas da linguagem, ou como premissas específicas de uma dedução hipotética). No caso indefinido, quando as infinitas sentenças do conjunto K_p, perfeitamente definido, estiverem presentes.

Um pouco de reflexão, aqui, mostra que a infinitude do conjunto K_p é de menor importância. A aparente dificuldade reside no fato de que, para a aplicação da regra indefinida em uma demonstração concreta, seria necessário possuir todas essas infinitas sentenças de K_p como teoremas, o que seria supostamente mais difícil do que possuir duas sentenças como teoremas. A questão, porém, não reside em saber qual situação é mais difícil de ocorrer na prática (e mesmo essa suposição está equivocada). A questão está em perceber que ambas as situações são igualmente claras do ponto de vista formal. Para ver isso, talvez seja adequado considerar o que se passa, em detalhes, no caso definido que tomamos como exemplo.

Nesse caso definido, encontramos uma regra – todos irão aceitar – bastante clara: Se duas sentenças específicas C2 e C3 forem teoremas, então a sentença C1 = "C2 e C3" também é teorema. Para a aplicação dessa regra em um caso concreto, no entanto, precisamos saber: As sentenças C2 e C3 são teoremas? Só assim poderemos estabelecer se a regra pode ou não ser aplicada para permitir a obtenção da conclusão C1 = "C2 e C3".

Infelizmente, essa pode ser uma questão totalmente indeterminada. Em muitos casos – e em muitos casos relevantes para a matemática e a aritmética – simplesmente não se sabe se certas sentenças (em número finito ou infinito) são ou não teoremas de uma linguagem. Talvez seja muito difícil vir a sabê-lo; talvez (fato esse que o teorema de Gödel demonstrou, ao construir sentenças indecidíveis), seja mesmo impossível vir a sabê-lo. Suponhamos que a sentença C2 seja o teorema de Fermat, e que a sentença C3 seja a conjectura de Goldbach. A aplicação da regra definida, nesse caso concreto, é de decisão extremamente difícil, na medida em que implica saber se duas complicadas proposições matemáticas (a segunda até hoje sem solução) são ou não teoremas de determinada linguagem formal. O conteúdo da regra de inferência, contudo, não poderia ser mais trivial: permitir a dedução, a partir das duas sentenças que são o teorema de Fermat e a conjectura de Goldbach, da sentença que é a conjunção (afirmação conjunta) de ambos, teorema de Fermat e conjectura de Goldbach.

O mesmo se pode dizer de uma regra de transformação indefinida, como a que estamos usando de exemplo. Também ela – há que se reconhecer agora – é bastante clara: Se as sentenças do conjunto K_p forem teoremas, então a sentença "$\forall x P(x)$" também é teorema. O problema, novamente, reside na aplicação da regra em um caso concreto qualquer, ou seja, na eventual dificuldade em saber se todas as infinitas sentenças de K_p são de fato teoremas. Mas se trata de um problema que, como acabamos de ver, pode surgir igualmente para o caso definido, ou seja, para o caso de um número finito de sentenças no conjunto-base, sem que com isso a regra perca sua clareza ou seu conteúdo como regra formal. Mais do que isso, é perfeitamente possível saber, em muitos casos, que infinitas sentenças são teoremas. Isso acontecerá, por exemplo, se possuirmos uma demonstração desse fato na linguagem sintaxe (ou uma "meta-demonstração", como poderíamos chamá-la), ou se possuirmos um esquema sintático de provas para as demonstrações de cada uma das sentenças do conjunto K_p. Em todo caso, o problema não diz respeito, em nenhum sentido, ao conteúdo ou à possibilidade da regra como regra formal.

Podemos ver agora em que sentido a argumentação de Carnap, oferecida para justificar a admissibilidade de termos indefinidos, pode ser transportada para uma discussão acerca da admissibilidade de regras indefinidas de transformação. No extenso trecho que citamos na seção IV, o raciocínio que Carnap desenvolve parece seguir exatamente a linha que tentamos esclarecer.

Tratava-se, ali, de mostrar a admissibilidade de um termo indefinido, como por exemplo o predicado numérico P_2, cuja definição era: $P_2(x) \equiv (\forall y)\ (Q(x,y))$. O aparente problema, apontado por alguns pensadores em relação a predicados indefinidos como P_2, residiria no fato de que, para poder decidir sua aplicação a qualquer caso concreto, digamos $P_2(5)$, seria necessário saber (ter como teorema) um conjunto infinito de sentenças, no caso o conjunto $K_{q,5}=\{Q(5,0), Q(5,1), Q(5,2), \ldots , Q(5,n), \ldots \}$. Contudo, Carnap defende que esse não deve ser visto um verdadeiro problema do método formal, capaz de invalidar todas as construções "não-finitistas". Tal problema, na verdade, surgiria apenas se supuséssemos a necessidade de saber, sempre, se o predicado P_2 é ou não aplicável a um caso concreto.

Como mostramos acima, porém, trata-se de um tipo de exigência pouco razoável. Levado ao limite, ele atenta até mesmo contra a possibilidade de aceitar simples regras de transformação definidas, da espécie mais trivial possível. Quando Carnap observa que "*... it is true that we know of no method of searching for the answer, but we do know what form the discovery of the answer would take – that is to say, we know under what conditions we should say that the answer had been found*", o que ele está afirmando é que o conteúdo do termo indeterminado é perfeitamente claro, para além da possibilidade de estabelecer sua aplicação a cada caso concreto. Da mesma maneira, o conteúdo de uma regra de transformação *definida* é perfeitamente claro, para além da possibilidade de estabelecer sua aplicação a cada caso concreto (o que, como vimos, nem sempre é possível). Finalmente, o conteúdo de uma regra de transformação *indefinida* também é perfeitamente claro – desde que ela esteja formulada de maneira formalmente correta, com o conjunto-base de sentenças corretamente descrito.

Uma vez que meditamos sobre esse assunto, o quadro teórico vislumbrado por Carnap afigura-se bastante coerente. Pois, ao descrever um conjunto infinito de sentenças, como aquelas que compõem o conjunto K_P ou $K_{q,5}$, é possível considerar que estamos realmente comparando um número infinito de seqüências de símbolos, para subsumi-los sob uma mesma forma. A mera possibilidade de falar do conjunto bem determinado K_P={P(0), P(1), P(2), ... , P(n), ... } envolve a possibilidade de comparar todas as infinitas sentenças do conjunto para verificar que elas compartilham uma mesma forma sintática, a forma "P(α)" (em que o símbolo "α" pode ser substituído por qualquer número natural n), e a possibilidade de comparar essas infinitas sentenças com a sentença que figura na conclusão, a saber, "$\forall x P(x)$".

Inversamente, a possibilidade de se falar na *forma sintática* "P(α)" (em que o símbolo "α" pode ser substituído por qualquer número natural n) envolve a possibilidade de desdobrar essa forma sintática em suas infinitas manifestações. Todas essas infinitas manifestações, por sua vez, são infinitas seqüências de símbolo da linguagem, sobre as quais efetuou-se, *ipso facto*, uma operação de comparação.

Dessa maneira, podemos concluir que a argumentação oferecida por Carnap para justificar a admissibilidade de termos indefinidos poderia servir também, com as devidas adaptações, para justificar a formulação de regras de transformação indefinidas para uma linguagem formal, sem que ela perca, com isso, seu *status* formal. Chegamos, portanto, ao seguinte ponto: Se considerarmos o que significa, para o sistema formal de *SLL*, a introdução de regras de transformação indefinidas, vemos que esse movimento teórico de Carnap encontra-se ao mesmo tempo motivado por importantes resultados (vale dizer, pela necessidade de superar certas limitações intrínsecas que, conforme havia sido revelado pelos teoremas de Gödel, restringiam o alcance de sistemas lógicos dotados exclusivamente de regras definidas – ver seção II acima) e justificado do ponto de vista do tratamento formal (de acordo com a visão do método formal defendida no livro – tema que abordamos na presente seção).

Ainda resta, contudo, um problema: a aplicabilidade dessas regras. Nos parágrafos acima, vimos que os problemas relacionados à aplicabilidade de uma regra indefinida não devem atentar contra o caráter formal desse tipo de regra, assim como não atentam contra o caráter formal de regras definidas (cuja aplicação concreta, já sabemos, pode revelar-se igualmente problemática). Não obstante, embora situações desse tipo não descaracterizem um sistema como formal, outros problemas irão surgir. É importante, porém, registrar desde agora: São problemas que derivam, não tanto da formulação de regras de transformação indefinidas, mas das conseqüências que Carnap espera poder tirar dessa nova possibilidade por ele vislumbrada. Deixaremos esse assunto, contudo, para ser analisado no capítulo seguinte.

VI

Do ponto de vista da sintaxe geral, eis o fato essencial que é necessário agora ressaltar, como conclusão do presente capítulo: O conceito de "conseqüência direta", da maneira como Carnap o propõe (enriquecido por regras de transformação indefinidas), deve fixar a essência de toda a abordagem formal. É por meio dele – dentro de seus limites e segundo suas possibilidades – que se expressa toda a multiplicidade cabível de estruturas sintáticas. Visto do ponto de vista de cada linguagem formal, mais ainda, esse fato se traduz da seguinte maneira: As regras de conseqüência direta, peculiares a certa linguagem, esgotam (do ponto de vista sintático) completamente essa linguagem, e expressam tudo o que há para expressar relativamente a ela.

Uma vez fornecido o conceito de "conseqüência direta" para uma linguagem formal, tudo o que pode ser dito a seu respeito já está dito; toda a sua estrutura já se encontra plenamente determinada; todas as suas características formais, e todas as características formais de suas expressões, já estão estabelecidas. Eis um trecho em que Carnap enfatiza exatamente essa situação:

> *"In the treatment of Languages I and II we introduced the term 'consequence' only at a late stage.* From the systematic standpoint, however, it is the beginning of all syntax. If for any language the term 'consequence' is established, then everything that is to be said concerning the logical connections within this language is thereby determined. *(...)"* [58] (destaque do autor)

Certamente, será necessário estudar e analisar a linguagem, para desvendar sua estrutura e obter os resultados que ela pode oferecer. No entanto, trata-se de uma estrutura que já se encontra plenamente determinada. Se meditarmos, agora, na maneira como se propõe o esquema sintático de *SLL*, vemos que essa posição de Carnap é perfeitamente natural. Na verdade, ela aparece como conseqüência necessária das questões tratadas no Capítulo 1 deste trabalho. Lá, mostramos que os cálculos formais ficavam definidos como sistemas de regras referidas a símbolos e seqüências de símbolos. Aqui, percebemos que essa definição é seguida com grande coerência por Carnap, com um rigor que outros lógicos, freqüentemente, não haviam sabido manter. Uma linguagem formal *é* as regras (de conseqüência direta) que a definem; portanto, do ponto de vista formal, todas as suas características já estão determinadas por essas regras.

Após as análises precedentes, estamos também em condições de compreender essa posição e esse esforço de Carnap – no sentido de mostrar que uma linguagem formal encontra-se plenamente determinada por suas regras de conseqüência direta – sob uma nova perspectiva. Para tanto, é interessante comparar suas propostas, mais uma vez, com a maneira usual de conceber o método formal.

Logo de cara, devemos reconhecer que, ao menos em princípio, Carnap apenas explicita (e enfatiza) algo que já está implícito em qualquer abordagem formal para a lógica. Sistemas formais foram desenvolvidos exatamente para que a tarefa de dedução ficasse reduzida à aplicação de certos passos lógicos formais, que por sua vez estavam completamente descritos em um sistema de regras simples e claras. Uma vez construído um cálculo formal, a atividade relevante passava a consistir, aparentemente, na dedução de teoremas dentro do cálculo. Nesse sentido, a pergunta mais importante que se poderia fazer

[58] *SLL*, p. 168.

em relação a uma sentença qualquer era: Tal sentença é, ou não é, um teorema do cálculo? Não é difícil ver que essa pergunta deveria ser respondida com referência única e exclusivamente às regras que definiam o cálculo (eis, de fato, uma maneira de reformulá-la que deixa bem claro esse ponto: A sentença pode, ou não pode, ser deduzida pelas regras do cálculo?).

Em *SLL*, contudo, vemos Carnap insistir, e com grande ênfase, a respeito desse fato, que poderia ser visto como trivial: de que as regras de conseqüência direta caracterizam completamente uma linguagem formal. Por que essa insistência? Há dois motivos principais que a justificam. Em primeiro lugar, porque Carnap desejava expurgar a lógica de absolutamente quaisquer considerações que ultrapassassem o plano sintático-formal. Conforme já indicamos no Capítulo 1, até mesmo em tratamentos formais para a lógica certas considerações extra-formais eram amplamente utilizadas na hora de justificar – ou de tentar justificar – o conjunto de regras sintáticas adotadas. Acreditava-se que um ou outro conjunto de regras, vale dizer, que um ou outro sistema formal deveria mostrar-se correto (ou não) em função de certas relações lógicas que transcendiam (eram anteriores a) o plano formal (para a maioria dos pensadores, relações de significado entre as sentenças). Por isso mesmo, a importância da ênfase adotada por Carnap, a esse respeito, é bastante considerável. Trata-se de assunto que merecerá, para ser devidamente discutido, todo o Capítulo IV, no qual examinaremos o Princípio de Tolerância proposto em *SLL*.

Em segundo lugar, porém, a ênfase com que Carnap defende seu ponto de vista sintático explica-se – podemos agora perceber – pela nova situação teórica que havia sido criada pelos teoremas de Gödel. Aqui, a comparação com abordagens formais anteriores impõe-se de maneira ainda mais relevante para a compreensão de *SLL*. Até a publicação dos resultados de incompletude de Gödel, acreditava-se que sistemas lógicos formais, do tipo baseado em regras definidas de dedução, poderiam captar todas as verdades aritméticas e matemáticas, bem como todas as verdades lógicas. De fato, esses três conceitos sequer ficavam claramente distinguidos. Verdade matemática, verdade lógica e verdade dentro de um sistema formal (desde que ele fosse “correto”) eram tratadas – e parecia possível tratá-las – como se fossem uma única e mesma coisa[59].

Os resultados de Gödel, no entanto, despertaram os logicistas para este fato bastante incômodo: Aparentemente, nenhum sistema formal – pelo menos nenhum do tipo até então conhecido e estudado – seria capaz de esgotar sequer um conjunto tão básico de verdades como as verdades aritméticas. E é justamente por esse motivo, como vimos, que Carnap irá sugerir a formulação e o estudo sistemático de um novo tipo de sistemas formais, dotados de regras indefinidas de transformação.

Considerados apenas os sistemas lógicos tradicionais, no entanto, o que os resultados de Gödel mostravam é que não havia coincidência possível entre o conceito de “verdade formal” e o de “verdade aritmética”. Surgia, assim, uma oposição interessante. O conceito de “verdade formal” – ou de “verdade dentro de um sistema formal” – era certamente bastante claro: eram “verdades formais” de um sistema aquelas sentenças que pudessem ser deduzidas por meio de suas regras. Não obstante, não se sabia mais ao certo o que um tal conceito captava, pois o teorema de Gödel havia mostrado que ele era e seria sempre insuficiente até mesmo para captar as simples verdades aritméticas. Inversamente, esse conceito de “verdade aritmética”, implícito no teorema de Gödel, parecia bastante incerto do ponto de vista de sua compreensão teórica: ficava difícil enxergar o quê, exatamente, poderia estar por trás dele. Contudo, ninguém que realmente houvesse compreendido a situação estava disposto a negar, dada a clareza com que ele emergia da demonstração de Gödel, seu significado e sua capacidade de captar fatos essenciais a respeito das estruturas matemáticas.

De maneira pitoresca, poderíamos dizer que as “verdades formais” dos diversos cálculos lógicos eram indubitavelmente claras, mas pareciam-se cada vez menos com verdades; já as “verdades aritméticas” de Gödel eram obscuras, mas eram indubitavelmente verdades. Parecia haver, portanto, algo especificamente não-formal por trás de verdades

[59] Aqui e no restante deste capítulo, estamos usando o termo “verdade” de maneira propositalmente vaga, para abranger todos os conceitos e definições que, segundo uma ou outra posição teórica, costumavam ser utilizados, com maior ou menor grau de discernimento. No Capítulo 5, teremos de examinar detalhadamente as razões que levaram Carnap a rejeitar de seu sistema, completamente, todo e qualquer conceito de “verdade”.

lógicas e de verdades matemáticas, as quais não podiam coincidir nem mesmo extensionalmente com as verdades formais dos diferentes sistemas até então estudados.

Carnap não aceita essa maneira de ver o problema. Como já sabemos, sua solução é outra: Ele passa a estudar um novo tipo de sistemas formais, baseado em regras indefinidas de transformação, que escapava às condições de demonstração do teorema de Gödel. Em sistemas como os sugeridos por Carnap, poderiam ficar eliminadas as embaraçosas restrições que ameaçavam toda a abordagem formal. Carnap estava disposto a oferecer um novo esquema de compreensão do método formal – uma extensão desse método – no qual as verdades do teorema de Gödel podiam, sim, ser capturadas.

Contrariamente ao que o teorema de Gödel parecia anunciar, portanto, Carnap desejava oferecer um método (e acreditava tê-lo encontrado) suficientemente poderoso para tratar toda a aritmética e matemática clássica (e, por isso mesmo, adequado para realizar ciência), e suficientemente rigoroso para ser dito formal. Mas, justamente porque essa tarefa tinha se tornado improvável, e porque cada vez mais se suspeitava que seria inevitável encontrar algo que fosse além (ou que estivesse por trás) dos esquemas meramente formais, Carnap necessitava enfatizar o caráter exclusivamente sintático de suas técnicas.

Capítulo 3: O primeiro conceito de analiticidade

I

Após haver examinado, no capítulo precedente, o conceito de "conseqüência direta" – e a maneira como Carnap propõe estender o método sintático-formal para abarcar regras indefinidas de transformação – podemos agora verificar como se define outro conceito fundamental para o projeto de *SLL*: o conceito de "analiticidade" [60].

Desde já, porém, devemos fazer uma advertência mais do que importante. No presente capítulo, estaremos preocupados em definir exclusivamente *um* dos *dois* conceitos de analiticidade presentes no livro. Mais precisamente, estaremos preocupados em definir o conceito de analiticidade formulado no âmbito da sintaxe geral, ou seja, na Parte IV de *SLL*. Existe, porém, um outro conceito de analiticidade, desenvolvido por Carnap na parte III do livro, especificamente para a sua linguagem II. Esses dois conceitos, como veremos no momento oportuno, são essencialmente distintos.

As razões por que Carnap introduz dois conceitos distintos de analiticidade não ficam muito claras em nenhum momento do texto, e constituem um dos pontos mais complicados para a apreciação crítica de *SLL*. A esse respeito, por exemplo, Coffa observa:

[60] No restante deste capítulo, como de resto em toda a exposição mais à frente, teremos de falar constantemente a respeito dos diferentes conceitos sintáticos desenvolvidos por Carnap, tais como o conceito de "validade", de "analiticidade" etc. Ao nos referir a esses conceitos, às vezes iremos colocá-los entre aspas (como acabamos de fazer), às vezes não. É importante frisar que essa variação não tem, na maioria das situações, um significado técnico: ela é usada principalmente para facilitar a leitura do texto.
Há situações, certamente, em que o uso de aspas é necessário, e possui um significado preciso. Quando escrevemos, por exemplo, algo como "o termo 'analiticidade'...", as aspas envolvendo a palavra "analiticidade" têm uma função específica. Isso acontece sempre depois de palavras como "termo" ("o termo 'analiticidade'..."), "palavra" ("a palavra 'analiticidade'..."), "expressão" ("a expressão 'analiticidade'...") e outras que, como elas, designam exclusivamente a seqüência simbólica em questão.
Ao tratar de conceitos, porém, a situação é bem mais ambígua. É possível falar em "o conceito de 'analiticidade'" (com as aspas) ou também em "o conceito de analiticidade" (sem as aspas). No primeiro caso, acentua-se a idéia de que estamos nos referindo àquele conceito que é especificamente designado pela *palavra* "analiticidade", a qual precisa ser destacada do texto; no segundo caso, não há uma ênfase em sua designação pela palavra, mas apenas uma referência ao conceito. O mais correto, talvez, fosse nunca usar as aspas (procedimento que, não por acaso, adotamos no título deste capítulo), pelo motivo de que um conceito não é uma palavra, ou a seqüência simbólica que o designa. A utilização de aspas, no entanto, ajuda sobremaneira a clareza do texto, em não poucas ocasiões, e por isso não nos pouparemos desse recurso.

"The second *strategy for defining truth and consequence in SLS appears in section 34, when analyticity is defined for language II. (...).* *Carnap never explained the reason for this change of strategy, (...)."* [61] (destaques do autor; grifo meu)

Por enquanto, nós ainda não buscaremos investigar as razões para essa duplicidade de estratégias. Embora figure entre os objetivos deste trabalho oferecer algumas considerações a respeito do assunto, isso só poderá ser feito, de maneira adequada, depois de havermos exposto e analisado, com suficiente grau de detalhes, o primeiro conceito de analiticidade – tarefa a que nos dedicaremos neste capítulo. Duas observações de caráter mais geral, no entanto, devem ser adiantadas.

A primeira encontra-se em conexão direta com a citação de Coffa que transcrevemos logo acima. Naquela passagem, o eminente estudioso do positivismo lógico (e particularmente da obra de Carnap) fala em uma "segunda estratégia para definir verdade e conseqüência", e refere-se com isso às duas definições de analiticidade presentes em *SLL*. Essa formulação deixa a entender que ambos os conceitos de analiticidade formulados na obra buscam captar, de alguma forma, a idéia de "verdade", qualquer que ela pudesse ser. Isso, absolutamente, não é o caso.

Em relação ao conceito de analiticidade desenvolvido para a linguagem II – ao qual nos referiremos, sempre, como o "segundo" conceito de analiticidade do livro[62] – a situação é de fato bastante complexa. Embora o assunto necessite ser examinado com detenção (coisa que faremos no Capítulo 4 e 5), adiantamos que não é de todo incorreto tratá-lo da maneira como Coffa sugere: Essa segunda definição parece mesmo incorporar uma

---

[61] [Coffa, 1987]: pág. 550.

[62] Rigorosamente, de acordo com a seqüência de *SLL*, o conceito de "analiticidade" desenvolvido dentro da sintaxe geral (que estamos chamando de "primeiro") aparece *depois* do conceito de "analiticidade" desenvolvido para a linguagem II (que estamos chamando de "segundo"). A opção por essa inversão obedece a um critério de relevância dentro da obra, e busca traduzir alguns aspectos importantes da estrutura de *SLL*. (Conforme ficará claro mais à frente, o "primeiro" conceito é aquele que, do ponto de vista do projeto filosófico do livro, desempenha o papel mais relevante, situação às vezes encoberta pelo fato de o segundo conceito ser, do ponto de vista teórico, o mais interessante dos dois.)
Coffa, por um lado, parece adotar esse mesmo ponto de vista, ao compartilhar conosco a noção de que a definição de analiticidade da sintaxe geral deve ser tratada como a "primeira" das duas. Por outro lado, o fato justamente de considerar a segunda definição mais interessante do que a primeira (ele afirma isso explicitamente na continuação do trecho citado) faz com use, para descrever a situação geral da obra, essa formulação – "*the second strategy for defining truth and consequence...*" – que só se aplica ao segundo conceito.

estratégia "para definir verdade e conseqüência". Inversamente, porém, o primeiro conceito de analiticidade, formulado no âmbito da sintaxe geral, está no cerne de uma atitude teórica radicalmente diversa por parte de Carnap, e traduz justamente a vontade do autor de superar qualquer referência a uma esfera de proposições ou fatos "verdadeiros". Carnap é explícito a esse respeito na parte IV:

> *"*<u>For truth and falsehood are not proper syntactical properties</u>*; whether a sentence is true or false cannot generally be seen by its design, that is to say, by the kinds and serial order of its symbols."* [63] (destaque do autor; grifo meu)

Se lembrarmos que, para Carnap, a filosofia deve ser substituída pela análise exclusivamente sintática de linguagens formais, e se lembrarmos a ênfase com que ele deseja expurgar qualquer elemento não-formal das investigações lógicas – fato esse que já demonstramos suficientemente nos dois primeiros capítulos deste trabalho –, então a citação acima deixa claro que o conceito de analiticidade, estritamente formal e sintático como é, não pode constituir uma estratégia para "definir verdade e conseqüência". A negação de toda essa maneira de colocar o problema, em termos de "verdade" e outras noções correlatas, é uma das características mais conspícuas da sintaxe geral, e fica reforçada pela postura de absoluta liberdade formal que Carnap adota em relação ao conceito de "conseqüência direta".

Em segundo lugar, chamamos desde já a atenção para o fato de que, entre os dois conceitos de analiticidade formulados por Carnap, o da sintaxe geral é, *do ponto de vista do projeto filosófico de SLL*, o mais importante. Conforme já observamos no capítulo precedente, o estudo da sintaxe geral desempenha, no esquema teórico proposto por Carnap, um papel absolutamente central. É dentro da sintaxe geral que deveriam ser desenvolvidos aqueles conceitos sintáticos fundamentais, por meio dos quais seria possível analisar de maneira adequada a estrutura das diferentes linguagens formais (única tarefa capaz de tornar o estudo da lógica e da filosofia – reduzida agora à lógica – não apenas útil, como também necessário ao progresso científico).

---

[63] *SLL*: pág. 216.

Por esse motivo, torna-se realmente difícil compreender as exatas razões que levaram Carnap a desenvolver, para uma linguagem específica, um conceito de analiticidade completamente diferente daquele oferecido, com tanto esforço e elaboração, dentro da sintaxe geral. É a esse estado de coisas que Coffa se refere na passagem citada acima. E é esse estado de coisas que, no momento oportuno, tentaremos esclarecer, pois constitui uma das chaves para estabelecer os méritos e os limites teóricos da obra de Carnap. Por enquanto, precisamos dissecar, com o maior rigor possível, o primeiro conceito de analiticidade. Passemos a essa tarefa.

II

Para entender o conceito de analiticidade formulado na sintaxe geral, devemos começar – mais uma vez – por uma comparação com os sistemas tradicionais de lógica simbólica. A maioria dos conceitos formulados por Carnap em *SLL* apresenta-se como uma extensão de conceitos familiares, comuns às diferentes abordagens formais. Seu significado e suas características, certamente, são transformados, para desempenhar certas funções específicas essenciais ao projeto filosófico defendido na obra. Além disso, a complexidade técnica necessária ao seu desenvolvimento é bem maior do que no caso dos sistemas tradicionais, devido à incorporação de regras indefinidas de dedução. Mesmo assim, o paralelo com conceitos mais familiares – até mesmo para indicar o momento em que cessa a coincidência – fornece a melhor maneira de compreender os mecanismos descritos por Carnap (o qual opta freqüentemente, ele mesmo, por esse tipo de exposição comparativa).

De maneira resumida, podemos destacar o seguinte quadro conceitual geral, comum aos diferentes sistemas tradicionais de lógica simbólica. Algumas sentenças do cálculo são oferecidas como "sentenças iniciais". A partir delas, pode-se aplicar as regras de dedução próprias ao cálculo. As regras são aplicadas um número indefinido de vezes, e em diferentes combinações, para permitir a obtenção de outras sentenças. Todas as sentenças resultantes desse processo – tanto as ditas "sentenças iniciais" como as outras, obtidas a partir delas por meio das regras de inferência – compõem o corpo de **teoremas** do cálculo.

Os teoremas constituem, por assim dizer, o corpo de proposições afirmadas pelo cálculo, ou aceitas como "verdadeiras" pelo cálculo[64].

Uma cadeia de sucessivas aplicações das regras de transformação do sistema, capaz de levar – passando por quantas sentenças intermediárias se mostrem necessárias – das sentenças iniciais até uma outra sentença do cálculo (a qual se afirma, por esse mesmo processo, como teorema do cálculo), é chamada de **demonstração** dessa sentença. Uma sentença é dita **demonstrável** (ou seja, é um teorema) se existir para ela uma demonstração. Temos aí o conceito de **demonstrabilidade**.

É possível, por outro lado, aplicar as regras de dedução não somente a partir das sentenças iniciais, mas a partir de um conjunto qualquer de sentenças tomadas como premissas arbitrárias de um raciocínio específico. Nesse caso, as novas sentenças obtidas não são, necessariamente, teoremas; mas são vistas como conseqüência lógica das premissas adotadas.

Assim, uma cadeia de sucessivas aplicações das regras de transformação do sistema, a partir de um conjunto qualquer K de premissas (e também das sentenças iniciais), até uma sentença C, é chamada de **derivação** da sentença C (a partir das premissas K). Uma sentença é dita **derivável** a partir de certas premissas se, a partir dessas premissas (e das sentenças iniciais), existir uma derivação levando até ela. Surge assim o conceito de **derivabilidade**, extremamente importante. Uma demonstração, como podemos ver, é apenas uma derivação que não faz uso de nenhum conjunto especial de premissas, apenas das sentenças iniciais. Um teorema, analogamente, é uma sentença derivável (demonstrável) a partir exclusivamente das sentenças iniciais, sem necessidade de acrescentar nenhuma outra premissa específica.

Esse é o conjunto básico de conceitos que, extraídos do estudo tradicional de sistemas formais, irá nos interessar. Carnap, de sua parte, aceita essas construções como adequadas para descrever a estrutura das linguagens formais tradicionais, às quais se aplicam[65]. Esse fato ficará bastante claro quando tivermos verificado a maneira como ele as

[64] Carnap, obviamente, irá recusar a utilização do adjetivo "verdadeiro" nesse caso. Nos sistemas tradicionais, no entanto, os teoremas eram sim vistos como o corpo das "verdades" expressas e capturadas pelo cálculo.
[65] Pode-se reparar que a elaboração dos conceitos indicados nesta seção atende, realmente, a todos os requisitos impostos por Carnap como próprios ao estudo sintático (ver Capítulo 2, seção I). Em primeiro

reformula dentro do esquema de *SLL*; pois os conceitos sintáticos que iremos examinar neste capítulo surgem como extensão bastante fiel desse quadro conceitual, adaptado à nova visão do método formal.

III

Tal como acabamos de oferecê-la, na versão tradicional, a caracterização formal dos conceitos de "derivabilidade", "demonstrabilidade" e "teorema" é bastante simples, e não envolve, do ponto de vista técnico, nenhuma dificuldade mais aguda. Seu problema, para o esquema de *SLL*, irá residir na limitação de sua aplicabilidade, restrita que fica a sistemas formais dotados exclusivamente de regras definidas de transformação. Carnap precisa, portanto, estender a definição desses conceitos para permitir o tratamento das novas estruturas, cujo estudo sistemático ele havia sido o primeiro a sugerir. Infelizmente, uma vez admitidas regras indefinidas de transformação, a situação técnica complica-se razoavelmente. Antes de verificar exatamente como Carnap resolve essas questões, é importante fixar alguns pontos.

O primeiro é um fato puramente terminológico. As estruturas tradicionais, limitadas a regras definidas de inferência, são apenas uma instância particular do caso mais geral, vislumbrado por Carnap. Seu estudo, e mesmo sua utilização, não ficam afastados. Pelo contrário, elas continuam a constituir um tipo bastante importante de sistemas formais, além de fornecer um excelente modelo para comparações e analogias. Carnap chama o método formal, quando restringido a regras definidas de inferência, de **d-método**. O prefixo "d-", aqui, é uma referência ao conceito de "derivação" que, conforme assinalamos na seção anterior, traduz o processo dedutivo fundamental desses sistemas. No mesmo sentido, ele irá prefixar com a letra "d-" todos os conceitos que dizem respeito, exclusivamente, a sistemas dotados apenas de regras definidas.

lugar, os conceitos estão descritos de maneira puramente formal (fazem referência somente às regras formais de que se compõem as diferentes linguagens). Em segundo lugar, podem ser aplicados a todas as linguagens do tipo relevante, ou seja, do tipo formal tradicional, com regras definidas de transformação (em relação ao qual foram concebidos). Finalmente, esses conceitos serão vistos – em sentido que nos caberá ainda explicar – como os conceitos mais importantes das linguagens às quais se aplicam.

Por oposição, Carnap chama o método formal estendido, com regras indefinidas de transformação, de **c-método**. O prefixo "c-", nesse caso, faz referência ao conceito de "conseqüência direta", o qual incorpora, como também já vimos, o novo tipo de regra proposto em *SLL*. Todos os conceitos prefixados por um "c-" dizem respeito, assim, a sistemas nos quais são permitidas regras indefinidas.

Fixada essa terminologia, o segundo ponto a respeito do qual devemos observar diz respeito justamente à distinção entre d-métodos e c-métodos. Ou melhor: diz respeito a uma semelhança entre d-métodos e c-métodos, que se mostram, no sentido em que veremos, menos afastados do que se poderia em princípio imaginar. Acabamos de dizer que os c-métodos dizem respeito ao caso indefinido; e os d-métodos, ao caso definido. A verdade, porém, é que ambos os métodos revelam-se *indefinidos*, sob certo aspecto bastante importante, que precisamos desde agora manter em vista.

A questão a esclarecer, obviamente, é: Em que sentido os d-métodos são indefinidos, se utilizam somente regras definidas de transformação? À primeira vista, tudo parece realmente depender somente da natureza das regras de transformação, na medida em que são elas que definem uma linguagem (que constituem a linguagem). No caso dos d-métodos, as regras de inferência são definidas (baseadas em conjuntos-base finitos de premissas), e essa restrição deveria bastar para caracterizar sua estrutura dedutiva.

No entanto, pergunta-se: O que uma regra de inferência permite fazer? Permite passar de um conjunto de sentenças, *diretamente*, a uma outra sentença. Mas nós já vimos que, para obter os teoremas de um cálculo, a partir de suas sentenças iniciais, é possível aplicar as regras de transformação um número indefinido de vezes. O conjunto dos teoremas de um cálculo não se resume àquelas sentenças que podem ser obtidas, diretamente das sentenças iniciais, pela aplicação, uma única vez, de uma das regras de transformação disponíveis. A mesma coisa vale em relação ao processo de derivação de uma sentença a partir de um conjunto de premissas: o conjunto de sentenças deriváveis a partir das premissas não se limita àquelas sentenças que podem ser obtidas pela aplicação de uma das regras de dedução, diretamente, uma única vez.

Por isso mesmo, precisamos introduzir (seguindo Carnap), um novo termo. Uma sentença que se segue de certas outras sentenças pela aplicação, *uma única vez*, de uma das regras definidas de transformação, diz-se **diretamente derivável** dessas sentenças. O que dissemos no parágrafo anterior, portanto, resume-se ao seguinte: o conceito de "derivabilidade direta" não exaure o conceito de "derivabilidade". De posse dessa distinção, então, fica fácil entender o que se passa. No caso dos d-métodos, o que é definido é especificamente o conceito de derivabilidade *direta* (o qual reflete, simplesmente, a natureza das regras de transformação). No entanto, devemos indagar: E quanto ao conceito fundamental de derivabilidade, que traduz de maneira mais ampla os processos dedutivos da linguagem?

Como vimos, uma sentença é dita "derivável" a partir de certas outras (ou "demonstrável", no caso de não haver premissas específicas além das sentenças iniciais) se houver uma *cadeia* de passos dedutivos capaz de conduzir do conjunto de premissas à sentença em questão. Essa cadeia de derivação pode envolver a aplicação das diferentes regras de inferência disponíveis na linguagem, qualquer número de vezes que se deseje. Não há um limite máximo para o tamanho das derivações. A conseqüência dessa situação é que o conceito de derivabilidade será, contrariamente ao conceito de derivabilidade direta, *indefinido*.

