

Gramática Categorical Independente

Luiz Arthur Pagani (DELIN/UFPR)
<http://people.ufpr.br/~arthur>
arthur@ufpr.br

Fevereiro de 2011

Sumário

1	Introdução	3
1.1	Fundamentos	3
2	Regra de Inserção Lexical (Lx)	5
2.1	Exemplos do uso de Lx	6
2.2	Exercício	7
3	Regra de Aplicação Funcional ($R1$)	8
3.1	Aplicação funcional	9
3.2	Exemplo do uso de $R1$	12
3.3	Exercícios	13
4	Regra de Associatividade ($R3$)	14
4.1	Redução- β	15
4.2	Exemplo do uso de $R3$ e da redução- β	16
4.3	Exercícios	20
5	Regra de Composição ($R2$)	21
5.1	Exemplo de uso de $R2$	22
6	Regra de promoção ($R4$)	26
6.1	Exemplos de uso de $R4$	28

6.1.1	<i>Pedro corre</i>	28
6.1.2	<i>Pedro ama Maria</i>	30
7	Dedução de informações lexicais	34
7.1	Exercícios	40
8	Conclusão	41
A	Variantes notacionais	43
A.1	Ajdukiewicz [1]	43
A.2	Bar-Hillel [2]	43
A.3	Lambek [11]	43
A.4	Geach [8]	44
A.5	Steedman [23]	44
A.6	Jacobson [10]	44
A.7	Estrutura de Características [19]	44

Capítulo 1

Introdução

O objetivo deste texto é o de apresentar um dos modelos mais simples de Gramática Categorial (GC), de forma que o leitor possa se familiarizar com o tratamento que uma GC pode oferecer para explicar fenômenos relacionados às línguas naturais (no nosso caso, notadamente, o português brasileiro). Se, por um lado, o texto não pressupõe nenhum conhecimento de GC, nem de suas ferramentas auxiliares (o cálculo de predicados de primeira ordem [17] e o cálculo- λ [15, p. 50–51], que são empregados na GC, também serão apresentadas aqui; caso seja necessário), por outro, não faremos nenhuma apresentação de questões sintáticas, relacionadas à estrutura de constituintes, ou semânticas, relacionadas à forma lógica; espera-se que o leitor já tenha algum conhecimento inicial sobre estas questões, ou que possa recorrer a um manual de introdução a estes assuntos (como o de Lyons [12, ps. 219–238] ou o de Miotto, Silva e Lopes [13], sobre a estrutura de constituintes, e o de Chierchia [6], sobre a forma lógica). O que se pretende é que o leitor, ao final do presente texto, esteja mais apto a enfrentar a leitura de uma introdução mais avançada à GC, como a de Borges [18], a de Wood [25], a de Moortgat [14], as de Morrill [15, 16] ou a de Carpenter [5].

1.1 Fundamentos

As GCs são, basicamente, ferramentas que permitem reconstruir composicionalmente a história derivacional de uma expressão lingüística. Por “composicionalmente”, entende-se que a derivação da expressão irá representar não só a combinatória sintática, mas apresentará também, para cada unidade sintática, sua respectiva interpretação semântica. Nesse sentido, as GCs constituem uma opção metodológica para executar o Princípio da Composicionalidade (ou Princípio de Frege), segundo o qual o significado de uma expressão complexa é

composto a partir dos significados das expressões mais simples que a constituem e da maneira como essas expressões são combinadas.¹

Apesar de supor algum conhecimento sobre a estrutura de constituintes das expressões lingüísticas, as GCs recorrem a critérios para identificar a coesão das combinações entre as expressões que vão além dos testes clássicos de constituição (ver, por exemplo, [12, ps. 219–226]. Assim, sem abandonar a noção de constituinte imediato, um GC recorre não apenas a testes sintáticos (como a relação entre a ativa e a passiva, a operação de clivagem ou a ambigüidade estrutural das expressões), mas também a critérios da combinatória semântica; e como a ontologia de muitas teorias semânticas é habitada por *entidades funcionais*, a relação entre funções e argumentos têm um papel de destaque na determinação da coesão das expressões lingüísticas. Nas palavras de Wood [25, p. 3], para uma GC,

a língua é concebida em termos de função e argumento, e não através de uma estrutura de constituintes. É, portanto, uma gramática de valência ou de dependência, e não uma gramática de estrutura sintagmática (GES); na literatura, GESs e GCs são freqüentemente contrapostas, respectivamente, como gramáticas posicionais (ou configuracionais) e funcionais

Do trecho acima, não é preciso concluir que esta é uma questão de exclusão (segundo a qual a relação função-argumento excluiria a estrutura de constituintes); senão a autora estaria sendo incoerente com a relação entre as GESs e as GCs que ela mesma postula mais adiante [25, p. 6]: “a teoria da GC pode ser concebida como uma metateoria lógica para a teoria da GES.”

Merece também explicação o termo **categoria**, que dá nome às GCs. Como veremos, as estruturas das expressões lingüísticas serão construídas através de operações que afetam tanto sua classe sintática quanto sua interpretação semântica. Dessa maneira, daremos o nome de *categoria* a esse par de classe sintática e interpretação semântica. A relação entre os elementos do par, por sua vez, será representada por ‘:’; assim, uma categoria será escrita como <classe sintática>:<interpretação semântica>.

A versão da GC que será apresentada aqui é a que Wood [25, p. 35] chama de *Gramática Categorial Independente* (GCI, *Free Categorical Grammar*). Ela é composta por uma regra de inserção lexical² e mais quatro pares de regras combinatórias (aplicação, associatividade, composição e promoção), que passaremos a ver a seguir. Como já disse, incluirei também algumas explicações, ainda que muito sucintas, sobre duas ferramentas auxiliares para a representação semântica: a aplicação funcional e a redução- β , à medida que forem necessárias.

(Ao longo da sua pequena história, autores diferentes empregaram diversas variantes notacionais para representar a dependência entre as categorias. Darei preferência aqui à notação de Lambek. Para ajudar a localizar e a relacionar as diversas variantes, o leitor encontra no Apêndice A (p. 43) um resumo das variantes que consegui localizar.)

¹Há uma controvérsia sobre Frege ter afirmado explicitamente ou não o princípio da composicionalidade, mas essa questão é mais importante para os filósofos e os historiadores da ciência do que para os lingüistas. Para constatar a importância da composicionalidade para as teorias semânticas, consultar, por exemplo, [4, p. 4], [24, p. 39] ou [6, p. 39].

²É pouco comum incluir menção explícita à regra de inserção lexical, mas aqui preferi seguir o exemplo de Carpenter [5, p. 155].

Capítulo 2

Regra de Inserção Lexical (Lx)

Apesar de não ser muito comum na maioria das apresentações das GCs (porque normalmente isto é subentendido), a regra mais importante para começar a construção de uma análise gramatical através de uma GC é a regra de inserção lexical,¹ pois é ela que interliga o conjunto de itens lexicais (Léxico) e as regras da GC.

Podemos representar a regra de inserção lexical da seguinte maneira:

$$\bullet e \Longrightarrow X : \tau \qquad \qquad \qquad [(e, X, \tau) \in \text{Léxico}]$$

(Onde e é uma expressão lexical, X sua classe sintática e τ sua interpretação semântica.)

A partir do uso de Lx , podemos desenhar diagramas como:

$$\boxed{\frac{e}{X : \tau} \quad Lx}$$

que podem ser empregados como subdiagramas para deduzir² as estruturas gramaticais das expressões lingüísticas.³

¹Estamos assumindo explicitamente que a construção da análise gramatical é ascendente (*bottom-up*, começando da expressão). Como as GCs dispõem do conceito *totalmente conexo* (uma expressão é totalmente conexa se, ao final da derivação, todas as suas partes estiverem interligadas; ou seja, se a derivação resultar numa única árvore — em termos práticos, isso acontece quando a derivação termina numa única categoria), não é preciso, como nos analisadores descendentes (*top-down*), estabelecermos de antemão uma categoria inicial.

²O termo *deduzir* aqui remete diretamente à chamada “análise gramatical como dedução” (*parsing as deduction*), muito empregada na Lingüística Computacional derivada da Programação em Lógica.

³Estes diagramas são chamados de “diagramas ao estilo de Prawitz”, por terem sido usados pelo lógico polonês Dag Prawitz, no seu sistema de Dedução

2.1 Exemplos do uso de Lx

Nos nossos exemplos, vamos supor um subconjunto bastante restrito do conjunto de itens lexicais (o Léxico). Precisaremos minimamente que:

$$\{\langle \text{Pedro}, N, p \rangle, \langle \text{Maria}, N, m \rangle, \langle \text{corre}, N \setminus S, C \rangle, \langle \text{ama}, (N \setminus S)/N, A \rangle\} \subseteq \text{Léxico}.$$

Assim, já que $\langle \text{Pedro}, N, p \rangle \in \text{Léxico}$, podemos incluir nas derivações uma subderivação como:

$$\boxed{\frac{\text{Pedro}}{N : p} \quad Lx}$$

Como também $\langle \text{Maria}, N, m \rangle \in \text{Léxico}$, outra subderivação possível é:

$$\boxed{\frac{\text{Maria}}{N : m} \quad Lx}$$

E ainda, como $\langle \text{corre}, N \setminus S, C \rangle \in \text{Léxico}$, também dispomos da subderivação abaixo:

$$\boxed{\frac{\text{corre}}{N \setminus S : C} \quad Lx}$$

Finalmente, como $\langle \text{ama}, (N \setminus S)/N, A \rangle \in \text{Léxico}$, a última subderivação lexical que empregaremos em nossos exemplos é:

$$\boxed{\frac{\text{ama}}{(N \setminus S)/N : A} \quad Lx}$$

Natural [20].

2.2 Exercício

1. Suponha que o verbo intransitivo *fumar* denote a função F , represente a informação lexical associada a ele e desenhe o diagrama correspondente.
2. Suponha que o verbo transitivo direto *odiar* denote a função O , represente a informação lexical associada a ele e desenhe o diagrama correspondente.
3. Admitindo que uma descrição definida como *o rei da França* denota um único indivíduo, como os nomes próprios, e pode ser representada pela fórmula $\iota x(R x)$, represente-a como se fosse uma expressão lexical.

Capítulo 3

Regra de Aplicação Funcional ($R1$)

A regra de aplicação funcional recebe este nome devido à maneira como as interpretações das expressões envolvidas são combinadas: uma das denotações é funcional e a outra seu argumento, e a combinação de ambas resulta na aplicação da função denotada pela primeira ao argumento denotado pela segunda.¹

A regra de aplicação funcional algumas vezes também é chamada de regra de cancelamento, porque ela se assemelha, em sua parte sintática, com a operação de cancelamento de frações (que permite simplificar uma fração quando ela é multiplicada a um número igual ao seu denominador: $\frac{2}{3} \times \frac{3}{2} = 2$).

Assim, aplicando inversamente esta metáfora do cancelamento de frações, a partir da suposição de que *Pedro corre* é uma sentença, cuja denotação é $(C\ p)$, e de que *Pedro* é um nome, que denota o indivíduo p , podemos deduzir que *corre* é uma expressão que se combina com um nome antes dela para resultar numa sentença e que denota a função C (que mapeia os indivíduos que estão correndo para o valor verdadeiro, e os que não correm para o falso); vamos escrever isto da seguinte forma: $N \backslash S : C$.

