UNIVERSIDADE FEDERAL DO CEARÁ
DEPARTAMENTO DE COMPUTAÇÃO
MESTRADO EM CIÊNCIA DA COMPUTAÇÃO

Um Sistema Infinitário para a Lógica de Menor Ponto-Fixo

Alexandre Matos Arruda

Fortaleza, Ceará

24 de Agosto de 2007

Alexandre Matos Arruda

Um Sistema Infinitário para a Lógica de Menor Ponto-Fixo

Dissertação apresentada como exigência parcial para a obtenção do título de mestre, sob orientação da professora Dra. Ana Teresa de Castro Martins.

Orientador(a):
Ana Teresa de Castro Martins

Departamento de Computação
Universidade Federal do Ceará

Fortaleza, Ceará

24 de Agosto de 2007

Dedico este trabalho a Deus,
à minha família
e à Najara.

Agradecimentos

Agradeço a Deus por ter me guiado e mostrado o caminho certo, mesmo que por linhas tortas.

Agradeço à minha família pelo apoio moral dado.

Agradeço à minha amada Najara, pelo carinho e companheirismo.

Agradeço à minha orientadora Profa. Ana Teresa Martins, pela sua orientação, apoio e compreensão.

Agradeço a todos meus amigos do Departamento de Computação da UFC.

Resumo

A noção de menor ponto-fixo de um operador é amplamente aplicada na ciência da computação como, por exemplo, no contexto das linguagens de consulta para bancos de dados relacionais. Algumas extensões da Lógica de Primeira-Ordem (FOL)[1] com operadores de ponto-fixo em estruturas finitas, como a lógica de menor ponto-fixo (LFP)[2], foram propostas para lidar com problemas relacionados à expressividade de FOL. A ***LFP*** captura as classes de complexidade PTIME sobre a classe das estruturas finitas ordenadas. A caracterização descritiva de classes computacionais é uma abordagem central em Teoria do Modelos Finitos (FMT)[3]. O teorema de Trakhtenbrot, considerado o ponto de partida para FMT, estabelece que a validade sobre modelos finitos não é recursivamente enumerável, isto é, a completude falha sobre modelos finitos. Este resultado é baseado na hipótese de que qualquer sistema dedutivo é de natureza finita. Entretanto, nos podemos relaxar tal hipótese como foi feito no escopo da teoria da prova para aritmética. A teoria da prova tem raízes no programa de Hilbert. Consequências teóricas da noção de prova são, por exemplo, relacionadas a teoremas de normalização, consistência, decidibilidade, e resultados de complexidade. A teoria da prova para aritmética também é motivada pelos teoremas de incompletude de Gödel, cujo alvo foi fornecer um exemplo de um princípio matemático verdadeiro e significativo que não é derivável na aritmética de primeira-ordem. Um meio de apresentar esta prova é baseado na definição de um sistema de prova com uma regra infinitária, a $\boldsymbol{\omega}$-rule, que estabiliza a consistência da aritmética de primeira-ordem através de uma perspectiva de teoria da prova. Motivados por esta prova, iremos propor aqui um sistema infinitário de prova para ***LFP*** que nos permitirá investigar propriedades em teoria da prova. Com tal sistema dedutivo infinito, pretendemos apresentar uma teoria da prova para uma lógica tradicionalmente definida no escopo de FMT. Permanece aberto um caminho alternativo de provar resultados já obtidos com FMT e também novos resultados do ponto de vista da teoria da prova. Além disso, iremos propor um procedimento de normalização com restrições para este sistema dedutivo, que pode ser usado em um provador de teoremas para computar consultas em banco de dados relacionais.

Palavras-Chaves: Lógica de Menor Ponto-Fixo, Teoria dos Modelos Finitos, Teoria da Prova, Sistema de Dedução Natural Infinitário.

[1]Em inglês é a abreviação para “First-Order Logic”, por padronização adotaremos abreviaturas em inglês durante toda a dissertação.

[2]Abreviação para “Least Fixed-Point”.

[3]Abreviação para “Finite Model Theory”.

Abstract

The notion of the least fixed-point of an operator is widely applied in computer science as, for instance, in the context of query languages for relational databases. Some extensions of FOL with fixed-point operators on finite structures, as the least fixed-point logic (LFP), were proposed to deal with problem problems related to the expressivity of FOL. LFP captures the complexity class PTIME over the class of finite ordered structures. The descriptive characterization of computational classes is a central issue within finite model theory (FMT). Trakhtenbrot's theorem, considered the starting point of FMT, states that validity over finite models is not recursively enumerable, that is, completeness fails over finite models. This result is based on an underlying assumption that any deductive system is of finite nature. However, we can relax such assumption as done in the scope of proof theory for arithmetic. Proof theory has roots in the Hilbert's programme. Proof theoretical consequences are, for instance, related to normalization theorems, consistency, decidability, and complexity results. The proof theory for arithmetic is also motivated by Gödel incompleteness theorems. It aims to offer an example of a true mathematically meaningful principle not derivable in first-order arithmetic. One way of presenting this proof is based on a definition of a proof system with an infinitary rule, the ω-rule, that establishes the consistency of first-order arithmetic through a proof-theoretical perspective. Motivated by this proof, here we will propose an infinitary proof system for LFP that will allow us to investigate proof theoretical properties. With such infinitary deductive system, we aim to present a proof theory for a logic traditionally defined within the scope of FMT. It opens up an alternative way of proving results already obtained within FMT and also new results through a proof theoretical perspective. Moreover, we will propose a normalization procedure with some restrictions on the rules, such this deductive system can be used in a theorem prover to compute queries on relational databases.
Keywords: Least Fixed-Point Logic, Finite Model Theory, Proof Theory, Infinitary Natural Deduction System.

Sumário

1 Introdução

1.1 Objetivos

A noção de menor ponto-fixo de um certo operador é amplamente aplicada em teorias da ciência da computação como, por exemplo, na formalização de linguagens de programação. A semântica denotacional de funções recursivas é definida como o menor ponto-fixo de uma certa função F sobre um domínio D [20]. Como exemplo, nós podemos mencionar a semântica da função fatorial, fac, definida sobre o conjunto dos números naturais $\mathbb{N}$, como o menor ponto-fixo da função $F : (\mathbb{N} \to \mathbb{N}_\perp) \to (\mathbb{N} \to \mathbb{N}_\perp)$ descrita em notação lambda como $F = \lambda f.\lambda n.\ n\ igual\ a\ zero\ \to\ um\ []\ n\ vezes\ f(n\ menos\ um)$ [18].

Outro exemplo do uso do conceito de menor ponto-fixo em ciência da computação é no contexto de linguagens de consulta para bancos de dados relacionais. Usando o teorema de Fraïsse, nós podemos provar a impossibilidade de expressar a consulta de fechamento transitivo sobre relações finitas em Lógica de Primeira Ordem (FOL)[8]. Diversas extensões da lógica de primeira-ordem com operadores de ponto-fixo em estruturas finitas foram propostas e usadas para definir fechamento transitivo e outras consultas que não podem ser expressas em FOL [14, 5].

Nós particularmente estamos interessados na lógica de menor ponto-fixo (LFP), uma extensão de FOL com um predicado que computa o menor ponto-fixo de um operador F_φ indexado por uma certa fórmula φ. Com tal lógica, podemos definir consultas de fechamento transitivo e aciclicidade, funções recursivas aritméticas, conectividade em grafos, e diversas outras noções que não são expressíveis em FOL. De fato, LFP é extremamente importante no escopo da ciência da computação teórica devido ao teorema de Immerman-Vardi que estabelece que a LFP captura a classe de complexidade $PTIME$ sobre as classes de estruturas finitas com ordenação [16, 12].

O desenvolvimento da teoria dos modelos finito (a teoria dos modelos sobre estruturas finitas) foi fortemente influenciada pelas aplicações computacionais na teoria de banco de dados, complexidades computacionais e linguagens formais. Já que é relacionada com a teoria dos modelos clássica (a teoria que lida com estruturas de qualquer cardinalidade) a FMT tem seus próprios metódos para provar resultados de expressabilidade, já que a compacidade e o teorema de Löwenheim-Skolem, as ferramentas principais da teoria dos modelos clássica, não são suficientes para provar propriedades sobre conjuntos finitos.

Em 1950, Trakhtenbrot provou um teorema que é considerado o ponto de partida da FMT. Este teorema estabelece que a validade sobre conjuntos finitos não é recursivamente enumerável, isto é, a completude falha sobre modelos finitos [1]. Este resultado é baseado na hipótese de que o sistema dedutivo é de natureza finita, isto é, a noção de provas formais é inerentemente finita, recursiva. Tal hipótese é abandonada no escopo da teoria da prova para aritmética com a introdução de regras infinitárias.

A teoria da prova tem raízes no programa de Hilbert. Por um lado, a teoria da prova está interessada na análise estrutural de provas formais e, por outro lado, na análise teórica matemática através de suas provas e interpretação sintática de uma teoria formal em outra. As consequências teóricas da noção de prova são, por exemplo, relacionadas a teoremas de normalização, consistência, decidibilidade, e resultados em complexidade [22]. A teoria da prova para aritmética é também motivada pelo teorema da incompletude de Gödel, que aponta para um exemplo de um princípio matematicamente significativo e verdadeiro, o princípio da indução transfinita, não derivável em aritmética de primeira-ordem. Como consêquência, é provado que a arimética de primeira-ordem é consistente para indução transfinita até ε_0. Tais resultados podem ser vistos em Gentzen [9, 10] e Schütte [19].

Em [19]. Schütte definiu um sistema de prova infinitário com a ω-rule

$$\dfrac{\begin{matrix}\Pi_0 & \Pi_1 & & \Pi_i & \\ \varphi(0) & \varphi(1) & \ldots & \varphi(i) & \ldots\end{matrix}}{\forall x \varphi(x)}\ \omega$$

para estabelecer a consistência da aritmética de primeira-ordem pela perpectiva da teoria da prova. Motivada por esta regra, proporemos um sistema de prova infinitário para LFP que nos permitirá investigar propriedades de provas. Com tal sistema dedutivo infinitário, pretendemos apresentar uma teoria da prova para uma lógica trandicionalmente definida no escopo de FMT. Isto nos abre um caminho alternativo de provar resultados já obtidos com FMT e também novos resultados atráves do ponto de vista da teoria da prova. Além disso, iremos propor um procedimento de normalização com restrições para este sistema dedutivo, que pode ser usado em um provador de teoremas para computar consultas em banco de dados relacionais.

1.2 Apresentação dos Capítulos e Contribuições

Esta dissertação está dividida da seguinte forma:

A simbologia adotada durante esta dissertação será apresentada no capítulo 2. A linguagem, a semântica e as regras de substituição são como as apresentadas em [6], porém diferentemente de [6], adotamos provas como árvores dedutivas. Além disso, relaxamos a regra de igualdade.

No capítulo 3, descreveremos um sistema de prova que nós desenvolvemos nesta dissertação para a lógica clássica de primeira-ordem em modelos finitos (FOL_{fin}) estendendo o sistema de prova da lógica clássica de primeira ordem (FOL) com a regra ($FIN \ \perp$). Tal sistema é equivalente à noção semântica em modelos finitos e servirá de base para a criação de um sistema de prova para a lógica de menor ponto-fixo. Um exemplo de prova será fornecido no final do capítulo.

Descreveremos, no capítulo 4, noções preliminares para o entendimento das lógicas de ponto-fixo. Apresentaremos, em teoria dos conjuntos, as noções de menor ponto-fixo, ponto-fixo inflacionário e ponto-fixo parcial. Em seguida, apresentaremos a lógica de menor ponto-fixo, juntamente com as lógicas de ponto-fixo inflacionárias e parciais. Com exceção de nossa contribuição com os teoremas 4.4 e 4.5, os resultados deste capítulo seguem como [14]. Este capítulo servirá de base para a criação do sistema dedutivo LFP_{fin}, que será apresentado no capítulo seguinte.

Introduziremos, no capítulo 5, o sistema de prova (LFP_{fin}) desenvolvido nesta dissertação e provado correto e completo quanto ao conceito semântico da Lógica de Menor

Ponto-Fixo em Modelos Finitos. Apresentaremos a prova de corretude e a de completude neste mesmo capítulo. Também apresentaremos um exemplo e em seguida demonstraremos que a regra $LFP\ PF$ pode ser eliminada

Apresentaremos, no capítulo 6, como contribuição nossa, uma prova de normalização para FOL + axiomas da igualdade, que servirá como base para a normalização com restrição de LFP_{fin}. Quanto a normalização deste último descreveremos uma versão restrita para conjuntos que possuem uma certa propriedade.

E, finalmente, no capítulo 7 apresentaremos nossas conclusões e trabalhos futuros.

2 A Lógica de Primeira-Ordem (FOL)

2.1 Introdução

Neste capítulo, introduziremos a linguagem que adotaremos para a Lógica de Primeira Ordem (FOL) com a definição de termos e fórmulas. Em seguida apresentaremos a noção semântica com a definição da interpretação de fórmulas em FOL e as regras de substituição. Por fim, apresentaremos as regras de inferência do sistema de dedução natural para FOL. Quanto à linguagem, à semântica e à substituição estas seguirão as propostas por [6], onde a linguagem possui o símbolo de igualdade, as interpretações estendem a noção de estrutura com o mapeamento de variáveis no domínio e as substituições são simultâneas. Porém, diferentemente de [6], que adota um cálculo de seqüentes, iremos expor nossas provas através de árvores de dedução. Além disso relaxaremos uma das regras de igualdade.