No capítulo anterior, assinalamos que, em muitos casos, pode tornar-se extremamente difícil saber se determinada sentença é ou não teorema de uma linguagem formal. Fornecemos, a título de exemplo, as sentenças que afirmam o teorema de Fermat e a conjectura de Goldbach. Vemos agora o que está por trás da dificuldade de saber se sentenças como essas são ou não teoremas: O conceito de derivabilidade e, pelos mesmos motivos, o de demonstrabilidade (ou seja, o próprio conceito de teorema), são indefinidos. Não se pode saber sempre, por meio de um procedimento regular que chegue ao fim em um número finito de passos, se uma sentença é ou não teorema de um cálculo, ou se é ou não derivável a partir de certas outras sentenças[66]. As demonstrações e derivações não possuem um limite superior de passos, previamente estabelecido. Obviamente, elas não podem ser

[66] O sentido da oposição "definido" x "indefinido", aqui, assume contornos um pouco distintos daqueles que vínhamos usando para classificar as regras de transformação. Lá, havíamos encontrado para essa oposição uma formulação bastante objetiva, em termos a cardinalidade do conjunto-base das deduções; aqui, vemos novamente aparecer uma referência à finitude ou à infinitude, porém não de conjuntos de sentenças, mas de certos "procedimentos" dedutivos.
De fato, a utilização dessa oposição (definido x indefinido), em *SLL*, assume mais de um significado ao longo do texto. A nota logo abaixo, em que comentamos uma passagem do livro, foi motivada justamente por essa situação.

infinitas; qualquer derivação ou demonstração corresponde à aplicação das regras de transformação (no caso, regras definidas de transformação) apenas um número finito de vezes. Contudo, não havendo limite superior ao número de passos, elas podem ter *qualquer* tamanho finito que se deseje ou necessite. Eis como o próprio Carnap expõe o assunto, na mesma passagem em que esclarece as questões terminológicas já referidas:

> *"We may set up a criterion of validity which, although itself indefinite, is yet based upon definite rules. Of this kind is the method that is used in all modern systems which attempt to create a logical foundation for mathematics (...). We shall designate it as the* method of derivation *or the* d-method*. It consists of setting up primitive sentences and rules of inference (...). In the rules of inference only a finite number of premisses (usually only one or two) appear. (...) It is usual to construct the rules in such a way that the term 'directly derivable' is always a definite term; that is to say, that in every individual case it can be decided*[*] *whether or not we have an instance of a primitive sentence – or of the application of a rule of inference, respectively. We have already seen how the terms 'derivable', 'demonstrable', 'refutable', 'resoluble', and 'irresoluble' are defined on*

[*] Nesta passagem, que é extraída da parte III de *SLL*, vemos Carnap associar o caráter definido das regras de dedução (ou seja, do conceito de derivabilidade direta) à possibilidade de decisão, em qualquer caso concreto, de sua aplicação. No capítulo anterior, vimos que essa maneira de caracterizá-las, por oposição às regras indefinidas, não é totalmente exata. Também as regras indefinidas são claras quanto à possibilidade de decisão. O problema, em um ou outro caso, diz respeito à possibilidade de se conhecer o status de todas as sentenças do conjunto-base a partir do qual será aplicada a regra, o que pode ser fácil ou difícil, tanto no caso definido como no caso indefinido. O melhor mesmo é caracterizar a distinção entre os dois tipos de regra de maneira direta, com referência ao número – finito ou infinito – de sentenças do conjunto-base.
Nesse ponto, é necessário reconhecer que o próprio Carnap oscila na interpretação dessa situação. Isso acontece devido a razões que ficarão claras somente o Capítulo 5, quando examinaremos o segundo conceito de analiticidade e os motivos que determinam sua adoção. Ali, teremos de introduzir uma questão delicada, relativa à necessidade e à possibilidade, quando da adoção de regras indefinidas de transformação, de descrever de maneira formalmente correta o conjunto-base da transformação. Vimos, no capítulo precedente, que esse conjunto tem de ser bem descrito, para que o método formal faça sentido. Essa interpretação, mais ainda, é a única condizente com o esquema da sintaxe geral, e parece ficar clara pela leitura da parte IV de *SLL*. Não obstante, ela será colocada em cheque no desenvolvimento concreto do conceito de analiticidade para a linguagem II.
O tema, portanto, faz parte da importante série de cisões interpretativas que podem ser detectadas em relação a esses dois importantes momentos de *SLL*: o esquema da sintaxe geral e a "segunda estratégia" para definir o conceito de analiticidade; cisão que, segundo Coffa (ver seção I acima), o próprio livro não esclarece completamente. Por enquanto, insistimos no fato de que o esquema da sintaxe geral é aquele mais importante para o projeto filosófico de *SLL*. Por isso mesmo, salvo aviso em contrário, nosso ponto de vista será sempre o da sintaxe geral.

> *the basis of this d-method. Since no upper limit to the length of a derivation-chain is determined, the terms mentioned, although they are based upon the definite term 'directly derivable', are themselves indefinite."* [67] (destaques do autor)

Vemos aqui Carnap insistir na distinção que há entre o conceito de derivabilidade direta (o qual, no caso de d-métodos, é um conceito definido) e o conceito de derivabilidade (que é indefinido *mesmo* para o caso de d-métodos). Essa distinção, vale a pena observar, corresponde a uma característica bastante profunda das abordagens formais para a lógica. No que diz respeito à estrutura dedutiva das linguagens, o que os diferentes sistemas de lógica formal formalizam, rigorosamente falando, é apenas um tipo restrito de dedução, ou melhor, apenas uma parcela restrita do processo dedutivo: a dedução direta. Eles formalizam apenas os pequenos "passos" lógicos de que se compõe um raciocínio mais complexo, como uma demonstração ou derivação. Aí reside o próprio sabor, se assim podemos dizer, do método formal: Estabelecer regras formais claras e inequívocas para as "pequenas" passagens lógicas que conduzem, de maneira imediata, de uma a outra sentença, esperando que esse conjunto de pequenos "passos" possam ser suficientes para captar todos os raciocínios mais complexos (as demonstrações e derivações, as quais também chamaremos, de maneira geral, de "provas").

O conceito mais amplo de prova, por outro lado, não chega a ser diretamente formalizado. A prova surge como articulação possivelmente complexa desses pequenos passos lógicos, e tudo o que o método formal chega a dizer a seu respeito limita-se precisamente a isso: que uma prova deve ser composta por um número finito (nos sistemas tradicionais) desses pequenos passos formais. Contudo, o método formal não se propõe a descrever o caminho para se obter uma prova (ou para mostrar que ela não pode ser obtida), vale dizer, o caminho para se atingir um teorema ou uma derivação (ou para mostrar que eles não podem ser atingidos). Esse caminho permanece sempre um caminho complexo, formado por qualquer combinação finita de passos formais individuais – e daí Carnap falar que o conceito de derivabilidade é indefinido, mesmo sendo baseado em um conceito definido de derivabilidade direta.

[67] *SLL*, págs. 99 e 100.

IV

Carnap substitui o conceito de "derivabilidade direta" (dito "definido", ou seja, restrito a conjuntos-base finitos) pelo conceito mais amplo de "conseqüência direta" (dito "indefinido", com a admissão de conjuntos-base infinitos). Vimos, na seção anterior, que o conceito de "derivabilidade direta", em um sistema formal, representa apenas uma parte da idéia de prova. Ele representa o conjunto das operações atômicas (regras de inferência) a partir do qual as provas devem ser compostas. Já vimos, também, a maneira pela qual devem ser obtidos (provados) todos os teoremas do cálculo, e pela qual todas as derivações devem ser feitas: pela aplicação sucessiva das regras de inferência, ou seja, por meio do encadeamento desses passos lógicos atômicos codificados no conceito de "derivabilidade direta".

O mesmo vale, agora, para os c-métodos de Carnap, e para o seu conceito de "conseqüência direta". Este conceito funciona como elemento atômico do processo dedutivo (de maneira paralela, portanto, à do conceito de "derivabilidade direta"). Ele indica as transformações que podem ser feitas *diretamente* a partir de certo conjunto (possivelmente infinito) de premissas. Contudo, a idéia genérica de prova, aqui como antes, vai além da aplicação direta, uma única vez, de uma regra de transformação. As regras de transformação podem ser aplicadas diversas vezes, de maneira coordenada, para obter provas. Chegamos assim ao importante conceito de **conseqüência** (paralelo ao conceito de "derivabilidade").

Uma sentença C é conseqüência de certo conjunto K de sentenças se existir um caminho formal levando de K até C. O caminho formal, obviamente, deve ser composto pela aplicação coordenada das regras formais de transformação (codificadas, em seu conjunto, no conceito de "conseqüência direta"). A idéia, portanto, é essencialmente a mesma que expusemos para o caso dos d-métodos. No entanto, a admissão de regras indefinidas de dedução torna a situação razoavelmente mais complexa do ponto de vista técnico.

No caso dos d-métodos, podíamos falar na aplicação *sucessiva* das regras de inferência, um número finito de vezes, para constituir cadeias de derivação, ou provas. No caso dos c-métodos, tal abordagem revela-se insuficiente. Por esse motivo, temos falado na aplicação *coordenada* (em vez de *sucessiva*) das regras indefinidas, como forma de descrever a nova situação. Essa aplicação coordenada (e já esclareceremos, do ponto de vista técnico, a forma que essa "coordenação" assume) das regras de inferência permite construir aquilo que Carnap chama, em alguns momentos, de "*consequence-series*" (o correspondente de "derivação" ou "prova"). O foco da complicação reside, como não poderia deixar de ser, na possibilidade de utilizar, como conjunto-base de uma transformação, um conjunto contendo infinitas sentenças.

Para ilustrar as dificuldades que irão surgir, suponhamos que, a partir de um conjunto infinito K_2 de sentenças, seja possível obter, pela aplicação de uma regra de transformação, a sentença C. Suponhamos agora que, a partir de um outro conjunto K_1 de sentenças, possivelmente também infinito, seja possível obter, pela aplicação das regras de transformação, todas as sentenças do conjunto K_2, porém não, *diretamente*, a própria sentença C. Pergunta-se: Qual a relação entre a sentença C e o conjunto K_1?

Não é difícil ver que a sentença C deveria ser vista como conseqüência do conjunto K_1, na medida em que todas as sentenças de K_2 são conseqüência de K_1, e C é conseqüência de K_2. Em outras palavras, parece que um conceito adequado de "prova" ou "conseqüência", para o sistema formal em questão, deveria levar em conta a possibilidade de se alcançar – a partir do conjunto K_1, e por meio exclusivamente das regras de inferência do sistema – a sentença C. Afinal de contas, há um caminho formal conduzindo de K_1 até C (o qual passa pelo conjunto K_2), caminho este que deveria ficar abrangido pelo conceito formal de "prova" (ou "conseqüência"). Infelizmente, um tratamento mais simples para essa situação, como aquele utilizado para os sistemas tradicionais, mostra-se incapaz de lidar com a situação descrita. Nenhuma cadeia finita de aplicações sucessivas das regras de transformação permitirá alcançar o resultado desejado, pelo fato de que o conjunto intermediário K_2 possui infinitas sentenças.

A discussão dos dois parágrafos anteriores, conforme observamos, deve ser vista apenas como motivação para a formulação mais precisa do caso geral, cuja complexidade poderia parecer gratuita. Seu objetivo esteve em revelar o tipo de problema que irá surgir com a maneira usual de encarar a idéia de prova (como aplicação sucessiva das regras de transformação), uma vez aceitas regras do tipo indefinido. A idéia de que uma sentença é conseqüência, segundo as regras formais do sistema, de certas outras sentenças, precisa portanto ser reformulada. Eis como Carnap resolve a questão, indicando a construção rigorosa de um conceito de "conseqüência" para as novas estruturas formais:

> *"In what follows the K are always sentential classes. C_1* [uma sentença] *is called a* consequence *of K_1, if C_1 belongs to every sentential class K_i satisfying the following two conditions: 1. K_1 is a sub-class of K_i; 2. Every sentence which is a direct consequence of a sub-class of K_i belongs to K_i."* [68] (destaque do autor; colchete meu)

Um pouco de reflexão poderá mostrar que esta definição consegue realizar tudo o que dela se espera, e que desempenha o mesmo papel que a definição de derivabilidade desempenha para sistemas dotados exclusivamente de regras definidas de inferência. Não se trata, porém, de reflexão simples: o próprio Carnap demorou para ver corretamente o problema, e parece ter necessitado da colaboração de Tarski para obter uma formulação adequada[69]. Por enquanto, o importante é manter em mente – não custa repetir – que o papel dessa definição é análogo ao papel da definição de derivabilidade, como deixa claro o seguinte comentário de Carnap, inserido logo após o trecho citado acima:

> *"If d-rules* [regras definidas] *only are given, then the terms 'derivable' and 'consequence' coincide; (...)"* [70] (colchete meu)

V

Temos em mãos, portanto, devidamente tratado do ponto de vista formal, o conceito de "conseqüência". A partir dele, é razoavelmente fácil extrair todos os demais conceitos importantes de uma linguagem, culminando com o conceito de "analiticidade".

---

[68] *SLL*, pág. 172.

[69] Logo após a passagem citada anteriormente citada, Carnap insere a seguinte nota:

> *"(Note, 1935.) The above definition of 'consequence' is a correction of the German original, the need for which was pointed out to me by Dr. Tarski." – SLL*, pág. 172.

[70] *SLL*, pág. 172.

Quais seriam, nos novos sistemas de Carnap, os "teoremas" da linguagem? Já sabemos que os teoremas são as sentenças afirmadas pela linguagem, ou seja, são as sentenças que resultam de suas regras formais. No caso dos d-métodos, essas sentenças são as sentenças demonstráveis, vale dizer, as sentenças deriváveis a partir exclusivamente das sentenças iniciais. No caso dos c-métodos, o mesmo irá acontecer: As sentenças afirmadas pelo sistema são aquelas que são "conseqüência" (em vez de "deriváveis") das sentenças iniciais. Porém, devemos recordar que as próprias sentenças iniciais são vistas como um caso especial das regras de transformação; elas são "conseqüência direta" do conjunto vazio. Assim, o corpo de teoremas do cálculo, de acordo com o esquema geral formulado por Carnap, compõe-se de todas as conseqüências (no sentido técnico explicado na seção anterior) do conjunto vazio.

Podemos fixar a situação da seguinte maneira. As sentenças iniciais são as *conseqüências diretas* do conjunto vazio. Mas já sabemos que o conceito de conseqüência direta (passos atômicos do processo dedutivo) não exaure o conceito mais amplo de conseqüência (que traduz os processos dedutivos como um todo). Os teoremas, então, são as *conseqüências* do conjunto vazio. Essa formulação, aliás, ressalta seu caráter de sentenças afirmadas pelo sistema: ela mostra como os teoremas já estão implicados pela simples formulação das regras formais do sistema, na medida em que são conseqüência, de acordo com essas regras, do conjunto *vazio* de sentenças. Carnap irá chamar os teoremas de sentenças **válidas**. Eis o que ele escreve:

> *"K_1* [uma classe de sentenças] *is called* valid *if K_1 is a class of consequences of de null class (and hence of every class). [We do not use the term 'analytic' here because we wish to leave open the possibility that S* [uma linguagem] *contains not only logical rules of transformation (as do Languages I and II) but also physical rules such as natural laws (see §51). In relation to languages like I and II, the terms 'valid' and 'analytic' coincide]."* [71] (destaque do autor; colchetes, em português, meus)

[71] *SLL*, págs. 173 e 174.

O trecho acima nos mostra que o conceito de “validade” (ou de sentença “válida”) construído por Carnap já está muito próximo do conceito de “analiticidade” (ou de sentença “analítica”). Examinemos um pouco seus desdobramentos.

Uma sentença válida, como vimos, é uma sentença que resulta das regras formais do sistema. E se acontecer, porém, que a negação de uma sentença (e não ela própria) resulte das regras do sistema? Nesse caso, chamamos a sentença de **contraválida**. Uma sentença contraválida, nesse sentido, é uma sentença negada pelo sistema. Em um sistema consistente, pela própria definição usual de consistência, nunca é possível que duas sentenças C e não-C – das quais uma é a negação da outra – sejam, ao mesmo tempo, válidas. Se isso acontecer, então é possível demonstrar que todas as sentenças do sistema são ao mesmo tempo válidas e contraválidas, ou seja, todas as sentenças são ao mesmo tempo afirmadas e negadas pelas regras formais, situação essa bastante desinteressante tanto do ponto de vista teórico como do ponto de vista prático, de utilização da linguagem. Assim, temos dois casos excludentes entre si: Em um sistema consistente, ou pode acontecer que uma sentença C resulte das regras do sistema, ou pode acontecer que sua negação não-C resulte das regras do sistema; ou uma sentença C é válida, ou é contraválida; nunca as duas coisas ao mesmo tempo[72].

Questão ainda mais importante, porém, é aquela que iremos propor a seguir. Uma sentença, já sabemos, não pode ser válida e contraválida ao mesmo tempo. Mas será necessário que ela seja uma entre essas duas coisas? Em outras palavras: Temos alguma razão para supor que, dada uma sentença C, ou ela ou sua negação – uma entre as duas – deva necessariamente resultar das regras do sistema? Ou pode acontecer que nem ela, nem sua negação, resultem das regras do sistema? Nesse caso, o sistema seria incapaz de decidir quanto a uma ou outra; ele não afirmaria nem a sentença, nem sua negação; a respeito dessa sentença, o sistema não diria nada, nem que ela é válida, nem que é contraválida.

Em princípio, não temos razão para fazer a mencionada suposição. Dada uma sentença C, certamente não é necessário que ela resulte das regras do sistema; pelo mesmo

[72] Simplificamos um pouco a definição de sentença contraválida. Carnap define assim esse conceito: C é contraválida se todas as sentenças do cálculo forem conseqüência de C.
Essa formulação deve-se à sua vontade de fornecer, no estudo da sintaxe geral, o quadro mais amplo possível, capaz de tratar também linguagens desprovidas de “negação”. Em uma linguagem dotada de “negação”, no entanto (com as propriedades usuais), as duas definições são equivalentes.

raciocínio, também não é necessário que sua negação, que é simplesmente uma outra sentença, resulte das regras do sistema. Obviamente, pode acontecer que um sistema S tenha a seguinte característica estrutural: dado qualquer par de sentenças opostas C e não-C, exatamente uma das duas é conseqüência do conjunto vazio (ou seja, é afirmada pelo sistema). Porém, uma tal característica, embora perfeitamente possível, estaria longe de ser trivial. A existência de um tal conjunto de pares de sentenças revelaria uma característica estrutural profunda, a qual necessitaria ser demonstrada e, uma vez demonstrada, representaria um aspecto bastante notável do sistema em questão, que podemos denominar de "completude"[73]. Certamente, porém, nem todas as estruturas formais, tal como Carnap as caracteriza, são desse tipo.

No caso geral, de fato, pode acontecer que, para certa sentença C, nem ela nem sua negação sejam afirmadas; a sentença C não é nem válida, nem contraválida. Nesse caso, dizemos que tal sentença é **indeterminada**. Uma sentença indeterminada, nesse sentido, é uma sentença a respeito da qual o sistema mostra-se incapaz de decidir. Um exemplo trivial é fornecido por sentenças como "Luana é alta", a qual depende de uma observação externa ao sistema para ser afirmada ou negada (por outro lado, se essa observação for incorporada às regras do sistema, por exemplo como sentença inicial, então ela deixa de ser indeterminada, e passa a ser válida; falaremos mais a respeito desse tipo de sentença na próxima seção).

Finalmente, uma sentença é dita **determinada** se for válida ou se for contraválida, isto é, se não for indeterminada. As sentenças determinadas, portanto, são aquelas a respeito das quais o sistema é capaz de decidir, quer afirmando-as (se forem válidas), quer negando-as (se forem contraválidas).

Vale a pena observar ainda que essa classificação das sentenças de uma linguagem é exaustiva, pela própria definição de "sentença indeterminada", a qual engloba *todas* as sentenças que não são nem válidas, nem contraválidas. Assim, ou uma sentença é válida, ou é contraválida, ou é indeterminada, e é necessariamente uma entre essas três coisas.

[73] A questão da completude pode ser colocada, não para *todos* os pares de sentença C e não-C do sistema, mas apenas para uma classe (fechada sob a operação de negação) de sentenças, como as sentenças matemáticas ou as sentenças lógicas de uma linguagem. Os casos obviamente se confundem se a linguagem só possuir sentenças lógicas, ou se todas as sentenças matemáticas forem vistas como lógicas.

(Podemos também dizer: ou uma sentença é determinada, ou é indeterminada, e é necessariamente uma entre essas duas coisas.)

VI

Um sistema formal pode ter regras de transformação de diversos tipos; de fato, qualquer regra capaz de ser formalmente expressa, da maneira como mostramos no Capítulo 2, é admissível. Haveria, então, alguma razão para distinguir entre diferentes categorias de regras sintáticas?

Carnap acredita que sim; e cabe aqui abrir um parêntese. Um sistema formal, embora seja formal do ponto de vista de sua estrutura, é construído para ser utilizado como ferramenta do conhecimento científico. Portanto, por mais que o estudo sintático utilize métodos estritamente formalizados, o desenvolvimento de uma linguagem particular nunca é feito no vazio, como um mero jogo combinatorial. Pelo contrário: uma linguagem é desenvolvida sempre com vistas a um propósito, ou seja, com o objetivo de expressar as relações existentes em determinado domínio de objetos. Por isso mesmo, ela possui sempre aquilo que chamamos de uma "interpretação informal", que nos orienta em sua compreensão e análise. Eis como Carnap vê a situação:

> *"Our thesis that the logic of science is syntax must therefore not be misunderstood to mean that the task of the logic of science could be carried out independently of empirical science and without regard to its empirical results. The syntactical investigation of a system which is already given is indeed a purely mathematical task. But the language of science is not given to us in a syntactically established form; whoever wants to investigate it must accordingly take into consideration the language which is used in practice in the special sciences, and only lay down rules on the basis of this."* [74]

Carnap será fiel a esse princípio no desenvolvimento do quadro conceitual da sintaxe geral. Um quadro conceitual útil à descrição estrutural das linguagens formais

[74] *SLL*, pág. 332.

deveria ser capaz de captar certas distinções informais bastante relevantes para a atividade científica. E é precisamente daí que surgirá a distinção entre "validade" e "analiticidade". Em termos da interpretação informal de uma linguagem científica qualquer, parece sempre possível distinguir entre regras lógicas de transformação (normalmente comuns à maioria das linguagens, como a regra do *modus ponens* e outras) e regras de natureza não-lógica. Essas últimas – que Carnap irá chamar de regras "físicas" – são aquelas que incorporam certas leis empíricas próprias de domínios particulares e que, por isso mesmo, devem variar de linguagem para linguagem, dependendo do propósito com o qual cada uma é concebida. (Um exemplo – sempre informal – de regra de transformação "física", seria uma regra que permitisse a dedução, a partir da sentença "x é um elefante", da sentença "x pesa duas toneladas ou mais"; esse tipo de dedução, se estiver correto, é baseado em observações empíricas, e não parece basear-se em nada que estejamos dispostos a chamar de lógica.)

Do ponto de vista de *SLL*, porém, a questão que se coloca é: Seria possível capturar essa distinção, puramente informal, por meio de características sintáticas adequadas? Já sabemos que um conceito, para ser admissível dentro do esquema lógico-sintático proposto por Carnap, necessita estar formulado em termos puramente formais. A questão, portanto, pode ser vista assim: Existe um conceito formal capaz de se aproximar, segundo critérios bem estabelecidos, dessa separação intuitiva e informal entre os dois tipos de regras mencionadas, as regras lógicas e as regras físicas?

À primeira vista, pareceria que não. Qualquer regra de transformação do sistema tem de ser formulada de maneira sintática (tanto as regras supostamente "físicas" como as regras supostamente "lógicas"); e uma vez admitida dentro do sistema, qualquer regra passa a fazer parte do seu arsenal formal. Como, em consonância com o Princípio de Tolerância, qualquer sistema sintático bem formulado pode ser oferecido como um sistema lógico aceitável, então as regras supostamente "físicas", devidamente incorporadas ao sistema, deveriam ser tratadas também como regras lógicas, em pé de igualdade com qualquer outra regra. Em outras palavras, parece não haver mais nenhuma base para se traçar a distinção desejada entre um e outro tipo de regra.

A situação, porém, não é bem assim. Carnap acredita que há, sim, não apenas razões suficientes para traçar essa distinção, mas também meios estritamente formais adequados para o fazer. Precisamos ver agora, portanto, a construção que ele oferece. O primeiro passo nessa direção é traçar uma distinção, formalmente expressa, entre "expressões lógicas" e "expressões descritivas" (lembramos que o termo "expressão" refere-se a qualquer seqüência finita de símbolos da linguagem, aí incluídas expressões de um único símbolo). Essas últimas, informalmente falando, seriam aquelas expressões da linguagem que se referem a observações empíricas (expressões tais como o predicado "elefante", na sentença "x é um elefante"). Como diferenciar formalmente entre essas duas classes de termos, os lógicos e os descritivos? Começaremos com a exposição completa de Carnap, para depois esclarecê-la:

> *"Let K_1* [um conjunto, ou classe, de expressões] *be the product* [intersecção lógica] *of all expressional classes K_i of S* [uma linguagem], *which fulfill the following four conditions. [In the majority of the usual language-systems, there exists only one class of the kind K_i; this is then K_1.] 1. If A_1* [uma expressão] *belongs to K_i, then A_1 is not empty and there exists a sentence which can be sub-divided into partial expressions in such a way that all belong to K_i and one of them is A_1. 2. Every sentence which can be thus sub-divided into expressions of K_i is determinate*[*]*. 3. The expressions of K_i are as small as possible, that is to say, no expression belongs to K_i which can be sub-divided into several expressions of K_i. 4. K_i is as comprehensive as possible, that is to say, it is not a proper sub-class of a class which fulfils both (1) and (2). An* expression *is called* **logical** *(A_L) if it is capable of being sub-divided into expressions of K_1; otherwise it is called* **descriptive** *(A_d)."* [75] (destaques do autor; colchetes, em português, meus)

Qual o sentido dessa construção? Carnap parte da seguinte orientação: Nós estamos dispostos – de acordo com nossa visão informal das linguagens – a chamar algumas expressões de lógicas, outras de descritivas. Expressões como "não", ou como "ou", ou

[*] Lembramos que o conjunto das sentenças ditas "determinadas" é a união do conjunto das sentenças válidas com o conjunto das sentenças contraválidas, ou seja, abrange todas as sentenças a respeito das quais o sistema é capaz de decidir.
[75] *SLL*, págs. 177 e 178.

ainda como “todo”, parecem lógicas; já uma expressão como “elefante”, por outro lado, parece descritiva. Qual a diferença entre elas?

Até agora, dispomos de uma importante família de conceitos formais, capaz de dissecar toda a estrutura da linguagem em termos de suas sentenças: estamos nos referindo aos conceitos de “sentença válida”, “sentença contra-valida”, “sentença determinada” e “sentença indeterminada”. Essa família de conceitos, suficiente para classificar todas as sentenças de uma linguagem, foi obtida a partir do conceito de “conseqüência” (que é um conceito que relaciona sentenças em ter si), por sua vez obtido a partir do conceito de “conseqüência direta” (que, no que tange às regras de transformação, também é um conceito que relaciona sentenças entre si). O caminho aberto para buscar agora uma classificação das *expressões* da linguagem, portanto, deve passar pela utilização dessa classificação das *sentenças* da linguagem. Para Carnap, esse caminho revela-se perfeitamente natural.

De fato, Carnap percebe o seguinte fato, distintivo das expressões que, normalmente, estamos dispostos a chamar de “lógicas”: Quando combinadas exclusivamente entre si, essas expressões deveriam produzir sempre sentenças determinadas (quer válidas, quer contraválidas). É bem verdade que expressões lógicas podem aparecer em sentenças eventualmente indeterminadas, como por exemplo a expressão “ou”, aparentemente lógica, na sentença “x é um elefante ou x uma girafa”. Também é verdade que uma expressão que nos parece descritiva, como “elefante”, pode aparecer em uma sentença determinada, como por exemplo a sentença válida “x é um elefante ou x não é um elefante”. No entanto, se uma sentença fosse composta *somente* por expressões lógicas, então ela deveria ser determinada. Na visão de Carnap, não parece fazer muito sentido classificar certas expressões como lógicas se, combinada *entre si*, elas puderem produzir sentenças indeterminadas.

É esse o princípio do qual Carnap parte, e é exatamente esse o princípio que sua construção busca captar, por trás de todo o aparato técnico necessariamente complexo que ele utiliza para tratar a situação sintática em toda a sua generalidade. Basicamente, o que a construção transcrita acima diz é: A classe de expressões lógicas de uma linguagem é a maior classe de expressões dessa linguagem tal que todas as sentenças, formadas exclusivamente a partir de expressões pertencentes a essa classe, são sentenças determinadas.

Utilizando, portanto, apenas os conceitos formais que já haviam sido desenvolvidos para a classificação de *sentenças*, Carnap consegue obter uma maneira formal de classificar as *expressões* da linguagem, em lógicas e descritivas. Vale a pena observar que essa classificação também é exaustiva, porque todas as expressões que não são lógicas contam como descritivas. Mais ainda, vale a pena observar que ela é uma classificação interna a cada linguagem: Dependendo das regras de transformação disponíveis em cada cálculo, e portanto das sentenças que cada cálculo estabelece como determinadas, a classe de expressões descritivas pode variar bastante; cálculos dotados do mesmo conjunto de símbolos, e das mesmas regras de formação, podem classificar de maneira diferente suas expressões.

A esse respeito, Carnap aponta um exemplo interessante. Segundo ele, nos sistemas tradicionais, uma expressão como "todo" (ou como quer que se deseje designar a quantificação universal) funciona como expressão *descritiva*, e não como expressão lógica, como seria certamente a intenção de seus autores. Isso acontece porque, conforme Gödel havia demonstrado em seu teorema de incompletude, existe sempre um predicado numérico P tal que todas as sentenças do conjunto K_p = {P(0), P(1), ... , P(n), ... } são teoremas (portanto sentenças determinadas), mas tal que a sentença universal correspondente, "$\forall xP(x)$", não pode ser demonstrada, e portanto não é válida. Como essa sentença universal, por outro lado, certamente não pode ser contraválida, no sistema em questão, então ela é indeterminada. Sendo P é um predicado numérico, não é difícil demonstrar que, na maior parte dos sistemas desse tipo, ele deverá contar como expressão lógica. Assim, vemos que a quantificação universal aparece, ao lado de outras expressões lógicas, em uma sentença indeterminada. O resultado é que ela própria deve figurar entre as expressões descritivas da linguagem.

Por outro lado, se estendermos esses sistemas tradicionais para incluir uma regra indefinida de transformação, que permita a obtenção da universal "$\forall xP(x)$" diretamente a partir do conjunto infinito K_p, então a sentença universal deixa de ser indeterminada para ser válida, e a expressão da quantificação universal pode ser lógica. Vemos, assim, que não há uma classificação estanque para expressões lógicas e descritivas. A divisão entre os termos de uma linguagem, embora orientada por considerações extra-formais, não é absoluta, baseada em algum elemento que transcenda a formalização de cada linguagem em particular. Atua aqui, novamente, o Princípio de Tolerância.

VII

Uma vez classificadas as expressões de uma linguagem em lógicas e descritivas, Carnap procede à classificação das relações de conseqüência de uma linguagem, separando-as em conseqüências lógicas e conseqüências físicas. Para esclarecer a construção oferecida em *SLL*, adotaremos o mesmo método da seção anterior, iniciando com o trecho relevante do livro, para depois passar às explicações necessárias:

> *"Let C_2* [uma sentença] *be a consequence of K_1* [uma classe de sentenças] *in S* [uma linguagem]. *Here three cases are to be distinguished: 1. K_1 and C_2 are logical. 2. Descriptive expressions occur in K_1 and in C_2, but only as undefined symbols; here two further cases are to be distinguished: 2a. for any K_3 and C_4 which are formed from K_1 (or C_2) by the replacement of every descriptive symbol of K_1 (or C_2 respectively) by an expression of the same genus, and specifically of equal symbols by equal expressions, the following is true: C_4 is a consequence of K_3; 2b. the condition mentioned is not fulfilled for every K_3 and C_4. 3. In K_1 and C_2 defined descriptive symbols also occur; let K_1* and C_2* be constructed from K_1 (or C_2 respectively) by the elimination of every defined descriptive symbol (including those which are newly introduced as the result of an elimination); 3a. the condition given in 2a. for K_1 and C_2 is fulfilled for K_1* and C_2*; 3b. the said condition is not fulfilled. In cases 1, 2a, 3a, we call C_2 an* L-consequence *of K_1.; in cases 2b, 3b, we call C_2 a* P-consequence *of K_1. Thus the*

formal distinction between L- and P-rules is achieved." [76] (destaques do autor; colchetes meus).

Novamente, a idéia por trás dessa construção é bem mais simples do que o aparato técnico deixa em princípio transparecer. Basicamente, o critério é: Relações de conseqüência nas quais as sentenças envolvidas compõem-se somente de símbolos lógicos são ditas "lógicas" (a sentença inferida é dita "L-conseqüência" do conjunto de sentenças a partir do qual é obtida); relações de conseqüência nas quais as sentenças envolvidas contêm algum símbolo descritivo são ditas "físicas" (a sentença inferida é dita "P-conseqüência" do conjunto de sentenças a partir do qual é obtida).

Sendo a idéia básica tão simples (caso 1 da passagem citada acima), onde aparece a complicação? Os casos 3a e 3b são trivialmente reduzidos aos casos 2a e 2b: trata-se simplesmente de, em havendo símbolos definidos (símbolos introduzidos anteriormente por alguma definição explícita), expandir a definição. O problema reside, portanto, nos casos 2a e 2b. Que situação eles buscam captar?

Suponhamos que seja feita a seguinte transformação: a partir das duas sentença "x é um elefante" e "todo elefante é azul", obtém-se a sentença "x é azul". Essa transformação (que chamaremos de "T") envolve símbolos (expressões) que, na maior parte das linguagens, serão classificadas como descritivas: "elefante" e "azul". Se a classificação das relações de conseqüência obedecesse somente ao princípio básico exposto no item 1, então a relação expressa nessa transformação deveria ser vista como uma relação física.

Lembramos agora de toda a intenção de Carnap, ao oferecer essas construções, está em mostrar a possibilidade (e utilidade) de captar, formalmente, as distinções intuitivas que traçamos entre a esfera lógica e a esfera descritiva (ou física) de uma linguagem. De acordo com essa orientação, porém, a transformação T parece ter uma natureza claramente lógica. Uma boa divisão entre a esfera lógica e a esfera descritiva da linguagem, nesse sentido, não deveria contar T como um procedimento físico de dedução, baseado em observações empíricas, mas como um procedimento lógico. Exatamente por que razão, no entanto, tendemos a ver a dedução T como uma dedução lógica?

---

[76] *SLL*, pág. 181.

Carnap dá a resposta: Tendemos a ver a dedução T como lógica porque, embora envolva termos descritivos, ela será válida para qualquer termo descritivo que puder ser substituído no lugar de “elefante” e de “azul”[77]. É por essa razão que a relação de conseqüência entre as sentenças envolvidas na transformação T deve ser vista como uma relação lógica, e não como uma relação física (descritiva). Eis, portanto, a situação que o item 2a busca captar como instância de relação *lógica* de conseqüência: aqueles casos em que, embora apareçam uma ou mais expressões descritivas, seu caráter descritivo revela-se não-essencial, na medida em que essas expressões podem ser substituídas por qualquer outra expressão do mesmo “genus”, sem que com isso seja invalidada a relação de conseqüência.

O que obtivemos com essa nova construção, portanto, foi uma classificação formal das *relações de conseqüência* de uma linguagem, a partir da classificação formal anteriormente obtida das *expressões* da linguagem. A classificação, mais uma vez, é exaustiva.

VIII

O último andar, no edifício conceitual de Carnap, pode ser finalmente alcançado. Na classificação anterior das sentenças de uma linguagem, tínhamos a seguinte situação: Uma sentença é válida se for *conseqüência* do conjunto vazio de sentenças; é contraválida se sua negação for *conseqüência* do conjunto vazio de sentenças; é indeterminada se não for nem válida, nem contraválida. (Acrescente-se: é determinada se não for indeterminada, ou seja, se for válida ou contraválida.)

Após classificar as relações de conseqüência de uma linguagem em relações lógicas e físicas, estamos agora em condições de definir os seguintes novos conceitos: Uma sentença será dita **analítica** se for *conseqüência lógica* do conjunto vazio de sentenças; será dita **contraditória** se sua negação for *conseqüência lógica* do conjunto vazio de sentenças;

[77] Aqui, a idéia de “genus” (mencionada na explicação que Carnap oferece para 2a), refere-se justamente à classe dos símbolos que podem ser substituídos, significativamente (de acordo com as regras frmais de formação), no lugar de uma expressão. No lugar de elefante, por exemplo, devemos testar a substituição de expressões como “girafa”, “macaco” etc., mas não das expressões “vermelho” ou “pesado”. Inversamente, no lugar da expressão “azul”, podem ser substituídos, significativamente, apenas expressões da linguagem como “vermelho” e “pesado”, mas não como “girafa” ou “macaco”.

será dita **sintética** se não for nem analítica, nem contraditória. (Acrescente-se: será dita **L-determinada** se não for sintética, ou seja, se for analítica ou contraditória.)