Como as línguas naturais geralmente são sensíveis à ordem de combinação dos constituintes, as regras das GCs são apresentadas aos pares, indicando o lado no qual a categoria funcional deve encontrar o seu argumento. Assim, a regra de aplicação é representada pelo seguinte par:

- $X/Y : \phi \quad \oplus \quad Y : \alpha \quad \Longleftrightarrow \quad X : (\phi\ \alpha)$
- $Y : \alpha \quad \oplus \quad Y \backslash X : \phi \quad \Longleftrightarrow \quad X : (\phi\ \alpha)$

¹Por motivos práticos, vamos nos restringir a uma semântica extensional, mas isso não significa que não poderíamos usar uma semântica intensional.

A primeira fórmula deve ser entendida como ‘uma expressão pertencente à categoria (funcional) X/Y , que denota a função ϕ , quando antecede outra expressão da categoria Y , que denota α , resulta numa expressão da categoria X e denota $(\phi \alpha)$ — a aplicação da função ϕ ao argumento α ; ou, mais simplesmente, ϕ de α ’.

A segunda diz que ‘qualquer expressão da categoria (funcional) $Y \setminus X$, cuja denotação é a função ϕ , quando anteceda por outra expressão da categoria Y , denotando α , resulta numa expressão da mesma categoria X e que também denota $(\phi \alpha)$ ’.

A partir de $R1$ também podemos contruir diagramas como os seguintes:

$$\boxed{\frac{X/Y : \phi \quad Y : \alpha}{X : (\phi \alpha)} \quad R1} \qquad \boxed{\frac{Y : \alpha \quad Y \setminus X : \phi}{X : (\phi \alpha)} \quad R1}$$

que também podem ser empregados como subdiagramas para deduzir a estrutura gramatical das expressões lingüísticas.

3.1 Aplicação funcional

Antes de apresentarmos o exemplo do uso de $R1$, vamos falar um pouco sobre a aplicação funcional. Como o termo já indica, a aplicação funcional nada mais é do que a aplicação de uma função a um argumento.

Esse tipo de conceito tem sido empregado na semântica moderna para explicar uma parte do funcionamento do significado das expressões lingüísticas. Para falar do significado de uma sentença como *Pedro corre*, por exemplo, neste tipo de semântica, costuma-se reconhecer que uma sentença como esta serve para exprimir algo que, se corresponder aos fatos, é verdadeira; e é falsa, caso não corresponda à realidade. Se considerarmos ainda que um nome próprio como *Pedro* denota um indivíduo (o indivíduo que se chama *Pedro*), não é difícil responder em que situações a sentença *Pedro corre* é verdadeira: ela sempre será verdadeira quando o indivíduo cujo nome é *Pedro* pertencer ao conjunto dos indivíduos que estiverem correndo; usando a notação da teoria de conjunto, se representarmos *Pedro* por p (os indivíduos são sempre representados por letras minúsculas) e *corre* por C (os conjuntos são simbolizados por letras maiúsculas), a sentença será verdadeira quando $p \in C$.

Uma maneira alternativa de se dizer a mesma coisa é através do conceito de função. Ao invés de dizer que *Pedro corre* é verdadeira quando $p \in C$, podemos dizer que *corre* denota uma função que, quando toma a denotação de *Pedro* como argumento, resulta no valor verdadeiro se Pedro efetivamente estiver correndo, e no valor falso se ele não estiver correndo.

Vamos supor, a título de ilustração, um mundo no qual existam apenas o Pedro (p), a Maria (m) e o João (j). Imaginemos agora uma situação na qual apenas o João esteja correndo. Isso pode ser representado como função em forma de uma matriz que diz, para cada indivíduo do mundo, se ele está correndo (V) ou não (F), como abaixo:

$$C_1 = \begin{bmatrix} p & \rightarrow & F \\ m & \rightarrow & F \\ j & \rightarrow & V \end{bmatrix}$$

Vamos imaginar agora uma segunda situação na qual tanto o Pedro quanto a Maria estejam correndo, isso pode ser representado pela função C_2 , a seguir:

$$C_2 = \begin{bmatrix} p & \rightarrow & V \\ m & \rightarrow & V \\ j & \rightarrow & F \end{bmatrix}$$

Imaginemos finalmente uma terceira situação, na qual todos estejam correndo:

$$C_3 = \begin{bmatrix} p & \rightarrow & V \\ m & \rightarrow & V \\ j & \rightarrow & V \end{bmatrix}$$

Intuitivamente, sabemos que Pedro só corre na segunda e na terceira situação (só nestas é que *Pedro corre* é verdadeira, na primeira situação ela é falsa). Vejamos o que nos diz a nossa abordagem funcional. Se a sentença *Pedro corre* é o resultado da aplicação da função denotada por *corre* ao argumento denotado por *Pedro* — que representaremos como $(C_n p)^2$ (onde n representa a situação a ser considerada) — vejamos o que acontece em cada uma das três situações. Na primeira, $(C_1 p)$, é preciso olhar na função C_1 qual o valor correspondente ao argumento p ; vemos que é F , o que corresponde ao falso, como em nossa intuição. Na segunda situação, para descobrirmos o valor de $(C_2 p)$, precisamos observar em C_2 o valor correspondente ao argumento p , que é V , ainda de acordo com nossa intuição. Na última situação, finalmente, o valor de $(C_3 p)$ também é V (verdadeiro, novamente como em nossa intuição). Assim, constatamos que a representação de *corre* como uma função e da sentença *Pedro corre* como a aplicação dessa função a um argumento é uma teoria fiel a essa nossa pequena inquirição.

²Também no caso da aplicação funcional há variação notacional. Na notação mais clássica do cálculo de predicados de primeira ordem, como no livro do Mortari [17] por exemplo, a fórmula para um predicado de um argumento (uma propriedade) seria $C_n(p)$; para um predicado de dois argumentos (uma relação binária), teríamos $A(p, m)$ (Pedro ama Maria); generalizando, uma construção com predicado enário (uma relação entre uma quantidade arbitrária n de indivíduos) corresponderia a $R(a_1, \dots, a_n)$. A notação usada no texto é mais comum nas GCs e aparece também em algumas implementações de analisadores gramaticais (*parsers* em Prolog, onde a aplicação funcional é representada por ‘@’ (apesar da notação do próprio Prolog ser do tipo $C(p)$), e $(C p)$ corresponderia a ‘C@p’.

Essa mesma perspectiva pode ser facilmente estendida aos verbos transitivos diretos. Para uma sentença como *Pedro ama Maria*, por exemplo, podemos dizer que *ama* é uma função que toma como argumento a denotação do objeto direto e resulta numa outra função que, por sua vez, toma como argumento a denotação do sujeito e resulta num valor de verdade (V caso o indivíduo correspondente ao sujeito ame o indivíduo correspondente ao objeto; F caso não). Supondo novamente os mesmos três indivíduos, podemos imaginar uma certa situação na qual a função A , correspondente ao verbo *ama*, é representada pela seguinte matriz:

$$A = \left[\begin{array}{c} p \rightarrow \left[\begin{array}{c} p \rightarrow V \\ m \rightarrow V \\ j \rightarrow F \end{array} \right] \\ m \rightarrow \left[\begin{array}{c} p \rightarrow V \\ m \rightarrow F \\ j \rightarrow V \end{array} \right] \\ j \rightarrow \left[\begin{array}{c} p \rightarrow F \\ m \rightarrow F \\ j \rightarrow F \end{array} \right] \end{array} \right]$$

Nesta situação, a sentença *Pedro ama Maria* corresponderia a $((A\ m)\ p)$. Vamos resolver primeiro $(A\ m)$: para esta função, obtemos uma outra função — vamos chamá-la provisoriamente de F — como abaixo:

$$F = \left[\begin{array}{c} p \rightarrow V \\ m \rightarrow F \\ j \rightarrow V \end{array} \right]$$

Como $((A\ m)\ p)$ é equivalente a $(F\ p)$, e esta última tem como resultado V , concluímos que a sentença *Pedro ama Maria* é verdadeira nesta situação.

Nos exemplos que apresentaremos, como o interesse não é a significação lexical (que exigiria por si só uma profunda pesquisa empírica), vamos nos limitar a falar da aplicação das funções aos argumentos, sem especificarmos exatamente como são estas funções.

3.2 Exemplo do uso de $R1$

Para deduzir a estrutura da sentença *Pedro ama Maria*, por exemplo, antes de começarmos a empregar as regras categoriais, precisamos aplicar a regra de inserção lexical para identificar os itens lexicais que aparecem na sentença. Usando os exemplos apresentandos na seção 2.1, podemos construir o seguinte diagrama:

$$\boxed{\begin{array}{ccc} \frac{\text{Pedro}}{N : p} & Lx & \frac{\text{ama}}{(N \backslash S)/N : A} & Lx & \frac{\text{Maria}}{N : m} & Lx \end{array}}$$

Nesta condição, podemos empregar $R1$ para juntar *ama* com *Maria*, já que suas respectivas categorias se encaixam com a primeira versão de $R1$, da seguinte maneira:³

$$\begin{array}{ccc} \underbrace{(N \backslash S)} & / & N : A \\ \downarrow & & \downarrow \\ X & / & Y : \phi \end{array} \oplus \begin{array}{ccc} N : m \\ \downarrow \\ Y : \alpha \end{array} \iff \begin{array}{ccc} X & : & (\phi \quad \alpha) \\ \downarrow & & \downarrow \downarrow \\ \underbrace{N \backslash S} & : & (A \quad m) \end{array}$$

Assim, podemos acrescentar mais uma parte ao diagrama acima, de forma que obtemos o seguinte diagrama:

$$\boxed{\begin{array}{ccc} \frac{\text{Pedro}}{N : p} & Lx & \frac{\text{ama}}{(N \backslash S)/N : A} & Lx & \frac{\text{Maria}}{N : m} & Lx \\ & & & & & R1 \\ & & & & N \backslash S : (A \ m) \end{array}}$$

O que este diagrama nos diz é que temos a expressão *Pedro*, da categoria N , denotando p , seguida da expressão *ama Maria*, que é da categoria $N \backslash S$ e denota $(A \ m)$. Dessa maneira, podemos empregar agora a segunda versão de $R1$, da seguinte forma:

³Observe que, quando os parêntesis aparecem nas margens de uma categoria sintática, eles serão sistematicamente apagados, porque aí eles só servem para indicar a estrutura de combinação com os conectivos. Na aplicação funcional, porém, como eles a indicam, vamos preferir mantê-los; mas, quando já estivermos dominando melhor essa notação, eles também podem ser omitidos quando aparecem nas margens de um termo usado para representar a interpretação semântica.

$$\begin{array}{ccc}
N : p & N \setminus S : \underbrace{(A \ m)} & \\
\downarrow & \downarrow & \downarrow \\
Y : \alpha & \oplus & Y \setminus X : \phi
\end{array}
\iff
\begin{array}{ccc}
X : (& \phi & \alpha) \\
\downarrow & \downarrow & \downarrow \\
S : (& \underbrace{(A \ m)} & p)
\end{array}$$

Com isto, podemos completar o diagrama com a derivação da estrutura de *Pedro ama Maria*, que, como vemos no diagrama abaixo, é uma expressão totalmente conexa — mais especificamente, é uma sentença (S) que denota $((A \ m) \ p)$.