2.2 Linguagem

O alfabeto para a linguagem de primeira-ordem que iremos considerar possui os seguintes símbolos:

1. $x_0, x_1, x_2, \ldots$ (variáveis);
2. $\neg, \wedge, \vee, \rightarrow$ (conectivos);
3. $\forall, \exists$ (quantificadores);
4. $\equiv$ (símbolo da igualdade);
5. $), ($ (parênteses);

6. (a) para todo $n \geq 1$ um conjunto (possivelmente vazio) de símbolos relacionais $R_0^n, R_1^n, R_2^n, \ldots$ de aridade n;

   (b) para todo $n \geq 1$ um conjunto (possivelmente vazio) de símbolos funcionais $f_0^n, f_1^n, f_2^n, \ldots$ de aridade n.

   (c) um conjunto (possivelmente vazio) de constantes $c_0, c_1, c_2, \ldots$.

Iremos denotar por S o conjunto de símbolos de 6.

Definição 2.1 *Definimos os S-termos pelas seguintes regras formação:*

1. *Toda variável é um S-termo.*

2. *Toda constante é um S-termo.*

3. *Se $t_1, \ldots, t_n$ são S-termos e f é um símbolo funcional de aridade n em S, então $ft_1 \ldots t_n$ é também um S-termo.*

Utilizaremos como meta-variáveis as letras t, v, y e z para termos (podendo ou não estarem subescritos). Denotaremos o conjunto de S-termos por T^S.

Definição 2.2 *As S-fórmulas são derivadas das seguintes regras:*

1. *Se t_1 e t_2 são S-termos, então $t_1 \equiv t_2$ é uma S-fórmula.*

2. *Se $t_1, \ldots, t_n$ são S-termos e R é um símbolo relacional de aridade n em S, então $Rt_1 \ldots t_n$ é uma S-fórmula.*

3. *Se φ é uma S-fórmula, então $(\neg\varphi)$ é também uma S-fórmula.*

4. *Se φ e ψ são S-fórmulas, então $(\varphi * \psi)$, onde $* \in \{\wedge, \vee, \rightarrow\}$ também são S-fórmulas.*

5. *Se φ é uma S-fórmula e x uma variável, então $\forall x\varphi$ e $\exists x\varphi$ também são S-fórmulas.*

As fórmulas atômicas são obtidas pelas regras 1 e 2. Chamaremos a fórmula $(\neg\varphi)$ de negação de φ, e $(\varphi \wedge \psi)$, $(\varphi \vee \psi)$ e $(\varphi \rightarrow \psi)$ de, respectivamente, a conjunção, disjunção, e implicação de φ e ψ.

Iremos usar, quando necessário, as meta-fórmulas φ, ψ, σ (podendo ou não estarem subescritos ou superescritos) para denotarem fórmulas quaisquer. O conjunto de S-fórmulas será denotado por L^S (ou seja, L^S é a linguagem de primeira-ordem associada com o símbolo S).

Algumas vezes iremos também omitir parênteses quando a ausência deles não apresentar ambigüidade. Por exemplo: em vez de $(\varphi \vee \psi)$ escreveremos $\varphi \vee \psi$.

Quando o conjunto S em questão for sem importância escreveremos "termos" e "fórmulas" em vez de "S-termos" e "S-fórmulas".

Utilizaremos as letras Γ, Δ e Φ (podendo ou não estarem subescritos ou superescritos) para conjuntos de fórmulas.

Definição 2.3 *O conjunto $VL(t)$ de variáveis livres de t é definido como:*

$$\begin{array}{lcl} VL(x) & := & \{x\} \\ VL(c) & := & \emptyset \\ VL(ft_1 \ldots t_n) & := & VL(t_1) \cup \ldots \cup VL(t_n) \end{array}$$

Estendemos o operador VL também para as fórmulas do seguinte modo:

Definição 2.4 *O conjunto $VL(\varphi)$ de variáveis livres de φ é definido como:*

$$\begin{array}{lcl} VL(t_1 \equiv t_2) & := & VL(t_1) \cup VL(t_2) \\ VL(Pt_1 \ldots t_n) & := & VL(t_1) \cup \ldots \cup VL(t_n) \\ VL(\neg\varphi) & := & VL(\varphi) \\ VL((\varphi * \psi)) & := & VL(\varphi) \cup VL(\psi) \ \textit{para} * \in \{\wedge, \vee, \rightarrow\} \\ VL(\forall x \varphi) & := & VL(\varphi) - \{x\} \\ VL(\exists x \varphi) & := & VL(\varphi) - \{x\} \end{array}$$

2.3 Semântica

Adotaremos a semântica apresentada em [6] onde o conceito de estrutura trata dos símbolos em S e o de interpretação a estende para as variáveis.

Definição 2.5 *Uma S-estrutura é um par $\mathfrak{A} = (A, \alpha)$ com as seguintes propriedades:*

(a) A é chamado domínio ou universo de $\mathfrak{A}$ e não pode ser vazio.

(b) α é um mapeamento em S que satifaz os seguinte itens:

1. *para todo símbolo relacional R de aridade n em S, $\alpha(R)$ é uma relação de aridade n em A.*
2. *para todo símbolo funcional f de aridade n em S, $\alpha(f)$ é uma função de aridade n em A.*
3. *para toda constante c em S, $\alpha(c)$ é um elemento de A.*

Obs.: Dada uma S-estrutura $\mathfrak{A} = (A, \alpha)$, abreviaremos $\alpha(R)$ como $R^{\mathfrak{A}}$, $\alpha(f)$ como $f^{\mathfrak{A}}$ e $\alpha(c)$ como $c^{\mathfrak{A}}$.

Definição 2.6 *Um assinalamento em uma S-estrutura $\mathfrak{A}$ é um mapeamento $\beta : \{x_n \mid n \in N\} \to A$ do conjunto de variáveis para o domínio A.*

Definição 2.7 *Uma S-interpretação $\mathfrak{I}$ é um par $(\mathfrak{A}, \beta)$ consistindo de uma S-estrutura $\mathfrak{A}$ e um assinalamento β em $\mathfrak{A}$.*

Da mesma forma que fizemos para termos e fórmulas, quando o conjunto S for sem importância, iremos falar simplesmente "estrutura" ou "interpretação" ao invés de S-estrutura ou S-interpretação.

Se β é um assinalamento em $\mathfrak{A}$, $a \in A$ e x é uma variável, então definimos $\beta\frac{a}{x}$ como um assinalamento em $\mathfrak{A}$ que mapeia x em a e concorda com β em todas as variáveis distintas de x:

$$\beta\frac{a}{x}(y) := \begin{cases} \beta(y) & \text{se } y \neq x \\ a & \text{se } y = x. \end{cases}$$

Para $\mathfrak{I} = (\mathfrak{A}, \beta)$ consideraremos $\mathfrak{I}\frac{a}{x} := (\mathfrak{A}, \beta\frac{a}{x})$.

Definição 2.8 (Interpretação de Termos) *(a) Para uma variável x definimos $\mathfrak{I}(x) := \beta(x)$.*

(b) Para uma constante $c \in S$ definimos $\mathfrak{I}(c) := c^{\mathfrak{A}}$.

(c) Para um símbolo funcional $f \in S$ e termos $t_1, \ldots, t_n$ definimos

$$\mathfrak{I}(ft_1 \ldots t_n) := f^{\mathfrak{A}}(\mathfrak{I}(t_1), \ldots, \mathfrak{I}(t_n)).$$

Definição 2.9 (Relação de Satisfação) *Seja $\mathfrak{I}$ uma interpretação qualquer. Definimos quando uma interpretação $\mathfrak{I}$ satisfaz uma fórmula φ (notação $\mathfrak{I} \models \varphi$) da seguinte forma:*

$$\begin{array}{lll}
\mathfrak{I} \models t_1 \equiv t_2 & sse & \mathfrak{I}(t_1) = \mathfrak{I}(t_2) \\
\mathfrak{I} \models Rt_1 \dots t_n & sse & R^{\mathfrak{A}}\mathfrak{I}(t_1)\dots\mathfrak{I}(t_n) \\
\mathfrak{I} \models \neg\varphi & sse & \text{não } \mathfrak{I} \models \varphi \\
\mathfrak{I} \models (\varphi \wedge \psi) & sse & \mathfrak{I} \models \varphi \text{ e } \mathfrak{I} \models \psi \\
\mathfrak{I} \models (\varphi \vee \psi) & sse & \mathfrak{I} \models \varphi \text{ ou } \mathfrak{I} \models \psi \\
\mathfrak{I} \models (\varphi \rightarrow \psi) & sse & \text{se } \mathfrak{I} \models \varphi \text{ então } \mathfrak{I} \models \psi \\
\mathfrak{I} \models \forall x\varphi & sse & \text{para todo } a \in A,\ \mathfrak{I}\frac{a}{x} \models \varphi \\
\mathfrak{I} \models \exists x\varphi & sse & \text{existe um } a \in A \text{ tal que } \mathfrak{I}\frac{a}{x} \models \varphi\ .
\end{array}$$

Quando $\mathfrak{I} \models \varphi$ diremos também que $\mathfrak{I}$ é modelo de φ ou $\mathfrak{I}$ satisfaz φ ou φ vale em $\mathfrak{I}$.

Dado um conjunto Φ de S-fórmulas, dizemos que $\mathfrak{I}$ é um modelo de Φ ou $\mathfrak{I}$ satisfaz Φ, e escrevemos $\mathfrak{I} \models \Phi$, se $\mathfrak{I} \models \varphi$ para todo $\varphi \in \Phi$.

Definição 2.10 *Dado um conjunto de fórmulas Φ e uma fórmula φ, dizemos que φ é uma consequência de Φ se e somente se toda interpretação que é um modelo de Φ também é um modelo de φ. Escreveremos como $\Phi \models \varphi$ se φ é uma consequência de Φ.*

Podemos (e iremos) generalizar a definição de $\mathfrak{I}\frac{a}{x}$. Dado $x_0, \dots, x_r$ distintos e uma interpretação $\mathfrak{I} = (\mathfrak{A}, \beta)$, e $a_0, \dots, a_r \in A$, definimos $\beta\frac{a_0 \dots a_r}{x_0 \dots x_r}$ como um assinalamento em $\mathfrak{A}$ tal que

$$\beta\frac{a_0 \dots a_r}{x_0 \dots x_r}(y) := \begin{cases} \beta(y) & \text{se } y \neq x_0, \dots, y \neq x_r \\ a_i & \text{se } y = x_i \end{cases}$$

e

$$\mathfrak{I}\frac{a_0 \dots a_r}{x_0 \dots x_r} := \left(\mathfrak{A}, \beta\frac{a_0 \dots a_r}{x_0 \dots x_r}\right).$$

2.4 Substituição

Definiremos agora a substituição de variáveis por termos e, em seguida, o critério de substituição variáveis por termos em fórmulas.

Definição 2.11

$$
\begin{array}{llcl}
(i) & x\frac{t_0 \ldots t_r}{x_0 \ldots x_r} & := & \begin{cases} x & se\ x \neq x_0, \ldots, x \neq x_r \\ t_i & se\ x = x_i \end{cases} \\
(ii) & c\frac{t_0 \ldots t_r}{x_0 \ldots x_r} & := & c \\
(iii) & ft'_0, \ldots, t'_p\frac{t_0 \ldots t_p}{x_0 \ldots x_p} & := & ft'_0\frac{t_0 \ldots t_p}{x_0 \ldots x_p}, \ldots, t'_p\frac{t_0 \ldots t_p}{x_0 \ldots x_p}.
\end{array}
$$

Quanto a substituição em fórmulas, temos:

Definição 2.12 .

1. $t'_1 \equiv t'_2\frac{t_0 \ldots t_r}{x_0 \ldots x_r} := t'_1\frac{t_0 \ldots t_r}{x_0 \ldots x_r} \equiv t'_2\frac{t_0 \ldots t_r}{x_0 \ldots x_r}$

2. $Rt'_1 \ldots t'_n\frac{t_0 \ldots t_r}{x_0 \ldots x_r} := Rt'_1\frac{t_0 \ldots t_r}{x_0 \ldots x_r} \ldots t'_n\frac{t_0 \ldots t_r}{x_0 \ldots x_r}$

3. $(\neg\varphi)\frac{t_0 \ldots t_r}{x_0 \ldots x_r} := (\neg\varphi\frac{t_0 \ldots t_r}{x_0 \ldots x_r})$

4. $(\varphi * \psi)\frac{t_0 \ldots t_r}{x_0 \ldots x_r} := (\varphi\frac{t_0 \ldots t_r}{x_0 \ldots x_r}\square\psi\frac{t_0 \ldots t_r}{x_0 \ldots x_r})$, *para* $* \in \{\wedge, \vee, \rightarrow\}$

5. *Considere as variáveis* $x_{i_1}, \ldots, x_{i_k}$ $(i_1 < \ldots < i_k)$ *sendo exatamente as variáveis dentre* $x_0, \ldots, x_r$, *tal que*
$$x_i \in VL(\exists x\varphi) \text{ e } x_i \neq t_i$$
A substituição neste caso é feita da seguinte forma:
$$[\exists x\varphi]\frac{t_0 \ldots t_r}{x_0 \ldots x_r} := \exists u[\varphi\frac{t_{i_1} \ldots t_{i_k} u}{x_{i_1} \ldots x_{i_k} x}]$$
onde u *é* x *se* $x \notin VL(t_{i_1}) \cup \ldots VL(t_{i_k})$, *ou então* u *é* x_i, *onde* x_i *é a primeira variável que não ocorre em* $\varphi, t_{i_1}, \ldots, t_{i_k}$.

6. *O caso* $\forall x\varphi\frac{t_0 \ldots t_r}{x_0 \ldots x_r}$ *é similar a* (5).

Portanto, a substituição $\varphi\frac{t}{x}$ nos garante que as variáveis de t não serão capturadas pelos quantificadores, renomeando quando necessário as varíaveis ligadas de φ.