Vemos assim que a classificação das sentenças em analíticas, contraditórias e sintéticas nasce de uma restrição das relações de conseqüência de uma linguagem, para se considerar somente as relações de conseqüência lógica (tal como formalmente caracterizadas). Carnap chama essa nova linguagem, em que as relações de conseqüência ficam restringidas às relações de conseqüência lógica, de a "L-sub-linguagem" associada a uma linguagem qualquer. Os novos conceitos, por esse motivo, são chamados de L-conceitos (sentenças analíticas são também chamadas de "L-válidas"; sentenças contraditórias de "L-contraválidas"; sentenças sintéticas de "L-indeterminadas").

O resultado prático dessa nova classificação também não é difícil de entender. Restringidas as relações de conseqüência de uma linguagem, algumas sentenças que antes eram determinadas passam para o campo das sentenças L-indeterminadas: de acordo com as regras de transformação originais do cálculo, elas (ou suas negações) podiam ser obtidas a partir do conjunto vazio, ou seja, figuravam entre as conseqüências do conjunto vazio, e por isso classificavam-se como determinadas; contudo, elas não figuram entre as conseqüências *lógicas* do conjunto vazio, e por isso passam a contar como sentenças L-indeterminadas.

Se concentrarmos nossa atenção no conceito de analiticidade, portanto, podemos dizer que ele surge como uma restrição do conceito de validade. As sentenças válidas de um cálculo são todas as sentenças que podem ser obtidas a partir das regras formais do cálculo (são todas as sentenças afirmadas pelo cálculo). Já as sentenças analíticas são todas aquelas sentenças que podem ser obtidas exclusivamente por meio dos recursos lógicos do cálculo (são afirmadas como sentenças *logicamente* válidas do cálculo). Aqui, o fato importante é que a divisão entre os recursos lógicos e os recursos não-lógicos (físico-descritivos) de um cálculo é oferecida de maneira rigorosamente formal.

IX

Para encerrar este capítulo, será útil resumir – de maneira esquemática e organizada – o caminho percorrido por Carnap para a obtenção do quadro conceitual que descrevemos acima. Ao realizar essa tarefa, poderemos realçar a interconexão entre os diversos conceitos e verificar o encadeamento das diversas etapas por que se constrói a sintaxe geral, bem como fixar seus aspectos mais destacados.

Logo de cara, devemos ter bem claro o ponto de partida da construção de Carnap: o conceito de "conseqüência direta". É este conceito que, no esquema sintático de *SLL*, fornece a totalidade da estrutura dedutiva de uma linguagem. Uma linguagem formal é um conjunto de regras referidas a um conjunto de símbolos. Entre as regras, há regras de formação (as quais estabelecem e delimitam a capacidade expressiva da linguagem) e regras de transformação (as quais determinam a estrutura dedutiva de uma linguagem). O conceito de "conseqüência direta" condensa o conjunto das regras de transformação do cálculo: uma sentença C é conseqüência direta de um conjunto K de sentenças se existir uma regra de transformação que, utilizando K como conjunto-base, permita a obtenção de C.

A respeito da admissibilidade de regras de transformação indefinidas, utilizadas por Carnap como parte do método formal, já tratamos no Capítulo 2. Tudo o que ficou dito ali, portanto, aplica-se ao conceito de conseqüência direta. Em especial, chamamos a atenção para o fato de que o conceito de conseqüência direta não deve ser considerado essencialmente mais problemático do que o conceito de derivabilidade direta, seu equivalente para linguagens que utilizam apenas regras definidas de transformação. Segundo resultou da nossa análise (Capítulo 2, seção V), é necessário garantir apenas que o conjunto-base das diversas regras de transformação esteja bem descrito, de maneira que seja possível sempre reconhecer os casos de aplicação de cada regra.

Eis então o ponto de partida da sintaxe geral. Qual o passo seguinte? O passo seguinte é a construção, a partir do conceito de conseqüência direta, do conceito de "conseqüência" (cuja definição nós fornecemos na seção IV do capítulo anterior). Trata-se de um passo *aparentemente* simples: O conceito de "conseqüência direta" expressa uma relação entre sentenças (entre uma sentença e um conjunto de sentenças); de igual maneira, o conceito de "conseqüência" também expressa uma relação entre sentenças (entre uma sentença e um conjunto de sentenças). Essa segunda relação define-se como uma extensão da relação anterior; por isso mesmo, o procedimento, embora tecnicamente complexo, dá a impressão de não envolver nenhum problema mais sério (veremos, contudo, que a situação é aqui bem mais complicada do que aparenta).

Após obter o conceito de conseqüência, Carnap vê-se em condições de definir diretamente, e sem grandes dificuldades técnicas, os conceitos de “validade”, “contra-validade”, “sentença determinada” e “sentença indeterminada”. É importante observar que esses quatro novos conceitos, diferentemente dos anteriores, não expressam *relações entre* sentenças (entre uma sentença e um conjunto de sentenças). Eles se aplicam diretamente às sentenças, classificando-as exaustiva e exclusivamente[78] por meio do par determinadas/indeterminadas ou, o que é essencialmente a mesma coisa, pelo trio válidas/contraválidas/indeterminadas.

Tecnicamente, a passagem de um conceito que *relaciona* conjuntos de sentenças para outros conceitos que *classificam* diretamente essas sentenças é realizada por meio da fixação de um conjunto de referência: o conjunto vazio. Assim, recordamos que as sentenças válidas definem-se como *conseqüência* do *conjunto vazio*; as sentenças contraválidas definem-se como aquelas cuja negação é *conseqüência* do *conjunto vazio*; as sentenças indeterminadas são todas as restantes.

De posse dessa classificação exaustiva e exclusiva das *sentenças* de uma linguagem, Carnap passa para a classificação das *expressões* da linguagem. Ele as divide, também de maneira exaustiva e exclusiva, em duas classes: “expressões lógicas” e “expressões descritivas”. A construção dessa divisão é orientada, abertamente, por certa visão intuitiva que temos a respeito daquilo que deva contar como termo descritivo (empírico) de uma linguagem, e daquilo que deva contar como termo lógico. No entanto, Carnap descreve certos meios capazes de traduzir para o âmbito formal, segundo sua opinião, o mecanismo por trás dessa divisão intuitiva.

Uma vez obtida essa classificação das expressões da linguagem, em lógicas e descritivas, Carnap retorna ao conceito de “conseqüência”. Ele subdivide esse conceito em outros dois: “conseqüência lógica” e “conseqüência física”. Esses dois novos conceitos são,

[78] Embora seja usual, vale a pena deixar bem clara essa terminologia, que estamos empregando desde o capítulo anterior: Por uma classificação *exaustiva*, queremos dizer que nenhum objeto de certa classe de objetos fica de fora da classificação; uma classificação exaustiva de *sentenças*, assim, é uma classificação que abrange *todas as sentenças* da linguagem em um sistema de conceitos. Por uma classificação *exclusiva* queremos dizer que nenhum dos objetos classificados pertence a mais de uma categoria do sistema conceitual.

assim como o conceito de conseqüência, conceitos que relacionam sentenças (uma sentença a um conjunto de sentenças). Entre as sentenças que antes estavam relacionadas pelo conceito de conseqüência, nem todas estarão relacionadas pelo conceito de conseqüência lógica; e nem todas estarão relacionadas pelo conceito de conseqüência física. Mas, se duas sentenças (uma sentença e um conjunto de sentença) estavam relacionadas pelo conceito de conseqüência, então elas estarão relacionadas, ou pelo conceito de conseqüência lógica, ou pelo conceito de conseqüência física. De fato, esse último é definido por exclusão: todas as relações de conseqüência que não são lógicas, são físicas.

Se olharmos para o conceito de conseqüência lógica, portanto, constatamos que ele surge como uma restrição do conceito de conseqüência. As mesmas definições que, a partir do conceito de conseqüência, conduziram aos conceitos de validade, contravalidade, sentença determinada e sentença indeterminada, podem agora ser refeitas a partir do conceito mais restrito de conseqüência lógica. Fixado novamente o conjunto vazio como conjunto de referência, obtêm-se um novo conjunto de conceitos que se aplicam diretamente a sentenças. Esses novos conceitos classificam as sentenças de uma linguagem, exaustiva e exclusivamente, em sentenças analíticas, contraditórias e sintéticas. (As sentenças analíticas definem-se como *conseqüência lógica* do *conjunto vazio*; sentenças contraditórias definem-se como aquelas cuja negação é *conseqüência lógica* do *conjunto vazio*; sentença sintéticas são todas aquelas que não são nem analíticas, nem contraditórias; vistas em conjunto, as sentenças analíticas e contraditórias são ditas “L-determinadas”, ao passo que as sintéticas são também chamadas de “L-indeterminadas”.)

É assim, portanto, que Carnap chega ao conceito de analiticidade. Cumpre agora examinar atentamente, quanto à sua validade e pressupostos, todas as passagens que descrevemos acima.

Capítulo 4: O segundo conceito de analiticidade[79]

I

No presente capítulo, iremos expor a maneira como Carnap constrói o conceito de analiticidade para a sua linguagem II (e que estamos chamando de "segundo" conceito de analiticidade[80]). Para que isso possa ser feito, devemos indicar – ainda que muito brevemente – algumas características que Carnap confere a essa linguagem.

Recordemos que, em *SLL*, Carnap desenvolve explicitamente a sintaxe de duas linguagens formais: a linguagem I (na parte I do livro) e a linguagem II (na parte III do livro). Seu propósito, ao desenvolver essas linguagens, é variado. Elas servem de referência para a maior parte das discussões propostas na obra. Por um lado, ambas permitem a Carnap expor o seu método de elaboração sintática de sistemas lógicos; por outro lado, o caráter essencialmente distinto de uma e outra permite-lhe também demonstrar na prática as vantagens que o Princípio de Tolerância oferece para o debate acerca dos problemas de fundamentação da matemática e das ciências. São essas duas linguagens que servem ainda de suporte para a discussão do método de aritmetização de Gödel, e dos resultados de incompletude obtidos por meio desse método. Finalmente, a linguagem II em particular fornece a Carnap a base para expor um "critério de validade" para a matemática clássica.

[79] É interessante deixar anotado desde o início que a exposição realizada neste capítulo recorrerá – ao contrário do que vinha sendo feito neste trabalho – a relativamente poucas citações do texto de *SLL*, principalmente nas seções finais, em que a parte técnica torna-se mais complicada. Essa opção, no entanto, parece-nos bastante justificada, pelas seguintes considerações.
A definição do segundo conceito de analiticidade é realizada, ao longo de diversas seções, por meio de quatro etapas distintas, em um total de mais de 50 regras. Muitas dessas regras fazem referência cruzada; vistas em conjunto, portanto, elas compõem uma rede nada simples de ser compreendida em suas múltiplas articulações. Nosso objetivo, no entanto, reside justamente em tentar esclarecer a construção de Carnap, de maneira a poder apreender (e explicar) suas principais características. Essa tarefa, que assumimos, demanda que possamos separar seus diversos elementos. Devemos, em outras palavras, reorganizar o material oferecido por Carnap – reorganizar a estrutura de sua definição – de modo a tornar mais evidente seu significado. Essa reorganização, no entanto, determinou que não fosse aconselhável (ou mesmo possível) citar os trechos do texto de onde as idéias são extraídas: muitas vezes, uma idéia que expomos encontra-se espalhada em diversos trechos; e alguma característica da definição, para a qual chamamos a atenção, só pode ser compreendida pela articulação de diferentes momentos do texto.
A ausência de citações explícitas, contudo, não significa - *e esse fato é importante de consignar* – que não estejamos sendo fiéis à construção. Pelo contrário: continuamos com a nossa orientação, presente em todo este trabalho, de permanecer extremamente próximos ao texto de *SLL*, para fazer-lhe uma análise que se baseie, na maior medida possível, em seus elementos endógenos.
[80] A esse respeito, ver a seção I do Capítulo 3 e, mais especificamente, a nota 62.

Isso acontece porque somente a linguagem II possui recursos suficientes para expressar toda a matemática clássica. Eis o que o próprio Carnap escreve, logo no início da parte III de *SLL*:

> *"Language I, with which we have been concerned up to the present, contains only definite concepts; in the domain of mathematics it contains only the arithmetic of natural numbers, and that only to an extent which corresponds approximately to a finitist or intuitionist standpoint. Language II includes Language I as a sub-language; (...). But Language II is far richer in modes of expression than Language I. It also contains indefinite concepts; it includes the whole of classical mathematics (functions with real and complex arguments; limiting values; the infinitesimal calculus; the theory of aggregates); and in it, in addition, the sentences of physics may be formulated."* [81]

A linguagem II, portanto, é construída para ser uma linguagem bastante rica em modos de expressão. O que isso significa? Na prática, isso significa duas coisas. Em primeiro lugar, que Carnap dotou-a de predicados (e funtores[82]) de todos os níveis lógicos possíveis, e não apenas de predicados e funtores numéricos (de primeira ordem), como era o caso para a linguagem I. Cumpre aqui mencionar que o conceito de nível lógico (e de tipo lógico) adotado em *SLL* baseia-se, diretamente (com algumas poucas particularidades sem grande relevância), na revisão proposta por Ramsey para a teoria clássica de tipos elaborada por Russell. Mais especificamente, Carnap utiliza a chamada "teoria simplificada" de tipos, que Ramsey ofereceu, ainda na década de 1920, como um aprimoramento da "teoria ramificada" de tipos[83]. Carnap é explícito em relação à perspectiva que deseja adotar:

---

[81] *SLL*, pág. 83.

[82] Os funtores – que aparecem como extensão do conceito matemático de função – são entidades formais que levam de um argumento a um valor. Sua estrutura geral é: F(U1)=U2, em que tanto U1 (que Carnap denomina de "argument-expression") como U2 (que Carnap denomina de "value-expression") são expressões de um tipo lógico determinado.

[83] [Ramsey, 1925]

> *"The* classification of types *outlined above is, in its essential points, the so-called* simple *classification of types proposed by Ramsey."* [84] (destaques do autor)

O segundo ponto importante, em relação à riqueza de modos de expressão da linguagem II, diz respeito ao seu repertório de variáveis, e ao tipo de quantificação admitida sobre essas variáveis. Na linguagem II, existem variáveis para todos os tipos lógicos possíveis, além de variáveis sentenciais (e não apenas variáveis numéricas, como era o caso para a linguagem I). Em relação a todas essas variáveis, mais ainda, Carnap aceita aquilo que chama de "operadores ilimitados" ("unlimited operators"). Esse tipo de operador traduz a idéia de quantificação irrestrita, ausente da linguagem I.

Na linguagem I, com efeito, os dois operadores disponíveis – o operador existencial e o operador universal – são construídos como operadores limitados. Aplicados a uma variável numérica (as únicas admitidas), eles ficam sempre limitados por uma expressão auxiliar, cuja função é indicar um campo finito para a quantificação. Dessa maneira, não há na linguagem I recursos suficientes para escrever uma sentença como: "Todos os números naturais têm a propriedade P"; ou ainda: "Existe um número natural com a propriedade P". Esse tipo de quantificação deveria abranger todos os números naturais, ilimitadamente, e está excluída da linguagem I. Nela, é necessário sempre limitar a quantificação, dentro de uma faixa restrita de números naturais (vistos como uma classe específica de expressões formais). São possíveis apenas sentenças como: "Todo número natural *até 7* têm a propriedade P"; "Existe um número natural até *105.899* com a propriedade P".

Na linguagem II, desaparecem essas restrições, e os operadores podem quantificar de maneira ilimitada. No caso de variáveis numéricas, já sabemos o que isso significa: a possibilidade de quantificar sobre todos os números naturais, ou sobre todas as expressões lingüísticas que desempenham o papel de números naturais[85]. Por outro lado, no caso de variáveis para outros tipos lógicos, a situação é mais complexa, e nós aceitamos por enquanto a seguinte resposta provisória: A quantificação irrestrita sobre uma variável de determinado tipo lógico indica uma quantificação sobre *todos* os elementos daquele

---

[84] *SLL*, pág. 86.

[85] Tanto no caso da linguagem I como da linguagem II, as expressões lingüísticas que desempenham o papel de números naturais são: 0, 0', 0'', 0''', ...

determinado tipo lógico. Por exemplo: Pela quantificação irrestrita sobre uma variável para predicados de primeira ordem do tipo lógico que Carnap designa por “(0)”, indica-se a quantificação sobre *todos* os predicados de primeira ordem desse tipo lógico. Haverá, certamente, divergências importantes na hora de interpretar a abrangência do termo “todos”. No entanto, o que nos interessa aqui, no momento em que sumariamos as características da linguagem II, é apenas indicar que ela possui esses recursos de expressão, ou seja, a quantificação irrestrita para todos os tipos lógicos.

Assim, podemos encerrar esta seção com o seguinte resumo: A linguagem II, desenvolvida por Carnap na parte III de *SLL*, segue os moldes deitados por Russell e Whitehead, nos *Principia Mathematica*, para sistemas lógicos formais. Incorpora, porém, a teoria simplificada de tipos, na linha sugerida por Ramsey. Finalmente, de suas características específicas, interessa-nos o fato de que a linguagem II possui variáveis para todos os tipos lógicos, e admite quantificação irrestrita sobre todas as suas variáveis. Essa estrutura geral da linguagem II determinará boa parte dos problemas com que Carnap terá de lidar na hora de fornecer sua segunda definição de analiticidade.

II

No esquema da sintaxe geral, conforme enfatizamos no capítulo precedente, o conceito de “conseqüência direta” é o primeiro conceito que deveria ser oferecido para caracterizar uma linguagem. Na verdade, o conceito de “conseqüência direta” é *o conceito* que caracteriza um cálculo formal qualquer: uma vez indicado, já contém todas as informações relevantes a respeito de sua estrutura. É a partir das regras de conseqüência direta que todos os conceitos sintáticos ficam determinados, aí incluída a classificação das sentenças em analíticas, contraditórias e sintéticas. No caso da linguagem II, porém, Carnap opta por uma estratégia diferente. Que estratégia é essa?

De início (seções 26 a 33 de *SLL*), Carnap trata a linguagem II como se fosse uma linguagem do tipo definido, isto é, como se ela contivesse apenas regras definidas de transformação. Devemos explicar o que queremos dizer com isso.

Carnap começa por fornecer as regras de formação da linguagem II. Essas regras, como vimos, são sempre definidas (Carnap não encontra nenhuma razão para estender seu método sintático desnecessariamente nessa direção, pela adoção de regras indefinidas de formação; ver seção III do Capítulo 2 e, em especial, a nota 41). Nesse sentido, podemos dizer que Carnap opta por expor, primeiramente, a parte supostamente não-problemática da linguagem: Na seção 26, ele indica o seu aparato simbólico; na seção 27, esclarece a classificação de tipos que pretende usar; nas seções 28 e 29, fornece as regras de formação propriamente ditas (essas quatro seções compõem uma subparte da parte III, intitulada "A: Rules of Formation for Language II"). Dessa maneira, ele consegue descrever sem problemas os recursos expressivos da linguagem II. As características da linguagem II, por exemplo, que indicamos na seção anterior, são estabelecidas nesse momento. Resta agora descrever a estrutura dedutiva da linguagem II. Como sabemos, é aí que – segundo o esquema de *SLL* – residem os maiores problemas.

As seções 30 a 33 compõem a subparte B da parte III, intitulada "B: Rules of Transformation for Language II". Nas seções 30 e 31, respectivamente, Carnap oferece as sentenças primitivas da linguagem II, e as regras de inferência da linguagem II. As seções 32 e 33 são dedicadas, simplesmente, a fornecer exemplos e comparações com outros sistemas lógicos. O fato importante a notar é que as regras de inferência descritas na seção 31 são regras definidas, do tipo mais simples possível (Carnap oferece apenas duas regras: uma delas baseia-se em duas premissas; a outra, em uma única premissa).

As sentenças primitivas oferecidas na seção 30, juntamente com as regras de inferência oferecidas na seção 31, estabelecem uma estrutura dedutiva para a linguagem II. Trata-se de uma estrutura dedutiva definida (porque as regras de inferência são definidas). Como tal, ela se assemelha muito à estrutura dedutiva dos sistemas de lógica formal tradicionais, já existentes na época, e não escapa aos resultados de incompletude já então demonstrados por Gödel. Como nós sabemos, porém, Carnap não deseja parar por aí. Seu objetivo, ao desenvolver a linguagem II, é justamente oferecer uma linguagem formal que consiga captar as sentenças da matemática clássica – e isso ele tentará fazer por meio de uma estrutura dedutiva indefinida.

Ao construir a linguagem I, Carnap já havia recorrido a essa mesma tática de exposição: primeiramente, ofereceu um conjunto de regras definidas de inferência; somente depois acrescentou uma regra indefinida, a que chamou de "regra de conseqüência". No esquema de Carnap, portanto, a estrutura indefinida de dedução é colocada, de certa soma, "por cima" de uma estrutura mais simples, definida. Para a estrutura dedutiva definida – que é, tanto no caso da linguagem I como no caso da linguagem II, uma estrutura desenvolvida segundo os moldes tradicionais –, ele reserva a nomenclatura tradicional, a respeito da qual já dissemos alguma coisa no Capítulo 3: uma cadeia de inferências é uma "derivação" (sentenças são "deriváveis" a partir de outras); uma cadeia de inferências que utiliza somente as sentenças iniciais é uma "demonstração" ("prova"); as sentenças dividem-se em "demonstráveis" (os teoremas), refutáveis (aquelas cuja negação é demonstrável) e irresolúveis (aquelas que não são nem demonstráveis, nem refutáveis).

É para a segunda estrutura dedutiva – aquela baseada em regras indefinidas – que ele reserva a classificação conceitual exposta no capítulo anterior. Rigorosamente falando, estruturas dedutivas distintas correspondem a linguagens distintas. No caso das linguagens I e II, no entanto, sua intenção é suficientemente clara para não dar margem a nenhuma dúvida. Cada uma dessas linguagens tem uma estrutura expressiva própria (regras de formação), e duas estruturas dedutivas compatíveis: a segunda estrutura – a indefinida – funciona como uma extensão da primeira – a definida – no sentido de que todas as sentenças demonstráveis são também analíticas, todas as sentenças refutáveis são também contraditórias, toda sentença derivável a partir de outras é também conseqüência dessas outras, etc.

No caso da linguagem I, no entanto, bastava acrescentar uma única regra de conseqüência para fixar essa nova estrutura dedutiva, indefinida. Trata-se da regra exposta na página 38 de *SLL*, com o nome de DC2 (DC1 retoma as regras definidas da linguagem I). Essa abordagem assim tão simples é possível – apenas uma única regra indefinida de transformação, e nada complicada – porque a linguagem I possui apenas variáveis numéricas. Basicamente, o que DC2 estabelece é que, a partir do conjunto-base infinito K_P = {P(0), P(0'), P(0''), ... , P(0'''$^{...}$'), ... }, pode-se deduzir, como conseqüência, a sentença aberta correspondente: P(x). (Note-se que, como não existe quantificação irrestrita na linguagem I, não se pode falar em deduzir a sentença universal (fechada) correspondente, a qual seria expressa – em uma linguagem com quantificação irrestrita como a linguagem II – por "$\forall x P(x)$", ou outro simbolismo equivalente.)

No caso da linguagem II, que possui variáveis para todos os tipos lógicos, essa abordagem tão direta e simples não se mostra mais possível. Nas próximas seções, tentaremos compreender os problemas que Carnap terá de enfrentar, bem como a solução que irá oferecer para eles. Esse esforço traduz-se na definição do conceito de analiticidade para a linguagem II. Carnap empreende essa tarefa de oferecer uma estrutura dedutiva completa para a linguagem II – não mais somente a estrutura dedutiva definida, mas também a estrutura dedutiva indefinida – na subparte da parte III intitulada "C: Rules of Consequence for Language II", a qual compreende diversas seções[86]. É esse o esquema que devemos agora examinar.

III

A primeira coisa que salta aos olhos, quando verificamos a maneira como Carnap constrói a estrutura dedutiva (completa) da linguagem II, é o seguinte fato peculiar, de notável importância teórica: As regras gerais de transformação (regras indefinidas de conseqüência) não são fornecidas diretamente. É o oposto do que havia acontecido com a linguagem I e, mais intrigante, o oposto daquilo que será preconizado na sintaxe geral. Não custa lembrar que, no sistema sintático de *SLL* (expresso da maneira mais organizada justamente na sintaxe geral), são precisamente as regras de transformação que fornecem a própria substância de um cálculo formal, e é a partir delas que todos os outros conceitos lógico-formais deveriam ficar determinados.

Para a linguagem II, contudo, Carnap adota outro método. Ele começa por expor, diretamente, o conceito de "analiticidade". Somente depois – a partir desse conceito de analiticidade – é que ele constrói o conceito de conseqüência. Eis as palavras que iniciam a seção 34b:

> *"Our procedure in laying down the consequence-rules for language I (§ 14) was first to define the term 'consequence' by means of the expansion of the rules of inference and then, with its help, the terms 'analytic' and 'contradictory'. In laying down the consequence-rules for language II, we*

[86] A subparte C compõe-se das seguintes seções: 34a até 34i, 35 e 36. Dessas, as seções 34a a 34f são dedicadas diretamente a expor a estrutura dedutiva indefinida da linguagem II.

shall, for technical reasons, do just the reverse: first we shall define 'analytic' and 'contradictory' and then, with the help of these terms, the term 'consequence'." [87]

Somente esse aspecto, isoladamente considerado, já representaria uma diferença bastante significativa no modo de conceber a abordagem sintática, a demandar algum esclarecimento mais preciso. Contudo, na exposição da estrutura dedutiva da linguagem II, não é apenas a ordem de definição dos termos que será diferente ("por razões técnicas", como Carnap aponta com vagueza incomum em seu trabalho). A própria maneira como o conceito de analiticidade é construído revela-se bastante distinta da maneira como aparecem todos os outros conceitos sintáticos de *SLL*. Em vez de indicar diretamente as condições a que uma sentença deve obedecer – ou as características que deve possuir – para contar como analítica, Carnap estabelece um complicado procedimento que, ao final de certo número de passos e etapas, deve permitir determinar o seu *status* (analítica, contraditória ou sintética).

A primeira etapa desse procedimento – dita "redução" – é descrita na seção 34b. Seu objetivo é levar as sentenças da linguagem II (qualquer sentença) para uma forma padrão – chamada de forma "reduzida" –, a partir da qual as próximas etapas possam se desenvolver. Trata-se de uma "receita" que, por meio da aplicação articulada de 32 regras diferentes, permite converter qualquer sentença dada em uma outra sentença sintaticamente equivalente, adequada ao prosseguimento das etapas seguintes.

Encontramos aqui, na padronização da forma sintática, um tipo de operação bastante comum. Em princípio, essa padronização possibilita visualizar as características lógicas relevantes de uma sentença, para tratá-las de maneira simplificada e organizada. Os termos introduzidos por definição são eliminados (substituídos pelas respectivas definições); os quantificadores são todos levados para o início da sentença; os conectivos lógicos são reduzidos em número e organizados segundo uma ordenação específica; e outras operações desse gênero. A partir de uma sentença C, obtém-se a reduzida C^r. Três aspectos dessa transformação, no entanto, devem ser destacados.

[87] *SLL*, pág. 102.

Primeiramente, deve-se enfatizar o fato, mencionado logo acima, de que todos os quantificadores são trazidos para o início da sentença reduzida C^r. Na verdade, os quantificadores numéricos restritos (que também existem na linguagem II, e que tem sua faixa de quantificação limitada por uma expressão numérica) são eliminados. Isso é feito de maneira simples. Uma sentença, por exemplo, como "Todo número até 2 possui a propriedade P" é transformada em uma série de conjunções: "0 possui a propriedade P e 1 possui a propriedade P e 2 possui a propriedade P". Sobram apenas os quantificadores irrestritos, aplicados a variáveis de qualquer tipo lógico. Esses sim, são todos trazidos para as primeiras posições da sentença reduzida.

Em segundo lugar, chamamos a atenção para o fato essencial de que as duas sentenças – a sentença original e a sua forma reduzida – são equivalentes do ponto de vista formal. Esse fato é essencial porque confere o próprio significado da transformação. O objetivo geral do procedimento oferecido por Carnap é, em vista de uma sentença C, determinar seu *status* sintático (se é analítica, contraditória ou sintética). A estratégia é: a partir de C, obtém-se a reduzida C^r; determina-se então o *status* dessa reduzida C^r, e conclui-se daí para o *status* de C. Isso só funciona, obviamente, se o *status* sintático de C e C^r forem sempre iguais. E é isso certamente o que a operação de redução tem de garantir: a sentença reduzida tem de ser equivalente à sentença original. Do ponto de vista formal, isso significa que ambas devem ser mutuamente deriváveis: a partir de C, deve ser possível derivar C^r; a partir de C^r, deve ser possível derivar C. Carnap, de fato, fornece o seguinte teorema:

*"**Theorem 34b.I:** C and C^r are always mutually derivable."* [88]

É interessante observar, aqui, que o conceito de derivabilidade a respeito do qual estamos falando, e que aparece no teorema acima, é o conceito de derivabilidade extraído da estrutura dedutiva "fraca" da linguagem II – aquela estrutura dedutiva baseada somente em regras definidas de transformação, fornecida por Carnap na subparte B da parte III. Essa circunstância, porém, não nos deve espantar. A definição da estrutura dedutiva "forte" da linguagem II, conforme já observamos, é concebida como extensão da estrutura fraca, no

[88] *SLL*, pág. 105.

sentido de estabelecer novas relações de conseqüência anteriormente inexistentes, sem contudo descartar nenhuma das relações de derivabilidade já estabelecidas.

Do ponto de vista formal, seria perfeitamente possível fornecer as operações de redução sem referência a nenhuma equivalência prévia entre as sentenças C e C^r. As regras de redução poderiam servir, elas mesmas, como definidoras de uma nova relação de equivalência, afirmada exclusivamente em relação à estrutura dedutiva forte. O *status* sintático de C, nesse sentido, ficaria definido pelo *status* sintático de C^r. Carnap, por outro lado, prefere sempre manter uma relação próxima – cuja nota essencial é dada justamente pela idéia de extensão dos recursos dedutivos – entre o novo procedimento que está buscando construir (e que fornecerá uma estrutura dedutiva mais forte), e os conceitos mais usuais de sistemas lógicos tradicionais, incorporados à linguagem II por meio de suas regras definidas de inferência (aquilo que estamos chamando de estrutura dedutiva fraca). O teorema 34b.I, assim, serve como amarra entre as duas estruturas dedutivas da linguagem II, e garante a compatibilidade entre uma e outra.

Finalmente, o terceiro aspecto que devemos destacar é a introdução, durante a exposição feita por Carnap do processo de redução, do símbolo especial "Ŋ". Esse símbolo é uma abreviação para a sentença "0=0", e desempenhará um papel importante na definição do conceito de analiticidade para a linguagem II. Como veremos, o símbolo "Ŋ" servirá como paradigma – ou modelo – para as sentenças analíticas (~Ŋ, a negação de Ŋ, servirá de modelo para as sentenças contraditórias). Assim, na seção 34b, encontramos regras de redução como: "*RR5a: Ŋ results from* U_1=U_1"; ou "*RR3c: If* C_2 [uma sentença] *is a disjunction of which one member is Ŋ, then Ŋ results from* C_2" [89]. Mais à frente em *SLL*, na etapa final da construção do conceito de analiticidade, em que sua definição é finalmente fornecida, a sentença Ŋ (e portanto todas as que foram reduzidas a ela pelas etapas anteriores) será definida como analítica.

IV

A segunda etapa do procedimento que estamos examinando – cujo objetivo é determinar o *status* sintático de uma sentença qualquer da linguagem II – é aquela que

[89] As duas regras citadas encontram-se em *SLL*, pág. 103.

Carnap chama de “valoração” (“*valuation*”), exposta na seção 34c. É nessa idéia de valoração, indicamos desde já, que Carnap dá o passo decisivo em direção a uma teoria lógica absolutamente nova.

O desejo de Carnap é reproduzir, para variáveis de todos os tipos lógicos admitidos na linguagem II, a mesma linha de raciocínio que, na linguagem I, havia possibilitado captar o conceito de analiticidade por meio de uma única regra de conseqüência. Na linguagem I, a regra DC2 permite a dedução da sentença “P(x)” (em que “P” designa um predicado numérico e “x” designa uma variável numérica) a partir do conjunto K_P = {P(0), P(0’), P(0’’), ... , P(0’’’^{...}’), ... }. Obviamente, P(x) será analítica se todas as sentenças do conjunto K_P forem analíticas (pode-se mostrar que essa condição não é apenas suficiente, mas também necessária). O conjunto K_P, por outro lado, nasce de P(x) pela substituição da variável x por todas as expressões numéricas da linguagem.

Tomemos agora um exemplo semelhante, porém com tipos lógicos mais complexos. Suponhamos que “F” designe uma variável para predicados numéricos, e que “M” designe um predicado de segunda ordem (M é um predicado que se aplica a predicados numéricos). A sentença “M(F)”, aberta[90], diz que o predicado de segunda ordem M aplica-se a todos os predicados numéricos. A pergunta que Carnap faz é: Em que ocasiões devemos considerar uma tal sentença como analítica?

Se, no caso de variáveis numéricas, devíamos olhar para todos os valores possíveis para essas variáveis – ou seja, todos os números – o mesmo deve ser feito agora para variáveis de outros tipos lógicos. A sentença “P(x)” é analítica se, e somente se, ao substituirmos a variável numérica x por todos os seus valores, obtivermos sentenças analíticas. No caso de “M(F)”, Carnap desejaria fazer a mesma afirmação, tomando como base todos os valores possíveis da variável F. No entanto, há nesse ponto um grave problema a ser solucionado: Quais são os valores possíveis para F?

No caso das variáveis numéricas, essa questão não chega a representar uma dificuldade. Pode-se facilmente supor uma correspondência biunívoca entre os valores que essas variáveis devem representam – números naturais – e certa classe de expressões

[90] Poderíamos ter tomado a sentença fechado correspondente, mas preferimos seguir o exemplo fornecido por Carnap na pág. 106 de *SLL*, que simplifica a exposição do problema.

formais disponíveis na linguagem, pelas quais elas podem ser explicitamente substituídas. Na maioria das linguagens formais, existe de fato uma classe de expressões sintáticas que representam esses números naturais. No caso das linguagens I e II de Carnap, por exemplo, o conjunto de expressões que desempenha esse papel é dado por En = {0, 0’, 0’’, ... , 0’’’’...’, ... }. Por isso é possível expressar a analiticidade da sentença “P(x)” em termos da analiticidade das sentenças de um certo conjunto – o conjunto K_P = {P(0), P(0’), P(0’’), ... , P(0’’’...’), ... }. As sentenças desse conjunto estão disponíveis na linguagem, isto é, o conjunto K_P pode ser construído, porque os valores assumidos pela variável numérica estão disponíveis na linguagem, como expressões sintáticas explícitas.

O mesmo, porém, não é válido para tipos lógicos mais complexos. Ou, ao menos, esse foi o enfoque adotado por Carnap. Eis a passagem relevante de *SLL*:

> *“In the case of a predicate- or a functor-variable, however, the analogous method does not succeed; a fact which has been pointed out by Gödel. Let C_1 be, for example, ‘M(F)’ (in words: ‘M is true for all properties’). Now, if from C_1 we refer back to the sentences ‘M(P_1)’, ‘M(P_2)’, and so on, which result from C_1 by substituting for ‘F’ each of he predicates of the type in question which are definable in II, in turn, then it may happen that, though all these sentences are true, ‘M(F)’ is nevertheless false – in so far as M does not hold for a certain property for which no predicate can be defined in II. (...)”* [91]

Em outra ponto do livro, Carnap defende que um predicado numérico equivale a um conjunto de números: o conjunto dos números ao qual o predicado se aplica[92]. No entanto, basta um simples argumento de cardinalidade (em vez de Gödel, Carnap poderia socorrer-se de Cantor) para mostrar que nenhuma linguagem formal, baseada em seqüências finitas de símbolos, pode dispor de descrições suficientes para todos os conjuntos de números naturais. Em outras palavras, nenhuma linguagem formal consegue expressar todos os

[91] *SLL*, pág. 106.