Pedro		ama		Maria
$\frac{}{N : p}$	Lx	$\frac{}{(N \setminus S)/N : A}$	Lx	$\frac{}{N : m}$
				$R1$
		$N \setminus S : (A \ m)$		
				$R1$
		$S : ((A \ m) \ p)$		

3.3 Exercícios

1. Faça a derivação da sentença *João corre*.
2. Suponha que apenas João fume; Faça a matriz que representa o predicado de um lugar F .
3. Suponha que Pedro odeie João e Maria, que Maria odeie Pedro e se odeie, que João não odeie ninguém. Faça a matriz que representa o predicado de dois lugares O .
4. Faça a derivação de *Maria odeia João*.

Capítulo 4

Regra de Associatividade ($R3$)

O que a regra de associatividade faz é mudar a ordem na qual uma expressão de categoria funcional biargumental se combina com seus dois argumentos, sem afetar a interpretação semântica final.¹ A associatividade é definida pelo seguinte par de fórmulas:

- $(X \setminus Y)/Z : \phi \iff X \setminus (Y/Z) : \lambda x_n. \lambda x_m. ((\phi \ x_m) \ x_n)$
- $X \setminus (Y/Z) : \phi \iff (X \setminus Y)/Z : \lambda x_n. \lambda x_m. ((\phi \ x_m) \ x_n)$

[para variáveis x_n e x_m novas na derivação]

A primeira diz que ‘uma expressão que pertence à categoria (funcional) $(X \setminus Y)/Z$ e que denota uma função ϕ também pertence à categoria $X \setminus (Y/Z)$, denotando a função $\lambda x_n. \lambda x_m. ((\phi \ x_m) \ x_n)$ ’ (como veremos na próxima seção, este uso do operador- λ garante que a função continua recebendo seus argumentos no lugar certo, apesar de sua ordem de combinação ter sido invertida). A leitura da segunda fórmula é uma imagem invertida da primeira: ‘qualquer expressão da categoria $X \setminus (Y/Z)$, que denota ϕ , também é da categoria $(X \setminus Y)/Z$ e denota $\lambda x_n. \lambda x_m. ((\phi \ x_m) \ x_n)$ ’. Para ambas as fórmulas, contudo, é preciso que as variáveis x_n e x_m ainda não tenham sido usadas em nenhuma parte da derivação feita até o momento em que a regra $R3$ é empregada. (Para garantirmos isto, os índices n e m devem ser

¹Na matemática, o termo “associatividade” geralmente é empregado para designar uma propriedade de um único operador, quando sua ordem de combinação não é relevante; numa soma, por exemplo, que é associativa, $x + (y + z) = (x + y) + z$. A associatividade na nossa GCI designa, na verdade, uma relação entre dois operadores; por isso, alguns autores preferem chamá-la de permutação (*swapping*).

substituídos por números inteiros nas nossas derivações e, para que essa restrição seja obedecida, basta substituírmos n e m por números que ainda não tenham aparecido em nenhum passo anterior da derivação.)²

Através desta regra, podemos desenhar diagramas como os dois abaixo, que também podem ser empregados como subdiagramas nas deduções das estruturas gramaticais de expressões lingüísticas complexas.

$$\boxed{\frac{(X \setminus Y)/Z : \phi}{X \setminus (Y/Z) : \lambda x_n. \lambda x_m. ((\phi \ x_m) \ x_n)} \quad R3} \qquad \boxed{\frac{X \setminus (Y/Z) : \phi}{(X \setminus Y)/Z : \lambda x_n. \lambda x_m. ((\phi \ x_m) \ x_n)} \quad R3}$$

4.1 Redução- β

Também antes de passarmos ao exemplo do uso de $R3$ será preciso apresentar a redução- β . Esta operação desempenha dois papéis nas derivações de uma Gramática Categorial: primeiro, ela simplifica a leitura de um termo que, sem ela, seria bem mais complexo; depois,

²Alternativamente, ao invés de empregar o operador- λ , seria possível postular um predicado $PERM$, que intercambiase as duas posições argumentais de um predicado (ou seja, tal que $(PERM \ \phi) = \lambda x_n. \lambda x_m. ((\phi \ x_m) \ x_n)$), de forma que o par de fórmulas para $R3$ poderia ser definido como:

- $(X \setminus Y)/Z : \phi \quad \Longleftrightarrow \quad X \setminus (Y/Z) : (PERM \ \phi)$
- $X \setminus (Y/Z) : \phi \quad \Longleftrightarrow \quad (X \setminus Y)/Z : (PERM \ \phi)$

Em termos da aplicação funcional, esse predicado $PERM$, quando aplicado à mesma função A (atribuída ao verbo *ama*), na página 11, apresentaria o seguinte resultado:

$$(PERM \ A) = \left[\begin{array}{l} p \rightarrow \left[\begin{array}{ll} p & \rightarrow V \\ m & \rightarrow V \\ j & \rightarrow F \end{array} \right] \\ m \rightarrow \left[\begin{array}{ll} p & \rightarrow V \\ m & \rightarrow F \\ j & \rightarrow F \end{array} \right] \\ j \rightarrow \left[\begin{array}{ll} p & \rightarrow F \\ m & \rightarrow V \\ j & \rightarrow F \end{array} \right] \end{array} \right]$$

Este tipo de solução é empregada na Gramática Categorial Combinatória, de Steedman [23, pp. 30–48], e segue algumas tentativas de eliminação das variáveis nas expressões lógicas, como a da Lógica Combinatória, de Curry e Feys [7], ou a Quine [21].

ela mostra mais claramente que alguns termos podem ser equivalente, apesar de terem formas bem diferentes (se, depois de aplicarmos a redução- β (possivelmente mais de uma vez) a um termo τ_1 , ele tiver a forma τ_2 , isso significa que os dois termos (τ_1 e τ_2) são equivalentes).³

A redução- β é definida pela seguinte fórmula:

$$\bullet (\lambda x_n. \tau \ \alpha) \quad \Longleftrightarrow \quad \tau^{[x_n \mapsto \alpha]}$$

de forma que $\tau^{[x_n \mapsto \alpha]}$ é o resultado de substituir no termo τ todas as ocorrências livres da variável x_n por α . (Numa fórmula como $\lambda x_n. \tau$, todas as ocorrências da variável x_n em τ são ligadas; todas as ocorrências das outras variáveis, se não estiverem ligadas por outro uso do operador- λ , serão livres em τ .)

Como a redução- β só afeta a representação da interpretação semântica (ela não modifica a categoria sintática), e como ela pode ser usada mais de uma vez e mesmo em sub-termos de um termo, o seu uso nos permitirá acrescentar nos diagramas o seguinte desenho:

$$\boxed{\begin{array}{c} \hline X : \quad \quad \quad \vdots \\ \quad \quad \quad \dots (\lambda x_n. \tau \ \alpha) \dots \\ \quad \quad \quad =_{red.\beta} \dots \tau^{[x_n \mapsto \alpha]} \dots \end{array}}$$

4.2 Exemplo do uso de $R3$ e da redução- β

Podemos fazer, para a mesma sentença *Pedro ama Maria*, uma derivação diferente da que vimos na seção 3.2, através agora do uso de $R3$. Começamos, porém, da mesma maneira, através da regra de inserção lexical:

$$\boxed{\begin{array}{ccc} \text{Pedro} & \text{ama} & \text{Maria} \\ \hline \frac{}{N : p} & Lx & \frac{}{(N \setminus S)/N : A} & Lx & \frac{}{N : m} & Lx \end{array}}$$

³A redução- β , na verdade, é uma parte do cálculo- λ ([15, ps. 50–51] ou [5, p. 50]. Este cálculo é definido pelas seguintes três convenções:

Convenção- α : $\lambda x_n. \tau \equiv \lambda x_m. \tau^{[x_n \mapsto x_m]}$

Convenção- β : $(\lambda x_n. \tau \ \alpha) \equiv \tau^{[x_n \mapsto \alpha]}$

Convenção- η : $\lambda x_n. (\phi \ x_n) \equiv \phi$

A redução- β é a segunda convenção usada apenas da esquerda para a direita.

Na seqüência, antes de usarmos *R1*, podemos empregar a primeira fórmula de *R3* no verbo *ama*, da seguinte maneira (observe a substituição de *n* por 0 e de *m* por 1, já que nenhuma variável tinha sido usada antes na derivação):

$$\begin{array}{c}
 (N \setminus S) / N : A \\
 \downarrow \quad \downarrow \quad \downarrow \quad \downarrow \\
 (X \setminus Y) / Z : \phi
 \end{array} \iff \begin{array}{c}
 X \setminus (Y / Z) : \lambda x_n . \lambda x_m . ((\phi x_m) x_n) \\
 \downarrow \quad \downarrow \quad \downarrow \quad \downarrow \quad \downarrow \quad \downarrow \\
 N \setminus (S / N) : \lambda x_0 . \lambda x_1 . ((A x_1) x_0)
 \end{array}$$

Esse emprego de *R3* nos permite aumentar o diagrama da seguinte maneira:

$\frac{\text{Pedro}}{N : p} \quad Lx$	$\frac{\text{ama}}{(N \setminus S) / N : A} \quad Lx$	$\frac{\text{Maria}}{N : m} \quad Lx$
$\frac{N \setminus (S / N) : \lambda x_0 . \lambda x_1 . ((A x_1) x_0)}{R3}$		

Agora, o verbo *ama* busca primeiro um argumento antes, para só então buscar um argumento depois. Com isso, podemos agora usar *R1* para juntar *Pedro* e *ama*:

$$\begin{array}{c}
 N : p \quad N \setminus \underbrace{(S / N)} : \underbrace{\lambda x_0 . \lambda x_1 . ((A x_1) x_0)} \\
 \downarrow \quad \downarrow \quad \downarrow \quad \downarrow \quad \downarrow \\
 Y : \alpha \quad Y \setminus X : \phi
 \end{array} \iff \begin{array}{c}
 X : (\quad \phi \quad \alpha) \\
 \downarrow \quad \downarrow \quad \downarrow \\
 \underbrace{S / N} : (\quad \underbrace{\lambda x_0 . \lambda x_1 . ((A x_1) x_0)} \quad p)
 \end{array}$$

Depois deste uso de *R1* a derivação passa a ser:

$\frac{\text{Pedro}}{N : p} \quad Lx$	$\frac{\text{ama}}{(N \setminus S) / N : A} \quad Lx$	$\frac{\text{Maria}}{N : m} \quad Lx$
$\frac{N \setminus (S / N) : \lambda x_0 . \lambda x_1 . ((A x_1) x_0)}{R3}$		
$\frac{S / N : (\lambda x_0 . \lambda x_1 . ((A x_1) x_0) p)}{R1}$		

Neste ponto, encontramos uma condição que nos permite usar a redução- β .⁴ O termo que representa a interpretação semântica de *Pedro ama* é composto exatamente por um termo iniciado com o operador- λ aplicado a um argumento:

$$\begin{array}{ccc}
 (\lambda \ x_0 \ . \ \underbrace{\lambda x_1.((A \ x_1) \ x_0)} & p &) \\
 \downarrow & \downarrow & \downarrow \\
 (\lambda \ x_n \ . \ \tau & \alpha &) \iff \tau^{[x_n \mapsto \alpha]} \\
 & & \downarrow \\
 & & \underbrace{\lambda x_1.((A \ x_1) \ p)}
 \end{array}$$

Com isto, podemos reformular o diagrama anterior, da seguinte forma:

$\frac{\text{Pedro}}{N : p} \quad Lx$	$\frac{\text{ama}}{(N \setminus S)/N : A} \quad Lx$	$\frac{\text{Maria}}{N : m} \quad Lx$
$\frac{(N \setminus S)/N : A}{N \setminus (S/N) : \lambda x_0. \lambda x_1. ((A \ x_1) \ x_0)} \quad R3$		
$\frac{N \setminus (S/N) : \lambda x_0. \lambda x_1. ((A \ x_1) \ x_0)}{S/N : (\lambda x_0. \lambda x_1. ((A \ x_1) \ x_0) \ p)} \quad R1$		
$\frac{S/N : (\lambda x_0. \lambda x_1. ((A \ x_1) \ x_0) \ p)}{=_{red.\beta} \lambda x_1. ((A \ x_1) \ p)}$		

Agora podemos aplicar novamente *R1*, desta vez juntando *Pedro ama* com *Maria*:

$$\begin{array}{ccc}
 S \ / \ N : \underbrace{\lambda x_1.((A \ x_1) \ p)} & N : m & \\
 \downarrow & \downarrow & \downarrow \\
 X \ / \ Y : \phi & Y : \alpha & \\
 \iff & & \\
 X : (\phi & \alpha &) \\
 \downarrow & \downarrow & \downarrow \\
 S : (\underbrace{\lambda x_1.((A \ x_1) \ p)} & m &)
 \end{array}$$

O diagrama então passa a ser:

⁴Caso tivéssemos preferido a solução alternativa, sem o operador- λ , depois de *R3* teríamos $(PERM \ A)$, ao invés de $\lambda x_0. \lambda x_1. ((A \ x_1) \ x_0)$; e depois de *R1* teríamos $((PERM \ A) \ p)$ (o conjunto dos indivíduos que Pedro ama), ao invés de $(\lambda x_0. \lambda x_1. ((A \ x_1) \ x_0) \ p)$. Isto tornaria a redução- β desnecessária.

Pedro		ama		Maria
$\frac{}{N : p}$	Lx	$\frac{}{(N \setminus S)/N : A}$	Lx	$\frac{}{N : m}$
		$\frac{}{N \setminus (S/N) : \lambda x_0. \lambda x_1. ((A x_1) x_0)}$	$R3$	
		$\frac{}{S/N : (\lambda x_0. \lambda x_1. ((A x_1) x_0) p)}$	$R1$	
		$\frac{}{=_{red.\beta} \lambda x_1. ((A x_1) p)}$		
		$\frac{}{S : (\lambda x_1. ((A x_1) p) m)}$	$R1$	

Podemos executar então o último passo, uma redução- β :

$$\begin{array}{c}
 (\lambda \ x_1 \ . \ \underbrace{((A x_1) p)} \ m) \\
 \downarrow \qquad \qquad \downarrow \qquad \downarrow \\
 (\lambda \ x_n \ . \ \tau \ \alpha) \iff \tau^{[x_n \mapsto \alpha]} \\
 \qquad \qquad \qquad \downarrow \\
 \qquad \qquad \qquad \underbrace{((A m) p)}
 \end{array}$$

O que resulta finalmente no diagrama da análise completa da sentença *Pedro ama Maria*:⁵

Pedro		ama		Maria
$\frac{}{N : p}$	Lx	$\frac{}{(N \setminus S)/N : A}$	Lx	$\frac{}{N : m}$
		$\frac{}{N \setminus (S/N) : \lambda x_0. \lambda x_1. ((A x_1) x_0)}$	$R3$	
		$\frac{}{S/N : (\lambda x_0. \lambda x_1. ((A x_1) x_0) p)}$	$R1$	
		$\frac{}{=_{red.\beta} \lambda x_1. ((A x_1) p)}$		
		$\frac{}{S : (\lambda x_1. ((A x_1) p) m)}$	$R1$	
		$\frac{}{=_{red.\beta} ((A m) p)}$		

Observe que, apesar de termos executado a derivação com passos diferentes da derivação feita na seção 3.2, chegamos à mesma representação semântica. A conclusão desta observação é que, independentemente da forma como as expressões foram internamente

⁵Sem o operador- λ , a representação semântica seria $((((PERM A) p) m)$. A desvantagem desta solução é que a equivalência de $((((PERM A) p) m)$ e $((A m) p)$ não é imediatamente demonstrável como a de $((\lambda x_0. \lambda x_1. ((A x_1) x_0) p) m)$ e $((A m) p)$ com a redução- β . Para isso, precisaríamos demonstrar que $\forall F. \forall x. \forall y. (((PERM F) y) x) \equiv ((F x) y)$.

combinadas (primeiro juntando *ama* a *Maria*, para depois juntar isto a *Pedro*; ou então primeiro concatenando *Pedro* a *ama*, para depois concatenar a *Maria*), o resultado semântico é o mesmo. Essa diferença de derivação, no entanto, terá um papel fundamental na explicação da coordenação dos chamados não-constituintes, por exemplo.

4.3 Exercícios

1. Na derivação de *Pedro ama Maria* através de *R3*, todas as reduções- β foram feitas imediatamente, assim que o operador- λ aparecia no contexto adequado para a redução; no entanto, isso não é necessário: as reduções poderiam ser todas feitas na última etapa da derivação. Refaça a derivação de *Pedro ama Maria* deixando para fazer as reduções- β todas no final.
2. Faça a derivação de *Maria odeia João*, aplicando *R3* em *odeia*. Faça uma reduzindo imediatamente, e outra com as reduções todas no fim.

Capítulo 5

Regra de Composição (*R2*)

Intuitivamente, a regra de composição pode ser explicada pela mesma metáfora do cancelamento de frações, só que agora quando o numerador de uma fração é igual ao denominador da outra: $\frac{2}{3} \times \frac{3}{5} = \frac{2}{5}$. Na sua parte semântica, o que a regra *R2* faz é a composição de duas funções (daí seu nome).

Esta regra é definida da seguinte maneira:

- $X/Y : \phi \quad \oplus \quad Y/Z : \psi \quad \Longleftrightarrow \quad X/Z : \lambda x_n.(\phi (\psi x_n))$
- $Z \backslash Y : \psi \quad \oplus \quad Y \backslash X : \phi \quad \Longleftrightarrow \quad Z \backslash X : \lambda x_n.(\phi (\psi x_n))$

[para uma variável x_n nova na derivação]

Em ambas as fórmulas, duas expressões funcionais são combinadas, quando o resultado de uma é argumento adequado para a outra, resultando na composição funcional (representada pelo termo $\lambda x_n.(\phi (\psi x_n))$), mais uma vez para uma variável x_n que ainda não tenha sido usada na derivação).¹ Com *R2*, podemos incluir nas derivações as seguintes subderivações:

¹Estritamente falando, a termo $\lambda x_n.(\phi (\psi x_n))$ não é propriamente uma composição funcional, mas uma espécie de ‘atrasador’ da aplicação funcional: uma função ϕ é concatenada a uma função ψ , de forma que elas ficam ‘esperando’ o argumento adequado para a função ψ , e só depois disso este resultado é aplicado à função ϕ . Normalmente a composição funcional é representada pelo símbolo $\bullet$, de forma que se imaginarmos que uma função g diz, para cada numeral romano, qual é seu correspondente arábico, e que uma função f diz de um numeral arábico se ele exprime um número par ou não, então a composição funcional $f \bullet g$ diria quais são os numerais romanos que exprimem um número par, como podemos ver abaixo.

$$\boxed{\frac{X/Y : \phi \quad Y/Z : \psi}{X/Z : \lambda x_n.(\phi (\psi x_n))} \quad R2} \quad \boxed{\frac{Z/Y : \psi \quad Y/X : \phi}{Z/X : \lambda x_n.(\phi (\psi x_n))} \quad R2}$$

5.1 Exemplo de uso de $R2$

Recorrendo as mesmas informações lexicais para as expressões *Pedro*, *ama* e *Maria*, e supondo ainda que a expressão *não* seja um item lexical da categoria $(N \setminus S)/(N \setminus S)$ e que denote $\lambda x_n. \lambda x_m. (\neg (x_n x_m))$ (de novo para variáveis x_n e x_m novas na derivação), podemos construir o diagrama abaixo através apenas da regra de inserção lexical.²

$$\boxed{\frac{\text{Pedro}}{N : p} \quad Lx \quad \frac{\text{não}}{(N \setminus S)/(N \setminus S) : \lambda x_0. \lambda x_1. (\neg (x_0 x_1))} \quad Lx \quad \frac{\text{ama}}{(N \setminus S)/N : A} \quad Lx \quad \frac{\text{Maria}}{N : m} \quad Lx}$$

Nesta condição, é possível empregar a regra $R2$ para juntar *não ama*, da seguinte maneira (como é preciso usar uma variável nova na derivação, e x_0 e x_1 já tinham sido usadas, empregamos agora x_2):

$$g = \begin{bmatrix} I & \rightarrow & 1 \\ II & \rightarrow & 2 \\ III & \rightarrow & 3 \\ IV & \rightarrow & 4 \\ V & \rightarrow & 5 \\ \vdots & \vdots & \vdots \end{bmatrix} \quad f = \begin{bmatrix} 1 & \rightarrow & F \\ 2 & \rightarrow & V \\ 3 & \rightarrow & F \\ 4 & \rightarrow & V \\ 5 & \rightarrow & F \\ \vdots & \vdots & \vdots \end{bmatrix} \quad f \bullet g = \begin{bmatrix} I & \rightarrow & F \\ II & \rightarrow & V \\ III & \rightarrow & F \\ IV & \rightarrow & V \\ V & \rightarrow & F \\ \vdots & \vdots & \vdots \end{bmatrix}$$

²O uso do símbolo $\neg$ para representar a negação proposicional é bastante comum, mas também são usados símbolos como $\sim$, $-$, ou mesmo uma barra sobre toda a expressão negada (como $\lambda x_n. \lambda x_m. (\overline{x_n x_m})$); a interpretação da negação normalmente é definida através de uma tabela de verdade como:

A	$\neg A$
V	F
F	V

$$\begin{array}{c}
\begin{array}{c} \underbrace{(N \setminus S)} \\ \downarrow \\ X \end{array} / \begin{array}{c} \underbrace{(N \setminus S)} \\ \downarrow \\ Y \end{array} : \underbrace{\lambda x_0. \lambda x_1. (\neg (x_0 \ x_1))}_{\downarrow \phi} \oplus \begin{array}{c} \underbrace{(N \setminus S)} \\ \downarrow \\ Y \end{array} / \begin{array}{c} \downarrow \\ Z \end{array} : \begin{array}{c} \downarrow \\ \psi \end{array} \\
\end{array} \iff \begin{array}{c} X / Z : \lambda x_n. (\begin{array}{c} \downarrow \phi \end{array} \begin{array}{c} \downarrow (\psi \ x_n) \end{array}) \\
\downarrow \quad \downarrow \quad \downarrow \\
\begin{array}{c} \underbrace{(N \setminus S)} \\ \downarrow \end{array} / N : \lambda x_2. (\underbrace{\lambda x_0. \lambda x_1. (\neg (x_0 \ x_1))}_{\downarrow} \begin{array}{c} \downarrow (A \ x_2) \end{array})
\end{array}$$