Lema 2.1 (Lema da Substituição) 1. *Para todo termo t, temos*
$$\mathfrak{I}\left(t\frac{t_0 \ldots t_r}{x_0 \ldots x_r}\right) = \mathfrak{I}\frac{\mathfrak{I}(t_0) \ldots \mathfrak{I}(t_r)}{x_0 \ldots x_r}(t).$$

2. *Para toda fórmula* φ,
$$\mathfrak{I} \models \varphi\frac{t_0 \ldots t_r}{x_0 \ldots x_r} \text{ sse } \mathfrak{I}\frac{\mathfrak{I}(t_0) \ldots \mathfrak{I}(t_r)}{x_0 \ldots x_r} \models \varphi.$$

Prova: Como em [6].

Lema 2.2 (Lema da Permutação) *Para toda permutação dos numeros* $0,\ldots r$*, temos que*

$$\varphi\frac{t_0\ldots t_r}{x_0\ldots x_r} = \varphi\frac{t_{\pi(0)}\ldots t_{\pi(r)}}{x_{\pi(0)}\ldots x_{\pi}(r)}.$$

Prova: Como em [6].

2.5 Dedução Natural

Nesta dissertação, as deduções (ou derivações) serão apresentadas como árvores, onde todos os nós são fórmulas, sendo as folhas chamadas de hipóteses e a raiz sendo a conclusão de toda a dedução. Cada fórmula na árvore é obtida de um nó antecessor atráves de uma regra de inferência.

Quanto as hipóteses (ou suposições) em uma árvore de dedução, são de dois tipos: as hipóteses fechadas, que são aquelas que foram eliminadas por alguma regra de inferência na derivação; e as hipóteses abertas, que não são fechadas.

As letras gregas Π e Σ (sub ou super-escritos) denotarão derivações e/ou sequências de derivações (incluindo a sequência vazia).

As notações

$$\begin{array}{c} [\psi]^k \\ \Pi \\ \varphi \end{array} \qquad \begin{array}{c} \Pi \\ \varphi \end{array}$$

significam, respectivamente, uma dedução Π com conclusão φ e um conjunto de hipóteses abertas $[\psi]^k$, consistindo de todas as ocorrências de fórmulas ψ que possuem o marcador k; e, uma dedução Π com conclusão φ.

As derivações são apresentadas como a seguir:
Base: Uma árvore com um único nó rotulado com φ é uma derivação com a hipótese φ aberta.

Passo Indutivo: Considere Π_1, Π_2, Π_3 derivações. Podemos construir uma derivação Π de acordo com uma das regras abaixo. As fórmulas entre colchetes $[\varphi]^k, [\neg\varphi]^k, [\varphi_1]^{k_1}, [\varphi_2]^{k_2}$

logo a seguir contém hipóteses abertas que são fechadas após a aplicações das regras que as eliminam (Note que escrevemos o número da hipótese ao lado da regra que a fechou).

As regras de inferência são:

$$\frac{\begin{matrix}\Pi_1 & \Pi_2 \\ \varphi_1 & \varphi_2\end{matrix}}{\varphi_1 \wedge \varphi_2}\ (\wedge I) \qquad \frac{\begin{matrix}\Pi_1 \quad \Pi_2 \\ \varphi_1 \wedge \varphi_2\end{matrix}}{\varphi_i}\ (\wedge E), \text{ para } i \in \{1,2\}$$

$$\frac{\begin{matrix}\Pi_i \\ \varphi_i\end{matrix}}{\varphi_1 \vee \varphi_2}\ (\vee I), \text{ para } i \in \{1,2\} \qquad \frac{\begin{matrix} & [\varphi_1]^{k_1} & [\varphi_2]^{k_2} \\ \Pi_1 & \Pi_2 & \Pi_3 \\ \varphi_1 \vee \varphi_2 & \psi & \psi\end{matrix}}{\psi}\ (\vee E), k_1, k_2$$

$$\frac{\begin{matrix}[\varphi]^k \\ \Pi_1 \\ \psi\end{matrix}}{\varphi \to \psi}\ (\to I), k \qquad \frac{\begin{matrix}\Pi_1 & \Pi_2 \\ \varphi \to \psi & \varphi\end{matrix}}{\psi}\ (\to E)$$

$$\frac{\begin{matrix}[\varphi]^k \\ \Pi_1 \\ \bot\end{matrix}}{\neg\varphi}\ (\neg I), k \qquad \frac{\begin{matrix}\Pi_1 & \Pi_2 \\ \neg\varphi & \varphi\end{matrix}}{\bot}\ (\neg E)$$

$$\frac{\begin{matrix}\Pi_1 \\ \bot\end{matrix}}{\varphi}\ (\bot_i) \qquad \frac{\begin{matrix}[\neg\varphi]^k \\ \Pi_1 \\ \bot\end{matrix}}{\varphi}\ (\bot_c), k$$

$$\frac{\begin{matrix}\Pi_1 \\ \varphi(y)\end{matrix}}{\forall x \varphi\frac{x}{y}}\ (\forall I) \qquad \frac{\begin{matrix}\Pi_1 \\ \forall x \varphi\end{matrix}}{\varphi\frac{t}{x}}\ (\forall E)$$

$$\frac{\begin{matrix}\Pi_1 \\ \varphi\frac{t}{x}\end{matrix}}{\exists x \varphi}\ (\exists I) \qquad \frac{\begin{matrix} & [\varphi\frac{y}{x}]^k \\ & \Pi_1 \\ \exists x\varphi & \psi\end{matrix}}{\psi}\ (\exists E), k$$

Restrições: para a regra $\forall I$ temos que a variável y não pode ocorrer nas hipóteses abertas que $\varphi(y)$ depende; quanto a regra $\exists E$ temos que y não pode ocorrer em $\exists x\varphi$, nem em ψ ou em qualquer hipótese que ψ dependa a menos de $\varphi\frac{y}{x}$.

Temos também as regras da igualdade:

$$\frac{}{t \equiv t}\,(EQ1) \qquad \frac{\begin{array}{cc}\Pi_1 & \Pi_2 \\ \varphi(z)\frac{t}{z} & t \equiv t'\end{array}}{\varphi(z)\frac{t'}{z}}\,(EQ2)$$

Restrições: a variável z ocorre livre em φ ($z \in VL(\varphi)$). Note que com essa restrição, necessariamente temos que $\varphi(z)\frac{t}{z}$ deve ter a forma diferente de $\varphi(z)\frac{t'}{z}$ se t e t' possuírem formas diferentes.

Em uma regra de eliminação, a premissa contendo a ocorrência de um conectivo lógico a ser eliminado é chamada de premissa maior. As outras premissas são chamadas de premissas menores da aplicação da regra.

Podemos (e iremos) relaxar a regra $EQ2$, utilizando o lema 2.2 (Lema da Permutação), da seguinte forma:

$$\frac{\begin{array}{cc}\Pi_1 & \Pi_2 \\ \varphi(z)\frac{tt_0t_1\ldots t_n}{zv_0v_1\ldots v_n} & t \equiv t'\end{array}}{\varphi(z)\frac{t't_0t_1\ldots t_n}{zv_0v_1\ldots v_n}}\,(EQ2)$$

Já que, pelo lema da permutação temos que:

$$\varphi(z)\frac{tt_0t_1\ldots t_n}{zv_0v_1\ldots v_n} = \varphi(z)\frac{t_0t_1\ldots t_nt}{v_0v_1\ldots v_nz}$$

e

$$\varphi(z)\frac{t't_0t_1\ldots t_n}{zv_0v_1\ldots v_n} = \varphi(z)\frac{t_0t_1\ldots t_nt'}{v_0v_1\ldots v_nz}.$$

Chamaremos a primeira premissa ocorrendo da esquerda para a direita na regra $EQ2$ de premissa primeira de uma aplicação de uma regra $EQ2$.

Definição 2.13 (Relação de Consequência Sintática) *A relação $\Gamma \vdash \varphi$ denota uma derivação de φ a partir de um conjunto Γ de hipóteses.*

Usaremos também as seguintes convenções sobre árvores de prova:

1. $\dfrac{\Pi_1 \quad \Pi_2 \quad \ldots \quad \Pi_n}{\varphi}$ denota uma derivação, cujas subderivações imediatas são $\Pi_1, \Pi_2, \ldots, \Pi_n$, e φ é a fórmula final.

2. $r(\Pi)$ denota a última regra de inferência em Π.

3. $\begin{matrix}\Gamma\\ \Pi\end{matrix}$ denota uma derivação Π que tem Γ como o conjunto de suas top-fórmulas.

4. $\begin{matrix}\Sigma\\ \overline{\Gamma}\\ \Pi\end{matrix}$ denota a árvore obtida, escrevendo Σ acima de cada tóp-formula de Π pertencendo à Γ.

5. Σ_x^t denota o resultado de substituir a variável x pelo termo t em todas as ocorrências de x em Σ.

3 A Lógica de Primeira-Ordem em Modelos Finitos (FOL_{fin})

3.1 Introdução

Um sistema de dedução é considerado finitário quando o conjunto de seus teoremas é recursivamente enumerável, ou seja, existe um algoritmo que lista todos os teoremas derivados neste sistema. Caso contrário, o sistema dedutivo é considerado infinitário. Por exemplo, o sistema de dedução natural apresentado no capítulo anterior é finitário. Em particular, as regras de inferência deste sistema são finitárias no sentido de que são definidas a partir de um conjunto finito de fórmulas (ver [6] e [5]).

A Lógica de Primeira-Ordem em Modelos Finitos (abreviaremos aqui como FOL_{fin}) será apresentada através de um sistema de dedução infinitário correto e completo em relação aos modelos finitos da FOL. O teorema de Trakhtenbrot, citado no capítulo 1, nos afirma que nenhum sistema de dedução finitário para FOL pode recursivamente enumerar o conjunto de verdades finitamente válidas (fórmulas que são válidas em modelos finitos). Isto implica que qualquer sistema de dedução finitário para FOL não é completo com relação aos modelos finitos de FOL. Nesta dissertação contribuímos com um cálculo que introduziremos neste capítulo que não é limitado por esse teorema devido ao fato de ser um sistema infinitário de prova. Nosso intuito em apresentá-lo, é que sua análise nos permitirá estendê-lo para a lógica de menor ponto-fixo restrita a modelos finitos.

Com exceção das definições 3.1 e 3.2, o resto das definições, lemas e teoremas apresentados no capítulo segue como contribuição nossa.

3.2 Dedução Natural

Antes de mostrar o cálculo iremos abordar aqui algumas definições e noções preliminares. Vamos começar pelas definições de finitamente satisfatível e finitamente válido.

Definição 3.1 *Dizemos que um conjunto* Γ *de fórmulas é finitamente satisfatível quando existe um modelo finito que o satisfaz.*

Definição 3.2 *Dizemos que um conjunto* Γ *de fórmulas é finitamente válido quando todo modelo finito o satisfaz.*

A seguir introduziremos fórmulas do tipo $\lambda_{\geq i}$ que falam da cardinalidade dos domínios de seus respectivos modelos.

A sentença $\lambda_{\geq 2} := \exists v_0 \exists v_1 (\neg v_0 \equiv v_1)$ é uma formalização de "Há pelo menos dois elementos". Mais precisamente, para todo vocabulário S e todas S-interpretações $\mathfrak{I} = (A, \mathfrak{A})$,

$$\mathfrak{I} \models \lambda_{\geq 2} \text{ sse } A \text{ contém pelo menos dois elementos.}$$

Podemos generalizar este tipo de fórmula para um número n qualquer de elementos:

$$\lambda_{\geq n} := \exists v_0 \ldots \exists v_{n-1} (\neg v_0 \equiv v_1 \wedge \ldots \wedge \neg v_0 \equiv v_{n-1} \wedge \ldots \wedge \neg v_{n-2} \equiv v_{n-1})$$

estabelece que há pelo menos n elementos, enquanto que as sentenças $\neg\lambda_{\geq n}$ e $\lambda_{\geq n} \wedge \neg\lambda_{\geq n+1}$ expressam que há menos que n elementos e exatamente n elementos, respectivamente. Agora considere o seguinte conjunto:

$$\Phi_\infty := \{\lambda_{\geq n} \mid n \geq 2\}.$$

Os modelos de Φ_∞ são precisamente as interpretações infinitas. Isto é, para todo S e todas S-interpretações $\mathfrak{I}$,

$$\mathfrak{I} \models \Phi_\infty \text{ sse } A \text{ contém um número infinito de elementos.}$$

O cálculo FOL_{fin} estende o cálculo FOL com a adição da seguinte regra:

$$\dfrac{\begin{array}{c}\Pi_1 \\ \exists v_1 \exists v_2 (\neg v_1 \equiv v_2)\end{array} \quad \begin{array}{c}\Pi_2 \\ \exists z_1 \exists z_2 \exists z_3 (\neg z_1 \equiv z_2 \wedge \neg z_1 \equiv z_3 \wedge \neg z_2 \equiv z_3)\end{array} \quad \ldots}{\perp} \; (FIN \perp)$$

ou

$$\dfrac{\begin{array}{c}\Pi_1 \\ \lambda_{\geq 2}\end{array} \quad \begin{array}{c}\Pi_2 \\ \lambda_{\geq 3}\end{array} \quad \ldots \quad \begin{array}{c}\Pi_{i-1} \\ \lambda_{\geq i}\end{array} \quad \ldots}{\perp} \; (FIN \perp).$$

O significado intuitivo desta regra é dado em relação à noção semântica em modelos finitos: se cada $\lambda_{\geq n}$ é consequência lógica de um conjunto de fórmulas Γ, então Γ não é finitamente satisfatível. Portanto, Γ deriva o absurdo.