[92] A esse respeito, basta ver a seção 38 de *SLL*, intitulada “The Elimination of Classes”. Nela, Carnap defende que um simbolismo específico para classes (conjuntos) pode simplesmente ser eliminado das linguagens formais, na medida em que a estrutura de classes é exatamente refletida pela estrutura de predicados. Nesse sentido, a cada classe corresponderia um predicado (o predicado “ser membro daquela classe”) e a cada predicado corresponderia uma classe (a classe de todos os elementos ao qual o predicado se aplica).

predicados possíveis. Ainda uma outra maneira de ver o problema, que torna essas dificuldades ainda mais palpáveis, é fornecida pelos números reais. Carnap aceita a associação entre predicados numéricos e números reais (para ele, os predicados numéricos são uma das maneiras possíveis de representar números reais na linguagem II[93]). Na continuação da passagem citada acima, ele observa:

> *"(...) As a result of Gödel's researches it is certain, for instance, that for every arithmetical system there are* numerical properties which are not definable*, or, in other words, indefinable real numbers (see Theorem 60d.1, p. 221). Obviously it would not be consistent with the concept of validity of classical mathematics if we were to call the sentence: 'All real numbers have the property M' an analytic sentence, when a real number can be stated (not, certainly, in the linguistic system concerned, but in a richer system) which does not possess this property."* [94] (destaque do autor)

Em todo caso, ficou claro para Carnap que uma sentença como M(F) jamais poderia ter sua analiticidade definida por referência à analiticidade de um conjunto de sentenças como "$M(F_1)$", "$M(F_2)$" etc., em que "F_1", "F_2" etc. representam os possíveis valores da variável de segunda ordem F. Isso porque nenhuma linguagem possui um estoque simbólico suficiente para fornecer, como expressões sintáticas explícitas, todos os valores que F deveria assumir. A pergunta que Carnap faz, então, é: No caso de variáveis de tipos lógicos superiores, como F, quais os possíveis valores que devemos levar em conta?

A solução de Carnap para essa situação é justamente o método de "valorações". O que é uma valoração? A idéia é simples. As valorações de uma variável são os possíveis valores que ela pode assumir. Uma valoração para uma variável numérica é, como seria de se esperar, um número – mais precisamente, uma expressão numérica, pois são esses os elementos sintáticos que representam os números. Mas o que seria a valoração, por exemplo, para um predicado de primeira ordem? A discussão acima fornece a resposta. Um predicado de primeira ordem é, na visão de Carnap, um conjunto de números. A valoração

[93] Ver a seção 39 de *SLL*.
[94] *SLL*, pág. 106.

de um predicado numérico é, portanto, um conjunto de números – ou melhor, de expressões numéricas. Assim como o número três – representado sintaticamente pela expressão “ 0’’’ ” – é uma valoração possível para a variável numérica x, o conjunto de números {1,3,4} – representado sintaticamente pelo conjunto de expressões numéricas {0’,0’’’,0’’’’} – é uma valoração possível para a variável F.

Carnap determinará quais são as valorações possíveis para as variáveis de todos os tipos lógicos. Não é difícil de entender a sua idéia. A valoração de um funtor de primeira ordem, que leva números em números (que assume uma expressão numérica como argumento, e assume outra expressão numérica como valor), será dada por um par ordenado de expressões numéricas. Já as valorações possíveis para um predicado de segunda ordem – um predicado que se aplica a predicados numéricos – serão dadas por conjuntos de conjuntos de expressões numéricas.

Mais precisamente, Carnap indicará uma classe de valorações para cada tipo lógico da linguagem II. A classe de valorações associada a certo tipo lógico fornece, para os propósitos da definição de analiticidade, a classe dos elementos que as variáveis daquele tipo lógico podem assumir como valor. O segredo que permitirá a Carnap construir uma classe de valorações para todos os tipos lógicos da linguagem II é a utilização do método recursivo. O primeiro conjunto de regras de valoração (regras VR1a-d), nesse sentido, indica como realizar a construção da classe de valorações de tipos lógicos cada vez mais elevados, com base nas classes de valorações de tipos lógicos mais simples. Todas essas regras remetem, portanto, à valoração de tipo lógico mais baixo, que é a valoração de variáveis numéricas como expressões numéricas. O resultado é fácil de imaginar: as valorações são dadas sempre por algum conjunto – por mais complicado que seja seu tipo lógico – que, tratado como agregado, é constituído finalmente por expressões numéricas.

Carnap fornece ainda um segundo tipo de regras que estabelecem valorações para novos tipos de expressões, sujeitas a certas restrições de acordo com sua função *dentro de uma sentença*. O primeiro conjunto de regras de valoração (VR1) indicava as classes de valorações associadas a cada tipo lógico. As novas regras (VR2) indicam como, dentro de uma sentença, a valoração das expressões deve ser feita. Trata-se aqui, simplesmente, de garantir certa coerência na valoração dos diferentes termos de uma sentença. A idéia básica é a seguinte: A valoração das variáveis livres de uma sentença pode ser escolhida arbitrariamente, entre aquelas possíveis segundo as regras VR1; no entanto, uma vez escolhidas essas valorações, a valoração de alguns outros termos fica determinada segundo as regras VR2.

Assim, se em uma sentença aparece a variável livre z e, em outra posição dessa mesma sentença, aparece a expressão “ z’ ” (que indica o sucessor de z), a valoração de z’ fica determinada pela valoração de z, de maneira óbvia. Se z receber como valoração uma expressão numérica qualquer, digamos “ 0’’ ”, então z’ deve receber como valoração o sucessor dessa expressão, no caso, “ 0’’’ ”. A papel que essas regras desempenham, portanto, não representa nenhum mistério: Se as valorações fornecem os valores possíveis que uma variável pode assumir, é necessário garantir – se se deseja valorar as diversas expressões passíveis de valoração de uma sentença – que a valoração de expressões relacionadas entre si sejam, também, relacionadas entre si.

O que expusemos no parágrafo anterior, basicamente, é a regra VR2b. As outras regras dessa família, embora mais complexas, impõem restrições essencialmente similares, que visam garantir a coerência interna das valorações dos termos de uma sentença. Não nos deteremos aqui em explicá-las uma a uma. Será interessante indicar, no entanto, o conteúdo da regra VR2a, que trata do símbolo “Ŋ”, introduzido pelas regras de redução (ver seção III acima). Ei-la: “**a:** *Ŋ itself shall be taken as the valuation for Ŋ*” [95]. Uma outra regra importante é aquela que impõe valorações idênticas para expressões idênticas, ou seja, se uma expressão aparece em diferentes posições da sentença, deve receber a mesma valoração em todas as posições em que aparece. Finalmente, devemos mencionar também que, como caso especial de valoração, Carnap inclui as próprias expressões numéricas, tais como “0” ou “ 0’’’ ”. No caso dessas expressões, a valoração é a própria expressão.

V

Na mesma seção 34c em que expõe a idéia de valoração, Carnap indica as regras de “avaliação” (“*evaluation*”). A avaliação de uma sentença, *feita com base em determinada*

[95] *SLL*, pág. 108.

valoração para todos os seus termos valoráveis, é a transformação dessa sentença em Ŋ ou ~Ŋ.

Da seção anterior, nós já sabemos que a classe de valorações de uma expressão corresponde aos possíveis valores que ela pode assumir; a escolha de uma valoração para uma expressão, portanto, corresponde à escolha de um possível valor para essa expressão. O que significa, então, a escolha de uma valoração para cada expressão de uma sentença? Uma sentença, na qual aparecem variáveis livres, não chega a fazer uma afirmação, e por isso não pode receber um *status* sintático (ou um valor de verdade). Porém, quando as variáveis são substituídas por valores concretos, a sentença se transforma, não somente em uma afirmação, mas em uma afirmação individual, que pode ser diretamente verificada. É isso o que as regras de avaliação buscam captar. Elas indicam se uma sentença, diante de uma valoração para suas expressões, faz uma afirmação correta – caso em que deve ser substituída por Ŋ, o modelo das sentenças sintéticas – ou faz uma afirmação incorreta – caso em que deve ser substituída por ~Ŋ, o modelo das sentenças contraditórias.

Carnap fornece apenas duas regras, EvR1 e EvR2, cujo conteúdo não é difícil compreender. A primeira delas estabelece o seguinte: Suponhamos uma sentença parcial, cuja forma sintática seja “P(A)”. Aqui, o símbolo “P” indica uma variável para predicados, de determinado tipo lógico, e o símbolo “A” indica um argumento do tipo lógico exigido por P. Para poder fazer a avaliação, devemos ter escolhido, primeiramente, uma valoração tanto para P, como para A (pois uma avaliação só é possível, não custa repetir, em vista da valoração de todos os termos valoráveis da sentença). Chamemos de “Vp” a valoração de P, e de “Va” a valoração de A.

Pelas regras de valoração, que relacionam justamente os diversos tipos lógicos por meio de regras recursivas, a valoração de P será dada por um conjunto de possíveis valorações para A, qualquer que seja o tipo lógico de ambos. No exemplo mais simples, se P é uma variável para predicados numéricos, as regras sintáticas impõem que A seja uma variável numérica (ou uma expressão numérica). Correspondentemente, a valoração de A é uma específica expressão numérica, ao passo que a valoração de P é um específico conjunto de expressões numéricas (conjunto de possíveis valorações de A). Podemos também tomar um exemplo mais complicado, e o resultado será o mesmo. Suponhamos que

P seja uma variável para predicados de nível dez (podemos supor que P é uma variável para predicados que aceitam, como argumento, predicados de certo tipo lógico T, de nível nove). As regras sintáticas impõem que A seja exatamente um desses predicados de tipo lógico T. A valoração de A será um elemento da classe K_t de valorações para o tipo lógico T. A valoração de P será um conjunto de elementos de K_t (pois as regras de valoração determinam, justamente, que a classe de valorações adequadas ao tipo lógico de P seja a classe dos conjuntos formados com elementos de K_t).

A regra de avaliação EvR1 estabelece, então, que a sentença P(A) deve ser substituída por Ŋ caso Va (a valoração escolhida para A) pertença a Vp (a valoração escolhida para P); caso contrário, deve ser substituída por ~Ŋ.

Novamente, a visualização do que está acontecendo fica simples se voltarmos nossa atenção para números e predicados numéricos. Suponhamos, mais uma vez, que P seja uma variável para predicados numéricos, e A uma variável numérica. Escolher uma valoração para A equivale a escolher uma expressão numérica (que representa um número natural); escolher uma valoração para P equivale a escolher um conjunto de expressões numéricas (que representa um predicado numérico). Como valoração para A, por exemplo, podemos tomar Va = 0’’ (é como se escolhêssemos o número 2 para o lugar da variável numérica A). Como valoração para P, podemos escolher Vp = {0, 0’, 0’’} (a variável P, nesse caso, estaria sendo valorada pelo predicado numérico {0,1,2,}; lembramos que um predicado numérico, para Carnap, é simplesmente um conjunto de números: mais especificamente, *o conjunto de números ao qual o predicado se aplica*.)

Agora, é fácil ver que a sentença P(A), após a escolha das valorações indicadas, está dizendo que o predicado {0,1,2} (que é o predicado que se aplica aos números 0, 1 e 2) aplica-se ao número 2. Em termos das valorações, isso é expresso pelo fato de que a expressão numérica “ 0’’ ” pertence ao conjunto de expressões {0, 0’, 0’’}, ou seja, pelo fato de que a valoração Va pertence a Vp. No caso dessas valorações, portanto,, a regra EvR1 determina que a sentença P(A) seja substituída por Ŋ. Se A fosse valorada por outra expressão numérica, digamos “ 0’’’’ ” (Va = 0’’’’), então Va não pertenceria a Vp, e a sentença deveria ser substituída por ~Ŋ.

A segunda regra de avaliação segue a mesma idéia geral de EvR1, que acabamos de explicar. Na verdade, sua compreensão é imediata. Eis o que ela estabelece: Suponhamos que uma sentença parcial[96] tenha a forma sintática "$U_1 = U_2$", em que tanto "U_1" como "U_2" são expressões valoráveis (que aceitam valoração). Escolhemos, para cada uma dessas expressões, valorações V_{U1} e V_{U2} adequadas aos respectivos tipos lógicos. Somente se V_{U1} for igual a V_{U2}, ou seja, se V_{U1} e V_{U2} forem a mesma valoração, a sentença deve ser substituída por Ŋ; caso contrário, por ~Ŋ. O significado da regra é mais do que claro: As valorações V_{U1} e V_{U2} correspondem à escolha de possíveis valores para as expressões U_1 e U_2. Em face das valorações, a igualdade original "$U_1 = U_2$" (que, por envolver expressões variáveis, não recebia um valor de verdade) transforma-se na igualdade "$V_{U1} = V_{U2}$", diretamente aferível. Essa igualdade, como é óbvio, deverá ser substituída por Ŋ somente se as duas valorações, de fato, forem iguais.

VI

Podemos fazer uma pausa, neste ponto, para avaliar o percurso realizado por Carnap até aqui, e mostrar a situação exata em que ele nos deixa. Relembramos que, nas seções anteriores, nós expusemos – ainda que sumariamente, sem entrar no emaranhado de detalhes que dificultam a leitura desse trecho de *SLL* – as três primeiras etapas do procedimento elaborado por Carnap para determinar o *status* sintático das sentenças da linguagem II. Mais especificamente, nós explicamos o processo de redução, valoração e avaliação.

A redução nos conduz de uma sentença C, cujo *status* sintático se trata de determinar, para a sua reduzida C^r. Essa sentença C^r, que de resto é equivalente a C (no

[96] Em EvR2, assim como em EvR1, Carnap fala em sentença *parciais*. As regras de formação da linguagem II, a respeito da qual pouco ou nada comentamos, mostram que uma sentença complexa pode ser formada a partir de sentenças atômicas a partir de certas operações (aplicação de conectivos etc.), da maneira mais comum em lógica formal. É a avaliação dessas sentenças atômicas – que possuem as duas formas básicas tratadas, respectivamente, pelas regras EvR1 e EvR2 – que é realizada em primeiro lugar. Carnap menciona explicitamente que, se uma sentença não reduzida resultar de uma transformação, então ela deve ser novamente reduzida, pelas regras de redução. Ele se refere, aqui, às regras de redução que se aplicam especificamente às duas sentenças canônicas Ŋ e ~Ŋ. No curso da avaliação de sentenças *parciais*, elas serão transformadas em uma dessas duas formas canônicas; a sentença total resultante, então, poderá ser novamente reduzida, por meio daquelas regras específicas de redução (por exemplo, a já citada regra RR3c – ver seção III acima).

sentido de derivabilidade recíproca), distingue-se por possuir uma forma padronizada. O aspecto mais relevante dessa padronização, que desejamos novamente enfatizar, é a posição dos quantificadores. Após a aplicação das regras de redução, todos os quantificadores (irrestritos) são trazidos para o início de C^r. Dessa maneira, C^r passa a ter a seguinte forma lógica: $[v_1]$ $[v_2]$ ... $[v_n]$ C_L. Aqui, os símbolos "v_1", "v_2", ... , "v_n" indicam variáveis de diferentes tipos lógicos; os colchetes indicam a quantificação irrestrita, seja ela universal ou existencial; e a sentença C_L é uma sentença sem quantificação, na qual todas as variáveis são livres. (Indicaremos a forma geral da sentença reduzida, acima indicada, por "[...] C_L".)

As regras de valoração, então, estabelecem como valorar (atribuir possíveis valores retirados de uma classe adequada) as diferentes expressões de uma sentença. É interessante indicar, no que segue, a lista completa de todas as expressões sintáticas que recebem valoração. Em primeiro lugar, as próprias expressões numéricas, cuja valoração é dada por elas mesmas (a valoração da expressão " 0'' ", por exemplo, é a própria expressão " 0'' "). Em segundo lugar, a sentença Ŋ (lembramos que o símbolo "Ŋ" indica a sentença "0=0"), que também é valorada por si própria (lembramos a regra VR2a: "*Ŋ itself shall be taken as the valuation for Ŋ*").

Em terceiro lugar, variáveis de todos os tipos lógicos. É em relação a essas variáveis que Carnap indica regras recursivas, suficientes para criar valorações de diferentes tipos lógicos. De fato, para cada possível tipo lógico, há uma correspondente classe de valorações possíveis. E todas as classes de valoração são construídas a partir dos elementos básicos da valoração: as expressões numéricas (que fornecem a classe de valoração para as variáveis mais simples, as variáveis numéricas). Assim, a valoração para uma variável numérica é uma expressão numérica; a valoração para uma variável para predicados numéricos é um conjunto de expressões numéricas; e assim por diante.

Em quarto lugar, predicados e funtores descritivos (que nós não havíamos mencionado até agora). Em relação a essas expressões, valem considerações análogas às realizadas no parágrafo anterior: Cada predicado ou funtor descritivo possui um tipo lógico específico, e deve receber uma valoração adequada para esse tipo lógico específico. No entanto, como os tipos lógicos desses predicados e funtores são os mesmos tipos lógicos das variáveis, não se acrescenta aqui nenhuma nova classe de valorações. As razões por que

Carnap coloca predicados e funtores *descritivos* entre as expressões valoráveis – uma escolha que a princípio pode parecer estranha – só ficarão completamente claras quando atingirmos a definição de analiticidade propriamente dita.

Dessas quatro classes de expressões – as expressões numéricas, a sentença Ŋ, as variáveis e os predicados/funtores descritivos – apenas as duas últimas podem receber diferentes valorações. As expressões numéricas, bem como Ŋ, recebem sempre a si próprias como valoração; para cada tipo de variável, no entanto, e para cada tipo de predicado/funtor descritivo, há toda uma classe de possíveis valorações. Cabe aqui enfatizar que a valoração, das variáveis de uma sentença, só é feita para variáveis *livres*. Dentro de uma sentença, essas variáveis livres podem receber qualquer uma das possíveis valorações de seu específico tipo lógico. Lembramos também que uma sentença reduzida C^r possui – conforme ressaltamos logo acima – a forma sintática [...] C_L, em que C_L possui somente variáveis livres. Encontramos aí uma das razões para Carnap, por meio das regras de redução, colocar todas as quantificações no início da sentença. Assim, todas as variáveis da sentença C_L podem ser valoradas (as quantificações podem ser deixadas para um momento posterior, em que receberão o tratamento adequado).

Há ainda, para encerrar nossa lista, uma última classe de expressões sintáticas que recebem valoração. Sua valoração dentro de uma sentença, no entanto, é diretamente determinada pela valoração escolhida para outras expressões, e não pode ser livremente escolhida dentro da classe de valorações possíveis para o seu tipo lógico. Trata-se das expressões sintáticas construídas justamente a partir de outras expressões sintáticas valoráveis. Sua valoração, como não poderia deixar de ser, dependerá da valoração das expressões a partir da qual são construídas.

Finalmente, Carnap fornece regras de avaliação. De maneira direta, essas regras aplicam-se somente a sentenças de forma sintática muito simples. Em primeiro lugar, sentenças da forma “P(A)”, em que P é um predicado ou uma variável para predicado (de um tipo lógico qualquer), e A um argumento do tipo lógico adequado. Em segundo lugar, sentenças da forma “$U_1 = U_2$”, em que tanto “U_1” como “U_2” são expressões valoráveis.

Essas duas formas sintáticas, contudo, representam as duas formas básicas das sentenças de uma linguagem como a linguagem II (que, nesse particular, segue o esquema mais comum adotado em lógica formal). Elas compõem as duas formas lógicas normalmente admitidas para as sentenças atômicas, a partir das quais todas as outras sentenças da linguagem são formadas, por meio de conectivos lógicos e de quantificação.

Ao fornecer regras para a avaliação dessas sentenças, portanto, Carnap fornece as regras de avaliação para todas as sentenças atômicas da linguagem. Mas o que é uma avaliação? Uma avaliação é uma operação que transforma essas sentenças atômicas em uma das duas seguintes sentenças: $\mathfrak{N}$, ou $\sim\mathfrak{N}$. Vemos aqui que o esquema proposto por Carnap começa a ganhar forma. Essas duas sentenças representam os dois modelos para as sentenças analíticas ($\mathfrak{N}$) e contraditórias ($\sim\mathfrak{N}$).

O método da avaliação é baseado nas valorações. Ele não diz, incondicionalmente, se uma sentença atômica deve ser transformada em $\mathfrak{N}$ ou $\sim\mathfrak{N}$. Nem isso seria possível, na medida em que existem, entre as sentenças atômicas, sentenças abertas com variáveis livres. Por exemplo: o que dizer que uma sentença como “x = y”, em que tanto x como y são variáveis numéricas? Obviamente, não deve ser possível decidir com relação ao *status* dessa sentença (a não ser que se considere, implicitamente, o seu fecho universal; mas essa não é a única opção possível de quantificação sobre as variáveis livres dessa sentença; Carnap, é claro, prefere guardar para uma outra etapa – a etapa final – o tratamento da quantificação).

O método fornecido por Carnap transforma uma sentença atômica em $\mathfrak{N}$ ou $\sim\mathfrak{N}$, *em vista de determinadas valorações para todas as suas expressões valoráveis*. Sua forma geral é: Dadas tais e tais valorações, para todos os elementos valoráveis de uma sentença atômica, então ela deve ser transformada em $\mathfrak{N}$, ou em $\sim\mathfrak{N}$.

Tomemos agora, novamente, uma sentença reduzida C^r. Sua forma, como sabemos, é [...] C_L, em que C_L é uma sentença que contém apenas variáveis livres (uma sentença sem quantificação). Se considerarmos C_L, perceberemos que ela tem de ser composta por uma série de sentenças atômicas, ligadas por conectivos lógicos[97] (mas sem nenhuma quantificação). Se forem escolhidas valorações para todos as expressões valoráveis da sentença C_L, então poderemos avaliar todas as (sub-)sentenças atômicas de que C_L é

[97] Estamos incluindo a negação entre os conectivos. Na forma reduzida de uma sentença aparecem, na verdade, somente três conectivos: “não” (ou “~”, de acordo com o simbolismo utilizado por Carnap para a linguagem II), “e” (ou “•”) e “ou” (ou “^”).

composta. Cada uma dessas (sub-)sentenças atômicas será transformada, em vista daquele específico conjunto de valorações, em Ŋ ou ~Ŋ. Como resultado, teremos uma série de sentenças Ŋ e ~Ŋ ligadas entre si por conectivos lógicos. As regras de redução (que tratam, conforme já observamos, da redução de sentenças nas quais aparece Ŋ) devem agora ser novamente aplicadas, para transformar C_L, como um todo, em Ŋ ou ~Ŋ. Por exemplo: Se C_L tem a forma "C_1 e C_2", em que tanto C_1 como C_2 são sentenças atômicas; e se, para certa valoração das expressões de C_L, C_1 é transformada em Ŋ, e C_2 é transformada ~Ŋ; então C_L é transformada (após a avaliação), na sentença "Ŋ e ~Ŋ". As regras de redução, finalmente, poderão ser novamente aplicadas para reduzir essa sentença para ~Ŋ. O resultado global é que C_L é transformada, por essa avaliação, em Ŋ. Carnap fornece esse resultado por meio de um teorema:

> *"**Theorem 34c.1.** Let C1 be a reduced sentence without operators. The evaluation of C1, on the basis of any valuations for the **b*** [as expressões valoráveis] *which occur, leads in every case, in a finite number of steps, to the final result; this is either Ŋ or ~Ŋ."* [98] (colchetes meus)

Chegamos, portanto, à seguinte situação: Dada uma sentença C, obtemos a reduzida C^r; essa reduzida C^r possui a forma "[...] C_L", em que C_L é uma sentença sem quantificação; para certa escolha de valorações para todas as expressões valoráveis de C_L, C_L é transformada em Ŋ ou ~Ŋ, por meio das regras de avaliação. É a partir desse ponto que Carnap irá oferecer sua definição de analiticidade.

VII

Conforme deve ter ficado claro após as explicações da seção anterior, resta apenas um problema que Carnap precisa resolver para poder fornecer uma definição de analiticidade para todas as sentenças da linguagem II: ele precisa encontrar uma maneira de tratar as quantificações que, após a redução de uma sentença, foram todas posicionadas no início da sentença reduzida. Vejamos como ele aborda essa questão.

[98] *SLL*, pág. 110.

Dada uma sentença C, já sabemos que a sua reduzida C^r possui a forma “[...] C_L”. Sabemos também que a sentença C_L, sem nenhuma quantificação, será avaliada como Ŋ ou como ~Ŋ, dependendo das valorações que se façam para sua expressões. Em outras palavras: para cada conjunto de valorações (uma valoração para cada expressão valorável da sentença), teremos uma avaliação de C_L, como Ŋ ou ~Ŋ. As expressões valoráveis, por excelência, são as variáveis livres de C_L; e as quantificações, antepostas a C_L, atuam justamente sobre essas variáveis livres. Com a situação assim descrita, não é difícil ver qual a solução encontrada por Carnap: A quantificação deverá atuar sobre as valorações. Examinemos mais detalhadamente essa idéia.

Podemos deixar momentaneamente de lado os predicados e funtores descritivos (eles aparecem em um tipo bastante peculiar de sentença, cujo tratamento pode ser feito separadamente, e sequer aparecem necessariamente como parte da linguagem II[99]; por isso, o melhor é retomá-los um pouco mais à frente, quando o restante da discussão já estiver mais claro). As variáveis livres de uma sentença, então, são as únicas expressões da sentença que podem ser livremente valoradas (ou seja, que podem receber qualquer uma das diferentes valorações possíveis para o seu tipo lógico). Todas as outras expressões passíveis de valoração, ou possuem apenas uma valoração possível, ou têm sua valoração estritamente determinada pela valoração das variáveis.

A sentença C_L não pode receber diretamente um *status* sintático (algo que corresponde, no esquema de Carnap, a um valor de verdade) justamente porque é uma sentença aberta (possui variáveis livres); isso significa que, estritamente falando, ela sequer chega a ser uma sentença, na medida em que não faz nenhuma afirmação[100]. Nesse sentido, as diversas valorações possíveis para as expressões de uma sentença aberta correspondem às diversas determinações possíveis para essa sentença. Tomemos como exemplo um caso bastante simples, em que C_L é a sentença (aberta) “x = y” (os símbolos “x” e “y”, aqui, indicam variáveis numéricas). Se valorarmos x pela expressão numérica

[99] A linguagem II possui um aparato simbólico básico de natureza essencialmente lógica, que não inclui predicados ou funtores descritivos. O que acontece é que, a esse aparato simbólico básico, podem ser *acrescidos* predicados e funtores descritivos.
[100] Na verdade, Carnap irá considerar essas sentenças como equivalentes ao seu fecho universal, ou seja, a sentença obtida pela quantificação universal de todas as variáveis livres. Isso significa que Carnap irá sim fornecer um *status* sintático para essas sentenças abertas (como analíticas, contraditórias ou sintéticas). Porém, isso só será feito por meio da consideração de uma sentença fechada que, por uma escolha interpretativa, é associada a ela.

“ 0’’ ” (V_x = 0’’) e y pela mesma expressão numérica “ 0’’ ” (V_y = 0’’), então a sentença aberta C_L fica “determinada”, *em vista dessas valorações*, como a sentença “ 0’’ = 0’’ ”. Não é à toa que, nesse caso, C_L é avaliada como Ŋ (modelo de sentença analítica). Por outro lado, se fizermos V_x = 0’’, e V_y = 0’’’, então C_L fica “determinada” como a sentença “ 0’’ = 0’’’ ”, cuja avaliação será ~Ŋ (modelo se sentença contraditória).

A sentença C^r, para a qual nossa atenção deve finalmente se dirigir, é a sentença fechada na qual todas as variáveis livres de C_L estão devidamente quantificadas. Como determinar seu *status* sintático? Já sabemos avaliar C_L em vista de cada conjunto possível de valorações. Trata-se agora de ver como as quantificações agem sobre as valorações. A solução, mais uma vez, é mais simples do que o aparato lógico da definição deixa transparecer. Sua idéia básica ganhará contornos nítidos se levarmos adiante o exemplo do parágrafo anterior. Suponhamos que C^r seja a sentença “(x) (y) (x = y)”, ou seja, a sentença “Para todo x, e para todo y, x = y” [101]. Essa sentença será analítica (segundo a concepção bastante razoável de Carnap) somente se a sentença aberta “x = y” for avaliada como Ŋ para toda valoração possível da expressão x, e para toda valoração possível da expressão y. Lembramos que as valorações de x e y representam os valores que essas variáveis podem assumir. A sentença “(x) (y) (x = y)” afirma, justamente, que x = y para todos os valores de x e de y. A definição de Carnap apenas traduz essa situação: “(x) (y) (x = y)” será analítica se “x = y” for avaliada como Ŋ para toda valoração de x e de y.

Vemos assim que, no esquema de Carnap, as expressões “para todo x” e “para todo y” – que são as quantificações originais presentes na sentença C^r – são transformadas, para efeitos da definição de analiticidade, em: “para toda *valoração* de x” e “para toda *valoração* de y”. (No caso do nosso exemplo, não é difícil perceber que a condição apresentada não se verifica. De fato, não é para toda valoração de x, e para toda valoração de y, que “x = y” á avaliada como Ŋ. No final do parágrafo anterior, indicamos um par de valorações que mostra bem esse fato.)

Estamos agora em condições de oferecer, de maneira concisa, a definição de analiticidade elaborada por Carnap na seção 34d de *SLL*. Essa seção contém nada menos do que 13 regras. Parte dessa abundância explica-se pela necessidade de tratar todos os casos

[101] Seguimos aqui a notação adotada por Carnap, em que os parênteses ao redor de uma variável indicam a quantificação universal; a quantificação existencial fica indicada pelo símbolo “∃”.

possíveis. Assim, Carnap indica que uma sentença é analítica (contraditória) se sua reduzida for analítica (contraditória); que uma sentença aberta é analítica (contraditória) se seu fecho universal for uma sentença analítica (contraditória); etc. São assuntos que já abordamos em seções anteriores (ao menos os casos de maior relevância).

Mais importante é a complicação resultante da necessidade de levar em conta diversas quantificações em uma mesma sentença. Devemos recordar: Após a aplicação das regras de redução, uma sentença C fica sob a forma "[...] C_L". Aqui, a notação "[...]" indica uma seqüência de quantificações, vale dizer, todas as quantificações irrestritas que apareciam na sentença original, e que foram trazidas para o início da sentença reduzida. Para lidar com essa seqüência de quantificações, a solução adotada por Carnap, como é comum nesse tipo de situação, faz uso – mais uma vez – de um método recursivo. Para entender o que se passa, portanto, precisamos começar pelo caso mais simples – o caso em que há apenas uma única quantificação anteposta a C_L[102] – e indicar como proceder para os casos mais complicados. Lembramos ainda que a linguagem II possui dois tipos distintos de quantificação que precisam ser considerados: a quantificação universal (que vem nos servindo mais freqüentemente como exemplo) e a quantificação existencial. Podemos passar, finalmente, às definições relevantes.

Seja uma sentença C. Iniciamos com o caso em que aparece, na sentença reduzida C^r (que tem a forma "[...] C_L"), apenas uma única quantificação precedendo a sentença aberta C_L. Chamemos de "A" a variável (de qualquer tipo lógico) sobre a qual essa quantificação opera. Há dois casos possíveis: 1) A quantificação é universal; e 2) a quantificação é existencial. Se a quantificação sobre a variável A for universal, defini-se:

[102] O caso em que não há nenhuma quantificação é tratado de maneira isolada, e podemos reservá-lo para esta nota. Lembramos que, se a sentença reduzida C^r for aberta, deve-se tomar (segundo a definição de analiticidade) o seu fecho universal. Por esse motivo, se uma sentença reduzida não possui quantificadores, é porque não possui variáveis.
Lembramos também que, neste ponto, ainda não estamos considerando as sentenças em que aparecem predicados e funtores descritivos, de que trataremos mais à frente. Na linguagem II, a divisão entre termos lógicos e descritivos é dada desde o início, de maneira extremamente simples: todos os termos básicos da linguagem (seu aparato simbólico original) são lógicos; os termos descritivos (predicados e funtores) devem ser explicitamente acrescentados.
As duas considerações anteriores servem para mostrar que uma sentença reduzida sem quantificadores possuiria a única forma possível de uma igualdade entre expressões numéricas. Pelas próprias regras de redução, no entanto, uma igualdade desse tipo deve ser reduzida para $\mathfrak{N}$ (no caso de uma igualdade verdadeira, tomada aqui como igualdade metateórica entre certas expressões simbólicas, as expressões numéricas) ou $\sim\mathfrak{N}$ (no caso de um igualdade falsa). A definição de analiticidade, então, torna-se trivial: É analítica a sentença cuja reduzida é $\mathfrak{N}$; é contraditória a sentença cuja reduzida é $\sim\mathfrak{N}$.

1a) A sentença C é analítica se C_L for avaliada como Ŋ para todas as valorações possíveis de A; e 1b) A sentença C é contraditória se C_L for avaliada como ~Ŋ para ao menos uma valoração possível para A. No outro caso, se a quantificação sobre A for existencial, temos: 2a) A sentença C é analítica se C_L for avaliada como Ŋ para ao menos uma valoração possível de A; e 2b) A sentença C é contraditória se C_L for avaliada como ~Ŋ para todas as valorações possíveis de A.

Se considerarmos o que significam as valorações e avaliações (de acordo com o que explicamos anteriormente), veremos que essas definições são bastante sensatas. Carnap deseja chamar uma sentença universal de "analítica" se ela for "válida" para "todos os valores possíveis da variável". Em sua definição, a idéia expressa informalmente por "todos os valores possíveis da variável" é traduzida por "todas as valorações possíveis para a variável"; e a idéia expressa informalmente pelo adjetivo "válida"[103] é traduzida formalmente pelas regras de avaliação (uma sentença é "válida para certo valor da variável" se for avaliada como Ŋ para certa valoração da variável).

Devemos agora considerar o caso em que a sentença reduzida apresenta mais de uma quantificação. Como dissemos, a definição de analiticidade será desenvolvida de maneira recursiva. No caso, a recursão será feita de fora para dentro sobre os operadores de quantificação. Tomemos uma sentença C cuja reduzida apresente n+1 quantificações; a reduzida, portanto, terá a forma: "[v_1] [v_2] ... [v_{n+1}] C_L"[104]. A analiticidade dessa sentença C é definida pelas seguintes cláusulas. Se a quantificação sobre a variável v_1 for universal, então: 1a') A sentença C é analítica se a sentença "[v_2] ... [v_{n+1}] C_L" for analítica para todas as valorações possíveis de v_1; e 1b') a sentença C é contraditória se a sentença "[v_2] ... [v_{n+1}] C_L" for contraditória para ao menos uma valoração possível de v_1. Por outro lado, se a quantificação sobre a variável v_1 for existencial, então: 2a') A sentença C é analítica se a sentença "[v_2] ... [v_{n+1}] C_L" for analítica para ao menos uma valoração possível de v_1; e 2b') a sentença C é contraditória se a sentença "[v_2] ... [v_{n+1}] C_L" for contraditória para todas as valorações possíveis de v_1.

[103] Não confundir com o conceito – *formal* – de "sentença válida" da sintaxe geral. Aqui estamos usando o adjetivo "válido", por falta de um outro adjetivo menos problemático, de maneira informal.
[104] Aqui, os colchetes indicam uma quantificação qualquer, seja universal ou existencial.

Vemos que, pelas quatro cláusulas acima, o *status* analítico ou contraditório da sentença C fica definido em função do *status* analítico ou contraditório da sentença "[v_2] ... [v_{n+1}] C_L". Essa sentença "[v_2] ... [v_{n+1}] C_L", por sua vez, possui apenas n quantificações. Dessa maneira, conseguimos definir o conceito de analiticidade para uma sentença com n+1 quantificações com base na definição do conceito de analiticidade de uma sentença com n quantificações. Como nós já oferecemos a definição de analiticidade para sentenças com apenas uma quantificação (e até, como caso especial[105], com nenhuma quantificação), a definição está completa para sentenças com qualquer número natural finito de quantificações. Como, na linguagem II, as sentenças são sempre seqüências finitas de símbolos, a definição está completa para qualquer sentença da linguagem II.

VIII

Até agora, expusemos as condições a que uma sentença deve obedecer para ser dita analítica (casos que indicamos por 1a e 2a na seção anterior), ou contraditória (casos 1b e 2b). Deve-se agora perguntar: É necessário que todas as sentenças da linguagem II sejam analíticas ou contraditórias? Em outras palavras: É necessário que todas as sentenças da linguagem II sejam L-determinadas?