Como a fórmula que representa a interpretação semântica ainda permite uma redução- β , vejamos como ela é aplicada:

$$\begin{array}{c}
\lambda x_2. (\lambda \ x_0 \ . \ \underbrace{\lambda x_1. (\neg (x_0 \ x_1))}_{\downarrow \tau} \ \underbrace{(A \ x_2)}_{\downarrow \alpha}) \\
\downarrow \quad \downarrow \quad \downarrow \\
(\lambda \ x_n \ . \ \tau \ \alpha) \iff \tau^{[x_n \mapsto \alpha]} \\
\downarrow \\
\lambda x_2. \ \underbrace{\lambda x_1. (\neg ((A \ x_2) \ x_1))}
\end{array}$$

Com isto, podemos expandir a derivação, da seguinte forma:

$\frac{\text{Pedro}}{N : p} \quad Lx$	$\frac{\text{não}}{(N \setminus S) / (N \setminus S) : \lambda x_0. \lambda x_1. (\neg (x_0 \ x_1))} \quad Lx$	$\frac{\text{ama}}{(N \setminus S) / N : A} \quad Lx$	$\frac{\text{Maria}}{N : m} \quad Lx$
$\frac{\quad}{(N \setminus S) / N : \lambda x_2. (\lambda x_0. \lambda x_1. (\neg (x_0 \ x_1)) (A \ x_2))} \quad R2$			
$\quad \quad \quad =_{red.\beta} \lambda x_2. \lambda x_1. (\neg ((A \ x_2) \ x_1))$			

Podemos agora usar a regra $R1$ para juntar *não ama* e *Maria*:

$$\begin{array}{c}
\begin{array}{c} \underbrace{(N \setminus S)} \\ \downarrow \\ X \end{array} / \begin{array}{c} \downarrow \\ Y \end{array} : \underbrace{\lambda x_2. \lambda x_1. (\neg ((A \ x_2) \ x_1))}_{\downarrow \phi} \oplus \begin{array}{c} \downarrow \\ Y \end{array} : \begin{array}{c} \downarrow \\ \alpha \end{array} \\
\end{array} \iff \begin{array}{c} X : (\begin{array}{c} \downarrow \phi \end{array} \begin{array}{c} \downarrow \alpha \end{array}) \\
\downarrow \quad \downarrow \\
\begin{array}{c} \underbrace{N \setminus S} \\ \downarrow \end{array} : (\underbrace{\lambda x_2. \lambda x_1. (\neg ((A \ x_2) \ x_1))}_{\downarrow} m)
\end{array}$$

E depois, podemos fazer a seguinte redução- β :

$$\begin{array}{c}
 (\lambda \ x_2 \ . \ \underbrace{\lambda x_1.(\neg ((A \ x_2) \ x_1))}_{\downarrow} \ m \) \\
 \downarrow \qquad \qquad \downarrow \qquad \qquad \downarrow \\
 (\lambda \ x_n \ . \ \qquad \qquad \tau \qquad \qquad \alpha \) \iff \tau^{[x_n \mapsto \alpha]} \\
 \qquad \qquad \qquad \qquad \qquad \qquad \qquad \downarrow \\
 \qquad \qquad \qquad \qquad \qquad \qquad \qquad \underbrace{\lambda x_1.(\neg ((A \ m) \ x_1))}_{\downarrow}
 \end{array}$$

Assim, um novo passo pode ser acrescentado à derivação:

$\frac{\text{Pedro}}{N : p} \quad Lx$	$\frac{\text{não}}{(N \setminus S)/(N \setminus S) : \lambda x_0. \lambda x_1. (\neg (x_0 \ x_1))} \quad Lx$	$\frac{\text{ama}}{(N \setminus S)/N : A} \quad Lx$	$\frac{\text{Maria}}{N : m} \quad Lx$
$\frac{(N \setminus S)/(N \setminus S) : \lambda x_0. \lambda x_1. (\neg (x_0 \ x_1)) \quad Lx}{(N \setminus S)/N : \lambda x_2. (\lambda x_0. \lambda x_1. (\neg (x_0 \ x_1)) (A \ x_2))} \quad R2$		$\frac{(N \setminus S)/N : \lambda x_2. (\lambda x_0. \lambda x_1. (\neg (x_0 \ x_1)) (A \ x_2))}{=_{red.\beta} \lambda x_2. \lambda x_1. (\neg ((A \ x_2) \ x_1))} \quad R1$	
$\frac{(N \setminus S)/N : \lambda x_2. (\lambda x_0. \lambda x_1. (\neg (x_0 \ x_1)) (A \ x_2))}{=_{red.\beta} \lambda x_2. \lambda x_1. (\neg ((A \ x_2) \ x_1))} \quad R1$			

Finalmente, mais um uso da regra $R1$, juntando *Pedro* e *não ama Maria*, e outro da redução- β , como abaixo:

$$\begin{array}{c}
 N : p \quad N \setminus S : \underbrace{\lambda x_1.(\neg ((A \ m) \ x_1))}_{\downarrow} \\
 \downarrow \quad \downarrow \quad \downarrow \quad \downarrow \\
 Y : \alpha \oplus Y \setminus X : \phi \iff X : (\quad \phi \quad \alpha \) \\
 \downarrow \quad \downarrow \quad \downarrow \\
 S : (\quad \underbrace{\lambda x_1.(\neg ((A \ m) \ x_1))}_{\downarrow} \quad p \) \\
 \\
 (\lambda \ x_1 \ . \ \underbrace{(\neg ((A \ m) \ x_1))}_{\downarrow} \ p \) \\
 \downarrow \quad \downarrow \quad \downarrow \\
 (\lambda \ x_n \ . \ \qquad \qquad \tau \qquad \qquad \alpha \) \iff \tau^{[x_n \mapsto \alpha]} \\
 \qquad \qquad \qquad \qquad \qquad \qquad \qquad \downarrow \\
 \qquad \qquad \qquad \qquad \qquad \qquad \qquad \underbrace{(\neg ((A \ m) \ p))}_{\downarrow}
 \end{array}$$

terminam nossa derivação numa expressão totalmente conexa:

Pedro	não	ama	Maria
$\frac{}{N : p} \quad Lx$	$\frac{}{(N \setminus S)/(N \setminus S) : \lambda x_0. \lambda x_1. (\neg (x_0 \ x_1))} \quad Lx$	$\frac{}{(N \setminus S)/N : A} \quad Lx$	$\frac{}{N : m} \quad Lx$
	$\frac{}{(N \setminus S)/N : \lambda x_2. (\lambda x_0. \lambda x_1. (\neg (x_0 \ x_1)) (A \ x_2))} \quad R2$		
	$\quad \quad \quad =_{red.\beta} \lambda x_2. \lambda x_1. (\neg ((A \ x_2) \ x_1))$		
	$\frac{}{N \setminus S : (\lambda x_2. \lambda x_1. (\neg ((A \ x_2) \ x_1)) \ m)} \quad R1$		
	$\quad \quad \quad =_{red.\beta} \lambda x_1. (\neg ((A \ m) \ x_1))$		
	$\frac{}{S : (\lambda x_1. (\neg ((A \ m) \ x_1)) \ p)} \quad R1$		
	$\quad \quad \quad =_{red.\beta} (\neg ((A \ m) \ p))$		

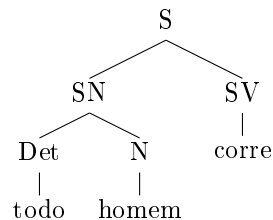
Capítulo 6

Regra de promoção ($R4$)

Algumas vezes, a última regra da Gramática Categorial Livre também é chamada de deslocamento de tipo (*type shifting*) ou alçamento de tipo (*type raising*); estamos preferindo aqui o termo “promoção” para evitar qualquer confusão terminológica, já que os outros termos estão fortemente associados à Gramática Gerativa.

Uma versão desta mesma regra desempenhou um papel fundamental no desenvolvimento recente da semântica formal. Desde que Russell [22] postulou um formato para a interpretação dos sintagmas nominais (SN) quantificados, cuja estrutura estava em desacordo com a estrutura sintática tradicionalmente admitida,¹ era impossível tratar composicionalmente a interpretação destes SN; além disso, eram necessários procedimentos diferentes para interpretar um SN quantificado e um SN composto apenas por um nome próprio. Este problema só foi resolvido quando Montague, usando o operador- λ , atribuiu a este SN quantificado uma categoria funcional cuja representação semântica era $\lambda P[\forall x[H(x) \rightarrow P(x)]]$; além disso, postulando uma regra que transforma a representação do nome próprio de p para

¹Numa gramática de estrutura sintagmática, esta estrutura está relacionada às regras $S \rightarrow SN\ SV$ e $SN \rightarrow Det\ N$, que, para uma expressão como *Todo homem corre*, são responsáveis por uma árvore como:



que tem como representação semântica a fórmula $\forall x[H(x) \rightarrow C(x)]$, cuja estrutura é:

$\lambda P[P(p)]$ (junto com a mudança categorial de N para $S/(N \setminus S)$), Montague conseguiu inclusive unificar o tratamento semântico dos SN.

De um ponto de vista intuitivo, da mesma maneira que deduzimos que *corre* era um $N \setminus S$, podemos também deduzir que *Pedro* também é um $S/(N \setminus S)$: se *Pedro corre* é um S , e *corre* um $N \setminus S$, então *Pedro* também pode ser uma expressão que se combina com uma expressão de tipo $N \setminus S$, depois dela, para resultar numa sentença.

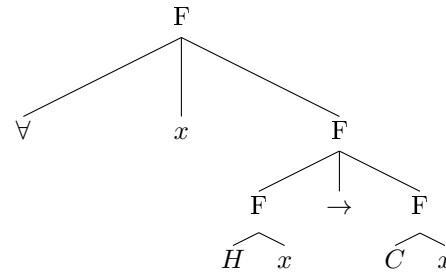
Funcionalmente, o que a regra *R4* faz é transformar uma expressão argumental numa expressão funcional que toma como argumento uma expressão funcional da qual ela seria argumento, resultando numa expressão do mesmo tipo que resultaria da aplicação da função inicial ao argumento. Essa operação é definida pelo par de fórmulas abaixo:

- $X : \alpha \quad \Longleftrightarrow \quad Y/(X \setminus Y) : \lambda x_n.(x_n \alpha)$
- $X : \alpha \quad \Longleftrightarrow \quad (Y/X) \setminus Y : \lambda x_n.(x_n \alpha)$

[para uma variável x_n nova na derivação]

Observe que, em ambos os casos, uma expressão qualquer (independentemente de sua categoria) também pertence a outras categorias funcionais mais complexas. Intuitivamente, esta idéia pode ser explicada usando uma reflexão dedutiva: da mesma forma como se deduz, a partir das informações que *Pedro* é um N e que *Pedro corre* é um S , que *corre* é $N \setminus S$, porque ele toma um N anterior para resultar num S , também podemos deduzir que *Pedro* ainda pode ser um $S/(N \setminus S)$, porque ele também pode ser considerado uma expressão que toma um $N \setminus S$ posterior para resultar num S (o mesmo raciocínio valeria caso o argumento estivesse originalmente depois da função, bastando inverter os qualificadores posicionais). Na sua parte semântica, o operador- λ garante apenas que a denotação funcional original continuará agindo como função da denotação argumental.