Agora definiremos mais formalmente a noção de satisfação em modelos finitos, a de consequência lógica e a noção de prova:

Definição 3.3 $\Gamma \models_{FOL_{fin}} \varphi$ *se toda interpretação finita que satisfaz* Γ *satisfaz* φ.

Definição 3.4 $\Gamma \vdash_{FOL_{fin}} \varphi$ *se existe uma derivação de* φ *em* FOL_{fin} *utilizando como hipóteses fórmulas de* Γ.

Antes de provarmos a corretude e completude quanto à semântica da teoria dos modelos finitos ($\Gamma \models_{FOL_{fin}} \varphi$ sse $\Gamma \vdash_{FOL_{fin}} \varphi$) precisamos do seguinte teorema:

Lema 3.1 $\Gamma \models_{FOL_{fin}} \varphi$ *sse* $\Gamma \cup \{\neg\varphi\}$ *não é finitamente satisfatível.*

Prova: $\Gamma \models_{FOL_{fin}} \varphi$

sse toda interpretação finita que satisfaz Γ satisfaz φ

sse não existe interpretação finita que satisfaz Γ e não satifaz φ

sse não existe interpretação finita que satisfaz Γ e $\neg\varphi$

sse $\Gamma \cup \{\neg\varphi\}$ não é finitamente satisfatível. $\square$

3.3 Corretude

Iremos agora provar que o sistema FOL_{fin} é correto ($\Gamma \vdash_{FOL_{fin}} \varphi \Rightarrow \Gamma \models_{FOL_{fin}} \varphi$):

Teorema 3.1 (Corretude) *Para $\Gamma \in L^S$ e $\varphi \in L^S$ temos:*

$$\textit{Se } \Gamma \vdash_{FOL_{fin}} \varphi \textit{ então } \Gamma \models_{FOL_{fin}} \varphi.$$

Prova: Para as regras idênticas à da FOL a corretude de FOL_{fin} é a mesma já que:

$$\Gamma \vdash \varphi \Rightarrow \Gamma \models \varphi \Rightarrow \Gamma \models_{FOL_{fin}} \varphi.$$

Resta, portanto, provarmos a corretude para a regra $(FIN\ \bot)$. Por indução temos:

$$\Gamma \models_{FOL_{fin}} \lambda_{\geq 2},\ \Gamma \models_{FOL_{fin}} \lambda_{\geq 3}\ \ldots\ .$$

Seja $\mathfrak{I}$ uma interpretação finita tal que $\mathfrak{I} \models_{FOL_{fin}} \Gamma$. Logo, pela hipótese, temos que $\mathfrak{I} \models_{FOL_{fin}} \lambda_{\geq 2}$, $\mathfrak{I} \models_{FOL_{fin}} \lambda_{\geq 3}$... e o conjunto $\{\lambda_{\geq 1}, \lambda_{\geq 2}, ...\}$ é satisfeito por $\mathfrak{I}$, portanto $\mathfrak{I}$ é uma interpretação infinita, absurdo. Concluímos que Γ não é finitamente satisfatível e portanto $\Gamma \models_{FOL_{fin}} \bot$. $\square$

3.4 Completude

Nesta seção iremos provar a implicação inversa da que foi apresentada na corretude, ou seja, a completude ($\Gamma \models_{FOL_{fin}} \varphi \Rightarrow \Gamma \vdash_{FOL_{fin}} \varphi$):

Teorema 3.2 (Completude) *Para $\Gamma \in L^S$ e $\varphi \in L^S$ temos:*

$$\textit{Se } \Gamma \models_{FOL_{fin}} \varphi \textit{ então } \Gamma \vdash_{FOL_{fin}} \varphi.$$

Prova: Dado $\Gamma \models_{FOL_{fin}} \varphi$ temos que provar $\Gamma \vdash_{FOL_{fin}} \varphi$. Assuma $\Gamma \models_{FOL_{fin}} \varphi$ e $\Gamma \not\vdash_{FOL_{fin}} \varphi$, logo $\Gamma \not\vdash \varphi$ e portanto o conjunto $\Gamma \cup \{\neg\varphi\}$ é consistente. Assim temos que $\Gamma \cup \{\neg\varphi\}$ é satisfatível e, pela hipótese, não é finitamente satisfatível (Lema 3.1). Concluímos que somente interpretações infinitas satisfazem $\Gamma \cup \{\neg\varphi\}$. Dessa forma temos:

$\Gamma \cup \{\neg\varphi\} \models \lambda_{\geq i}$

$\Rightarrow \Gamma \cup \{\neg\varphi\} \vdash \lambda_{\geq i}$

$\Rightarrow \Gamma \cup \{\neg\varphi\} \vdash_{FOL_{fin}} \lambda_{\geq i}$

$\Rightarrow \Gamma \cup \{\neg\varphi\} \vdash_{FOL_{fin}} \bot$

$\Rightarrow \Gamma \vdash_{FOL_{fin}} \varphi$. $\square$

3.5 Exemplo de Prova em FOL_{fin}

Este cálculo nos permite a prova de sentenças que são finitamente válidas. Por exemplo, a sentença $(\forall x \forall y(fx \equiv fy \to x \equiv y) \to \forall x \exists y(x \equiv fy))$ diz que se f é uma função injetiva, então ela é sobrejetiva. Esta fórmula só é válida sobre modelos finitos.

Como a sentença $(\forall x \forall y(fx \equiv fy \to x \equiv y) \to \forall x \exists y(x \equiv fy))$ é valida em todas as interpretações finitas, iremos provar em FOL_{fin} que

$$\vdash_{FOL_{fin}} (\forall x \forall y(fx \equiv fy \to x \equiv y) \to \forall x \exists y(x \equiv fy)).$$

Podemos esquematizar da seguinte forma:

$$\dfrac{\dfrac{\begin{matrix}\Pi_1 & \Pi_2 & \\ \varphi_{\geq 2} & \varphi_{\geq 3} & \cdots\end{matrix}}{\bot}\,(FIN\,\bot)}{(\forall x \forall y(fx \equiv fy \to x \equiv y) \to \forall x \exists y(x \equiv fy))}\,(\bot_c), 0$$

onde $\Pi_1 =$

$$\dfrac{\dfrac{\dfrac{\dfrac{\neg(\forall x \forall y(fx \equiv fy \to x \equiv y) \to \forall x \exists y(x \equiv fy))^0}{\forall x \forall y(fx \equiv fy \to x \equiv y) \wedge \neg\forall x \exists y(x \equiv fy)}}{\neg\forall x \exists y(x \equiv fy)}}{\exists x \forall y \neg(x \equiv fy)} \qquad \dfrac{\dfrac{\dfrac{\forall y \neg(z \equiv fy)^1}{\neg(z \equiv fz)}}{\exists v_1 \neg(z \equiv v_1)}}{\exists v_0 \exists v_1 \neg(v_0 \equiv v_1)}}{\exists v_0 \exists v_1 \neg(v_0 \equiv v_1)}$$

e $\Pi_2 =$

$$
\dfrac{
\dfrac{
\dfrac{
\dfrac{\neg(\forall x\forall y(fx \equiv fy \to x \equiv y) \to \forall x\exists y(x \equiv fy))^0}{\forall x\forall y(fx \equiv fy \to x \equiv y) \wedge \neg\forall x\exists y(x \equiv fy)}
}{\neg\forall x\exists y(x \equiv fy)}
}{\exists x\forall y\neg(x \equiv fy)}
\qquad
\dfrac{
\dfrac{
\dfrac{
\dfrac{
\dfrac{\forall y\neg(z \equiv fy)}{\neg(z \equiv fz)} \quad \Sigma
}{\neg(z \equiv fz) \wedge \neg(z \equiv ffz) \wedge \neg(fz \equiv ffz)}
}{\exists v_2(\neg(z \equiv fz) \wedge \neg(z \equiv v_2) \wedge \neg(fz \equiv v_2))}
}{\exists v_1\exists v_2(\neg(z \equiv v_1) \wedge \neg(z \equiv v_2) \wedge \neg(v_1 \equiv v_2))}
}{\exists v_0\exists v_1\exists v_2(\neg v_0 \equiv v_1 \wedge \neg v_0 \equiv v_2 \wedge \neg v_1 \equiv v_2)}
}{\exists v_0\exists v_1\exists v_2(\neg v_0 \equiv v_1 \wedge \neg v_0 \equiv v_2 \wedge \neg v_1 \equiv v_2)}
$$

onde $\Sigma =$

$$
\dfrac{
\dfrac{\forall y\neg(z \equiv fy)}{\neg(z \equiv ffz)}
\qquad
\dfrac{
\dfrac{
\dfrac{\forall y\neg(z \equiv fy)}{\neg(z \equiv fz)}
\quad
\dfrac{fz \equiv ffz \qquad
\dfrac{
\dfrac{
\dfrac{
\dfrac{\neg(\forall x\forall y(fx \equiv fy \to x \equiv y) \to \forall x\exists y(x \equiv fy))^0}{\forall x\forall y(fx \equiv fy \to x \equiv y) \wedge \neg\forall x\exists y(x \equiv fy)}
}{\forall x\forall y(fx \equiv fy \to x \equiv y)}
}{\forall y(fz \equiv fy \to z \equiv y)}
}{fz \equiv ffz \to z \equiv fz}
}{(z \equiv fz)}
}{\perp}
}{\neg(fz \equiv ffz)}
}{\neg(z \equiv ffz) \wedge \neg(fz \equiv ffz)}
$$

e $\Pi_i = \ldots.$

Obs.: A notação $\overset{\Sigma}{\overline{\overline{\Pi}}}$ acima, unicamente neste caso, é uma forma de dizer que são teoremas provados em FOL. Correspondendo a prova de $\neg(\varphi_1 \to \varphi_2) \vdash (\varphi_1 \wedge \neg\varphi_2)$ e $\neg\forall x\exists y\varphi_1 \vdash \exists x\forall y\neg\varphi_1$ quando é o caso.

Vemos que, para um termo z acima, podemos obter $\neg(z \equiv ff^{[n]}z)$ para qualquer $n \in N$. Da mesma forma obtemos $\neg(ff^{[n]}z \equiv ff^{[m]}z)$ onde $m, n \in N$ e $m \neq n$. Portanto podemos definir uma conjunção de qualquer tamanho onde $(\neg(z \equiv fz) \wedge \neg(z \equiv ffz) \wedge \ldots \wedge \neg(fz \equiv ffz) \wedge \ldots \wedge \neg(f^{[n-1]}z \equiv f^{[n]}z))$ sendo necessário apenas aplicarmos a regra $(\exists I)$ $n+1$ vezes para obtermos $\lambda_{\geq n}$. Logo há uma prova para cada $\lambda_{\geq n}$.

4 Lógicas de Ponto-Fixo

4.1 Introdução

Neste capítulo, abordaremos o conceito de operadores de ponto-fixo que é adequado para a definição de muitas propriedades que não podem ser expressas em lógica de primeira-ordem, como conectividade e alcançabilidade. A partir desta noção apresentaremos uma semântica para a lógica de menor ponto-fixo, que estende a lógica de primeira ordem com a inclusão destes operadores.

Na seção 4.2 descreveremos a noção de operadores de menor ponto-fixo via teoria dos conjuntos (particularmente em conjuntos finitos) como descrita em [14]. Como foge ao escopo desta dissertação, assumiremos que o leitor esteja familiarizado com alguns conceitos em teoria dos conjuntos que podem ser encontrados em [11]. Abordaremos também outros operadores de ponto-fixo, são estes: operadores inflacionários e parciais. Eles diferem dos operadores de menor ponto-fixo por não serem necessariamente monotónos.

Apresentaremos na seção 4.3 uma semântica para a lógica de menor ponto-fixo que estende a semântica da lógica de primeira-ordem clássica com a inclusão de fórmulas que representam o menor ponto-fixo de operadores monotónos (como apresentado em [14]). Temos, como única contribuição nossa neste capítulo, os teoremas 4.4 e 4.5 que servirão de base para a elaboração de um sistema dedutivo no próximo capítulo.

4.2 Operadores de Ponto-Fixo

A teoria dos operadores de ponto-fixo é apresentada para reticulados completos, isto é, conjuntos parcialmente ordenados $(A, \preccurlyeq)$ onde todo subconjunto de A tem um ínfimo e um supremo na ordem $\preccurlyeq$. Lembramos que nesta dissertação nosso foco principal são as interpretações com domínio finito, o que estabelece que os teoremas aqui apresentados são provados para conjuntos finitos. Para maiores detalhes, consulte [14].

Primeiramente vamos estabelecer o que é um operador:

Definição 4.1 (Operador) *Dado um conjunto A, considere $\wp(A)$ seu conjunto das partes. Um operador F em A é um mapeamento $F : \wp(A) \to \wp(A)$.*

A partir disto, podemos definir quando um operador é monotóno ou inflacionário:

Definição 4.2 (Operador Monotóno e Inflacionário) *Dizemos que um operador F é monótono se*

$$X \subseteq Y \text{ implica } F(X) \subseteq F(Y),$$

e inflacionário se

$$X \subseteq F(X)$$

para todo $X \in \wp(A)$.

A partir deste conceito, agora podemos definir o menor ponto-fixo de um operador monotóno:

Definição 4.3 (Menor Ponto-Fixo de um Operador) *Dado $F : \wp(A) \to \wp(A)$, um conjunto $X \subseteq A$ é um ponto-fixo de F se $F(X) = X$. Um conjunto $X \subseteq A$ é o menor ponto-fixo de F se ele é um ponto-fixo e, para todo outro ponto-fixo Y de F, temos $X \subseteq Y$. Iremos denotar o menor ponto-fixo de um operador F como $\boldsymbol{lfp}(F)$.*

Definição 4.4 (Operador Indutivo) *Chamamos um operador F de indutivo se a sequência*

$$X^0 = \emptyset,\ X^{i+1} = F(X^i). \tag{4.1}$$

é crescente, ou seja, $X^i \subseteq X^{i+1}$ para todo i.