Em princípio, não. Para cada tipo de quantificação, demos uma condição a que a sentença deve obedecer para ser analítica, e uma outra condição a que deve obedecer para ser contraditória. Nada indica que, dada uma sentença qualquer da linguagem II, ela tenha de obedecer necessariamente a uma entre essas duas condições (essas condições não são a simples negação uma da outra). Uma sentença que não obedeça a nenhuma dessas condições, como é perfeitamente possível, será dita sintética.

Acontece que, até agora, estivemos trabalhando somente com a parte "lógica" da linguagem II. A utilização do adjetivo "lógico", aqui, não está ainda totalmente justificada: por enquanto, podemos dizer que ele foi introduzido por Carnap, arbitrariamente, no momento em que ofereceu as regras de formação para a linguagem II. Ali, Carnap estabeleceu que todos os símbolos básicos da linguagem II devem ser vistos como símbolos lógicos; e as sentenças compostas por eles, como sentenças lógicas. É com esses símbolos e

[105] Ver nota 102 acima.

sentenças que, até a presente seção, nós nos temos preocupado. Após oferecer sua definição de analiticidade, contudo, Carnap expõe um teorema – na seção 34e – que justifica plenamente a adoção do adjetivo “lógico” para todo esse aparato simbólico fundamental da linguagem II. Ei-lo:

> *“**Theorem 34e.11.** Every logical sentence is L-determinate, that is to say it is either analytic or contradictory.”* [106]

Esse teorema significa uma resposta positiva à pergunta com que iniciamos a seção: Sim, dada uma sentença da linguagem II (uma sentença composta com os símbolos básicos, ou primitivos, da linguagem II), ela será ou analítica, ou contraditória. Não é difícil ver as razões por que isso ocorre. Lembremos que, dada uma sentença C, ela é reduzida para a forma “[...] C_L”, em que C_L não possui quantificação. No caso da quantificação universal, as condições estabelecidas para uma sentença ser L-determinada (analítica ou contraditória) eram: 1a) analítica se C_L for avaliada como Ŋ para todas as valorações possíveis de A; e 1b) contraditória se C_L for avaliada como ~Ŋ para ao menos uma valoração possível para A. A conclusão é imediata: para que uma sentença não seja nem analítica, nem contraditória, é necessário que, para alguma valoração de suas expressões, C_L não seja avaliada nem como Ŋ, nem como ~Ŋ[107]. Das regras de avaliação, no entanto, sabemos que uma sentença sem quantificação é sempre avaliada como Ŋ ou ~Ŋ (ver teorema 34c.1, reproduzido na seção IV acima).

Podemos então dizer que a linguagem II não possui sentenças sintéticas? Sim e não. Até agora, como dissemos, estivemos lidando exclusivamente com o aparato simbólico básico da linguagem II. A esse aparato simbólico, porém, podem ser acrescentados predicados e funtores descritivos. Trata-se dos típicos predicados e funtores utilizados para descrever domínios empíricos de conhecimento, como a física, e cujos valores (no caso de funtores) ou valores de verdade (no caso de predicados) são estabelecidos de maneira externa à linguagem.

[106] *SLL*, pág. 116.
[107] De fato, se C_L for avaliada sempre como Ŋ ou ~Ŋ, então ou bem aparece alguma avaliação ~Ŋ (caso em que C é contraditória), ou todas as avaliações são Ŋ (caso em que C é analítica).

De maneira geral, podemos dizer que os recursos básicos da linguagem II são adequados a lidar, fundamentalmente, com proposições matemáticas. Essas proposições, na linguagem II, aparecerão sempre como proposições lógicas L-determinadas – e é nesse sentido que podemos dizer que não aparecem sentenças sintéticas na linguagem II[108]. No entanto, podem ser acrescentados à linguagem II predicados e funtores descritivos, introduzidos especialmente para permitir a descrição de algum domínio científico específico. As sentenças em que tais predicados e funtores aparecem é que serão, eventualmente, sintéticas. (Caso não sejam introduzidos predicados e funtores descritivos, a linguagem II é mesmo uma linguagem puramente lógica, em que todas as sentenças são determinadas).

O tratamento que Carnap confere às sentenças descritivas (aquelas nas quais aparece algum termo descritivo) espelha o tratamento dado na sintaxe geral. Em geral, sentenças descritivas serão sintéticas, na medida justamente em que não é possível determinar seu *status* somente com base nos recursos sintáticos internos à linguagem, seu valor de verdade dependendo diretamente de alguma observação (ou decisão) exterior à linguagem. Não obstante, é possível que algumas sentenças descritivas, por possuírem certa forma especial, sejam sim analíticas ou contraditórias. Suponhamos que o predicado "Vermelho" (de um tipo lógico qualquer) seja um predicado descritivo introduzido na linguagem II. Em geral, a sentença "Vermelho(A)" – em que A é um argumento (variável ou constante) do tipo lógico adequado – será sintética. Porém, uma sentença como "Vermelho(C) ou não-Vermelho(C)" (para facilidade, estamos tomando C como uma constante) deverá certamente ser considerada analítica. Já uma sentença como "Vermelho(C) e não-Vermelho(C)" deverá ser considerada contraditória. Como Carnap trata essas situações?

Carnap fornece uma definição de analiticidade e uma definição de contraditoriedade específica para sentenças em que aparece algum termo descritivo (qualquer termo que não esteja, em princípio, entre os termos primitivos da linguagem[109]). Comecemos pela

[108] Vemos aqui a coincidência que há, na classificação dos termos da linguagem II, com o esquema conceitual que será estabelecido pela sintaxe geral. Na sintaxe geral, com efeito, os termos lógicos ficavam definidos como aqueles termos que, combinados entre si, resultavam sempre em sentenças determinadas. É exatamente o que acontece com o aparelhamento simbólico básico da linguagem II.

[109] É sempre possível que um termo introduzido, em princípio tido como descritivo, possa revelar-se um termo lógico: isso acontecerá se todas as sentenças em que ele aparecer, combinado com outros termos

definição de analiticidade. Uma sentença descritiva será analítica se a sentença que resultar da substituição de todos os seus termos descritivos por variáveis de tipo lógico adequado (uma mesma variável para ocorrências distintas do mesmo termo; variáveis diferentes para termos diferentes) for analítica. Não há muito mistério por trás dessa definição. A condição estabelecida significa que os termos descritivos da sentença, de certa forma, eram supérfluos, já que a sentença permanece válida para quaisquer outros termos que se possa colocar em seus lugares (é isso o que significa a sua substituição por variáveis).

A definição para as sentenças descritivas contraditórias também não é complicada, mas requer um pouco mais de atenção. Recordamos, logo de cara, que os predicados e funtores lógicos também estão entre as expressões sintáticas que podem receber valoração. Uma sentença descritiva será contraditória, então, *se para qualquer valoração de seus termos*, a sentença resultante for contraditória. As valorações, já sabemos, representam todos os possíveis valores que certo termo é capaz de assumir, de acordo com seu tipo lógico. Após a valoração dos termos descritivos de uma sentença descritiva, o que sobra é uma sentença lógica, que deve ser analítica ou contraditória. A definição acima estabelece, portanto, que uma sentença descritiva é contraditória quando não há valoração possível para seus termos descritivos que torne analítica a sentença lógica resultante. Em outras palavras: Uma sentença descritiva é contraditória quando, independentemente dos valores que assumam os termos descritivos, ela fará sempre uma afirmação logicamente contraditória.

Perguntamos agora: Diante dessas definições, uma sentença descritiva necessita ser sempre analítica ou contraditória? Ou: Dada uma sentença descritiva, ela precisa necessariamente encaixar-se em uma dessas duas definições? É fácil ver que não. Uma sentença descritiva, ao contrário das sentenças lógicas, pode escapar tanto à definição para sentença analítica como à definição para sentença contraditória. De fato, pode acontecer – e é o que normalmente deveria acontecer – que uma sentença descritiva torne-se analítica para certa valoração de seus termos descritivos, e contraditória para outra valoração dos termos descritivos. Nesse caso, ela não é analítica, pois a substituição dos termos descritivos por variáveis não resulta em uma sentença analítica, e sim em uma sentença

lógicos, resultar determinada. A esse respeito, ver a definição de termos lógicos e descritivos da sintaxe geral, que nós expusemos na seção VI do Capítulo 3.

contraditória[110]. Mas também não é contraditória, pois existe uma valoração para a qual a sentença lógica resultante é analítica.

IX

Até aqui, vimos como Carnap define, para a linguagem II, o conceito de "sentença analítica", de "sentença contraditória" e "de sentença sintética". A partir desses conceitos é que ele define, na seção 34f, o conceito de "conseqüência" para a linguagem II. Explicaremos muito brevemente a sua construção, que não é complicada.

Suponhamos – de maneira ainda informal – que uma sentença C seja considerada como conseqüência de um conjunto K de sentenças. Podemos imaginar, para efeitos desta explicação preliminar, uma linguagem normal de comunicação, não-formalizada, como por exemplo o português. Em qualquer linguagem desse tipo, se considerarmos que uma sentença C decorre de certas outras sentenças (o conjunto K de sentenças), é comum supor que essas outras sentenças não sejam compatíveis com a negação de C. Em outras palavras: Se a partir de K, eu deduzo C, então não posso considerar K como compatível com a negação de C. É esse o esquema informal que Carnap busca captar por meio de uma definição formal. Eis a passagem relevante de *SLL*:

> *"A sentence is (in material interpretation) a logical consequence of certain other sentences if, and only if, its antithesis is incompatible with these sentences. Hence we define as follows:* C_1 *is called a* **consequence** *of* K_1 *in II, if* $K_1 + \{\sim(\ )\ (C_1)\}$ *is contradictory."* [111] (destaque do autor)

Algumas observações são aqui necessárias. Em primeiro lugar, observamos que a notação "$\sim(\ )\ (C_1)$" indica a negação do fecho universal da sentença C_1, a qual pode ser aberta[112]. Em segundo lugar – e mais importante – chamamos a atenção para o fato de que, na definição acima, o conceito de conseqüência é definido a partir do conceito de *conjunto*

[110] Recordamos que, após a substituição dos termos descritivos por variáveis, deve-se tomar o fecho universal da nova sentença.
[111] *SLL*, pág. 117.
[112] Conforme já observamos, uma sentença aberta é sempre interpretada como sendo equivalente ao seu fecho universal.

contraditório, e não do conceito de sentença contraditória, que foi (este último) o conceito exposto nas seções anteriores (a expressão "$K_1 + \{\sim(\ )\ (C_1)\}$" indica um conjunto de sentenças, isto é, o conjunto K_1 de sentenças, acrescido da sentença "$\sim(\ )\ (C_1)$").

De fato, nas seções anteriores, para simplificar nossa exposição, nós optamos por tratar somente da definição de analiticidade para sentenças. Na seção 34d de *SLL*, porém, Carnap fornece a definição de analiticidade para sentenças e para conjuntos de sentenças, de maneira integrada. Na verdade, a analiticidade de sentenças é definida como um caso particular da analiticidade de conjuntos: C_1 é analítica se o conjunto unitário $\{C_1\}$ for analítico. Isso, contudo, não precisa nos preocupar, pois a relação entre esses conceitos é previsível.

No caso de possuir apenas sentenças lógicas, um conjunto será contraditório se uma dessas sentenças for contraditória. Se, por outro lado, aparecerem sentenças descritivas no conjunto, a situação torna-se um pouco mais complicada. Deve-se levar em conta o caso de conjuntos como, por exemplo, {Vermelho(C), não-Vermelho(C)}. Embora esse conjunto seja claramente contraditório, nenhuma de suas sentenças, isoladamente considerada, é contraditória (ambas são sintéticas; no caso de sentenças lógicas – que como sabemos são sempre determinadas – ao menos uma teria de ser contraditória).

Essa situação, porém, pode ser facilmente resolvida, e de mais de uma maneira. Se o conjunto for finito, por exemplo, seria possível tomar a junção lógica de todas as suas sentenças. O conjunto seria contraditório se a sentença obtida pela junção (produto lógico) de todas as suas sentenças fosse contraditória. Para o caso geral, em que o conjunto é potencialmente infinito, Carnap dá a definição em termos de valorações. O conjunto será contraditório se, para qualquer valoração de todos os termos descritivos presentes em suas sentenças, ao menos uma sentença for contraditória. A idéia é a mesma já exposta anteriormente, quando explicamos a definição de sentença descritiva contraditória.

Chegamos, assim, ao seguinte ponto: Carnap fornece, para a linguagem II, uma estrutura dedutiva forte, que representa uma extensão da estrutura dedutiva fraca exposta na seção 31 de *SLL*, por meio de regras definidas de inferência. O caminho que ele faz, contudo, não é aquele descrito na sintaxe geral. Em vez de iniciar pela elaboração de um conceito indefinido de "conseqüência direta", a partir do qual a analiticidade de sentenças –

bem como toda a estrutura dedutiva da linguagem – ficaria devidamente determinada, ele faz justamente o oposto. Constrói diretamente – e de forma bastante laboriosa, conforme pudemos verificar – uma definição de analiticidade, para depois, com base nela, obter uma definição para o conceito de "conseqüência".

A definição de analiticidade construída por Carnap para a linguagem II apresenta diversas peculiaridades. Ela pode ser considerada, em certo sentido, o ponto alto de *SLL*, na medida em que anuncia uma nova teoria lógica (a teoria semântica) que Carnap ele próprio, infelizmente, não soube apreciar devidamente. Por outro lado, a construção dessa definição permanece envolta em algumas obscuridades. Por que Carnap teria desviado, nesse exato ponto, do esquema da sintaxe geral?

Algumas questões, nesse sentido, devem ser abordadas. A definição de analiticidade para a linguagem II pode ser considerada como uma definição sintática? Por que não? O tratamento oferecido por Carnap para a estrutura dedutiva da linguagem II é ao menos compatível com o esquema da sintaxe geral? Qual o objetivo de Carnap ao desenvolver o conceito de analiticidade para a linguagem II? No Capítulo 5, iremos abordar precisamente esses assuntos.

Capítulo 5: Verdade, matemática e o Princípio de Tolerância

I

Ao desenvolver o conceito de analiticidade para a linguagem II, a preocupação central de Carnap reside na questão da *verdade matemática*. Essa afirmação, embora aparentemente simples, necessita diversos esclarecimentos. Tantos esclarecimentos, de fato, que dedicaremos este capítulo inteiro a discuti-la – em todos os seus detalhes e desdobramentos. Por esse caminho, poderemos chegar ao cerne de alguns dos principais problemas interpretativos de *SLL*.

Por um lado, é bastante óbvio que, ao desenvolver o conceito de analiticidade para a linguagem II, Carnap está lidando com o antigo problema das proposições matemáticas e do *status* que elas devem receber. Por outro lado, é necessário compreender as razões que levam Carnap a recusar, veementemente, o conceito de "verdade" como um conceito logicamente admissível, ao mesmo tempo em que trabalha – conforme veremos – sob sua influência ineludível e poderosa.

De maneira geral, pode-se dizer que há uma constante tensão, fundamental para a compreensão de *SLL*, entre a noção sintática de analiticidade – solução oferecida por Carnap para os problemas teóricos que enfrenta – e a noção – que Carnap considera como não-sintática, e portanto intratável do ponto de vista lógico – de verdade. Essa tensão perpassa todo o livro. Mais ainda, Carnap não poderá livrar-se dela, da maneira como desejaria, justamente devido à necessidade que sente de oferecer um tratamento adequado para as questões da matemática. É em relação à matemática, de fato, que o conceito de "verdade" revela toda a sua complexidade: em *SLL*, ele assumirá diferentes significados, cujas mutações e relações de interdependência não são fáceis de se seguir, e não consegue, em última instância, ser totalmente descartado.

Em nenhum momento da obra de Carnap, portanto, essa tensão de que estamos falando – entre o conceito de "analiticidade" e o de "verdade" – torna-se tão evidente quanto na definição de analiticidade para a linguagem II. O problema da verdade matemática surge ali em toda a sua amplitude; e recebe, fruto que é de tantas complicações, um tratamento bastante inovador. É essa a situação que vamos agora explorar, em todos os seus detalhes.

A questão das proposições matemáticas, como se sabe, é absolutamente central para o pensamento logicista. E Carnap é – para dizer o mínimo – herdeiro direto dessa escola de pensamento. Só por isso, o tema já estaria fadado a ocupar um lugar de destaque em *SLL*. Há, contudo, uma segunda força, um segundo e novo impulso teórico, que impelia Carnap nessa direção: o recém-publicado teorema da incompletude de Gödel (o primeiro teorema de incompletude). De certa maneira, esse teorema representa uma das fontes que forçará o problema da *verdade* matemática – e não outro – sobre o trabalho de Carnap.

Um dos maiores méritos do teorema de Gödel, com efeito, é expor a questão da verdade matemática em toda a sua extensão. Pode-se até mesmo dizer que coube aos resultados de Gödel retirar esse problema – fundamental como é – da sombra em que ele havia sido colocado pelos desenvolvimentos da lógica simbólica e formal no final do século XIX e, principalmente, no início do século XX. Carnap esteve entre os pensadores que mais rapidamente compreendeu o significado profundo do teorema de incompletude para a lógica, e sentiu-se compelido a enfrentar as difíceis questões que ele propunha. A esse respeito, Coffa observa:

> *"It is ironic that Carnap´s syntactical philosophy is sometimes thought to be refuted by Gödel's discoveries which, we are told, establish the need to go beyond syntax. In fact, Gödel's discoveries were the decisive factor in determining both the technical problems that Carnap faced and his solutions for them. Far from having been written in ignorance of Gödel's results, Carnap's LSL was inspired by an appreciation of the significance of Gödel´s work that only a handful of logicians could match at the time. (...)"* [113]

O comentário de Coffa não poderia ser mais exato. Em *SLL*, Carnap discute extensamente o teorema de incompletude de Gödel, e oferece para ele uma formulação que supera em muito – em termos de clareza, abrangência e profundidade de compreensão –

[113] Coffa, 1987, p. 548.

qualquer texto que houvesse sido publicado, até aquele momento, a respeito do assunto. No Capítulo 2 acima, tivemos ocasião de abordar brevemente a questão da influência exercida pelo trabalho de Gödel em *SLL* (ver Capítulo 2, seção III). Devemos agora retomar essa discussão, de um ponto de vista mais amplo, e verificar exatamente qual o enigma *proposto* (não aquele resolvido) pelos resultados de Gödel. O melhor, portanto, é iniciar nossa indagação por um breve exame do famoso teorema.

II

Essencialmente, o teorema de incompletude de Gödel estabelece que nenhum conjunto de proposições recursivamente determinado pode abranger todas as *verdades da aritmética*.

Por detrás do complicado aparato técnico necessário à sua obtenção, eis aí o cerne – até certo ponto simples de ser compreendido e menos espetacular do que se costuma imaginar – do teorema em questão. Contudo, não devemos deixar a aparente simplicidade da formulação acima nos enganar. Esse núcleo do teorema de Gödel, da maneira como o expusemos, já assinala abertamente a dificuldade central que nele se revela: o problema da verdade aritmética (e da verdade matemática em geral), bem como a correspondente necessidade – sempre renovada em filosofia – de oferecer para ele um conteúdo claro, em vez de tentar contorná-lo. É o fato de trazer esse problema à tona, sob uma nova perspectiva, que determina toda a imensa importância conceitual dos resultados de Gödel para a evolução da Lógica e da matemática no século XX.

De fato, quais são essas "verdades da aritmética" que surgem no teorema de Gödel e que não podem ser exauridas por nenhum sistema formal[114]? A questão aqui é bastante sensível, principalmente em vista do pensamento logicista, até então prevalente. Segundo a proposta dos logicistas, os sistemas formais eram concebidos para captar a noção de prova lógica e de teorema; e essas duas noções deveriam servir justamente para captar a idéia de

[114] Aqui, a expressão "sistema formal" está sendo usada para indicar sistemas aptos a gerar enumerações recursivas de sentenças. Era esse o significado corrente da expressão entre os lógicos daquela época e, em certa medida, até hoje. É esse o significado, além do mais, abrangido pelo teorema de Gödel. No restante deste capítulo, portanto – salvo menção explícita em contrário – usaremos a expressão "sistemas formais" sempre com o enfoque restrito da teoria tradicional, e não com o enfoque alargado que Carnap tenta desenvolver.

"verdade aritmética" (e, um passo mais à frente, de "verdade matemática"). O teorema de Gödel, no entanto, parecia forçar uma ruptura entre qualquer noção formal, de um lado, e a noção de "verdade aritmética", do outro. Torna-se, porém, necessário perguntar: Que noção é essa de "verdade aritmética" – capaz de transcender as "provas" e os "teoremas" – que aparece no teorema de Gödel?

Hintikka expõe da seguinte maneira a situação:

> *"(...) But what Gödel claimed to have proven was that there are (...) arithmetical propositions which are true but which cannot be proved logically in that system* [um sistema axiomático para a aritmética]. *The only assumption Gödel had to make is that the axiom system is not formally inconsistent (...). How can Gödel prove that his crucial proposition is not logically provable by using the very same logic? <u>And how can he know that the proposition in question is true if he cannot prove it?</u>"* [115] (grifo meu)

Em outra passagem, ao comentar o teorema de completude de Gödel, o mesmo Hintikka observa:

> *"(...) In brief, we have to establish both what logic (or a system of logic) actually can do and also what it <u>ought to do</u>.*
>
> *Saying this already steps in many toes. Many a logical Protagoras will tell you that in logic our actual proof methods are the measure of all things. For instance, they claim that the meaning of logical constants like propositional connectives and quantifiers is determined by the rules of proof that govern them. If so, it would make no sense to speak of <u>what our logic ought to do</u> (...)."* [116] (grifos meus)

Vemos, assim, a distinção que surge entre a noção de *prova* lógica, realizada dentro de um sistema lógico formal axiomatizado, e a noção de "verdade" para além da prova formal, ou seja, daquilo que a lógica "deveria fazer", e que a aritmética parece exigir.

[115] Hintikka, 2000, p. 29.
[116] *Op. cit.*, p. 16.

Historicamente, foi precisamente o teorema da incompletude de Gödel que cobrou claramente – ao oferecer um resultado tão exato como inequívoco – o significado essencial da distinção entre "prova" e "verdade".

Que essa distinção, embora por vezes confusamente percebida, permanecia envolta em todo tipo de brumas, demonstra-o a posição que a esse respeito assume um matemático e lógico tão brilhante quanto Herbrand, em texto significativamente escrito em 1930 (pouco anterior, portanto, aos resultados de Gödel). De fato, ao falar do sistema de Russell e Whitehead, ele comenta:

> *"Mais on peut encore se demander s'il n'y a pas des proposition que le système de signes ne saura traduire, ou des <u>démonstrations qui échapperont à ses regles</u>.*
>
> *Il ne faut pas cacher que cela n'est qu'un résultat expérimental (dont la validité ne peut être que confirmée par une dialectique philosophique); sa preuve réside, en somme, dans le fait que Russell et Whitehead ont réussi, dans les trois tomes de* Principia Mathematica*, à reproduire tous les raisonnements des débuts des mathématiques et de la theorie des ensembles. <u>On peut considérer comme um des faits le plus parfaitement vérifiés dans notre connaissance logique du monde que tout raisonnement que peut actuellement faire un mathématicien raisonnable trouve immédiatement sa traduction dans le système de signes étudié</u>."* [117] (grifos meus)

Nessas palavras de Herbrand, encontramos uma exposição clara das contradições a que o pensamento teórico, em matemática, havia sido submetido pelos primeiros logicistas. Pode-se perceber nelas, com toda nitidez, a tensão de que falamos entre a noção de verdade e a noção de prova formal, que Carnap tentará enfrentar em seu novo sistema sintático. Por um lado, Herbrand parece estar ciente da distinção que há entre aquilo que *pode* ser provado dentro de um sistema formal (no caso, o sistema de Russell e Whitehead), e aquilo que, por ser matematicamente verdadeiro – segundo um conceito de verdade anterior a

[117] Herbrand, 1930, p. 36.

qualquer formalização – *deveria poder* ser provado. Essa percepção transparece em sua pergunta pela possível existência de "demonstrações que escapam às regras" dos *Principia*.

Logo a seguir, porém, Herbrand não consegue manter essa distinção em termos claros, e mostra assim como as noções envolvidas permaneciam irremediavelmente confusas. Ao defender a posição de que o sistema dos *Principia* é, de fato, o sistema correto dentro do qual formular a matemática (posição, aliás, que se mostraria tragicamente equivocada), ele não encontra nenhuma maneira adequada de caracterizar esse fato, em termos das sentenças verdadeiras da matemática. Ao contrário, ele prefere falar que o sistema dos *Principia* consegue traduzir "todo raciocínio que um matemático razoável pode fazer", sem esclarecer quais são esses raciocínios, nem o que faria deles raciocínios "razoáveis" (o fato, vale dizer, de conduzirem a sentenças matemáticas verdadeiras, segundo algum sentido independente do termo "verdade"). Em resumo: Embora Herbrand perceba que um sistema matemático formal é concebido para atingir certo propósito, determinado de maneira exterior ao próprio sistema, ele não consegue caracterizar exatamente esse propósito, na medida em que não consegue fornecer uma caracterização independente (independente de esquemas dedutivos) para a classe das sentenças matemáticas verdadeiras.

É Gödel quem abre o caminho para superar esse impasse – ou, antes, mostra que se trata de um impasse. Pois, diante de seus resultados, somos forçados a indagar: Se a noção de "verdade aritmética", pressuposta no teorema de incompletude, ultrapassa a noção de "prova formal" (ou simplesmente não coincide com ela), qual exatamente o conteúdo que lhe pode ser atribuído? A evolução posterior dos estudos lógicos e matemáticos revelou que esse conteúdo é dado pela noção de "verdade", tal como concebida pela teoria de modelos. Nesse sentido, Hintikka esclarece:

> *"... there are several truly remarkable things about Gödel's result* [e Hintikka refere-se aqui ao teorema da completude; a observação, porém, vale igualmente para o teorema de incompletude, que constitui a sua extensão, radicalmente criativa e inovadora, para o âmbito da aritmética]. *The most fundamental one is perhaps the very conceptual distinction between what one can prove in a logic and what should be provable in it.*

What makes this feature remarkable is that the 'should be' idea is what is called model-theoretical *notion, not a* proof-theoretical *one."* [118] (destaques do autor)

O fato importante a reter, porém, é que tal situação não estava clara nem mesmo para o próprio Gödel. Sua brilhante demonstração não dependia de uma análise precisa e rigorosa a respeito do sentido que se deveria atribuir à "verdade" das famosas "proposições indemonstráveis" (indemonstráveis dentro de determinado sistema formal). Tanto assim que coube a outro lógico, Alfred Tarski, desenvolver a definição semântica de verdade que haveria de preencher, com um conteúdo mais claro, os resultados de Gödel (uma teoria de modelos plenamente desenvolvida, porém, só veio a ser elaborada na década de 1950, pelo mesmo Tarski e seus colaboradores). Citamos novamente Coffa, pela clareza de seu comentário:

"(...) Few people realized as clearly as Carnap the extent to which Gödel's (1931) had reactivated an old philosophical problem: what precise sense can we make of the notion of truth involved in Gödel's two major theorems? It is well known that Tarski not only saw the problem but also solved it. It is less widely known that, next to him, Carnap deserves credit for having come closer than anyone else to a solution." [119]

Vemos que em *SLL*, portanto, Carnap está lidando com os mesmos problemas, agudamente apontados pelos resultados de Gödel, que preocupavam Tarski e o próprio Gödel. Mais do que isso, com seu conceito de analiticidade para a linguagem II, Carnap desenvolve um método extremamente semelhante aos expostos por Tarski em seu clássico *Der Wahrheitsbegriff in den formalisierten Sprachen*, de 1935[120]. Na verdade, o método de

---

[118] Hintikka, 2000, p. 16.

[119] Coffa, 1987, p. 548. Esse trecho constitui a continuação do trecho anteriormente citado.

[120] Esse é o título da versão em alemão, publicada em 1935 com importantes alterações e correções em relação ao texto anterior. Em relação a essa data, é interessante observar, de maneira geral, a grande proximidade que havia entre os trabalhos de Carnap, Gödel e Tarski (os resultados que, ao longo da década de 1930, foram sendo publicados atestam vivamente essa evolução, em certa medida paralela e complementar, do pensamento dos três). Essa proximidade encontra explicações tanto temporais – eles tinham quase a mesma idade – como geográficas e culturais.
Do ponto de vista intelectual, os três se movimentavam no eixo Praga-Viena (Tarski menos do que os outros), duas cidades que à época mantinham um vínculo extremamente forte, no contexto do Império Austro-

Carnap é ligeiramente superior ao de Tarksi em alguns aspectos técnicos (posteriormente, Tarski veio a corrigir seus resultados no sentido apontado por Carnap), e parece gozar de certa precedência histórica. Exatamente em que ponto, então, Carnap falhou ao tentar oferecer uma solução abrangente para a nova situação delineada pelo teorema de incompletude? É essa a questão que a seguir nos ocupará.

III

Logo de início, é importante que penetremos um pouco mais a fundo no problema da verdade *aritmética*, tal como diretamente proposto pelo teorema de incompletude de Gödel. Na seção precedente, falamos a respeito do enigma colocado por esse teorema, ao demonstrar a existência – dentro de qualquer sistema formal de uma certa classe – de proposições "verdadeiras", porém "indemonstráveis". Aqui, examinaremos o teorema segundo uma perspectiva ligeiramente diversa: levaremos em conta, um pouco mais de perto, o mecanismo de sua construção, estratégia que nos permitirá enxergar a maneira simples como a demonstração de Gödel expõe a estrutura do problema da verdade aritmética. Trata-se de um tema, aliás, que de certo modo já apareceu em nosso trabalho. Não custa retomá-lo.

Resumidamente, Gödel mostra o seguinte: Para qualquer sistema formal que permita estabelecer uma enumeração recursiva das sentenças dedutíveis (teoremas), existe um predicado numérico G tal que a sentença "G(n)" é verdadeira para todos os números naturais n, mas tal que a sentença universal correspondente – seja ela a sentença fechada propriamente dita, "$\forall xG(x)$", ou a aberta "G(x)", em que x é uma variável numérica – não é demonstrável. Em outras palavras: todas as sentenças do conjunto K_G = {G(1), G(2), ... ,

Húngaro. Gödel era natural de Praga, mas trabalhou algum tempo em Viena, ao lado de Carnap, tomando parte no conhecido Círculo organizado por M. Schlick. Carnap, alemão de nascença, chegou a lecionar algum tempo na Universidade Alemã de Praga, mas estabeleceu definitivamente sua reputação como filósofo no importante período em que permaneceu em Viena, como um dos mais destacados membros do Círculo. Em Viena, Carnap e Gödel mantiveram contato intenso entre si, e o segundo teve várias ocasiões de explicar pessoalmente seus teoremas a Carnap.
Também Tarski, embora fosse polonês, manteve diversos contatos com Carnap, em diferentes ocasiões, antes e depois do período do Círculo. Tanto Gödel como Tarski, aliás, fizeram sugestões e comentários em relação ao manuscrito de *A Sintaxe Lógica da Linguagem*, fato esse que é amplamente reconhecido por Carnap no texto do próprio livro.

G(n), ... } são verdadeiras; portanto, também a sentença aberta[121] G(x), que afirma exatamente esse fato, é verdadeira; contudo, G(x) não é demonstrável.

Devemos chamar a atenção para o fato de que essa sentença G(x) é uma sentença puramente aritmética, na medida em que faz uma afirmação exclusivamente a respeito dos números naturais. Eis por que observamos, no início desta seção, que pretendíamos tratar da estrutura do problema da verdade *aritmética*, proposto pelo teorema de Gödel.

Há ainda um fato adicional que precisamos fixar, bastante relevante para os nossos propósitos. Em linguagens como a linguagem I e a linguagem II de Carnap – assim como na absoluta maioria dos sistemas lógico-formais – as sentenças do conjunto K_G não apenas são verdadeiras, como também são demonstráveis. (De fato, essas sentenças são todas "definidas", para utilizar uma terminologia de Carnap: cada uma delas realiza uma afirmação aritmética finita e individual que pode ser deduzida até mesmo em sistemas bastante econômicos em seus meios de dedução, como é o caso dos sistemas finitistas ou intuicionistas). Para essas linguagens, a estrutura do problema da verdade aritmética pode ser resumida de maneira ainda mais clara, assim: Todas as sentenças do conjunto K_G são verdadeiras e demonstráveis; a sentença G(x) correspondente também é verdadeira, porém não é demonstrável.

Posta a questão nesses termos, podemos agora constatar o seguinte fato importante. Carnap já havia conseguido fornecer um tratamento perfeitamente adequado para o problema específico da verdade *aritmética* em sua linguagem I (por meio do correspondente conceito de analiticidade). Na linguagem I, com efeito, Carnap propõe uma regra formal de conseqüência direta (trata-se da regra DC2[122]) nestes termos: dado qualquer predicado numérico P, pode-se inferir a sentença P(x) a partir do conjunto sentencial K_p = {P(0), P(1), P(2), ... , P(n)}. Não é difícil ver que uma tal regra, assim formulada, basta para resolver o problema de que estamos tratando.

A regra DC2, de fato, aplicada ao predicado de Gödel G (devemos recordar o fato, para o qual chamamos a atenção logo acima, de que as sentenças de K_G são todas demonstráveis), permite demonstrar a sentença G(x). Portanto, pelo acréscimo dessa

[121] Preferimos, no que segue, trabalhar sempre com a sentença aberta; o mesmo é válido, porém, para a sentença fechada correspondente.
[122] Ver *SLL*, pág. 38.

simples regra indefinida (baseada em um conjunto infinito de premissas), Carnap consegue resolver o problema da sentença de Gödel – verdadeira porém indemonstrável. Na linguagem I, ela passa a ser verdadeira e *demonstrável*, como seria desejável para qualquer sistema formalizado de aritmética. Em outras palavras, o conceito de analiticidade obtido para a linguagem I – formulado de maneira razoavelmente simples e, o que é mais importante, em direto acordo com o esquema da sintaxe geral – consegue solucionar o problema da verdade aritmética proposto pelo teorema de Gödel.

No Capítulo 2, mostramos que o teorema de Gödel forneceu a motivação central para que Carnap estendesse sua maneira de conceber o método formal, e incluísse também regras indefinidas de transformação. Mostramos ainda que esse tipo de regra podia sim ser considerado compatível com a teoria sintática/formal que está por trás de todo o projeto filosófico de *SLL*. Devemos agora reconhecer, no entanto, que Carnap não se detém nesse ponto. O conceito de analiticidade da linguagem I, embora de certa maneira represente uma resposta satisfatória para a questão das verdades aritméticas (na medida em que consegue captá-las de um ponto de vista extensional), acaba parecendo uma mera sombra, que pouca atenção desperta, em face do conceito de analiticidade para a linguagem II.

Ali, Carnap não se satisfaz com uma simples regra do tipo de DC2. Devemos chamar a atenção, contudo, para o fato de que essa regra (ou seu equivalente) bastaria para permitir a dedução de todas as verdades aritméticas de Gödel, mesmo na linguagem II. Por que, então, Carnap não se satisfaz com essa estratégia? O que ele busca obter com sua definição de analiticidade para a linguagem II?

A resposta é clara: Carnap deseja solucionar não apenas o problema da verdade aritmética, mas o problema mais amplo da verdade matemática em geral. Conforme já observamos no Capítulo 4 (ver seção I), a linguagem I não possui os recursos necessários para expressar todas as proposições da matemática clássica. Trata-se de uma linguagem essencialmente aritmética, e por isso Carnap considerou-se satisfeito com um conceito de analiticidade que captasse as verdades aritméticas. A linguagem II, no entanto, possui recursos expressivos suficientes para lidar com toda a matemática clássica. Carnap precisava agora encontrar um ponto de vista adequado para tratar essa questão.

