Notas em Teoria dos Tipos

Mateus Galdino

Recife, 2023

Contents

1 Introdução à Teoria dos Tipos **3**
- 1.1 História da Teoria dos Tipos 3
  - 1.1.1 Russel e os problemas da teoria dos conjuntos 3

2 Cálculo Lambda não-tipado ($\lambda_{\beta\eta}$) **4**
- 2.1 Definições 4
- 2.2 Sintáxe do Cálculo Lambda 4
- 2.3 Conversão 6
- 2.4 Substituição 7
- 2.5 Beta redução 7

Esse conjunto de notas reunidas aqui estão relacionadas à Teoria dos Tipos, suas subáreas (Por exemplo, a Teoria dos Tipos Homotópica) e suas áreas irmãs (Teoria das Categorias, etc). Para isso, vou me basear nos seguintes textos:

Para algumas sitações que eu queria explicitar, usarei as seguintes abreviações:
[HoTT] = Homotopy Type Theory: Univalent foundations of mathematics. Voevodsky et al.
[TTFP] = Type Theory and Formal Proof. Geuvers e Nederpelt
[TPL] = Types and Programming Languages. Benjamin Pierce
[IHoTT] = Introduction to Homotopy Type Theory. Egbert Rijke

1 Introdução à Teoria dos Tipos

1.1 História da Teoria dos Tipos

Para entender um pouco do que é a teoria dos tipos, é necessário entender sua história de desenvolvimento. Para isso, é necessário voltar até sua criação.

1.1.1 Russel e os problemas da teoria dos conjuntos

Do final do século XIX até o início do século XX, um dos temas de maior relevância para a matemática era a chamada Teoria dos Conjuntos. A primeira publicação que tratava do conceito de **conjunto** foi o artigo de **Georg Cantor** "Über eine Eigenschaft des Inbegriffes aller reellen algebraischen Zahlen" (Sobre uma propriedade da coleção de todos os números reais algébricos), no contexto de estudos em álgebra. Para Cantor, um conjunto (*Menge* ou *Inbegriff*) seria definido como "qualquer combinação M de certos objetos completamente distintos m em nossa intuição [Anschauung] ou em nosso pensamento (que são conhecidos como "Elementos" de M) como um todo."

Uma das primeiras pessoas a utilizar o conceito desenvolvido por Cantor para formular uma teoria sobre as bases da matemática seria **Gottlob Frege** no seu livro "Grundlagen der Arithmetik" (Fundamentos da aritmética). Nesse livro, Frege tenta explicar filosoficamente as bases da matemática e para isso ele se vale da teoria dos conjuntos, também da definição de número feita por **Giuseppe Peano**.

Um grande leitor das obras de Frege foi o matemático **Bertrand Russel**. Ele percebeu em seus estudos um problema na teoria dos conjuntos que fazia essa teoria, no seu estado naquela época, ser insuficiente para uma construção sólida das bases da matemática. Esse problema seria um paradoxo desenvolvido através das próprias definições de conjunto encontradas até então. De forma básica, o *Paradoxo de Russell* era construido através da seguinte definição. Seja o conjunto R definito por $R = \{x | x \notin x\}$, ou seja, o conjunto de todos os conjuntos que não possuem a si mesmos como elemento. A pergunta que fica é $R \in R$? Se sim, pela própria definição de R, então $R \notin R$. Mas se não, ou seja $R \notin R$, então, novamente pela definição de R, $R \in R$. Logo, chega-se a um paradoxo.

Tendo em vista essa questão, os autores **Bertrand Russel** e **Alfred North Whitehead** decidem desenvolver uma obra para a formalização completa da matemática desde suas bases até áreas mais avançadas, chamada de Principia Mathematica. Nessa obra, Russel coloca novamente a questão desse paradoxo, mas dessa vez ele introduz uma possível solução. Nessa solução, ele divide os conjuntos em tipos que só podem possuir elementos de outros tipos. Isso faz com que esses tipos não deem brecha para possuirem a si mesmos.

Dessa forma, a noção de tipos ainda estava vinculada à noção de conjuntos, mas aparecia para resolver problemas relacionados à, como ficou chamada poteriormente, Teoria Ingênua dos conjuntos.

2 Cálculo Lambda não-tipado ($\lambda_{\beta\eta}$)

Para entender melhor o que motiva a criação da teoria dos tipos, essas notas serão iniciadas com o cálculo λ não tipado.