Teorema 4.1 *Todo operador monótono F é indutivo.*

Prova (como em [14]): Por indução temos $X^0 \subseteq X^1$ já que $X^0 = \emptyset$. Se $X^i \subseteq X^{i+1}$ então, pela monotonicidade, $F(X^i) \subseteq F(X^{i+1})$, isto é, $X^{i+1} \subseteq X^{i+2}$. Isto mostra que $X^i \subseteq X^{i+1}$ para todo $i \in N$. □

Definição 4.5 *Se F é indutivo, definimos*

$$X^\infty = \bigcup_{i=0}^{\infty} X^i. \tag{4.2}$$

A sequência (4.2) estabiliza (pela finitude do domínio A) depois de uma quantidade finita de passos. Portanto existe um número n tal que $X^\infty = X^n$. Como exemplo:

Dada uma relação binária E de arestas em um grafo, podemos obter relações $E^0, E^1, \ldots$, tal que se $(x,y) \in E^i$ então existe um caminho de tamanho no máximo i de x para y. O predicado binário "caminho" deste grafo pode ser expresso como a união de todas essas relações:

$$E^\infty = \bigcup_{0}^{\infty} E^i.$$

Como estamos considerando apenas domínios finitos, E é finito. Para algum ponto n da sequência $E^0, E^1, \ldots$ temos $E^n = E^{n+1} = E^{n+2} = \ldots$. Logo $E^\infty = E^n$ e portanto E^n é o limite da sequência e podemos considerá-lo um ponto-fixo, já que para $i \geq n$ implica que $E^i = E^{i+1}$.

Agora provaremos que todo operador monótono tem um menor ponto-fixo, que é o conjunto X^∞ definido como a união da sequência crescente de (4.2).

Teorema 4.2 (Tarski-Knaster, [14]) *Todo operador monótono $F : \wp(A) \rightarrow \wp(A)$ tem um menor ponto-fixo $\boldsymbol{lfp}(F)$ que pode ser definido como*

$$\boldsymbol{lfp}(F) = \bigcap \{Y \mid Y = F(Y)\}.$$

Além disso, $\boldsymbol{lfp}(F) = X^\infty = \bigcup_i X^i$, para a sequência X^i definida por (4.2).

Pela grande importância deste teorema, em vez de apenas citá-lo vamos expor a prova apresentada em [14]:

Prova: Dado $W = \{Y \mid F(Y) \subseteq Y\}$. Claramente $W \neq \emptyset$, já que $A \subseteq W$. Primeiramente mostraremos que $S = \bigcap W$ é um ponto-fixo de F. Além disso, para qualquer $Y \in W$, temos $S \subseteq Y$ e portanto $F(S) \subseteq F(Y) \subseteq Y$. Logo $F(S) \subseteq \bigcap W = S$. Inversamente já que $F(S) \subseteq S$, temos $F(F(S)) \subseteq F(S)$, e dessa forma $F(S) \in W$. Portanto $S = \bigcap W \subseteq F(S)$, provando que $S = F(S)$.

Dado $W' = \{Y \mid F(Y) = Y\}$ e $S' = \bigcap W'$, então $S \in W'$ e $S' \subseteq S$. Inversamente, $W' \subseteq W$, então $S = \bigcap W \subseteq \bigcap W' = S'$. Portanto, $S = S'$. Consequentemente, $S = \bigcap\{Y \mid Y = F(Y)\}$ é um ponto-fixo de F. Já que é a interseção de todos os pontos fixos de F, S é o menor ponto-fixo de F. Isto mostra que

$$\mathbf{lfp}(F) \;=\; \bigcap\{Y \mid Y = F(Y)\} \;=\; \bigcap\{Y \mid F(Y) \subseteq Y\}.$$

Para provar que $\mathbf{lfp}(F) = X^\infty$, note que a sequência X^i aumenta e, portanto, para algum $n \in N$, $X^n = X^{n+1} = \ldots = X^\infty$. Logo, $F(X^\infty) = X^\infty$ e X^∞ é um ponto-fixo. Para mostrar que é o menor ponto-fixo, é suficiente provar que $X^i \subseteq Y$ para todo i e para todo $Y \in W$. Provamos por indução em i. Claramente $X^0 \subseteq Y$ para todo $Y \in W$. Precisamos provar a afirmação para X^{i+1}. Dado $Y \in W$, temos $X^{i+1} = F(X^i)$. Pela hipótese, $X^i \subseteq Y$. Isto mostra que todo X^i's estão contidos em todos os conjuntos de W, o que completa a prova do teorema. □

Apresentaremos agora dois tipos de operadores não-monotónos, são eles: operadores inflacionários e operadores parciais.

Dado que F é inflacionário, a sequência (4.1) é crescente, e portanto alcança um ponto-fixo X^∞. Suponha agora um operador G qualquer, podemos associar a G um operador inflacionário G_{infl} dado por:

Definição 4.6 (Ponto-Fixo Inflacionário de um Operador) *Chamamos o ponto-fixo inflacionário de um operador G como o X^∞ da sequeência 4.2 do operador*

$$G_{infl}(Y) = Y \cup G(Y).$$

e será denotado por $\boldsymbol{ifp}(G)$.

Portanto, $\mathbf{ifp}(G)$ é a união de todos os conjuntos X^i's onde $X^0 = \emptyset$ e

$X^{i+1} = X^i \cup G(X^i)$.

Finalmente, dado um operador F e a sequência (4.1), temos a opção de esta sequência não precisar ser indutiva. Neste caso ou a sequência atinge um ponto-fixo (para algum $n \in N$ ocorre $X^n = X^{n+1}$), e teremos necessariamente que $n \leq 2^{|A|}$, caso contrário tal n não existe. Com base nisto, o ponto-fixo parcial pode ser definido como:

Definição 4.7 (Ponto-Fixo Parcial de um Operador) *Definimos o ponto-fixo parcial de um operador F como*

$$\boldsymbol{pfp}(F) = \begin{cases} X^n & \text{, se } X^n = X^{n+1} \\ \emptyset & \text{, se } X^n \neq X^{n+1} \text{ para todo } n \leq 2^{|A|}. \end{cases}$$

Como relação desses três operadores (menor ponto-fixo, inflacionário e parcial), temos o seguinte teorema:

Teorema 4.3 *Se F é monótono então $\boldsymbol{lfp}(F) = \boldsymbol{ifp}(F) = \boldsymbol{pfp}(F)$.* $\square$

A partir deste ponto em diante, por todo o texto da dissertação, lidaremos unicamente com vocabulários relacionais, ou seja, interpretações sem símbolos funcionais. Portanto, variáveis e constantes serão os únicos termos da linguagem. Lembramos novamente ao leitor que toda interpretação a que iremos nos referir possui domínio finito.

4.3 A Lógica de Menor Ponto-Fixo

Agora adicionaremos operadores de ponto-fixo para a lógica de primeira-ordem. Suponha que tenhamos um vocabulário relacional S, e um símbolo relacional adicional $R \notin S$ de aridade k. Considere $\varphi(R, x_1, \ldots, x_k)$ uma fórmula do vocabulário $S \cup \{R\}$. Colocamos o símbolo R explicitamente como parâmetro, já que esta fórmula irá dar origem a um operador sobre S-interpretações.

Para cada S-interpretação $\mathfrak{I}$, a fórmula $\varphi(R, \vec{x})$ dá origem a um operador $F_\varphi : \wp(A^k) \rightarrow \wp(A^k)$ definido como:

$$F_\varphi(X) = \{(a_1, \ldots a_k) \mid \mathfrak{I}\frac{a_1 \ldots a_k}{x_1 \ldots x_k} \models \varphi(X/R, x_1, \ldots, x_k)\}. \tag{4.3}$$

Aqui a notação $\varphi(X/R,\vec{x})$ significa que R é interpretado como X em φ. Mais precisamente, se $\Im$' é uma $S \cup \{R\}$-interpretação expandindo $\Im$, onde R é interpretado como X, então $\Im' \frac{a_1 \ldots a_k}{x_1 \ldots x_k} \models \varphi(x_1, \ldots, x_k)$.

A idéia das lógicas de ponto-fixo é adicionarmos fórmulas para computar pontos-fixos dos operadores F_φ. Isto nos dá uma definição formal das lógicas IFP e PFP.

Definição 4.8 *As lógicas IFP e PFP são definidas como extensões da lógica de primeira-ordem com as seguintes regras de formação:*

- *(Para IFP): se $\varphi(R,\vec{x})$ é uma fórmula, onde R é de aridade k, e $\vec{t}$ é uma tupla de termos, onde $|\vec{x}| = |\vec{t}| = k$, então*

  $$[\boldsymbol{ifp}_{R,\vec{x}}\varphi(R,\vec{x})](\vec{t})$$

  é uma fórmula cujas variáveis livres são aquelas de $\vec{t}$.

- *(Para PFP): se φ é uma fórmula, onde R é de aridade k, e $\vec{t}$ é uma tupla de termos, onde $|\vec{x}| = |\vec{t}| = k$, então*

  $$[\boldsymbol{pfp}_{R,\vec{t}}\varphi(R,\vec{x})](\vec{t})$$

  é uma fórmula cujas variáveis livres são aquelas de $\vec{t}$.

A semântica é definida como:

- *(Para IFP): $\Im \models [\boldsymbol{ifp}_{R,\vec{x}}\varphi(R,\vec{x})](\vec{t})$ sse $(\Im(t_1)\ldots\Im(t_k)) \in \boldsymbol{ifp}(F_\varphi)$.*

- *(Para PFP): $\Im \models [\boldsymbol{pfp}_{R,\vec{x}}\varphi(R,\vec{x})](\vec{t})$ sse $(\Im(t_1)\ldots\Im(t_k)) \in \boldsymbol{pfp}(F_\varphi)$.*

Não podemos definir analogamente uma extensão com o conceito do menor ponto-fixo porque menores pontos-fixos são garantidos existirem apenas para operadores monótonos. Entretanto, o seguinte teorema nos diz que essa propriedade não é fácil de se obter:

Lema 4.1 *Testar se F_φ é monótono é indecidível para fórmulas de primeira-ordem φ.*

Prova: Uma prova pode ser encontrada em [14]. □

Portanto, é necessário algumas restrições sintáticas para garantirmos que menores pontos-fixos são tomados apenas para operadores monótonos. Para este fim, iremos definir o conceito de ocorrência negativa de um símbolo relacional R em uma fórmula φ: Dada uma fórmula φ que contém um símbolo relacional R, dizemos que uma ocorrência de R é negativa se está sob o escopo de um número ímpar de negações.

Por exemplo, na fórmula $\exists x \neg R(x) \vee \neg \forall y \forall z \neg (R(y) \vee \neg R(z))$, a primeira ocorrência de R é negativa, a segunda é positiva (pois está no escopo de um número par de negações) e a terceira é negativa novamente. Dizemos que uma fórmula é positiva em R se não há ocorrências negativas de R nela, ou seja, se todas as ocorrências de R são positivas, ou não existem.

Definição 4.9 *A lógica LFP estende a lógica de primeira-ordem com a seguinte regra de formação:*

- *se $\varphi(R,\vec{x})$ é uma fórmula positiva em R, onde R é de aridade k e $\vec{t}$ é uma tupla de termos, onde $|\vec{x}| = |\vec{t}| = k$, então*

$$[\boldsymbol{lfp}_{R,\vec{x}}\varphi(R,\vec{x})](\vec{t})$$

é uma fórmula, cujas variáveis livres são aquelas de $\vec{t}$.

A semântica é definida como:

$$\mathfrak{I} \models [\boldsymbol{lfp}_{R,\vec{x}}\varphi(R,\vec{x})](\vec{t}) \text{ sse } (\mathfrak{I}(t_1)\ldots\mathfrak{I}(t_k)) \in \boldsymbol{lfp}(F_\varphi).$$

Ainda resta ser provado:

Lema 4.2 *Se $\varphi(R,\vec{x})$ é positivo em R, então F_φ é monótono.* □

Prova: Como em [14].

Agora suponha que temos uma fórmula $\varphi(R,\vec{x})$. Assuma que φ é positivo em R. Para construir o menor ponto-fixo de φ sobre uma interpretação $\mathfrak{I}$, indutivamente calculamos

$X^0 = \emptyset$, $X^{i+1} = F_\varphi(X^i)$, e então o ponto-fixo é dado por $X^\infty = \bigcup_i X^i$. Iremos referir X^i's como estágios da computação do ponto-fixo, com X^i sendo o i-ésimo estágio.