No início da seção 34a, Carnap mostra-se bastante claro a esse respeito. Eis o que ele escreve:

> *"One of the chief tasks of the logical foundation of mathematics is to set up a formal criterion of validity, that is, to state the necessary and sufficient conditions which a sentence must fulfill in order to be valid (correct, true) in the sense understood in classical mathematics. Since Language II is constructed in such a way that classical mathematics may be formulated in it, we can state the problem as that of setting up a formal criterion of validity for the sentences of Language II."* [123]

Ao construir o conceito de analiticidade para a linguagem II, portanto, eis exatamente o que Carnap busca obter: uma maneira de captar o conceito de verdade matemática ou, como ele mesmo coloca, um critério de validade para a matemática *clássica*. E é importante enfatizar esse termo utilizado pelo próprio Carnap – *matemática clássica* –, por circunstâncias que ficarão claras um pouco mais à frente em nossa análise.

As razões por que uma simples regra indefinida de transformação como DC2, embora perfeitamente adequada para capturar as verdades aritméticas de Gödel, já não se mostra suficiente para capturar formalmente, como teoremas, esse conjunto mais amplo de sentenças – as sentenças verdadeiras da matemática clássica – já foram abordadas anteriormente (ver Capítulo 4, seção IV). Vale a pena retomar brevemente esse assunto.

Os números naturais, como se sabe, formam um conjunto enumerável. Assim, é possível construir, dentro de uma linguagem formal, uma estrutura isomorfa a eles. No caso das linguagens I e II de Carnap, essa estrutura isomorfa aos números naturais (uma "aritmética", como Carnap irá chamar tais estruturas na sintaxe geral[124]) é dada pelo conjunto A = {0, 0', 0'', ... , 0''''$^{\cdots}$' , ... }. Esse conjunto é um conjunto de *expressões numéricas*, ou seja, termos sintáticos da linguagem formal (símbolos da linguagem). Essa disponibilidade de expressões numéricas significa que toda afirmação individual acerca de um número natural encontra um correspondente sintático na linguagem I ou II (sob a forma de sentenças individuais). Como conseqüência, todas as instâncias individuais de uma

[123] *SLL*, pág. 98.
[124] Ver seção 58 de *SLL*.

afirmação geral acerca dos números naturais – afirmação geral essa que é representada na linguagem I e II por uma sentença como "P(x)" – existem como sentenças da linguagem (mais especificamente, as sentenças individuais P(0), P(0'), P(0'') etc.). É por isso, finalmente, que uma regra como DC2 pode ser formulada: Basicamente, ela diz que a sentença numérica geral pode ser deduzida a partir de todas as suas instâncias individuais, as quais estão disponíveis como sentenças da linguagem.

O mesmo, porém, não ocorre para tipos lógicos mais complexos. A dificuldade central manifesta-se plenamente já para o caso dos predicados numéricos, que correspondem ao tipo lógico mais simples depois das expressões numéricas. É fácil mostrar que nenhuma linguagem formal baseada em seqüências finitas de símbolos possui expressões suficientes para representar, de maneira sintática explícita, todos os predicados numéricos. A própria maneira de colocar a questão, aqui, mostra que estamos utilizando uma visão característica a respeito do que sejam predicados numéricos. Segundo essa visão, um predicado numérico é, de certa forma, qualquer coisa que transcende os recursos de qualquer linguagem formal específica. Uma sentença da linguagem II, por exemplo, que faça uma afirmação a respeito de "todos os predicados numéricos", não está se referindo somente a todos os predicados numéricos que podem ser definidos na linguagem, mas sim a *todos* os predicados numéricos.

Podemos ver essa mesma situação do seguinte ponto de vista. Carnap identifica predicados numéricos com conjuntos numéricos (conjuntos de expressões numéricas) e, equivalentemente, com números reais[125]. Por argumentos de cardinalidade que Carnap prontamente aceitaria (embora ele prefira invocar argumentos extraídos do método de aritmetização de Gödel[126]), pode-se mostrar que nenhum sistema formal pode representar explicitamente todos os números reais ou todos os conjuntos numéricos. Não obstante, um sistema formal pode representar implicitamente esses conjuntos: por meio, justamente, do quantificador universal. De fato, Carnap opta por entender as sentenças da linguagem II em que aparece a expressão "todos os predicados numéricos" como uma sentença que faz uma afirmação, não somente sobre os predicados numéricos que podem ser definidos

[125] Seções 37 e 39 de *SLL*.
[126] Ver citação logo à frente.

(representados) na linguagem, mas sobre todos os predicados numéricos (todos os números reais, ou todos os conjuntos numéricos)[127].

Carnap é bastante claro a respeito dessa opção. Citamos aqui novamente uma passagem que já foi citada no Capítulo 4, mas que pode agora ser compreendida sob novo enfoque:

> *"In the case of a predicate- or a functor-variable, however, the analogous method does not succeed; a fact which has been pointed out by Gödel. Let C_1 be, for example, 'M(F)' (in words: 'M is true for all properties'). Now, if from C_1 we refer back to the sentences 'M(P_1)', 'M(P_2)', and so on, which result from C_1 by substituting for 'F' each of he predicates of the type in question which are definable in II, in turn, then it may happen that, though all these sentences are true, 'M(F)' is nevertheless false – in so far as M does not hold for a certain property for which no predicate can be defined in II. As a result of Gödel's researches it is certain, for instance, that for every arithmetical system there are* numerical properties which are not definable*, or, in other words, indefinable real numbers (see Theorem 60d.1, p. 221). Obviously it would not be consistent with the concept of validity of classical mathematics if we were to call the sentence: 'All real numbers have the property M' an analytic sentence, when a real number can be stated (not, certainly, in the linguistic system concerned, but in a richer system) which does not possess this property."* [128] (destaque do autor)

[127] Para colocar a questão de maneira mais precisa: Carnap adota uma certa visão – exterior à linguagem II – a respeito do que seja o conjunto dos números reais e (de maneira equivalente) do que seja o conjunto de todos os conjuntos de números naturais. Ele adota, portanto, uma certa visão a respeito do que devem significar as expressões informais "todos os números reais" e "todos os conjuntos numéricos" (essa visão é uma visão extensional, em que um conjunto de números naturais é qualquer conjunto extensionalmente possível de números naturais). Finalmente, como ele considera que predicados numéricos da linguagem II representam números reais (ou conjuntos numéricos), ele equaciona as duas expressões anteriormente mencionadas com a expressão formal da linguagem II "todos os predicados numéricos".
Todas as muitas e complicadas questões envolvidas na explicação acima ficarão mais claras à medida que avançarmos neste capítulo.

[128] *SLL*, pág. 106.

Nesse trecho, Carnap manifesta com todas as letras sua intenção: a) de que a noção de analiticidade para a linguagem II deve servir como critério de validade para a *matemática clássica*; e b) de que as sentenças da linguagem II, nas quais apareça uma variável livre (ou universalmente quantificada) para predicados numéricos, referem-se a todos os números reais, e não somente aos números reais definíveis na própria linguagem II. E é por isso que uma regra simples como DC2 não funciona mais: porque não existem expressões suficientes na linguagem II para representar todos os predicados pretendidos, e consequentemente não é possível construir um conjunto adequado de sentenças a partir das quais deduzir imediatamente a sentença geral sobre predicados.

Essa opção de Carnap[129], porém, não pode acontecer sem despertar graves problemas, alguns dos quais já foram adiantados. Dissemos acima que, segundo a concepção adotada em *SLL*, as sentenças gerais acerca de predicados numéricos referem-se a *todos* os predicados numéricos (ou a *todos* os números reais), em algum sentido que transcende a mera possibilidade de representação direta na linguagem. É necessário esclarecer exatamente, então, em que sentido deve-se compreender essas sentenças, em especial a expressão "todos" que nelas aparece. A resposta de Carnap para essa questão é dada pela idéia de valoração, que descrevemos no capítulo anterior, e que podemos agora explorar.

IV

Carnap deseja fornecer, com a definição de analiticidade para a linguagem II, um critério de validade para a matemática clássica; e a matemática clássica constitui-se como o domínio de todos os números reais. É por essa razão que as sentenças gerais acerca de predicados numéricos da linguagem II não podem se referir somente a alguns (relativamente poucos, como sabemos) predicados numéricos definíveis na própria linguagem II. Carnap, então, precisa fornecer uma solução para o problema de identificar essas condições de verdade que valem para as proposições da matemática clássica –

[129] Trata-se de uma opção oposta à opção original de Russell, ao construir o sistema dos *Principia*. Para o pensador inglês, as sentenças universais sobre entidades de qualquer tipo lógico deveriam referir-se somente a entidades definíveis dentro da linguagem. A esse respeito, ver [Hintikka, 1995].

proposições acerca de *todos* os números reais. Sua resposta está na elaboração do sistema de valorações para os diversos tipos lógicos da linguagem II.

Na teoria das valorações (descrita por nós no Capítulo 4, seção IV), encontramos o foco para o qual convergem as maiores contradições de *SLL* – bem como, possivelmente, seus vôos mais elevados. Examinemos exatamente o que ela propõe, e o que consegue obter.

A linguagem II é uma linguagem que possui, como conjunto de expressões sintáticas, uma estrutura isomorfa aos números naturais. Trata-se, como sabemos, do conjunto de expressões numéricas dado por A = {0, 0', 0'', ... , 0''''$^{...}$' , ... }. A valoração das variáveis numéricas da linguagem é obtida trivialmente por meio desse conjunto sintático. Em outras palavras: o conjunto de valorações possíveis para uma variável numérica é dado pelo conjunto A de expressões sintáticas. Dizer isso é simplesmente uma nova maneira de dizer que uma sentença como "P(x)" – em que P é um predicado numérico e x uma variável numérica – equivale ao conjunto de sentenças obtidas pela substituição de x por todas as expressões numéricas, vistas agora como possíveis valorações dessa variável (e que representam, em última instância, o conjunto dos números naturais).

Carnap, agora, precisa estender esse esquema de valorações para outros tipos lógicos. Tratemos novamente do caso mais simples, que é dado pelos predicados numéricos. Uma variável para predicados numéricos, conforme mostramos na seção anterior, deve "referir-se"[130] a *todos* os predicados numéricos. Já observamos também que, diante dessa afirmação, a seguinte pergunta se impõe: Que sentido deve assumir o termo "todos" que nela aparece? Carnap, por sua vez, fornece uma resposta surpreendentemente precisa a essa pergunta: O termo "todos" deve ser compreendido no sentido que será dado pelas valorações. As valorações são introduzidas, assim, justamente para fixar o conjunto de valores que uma variável pode assumir (e em vista desse conjunto de valores estabelecer a maneira como ela será avaliada).

[130] Não precisamos entrar agora em uma discussão a respeito de supostos "referentes" para uma variável ou expressão lingüística. Uma tal discussão já viciaria, na origem, a discussão que estamos realizando. Aqui, dizemos que uma variável para predicados numéricos "refere-se" (entre aspas) a todos os predicados numéricos no sentido inequívoco, manifestamente expresso por Carnap, de que as sentenças em que tais variáveis aparecem fazem afirmações acerca de todos os números reais (ou todos os conjuntos numéricos, o que é equivalente).

Carnap estava diante, portanto, do problema de determinar a classe de possíveis valorações das variáveis para predicados numéricos. Essa classe de valorações é que deveria servir para captar o conjunto de todos os números reais (ou o conjunto de todos os conjuntos de números naturais). Pois é com base nessa classe de valorações que as sentenças da linguagem II, nas quais aparecem variáveis para predicados numéricos, serão avaliadas e, em última instância, receberão seu *status* de analiticidade. Carnap, então, extrai a única conclusão possível para a própria maneira – absolutamente inovadora – como formulou o problema: Se a classe de valorações para predicados numéricos deve captar a classe de todos os conjuntos de números naturais; se os números naturais são valorados como o conjunto de expressões numéricas da linguagem (conjunto A); então as valorações para predicados numéricos têm de ser dadas por conjuntos de expressões numéricas, vale dizer, por subconjuntos do conjunto A, e a classe das valorações para predicados numéricos como um todo tem de ser dada pela totalidade desses conjuntos, ou seja, por aquilo que em teoria dos conjuntos costuma-se chamar de "partes de A", simbolicamente indicada como "$\pi(A)$".

A partir desse raciocínio, todo o resto do sistema de valorações é construído de maneira recursiva, seguindo o mesmo princípio indicado. Por exemplo, as valorações para predicados de segunda ordem (que aceitam predicados numéricos como argumento), são dadas como conjuntos de conjuntos de expressões numéricas; e a classe dessas valorações é dada pela totalidade desses conjuntos, ou seja, por $\pi(\pi(A))$.

Esse sistema de valorações fornece toda a base para a avaliação das sentenças da linguagem II, e é com base nele que Carnap conseguirá capturar as "verdades" da matemática clássica. Em outras palavras: é o sistema de valorações que permitirá a Carnap avaliar como analíticas todas as sentenças verdadeiras no sentido da matemática clássica, pois é esse sistema de valorações que permite que as sentenças da linguagem II sejam *interpretadas* como sentenças da matemática clássica.

E chegamos aqui a um ponto fundamental. Falar em interpretação, nesse contexto, pode de fato parecer duvidoso. Como se sabe, o termo "interpretação" é um termo que costuma aparecer dentro da teoria de modelos, e que só nela ganha seu pleno significado. Carnap, por outro lado, nunca concebeu nada como uma teoria de modelos e, no sistema de

SLL, deveria mostrar-se mesmo refratário a uma tal abordagem, na medida em que defende um enfoque puramente sintático, com exclusão de conceitos semânticos.

Devemos examinar, portanto, cuidadosamente essa questão, dedicando-lhe minuciosa atenção. Desde logo, porém, cabe observar o seguinte a respeito do caminho que iremos adotar. É lugar comum falar a respeito das semelhanças que existem entre o método que Carnap desenvolveu para definir analiticidade na linguagem II de *SLL* – método esse que estamos tentando esclarecer – e os métodos semânticos pioneiramente formulados por Tarski em seu clássico *Der Wahrheitsbegriff in den formalisierten Sprachen*[131]. Examinar e comparar diretamente esses dois métodos, aqui, implicaria expor toda a teoria de Tarski, o que ultrapassa claramente os propósitos deste trabalho. Nossa abordagem, assim, terá de ser outra. O melhor é prosseguir com o mesmo enfoque que estamos adotando desde o início, e realizar uma análise endógena da proposta de Carnap. Por esse caminho, poderemos esclarecer suficientemente as questões levantadas no parágrafo anterior. Como próximo passo do nosso estudo, portanto, devemos avaliar a importância dos elementos semânticos para o método de Carnap. É o que faremos.

V

Começaremos nossa exposição por uma indicação clara do ponto a que pretendemos chegar: O método desenvolvido por Carnap para definir analiticidade para a linguagem II de *SLL* é um método plenamente semântico e, mais do que isso, traz implícita a idéia de "modelo"; Carnap, no entanto, não chegou a perceber exatamente aquilo que havia obtido, por razões que tentaremos explicar.

Uma teoria semântica caracteriza-se por interpretar as sentenças de uma linguagem em termos da referência que fazem a certo domínio de objetos. Aqui, os termos "interpretar" e "referência" estão corretamente utilizados, como termos semânticos próprios que são. Na descrição da teoria das valorações de Carnap, tivemos que lutar contra a tendência constante de utilizar esses termos: Carnap não via a sua teoria como uma teoria semântica, e sim como uma teoria sintática. Contudo, esse ponto de vista é – como, aliás, ele próprio viria mais tarde a admitir – insustentável.

[131] Ver nota 120 acima.

A chave do problema reside, em grande parte, em um aspecto específico das linguagens I e II de Carnap. Nessas linguagens, como sabemos, aparece o conjunto de expressões numéricas A = {0, 0', 0'', ... , 0''''...' , ... }. Essas expressões, mais ainda, são as únicas constantes – os únicos nomes individuais – existentes na linguagem. A relação entre elas, baseada no símbolo de sucessão " ' ", é introduzida por meio dos axiomas usuais da aritmética[132]. Não à toa, Carnap chama estruturas sintáticas desse tipo, na sintaxe geral, de "aritméticas".

Basicamente, o que a presença dessas estruturas na linguagem I e II representa é a adoção de um modelo para os números naturais. É com base nesse modelo dos naturais, embutido na linguagem, que os números reais são construídos. Precisamos esclarecer melhor, porém, em que sentido exatamente podemos afirmar que o conjunto A desempenha o papel de modelo e determina, assim, a adoção – ainda que inconsciente – de uma teoria semântica por parte de Carnap.

Como vimos, o problema principal surge na hora de tratar as expressões de tipo lógico superior. As variáveis numéricas podem ser substituídas diretamente pelos elementos do conjunto A, para formar sentenças individuais e permitir a elaboração de regras indefinidas de transformação. O mesmo não acontece com outros tipos de variáveis. (Na seqüência de nossa discussão, por uma questão de clareza, utilizaremos – como aliás tem sido sempre a nossa prática – o exemplo de variáveis para predicados numéricos). O ponto importante, para o qual devemos agora chamar a atenção, é que a questão toda é colocada, pelo próprio Carnap, em termos da *interpretação* que deve ser dada para as sentenças em que tais variáveis para predicados numéricos aparecem.

Na extensa passagem citada na seção III acima, esse aspecto da discussão surge muito claramente. A pergunta que Carnap faz é: O que quer dizer uma sentença como "M(F)", em que F é uma variável para predicados numéricos? Podemos saber que Carnap faz exatamente essa pergunta – embora não a formule com todas as letras – porque a resposta que ele dá é uma resposta adequada exatamente para essa pergunta. Mais

[132] Os axiomas usuais da aritmética aparecem, dentro do sistema sintático (regras sintáticas da linguagem), como sentenças iniciais. Por meio do segundo conjunto de regras de valoração (aquelas regras cuja função é exigir coerência na valoração de diferentes expressões de uma mesma sentença), esses axiomas são transpostos para a classe de valorações de nível zero, que passam a constituir assim um verdadeiro modelo dos números naturais.

especificamente, Carnap diz que uma tal sentença não pode ser tida como analítica se existir um número real para o qual não valha a propriedade M, mesmo que esse número real não possa ser definido na linguagem II.

A questão que o preocupa, portanto, pode ser colocada ainda mais claramente da seguinte maneira: *Sobre quem* (a respeito de que elementos) fala uma sentença como "M(F)"? A resposta é explícita: essa sentença fala a respeito de todos os números reais, independentemente de poderem ou não ser definidos na linguagem II. Repetimos o trecho relevante:

> "*Obviously it would not be consistent with the concept of validity of classical mathematics if we were to call the sentence: <u>'All real numbers have the property M'</u> an analytic sentence, when a real number can be stated (not, certainly, in the linguistic system concerned, but in a richer system) which does not possess this property.*" [133] (grifo meu)

Uma vez colocada a questão nesses termos, porém, ela já escapa inevitavelmente ao plano sintático. A expressão "todos os números reais", que Carnap equaciona explicitamente com a utilização de uma variável para predicados numéricos, é uma expressão que faz sentido em relação a um modelo dos números reais. E Carnap – o que é mais surpreendente – constrói esse modelo de maneira absolutamente sofisticada: ele o faz por meio da sua teoria das valorações. Nessa teoria, o modelo dos números naturais é construído com base no conjunto A de expressões numéricas, que entram no sistema de valorações como valorações para as variáveis de nível zero (variáveis numéricas).

Como, então, Carnap não percebeu esse fato?

A resposta, aqui, tem de permanecer no campo da conjectura. Parece, no entanto, que a própria moldura em que Carnap formulou o problema, segundo o esquema que estamos descrevendo, induziu-o a cometer esse deslize. Para Carnap, seguindo a tendência absolutamente generalizada na matemática do seu tempo (que é também a tendência predominante nos dias de hoje), os números reais deviam ser construídos a partir dos números naturais. O modelo que Carnap constrói para os números reais, então, baseia-se

---

[133] *SLL*, pág. 106.

em dois elementos lógicos que, de um ponto de vista atual, podemos identificar claramente: 1) O conjunto A das expressões numéricas da linguagem II, que representam os números naturais; e 2) Uma interpretação extensional para a teoria de conjuntos, ou seja, uma interpretação em que a classe formada por todos os subconjuntos do conjunto A – ou seja, $\pi(A)$ – é interpretada de maneira extensional: os subconjuntos de A são todos os subconjuntos extensionalmente possíveis de A, e não somente aqueles definíveis em certa linguagem.

A confusão em que Carnap submerge – o que já seria bastante natural devido à complexidade e à novidade das idéias envolvidas – é portanto estimulada pelo fato de que o modelo dos números naturais em que Carnap baseia sua construção dos reais tem por base o conjunto A, que é um conjunto de expressões sintáticas da linguagem. Pior: a estrutura que torna esse conjunto um modelo para os números naturais é dada, formalmente, por sentenças iniciais (sintáticas) da linguagem II, as quais expressam os axiomas usuais da aritmética[134]. Em outras palavras, o que parece ter atrapalhado Carnap foi um conjunto de circunstâncias, que podemos resumir da seguinte maneira.

O modelo dos naturais que ele utiliza, dado pelo conjunto de valorações para as variáveis numéricas de nível zero, tem dupla existência. Uma existência sintática, como conjunto de expressões da linguagem II, e uma existência semântica, como conjunto de valorações. (Não devemos esquecer, a esse respeito, que as valorações representam os *valores* que uma variável pode *assumir*, e que esses valores – como o caso dos números reais demonstra claramente – não se restringem a elementos com existência sintática dentro da linguagem II.) O conjunto A, portanto, desempenha o papel de conjunto sintático de expressões e também de conjunto semântico de valorações (o que só é possível porque o conjunto dos números naturais é enumerável). A estrutura aritmética desse conjunto, mais ainda, é formalizada *dentro* da linguagem, o que aumenta a "promiscuidade" entre suas duas utilizações.

Com base na utilização semântica do conjunto A – que Carnap, pelos motivos aventados acima, não distinguiu claramente – é construída toda a teoria de valorações. Essa teoria é uma teoria semântica, por meio da qual Carnap oferece um domínio de referência

[134] Ver nota 132 acima.

para as sentenças da linguagem II. As valorações, como valores possíveis para uma variável, remetem para fora do âmbito sintático da linguagem II. Elas representam, de fato, algo externo à linguagem, com base no qual suas sentenças são avaliadas. A teoria de valorações separa uma classe de valorações possíveis para cada tipo lógico; e essas classes são compostas por elementos externos à linguagem, na medida em que não podem ser representados por meio de seus recursos simbólicos[135] (e uma linguagem é vista, dentro de uma teoria sintática como deveria ser a de Carnap, como um conjunto de expressões simbólicas). A teoria de valorações é construída com base no conjunto A e na operação que passa de um conjunto à totalidade de seus subconjuntos (além de outras operações do mesmo gênero, como a formação de correlações, as quais funcionam como valoração para funtores); e essa totalidade é interpretada em termos extensionais. A operação que leva de A a π(A), nesse sentido, é uma operação interpretada dentro de uma teoria extensional de conjuntos ou, o que é a mesma coisa, dentro de um modelo extensional para a teoria de conjuntos.

Desse modo, podemos dizer que a confusão de Carnap tem – *do ponto de vista técnico* – uma origem clara. Ele acreditou, pelo fato de utilizar elementos sintáticos como valorações para variáveis numéricas de tipo zero (as expressões sintáticas do conjunto A), que toda a teoria de valorações poderia manter um significado sintático. Essa posição, como vimos, é insustentável. No entanto, a confusão técnica não é o único motivo para Carnap não ter desenvolvido um ponto de vista abertamente semântico. Há ainda outros fatores a considerar.

VI

Um aspecto que nós temos de considerar mais de perto, agora, é o fato de Carnap rejeitar, no esquema teórico de *SLL*, o conceito de verdade (falaremos, alternativamente, no conceito de “verdadeiro” e no conceito de “falso”) como ilegítimo em lógica. Precisamos buscar as causas dessa rejeição, bem como suas conseqüências.

---

[135] Os elementos de π(A), por exemplo, que compõem uma das classes de valorações, não podem ser todos representados com os recursos da linguagem II (nem de nenhuma linguagem finita), quando se compreende π(A) de um ponto de vista extensional (como Carnap faz).

Logo de cara, um ponto pode parecer estranho em nossa análise. Ao longo do presente capítulo, insistimos que boa parte do esforço lógico realizado por Carnap – ao oferecer as definições de analiticidade para a linguagem I e para a linguagem II, por exemplo – representa justamente uma tentativa de captar a idéia de verdade aritmética e de verdade matemática. Mais ainda, exibimos citações que parecem não dar margem a opinião diversa. Precisamos mostrar, portanto, como o projeto de Carnap se articula, de maneira a permitir esse duplo movimento: a tentativa de capturar *certos* conceitos de "verdade", e a rejeição explícita desse conceito.

Como já observamos (ver Capítulo 1), Carnap deseja reduzir toda a lógica à análise sintática. Ele acredita que a lógica possa e deva ser formulada de maneira puramente formal, como um cálculo simbólico. Segundo o seu ponto de vista, não é adequado (e nem muito menos necessário) que a lógica busque determinar suas relações, de maneira absoluta, em alguma esfera transcendente de "significados". Uma tal tentativa tenderia a engessar o estudo lógico, e a envolvê-lo em todo tipo de discussões pouco frutíferas, na medida em que o acesso a esses reinos transcendentais permanece sempre encoberto por um véu de incertezas ou, o que pior, de dogmas injustificados.

Para Carnap, é necessário desenvolver diferentes sistemas lógicos, cada um dos quais dotado de regras sintáticas claras. Um sistema lógico, nesse sentido, é um cálculo simbólico descrito de maneira clara o suficiente para permitir a compreensão de suas regras (ver Capítulo 2). Qualquer cálculo simbólico, desde que bem formulado, é admissível como sistema lógico. A única medida que temos para comparar esses diferentes sistemas deve ser procurada na ciência. Isso porque uma linguagem formal não é desenvolvida para ser um mero "brinquedo" formal, mas para permitir a descrição de certos domínios de interesse científico. As diferentes linguagens, portanto, poderão mostrar-se mais ou menos adequadas de acordo com as necessidades de uma ou outra ciência. As relações lógicas verificadas dentro das diferentes linguagens (sua estrutura dedutiva) poderão ou não espelhar relações encontradas em um ou outro ramo do conhecimento. E somente essa utilização prática é que permitirá selecionar entre os muitos sistemas lógicos possíveis – não de maneira absoluta, com o objetivo de escolher ou encontrar o único sistema lógico verdadeiro, mas de maneira relativa, segundo os critérios e a metodologia local de cada ciência.

Esse é, em resumo, o significado do Princípio de Tolerância defendido por Carnap em *SLL*. Sua formulação é oferecida logo na seção 17 do livro, intitulada "The Principle of Tolerance in Synrax". Ainda no início dessa seção, Carnap escreve:

> *"In the foregoing we have discussed several examples of negative requirements (...) by which certain common forms of language – methods of expression and of inference – would be excluded. Our attitude to requirements of this kind is given a general formulation in the* Principle of Tolerance: It is not our business to set up prohibitions, but to arrive at conventions*."* [136] (destaque do autor)

Um pouco mais à frente, Carnap é ainda mais claro:

> *"*In logic, there are no morals*. Everyone is at liberty to build up his own logic, i.e. his own form of language, as he wishes. All that is required of him is that, if he wishes to discuss it, he must state his methods clearly, and give syntactical rules instead of philosophical arguments."* [137] (destaque do autor)

Do nosso ponto de vista, interessa agora enfatizar que a adoção do Princípio de Tolerância corresponde, do ponto de vista de Carnap, a uma rejeição de qualquer idéia de verdade absoluta em matéria de lógica. Para ele, não é possível falar em uma única lógica "verdadeira", que traduza algo como as "verdadeiras" relações (lógicas) verificadas no mundo. Eis uma passagem de *SLL*, retirada ainda do "foreword", que confirma bem essa opinião:

> *"The fact that no attempts have been made to venture still further from the classical forms [da Lógica] is perhaps due to the widely held opinion that any such deviations must be justified – that is, that the new language-form must be proved to be 'correct' and to constitute a faithful rendering of 'the true logic'.*

[136] *SLL*, pág. 51.
[137] *SLL*, pág. 52.

To eliminate this standpoint, together with the pseudo-problems and wearisome controversies which arise as result of it, is one of the chief tasks of this book. (...)" [138]

Carnap deseja fornecer um quadro teórico abrangente e adequado dentro do qual formular sistemas lógico-sintáticos de maneira clara e correta, sem que nenhum goze de prioridade sobre os outros. É a essa tentativa que corresponde, como vimos, o desenvolvimento da sintaxe geral, descrita na parte IV de *SLL*. A sintaxe geral, portanto, desempenha um papel fundamental para a arquitetura teórica de Carnap como um todo. No que diz respeito ao projeto filosófico oferecido no livro, ela constitui a peça teórica central para o qual devemos voltar nossa atenção sempre que se trata de compreender as motivações e aspirações do sistema de Carnap.

A estrutura da sintaxe geral, por sua vez, pode ser resumida de maneira simples. Em primeiro lugar, Carnap fornece uma moldura teórica estendida (em relação aos esquemas tradicionais) dentro da qual formular as regras sintáticas – regras de formação e transformação – que caracterizam uma linguagem. A partir daí, ele formula um quadro conceitual que deve servir para analisar a estrutura dessas linguagens. São esses os dois momentos básicos da sintaxe geral. Eles estão ligados da seguinte maneira: todos os conceitos sintáticos usados para analisar a estrutura de uma linguagem devem ser estabelecidos somente com base nas regras sintáticas que caracterizam essa linguagem. Devem ser, em outras palavras, conceitos formais.

Carnap elabora um sistema, portanto, em que a análise da linguagem é realizada internamente, ou seja, por referência exclusiva a suas regras sintáticas internas. O conceito básico de analiticidade, tal como concebido na sintaxe geral, obedece estritamente a essa exigência. São analíticas as sentenças que podem ser obtidas pelas regras da linguagem[139]. Por outro lado, Carnap não esconde que os conceitos formais de uma linguagem podem ser inspirados em todo tipo de discussão informal. Posto de outra maneira: Um conceito sintático, estabelecido com base em critérios puramente formais, pode buscar traduzir certa concepção intuitiva e informal que se tenha a respeito das estruturas lógicas. Na verdade,

[138] [SLL], págs. xiv e xv.
[139] Não nos preocuparemos, no que segue, com a distinção entre sentenças válidas e analíticas. Em uma linguagem lógica, como é o núcleo da linguagem II, essa distinção desaparece.

esse é o único método possível para a construção de uma linguagem; nenhuma linguagem é desenvolvida, por assim dizer, no vazio.

Esse princípio está presente de maneira muito clara nos desenvolvimentos concretos da sintaxe geral. Foi exatamente esse o ponto que enfatizamos, por exemplo, na seção VI do Capítulo 3, quando discutimos a divisão formal que Carnap propõe entre as expressões lógicas e as expressões descritivas de uma linguagem. Essa distinção recebe uma definição puramente formal; no entanto, busca captar certa idéia intuitiva que nós temos a respeito do que sejam termos descritivos e termos lógicos. Podemos aqui repetir uma citação, que já oferecemos naquela seção, e que mostra exatamente a orientação adotada por Carnap a esse respeito:

> *"Our thesis that the logic of science is syntax must therefore not be misunderstood to mean that the task of the logic of science could be carried out independently of empirical science and without regard to its empirical results. The syntactical investigation of a system which is already given is indeed a purely mathematical task. But the language of science is not given to us in a syntactically established form; whoever wants to investigate it must accordingly take into consideration the language which is used in practice in the special sciences, and only lay down rules on the basis of this."* [140]

As mesmas considerações são válidas no que diz respeito ao conceito de "analiticidade". Trata-se, na visão de Carnap, de um conceito que deve ser estabelecido de maneira puramente formal. O conceito de analiticidade é um conceito sintático, interno às regras de uma linguagem. Nada obsta, porém, a que ele busque capturar certa visão informal (no sentido de pré-formal) que se tenha de respeito de um certo domínio do conhecimento. É exatamente isso o que o conceito de analiticidade para a linguagem II tenta fazer para o domínio da *matemática clássica*. Esse conceito traduz a tentativa que Carnap faz para formalizar um conceito de verdade que permanecia, segundo o seu ponto de vista, informal e não esclarecido.

[140] *SLL*, pág. 332.

É importante compreender bem esse ponto[141]. Carnap encontrava diante de si um corpo de teoria a que se pode chamar, propriamente, de “matemática clássica”. Trata-se do corpo de conhecimento que se erigiu, com os métodos da análise matemática moderna (a partir de fins do século XVIII), em torno dos números reais. Esses métodos da matemática clássica, é bem verdade, podiam ser atacados por uma ou outra tendência do fundacionismo matemático; os intuicionistas, por exemplo, não aceitarão a totalidade desses métodos, enquanto outros pensadores sequer aceitarão a noção usual de número real, passando a adotar uma perspectiva construtivista segundo a qual só são admissíveis números reais que possam ser explicitamente construídos. Carnap, no entanto, não duvida em nenhum momento que a maioria dos matemáticos atuantes – aquilo que a língua inglesa chama de “working mathematicians” – trabalha dentro de uma moldura teórica comum, em que os números reais são de fato não-denumeráveis.

Carnap aceita essa moldura, no sentido de que buscará justamente formalizá-la de maneira adequada. Isso não significa de sua parte, porém, nenhum compromisso com um conceito absoluto de verdade matemática. Por meio de seu conceito de analiticidade para a linguagem II, ele procura encontrar um equivalente formal bem construído para o corpo de “verdades” da matemática clássica, como domínio científico específico[142]. Como vimos, ele

[141] A natureza do tratamento que Carnap oferece para a matemática tem sido foco de inúmeras controvérsias, sem que se tenha chegado a qualquer acordo sobre o assunto. Opiniões bastante diversas são encontradas na literatura mais recente, algumas delas opostas. Tome-se, por exemplo, essas duas passagens, de estudiosos tão eminentes como T. Ricketts e M. Friedman:

> *“ (...) Furthermore, it is not apt to attribute to Carnap a conventionalist philosophy of mathematics. Such an account addresses the question ‘What is the nature of pure mathematics and its applications in science?’ that Kant, Frege, Russell, and Schlick all variously address. A rejection of this question is implicit in* Logical Syntax*.”* (grifo meu) – ver [Ricketts, 1994]: pág. 177.
>
> Em sentido oposto, porém:
>
> *“Throughout his philosophical career, Carnap places the foundations of logic and mathematics at the center of his inquiries: he is concerned above all with the Kantian question ‘How is mathematics (both pure and applied) possible?’.”* (grifo meu) – ver [Friedman, 1999]: pág. 165. (Friedman deixa claro, na seqüência do artigo, que está se referindo principalmente às propostas de Carnap em SLL: *“Yet when one looks at Logical Syntax, which is clearly Carnap´s richest and most systematic discussion of these foundational questions, (...).”*, pág. 166.)

[142] Coffa escreve – e nós em parte *discordamos* – o seguinte:

> *“Let us note in passing that Carnap was never tempted to say that perhaps some mathematical statements are neither true or false. (...) Even though Carnap’s extreme reluctance to endorse a concept of truth different from that of well grounded belief was*

é explícito ao afirmar que buscará encontrar um “critério de validade para a matemática clássica” (ver seção III acima). Mais do que isso, Carnap consegue obter – por meio de um método então absolutamente inovador – exatamente aquilo que deseja: uma formalização perfeitamente adequada desse corpo de “verdades” até então informais ou, de toda maneira, não suficientemente formalizadas, do ponto de vista adotado em *SLL*.

Podemos acrescentar que sua definição é perfeitamente correta no mesmo sentido em que a definição de Tarski é correta. Por meio do método de valorações, ele capta exatamente o sentido em que as proposições da matemática clássica são verdadeiras: como afirmações acerca de um corpo de números – os números reais clássicos – essencialmente não-denumerável. Em termos de valorações, ele encontra uma base para interpretar as proposições da linguagem II que tratam desses números reais (vistos aqui como predicados numéricos) por meio do conjunto π(A) (o conjunto das partes do conjunto A), segundo uma interpretação extensional da teoria dos conjuntos (ver seção IV acima).

Em resumo: a linguagem II, como já mostramos (ver a primeira citação da seção I do Capítulo 4), foi construída para traduzir a matemática clássica. Seu conceito de analiticidade, que Carnap estabelece de maneira puramente formal, consegue captar precisamente o conjunto das proposições “verdadeiras” de acordo com a matemática

---

defeated only after he learned of Tarski’s work, he never seems to have seriously doubted that in the range of mathematics, proof was one thing and truth an entirely different one. When Gödel convinced him that proof could not even grasp extensionally the concept of mathematical truth his instinctive reaction was: something else must. The most interesting technical portions of LSL are devoted to the task of explicating this new notion of mathematical truth.” (grifo meu) – ver [Coffa, 1987]: pág. 549.