2.1 Definições

O cálculo lambda serve como uma abstração em cima do conceito de função. Uma função é algo que pega um *input* e retorna um *output*, por exemplo a função $f(x) = x^2$ pega um input x e retorna seu quadrado x^2. No cálculo lambda, essa função é denotada por $\lambda x.x^2$, onde $\lambda x.$ simboliza que essa função espera receber como entrada x. Quando queremos receber uma resposta específica de uma função, usamos números no lugar das variáveis, como por exemplo $f(3) = 3^2 = 9$. No cálculo lambda, isso é feito na forma de $(\lambda x.x^2)(3)$.

Esses dois principrios de construção são definidos como:

- **Abstração**: Seja M uma expressão e x uma variável, podemos construir uma nova expressão $\lambda x.M$. Essa expressão é chamada de Abstração de x sobre M

- **Aplicação**: Sejam M e N duas es expressões, podemos construir uma expressão MN. Essa expressão é chamada de Aplicação de M em N.

Dadas essas operações, precisamos também de uma definição que dê conta do processo de encontrar o resultado após a aplicação em uma função. Esse processo é chamado de β-redução. Ela faz uso da substituição e usa como notação os colchetes.

Definição 2.1 (β-redução)**.** A β-redução é o processo de resscrita de uma expressão da forma $(\lambda x.M)N$ em outra expressão $M[x := N]$, ou seja, a expressão M na qual todo x foi substituido por N.

2.2 Sintáxe do Cálculo Lambda

É interessante definir a sintaxe do cálculo lambda de forma mais formal. Para isso, são utilizados métodos que podem ser familiares para aqueles que já trabalharam com lógica proposicional, lógica de primeira ordem ou teoria de modelos.

Definição 2.2. (i) Os *termos lambda* são palavras em cima do seguinte alfabeto:

- variáveis: $v_0, v_1, \ldots$

- abstrator: λ

- parentesis: $($, $)$

(ii) O conjunto de λ-termos Λ é definido de forma indutiva da seguinte forma:

- Se x é uma variável, então $x \in \Lambda$

- $M \in \Lambda \to (\lambda x.M) \in \Lambda$
- $M, N \in \Lambda \to MN \in \Lambda$

Na teoria dos tipos e no cálculo lambda, é utilizada uma forma concisa de definir esses termos chamada de Formalismo de Backus-Naur ou Forma Normal de Backus (BNF, em inglês). Nessa forma, a definição anterior é reduzida à:

$$\Lambda = V|(\Lambda\Lambda)|(\lambda V \Lambda)$$

Onde V é o conjunto de variáveis $V = \{x, y, z, \dots\}$

Para expressar igualdade entre dois termos de Λ utilizamos o simbolo $\equiv$.

Algumas definições indutivas podem ser formadas a partir da definição dos λ-termos.

Definição 2.3 (Multiconjunto de subtermos)**.**

1. (Base) $Sub(x) = \{x\}$, para todo $x \in V$
2. (Aplicação) $Sub((MN)) = Sub(M) \cup Sub(N) \cup \{(MN)\}$
3. (Abstração) $Sub((\lambda x.M)) = Sub(M) \cup \{(\lambda x.M)\}$

Observações:
(i) Um subtermo pode ocorrer múltiplas vezes, por isso é escolhido chamar de multiconjunto

Lemma 2.1 (Propriedades de Sub)**.**

- *(Reflexividade) Para todo λ-termo M, temos que $M \in Sub(M)$*
- *(Transitividade)Se $L \in Sub(M)$ e $M \in Sub(N)$, então $L \in Sub(N)$.*

Definição 2.4 (Subtermo próprio)**.** L é um subtermo próprio de M se L é subtermo de M e $L \not\equiv M$

Exemplos:

1. Seja o termo $\lambda x.\lambda y.xy$, vamos calcular seus subtermos:

$$\begin{aligned} Sub(\lambda x.\lambda y.xy) &= \{\lambda x.\lambda y.xy\} \cup Sub(\lambda y.xy) \\ &= \{\lambda x.\lambda y.xy\} \cup \{\lambda y.xy\} \cup Sub(xy) \\ &= \{\lambda x.\lambda y.xy\} \cup \{\lambda y.xy\} \cup Sub(x) \cup Sub(y) \\ &= \{\lambda x.\lambda y.xy, \lambda y.xy, x, y\} \end{aligned}$$