Primeiramente, notamos que cada estágio é definível por uma fórmula LFP, se φ é positivo em R. Isto é, para cada estágio i, temos uma fórmula $\varphi^i(\vec{x_i})$, tal que $\varphi^i(\mathfrak{I})$[1] é exatamente X^i. Estas são definidas indutivamente como:

Definição 4.10 (Sequência de Fórmulas $\varphi^i(\vec{x_i})$)

$$\begin{array}{ll} \varphi^0(\vec{x_0}) \equiv \neg(x = x) & , \textit{ onde } x \textit{ é uma variável em } \vec{x_0} \\ \varphi^{i+1}(\vec{x_{i+1}}) \equiv \varphi(\varphi^i/R, \vec{x_{i+1}}) & . \end{array}$$

Aqui a notação $\varphi(\varphi^i/R, \vec{x_{i+1}})$ significa que toda ocorrência de $R(\vec{y})$ em φ é substituída por $\varphi^i(\vec{y})$ e, portanto, todas as variáveis ligadas em φ são substituídas por novas. Por exemplo, considere a fórmula $\varphi(R,x,y) \equiv E(x,y) \vee \exists z(E(x,z) \wedge R(z,y))$. Seguindo a formação anterior obtemos as fórmulas:

$$\begin{array}{lcl} \varphi^0(x_0,y_0) & \equiv & \neg(x_0 = x_0) \\ \varphi^1(x_1,y_1) & \equiv & E(x_1,y_1) \vee \exists z_1(E(x_1,z_1) \wedge \varphi^0(z_1,y_1)) \\ & & \leftrightarrow E(x_1,y_1) \\ \varphi^2(x_2,y_2) & \equiv & E(x_2,y_2) \vee \exists z_2(E(x_2,z_2) \wedge \varphi^1(z_2,y_2)) \\ & & \leftrightarrow E(x_2,y_2) \vee \exists z_2(E(x_2,z_2) \wedge E(z_2,y_2)) \\ \ldots & & \ldots \end{array}$$

computando os estágios do operador de fechamento transitivo.

A seguir apresentaremos a parte com nossa contribuição no capítulo (definição de substituição e teoremas 4.4 e 4.5):

Vamos estender a substituição para o cálculo LFP_{fin} da seguinte forma:

Definição 4.11 (Substituição em LFP)

$$[\boldsymbol{lfp}_{R,\vec{x}}\ \varphi(R,\vec{x})](t_1,\ldots,t_n)\frac{t'_1\ldots t'_k}{x_1\ldots x_k} = [\boldsymbol{lfp}_{R,\vec{x}}\ \varphi(R,\vec{x})](t_1\frac{t'_1\ldots t'_k}{x_1\ldots x_k},\ldots,t_n\frac{t'_1\ldots t'_k}{x_1\ldots x_k})$$

Mesmo estendendo nossa noção de substituição, é fácil provar que o Lema da Substituição

[1] $\varphi(\mathfrak{I}) = \{(a_1,\ldots,a_m) \in A^m \mid \mathfrak{I}\frac{a_1\ldots a_m}{x_1\ldots x_m} \models \varphi(x_1,\ldots,x_m)\}$ onde A é o domínio de i.

(Lema 2.1) continua válido para LFP_{fin}.

Definimos também $\varphi^i(\vec{t}) = \varphi^i(x_1,\ldots,x_k)\frac{t_1\ldots t_k}{x_1\ldots x_k}$, para $\vec{t} = (t_1,\ldots,t_k)$.

Teorema 4.4 *Dada as fórmulas* $\varphi^i(\vec{t})$ *e* $[\boldsymbol{lfp}_{R,\vec{x}}\ \varphi(R,\vec{x})](\vec{t})$ *onde* $|\vec{x}| = |\vec{t}| = k$. *Considere uma interpretação* $\mathfrak{I}$ *tal que* $\mathfrak{I} \models \varphi^i(\vec{t})$, *então temos* $\mathfrak{I} \models [\boldsymbol{lfp}_{R,\vec{x}}\ \varphi(R,\vec{x})](\vec{t})$.

Prova: Como $\varphi^i(\mathfrak{I})$ define exatamente X^i e temos $X^i \subseteq X^\infty$, onde $\mathbf{lfp}(F_\varphi) = X^\infty$, logo $(\mathfrak{I}(t_1),\ldots,\mathfrak{I}(t_k)) \in X^i$ e portanto $\mathfrak{I} \models [\mathbf{lfp}_{R,\vec{x}}\ \varphi(R,\vec{x})](\vec{t})$. $\square$

Teorema 4.5 *Dada as fórmulas* $\varphi^{i^{|\vec{t}|}}(\vec{t})$ *e* $[\boldsymbol{lfp}_{R,\vec{x}}\ \varphi(R,\vec{x})](\vec{t})$ *onde* $|\vec{x}| = |\vec{t}| = k$. *Considere uma interpretação* $\mathfrak{I}$ *com domínio* $|A| < i+1$, *para algum* $i \in N$, *tal que* $\mathfrak{I} \models [\boldsymbol{lfp}_{R,\vec{t}}\ \varphi(R,\vec{x})](\vec{t})$, *então temos* $\mathfrak{I} \models \varphi^{i^{|\vec{t}|}}(\vec{t})$.

Prova: Note que $\mathbf{lfp}(F_\varphi) = X^{i^{|\vec{t}|}}$, já que a sequência de X^n's é crescente e $\mathfrak{I}$ possui domínio $|A| < i+1$. Como visto antes, a fórmula $\varphi^n(\vec{x_n})$ define o estágio X^n da computação do ponto-fixo. Portanto concluímos que $\mathfrak{I} \models \varphi^{i^{|\vec{t}|}}(\vec{t})$. $\square$

Para o próximo capítulo, iremos utilizar o símbolo $\models_{LFP_{fin}}$ para a nova semântica aqui apresentada em modelos finitos.

5 *A Lógica de Menor Ponto-Fixo em Modelos Finitos* (LFP_{fin})

5.1 Introdução

Neste capítulo, na seção 5.2, apresentaremos o sistema de dedução natural LFP_{fin} para a lógica de menor ponto-fixo em modelos finitos que é equivalente à semântica apresentada no capítulo 4. Na seção seguinte apresentaremos sua corretude. Em seguida, na seção 5.4, abordaremos definições e teoremas sobre a consistência deste cálculo. Apresentaremos, na seção 5.5, algumas definições e conceitos preliminares para finalmente, na seção 5.6, expor a prova da completude do cálculo introduzido. Também apresentaremos na seção 5.7 um exemplo de uma prova em LFP_{fin} e finalmente, na seção 5.8 apresentaremos a eliminação da regra ($LFP\ PF$).

Como contribuição nossa, podemos listar: as seções 5.2 (Dedução Natural), 5.3 (Corretude), o teorema 5.3 na seção 5.4 (já que a maioria das provas dessa seção segue como em FOL), a seção 5.5 com exceção da subseção 5.5.1 (Teorema de Henkin, já que as idéias seguem como as propostas por Henkin), a seção 5.6(Completude), a seção 5.7 e finalmente a eliminabilidade da regra ($LFP\ PF$) na seção 5.8.

A seção 5.4 possui definições e teoremas sobre consistência para LFP_{fin} que são similares às encontradas para FOL (ver [6]), com exceção do teorema 5.3. Da mesma forma o teorema de Henkin em 5.5.1 (ver [6]).

5.2 Dedução Natural

Iniciaremos apresentando o sistema LFP_{fin}.

Obtemos LFP_{fin} estendendo o sistema dedutivo FOL_{fin} adicionando as seguintes regras de inferência:

$$\frac{\begin{array}{c}\Pi\\ [\mathbf{lfp}_{R,\vec{x}}\ \varphi(R,\vec{x})](\vec{t})\end{array} \quad \begin{array}{c}\varphi^0(\vec{t})^0\\ \Pi_0\\ \sigma\end{array} \quad \begin{array}{c}\varphi^1(\vec{t})^1\\ \Pi_1\\ \sigma\end{array} \quad \ldots \quad \begin{array}{c}\varphi^i(\vec{t})^i\\ \Pi_i\\ \sigma\end{array} \quad \ldots}{\sigma}\ (LFP\ E), 0, 1, \ldots, i, \ldots,$$

com o descarte das hipóteses $0, 1, \ldots, i, \ldots$.

Com esta regra, torna-se evidente que as fórmulas da forma $[\mathbf{lfp}_{R,\vec{x}}\ \varphi(R,\vec{x})](\vec{t})$ podem ser vistas como disjunções infinitas das fórmulas que definem os estágios X^i's de seus operadores.

Também adicionamos

$$\frac{\varphi^i(\vec{t})}{[\mathbf{lfp}_{R,\vec{x}}\ \varphi(R,\vec{x})](\vec{t})}\ (LFP\ I), \text{ onde } i \geq 0 \text{ para } i \in N,$$

que intuitivamente nos diz que se provarmos uma das fórmulas $\varphi^i(\vec{t})$ então temos o ponto-fixo para o vetor de termos $\vec{t}$.

Além destas regras, também temos:

$$\frac{[\mathbf{lfp}_{R,\vec{x}}\ \varphi(R,\vec{x})](\vec{t}) \quad \neg\lambda_{\geq i+1}}{\varphi^{i^{|\vec{t}|}}(\vec{t})}\ (LFP\ FP), \text{ onde } i \geq 0 \text{ para } i \in N,$$

A idéia é que uma vez que, tenhamos alguma informação sobre a cardinalidade de um conjunto dada por $\neg\lambda_{\geq i+1}$, então saberemos como calcular seu menor ponto-fixo.

Definição 5.1 $\Gamma \vdash_{LFP_{fin}} \varphi$ *se existe uma derivação* Π *no sistema de dedução natural para* LFP_{fin} *com hipóteses não descartadas em* Γ *e conclusão* φ*.*

5.3 Corretude

Iremos agora provar que o cálculo LFP_{fin} é correto:

Teorema 5.1 (Corretude) *Se* $\Gamma \vdash_{LFP_{fin}} \varphi$ *então* $\Gamma \models_{LFP_{fin}} \varphi$.

Prova: Basta nos restringirmos somente às regras adicionadas, já que as outras regras ($(FIN\ \perp)$ e regras da lógica de primeira-ordem clássicas) já são provadas corretas.

1. $(LFP\ E)$: Suponha , por indução , que $\Gamma' \models_{LFP_{fin}} [\mathbf{lfp}_{R,\vec{x}}\ \varphi(R,\vec{x})](\vec{t})$ e $\Gamma_0, \varphi^0(\vec{t}) \models_{LFP_{fin}} \sigma$ e $\Gamma_1, \varphi^1(\vec{t}) \models_{LFP_{fin}} \sigma \ldots$, onde $\Gamma = \Gamma' \cup \Gamma_0 \cup \Gamma_1 \ldots$. Seja $\mathfrak{I}$ uma interpretação tal que $\mathfrak{I} \models_{LFP_{fin}} \Gamma$, então como $\Gamma' \subseteq \Gamma$, $\mathfrak{I} \models_{LFP_{fin}} [\mathbf{lfp}_{R,\vec{x}}\ \varphi(R,\vec{x})](\vec{t})$ e logo, pelo teorema 4.5, temos que para algum $i \in N$, $\mathfrak{I} \models_{LFP_{fin}} \varphi^i(\vec{t})$. Como $\mathfrak{I} \models_{LFP_{fin}} \Gamma_i$ e, usando a hipótese de indução, concluímos $\Gamma \models_{LFP_{fin}} \sigma$.

2. $(LFP\ I)$: Suponha que $\Gamma \models_{LFP_{fin}} \varphi^i(\vec{t})$. Dada uma interpretação $\mathfrak{I}$ tal que $\mathfrak{I} \models_{LFP_{fin}} \Gamma$, logo $\mathfrak{I} \models_{LFP_{fin}} \varphi^i(\vec{t})$ e, pelo teorema 4.4, $\mathfrak{I} \models_{LFP_{fin}} [\mathbf{lfp}_{R,\vec{x}}\ \varphi(R,\vec{x})](\vec{t})$.

3. $(LFP\ FP)$: Por indução, seja $\Gamma_0 \models_{LFP_{fin}} [\mathbf{lfp}_{R,\vec{x}}\ \varphi(R,\vec{x})](\vec{t})$ e $\Gamma_1 \models_{LFP_{fin}} \neg\lambda_{\geq i+1}$, onde $\Gamma = \Gamma_0 \cup \Gamma_1$. Dada uma interpretação $\mathfrak{I}$ tal que $\mathfrak{I} \models_{LFP_{fin}} \Gamma$, logo temos $\mathfrak{I} \models_{LFP_{fin}} [\mathbf{lfp}_{R,\vec{x}}\ \varphi(R,\vec{x})](\vec{t})$ e $\mathfrak{I} \models_{LFP_{fin}} \neg\lambda_{\geq i+1}$. A última afirmativa nos indica que $\mathfrak{I}$ possui um domínio $|A| < i+1$ e portanto, utilizando o teorema 4.5, temos $\mathfrak{I} \models_{LFP_{fin}} \varphi^{i^{|\vec{t}|}}(\vec{t})$.
$\square$

5.4 Sobre a Noção de Consistência em LFP_{fin}

Apresentaremos agora teoremas sobre a consistência do sistema dedutivo LFP_{fin}, para que possamos utilizá-los nas provas das seções 5.5 e 5.6. As definições e teoremas são apresentados similarmente à [6].

Definição 5.2 *(a)* Φ *é consistente (escreveremos como: Con* Φ*) se e somente se não existe fórmula* φ *tal que* $\Phi \vdash_{LFP_{fin}} \varphi$ *e* $\Phi \vdash_{LFP_{fin}} \neg\varphi$.

(b) Φ *é inconsistente (escreveremos como: Inc* Φ*) se e somente se* Φ *não é consistente, isto é, se existe uma fórmula* φ *tal que* $\Phi \vdash_{LFP_{fin}} \varphi$ *e* $\Phi \vdash_{LFP_{fin}} \neg\varphi$.

A partir da última definição, obtemos:

Teorema 5.2 *Para um conjunto Φ de fórmulas em LFP_{fin}, os itens seguintes são equivalentes:*

(a) Inc Φ.

(b) Para todo φ, $\Phi \vdash_{LFP_{fin}} \varphi$.