Na prática, a regra *R4* é mais simples do que as explicações acima. Com ela, os diagramas podem ser acrescidos pelos seguintes subdiagramas:



Observe que, na representação semântica, a interpretação de *homem* (H) e de *corre* (C) estão ligados mais coesamente, ao contrário da estrutura sintática, em que *todo* e *homem* são constituintes imediatos do SN.

$\frac{X : \alpha}{Y/(X \setminus Y) : \lambda x_n.(x_n \alpha)} \quad R4$	$\frac{X : \alpha}{(Y/X) \setminus Y : \lambda x_n.(x_n \alpha)} \quad R4$
--	--

6.1 Exemplos de uso de $R4$

Para exemplificar o uso de $R4$, vamos apresentar duas derivações: uma para a sentença *Pedro corre* e outra (a terceira neste texto) para a sentença *Pedro ama Maria*.

6.1.1 *Pedro corre*

Ainda que a primeira possa ser analisada empregando-se apenas uma vez a regra $R1$, faremos aqui uma derivação usando primeiro a regra $R4$ para transformar a categoria atômica de *Pedro* numa categoria funcional, para depois empregar $R1$ para aplicar a denotação funcional, modificada a partir da denotação original de *Pedro*, à denotação de *corre*. Recorrendo às suposições feitas para estes itens lexicais, depois de aplicada a regra Lx a eles teremos:

$\frac{\text{Pedro}}{N : p} \quad Lx$	$\frac{\text{corre}}{N \setminus S : C} \quad Lx$
---------------------------------------	---

Empregando a primeira versão da regra $R4$, fazendo o Y valer S e usando a variável x_0 (já que nenhuma variável ainda foi usada neste ponto da derivação), teremos:

$$\begin{array}{ccc}
 N & : & p \\
 \downarrow & & \downarrow \\
 X & : & \alpha
 \end{array}
 \iff
 \begin{array}{ccc}
 Y & / & (X \setminus Y) : \lambda x_n.(x_n \alpha) \\
 \downarrow & & \downarrow \\
 S & / & (N \setminus S) : \lambda x_0.(x_0 p)
 \end{array}$$

Assim, a construção da derivação ganha o seguinte passo:

Pedro	corre
$\frac{}{N : p} \quad Lx$	$\frac{}{N \setminus S : C} \quad Lx$
$\frac{}{S / (N \setminus S) : \lambda x_0. (x_0 p)} \quad R4$	

Aqui a regra $R1$ pode ser usada para juntar $Pedro$ e $corre$, e depois a denotação resultante pode ser reduzida, através das duas operações abaixo:

$$\begin{array}{c}
S \quad / \quad \underbrace{(N \setminus S)} \quad : \quad \underbrace{\lambda x_0. (x_0 p)} \quad \quad \underbrace{N \setminus S} \quad : \quad C \\
\downarrow \quad \quad \downarrow \quad \quad \downarrow \quad \quad \downarrow \quad \quad \downarrow \\
X \quad / \quad Y \quad : \quad \phi \quad \oplus \quad Y \quad : \quad \alpha \quad \Longleftrightarrow \quad X \quad : \quad (\quad \phi \quad \alpha \quad) \\
\quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \downarrow \quad \quad \downarrow \\
\quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad S \quad : \quad (\quad \underbrace{\lambda x_0. (x_0 p)} \quad C \quad) \\
\\
(\lambda \quad x_0 \quad . \quad \underbrace{(x_0 p)} \quad C \quad) \\
\downarrow \quad \quad \downarrow \quad \quad \downarrow \\
(\lambda \quad x_n \quad . \quad \tau \quad \alpha \quad) \quad \Longleftrightarrow \quad (\quad \tau^{[x_n \mapsto \alpha]} \quad) \\
\quad \quad \quad \quad \quad \quad \quad \quad \quad \downarrow \\
\quad \quad \quad \quad \quad \quad \quad \quad \quad (\quad C \quad p \quad)
\end{array}$$

Portanto, esta derivação termina com o seguinte diagrama totalmente conexo:

Pedro	corre
$\frac{}{N : p} \quad Lx$	$\frac{}{N \setminus S : C} \quad Lx$
$\frac{}{S / (N \setminus S) : \lambda x_0. (x_0 p)} \quad R4$	
$\frac{}{S : (\lambda x_0. (x_0 p) C)} \quad R1$	
$\quad \quad \quad =_{red.\beta} (C p)$	

6.1.2 *Pedro ama Maria*

A terceira derivação diferente (porém, chegaremos ao mesmo resultado) que apresentaremos aqui para a sentença *Pedro ama Maria* começa com a mesma aplicação da regra Lx :

Pedro	ama	Maria
$\frac{}{N : p} \quad Lx$	$\frac{}{(N \setminus S)/N : A} \quad Lx$	$\frac{}{N : m} \quad Lx$

A diferença começa quando empregamos a regra $R4$ em *Pedro*, atribuindo a Y o valor S e recorrendo à variável x_0 , já que nenhuma variável tinha sido usada na derivação:

$$\begin{array}{c}
 N : p \\
 \downarrow \\
 X : \alpha
 \end{array}
 \iff
 \begin{array}{c}
 Y / (X \setminus Y) : \lambda x_n . (x_n \alpha) \\
 \downarrow \quad \downarrow \quad \downarrow \\
 S / (N \setminus S) : \lambda x_0 . (x_0 p)
 \end{array}$$

Incluindo esta operação na derivação, temos:

Pedro	ama	Maria
$\frac{}{N : p} \quad Lx$	$\frac{}{(N \setminus S)/N : A} \quad Lx$	$\frac{}{N : m} \quad Lx$
$\frac{}{S/(N \setminus S) : \lambda x_0 . (x_0 p)} \quad R4$		

Com isto, podemos recorrer a $R2$ para juntar *Pedro* e *ama*, e como já usamos a variável x_0 , vamos usar agora x_1 ; depois também podemos executar a redução- β na representação semântica:

$$\begin{array}{c}
 S / \underbrace{(N \setminus S)} : \underbrace{\lambda x_0 . (x_0 p)} \quad \underbrace{(N \setminus S)} / N : A \\
 \downarrow \quad \downarrow \quad \downarrow \quad \oplus \quad \downarrow \quad \downarrow \quad \downarrow \\
 X / Y : \phi \oplus Y / Z : \psi
 \end{array}
 \iff
 \begin{array}{c}
 X / Z : \lambda x_n . (\phi \quad (\psi \ x_n)) \\
 \downarrow \quad \downarrow \quad \downarrow \quad \downarrow \quad \downarrow \\
 S / N : \lambda x_1 . (\underbrace{\lambda x_0 . (x_0 p)} \quad (A \ x_1))
 \end{array}$$

Chegamos assim à seguinte derivação:

Pedro	ama	Maria
$\frac{}{N : p} \quad Lx$	$\frac{}{(N \setminus S)/N : A} \quad Lx$	$\frac{}{N : m} \quad Lx$
$\frac{}{S/(N \setminus S) : \lambda x_0.(x_0 p)} \quad R4$		
$\frac{}{S/N : \lambda x_1.(\lambda x_0.(x_0 p) (A x_1))} \quad R2$		

Podemos aqui executar a redução- β internamente:

$$\begin{array}{c}
 \lambda x_1. (\lambda x_0. \underbrace{(x_0 p)}_{\downarrow \tau} \underbrace{(A x_1)}_{\downarrow \alpha}) \\
 \downarrow \\
 (\lambda x_n. \underbrace{\tau}_{\downarrow} \underbrace{\alpha}_{\downarrow}) \iff \tau^{[x_n \mapsto \alpha]} \\
 \downarrow \\
 \lambda x_1. \overbrace{((A x_1) p)}
 \end{array}$$

Com esta redução, a derivação passa a ser:

Pedro	ama	Maria
$\frac{}{N : p} \quad Lx$	$\frac{}{(N \setminus S)/N : A} \quad Lx$	$\frac{}{N : m} \quad Lx$
$\frac{}{S/(N \setminus S) : \lambda x_0.(x_0 p)} \quad R4$		
$\frac{}{S/N : \lambda x_1.(\lambda x_0.(x_0 p) (A x_1))} \quad R2$		
$\quad \quad \quad =_{red.\beta} \lambda x_1.((A x_1) p)$		

Aqui, devemos usar $R1$:

$$\begin{array}{ccc}
S / N : \underbrace{\lambda x_1.((A x_1) p)} & N : m & \\
\downarrow \quad \downarrow & \downarrow \quad \downarrow & \\
X / Y : \phi & \oplus Y : \alpha & \Longleftrightarrow X : (\begin{array}{cc} \phi & \alpha \\ \downarrow & \downarrow \end{array}) \\
& & S : (\underbrace{\lambda x_1.((A x_1) p)} m)
\end{array}$$

Desta maneira, chegamos a um diagrama totalmente conexo:

Pedro	ama	Maria
$\frac{}{N : p}$	$\frac{}{(N \setminus S)/N : A}$	$\frac{}{N : m}$
$\frac{}{S/(N \setminus S) : \lambda x_0.(x_0 p)}$		
$\frac{S/N : \lambda x_1.(\lambda x_0.(x_0 p) (A x_1))}{=_{red.\beta} \lambda x_1.((A x_1) p)}$		
$\frac{}{S : (\lambda x_1.((A x_1) p) m)}$		

Finalmente, através de uma última redução- β , se chega ao resultado final:

$$\begin{array}{ccc}
(\lambda x_1. \underbrace{((A x_1) p)} m) & & \\
\downarrow \quad \downarrow \quad \downarrow & & \\
(\lambda x_n. \tau \alpha) & \Longleftrightarrow & \tau^{[x_n \mapsto \alpha]} \\
& & \downarrow \\
& & ((A m) p)
\end{array}$$

Pedro		ama		Maria	
$\frac{}{N : p}$	Lx	$\frac{}{(N \setminus S)/N : A}$	Lx	$\frac{}{N : m}$	Lx
$\frac{}{S/(N \setminus S) : \lambda x_0.(x_0 p)}$	$R4$				
$\frac{S/N : \lambda x_1.(\lambda x_0.(x_0 p) (A x_1))}{=_{red.\beta} \lambda x_1.((A x_1) p)}$	$R2$				
$\frac{S : (\lambda x_1.((A x_1) p) m)}{=_{red.\beta} ((A m) p)}$	$R1$				

Capítulo 7

Dedução de informações lexicais

Estivemos até agora usando a GCL para construir análises gramaticais, no entanto, ela também pode ser usada para descobrir informações lexicais.