Carnap não coloca, de fato, a questão em termos de *as proposições matemáticas* serem *verdeiras ou falsas*. Essa maneira de formular a questão estaria em conflito aberto com o Princípio de Tolerância. Em primeiro lugar, não é sequer necessário que uma linguagem tenha meios para exprimir as proposições da matemática *clássica*; a linguagem I, por exemplo, não possui esses recursos.
Para Carnap, cada linguagem consegue exprimir certas proposições, e cada linguagem possui um conceito de analiticidade que lhe é próprio. Nenhum desses conceitos de analiticidade exprime “a verdade” ou “a falsidade” de “proposições matemáticas”. Isso não impede, porém, que um determinado conceito de analiticidade (no caso, o conceito de analiticidade para a linguagem II), seja construído para captar certas proposições de uma teoria específica: a matemática *clássica*. Aqui, “matemática clássica” designa apenas uma certa teoria, a respeito da qual há certo consenso (mas não um consenso absoluto) em relação ao que deva contar como “verdade”. Essa teoria, não-empírica, é extremamente importante para a ciência moderna, bem como para a história do pensamento teórico em geral, e por isso Carnap ocupa-se tão longamente dela.
A importância da matemática clássica, no entanto, não exclui a possibilidade de se formular outras “matemáticas”, igualmente não-empíricas, e possivelmente úteis em diferentes domínios científicos. Em nenhum caso, no entanto, estaríamos autorizados a falar em algum sentido absoluto de verdade - falar que as sentenças de tal ou qual matemática são “verdadeiras” ou “falsas” – como parece implicar o trecho de Coffa que nós citamos.

clássica. O problema é que essa formalização não era do tipo que Carnap acreditava que poderia ser.

VII

Carnap, embora dedique diversas seções de *SLL*[143] à elaboração de um conceito de analiticidade (a analiticidade para a linguagem II) cujo objetivo é captar um corpo específico de proposições – as proposições verdadeiras *no sentido da matemática clássica* – não perde de vista a idéia central do Princípio de Tolerância, que orienta toda a obra. Não se pode falar em verdade matemática em geral, em sentido absoluto, porque sequer há algo como uma matemática em geral, que como tal deva ser universalmente aceita. A discussão que Carnap faz a respeito das tendências intuicionistas em matemática, na seção intitulada "On Intuitionism" (seção 16), é bastante esclarecedora acerca de sua posição. Podemos reproduzir o seguinte trecho:

> *"(...) We hold that the problems dealt with by Intuitionism can be exactly formulated only by means of the construction of a calculus, and that all the non-formal discussions are to be regarded merely as more or less vague preliminaries to such a construction. (...)*
>
> *(...) Once the fact is realized that all the pros and cons of the intuitionist discussions are concerned with the forms of a calculus, questions will no longer be put in the form: 'What is this or that like?' but instead we shall ask: 'How* do we wish to arrange *this or that in the language to be constructed?' or, from the theoretical standpoint: 'What consequences will ensue if we construct a language in this or that way?'*
>
> *On this view the dogmatic attitude which renders so many discussions unfruitful disappears. When we here construct our language I in such a way that it is a definite language, and thus fulfils certain conditions laid down by Intuitionism, we do not mean thereby to suggest that this is the*

[143] Segundo Coffa, as seções mais "interessantes" do livro (ver citação oferecida na nota anterior).

only possible or justifiable form of language. (...)" [144] (destaque do autor; grifos meus)

Não parecem acertadas, portanto, as opiniões que buscam compreender a posição de Carnap em *SLL* – e particularmente seu esforço para desenvolver o conceito de analiticidade para a linguagem II – em termos de qualquer tipo de compromisso com um conceito absoluto de verdade matemática. Essa visão costuma ser expressa de diferentes maneiras. Em geral, a confusão ocorre porque, tanto na linguagem I como na linguagem II, Carnap faz questão de incluir a matemática na parte lógica (determinada; não-sintética) da linguagem – ou assim se acredita.

Essa crença, de fato, não é falsa, e encontra pleno respaldo nas partes de *SLL* em que Carnap efetivamente desenvolve os conceitos de analiticidade para a linguagem I e II. Daí a sua força interpretativa. Ela necessita, contudo, ser devidamente modulada (ou relativisada) de acordo com o projeto geral da obra, e de acordo com outras tantas passagens igualmente explícitas. O mais correto, então, seria dizer que Carnap faz questão de incluir *aquilo que dentro da linguagem I e da linguagem II conta como matemática* na parte lógica (determinada; não-sintética) *dessas linguagens*. Mas nem essas linguagens são as únicas possíveis ou válidas (embora desempenhem o importante papel de formalizar conceitos extremamente relevantes: o conceito de verdade aritmética[145] e o de verdade da matemática clássica); nem existe uma única estrutura universal, igual para todos os sistemas formais, que deva necessariamente ser chamada de "matemática"[146].

A questão torna-se um pouco mais complicada quando se trata de avaliar o papel da aritmética – e do conceito de verdade aritmética – para o sistema sintático de *SLL*. Já discutimos longamente a importância do teorema de Gödel para a maneira como Carnap concebe o método formal (ver Capítulo 2, seção III, e principalmente as seções I e II

[144] *SLL*, págs. 46 e 47.
[145] Do qual trataremos logo adiante.
[146] Essa posição que defendemos fica ainda mais clara quando se considera que, em *SLL*, Carnap mostra que não é sequer necessário que a "matemática" de uma linguagem (referimo-nos aqui às sentenças da linguagem na qual figurem expressões numéricas, mas mesmo isso *quando e se* a linguagem em questão possuir uma estrutura *aritmética*, o que também não é obrigatório) faça parte da parte lógica da linguagem. De fato, ele mostra que, segundo a sua concepção, isso é exatamente o que acontece em todos os sistemas lógicos tradicionais. Nessas linguagens tradicionais, a sentença de Gödel não é demonstrável, ou seja, não é analítica (nessas linguagens, os dois conceitos coincidem). Mas a sentença de Gödel – excluída da parte lógica dessas linguagens – é claramente uma sentença matemática e, mais ainda, uma sentença aritmética.

acima). Apesar dessa importância seminal, nosso raciocínio pode seguir aqui uma linha semelhante à que acabamos de expor, em relação ao conceito de "verdade para a matemática clássica".

Segundo esse ponto de vista, o conceito de "verdade aritmética" seria dado dentro de uma teoria específica, a teoria aritmética clássica. Por se tratar de uma teoria extremamente importante, do ponto de vista da ciência e da história do pensamento teórico em geral, é que se torna necessário captar esse conceito de maneira formal exata. Isso Carnap faz pela extensão do método formal, para incluir regras indefinidas de transformação, e pela formulação de conceitos de analiticidade (para a linguagem II e também para a linguagem I) que, ao contrário do que acontecia nos sistemas tradicionais, são suficientes para captar as tais "verdades aritméticas".

Há, não obstante, uma dificuldade com essa maneira de entender a situação, que precisamos esclarecer. Carnap não somente procura obter um conceito formal capaz de traduzir a noção de "verdade aritmética". Ele usa a aritmética como instrumento de formalização das linguagens. Quase toda a parte II de *SLL* (seções 18 a 25) é dedicada a explicar o método de aritmetização de Gödel, no qual Carnap julga ver uma importante ferramenta da análise formal. Trata-se de uma ferramenta importante porque fornece uma teoria combinatorial bastante poderosa e suficientemente desenvolvida, pronta para ser utilizada. A esse respeito, eis o que ele escreve:

> *"If this method of arithmetization is not applied, certain difficulties arise in the exact formulation of syntax. (...) In order to be able to express such a sentence about possibility in the non-arithmetized syntax (...), the syntax would have to be supplemented by a theory (not empirical but analytic) concerning the possible arrangement of any elements – that is to say, by pure combinatorial analysis. It proves, however, to be much simpler, instead of constructing a new combinatorial analysis of this kind in a non-arithmetical form, to use the arithmetic of the natural numbers which already contains within itself the whole of combinatorial analysis (...)."* [147]

[147] *SLL*, págs. 57 e 58.

O que parece se anunciar, nesse sentido, é uma utilização transcendental da aritmética, no sentido de ser ela uma teoria capaz de conter a própria idéia de formalização (por meio do método de aritmetização de Gödel). Segundo esse ponto de vista, a aritmética poderia ser encarada como metalinguagem universal que permite a Carnap formalizar e analisar a estrutura de diferentes linguagens. Essa é, mais ou menos, a opinião defendida por Proust, em artigo sugestivamente intitulado "Fomal Logic as Transcendental in Wittgenstein and Carnap" [148]. Eis o que ela escreve, no trecho para nós mais significativo de sua análise:

> *"In the third place, arithmetization permits Carnap to obtain what one would expect from a transcendental doctrine, namely to specify* a priori *the set of possibilities. Since arithmetic already contains the whole of the calculus of permutations and combinations, it allows us to express such concepts as demonstrability and undecidability (1937, sec. 19, p. 57). What Gödel's theorems signify is that arithmetic carries with it its own combinatorial means. The first theorem proves not only that the system of arithmetic is incomplete, but that is* incompletable*: (...). Arithmetization thus confers on logical syntax a meaning that a non-arithmetized, descriptive syntax could never attain. By its means the operational possibilities of a formal system can be expressed."* [149] (destaques da autora)

Não nos parece, contudo, que uma tal interpretação "transcendental" seja a melhor opção em relação aos esforços desenvolvidos por Carnap. Em primeiro lugar porque, como a passagem anterior de Carnap deixa claro, a aritmética fornece uma teoria combinatorial bastante útil, mas não a única possível. Carnap menciona especificamente a possibilidade de desenvolver outra teoria combinatorial, igualmente não-empírica (analítica), para analisar as possibilidades sintáticas de uma linguagem. A aritmética, nesse sentido, representa uma teoria que se encontra felizmente à mão, plenamente desenvolvida e

[148] [Proust, 1987].
[149] [Proust, 1987], pág. 509.

suficientemente rica, mas não uma teoria necessária de um ponto de vista transcendental qualquer.

O desenvolvimento de linguagens formais, por outro lado, pode perfeitamente ser realizado sem auxílio da aritmética. É precisamente isso o que Carnap faz, inicialmente, para a linguagem I (a aritmetização da sintaxe da linguagem I aparece somente em um momento posterior, depois que suas regras já foram descritas); é isso o que ele faz também para a parte definida da linguagem II (regras definidas de formação e transformação, descritas nas seções 26, 29, 30 e 31). A idéia de formalização, assim, não parece depender em nenhum sentido da teoria aritmética. No Capítulo 1 deste trabalho, nós procuramos, precisamente, examinar os pressupostos da concepção formal de Carnap. Mostramos, então, que ela se constitui com base na possibilidade de realizar certas operações sobre símbolos e seqüências finitas de símbolos, e que somente isso basta para instituí-la como tal.

Finalmente, devemos examinar o papel desempenhado pela aritmética na obtenção de resultados metateóricos específicos. Estamos nos referindo, está claro, aos resultados obtidos por meio do método de aritmetização de Gödel. Se esse método, conforme sustentamos no parágrafo anterior, não é imprescindível para a formalização propriamente dita das linguagens, poderia ser que mesmo assim que ele se revelasse imprescindível de algum outro ponto de vista, como para permitir a obtenção de algum resultado indispensável.

O resultado básico obtido com o método de aritmetização, o teorema de incompletude de Gödel, é um resultado a respeito da própria aritmética, e de um conjunto específico de proposições, as verdades aritméticas. Ele afirma que uma grande classe de cálculos formais não é capaz de captar esse conjunto de proposições, verdadeiras para a aritmética. A resposta de Carnap é, nesse sentido, a mesma que já havíamos verificado em relação à matemática clássica: Ele reconhece a importância de apreender, por meios formais, especificamente esse conjunto de proposições; e ele desenvolve duas linguagens em que todas as verdades aritméticas aparecem como sentenças analíticas.

Mais ainda, por meio do conceito de analiticidade para a linguagem II, Carnap consegue apreender o exato significado em que as sentenças da aritmética são válidas: como sentenças a respeito de um domínio específico, que é o domínio de valorações para as variáveis numéricas de nível zero. Como vimos, esse domínio de valorações fornece (juntamente com os axiomas da aritmética[150]) uma espécie de modelo para a aritmética *standard*, embutido no próprio aparelhamento simbólico da linguagem II.

A maneira como Carnap trata esse assunto, na seção 58 da sintaxe geral (intitulada "Arithmetic"), parece confirmar nossa opinião. Nessa seção, Carnap analisa de maneira geral as estruturas a que chama de "aritmética". Trata-se de estruturas do tipo fornecido pelo conjunto A (juntamente com certas sentenças iniciais) para as linguagens I e II. Basicamente, uma aritmética compõe-se de uma classe de expressões numéricas ordenadas, acrescidas da possibilidade de definir as operações usuais de soma e produto. Qualquer linguagem, em sua conformação simbólica e formal, pode apresentar *ou não* uma estrutura aritmética desse tipo. Se apresentar uma tal estrutura, pode *ou não* possuir um conceito formal capaz de apreender o conjunto das sentenças verdadeiras da aritmética no sentido do teorema de Gödel. Todos os sistemas tradicionais, por exemplo, não conseguiam capturar essa importante classe de proposições, mas nem por isso deixavam de fazer afirmações aritméticas.

Como vimos na seção II deste capítulo, o grande problema residia justamente em encontrar o sentido exato em que as proposições de Gödel podiam ser ditas verdadeiras[151]. E Carnap, ao desenvolver seu conceito de analiticidade para a linguagem II, encontrou esse sentido. Novamente, devemos afirmar que ele forneceu uma definição formalmente correta dessa classe de sentenças verdadeiras. Do ponto de vista da lógica atual – *pós* teoria de modelos – pode-se dizer que essas sentenças são verdadeiras quando interpretadas como sentenças a respeito de um modelo *standard* dos números naturais. Pelo sistema de valorações, Carnap obtém exatamente isso: um modelo *standard* para os naturais. Ele consegue formalizar corretamente, portanto, o sentido em que as sentenças de Gödel são sentenças verdadeiras da *aritmética* (sentenças verdadeiras de um modelo específico de aritmética).

---

[150] Ver nota 132 acima.

[151] Proust escreve que *"the first theorem proves not only that the system of arithmetic is incomplete, but that is* incompletable". Ela esquece, no entanto, de perguntar: "Incompletável" em que sentido? Aliás, o sistema de *qual aritmética*? São essas as perguntas importantes que o teorema de Gödel traz à tona, mais do que o resultado em si.

Em nenhum momento, porém, o sistema de Carnap obriga-nos a readmitir qualquer noção de verdade absoluta – mesmo a simples noção de verdade *aritmética* absoluta. Diferentes linguagens podem possuir diferentes conceitos de analiticidade; mais ainda, é perfeitamente aceitável que diversas dentre essas linguagens não possuam nem sequer algo a que se possa chamar de "aritmética", ou seja, são absolutamente incapazes de realizar afirmações aritméticas. Nenhuma delas goza de prioridade sobre as outras, a não ser do ponto de vista da utilização dentro de domínios específicos de pesquisa, sejam esses domínios empíricos ou não.

Eis, então, o que acontece em relação às linguagens I e II. Carnap – podemos agora afirmar com maior clareza – aceitaria que a linguagem I e a linguagem II são linguagens mais adequadas do que outras para lidar com a teoria matemática usualmente chamada de "aritmética", no sentido de que o conceito de analiticidade adotado por cada uma delas (por caminhos, aliás, bem distintos) corresponde ao conceito de verdade que costuma ser utilizado – sem, contudo, estar devidamente formalizado – dentro dessa teoria (e que se revela, como Carnap mostrou, mas não *explicou* adequadamente, o conceito de verdade em determinado modelo para os números naturais, não à toa chamado de "modelo *standard*"). Nesse mesmo espírito, Carnap aceitaria que a linguagem II mostra-se mais adequada para lidar com a teoria da matemática clássica, porque seu conceito de analiticidade captura corretamente o conceito de verdade normalmente aceito nessa teoria.

Fiel ao Princípio de Tolerância que informa toda a sua abordagem sintática, porém, Carnap não se vê em nenhum momento obrigado a aceitar uma noção transcendente ou absoluta de verdade – seja ela verdade aritmética ou verdade matemática.

VIII

A adoção do Princípio de Tolerância, peça central de sua visão sintática para a lógica, determina portanto que Carnap rejeite qualquer noção *absoluta* de verdade. E isso vale, como argumentamos acima, até mesmo para noções tão básicas como as noções de verdade matemática (verdade na matemática clássica) e de verdade aritmética, para as quais Carnap encontra um equivalente formal adequado. Em face dessa constatação, podemos agora examinar melhor o erro que Carnap cometerá ao avaliar o que está em jogo na construção que ele próprio desenvolve para o conceito de "analiticidade" da linguagem II.

A noção de verdade esteve constantemente ligada, na história da filosofia, a um ponto de vista absolutista. Ao rejeitar a noção de verdade absoluta – extra-sintática e não-formalizável – Carnap acredita que deva rejeitar qualquer noção de verdade. Como veremos, ele é bastante explícito a esse respeito.

A seção 60b de *SLL*, na qual Carnap trata desse assunto (a seção intitula-se "The Concepts 'True' and 'False'"), inicia-se com um argumento bastante semelhante ao argumento apresentado por Tarski na primeira parte de *Der Wahrheitsbegriff in den formalisierten Sprachen*. Carnap mostra que a inclusão dos predicados "verdadeiro" e "falso" dentro de uma linguagem (desde que dotados de certas propriedades usuais), e quando se pode aplicá-los às expressões dessa linguagem, leva a contradições. Ele conclui a exposição da seguinte maneira:

> *"This contradiction* [a contradição que ele mostrou surgir em relação ao uso dos predicados "verdadeiro" e "falso"] *only arises when the predicates 'true' and 'false' referring to sentences of a language S are used in S itself. On the other hand, it is possible to proceed without incurring any contradiction by employing the predicates 'true (in S_1)' and 'false (in S_1)' in a syntax of S_1 which is not formulated in S_1 itself but in another language S_2."* [152] (colchetes meus)

Vemos aqui, mais uma vez, que Carnap anda sempre muito próximo aos resultados de Tarski. Em mais de um sentido, ele chega perto de desenvolver uma teoria adequada para tratar a questão da verdade. Infelizmente, a passagem acima prossegue da seguinte maneira:

> *" (...) A theory of this kind formulated in the manner of a syntax would nevertheless not be a genuine syntax.* For truth and falsehood are not proper syntactical properties; *whether a sentence is true or false cannot*

[152] *SLL*, pág. 216.

generally be seen by its design, that is to say, by the kinds and serial order of its symbols." [153] (destaque do autor)

Coffa considera esse argumento um dos mais fracos apresentados por Carnap em toda a sua carreira[154]. O argumento, de fato, não se sustenta. Para entender o equívoco cometido por Carnap, porém, devemos compreender claramente a intenção com que ele é proposto. Propriedades sintáticas, que para Carnap são as únicas aceitáveis em lógica, são propriedades internas de cada linguagem; o conceito de "verdade" – segundo o argumento parece indicar – há de ser sempre um conceito estabelecido de maneira externa à linguagem, e por isso jamais será sintático.

A afirmação de Carnap, assim compreendida, estaria correta. O problema reside no fato de ele concluir, a partir daí, que a noção de verdade deva ser completamente rejeitada. Carnap rejeita a noção de verdade porque acredita que ela não é formalizável, e é nesse ponto que comete seu maior erro. Para enxergar esse ponto, basta atentar para a última sentença do trecho citado acima: "*whether a sentence is true or false cannot generally be seen by its design, that is to say, by the kinds and serial order of its symbols*". Sabemos que, no esquema de *SLL*, uma teoria que possa ser construída somente com referência a símbolos e seqüências de símbolos é uma teoria formal; a rejeição dos conceitos "verdadeiro" e "falso", portanto, é uma rejeição da possibilidade de formalizar adequadamente esses conceitos.

Acontece que, como vimos, o próprio Carnap havia construído definições formais adequadas para certas classes de verdades: as verdades aritméticas (linguagem I e II); e as verdades da matemática clássica (linguagem II). Como explicar, então, sua rejeição da possibilidade de formalizar esse conceito? Carnap – devemos reconhecer francamente – andou no limiar de desenvolver a idéia semântica de verdade que, suficientemente desenvolvida, conduz à teoria de modelos; mas não desenvolveu. Ele construiu definições que traduziam corretamente a idéia de verdade em um modelo – o modelo *standard* dos naturais, e o modelo clássico dos reais – mas não percebeu que havia construído modelos, dentro dos quais interpretou as sentenças da linguagem II.

[153] *SLL*, pág. 216.
[154] [Coffa, 1987], pág. 566.

Faltava a Carnap, do ponto de vista teórico, a noção relativa de *verdade em um modelo*[155]. O que as definições de Carnap conseguem fazer, se devidamente compreendidas (ver seções III e IV acima), é justamente captar a verdade de certas sentenças aritméticas e matemáticas em relação a certo modelo (o modelo construído em termos de valorações). Mas Carnap não dispunha da idéia geral de "modelo"; e não alcançou enxergar exatamente a noção de verdade em um modelo. Pelo contrário, a noção de verdade que se apresenta constantemente diante de Carnap é a noção *absoluta* de verdade, aquela mais freqüente em filosofia da ciência, e contra a qual ele lutava abertamente.

A confusão que Carnap faz pode ser vista também sob outro ponto de vista, estreitamente relacionado ao anterior. Trata-se agora de constatar uma confusão entre, de um lado, a divisão entre aspectos internos e externos de uma linguagem, e, do outro, aspectos formais e não-formais de uma linguagem. Pode-se dizer, em outras palavras, que a oposição interno/externo, relativa a uma linguagem, não fica devidamente distinguida da oposição formal/não-formal. Vejamos o que isso significa.

A intenção básica e original de Carnap, aquela que informa todo seu projeto filosófico em *SLL*, é manter-se estritamente dentro do campo formal de considerações. Ele rejeita, no que diz respeito a sistemas simbólicos lógicos, qualquer consideração não-formal[156]. Ao fazer isso, no entanto, Carnap acaba rejeitando, não apenas o campo não-formal, mas igualmente o campo de considerações externas à linguagem. Isso, no entanto, não é necessário – fato que Carnap não percebeu. A existência das teorias semânticas formais prova justamente esse ponto. Uma teoria de modelos nada mais é do que uma teoria estritamente formalizada que define a verdade das proposições de uma linguagem com base em um elemento exterior a ela: o modelo. Carnap não viu isso porque o modelo que construiu para os naturais (e, a partir dele, para os reais) compõe-se, conforme já observamos, de expressões sintáticas internas à linguagem II (o conjunto A de expressões numéricas). Esse modelo encontra-se, por assim dizer, embutido no próprio aparelhamento

[155] A idéia de "modelo", embora hoje bastante difundida, está longe de ser óbvia. Basta considerar que, mesmo após Tarski ter desenvolvido, ainda na década de 1930, um ponto de vista abertamente semântico que foi rapidamente assimilado por boa parte dos melhores lógicos (e também por Carnap), somente na década de 1950 veio a surgir (pelas mãos do próprio Tarski, e colaboradores) algo que realmente pudesse ser chamado de teoria de modelos.
[156] Exceção feita àquelas considerações abertamente informais, mas apenas como passo necessário na direção de sua formalização, vale dizer, na obtenção de sistemas e conceitos sintáticos formais.

simbólico da linguagem, de maneira que se tornou fácil para Carnap perder de vista sua exata natureza.

Podemos resumir a situação, então, da seguinte maneira. Carnap deseja eliminar todas as considerações *não-formais* acerca de uma linguagem. Por uma transformação característica, no entanto, acaba defendendo a eliminação – no que diz respeito ao estudo da estrutura lógica da linguagem – de todas as considerações externas a ela. O conceito de "verdade" é rejeitado, então, justamente por esse motivo (que é motivo errado): porque, seja elaborado da maneira que for, remete a uma esfera exterior à linguagem. Com diz o próprio Carnap: "*For truth and falsehood are not proper syntactical properties*". E, de fato, "verdade" e "falsidade" não são conceitos sintáticos (se por sintaxe se entende, como é usual, um estudo puramente interno acerca das possibilidades combinatoriais das expressões de uma linguagem). Não obstante, "verdade" e "falsidade" podem ser formalizados.

Vemos assim que Carnap deseja limitar-se à esfera sintática justamente na medida em que acredita que essa é a única esfera formalizável, como também a única esfera em que se pode derrubar o conceito absoluto de verdade. Nos dois casos, ele está errado. Do seu ponto de vista, as definições de analiticidade que ele mesmo constrói para a linguagem I e II captam certos conjuntos de sentenças, como poderiam captar outras. Ele sabe que, especificamente, elas captam dois conjuntos importantes de "verdades" – as "verdades" aritméticas e as "verdades" da matemática clássica. Contudo, em vez de perceber que sua definição de analiticidade para a linguagem II havia conseguido traduzir o exato sentido em que essas sentenças são de fato "verdades" da aritmética e da matemática clássica, ele acredita simplesmente que encontrou um equivalente formal para duas classes de sentenças até então não devidamente formalizadas. Esse equivalente que ele encontra, na sua opinião, é oferecido não apenas por meios puramente formais, mais por meios puramente sintáticos, ou seja, internos à linguagem.

Resta, finalmente, perguntar: Existem esses equivalentes sintáticos que Carnap julga haver encontrado? Em relação à definição de analiticidade para a linguagem I, podemos responder à pergunta definitivamente na afirmativa. Essa definição de analiticidade para a linguagem I, com efeito, é uma definição estritamente sintática, feita diretamente por meio de uma regra indefinida de transformação. Com relação à definição de analiticidade para a linguagem II, a situação é mais complicada. Tal como Carnap a constrói, nós já sabemos, ela é formal, mas não é sintática. É formal porque faz referência às sentenças somente em sua qualidade de seqüências ordenadas de símbolos. Não é sintática porque não é formulada em termos de regras de conseqüência direta (como é exigido por sua própria sintaxe geral); e porque faz recurso a um modelo externo à linguagem (o modelo dado pelas valorações), em relação ao qual a analiticidade das sentenças fica definida. Constitui-se, assim, como definição semântica (ver seção V acima).

Carnap, no entanto, define retroativamente um conceito de "conseqüência" para a linguagem II, com base na definição de analiticidade (ver Capítulo 4, seção IX). A esse respeito, porém, devemos reparar em dois pontos: 1) O conceito de "conseqüência" definido por Carnap *não* pode ser definido diretamente por meio de regras de transformação[157], e portanto não poderia possuir um equivalente sintático diretamente definido; e 2) Carnap *não* define um conceito de "conseqüência direta" para a linguagem II, do tipo que, segundo o esquema da sintaxe geral, representa o conceito fundamental de qualquer abordagem sintática.

Nesse sentido, devemos ainda considerar a hipótese de que pudesse haver algum conceito sintático adequado de "conseqüência direta" (construído em termos de regras indefinidas de transformação), capaz de gerar exatamente o conceito de conseqüência da linguagem II e, portanto, o conceito de analiticidade dessa linguagem. Essa hipótese é bem mais difícil de ser descartada. Se ela puder ser aceita, isso significaria que o conceito de analiticidade para a linguagem II, embora definido de maneira semântica, seria ao menos *compatível* com algum possível conceito propriamente sintático de "conseqüência direta", que Carnap no entanto não chegou a construir. Nós não temos condições de responder definitivamente a essa questão; mas, em vista das diferentes circunstância teóricas que já expusemos, consideramos como no mínimo improvável que uma tal definição sintática de conseqüência direta possa, finalmente, ser obtida para a linguagem II.

[157] Como vimos, não há sentenças suficientes na linguagem II (nem em nenhum outra linguagem composta por expressões finitas) para deduzir diretamente a sentença "M(F)" (em que F é uma variável para predicados numéricos).

Conclusão

I

Nosso estudo de *SLL* nos conduziu por um percurso que, com toda razão, poderíamos chamar de surpreendente. Na obra, confluem algumas das principais questões da lógica simbólica do século XX, muitas das quais formuladas de maneira pioneira ou – o que muitas vezes vem a ser o mesmo – em antecipação a seu tempo. O livro combina um domínio incomum das técnicas matemáticas em lógica (poucas vezes igualado por outros filósofos) com extremo rigor conceitual, além de notável abrangência no tratamento de uma grande variedade de temas lógicos e matemáticos; no entanto, sua concepção final torna-se difícil de sustentar, e isso acontece justamente devido a certos problemas técnicos e conceituais da abordagem proposta por Carnap.

Toda essa duplicidade, de resultados e – devemos salientar – também de intenções, coloca-nos diante de um enigma: Como situar o projeto de *SLL* dentro da história da lógica contemporânea? No presente trabalho, esperamos ter contribuído justamente para uma melhor apreciação dessa questão. Foi com esse objetivo que, a partir da análise direta do texto de Carnap, detectamos algumas características que podem ajudar, segundo acreditamos, a esclarecer as contradições que se verificam na obra e que constituem, também elas, parte de sua força e riqueza teórica.

Com efeito, o projeto lógico-filosófico elaborado por Carnap em *SLL* – o qual se revela *também*, mas somente em certo sentido, um projeto de fundamentação da matemática – caracteriza-se por uma tensão fundamental que nós procuramos identificar e examinar. Essa tensão opõe, de um lado, a visão sintática da lógica informada pelo Princípio de Tolerância, que constitui a peça mais importante do projeto filosófico defendido na obra. Do outro lado, encontramos a inovadora definição oferecida por Carnap para o conceito de analiticidade da linguagem II. Essa definição, na qual aparecem claros elementos de uma teoria semântica, acaba ocupando lugar de destaque no projeto de fundamentação matemática que, embora com escopo bem diferente do usual, deve ser igualmente reconhecido no texto.

Em relação ao primeiro termo dessa tensão, é interessante observar o seguinte. A visão filosófica defendida por Carnap em *SLL* – mas não só em *SLL* – mostra-se tão relevante para a evolução dos estudos filosóficos no século XX que pode ser vista como uma das raízes de toda a filosofia analítica que dominou os países de língua inglesa a partir da segunda guerra mundial. Obviamente, não temos condições de nos debruçar aqui sobre esse interessante tema da história da filosofia[158] – nem é esse nosso objetivo. De maneira geral, no entanto, podemos mencionar alguns aspectos de *SLL* que perduraram, não apenas na obra posterior de Carnap, como na ampla tradição filosófica que ele ajudou a construir.

De um ponto de vista mais específico, cabe destacar a proposta, tão arduamente defendida em *SLL*, de estudar filosofia somente a partir de uma análise cuidadosa, não-dogmática, da linguagem na qual as proposições filosóficas se expressam. Esse estudo não deve restringir-se a classificações categoriais supostamente *a priori*. Ele deve esquadrinhar, em toda sua complexidade, os mecanismos lingüísticos em sua variada relação com a atividade cognitiva, ou seja, como meio de apreender, organizar e expressar os resultados dessa atividade.

De um ponto de vista um pouco mais amplo, podemos destacar: a exigência de correção formal, que constantemente acompanha as análises de Carnap; sua preocupação constante com a clareza e precisão dos resultados, examinados sempre no máximo de detalhes; e sua rejeição de qualquer discussão transcendental ou *a priori*, sem um conteúdo que possa ser clara e diretamente relacionado à prática científica concreta, como prática por excelência voltada a conhecer o mundo (podemos identificar essa última tendência com a sua decantada tendência antimetafísica).

Em *SLL*, é bem verdade, todo esse conjunto de posições defendidas por Carnap ganha uma roupagem (uma formulação) estritamente *sintática*, que em pouco tempo revelou-se difícil de sustentar. Sintomaticamente, a teoria sintática da lógica e da filosofia veio a ser completamente renegada pelo próprio Carnap, em menos de uma década. Nesse contexto, o Princípio de Tolerância lógica ocupa um lugar extremamente relevante. Ele representa uma parte importante daquela porção da visão filosófica de *SLL*, aliás nada

[158] Ver, a esse respeito, [Friedman, 2000].

desprezível, que conseguiria ultrapassar as limitações (possivelmente insuperáveis) do sistema exposto na obra.

Já em relação ao segundo termo da tensão fundamental que apontamos em *SLL*, também cabe observar alguma coisa. A definição que Carnap oferece para o conceito de analiticidade de sua linguagem II é uma definição – conforme procuramos mostrar – claramente semântica segundo os métodos que aplica (embora não o fosse segundo a intenção de seu autor). Como tal, ela consegue captar exatamente uma das coisas que uma definição semântica é capaz de captar: a idéia de verdade *dentro de um modelo*. No caso, Carnap constrói – ainda que sem a devida consciência teórica – um verdadeiro modelo para os números reais e demais entidades da análise matemática clássica (funções de reais etc.). Ele faz isso a partir de um modelo *standard* para os números naturais, que se encontra embutido no próprio aparelhamento simbólico da linguagem II, acrescido de uma interpretação extensional para a teoria dos conjuntos.

Outro aspecto importante a considerar é que Carnap não desenvolve sua definição de analiticidade para a linguagem II no sentido até então usual que se poderia esperar de uma fundamentação para a matemática. Para Carnap, trata-se de uma tarefa localizada: encontrar um critério de validade *para a matemática clássica*. Do ponto de vista geral de *SLL*, no entanto, e de acordo com o Princípio de Tolerância, a matemática clássica não ocupa um lugar absoluto, como única ciência (supostamente *a priori*) dos números e das quantidades. Carnap rejeita, explicitamente, a discussão que opunha, em diferentes campos, construtivistas, intuicionistas, formalistas, matemáticos de linha mais tradicional ou ainda outras tendências fundacionistas. Para ele, a matemática clássica merece atenção particular por constituir um corpo bem desenvolvido de teoria, extremamente útil para a ciência contemporânea. Nesse sentido é que seria necessário encontrar uma linguagem – uma estrutura lógica – capaz de captar, por meio de seu conceito de analiticidade (ou seja, na parte determinada da linguagem), as proposições dessa teoria.

A linguagem II, porém, não é a única linguagem possível; e nem a abordagem pressuposta pela matemática clássica corresponde à única maneira possível de tratar as estruturas numéricas de uma linguagem (segundo uma específica concepção de estrutura numérica, que Carnap baseia no conceito de “aritmética”, formulado dentro da sintaxe geral), em sua qualidade de componentes de uma descrição científica do mundo. Encontramos, nesse ponto, uma das questões em que as propostas de Carnap revelam-se mais avançadas. O modo como ele defende, explicitamente, a livre transição entre sistemas lógicos e matemáticos distintos, cuja única medida estaria na possibilidade de aplicação a diferentes áreas do conhecimento, reflete-se atualmente na aceitação localizada de diferentes "lógicas", como a lógica quântica e outras, e de diferentes abordagens para matemática, incluindo abordagens com alto grau de restrição epistêmica, dentro da economia, ciências sociais, teoria da informação etc.

II

A tensão que nós detectamos na obra de Carnap é, de certa maneira, evidente. A compreensão adequada de sua definição de analiticidade para a linguagem II, e principalmente sua posição face ao restante da obra, já se tornou um tema clássico nos estudos críticos a respeito do positivismo lógico. Os termos em que o problema se manifesta, nesse sentido, são claros: A definição de analiticidade para a linguagem II apresenta elementos nitidamente semânticos, dentro de uma obra cujo propósito essencial deveria ser a defesa de uma visão sintática da lógica e da linguagem. De certa maneira, em nosso trabalho, tentamos apresentar um quadro claro dentro do qual formular e compreender justamente essa tensão.

O primeiro elemento desse quadro é aquilo que poderíamos chamar, de certa maneira, de uma "solução" da tensão. Essa solução deve ser vista, em um primeiro momento, como uma solução abertamente externa ao texto de Carnap. Em outras palavras: trata-se de buscar compreender, de um ponto de vista atual – com auxílio de certo aparato técnico e conceitual que se desenvolveu depois da publicação de *SLL*, mas de cuja história *SLL* representa uma etapa –, os verdadeiros problemas com que luta o sistema de Carnap, principalmente em sua abordagem para a matemática, e mostrar como eles podem ganhar uma expressão articulada e coerente, pois somente assim será possível penetrar os méritos e deficiências da obra.