Prova: (a) segue imediatamente de (b). Para provar o inverso, suponha que Inc Φ é válido, isto é, $\Phi \vdash_{LFP_{fin}} \psi$ e $\Phi \vdash_{LFP_{fin}} \neg\psi$ para alguma fórmula ψ. Seja φ uma fórmula arbitrária. Vamos provar que $\Phi \vdash_{LFP_{fin}} \varphi$.
Sabemos que existem dois conjuntos Γ_1 e Γ_2, que consistem de fórmulas de Φ, e derivações

$$\begin{array}{ccc} \Gamma_1 & & \Gamma_2 \\ \Pi_1 & & \Pi_2 \\ \psi & \text{e} & \neg\psi. \end{array}$$

Podemos então obter a seguinte derivação:

$$\dfrac{\dfrac{\begin{array}{cc} \Gamma_2 & \Gamma_1 \\ \Pi_2 & \Pi_1 \\ \neg\psi & \psi \end{array}}{\bot}}{\varphi}$$

para qualquer φ, o que conclui nossa prova. □

Corolário 5.1 *Dado um conjunto Φ qualquer de fórmulas em LFP_{fin} os seguintes itens são equivalentes:*

(a) Con Φ.

(b) Existe uma fórmula φ que não é derivável de Φ. □

Teorema 5.3 *Todo conjunto de fórmulas em LFP_{fin} finitamente satisfátivel é consistente.*

Prova: Suponha Inc Φ. Então para um φ adequado temos ambos $\Phi \vdash_{LFP_{fin}} \varphi$ e $\Phi \vdash_{LFP_{fin}}$ $\neg\varphi$. Portanto, pelo teorema da corretude, $\Phi \models_{LFP_{fin}} \varphi$ e $\Phi \models_{LFP_{fin}} \neg\varphi$, que implica que Φ não pode ser finitamente satisfatível. □

Teorema 5.4 *Para todo Φ e φ em LFP_{fin} as seguintes afirmativas são válidas:*

(a) $\Phi \vdash_{LFP_{fin}} \varphi$ *sse Inc* $\Phi \cup \{\neg\varphi\}$ *.*

(b) $\Phi \vdash_{LFP_{fin}} \neg\varphi$ *sse Inc* $\Phi \cup \{\varphi\}$*.*

(c) Se Con Φ*, então Con* $\Phi \cup \{\varphi\}$ *ou Con* $\Phi \cup \{\neg\varphi\}$*.*

Prova:

(a) Se $\Phi \vdash_{LFP_{fin}} \varphi$ então $\Phi \cup \{\neg\varphi\} \vdash_{LFP_{fin}} \varphi$. Já que $\Phi \cup \{\neg\varphi\} \vdash_{LFP_{fin}} \neg\varphi$, então $\Phi \cup \{\neg\varphi\}$ é inconsistente. Para a prova do inverso, considere $\Phi \cup \{\neg\varphi\}$ inconsistente, então para um conjunto Γ adequado consistindo de fórmulas de Φ há uma derivação $\Gamma \cup \{\neg\varphi\} \vdash_{LFP_{fin}} \varphi$. A partir disto obtemos a seguinte derivação:

$$\dfrac{\dfrac{\begin{matrix}\Gamma \cup [\neg\varphi]^{n_2} \\ \Pi_2 \\ \neg\varphi\end{matrix} \qquad \begin{matrix}\Gamma \cup [\neg\varphi]^{n_1} \\ \Pi_1 \\ \varphi\end{matrix}}{\perp}\ (\perp I), n_1, n_2}{\varphi}\ (\perp_c), \text{eliminando fórmulas da forma } [\neg\varphi].$$

Portanto $\Phi \vdash_{LFP_{fin}} \varphi$.

(b) Similar à prova de (a) porém invertendo as regras de φ e $\neg\varphi$.

(c) Se nem Con $\Phi \cup \{\varphi\}$ nem Con $\Phi \cup \{\neg\varphi\}$, isto é, se Inc $\Phi \cup \{\varphi\}$ e Inc $\Phi \cup \{\neg\varphi\}$, então (por (b) e (a)) $\Phi \vdash_{LFP_{fin}} \neg\varphi$ e $\Phi \vdash_{LFP_{fin}} \varphi$. Portanto Φ é inconsistente, uma contradição com a hipótese Con Φ. $\square$

5.5 Definições e Lemas Preliminares para a prova da Completude

Antes de provarmos o teorema da completude precisaremos das seguintes definições:

Definição 5.3 *O* $rk(\varphi)$ *de uma fórmula* φ *é definido como o menor ordinal* α *tal que:*

- $rk(\varphi) = \alpha = 0$*, se* φ *for atômico;*
- $rk(\neg\varphi) = \alpha$ *tal que* $rk(\varphi) = \alpha_1$ *e* $\alpha = \alpha_1 + 1$*;*
- $rk(\varphi * \psi) = \alpha$*, onde* $* \in \{\rightarrow, \wedge, \vee\}$*, tal que* $rk(\varphi) = \alpha_1$*,* $rk(\psi) = \alpha_2$ *e* $\alpha = max\{\alpha_1, \alpha_2\} + 1$*; onde* $max\{\alpha_1, \alpha_2\}$ *é uma função que retorna o maior elemento do conjunto* $\{\alpha_1, \alpha_2\}$*.*

- $rk(\exists x\varphi) = \alpha$ *tal que* $rk(\varphi) = \alpha_1$ *e* $\alpha = \alpha_1 + 1$*;*

- $rk(\forall x\varphi) = \alpha$ *tal que* $rk(\varphi) = \alpha_1$ *e* $\alpha = \alpha_1 + 1$*;*

- $rk([\mathbf{lfp}_{R,\vec{x}}\ \varphi(R,\vec{x})](\vec{t})) = \alpha$ *tal que* $rk(\varphi^i(\vec{t})) = \alpha_i$ *e* $\alpha = sup\{\alpha_0, \alpha_1, \ldots\}\alpha$*; onde* sup *é uma função que retorna o supremo do conjunto* $\{\alpha_0, \alpha_1, \ldots\}$.

Lema 5.1 $rk(\varphi\frac{t_0 \ldots t_r}{x_0 \ldots x_r}) = rk(\varphi)$. $\square$

Lema 5.2 *O conjunto* L^S *de fórmulas de um vocabulário contável e relacional* S *da lógica* LFP_{fin} *é contável.*

Prova: Basta notar que cada fórmula de L^S é finita e, pelas regras de formação apresentadas, existe um procedimento que decide se uma fórmula $\varphi \in L^S$. Portanto L^S é recursivamente enumerável e logo L^S é contável. $\square$

Definição 5.4 *O tamanho de uma árvore de prova* T *(iremos denominar por* $|T|$*) é definido como o menor ordinal* α *associado à uma derivação da seguinte forma:*

- *Se* φ *é uma hipótese ou um axioma então* $|\varphi| = 1$

- *Se* $\Pi = \dfrac{\Pi_0 \quad \Pi_1 \quad \ldots \quad \Pi_k}{\varphi}$ *tal que* $|\Pi_i| = \alpha_i$ *então* $|\Pi| = \alpha$ *onde* $\alpha = max\{\alpha_1, \ldots \alpha_k\} + 1$.

- *Se* $\Pi = \dfrac{\Pi_0 \quad \Pi_1 \quad \ldots \quad \Pi_k \quad \ldots}{\varphi}$ *tal que* $|\Pi_i| = \alpha_i$ *então* $|\Pi| = \alpha$ *onde* sup *é uma função que retorna o supremo do conjunto* $\{\alpha_0, \alpha_1, \ldots\}$.

5.5.1 Teorema de Henkin

Nesta seção abordaremos o teorema de Henkin que, na literatura clássica, nos diz que a partir de um conjunto consistente e completo para a negação e que contém testemunhas (maximal consistente, na literatura), podemos obter uma interpretação para este mesmo conjunto (ver [6]).

Esta seção segue similar ao teorema de Henkin proposto para FOL em [6], com a diferença fundamental do teorema 5.5 (teorema fundamental para nosso propósito) e das regras de inferência em questão.

Dado um conjunto consistente Φ de fórmulas, devemos achar uma interpretação $\mathfrak{I} = (\mathfrak{A}, \beta)$ satisfazendo Φ usando somente elementos sintáticos. Para este objetivo, definiremos uma interpretação $\mathfrak{I}^{\Phi} = (\mathfrak{T}^{\Phi}, \beta^{\Phi})$, mas antes introduziremos a relação binária $\backsim$ sobre o conjunto T^S de S-termos dado por:

Definição 5.5 *$t_1 \backsim t_2$ sse $\Phi \vdash_{LFP_{fin}} t_1 \equiv t_2$.*

Lema 5.3 *(a) $\backsim$ é uma relação de equivalência.*
(b) é compatível com os símbolos em S no seguinte sentido:
Se $t_1 \backsim t_1^{'}, \ldots, t_n \backsim t_n^{'}$ então para um $R \in S$ de aridade n

$$\Phi \vdash_{LFP_{fin}} Rt_1 \ldots t_n \text{ sse } \Phi \vdash_{LFP_{fin}} Rt_1^{'} \ldots t_n^{'}.$$

Prova: Utilizando os axiomas de igualdade. □

Considere $\bar{t}$ a classe de equivalência de t:

$$\bar{t} := \{t^{'} \in T^S \mid t \backsim t^{'}\},$$

e considere T^{Φ} o conjunto das classes de equivalências:

$$T^{\Phi} := \{\bar{t} \mid t \in T^S\}.$$

Definimos, portanto, a S-estrutura $\mathfrak{T}^{\Phi}$ sobre T^{Φ}, que na literatura é chamada de estrutura de termos correspondente a Φ, do seguinte modo:

Definição 5.6 *Para $R \in S$ de aridade n,*

$$R^{\mathfrak{T}^{\Phi}} \bar{t}_1 \ldots \bar{t}_n \text{ sse } \Phi \vdash_{LFP_{fin}} Rt_1 \ldots t_n.$$

E quanto as constantes temos:

Definição 5.7 *Para $c \in S$, $c^{\mathfrak{T}^{\Phi}} := \bar{c}$.*

E variáveis:

Definição 5.8 *$\beta^{\Phi}(x) := \bar{x}$.*

Chamaremos $\mathfrak{I}^{\Phi} := (\mathfrak{T}^{\Phi}, \beta^{\Phi})$ a interpretação de termos associadas com Φ.

Lema 5.4 *(a) Para todo* t, $\mathfrak{I}^{\Phi}(t) = \bar{t}$.
(b) Para toda fórmula atômica φ,

$$\mathfrak{I}^{\Phi} \models_{LFP_{fin}} \varphi \text{ sse } \Phi \vdash_{LFP_{fin}} \varphi.$$

(c) Para toda fórmula φ *e* $x_1, \ldots, x_n$ *variáveis distintas entre si,*

1. $\mathfrak{I}^{\Phi} \models_{LFP_{fin}} \exists x_1 \ldots \exists x_n \varphi$ *sse existem* $t_1, \ldots, t_n \in T^S$ *com* $\mathfrak{I}^{\Phi} \models_{LFP_{fin}} \varphi \frac{t_1 \ldots t_n}{x_1 \ldots x_n}$.
2. $\mathfrak{I}^{\Phi} \models_{LFP_{fin}} \forall x_1 \ldots \forall x_n \varphi$ *sse para todo* $t_1, \ldots, t_n \in T^S$ *com* $\mathfrak{I}^{\Phi} \models_{LFP_{fin}} \varphi \frac{t_1 \ldots t_n}{x_1 \ldots x_n}$.

Prova: (a) Vale para $t = x$ e para $t = c$ por (5.8 e 5.7).
(b) $\mathfrak{I}^{\Phi} \models_{LFP_{fin}} t_1 \equiv t_2$ *sse* $\mathfrak{I}^{\Phi}(t_1) = \mathfrak{I}^{\Phi}(t_2)$

sse $\bar{t_1} = \bar{t_2}$ (por (a))

sse $t_1 \backsim t_2$

sse $\Phi \vdash_{LFP_{fin}} t_1 \equiv t_2$.

$\mathfrak{I}^{\Phi} \models_{LFP_{fin}} Rt_1 \ldots t_n$ *sse* $R^{\mathfrak{T}^{\Phi}} \bar{t_1} \ldots \bar{t_n}$

sse $\Phi \vdash_{LFP_{fin}} Rt_1 \ldots t_n$ (por 5.6).

(c) (1) $\mathfrak{I}^{\Phi} \models_{LFP_{fin}} \exists x_1 \ldots \exists x_n \varphi$

sse existem $a_1, \ldots, a_n \in T^{\Phi}$ com $\mathfrak{I}^{\Phi} \frac{a_1 \ldots a_n}{x_1 \ldots x_n} \models_{LFP_{fin}} \varphi$

sse existem $t_1, \ldots, t_n \in T^S$ com $\mathfrak{I}^{\Phi} \frac{\bar{t_1} \ldots \bar{t_n}}{x_1 \ldots x_n} \models_{LFP_{fin}} \varphi$ (já que $T^{\Phi} = \{\bar{t} | t \in T^S\}$)

sse existem $t_1, \ldots, t_n \in T^S$ com $\mathfrak{I}^{\Phi} \frac{\mathfrak{I}^{\Phi}(t_1) \ldots \mathfrak{I}^{\Phi}(t_n)}{x_1 \ldots x_n} \models_{LFP_{fin}} \varphi$ (por (a))

sse existem $t_1, \ldots, t_n \in T^S$ com $\mathfrak{I}^{\Phi} \models_{LFP_{fin}} \varphi \frac{t_1 \ldots t_n}{x_1 \ldots x_n}$ (por (a)) (Pelo lema da substituição).

(2) segue similarmente de (1).

Definição 5.9 *(a) Φ é completo para negação (negation complete) sse para toda fórmula φ,*

$$\Phi \vdash_{LFP_{fin}} \varphi \text{ ou } \Phi \vdash_{LFP_{fin}} \neg\varphi.$$

(b) Φ contém testemunhas (witness) sse para toda fórmula da forma $\exists x\varphi$ existe um termo t tal que $\Phi \vdash_{LFP_{fin}} (\exists x\varphi \rightarrow \varphi\frac{t}{x})$.