Suponhamos que não conhecêssemos as informações lexicais relacionadas à expressão *não*; conhecendo todas as informações lexicais de outras expressões com as quais *não* se combina numa expressão complexa, e admitindo que esta combinação aconteça de uma determinada maneira, podemos ‘calcular’ as informações lexicais que *não* precisa possuir para poder entrar coerentemente na combinação. Usando então a sentença *Pedro não ama Maria*, teríamos a seguinte disposição:

Pedro	não	ama	Maria
$\frac{}{N : p} \quad Lx$	$\frac{}{?_0} \quad Lx$	$\frac{}{(N \setminus S)/N : A} \quad Lx$	$\frac{}{N : m} \quad Lx$

A interrogação sob *não* indica que ainda não sabemos nada a respeito das informações lexicais dela. Mas como *ama* e *Maria* são combináveis através de *R1*, vamos supor que isto efetivamente aconteça. Assim, a derivação fica:

Pedro	não	ama	Maria
$\frac{}{N : p} \quad Lx$	$\frac{}{?_0} \quad Lx$	$\frac{}{(N \setminus S)/N : A} \quad Lx$	$\frac{}{N : m} \quad Lx$
$\frac{}{N \setminus S : (A \ m)} \quad R1$			

Vamos supor agora que cada uma das outras duas expressões também se combinem com o resto da expressão complexa através de *R1*. Esta suposição é apoiada por um lado pela tradição gramatical em analisar esta sentença como sendo a combinação do sujeito *Pedro* com a predicado *não ama Maria*, e o predicado por sua vez através do acréscimo do advérbio de negação *não* ao predicado *ama Maria*; além disso, de um ponto de vista hipotético-dedutivo, esta suposição também tem a vantagem de assumir a estrutura mais simples — se conseguirmos chegar a um resultado satisfatório a partir desta suposição, teremos feito isto da forma mais simples que nossa teoria nos permite. Por suposição, então, a derivação completa seria:

Pedro	não	ama	Maria
$\frac{}{N : p} \quad Lx$	$\frac{}{?_0} \quad Lx$	$\frac{}{(N \setminus S)/N : A} \quad Lx$	$\frac{}{N : m} \quad Lx$
		$\frac{}{N \setminus S : (A m)} \quad R1$	
		$\frac{}{?_1} \quad R1$	
		$\frac{}{?_2} \quad R1$	

As duas novas interrogações significam que ainda não sabemos nada nem sobre a expressão toda, nem sobre o predicado. Mas para que possamos começar a deduzir quais são as informações sobre o item lexical *não*, precisamos ter estas informações, para aplicar um procedimento dedutivo semelhante à chamada regra de três. Assim, vamos aceitar que *Pedro não ama Maria* seja uma sentença, e que sua representação semântica seja $(\neg ((A m) p))^1$ — seguindo, por exemplo, a explicação de Chierchia [6, p. 202–207]. Assim, podemos acrescentar estas informações ao final da derivação:

¹Esta negação normalmente é explicada proposicionalmente como ‘não é verdade que Pedro ama Maria’; assim, como a representação semântica de *Pedro ama Maria* é $((A m) p)$, a negação é entendida como um operador proposicional que inverte o valor de verdade da proposição com a qual ele se combina (se α é verdadeiro, $(\neg \alpha)$ é falso; e se α é falso, $(\neg \alpha)$ é verdadeiro. Funcionalmente, a negação pode ser representada da seguinte maneira:

$$\neg = \begin{bmatrix} V & \rightarrow F \\ F & \rightarrow V \end{bmatrix}$$

Pedro	não	ama	Maria
$\frac{}{N : p} \quad Lx$	$\frac{}{?_0} \quad Lx$	$\frac{}{(N \setminus S)/N : A} \quad Lx$	$\frac{}{N : m} \quad Lx$
		$\frac{}{N \setminus S : (A \ m)} \quad R1$	
		$\frac{}{?_1} \quad R1$	
		$\frac{}{S : (\neg ((A \ m) \ p))} \quad R1$	

Agora sabemos que $S : (\neg ((A \ m) \ p))$ é o resultado de juntar através de $R1$ $N : p$ a alguma coisa que ainda não conhecemos. Podemos, portanto, desfazer a regra $R1$ para descobrir o valor desconhecido; como a categoria de *Pedro* é atômica, ela é inadequada à primeira versão de $R1$, de forma que precisamos usar então a outra versão:

$$\begin{array}{c}
N : p \\
\downarrow \\
Y : \alpha \oplus \overbrace{Y \setminus X : \phi}^{?_1} \Rightarrow \begin{array}{c} S : (\ ?'_1 \ p \) \\ \downarrow \quad \downarrow \downarrow \\ X : (\ \phi \ \alpha \) \end{array} \\
\downarrow \quad \downarrow \quad \downarrow \\
N \setminus S : ?'_1
\end{array}$$

O problema que é preciso resolver agora é: qual o valor adequado a se atribuir a ϕ de forma que $(\neg ((A \ m) \ p))$ seja resultado da aplicação funcional deste valor a p ? Para se encontrar este valor é preciso retirar de $(\neg ((A \ m) \ p))$ o p , de forma que a derivação pode ser expandida da seguinte maneira:

Pedro	não	ama	Maria
$\frac{}{N : p} \quad Lx$	$\frac{}{?_0} \quad Lx$	$\frac{}{(N \setminus S)/N : A} \quad Lx$	$\frac{}{N : m} \quad Lx$
		$\frac{}{N \setminus S : (A \ m)} \quad R1$	
		$\frac{}{N \setminus S : ?'_1} \quad R1$	
	$\frac{}{S : \ (?'_1 \ p)} \quad R1$		
	$\quad =_{red.\beta} \ (\neg ((A \ m) \ p))$		

Esta operação é feita através da expansão- β :²

$$\begin{array}{ccc}
 & p & (\neg ((A \ m) \ p)) \\
 & \downarrow & \downarrow \\
 & \alpha & \tau^{[x_n \mapsto \alpha]} \\
 & \iff & \\
 \underbrace{\left(\lambda x_n . \quad \tau \right)}_{?'_1} & & \\
 \downarrow & & \\
 \lambda x_1 . \quad \overbrace{(\neg ((A \ m) \ x_1))} & &
 \end{array}$$

Assim, acabamos de descobrir que o valor de $'_1$ deve ser $\lambda x_1.(\neg ((A \ m) \ x_1))$, completando um pouco mais a derivação:

²A expansão- β é a conversão- β usada da direita para a esquerda : $(\lambda x_n. \tau \ \alpha) \Leftarrow \tau^{[x_n \mapsto \alpha]}$.

Pedro	não	ama	Maria
$\frac{}{N : p} \quad Lx$	$\frac{}{?_0} \quad Lx$	$\frac{}{(N \setminus S)/N : A} \quad Lx$	$\frac{}{N : m} \quad Lx$
		$\frac{}{N \setminus S : (A \ m)} \quad R1$	
	$\frac{}{N \setminus S : \lambda x_1.(\neg ((A \ m) \ x_1))} \quad R1$		
	$\frac{}{S : (\lambda x_1.(\neg ((A \ m) \ x_1)) \ p)} \quad R1$		
	$\frac{}{=_{red.\beta} (\neg ((A \ m) \ p))}$		

Neste ponto, a única incógnita está relacionada às informações lexicais de *não*, que era justamente o que gostaríamos de descobrir. Para chegar a ela, basta empregar exatamente o mesmo procedimento que acabamos de usar: primeiro desfazer *R1* entre *não* e *ama Maria*, e depois desfazer uma redução- β .

Para desfazer *R1* agora, precisamos ser um pouco mais cuidadosos, porque *ama Maria* é uma expressão funcional e, por isso, podíamos achar que a aplicação se dá desta para a outra. Contudo, se observarmos os valores disponíveis, veremos que a operação exigiria que um $N \setminus S$ tomasse antes alguma coisa que resultasse num mesmo $N \setminus S$; mas um $N \setminus S$ só pode ser aplicado a um N , o que resultaria num S , e não num $N \setminus S$. Portanto, a única opção é primeira versão de *R1*:

$$\begin{array}{c}
\overbrace{X \quad / \quad Y \quad : \quad \phi}^{?_0} \oplus \underbrace{Y \quad : \quad \alpha}_{N \setminus S : (A \ m)} \iff \underbrace{X \quad : \quad (\phi \quad \alpha)}_{N \setminus S : (\ ?'_0 \quad (A \ m) \)} \\
\downarrow \quad \downarrow \quad \downarrow \quad \downarrow \quad \downarrow \quad \downarrow \\
\overbrace{(N \setminus S) \quad / \quad (N \setminus S) \quad : \quad ?'_0}
\end{array}$$

A questão agora passa a ser: qual o valor que deve ser atribuído a ϕ de forma que $\lambda x_1.(\neg ((A \ m) \ x_1))$ resulte da aplicação funcional deste valor a $(A \ m)$:

Pedro	não	ama	Maria
$\frac{}{N : p} \quad Lx$	$\frac{}{(N \setminus S)/(N \setminus S) : ?'_0} \quad Lx$	$\frac{}{(N \setminus S)/N : A} \quad Lx$	$\frac{}{N : m} \quad Lx$
		$\frac{}{N \setminus S : (A \ m)} \quad R1$	
	$\frac{}{N \setminus S : (?'_0 (A \ m))} \quad R1$		
	$\frac{}{=_{red,\beta} \lambda x_1.(\neg ((A \ m) \ x_1))} \quad R1$		
	$\frac{}{S : (\lambda x_1.(\neg ((A \ m) \ x_1)) \ p)} \quad R1$		
	$\frac{}{=_{red,\beta} (\neg ((A \ m) \ p))}$		

Recorreremos novamente à expansão- β :

$$\begin{array}{c}
 \overbrace{\lambda x_n . \tau}^{?'_0} \quad \underbrace{(A \ m)}_{\downarrow \alpha} \quad \underbrace{\lambda x_1.(\neg ((A \ m) \ x_1))}_{\downarrow \tau[x_n \mapsto \alpha]} \\
 \downarrow \quad \downarrow \\
 \lambda x_0 . \lambda x_1.(\neg (x_0 \ x_1))
 \end{array}
 \iff
 \begin{array}{c}
 \lambda x_n . \lambda x_1.(\neg (x_0 \ x_1))
 \end{array}$$

Com isto, conseguimos fechar totalmente nossa derivação, encontrando as informações lexicais relativas a *não*:

Pedro	não	ama	Maria
$\frac{}{N : p} \quad Lx$	$\frac{}{(N \setminus S)/(N \setminus S) : \lambda x_0. \lambda x_1. (\neg (x_0 \ x_1))} \quad Lx$	$\frac{}{(N \setminus S)/N : A} \quad Lx$	$\frac{}{N : m} \quad Lx$
		$\frac{}{N \setminus S : (A \ m)} \quad R1$	
			$\frac{}{N \setminus S : (\lambda x_0. \lambda x_1. (\neg (x_0 \ x_1)) (A \ m))} \quad R1$
			$\frac{}{=_{red.\beta} \lambda x_1. (\neg ((A \ m) \ x_1))}$
			$\frac{}{S : (\lambda x_1. (\neg ((A \ m) \ x_1)) \ p)} \quad R1$
			$\frac{}{=_{red.\beta} (\neg ((A \ m) \ p))}$

Como acontece com as regras, quando uma variável é introduzida na derivação, ela sempre precisa ser uma variável que ainda não foi empregada antes — não faria sentido que a representação da denotação de *não* fosse $\lambda x_0. \lambda x_1. (\neg (x_0 \ x_1))$, com variáveis pré-definidas, porque quando ele fosse usado em derivações nas quais estas variáveis já tivessem sido usadas, isto causaria ligações indevidas entre elas. Portanto, uma representação lexical adequada precisa ser $\lambda x_n. \lambda x_m. (\neg (x_n \ x_m))$, para variáveis x_n e x_m novas na derivação. Para que essas informações possam ser usadas pela regra Lx , é preciso registrar que $\langle \text{não}, (N \setminus S)/(N \setminus S), \lambda x_0. \lambda x_1. (\neg (x_0 \ x_1)) \rangle \in \text{Léxico}$ (para variáveis x_n e x_m novas na derivação). Observe que o resultado obtido é o mesmo que foi usado na exemplificação de R_4 (na Seção 5.1, p. 22).