Dessa maneira, a "solução" que procuramos oferecer deve: 1) tomar como ponto de partida o verdadeiro foco em torno do qual se expande a tensão, vale dizer, a tentativa de

Carnap de encontrar um "critério de validade" para a matemática clássica por meio do conceito de analiticidade para a linguagem II; e 2) tentar colocar essa definição – que não pode ser descartada, em sua natureza semântica, como simples erro ou desvio mas, pelo contrário, deve ser tomada como indicativa de um importante movimento dentro da obra – em harmonia com o projeto filosófico geral de *SLL*, naquilo em que ele ultrapassa a insistência na abordagem sintática, e que determina mesmo essa abordagem sintática (e que, portanto, pode e deve ser visto como mais essencial do que ela).

Não é possível, de fato, colocar um método essencialmente semântico como aquele utilizado para definir analiticidade para a linguagem II em harmonia com a formulação explicitamente sintática da lógica defendida por Carnap em *SLL*. Não é aí, portanto, que devemos procurar a compatibilidade. Contudo, é possível harmonizar esse método semântico (que em última instância supera a abordagem sintática, tanto em termos históricos como na própria evolução do pensamento de Carnap) com outras tendências filosóficas manifestadas em *SLL*, as quais se revelam, precisamente nesse sentido, mais fundamentais que o credo sintático. Estamos nos referindo, neste passo, ao impulso teórico contido no Princípio de Tolerância e, de maneira geral, à exigência de uma abordagem lingüística e formal (baseada em um cálculo simbólico suficientemente claro) para a lógica e a filosofia.

Em vista dessas considerações, os próprios termos em que formulamos o problema apontam sua solução. Eis como podemos ver a situação: Carnap desenvolveu um modelo para a aritmética e para a matemática clássica por meio do método das valorações; e pôde aceitar esse método como parte do esquema de *SLL* – um método que, como sabemos, não é sintático – porque o reconheceu como método devidamente formalizado. Carnap acreditou, certamente, que sua construção pudesse manter-se no plano da pura sintaxe. Essa confusão tornou-se possível, porém, somente porque, antes de ser sintática ou semântica, sua construção era formal. Em outras palavras, a natureza semântica desse método ficou obscurecida porque sua formalização pareceu a Carnap suficientemente adequada, e ele associava a idéia de formalização com a idéia de sintaxe.

Desse modo, a abordagem por meio de modelos – implícita na definição de analiticidade para a linguagem II e, de maneira geral, no tratamento que Carnap confere às discussões matemáticas – pode ser compatibilizada, em certa extensão, com os pressupostos filosóficos de *SLL*. A extensão dessa compatibilidade, mais ainda, é a maior que poderia ser, em vista das restrições apontadas.

Tal abordagem, em que se destaca o papel implícito que a construção de modelos desempenha em *SLL*, constitui uma solução para os problemas e contradições propostos à interpretação dessa obra, em mais de um sentido. Em primeiro lugar, a idéia de modelo resolve adequadamente a questão teórica básica proposta pelo teorema de incompletude de Gödel, na medida em que fornece um significado preciso para o conceito de "verdade" que aparece, de maneira essencial, nesse teorema. Isso é relevante para a compreensão de *SLL* porque Carnap, como mostramos, enfrenta-se diretamente com os resultados de Gödel (ele foi, aliás, um dos primeiros pensadores a entendê-los). Boa parte do livro é dedicada justamente à tarefa de encontrar uma maneira adequada de tratar esses resultados dentro de um sistema lógico formal; e é nesse percurso que ele desenvolve tanto seus conceitos sintáticos mais sofisticados como, finalmente, seu método semântico. Carnap encontra a resposta que buscava, e o faz precisamente ao construir modelos matemáticos.

Mais importante do que isso, porém, é o fato de que a teoria de modelos constitui uma solução para o verdadeiro coração das contradições que aparecem em *SLL*. Isso porque ela constitui uma teoria *formal* – segundo um significado da palavra "formal" que Carnap não demorou em aceitar – capaz de explicar e captar o problema mais amplo da *verdade* matemática, que Carnap abordou sob diversos aspectos em *SLL* (apesar de negar veementemente a relevância do conceito de "verdade"), e que afinal constituiu sua principal motivação na direção de desenvolver um método semântico.

O ponto essencial a destacar, em relação à situação que apontamos acima, é que Carnap rejeita a noção de "verdade", principalmente, por dois motivos. Por um lado, porque ele não acredita que ela possa ser devidamente formalizada. (Recordemos, aqui, a passagem de *SLL* em que ele rejeita, definitivamente, o conceito de "verdade": "*For truth and falsehood are not proper syntactical properties*; whether a sentence is true or false cannot generally be seen by its design, that is to say, <u>by the kinds and serial order of its symbols</u>."[159]) Por outro lado, Carnap rejeita a noção de "verdade" na medida em que rejeita

[159] *SLL*, pág. 216. O grifo, na passagem citada, é nosso.

– como parte de sua tendência antimetafísica – qualquer conceito absoluto, como acreditava que devesse ser o conceito de verdade.

Acontece, porém, que a idéia de "modelo" representa uma resposta adequada justamente para essas duas objeções. Por um lado, o conceito de "verdade" que aparece na teoria de modelos é um conceito relativo ("verdade" relativa a um modelo); por outro lado, ele pode ser, em grande medida, formalizado. É por isso que Carnap pôde trabalhar com uma definição semântica de analiticidade e não se dar conta do que realmente acontecia.

III

Apresentar a idéia de "modelo" (bem como idéias semânticas em geral) como uma solução *externa* para a tensão presente em *SLL*, contudo, não é suficiente. Por um lado, é fato bastante conhecido que Carnap não demorou a assimilar e defender os métodos de Tarski, reconhecendo claramente sua natureza semântica (métodos esses que conduziriam, cerca de duas décadas depois, à formulação de uma teoria completa de modelos). A assimilação das idéias semânticas de Tarski, mais ainda, ocorreu pouco depois da publicação de *SLL*, o que apenas confirma a hipótese de que o próprio Carnap teria encontrado, nesse novo tipo de abordagem, a solução (e a superação) dos problemas e contradições que havia enfrentado em sua obra, e que nela se revelam. Por outro lado, uma análise adequada da situação não pode, certamente, deter-se nesse ponto.

A questão essencial passa a ser, nesse sentido, a de determinar exatamente as forças internas, presentes no texto de *SLL*, em relação às quais a construção de certos modelos para a matemática se articulam. Existem duas questões principais que precisamos ponderar.

Em primeiro lugar, devemos ser capazes de mostrar que essa abordagem está realmente presente em *SLL*, e esclarecer, segundo critérios objetivos, o quão próximo Carnap chegou de uma teoria semântica. Em segundo lugar, devemos poder dizer alguma coisa a respeito das razões que contribuíram para que ele não visse, corretamente, a situação teórica que se havia descortinado, e os fatores que, de certa maneira, desviaram-no de desenvolver um ponto de vista *abertamente* semântico. Ao contrário da primeira, essa segunda questão possui uma natureza marcadamente subjetiva, e tem de permanecer, em última instância, no plano conjetural. Em nosso trabalho, tentamos nos dirigir a ambas essas linhas de indagação.

Como ponto inicial, nós mostramos que a construção da definição de “analiticidade” para a linguagem II utiliza um método indubitavelmente semântico: o método das valorações. Mais ainda, examinamos o sentido exato em que o sistema de valorações, baseado em uma interpretação extensional para a teoria dos conjuntos, corresponde à construção de um modelo para os números naturais, para os números reais e para todas as demais entidades lógicas de tipo superior (que encontram sua expressão matemática clássica dentro do campo da análise funcional). Verificamos também como, a partir justamente desse sistema de valorações – e da quantificação que é *interpretada* sobre esse sistema de valorações –, as sentenças da linguagem II são avaliadas e têm seu *status*, que Carnap acreditava sintático, determinado (como “analíticas”, “contraditórias” ou “sintéticas”).

Na seqüência de nossa análise, então, enfatizamos um fato importante, para o qual já chamamos a atenção um pouco mais acima (nas seções I e II desta Conclusão), mas que merece ser considerado, agora, sob novo enfoque. O desenvolvimento desse método de valorações, por parte de Carnap, pode e deve ser compreendido em relação com o Princípio de Tolerância, e isso mesmo depois de revelada sua natureza semântica.

O Princípio de Tolerância, como sabemos, é formulado primordialmente dentro do quadro teórico *sintático* de *SLL*, e por isso é muitas vezes esquecido na hora em que se analisa a definição de analiticidade para a linguagem II. É devido a essa cisão, entre a moldura sintática geral de *SLL* e a natureza semântica de seu tratamento para a matemática (aliada, muitas vezes, à desconsideração da verdadeira natureza da construção semântica de Carnap, como construção de certo *modelo* para a matemática), que muitos comentadores são levados a crer em uma visão absoluta acerca da verdade matemática por parte de Carnap. De acordo com nossa orientação, no entanto, acreditamos que seja equivocado manter esse ponto de vista, segundo o qual Carnap entretém alguma noção absoluta de verdade matemática, ou deseja incluir as proposições matemáticas definitivamente no campo lógico. Para Carnap, de acordo com o Princípio de Tolerância, nem sequer o campo lógico encontra-se definitivamente fixado. Pelo contrário: ele fica estabelecido de maneira somente relativa, com referência a cada linguagem em particular (entre as infinitas linguagens possíveis), como o campo essencialmente determinado *daquela* linguagem.

Tampouco a matemática, portanto, pode possuir um *status* definitivo. Nem todas as linguagens tratam a matemática da mesma maneira, e nenhuma possui privilégio intrínseco em relação às outras. Pior: aquilo que conta como "matemática", dentro de cada linguagem, nem sempre será a mesma coisa. Trata-se de estruturas possivelmente distintas dentro de cada sistema lógico, cujo ponto em comum é serem construídas a partir de um tipo específico de estrutura simbólica – que Carnap chama de "aritmética" – caracterizado de maneira exterior às linguagens. (A "aritmética" deve ser apreendida como determinado tipo de estrutura simbólica particular, e *infinita*; além do mais, ela deve ser *interpretada* como a estrutura de *todos* os números naturais).

Diante dessa situação, nascida da consideração fundamental do Princípio de Tolerância, a noção de que Carnap trabalha, ainda que implicitamente, com um verdadeiro modelo para a matemática ganha força. Essa idéia permite entender seu tratamento para as questões matemáticas, em que ele parece procurar e encontrar critérios de "verdade" (ou "validade", como ele prefere dizer) para a matemática, com a tendência básica ditada pelo Princípio de Tolerância. Nossa orientação interpretativa, nesse sentido, fica plenamente justificada, devido à sua capacidade de integrar os diversos momentos de *SLL*.

O modelo de matemática que Carnap chega a desenvolver de maneira tão sofisticada, mas com o qual trabalha apenas implicitamente, é o modelo pressuposto – e por muito tempo não devidamente esclarecido ou reconhecido como tal – pela matemática clássica. Trata-se, porém, apenas de um modelo. No sistema de Carnap (o qual é formulado de maneira sintática), assim como na teoria abertamente semântica de modelos, ele não deve ser visto como único.

Em *SLL*, portanto, as coisas se passam da seguinte maneira. Um modelo da matemática, dentro do qual as proposições numéricas da linguagem II são interpretadas, é fornecido pelo sistema de valorações. O fato de esse modelo não ser único se expressa – ou antes seu "paralelo" sintático se expressa – por meio do Princípio de Tolerância. Segundo esse princípio, a linguagem II não é a única possível, como tampouco é a única possível a matemática que por meio dela pode ser formulada, e que encontra seu "critério de validade" em uma definição de analiticidade que também não é a única possível. Qualquer visão absoluta a respeito da verdade matemática fica, desse modo, excluída – a despeito da impressão em contrário que a definição de analiticidade para a linguagem II possa dar.

As discussões acerca de como Carnap via a matemática, assim, tendem a se diluir, ou a ganhar contornos menos problemáticos. Carnap não via *a matemática* desta ou daquela maneira: Ele reconhecia, na *matemática clássica*, um corpo bem estabelecido de conhecimento, para o qual merecia ser encontrado um "critério de validade". Isso equivale, em seu sistema, a encontrar uma definição de analiticidade que consiga capturar, formalmente, dentro da parte definida (lógica) de uma linguagem específica, as sentenças da matemática clássica. Ele faz isso por meio da linguagem II e de sua definição de analiticidade; contudo, essa definição ultrapassa, ao contrário do que ele esperava, o plano sintático, pois é obtida somente por meio da construção de um modelo matemático específico dentro do qual as sentenças da linguagem II são interpretadas.

IV

Chegamos ao ponto, assim, em que tivemos de examinar a segunda questão que propusemos: Por que Carnap não viu corretamente o próprio caminho teórico que havia aberto de maneira tão inovadora e, ao mesmo tempo, tecnicamente tão precisa? Como já observamos, trata-se de uma questão cuja resposta tem de permanecer, finalmente, no plano conjetural. Nem por isso, contudo, será impossível apontar alguns indícios, mais ou menos claros, dos rumos que tomou o pensamento de Carnap. De nosso trabalho, surgem três possíveis explicações – três fatores que podem ter se combinado para afastar Carnap de uma teoria explicitamente semântica, de resto vislumbrada por ele em mais de um momento[160].

Em primeiro lugar, temos o fato, indicado logo acima, de que o Princípio de Tolerância desempenha – segundo o esquema sintático de *SLL* – o papel relativizador, em relação à fundamentação da matemática, que uma teoria de modelos tornaria patente. Embora uma tal relativização do problema de fundamentação da matemática (como aliás

[160] Na seção seguinte – a seção final desta Conclusão – ofereceremos uma longa citação que demonstra bem esse ponto.

dos problemas de fundamentação em geral) seja explicitamente defendida em *SLL*, e pudesse conduzir a uma teoria semântica (como conduziu no caso na definição de analiticidade para a linguagem II), Carnap podia dar-se por satisfeito com a formulação aparentemente sintática que havia encontrado para ela, por meio do Princípio de Tolerância.

Em segundo lugar, podemos identificar a rejeição que Carnap devotava – estritamente relacionada à sua adoção do Princípio de Tolerância – a qualquer tentativa de estabelecer um conceito absoluto, cuja discussão só pudesse ocorrer no plano metafísico (desvinculado, portanto, da atividade científica prática, voltada a conhecer o mundo a partir de alguma base empírica). Essa sua posição levou-o a rejeitar, de maneira renitente, o conceito de verdade. Como sabemos, porém, o conceito de verdade não precisa remeter, necessariamente, a um plano absoluto de considerações, como mostra justamente a teoria de modelos. Essa rejeição inicial e arraigada de Carnap[161], no entanto, fez com que ele não conseguisse enxergar a possibilidade de uma definição relativa de verdade (embora a tenha desenvolvido, concretamente, para a linguagem II).

Finalmente, detectamos em nosso trabalho uma característica técnica de *SLL* que pode ter contribuído para que Carnap perdesse de vista a natureza semântica de sua definição de analiticidade para a linguagem II. Trata-se de uma característica das linguagens I e II, as quais possuem uma "aritmética" (no sentido da sintaxe geral) dada pelo conjunto de símbolos $A = \{0, 0', 0'', \ldots , 0''''^{\cdots}' , \ldots \}$[162]. Esse conjunto faz parte do aparelhamento simbólico dessas linguagens. Como tal, representa uma estrutura sintática isomorfa aos números naturais, que pode ser utilizada, ela mesma, como modelo para esses números.

---

[161] Devemos lembrar a observação de Coffa, em [Coffa, 1987], segundo a qual a objeção de Carnap ao conceito de "verdade" funda-se no pior argumento oferecido pelo pensador em toda a sua carreira.

[162] Carnap fala que as linguagens I e II são "linguagens de posição". Não nos debruçamos sobre esse fato, em nosso trabalho, porque ele se revela de pouca conseqüência para as questões de que tratamos (sua principal relevância manifesta-se, possivelmente, em relação a certas questões existenciais em lógica, tratadas na seção 38a de *SLL*; de toda maneira, esse tema sequer volta a aparecer na sintaxe geral).
Na sintaxe geral, com efeito, a estrutura simbólica chamada de "aritmética", que pode estar presente (ou não) em uma linguagem qualquer, é capaz de assumir formas sintáticas bastantes complexas. Embora tal estrutura tenha de ser, como aliás é óbvio, isomorfa aos números naturais, ela pode manifestar-se, por exemplo, por meio de um conjunto de entidades lógicas de tipo superior, relacionadas por um funtor adequado.
Nesse sentido, o fato importante a constatar, em relação a linguagens de posição como são as linguagens I e II (mas não somente ou necessariamente em relação a elas), é que sua "aritmética" é formada por um conjunto bastante simples de expressões sintáticas: o conjunto A. Esse conjunto é composto por expressões de nível zero, e contém, além do mais, *todas* as constantes individuais das linguagens I e II, ou seja, todos os seus nomes individuais. Esse fato, certamente, contribui para obscurecer a situação.

Na hora em que Carnap constrói o seu método de valorações, utiliza os próprios símbolos do conjunto A como valoração para as variáveis numéricas (variáveis de nível zero). A partir daí é que ele elabora todo o sistema de valorações para tipos lógicos superiores, o qual funciona como verdadeiro modelo para a matemática clássica (o sistema de valorações constitui-se, mais precisamente, com base no conjunto A de símbolos e em uma interpretação extensional para a teoria de conjuntos). Assim – como já dissemos no corpo do nosso trabalho – o conjunto A passa a ter dupla existência: uma existência sintática como conjunto de símbolos da linguagem II; e uma existência semântica como conjunto de valorações para as variáveis numéricas dessa linguagem. Essa duplicidade ajuda a explicar, ao menos em parte, por que Carnap acreditou – de maneira certamente equivocada – que pudesse manter sua definição de analiticidade para a linguagem II, pioneira justamente devido ao método semântico que utiliza, no campo puramente sintático.

IV

Iremos apresentar uma passagem relativamente longa de *SLL* que parece confirmar, de maneira particularmente interessante, muitas de nossas opiniões. Trata-se de um trecho complexo, no sentido de manifestar diversas das contradições que nós tentamos indicar. Nele, Carnap dá a impressão de estar no limiar de um novo ponto de vista; expõe certas questões complicadas de seu sistema, e acredita poder encaixá-las no plano sintático de discussão; ao fazer isso, porém, acaba por identificar os exatos elementos que deveriam determinar a superação do plano sintático, ao mesmo tempo em que aponta na clara direção de uma teoria semântica.

A passagem foi retirada da seção 34d, Parte III de *SLL*. É nessa seção, precisamente, que Carnap expõe o conceito de analiticidade para a linguagem II. Ou melhor: é nessa seção que Carnap, após haver preparado o terreno ao longo das seções anteriores (nas quais explica as regras de redução, valoração e avaliação), termina de expor seu conceito de analiticidade, cuja definição ele finalmente fornece por meio de um novo conjunto de regras. Depois de formular a definição completa, Carnap alonga-se em algumas explicações e considerações a seu respeito. Ele finaliza a seção da seguinte maneira (nós iremos sublinhar alguns trechos que mais de perto nos interessam, para comentá-los ao final):

"A certain point in the given definition of 'analytic in II' may appear dubious. For the sake of simplicity we will consider the corresponding definition of 'analytic in II_1' [II_1 é a linguagem II restrita ao primeiro nível de tipos lógicos]. *Let a language S be used as a formalized syntax-language (for example, a more extensive region of II, or II itself). Since in II_1 free ^1p* [uma variável para predicados de primeiro nível, ou seja, para predicados numéricos] *(...) occur, the definition of 'analytic in II_1' (...) will contain phrases such as 'for every valuation for a $^1p^1$* [variável para predicados do primeiro nível, com apenas um argumento] *...'; this, according to VR1a and VR1c* [regras de valoração], *is the same as saying 'for all syntactical properties of accented expressions...'* ["expressões acentuadas" são as expressões do conjunto A, ou seja, as expressões numéricas – 0, 0', 0'' etc. – da linguagem II]. *Now what is meant by this phrase and how is it to be formulated in the symbolic language S? If we said instead merely 'for all syntactical properties which are definable in S...', then the definition of 'analytic in II_1' would not effect what is required of it. For just as for every language there are numerical properties which are not definable in it (see p. 106), so there are also syntactical properties which are not definable in S. Thus it might happen that the sentence 'C_1 is analytic in II_1' was true (analytic) in the syntax-language S , and yet false (contradictory) in a richer syntax-language S', namely if the phrase, 'for all definable syntactical properties...', contained in the criterion for that sentence, although valid for all the properties definable in S, was not valid for a certain property which is only definable in S'. Thus the definition must not be limited to the syntactical properties which are definable in S, but must refer to all syntactical properties whatsoever. But do we not by this means arrive at a platonic absolutism of ideas, that is, at the conception that the totality of all properties, which is non-denumerable and therefore can never be exhausted by definitions, is something which subsists in itself, independent of all construction and definition? From our point of view, this metaphysical conception (...) is*

definitely excluded. We have here absolutely nothing to do with the metaphysical question as to whether properties exist in themselves or whether they are created by definition. The question must rather be put as follows: can the phrase 'for all properties...' (interpreted as 'for all properties whatsoever' and not 'for all properties which are definable in s') be formulated in the symbolic syntax-language S? This question may be answered in the affirmative. The formulation is effected by the help of a universal operator with a variable p, i.e. by means of '(F) (...)', for example. (That this phrase has in the language S the meaning intended is formally established by the fact that the definition of 'analytic in S' is formulated in the wider syntax-language S_2, again in accordance with previous considerations (pp. 106 f.), not by substitutions of the predicates ob S, but with the help of valuations.) This is correspondingly true for the valuations of higher types in the wider language regions." [163] (grifos e colchetes em português meus)

Esse trecho revela com agudeza as questões de que estivemos tratando. A despeito da insistência de Carnap no adjetivo "sintático", tudo se passa, aqui, no mais claro plano semântico. O quantificador universal precisa ser *interpretado* da maneira correta, para abranger realmente *todas* as propriedades; e essa maneira correta de interpretar o quantificador exprime-se em termos de valorações. De fato, Carnap fala que a definição de analiticidade para a linguagem II_1 deve ser feita com base em valorações (regras VR1a e VR1c), e que isso significa falar em "todas as propriedades *sintáticas* de expressões acentuadas". Esse ponto acentua bem a confusão que tentamos mostrar: Tais valorações são construídas, realmente, a partir de expressões sintáticas (as expressões acentuadas do conjunto A), e por isso Carnap acredita que sejam sintáticas, embora desempenhem papel claramente semântico. A peça semântica que realmente interessa, mais ainda, pode ser identificada na interpretação extensional para a teoria dos conjuntos: A formulação de Carnap, ao falar em "*todas* as propriedades sintáticas de expressões acentuadas", refere-se (como ele deixa claro aqui e alhures) a *todos* os conjuntos extensionalmente possíveis de

[163] *SLL*, págs. 113 e 114.

expressões acentuadas (ou seja, a *todos* os subconjuntos extensionalmente possíveis do conjunto A).

Na passagem, vemos também a rejeição explícita da metafísica (de um "absolutismo platônico de idéias") claramente associada à percepção errada que Carnap desenvolve em relação a seu próprio método. Carnap deseja excluir qualquer noção absoluta de seu sistema, e isso significa, para ele, a exclusão de qualquer noção que ultrapasse o plano sintático (como plano interno às linguagens), informado pelo Princípio de Tolerância, que ele confunde com o plano formal.

Finalmente, podemos observar que Carnap consegue solucionar o problema que propõe – e que diz respeito ao fato de que a noção de analiticidade de uma linguagem deve estar formulada em outra linguagem – apenas por meio da suposição, certamente extra-sintática, de que toda uma seqüência infinita de linguagens precisa ter sua definição de analiticidade baseada no mesmo esquema de valorações. Em outras palavras: a definição de analiticidade para a linguagem II_1 irá funcionar da maneira desejada desde que esteja formulada em uma outra linguagem, II_2, cujo conceito de analiticidade também seja definido com base em valorações; a linguagem no qual essa definição de analiticidade para II_2 deve ser formulada, por sua vez, deve ser uma outra linguagem II_3 cujo conceito de analiticidade também se baseie em valorações; e assim por diante. Em relação a essa seqüência infinita, não está claro se Carnap tinha uma noção exata acerca da linguagem *formal* a ser utilizada para formular a definição de analiticidade para a própria linguagem II, com seus infinitos tipos lógicos. O ponto importante, porém, é que essa seqüêjcia só pode ser compreendida (e só consegue realizar o que dela se espera) em termos do uso comum que faz do sistema de valorações desenvolvido por Carnap, ou seja, em termos do uso comum de um modelo dentro do qual interpretar as diferentes proposições, para avaliar seu *status* (sua "verdade", como Carnap chega a falar em certos momentos da passagem). O uso de um modelo, portanto (que é o que Carnap obtém por meio da construção do sistema de valorações), e não o uso de nenhuma sintaxe, é a única coisa que funciona como âncora e termo final nesse processo de recorrência *ad infinitum*. E isso Carnap percebeu apenas confusamente.

V

Podemos resumir da seguinte maneira, portanto, os resultados obtidos em nosso trabalho:

1) Carnap defende em *SLL* uma visão sintática da lógica e da filosofia, informada pelo Princípio de Tolerância. Essa visão notabiliza-se por ser formal e antimetafísica. Isso significa que as discussões lógicas e filosóficas devem ser formuladas em relação a sistemas simbólicos devidamente formalizados, ou seja, cálculos lingüísticos que possam ser operados segundo regras bem estabelecidas. Essas linguagens devem ser usadas na descrição científica do mundo, e somente nesse propósito encontram sua medida. Não há que se falar em uma linguagem privilegiada, capaz de refletir a *esfera do ser*, ou a maneira como as coisas *realmente são*.

2) Carnap estende o método sintático de construção de linguagens formais para abranger regras indefinidas (infinitas) de transformação. Esse movimento é motivado, em sua obra, pela compreensão da limitação que os resultados de Gödel haviam imposto aos sistemas tradicionais de lógica simbólica, baseados em regras definidas (finitas) de transformação. O sistema de Carnap, nesse sentido, pode ser descrito como uma tentativa de resgatar o projeto logicista da situação desconfortável em que havia sido colocado pelo teorema de incompletude de Gödel.

3) O resgate proposto por Carnap, no entanto, ocorre segundo um plano diferente do plano original dos logicistas. Não apenas as técnicas usadas são diferentes. Para Carnap, a própria idéia de fundamentação lógica da matemática (e do conhecimento em geral) assume novo contorno. Em *SLL*, não há que se falar propriamente em fundamentação lógica, na medida em que, pelo Princípio de Tolerância, não há uma *lógica* (uma única estrutura dedutiva a que se deva chamar, *corretamente*, de lógica). Para Carnap, trata-se da tarefa – menos espetacular e ao mesmo tempo mais útil – de encontrar sistemas formais capazes de captar as "proposições aritméticas" (no sentido do teorema de Gödel) entre seus teoremas, ou seja, entre suas sentenças analíticas. Por esse caminho, Carnap pretendia mostrar que o método formal, que ele via como método sintático, pode ser suficientemente flexível para lidar com as situações relevantes da ciência (posição que pareceria totalmente implausível

caso ele não fosse capaz de lidar nem ao menos com uma teoria tão básica como a aritmética, tal como normalmente compreendida).

4) Carnap oscila em relação à maneira de compreender o que sejam essas “proposições aritméticas”. Do ponto de vista sintático, elas deveriam ser vistas apenas como um conjunto particular de proposições. Para poder captá-las, no entanto, Carnap precisa desvendar aquilo que está por trás delas, ou o que as distingue como conjunto específico de proposições. Ele se vê, portanto, diante do problema de compreender em que sentido as proposições aritméticas que aparecem no teorema de Gödel são proposições “verdadeiras”.

5) Tanto por meio do conceito de analiticidade para a linguagem I, como do conceito de analiticidade da linguagem II, Carnap encontra uma resposta para essa questão. Ele consegue oferecer um critério pelo qual identificar as proposições aritméticas no sentido de Gödel, e determiná-las como analíticas. Nos dois casos, o critério oferecido por Carnap, uma vez bem compreendido, revela o exato sentido em que as proposições aritméticas de Gödel são “verdadeiras”: *como sentenças verdadeiras de um modelo da aritmética* (o modelo *standard*). Esse modelo é fornecido pelo conjunto A = {0, 0’, 0’’, ... , 0’’’’…’ , ... } de expressões sintáticas numéricas (ditas expressões acentuadas), presente no aparelhamento simbólico das duas linguagens.

6) No caso da linguagem I, cujos recursos expressivos são bastante limitados, o critério encontrado pode ser traduzido por meio de uma única regra sintática indefinida de transformação. É assim que Carnap o apresenta, e a situação teórica fica devidamente encoberta.

7) No caso da linguagem II, a indagação de Carnap o leva mais longe. A linguagem II possui recursos suficientes para expressar toda a matemática clássica. Carnap então formula o problema, explicitamente, como o problema de encontrar um “critério de validade para a *matemática clássica*”. A linha de solução oferecida por Carnap baseia-se no método das valorações. Esse método é claramente semântico: ele pressupõe a construção de um modelo para os números naturais (avaliações das expressões de nível zero, particularmente das variáveis numéricas), para os números reais (avaliações às expressões de nível um, particularmente das variáveis para predicados numéricos) e, em geral, para

todas as entidades da análise funcional clássica, real e complexa (valorações para expressões lógicas de tipo superior).

8) Carnap, portanto, soluciona – na prática – o problema de encontrar o sentido em que certas proposições matemáticas são "verdadeiras". Ele faz isso para as proposições *aritméticas*, diretamente sugeridas pelo teorema de Gödel, e também para as proposições da *matemática clássica*. As primeiras são verdadeiras para um modelo *standard* dos números naturais, fornecido pelo conjunto A (cujos elementos são assumidos como valorações na definição de analiticidade da linguagem II); as segundas são verdadeiras para um modelo da matemática clássica, construído a partir do conjunto A com auxílio de uma interpretação extensional para a teoria de conjuntos.

9) Carnap não compreendeu a exata natureza do procedimento que havia construído. A definição de analiticidade para a linguagem I é fornecida diretamente por meio de uma regra indefinida de transformação (essa regra *traduz* adequadamente, no plano sintático, a *interpretação* do conjunto A como um modelo para os números naturais). Ele acreditou que sua definição de analiticidade para a linguagem II também pudesse ser reduzida a regras sintáticas adequadas; mais do que isso, acreditou que sua construção fosse apenas uma maneira indireta – "por razões técnicas" – de indicar essas regras. Conforme mostramos, no entanto, é absolutamente improvável que um tal conjunto de regras sintáticas possa ser formulado; e mesmo que pudesse, isso pouco alteraria o significado do critério de validade encontrado por Carnap como um critério semântico, que capta a verdade de certas proposições em relação a um modelo.

10) Assim como, do ponto de vista de uma teoria matemática de modelos, os modelos dentro dos quais Carnap interpretou a aritmética e a matemática não são únicos, Carnap não oferece suas definições de analiticidade como únicas. De acordo com o Princípio de Tolerância, as linguagens I e II, com suas respectivas definições de analiticidade, representam apenas duas linguagens possíveis, entre outras tantas linguagens possíveis; as estruturas numéricas que nelas aparecem, e a maneira como são tratadas, constituem apenas duas possíveis estruturas auxiliares na descrição científica do mundo, entre outras tantas estruturas lingüísticas possíveis. (Carnap, contudo, reconhece a particular importância de captar, por meio de estruturas lingüísticas, a teoria da aritmética e

da matemática clássica, e o conjunto de proposições que nessas duas teorias aparecem como válidas. O papel dessas teorias para a ciência parecia bem estabelecido, e sua utilidade indiscutível. É por isso que coloca tanta ênfase na definição de analiticidade para a linguagem II.)

11) Carnap, além do mais, percebe que essas duas teorias – a aritmética e a matemática clássica – possuem uma estrutura própria, isto é, possuem um critério próprio que determina a "validade" de suas proposições. O critério procurado, porém, só poderia ser dado para além da sintaxe. Esse fato decorre da maneira mesmo como Carnap coloca o problema, como o problema de encontrar um equivalente sintático do "critério de validade da matenática clássica". Carnap efetivamente encontra esse critério, sob a forma de um modelo matemático. Infelizmente, tendo encontrado esse modelo, em relação ao qual às sentenças da matemática clássica são avaliadas, ele recai na visão sintática, e não percebe o alcance daquilo que havia estabelecido.

12) Esse trânsito constante de Carnap, entre uma visão puramente sintática da lógica, informada pelo Princípio de Tolerância, e a intenção de captar (por mecanismos que ele acredita sintáticos) certa estrutura teórica – a matemática clássica – exterior à linguagem, confunde muitas vezes o exame crítico da obra. O leitor é levado a crer, em certos momentos, que Carnap busca uma definição absoluta de verdade matemática. A maneira como ele obtém a definição de analiticidade *para a linguagem II*, por meio da construção de um modelo da matemática, aliada à consideração do próprio Princípio de Tolerância, que não é abandonado em nenhum momento de *SLL*, mostram porém que essa não é a sua intenção.

13) É nesse sentido que, seguindo uma linha bastante clara de evolução, Carnap não demorará a abraçar, após a publicação de *SLL*, o ponto de vista explicitamente semântico elaborado por Tarski. Esse ponto de vista satisfaz a alguns dos anseios filosóficos básicos de *SLL*, ao mesmo tempo em que revela a verdadeira natureza do tratamento já oferecido, naquela obra, para a aritmética e matemática. Há, certamente, um preço a pagar: os conceitos semânticos carecem do mesmo grau de formalização que Carnap julgava ter alcançado por meio das estruturas simbólicas sintáticas. Esse preço, no entanto, já havia

sido pago em *SLL*, como única maneira encontrada por Carnap para tratar a matemática clássica. É isso o que Carnap finalmente compreende.

Bibliografia

AYER, A. J. (ed.). **Logical Positivism**, Nova Iorque: Free Press, 1959.

CARNAP, R. **Logische Syntax der Sprache**, 1934, traduzido como **The Logical Syntax of Language**, London: Routledge & Kegan Paul Ltd., 1937.

COFFA, A. **The Semantic Tradition from Kant to Carnap: to the Vienna Station**, Cambridge: Cambridge University Press, 1991.

"*Carnap, Tarski and the Search for Truth*", em *Noûs 21 (4)*, 1987.

FRIEDMAN, M. **Reconsidering Logical Positivism**, Cambridge: Cambridge University Press, 1999.

A Parting of the Ways, Chicago: Open Court, 2000.

"*Carnap and the Evolution of the A Priori*", editado por Awodey, S. e Klein, C., **Carnap Brought Home: the View from Jena**, Chicago: Open Court, 2004.

GÖDEL, K. **Collected Works**, Volume III, Unpublished essays and lectures, editado por S. Feferman, Nova Iorque e Oxford: Oxford University Press, 1995.

HERBRAND, J. **Recherches sur la Théorie de la Démonstration**, 1930, republicado em [Herbrand, 1968].

Écrits Logiques, Paris: Presses Universitaires de France, 1968.

HINTIKKA, J. **On Gödel**, Belmont: Wadsworth, 2000.

The Principles of Mathematics Revisited, Cambridge: Cambridge University Press, 1996.

KIRKHAM, R. **Theories of Truth: a critical introduction**, Cambridge, Mass.: MIT Press, 1992.

LAKATOS, I. **Mathematics, Science and Epistemology, philosophical papers, vol. 1** – Cambridge: Cambridge University Press, 1978.

Mathematics, Science and Epistemology, philosophical papers, vol. 2 – Cambridge: Cambridge University Press, 1978.

PROUST, J. "*Fomal Logic as Transcendental in Wittgenstein and Carnap*", em **Noûs** 21(4)*,* 1987.

RICKETTS, T. "*Carnap's Principle of Tolerance, Empiricism, and Conventionalism*", 1994, em P. Clark e B. Hale (eds.), **Reading Putnam**, Oxford: Blackwell, p. 176-200.

RAMSEY, F. "*The Foundations of Mathematics and Other Logical Essays*", 1925, editado por R. B. Braithwaite, London: Kegan Paul, 1931.

RUSSELL, B. **Logic and Knowledge**, Trowbridge: Routledge, 1956.

My Philosophical Development, Trowbridge: Routledge, 1959.

TARSKI, A. "*Der Wahrheitsbegriff in den formalizierten Sprachen*", Studia Philosophica 1, 1936.

"*The Semantic Conception of Truth*", em **Philosophy and Phenomenological Research**, IV, 1944.

VAN HEIJENOORT, J. "*Logic as Calculus and Logic as Language*", 1967, em **Synthese** 17.