2. Seja o termo $(y(\lambda x.(xyz)))$, vamos calcular os seus subtermos:

$$\begin{aligned} Sub(y(\lambda x.(xyz))) &= Sub(y) \cup Sub((\lambda x.(xyz))) \\ &= \{y\} \cup \{(\lambda x.(xyz))\} \cup Sub((xyz)) \\ &= \{y\} \cup \{(\lambda x.(xyz))\} \cup Sub(x) \cup Sub(y) \cup Sub(z) \\ &= \{y\} \cup \{(\lambda x.(xyz))\} \cup \{x\} \cup \{y\} \cup \{z\} = \{y, (\lambda x.(xyz)), x, y, z\} \end{aligned}$$

Outro conjunto importante para a sintaxe do cálculo lambda é o de variáveis livres. Uma variável é dita *ligante* se está do lado do λ. Em um termo $\lambda x.M$, x é uma variável ligante e toda aparição de x em M é chamada de *ligada*. Se existir uma variável em M que não é ligante, então dizemos que ela é livre. Por exemplo, em $\lambda x.xy$, o primeiro x é ligante, o segundo x é ligado e y é livre.

O conjunto de todas as variáveis livres em um termo é denotado por FV e definido da seguinte forma:

Definição 2.5 (Multiconjunto de variáveis livres)**.**

1. (Base) $FV(x) = \{x\}$, para todo $x \in V$
2. (Aplicação) $FV((MN)) = FV(M) \cup FV(N) \cup \{(MN)\}$
3. (Abstração) $FV((\lambda x.M)) = FV(M) \setminus \{x\}$

Exemplos:

1. Seja o termo $\lambda x.\lambda y.xyz$, vamos calcular seus subtermos:

$$\begin{aligned} FV(\lambda x.\lambda y.xyz) &= FV(\lambda y.xyz) \setminus \{x\} \\ &= FV(xyz) \setminus \{y\} \setminus \{x\} \\ &= FV(x) \cup FV(y) \cup FV(z) \setminus \{y\} \setminus \{x\} \\ &= \{x, y, z\} \setminus \{y\} \setminus \{x\} \\ &= \{z\} \end{aligned}$$

2.

Vamos definir os termos fechados da seguinte forma:

Definição 2.6. O λ-termo M é dito *fechado* se $FV(M) = \emptyset$. Um λ-termo fechado também é chamado de *combinador*. O conjunto de todos os λ-termos fechados é chamado de Λ^0.

Os combinadores são muito utilizados na *Lógica Combinatória*, mas vamos explorá-los mais a frente.

2.3 Conversão

No cálculo Lambda, é possível renomear variáveis ligantes/ligadas, pois a mudança dos nomes dessas variáveis não muda na sua renomeação. Por exemplo, $\lambda x.x^2$ e $\lambda u.u^2$ podem ser utilizadas de forma igual, mesmo que com nomes diferentes. Podemos definir então essa renomeação

Definição 2.7. Seja $M^{x \to y}$ o resultado da troca de todas as livre-ocorrencias de x em M por y. A relação de renomeação é expressa pelo símbolo $=_\alpha$ e é definida como: $\lambda x.M =_\alpha \lambda y.M^{x \to y}$, dado que $y \notin FV(M)$ e y não seja ligante em M.

Podemos extender essa definição para a definição do renomeamento, chamado de α-conversão.

Definição 2.8 (α-conversão)**.**

1. (Renomeamento) $\lambda x.M =_\alpha \lambda y.M^{x \to y}$
2. (Compatibilidade) Sejam M, N, L termos. Se $M =_\alpha N$, então $ML =_\alpha NL$, $LM =_\alpha LN$ e, para um z qualquer, $\lambda z.M = \lambda z.N$
3. (Reflexividade) $M =_\alpha M$
4. (Simetria) Se $M =_\alpha N$, então $N =_\alpha M$
5. (Transitividade) Se $L =_\alpha M$ e $M =_\alpha N$, então $L =_\alpha N$

A partir dos pontos (3), (4) e (5) dessa definição, é possível dizer que a α-conversão é uma relação de equivalência, chamada de α-equivalência.

2.4 Substituição

Podemos definir agora a substituição de um termo por outro da seguinte forma:

Definição 2.9 (Substituição)**.**

1. $x[x := N] \equiv N$
2. $y[y := x] \equiv y$, se $x \not\equiv y$
3. $(PQ)[x := N] \equiv (P[x := N])(Q[x := N])$
4. $(\lambda y.P)[x := N] \equiv (\lambda z.P^{y \to z})[x := N]$ se $\lambda z P^{y \to z}$ é α-equivalente a $\lambda y P$ e $z \notin FV(N)$

A notação $[x := N]$ é uma meta-notação, pois não está definida na sintaxe do cálculo lambda.

2.5 Beta redução