7.1 Exercícios

1. Deduza as informações lexicais associadas ao advérbio *muito*, como usado na sentença *Pedro corre muito*. Considere inicialmente que ele é advérbio adsentencial [9, p. 407], e o significado da sentença inteira é $((M \ C) \ p)$.
2. A partir do resultado do exercício anterior, é possível fazer uma segunda derivação da mesma sentença *Pedro corre muito*, na qual o verbo e o advérbio se combinam imediatamente, usando R_4 no advérbio. Observe que, com esse resultado, a discussão sobre o escopo do advérbio (se *muito* se combina só com o verbo ou com toda a sentença) passa a ser muito mais semântica do que sintática (o que ele modifica na sentença: a forma como Pedro está correndo, ou só a propriedade de correr).
3. Considerando que o significado da sentença *um menino corre todo menino corre* são normalmente representados pelas fórmula $\exists x_0[(M \ x_0) \wedge (C \ x_0)]$ e $\forall x_0[(M \ x_0) \rightarrow (C \ x_0)]$, deduza as informações lexicais de *menino*, *um* e *todo*; considere também que *menino* é um nome comum (que tem distribuição diferente dos sintagmas nominais) e, portanto, sua classe sintática corresponde a N_c .

Capítulo 8

Conclusão

O leitor que acompanhou a leitura até aqui pode querer aprofundar seus conhecimentos, e a recomendação aqui é que ele tente seguir a lista de referências bibliográficas apresentadas no final do primeiro parágrafo da introdução deste texto.

Um dos principais avanços obtidos pela GC foi o de oferecer uma solução elegante para a chamada conjunção booleana, sem precisar recorrer a categorias vazias e permitindo a sua generalização mesmo para a conjunção de expressões que não são constituintes tradicionais. Assim, sentenças como *Pedro ama e João odeia Maria* podem ser construídas da seguinte maneira (a parte semântica está sendo omitida apenas por brevidade, mas ela também é compatível com a derivação):

Pedro		ama		e		João		odeia		Maria	
$\frac{}{N}$	Lx	$\frac{}{(N \backslash S)/N}$	Lx	$\frac{}{((S/N) \backslash (S/N))/(S/N)}$	Lx	$\frac{}{N}$	Lx	$\frac{}{(N \backslash S)/N}$	Lx	$\frac{}{N}$	Lx
		$\frac{}{N \backslash (S)/N}$	$R3$					$\frac{}{N \backslash (S)/N}$	$R3$		
		$\frac{}{S/N}$	$R2$					$\frac{}{S/N}$	$R2$		
				$\frac{}{(S/N) \backslash (S/N)}$					R		
				$\frac{}{S/N}$					$R1$		
				$\frac{}{S}$							$R1$

Ainda que, para esta derivação, a classe sintática de e tenha sido estabelecida como $((S/N) \setminus (S/N)) / (S/N)$, diferentemente da classe da conjunção de sentença (que seria $(S \setminus S) / S$), há uma solução mais geral, através da postulação de uma regra que determina o polimorfismo da conjunção [15, p. 77]:¹

$$\bullet (X \setminus X) / X : \phi \iff (Y \setminus Y) / Y : \lambda x_n. \lambda x_m. \lambda l. ((\phi (x_0 \ x_l)) (x_m \ x_l))$$

$$\left[\begin{array}{l} \text{tal que:} \\ Y \text{ pode ser } Z/W \text{ ou } Z \setminus W, \\ x_n \text{ e } x_m \text{ são de tipo } (tipo \ W), \\ x_l \text{ é de tipo } (tipo \ Z)^{(tipo \ W)} \text{ e} \\ x_n, \ x_m \text{ e } x_l \text{ são novas na derivação} \end{array} \right]$$

A partir desta regra e da atribuição lexical $\langle e, (S \setminus S) / S, \wedge \rangle \in \text{Léxico}$, podemos derivar a categoria $((S/N) \setminus (S/N)) / (S/N)$, para a derivação de *Pedro ama e João odeia Maria* (novamente, a interpretação semântica está sendo omitida por comodidade, mas é igualmente calculável). Mas há uma solução ainda mais simples para a conjunção no Cálculo de Lambek (cf. [5, Cap. 6]), que já fica além dos limites deste texto.

Depois da GCI, o próximo passo natural seria passar ao que Wood [25, p. 34] chama de *Gramática Categorial Clássica*, que inclui, além dos quatro pares de regras apresentadas aqui, mais dois pares (as regras de divisão normal (R_5) e divisão invertida (R_6)); na seqüência, o Cálculo de Lambek constitui uma generalização importante das GCs (há, por exemplo, um outro conectivo categorial, além de ‘/’ e ‘\’ — o produto ‘•’; e as regras são apresentadas aos pares (uma de introdução e outra de eliminação) para cada conectivo categorial), e também deve ser aprendido pelos mais interessados em continuar desenvolvendo pesquisas como esta ferramenta de análise. Além do artigo “The mathematics of sentence structure”, do próprio Lambek [11], outros clássicos são os artigos “Die syntaktische Konnexität”, de Ajdukiewicz [1], e “A quasi-arithmetical notation for syntactic description”, de Bar-Hillel [2] (que juntos estabeleceram o chamado *Modelo-AB* [25, p. 34], que é composto apenas pelo que corresponde à regra $R1$).² Textos mais avançados são os livros de Moortgat, van Benthem, Morrill e Carpenter [14, 3, 15, 5, 16].

Afinal, a pretensão do presente texto era mesmo a de apenas introduzir e interessar possíveis futuros praticantes.

¹Aqui a conjunção é representada também funcionalmente; assim, ao invés de escrever $p \wedge q$, escreve-se $((\wedge \ q) \ p)$, com a mesma interpretação.

²As traduções dos dois primeiros textos podem ser encontradas em minha página de internet (<http://www.ufpr.br/~arthur>), além da *Introdução às Gramáticas Categoriais*, do Borges [18].

Apêndice A

Variantes notacionais

A.1 Ajdukiewicz [1]

$$\boxed{\frac{\alpha}{\beta} \oplus \beta \rightarrow \alpha}$$

A.2 Bar-Hillel [2]

$$\boxed{\beta \oplus \frac{\alpha}{(\beta)} \rightarrow \alpha \quad \frac{\alpha}{[\beta]} \oplus \beta \rightarrow \alpha \quad \beta \times \frac{\alpha}{(\beta)[\gamma]} \oplus \gamma \rightarrow \alpha}$$

A.3 Lambek [11]

$$\boxed{\alpha/\beta \oplus \beta = \alpha \quad \beta \oplus \beta \backslash \alpha = \alpha}$$

A.4 Geach [8]

$$\boxed{: \alpha \beta \oplus \beta \rightarrow \alpha}$$

A.5 Steedman [23]

$$\boxed{\alpha / \beta \oplus \beta = \alpha \mid \beta \oplus \alpha \backslash \beta = \alpha}$$

A.6 Jacobson [10]

$$\boxed{\alpha /_D \beta \oplus \beta \rightarrow \alpha \mid \beta \oplus \alpha /_E \beta \rightarrow \alpha}$$

A.7 Estrutura de Características [19]

$\left[\begin{array}{cc} res & \alpha \\ dep & \beta \end{array} \right] \oplus \beta \rightarrow \alpha$	$\beta \oplus \left[\begin{array}{cc} res & \alpha \\ ant & \beta \end{array} \right] \rightarrow \alpha$
ou	
$\left[\begin{array}{cc} res & \alpha \\ arg & \beta \\ pos & depois \end{array} \right] \oplus \beta \rightarrow \alpha$	$\beta \oplus \left[\begin{array}{cc} res & \alpha \\ arg & \beta \\ pos & antes \end{array} \right] \rightarrow \alpha$

Referências Bibliográficas

- [1] Kazimiers Ajdukiewicz. Die syntaktische konnexität. *Studia Philosophica*, 1:1–27, 1935.
- [2] Yehoshua Bar-Hillel. A quasi-arithmetical notation for syntactic description. *Language*, 29:47–58, 1953.
- [3] Johan F. A. K. van Benthem. *Language in Action -- Categories, Lambdas and Dynamic Logic*. North-Holland & MIT Press, Amsterdam & Cambridge, 1995.
- [4] Ronnie Cann. *Formal Semantics*. Cambridge University Press, Cambridge, 1993.
- [5] Bob Carpenter. *Type-Logical Semantics*. The MIT Press, Cambridge, MA, 1997.
- [6] Gennaro Chierchia. *Semântica*. Editora da Unicamp & Editora da UEL, Campinas & Londrina, 2003. Traduzido por Luiz Arthur Pagani, Lígia Negri & Rodolfo Ilari.
- [7] H. B. Curry and R. Feys. *Combinatory Logic, I*. North-Holland, Amsterdam, 1958.
- [8] Peter T. Geach. A program for syntax. In Donald Davidson and Gilbert Harman, editors, *Semantics of Natural Language*, pages 483–497. D. Reidel, Dordrecht, 1972.
- [9] Rodolfo Ilari and Maria Helena de Moura Neves, editors. *Gramática do Português Culto Falado no Brasil — Volume 2: Classes de Palavras e Processos de Construção*. Editora UNICAMP, Campinas, 2008.
- [10] Pauline Jacobson. The syntax/semantics interface in categorial grammar. In Shalom Lappin, editor, *The Handbook of Contemporary Semantic Theory*. Blackwell, Oxford, 1996.

- [11] Joachim Lambek. The mathematics of sentence structure. *American Mathematical Monthly*, 65:154–170, 1958.
- [12] John Lyons. *Introdução à Lingüística Teórica*. Editora Nacional & EDUSP, São Paulo, 1979. Traduzido por Rosa Virgínia Mattos e Silva & Hélio Pimentel.
- [13] Carlos Miotto, Maria Cristina Figueiredo Silva, and Ruth Elisabeth Vasconcellos Lopes. *Manual de Sintaxe*. Insular, Florianópolis, 1999.
- [14] Michael Moortgat. *Categorical Investigations: Logical and Linguistic Aspects of the Lambek Calculus*. Foris, Dordrecht, 1988.
- [15] Glyn V. Morrill. *Type Logical Grammar*. Kluwer, Dordrecht, 1994.
- [16] Glyn V. Morrill. *Categorical Grammar -- Logical Syntax, Semantics, and Processing*. Oxford University Press, Oxford, 2011.
- [17] Cezar A. Mortari. *Introdução à Lógica*. Editora UNESP & Imprensa Oficial do Estado, São Paulo, 2001.
- [18] José Borges Neto. Introdução às gramáticas categoriais. 1999.
- [19] Luiz Arthur Pagani. Gramática categorial através de estrutura de características. *Revista Letras*, 60:385–410, 2003.
- [20] Dag Prawitz. *Natural Deduction – A Proof-Theoretical Study*. Almqvist and Wiksell, Uppsala, 1965.
- [21] W. V. O. Quine. Variables explained away. In *Proceedings of the American Philosophical Society*, volume 104, pages 343–347, 1960.
- [22] Bertrand Russell. On denoting. *Mind*, 14:479–493, 1905.
- [23] Mark Steedman. *Surface Structure and Interpretation*. The MIT Press, Cambridge, 1996.
- [24] Henriëtte de Swart. *Introduction to Natural Language Semantics*. CSLI, Stanford, 1998.
- [25] Mary McGee Wood. *Categorical Grammars*. Routledge, London, 1993.