Lema 5.5 *Suponha que Φ é consistente e completo para negação e que contém testemunhas. Então as seguintes afirmações são válidas para todo φ e ψ:*

(a) $\Phi \vdash_{LFP_{fin}} \neg\varphi$ sse não $\Phi \vdash_{LFP_{fin}} \varphi$.

(b) $\Phi \vdash_{LFP_{fin}} (\varphi \vee \psi)$ sse $\Phi \vdash_{LFP_{fin}} \varphi$ ou $\Phi \vdash_{LFP_{fin}} \psi$.

(c) $\Phi \vdash_{LFP_{fin}} (\varphi \wedge \psi)$ sse $\Phi \vdash_{LFP_{fin}} \varphi$ e $\Phi \vdash_{LFP_{fin}} \psi$.

(d) $\Phi \vdash_{LFP_{fin}} \forall x\varphi$ sse para todo termo t, $\Phi \vdash_{LFP_{fin}} \varphi\frac{t}{x}$.

(e) $\Phi \vdash_{LFP_{fin}} \exists x\varphi$ sse há um termo t com $\Phi \vdash_{LFP_{fin}} \varphi\frac{t}{x}$.

(f) $\Phi \vdash_{LFP_{fin}} [\boldsymbol{lfp}_{R,\vec{x}}\varphi(R,\vec{x})](\vec{t})$ sse $\Phi \vdash_{LFP_{fin}} \varphi^0(\vec{t})$ ou $\Phi \vdash_{LFP_{fin}} \varphi^1(\vec{t})$ ou $\ldots$ ou $\Phi \vdash_{LFP_{fin}} \varphi^n(\vec{t}) \ldots$.

Prova:

(a) Já que Φ é completo para negação, temos $\Phi \vdash_{LFP_{fin}} \varphi$ ou $\Phi \vdash_{LFP_{fin}} \neg\varphi$; e como Φ é consistente, $\Phi \vdash_{LFP_{fin}} \varphi$ sse não $\Phi \vdash_{LFP_{fin}} \neg\varphi$.

(b) Primeiramente seja $\Phi \vdash_{LFP_{fin}} (\varphi \vee \psi)$. Se não $\Phi \vdash_{LFP_{fin}} \varphi$, então $\Phi \vdash_{LFP_{fin}} \neg\varphi$ (já que Φ é completo para negação) e pelo silogismo disjuntivo em FOL temos $\Phi \vdash_{LFP_{fin}} \psi$. A outra direção segue imediatamente das regras de $\vee$.

(c) Como $\Phi \vdash_{LFP_{fin}} (\varphi \wedge \psi)$ e Φ é negation complete e consistente, então não pode ocorrer $\Phi \vdash_{LFP_{fin}} \varphi$ nem $\Phi \vdash_{LFP_{fin}} \psi$ o que conclui a ida. A volta é imediata a partir da regra $(\wedge I)$.

(d) A ida é trivial. Quanto a volta, suponha que $\Phi \vdash_{LFP_{fin}} \neg\forall x\varphi$. É fácil provar que $\neg\forall x\varphi \vdash_{LFP_{fin}} \exists x\neg\varphi$, portanto, como Φ contém testemunhas, há um termo t tal que $\Phi \vdash_{LFP_{fin}} \neg\varphi\frac{t}{x}$, um absurdo. Logo $\Phi \vdash_{LFP_{fin}} \forall x\varphi$.

(e) Seja $\Phi \vdash_{LFP_{fin}} \exists x\varphi$. Para a ida, já que Φ contém testemunhas, há um termo t tal que $\Phi \vdash_{LFP_{fin}} (\exists x\varphi \to \varphi\frac{t}{x})$. Usando Modus Ponens, temos $\Phi \vdash_{LFP_{fin}} \varphi\frac{t}{x}$. Para a volta, seja $\Phi \vdash_{LFP_{fin}} \varphi\frac{t}{x}$ para um termo t. Então a regra de introdução existencial nos dá $\Phi \vdash_{LFP_{fin}} \exists x\varphi$.

(f) Suponha $\Phi \vdash_{LFP_{fin}} [\mathbf{lfp}_{R,\vec{x}}\ \varphi(R,\vec{x})](\vec{t})$ e que não $\Phi \vdash_{LFP_{fin}} \varphi^0(\vec{t})$ e não $\Phi \vdash_{LFP_{fin}} \varphi^1(\vec{t})$ e .... Logo como Φ é completo para negação e consistente temos que $\Phi \vdash_{LFP_{fin}} \neg\varphi^0(\vec{t})$ e $\Phi \vdash_{LFP_{fin}} \neg\varphi^1(\vec{t})$, ... Portanto, pela regra de eliminação $(LFP\ E)$:

$$\dfrac{\begin{array}{c}\Pi\\ [\mathbf{lfp}_{R,\vec{x}}\ \varphi(R,\vec{x})](\vec{t})\end{array} \quad \dfrac{\varphi^0(\vec{t}) \quad \begin{array}{c}\Pi_0\\ \neg\varphi^0(\vec{t})\end{array}}{\bot} \quad \dfrac{\varphi^1(\vec{t}) \quad \begin{array}{c}\Pi_1\\ \neg\varphi^1(\vec{t})\end{array}}{\bot} \quad \cdots}{\bot}\ (LFP\ E).$$

Logo Φ não é consistente o que é um absurdo. O inverso segue da regra $(LFP\ I)$. □

O próximo teorema é uma contribuição nossa, e é fundamental para determinar a finitude do domínio que iremos tratar.

Teorema 5.5 *Seja Φ um conjunto consistente que possui testemunhas e completo para negação. Então $\mathfrak{I}^\Phi$ possui um domínio finito.*

Prova: Suponha que $\mathfrak{I}^\Phi$ possua um domínio infinito, ou seja, T^Φ é infinito. Como Φ é completo para negação, temos que para $\bar{t}_1 \neq \bar{t}_2$, tal que $t_1, t_2 \in T^S$, $\Phi \vdash \neg t_1 \equiv t_2$. Como existe uma quantidade infinita de classes $\bar{t}$, logo deduzimos que $\Phi \vdash \lambda_{\geq i}$, para todo $i \geq 2$. Portanto, usando a regra $(FIN\ \ \bot)$, Φ é inconsistente, absurdo. □

Teorema 5.6 *Seja Φ um conjunto consistente de fórmulas que é completo para negação e contém testemunhas. Então para todo φ pertencendo a LFP_{fin},*

$$\mathfrak{I}^\Phi \models_{LFP_{fin}} \varphi \ \textit{sse}\ \Phi \vdash_{LFP_{fin}} \varphi.$$

Prova: Provaremos por indução transfinita em $rk(\varphi)$. Se $rk(\varphi) = 0$, então φ é atômico, e (5.4) mostra que é válido. O passo de indução é separado nos seguintes casos:

1. $\varphi = \neg\psi$: $\mathfrak{I}^\Phi \models_{LFP_{fin}} \neg\psi$

   sse não $\mathfrak{I}^\Phi \models_{LFP_{fin}} \psi$

   sse não $\Phi \vdash_{LFP_{fin}} \psi$ (hipótese de indução)

   sse $\Phi \vdash_{LFP_{fin}} \neg\psi$ (por 5.5.(a));

2. $\varphi = (\psi \vee \varphi)$: $\mathfrak{I}^{\Phi} \models_{LFP_{fin}} (\psi \vee \varphi)$

   sse $\mathfrak{I}^{\Phi} \models_{LFP_{fin}} \psi$ ou $\mathfrak{I}^{\Phi} \models_{LFP_{fin}} \varphi$

   sse $\Phi \vdash_{LFP_{fin}} \psi$ ou $\Phi \vdash_{LFP_{fin}} \varphi$ (hipótese de indução)

   sse $\Phi \vdash_{LFP_{fin}} (\psi \vee \varphi)$ (por 5.5.(b));

3. $\varphi = (\psi \wedge \varphi)$: $\mathfrak{I}^{\Phi} \models_{LFP_{fin}} (\psi \wedge \varphi)$

   sse $\mathfrak{I}^{\Phi} \models_{LFP_{fin}} \psi$ e $\mathfrak{I}^{\Phi} \models_{LFP_{fin}} \varphi$

   sse $\Phi \vdash_{LFP_{fin}} \psi$ e $\Phi \vdash_{LFP_{fin}} \varphi$ (hipótese de indução)

   sse $\Phi \vdash_{LFP_{fin}} (\psi \wedge \varphi)$ (por 5.5.(c));

4. $\varphi = \forall x \psi$: $\mathfrak{I}^{\Phi} \models_{LFP_{fin}} \forall x \psi$

   para todo t, $\mathfrak{I}^{\Phi} \models_{LFP_{fin}} \psi\frac{t}{x}$ (Por 5.4.(c))

   para todo t, $\Phi \vdash_{LFP_{fin}} \psi\frac{t}{x}$ (hipótese de indução, já que $rk(\psi\frac{t}{x}) = rk(\psi) < rk(\varphi)$, conforme definição 5.3 e o lema 5.5.(d));

5. $\varphi = \exists x \psi$: $\mathfrak{I}^{\Phi} \models_{LFP_{fin}} \exists x \psi$

   sse existe um t tal que $\mathfrak{I}^{\Phi} \models_{LFP_{fin}} \psi\frac{t}{x}$ (pelo lema 5.4.(c))

   sse existe um t tal que $\Phi \vdash_{LFP_{fin}} \psi\frac{t}{x}$ (hipótese de indução, já que $rk(\psi\frac{t}{x}) = rk(\psi) < rk(\varphi)$, conforme definição 5.3);

   sse $\Phi \vdash_{LFP_{fin}} \exists x \psi$ (por 5.5.(e)). $\square$

6. $\varphi = [\mathbf{lfp}_{R,\vec{x}}\ \varphi(R,\vec{x})](\vec{t})$: $\mathfrak{I}^{\Phi} \models_{LFP_{fin}} [\mathbf{lfp}_{R,\vec{x}}\ \varphi](\vec{t})$

   sse $\mathfrak{I} \models_{LFP_{fin}} \varphi^0(\vec{t})$ ou $\mathfrak{I} \models_{LFP_{fin}} \varphi^1(\vec{t})$ ou .... (pelo teorema 4.5 temos $\mathfrak{I} \models_{LFP_{fin}} \varphi^i(\vec{t})$ para algum $i \in N$).

   sse $\Phi \vdash_{LFP_{fin}} \varphi^0(\vec{t})$ ou $\Phi \vdash_{LFP_{fin}} \varphi^1(\vec{t})$ ou .... (hipótese de indução)

   sse $\Phi \vdash_{LFP_{fin}} [\mathbf{lfp}_{R,\vec{x}}\ \varphi(R,\vec{x})](\vec{t})$ (por 5.5.(f)).

Teorema 5.7 *Se Φ é um conjunto consistente que é completo para negação e possui testemunhas, então $\mathfrak{I}^{\Phi} \models_{LFP_{fin}} \Phi$.* $\square$

5.6 Completude

Para provarmos a completude, definiremos o conceito de f-testemunha, que nos fala sobre a cardinalidade do conjunto na qual ela está inserida. Tal noção é útil para obtermos uma compacidade restrita, como veremos mais tarde, a fim de obtermos o teorema da completude atra´vés da utilização do teorema de Henkin.

Definição 5.10 *Um conjunto Φ possui f-testemunhas se para algum $i \geq 2$, $\neg\lambda_{\geq i} \in \Phi$.*

Teorema 5.8 *Se $\Phi \vdash_{LFP_{fin}} \neg\lambda_{\geq i}$ então $\Phi \vdash_{LFP_{fin}} \neg\lambda_{\geq i+1}$.*

Prova: Como é demonstrado na lógica clássica de primeira ordem que $\lambda_{\geq i+1} \rightarrow \lambda_{\geq i}$, basta utilizarmos a contrapositiva. □

Teorema 5.9 *Se um conjunto Φ possui f-testemunhas, então para toda fórmula da forma $[\boldsymbol{lfp}_{R,\vec{x}}\ \varphi(R,\vec{x})](\vec{t})$ temos $\Phi \vdash_{LFP_{fin}} \forall x_1 \ldots \forall x_n([\boldsymbol{lfp}_{R,\vec{x}}\ \varphi(R,\vec{x})](x_1,\ldots,x_n) \rightarrow \varphi^{i^{|\vec{t}|}}(x_1,\ldots,x_n))$, para algum i tal que $\neg\lambda_{\geq i+1} \in \Phi$.*

Prova:
Como $\neg\lambda_{\geq i+1} \in \Phi$, para algum $i \in N$

$$\cfrac{\cfrac{\cfrac{[\mathbf{lfp}_{R,\vec{x}}\ \varphi(R,\vec{x})](x_1,\ldots,x_n)^0 \quad \neg\lambda_{\geq i+1}}{\varphi^{i^{|\vec{t}|}}(x_1,\ldots,x_n)}\ (LFP-FP)}{[\mathbf{lfp}_{R,\vec{x}}\ \varphi(R,\vec{x})](x_1,\ldots,x_n) \rightarrow \varphi^{i^{|\vec{t}|}}(x_1,\ldots,x_n)}\ (\rightarrow I),\ \text{eliminando } 0}{\forall x_1 \ldots \forall x_n([\mathbf{lfp}_{R,\vec{x}}\ \varphi(R,\vec{x})](x_1,\ldots,x_n) \rightarrow \varphi^{i^{|\vec{t}|}}(x_1,\ldots,x_n))}\ (\forall I)^n \quad . \ \square$$

Teorema 5.10 *Para todo conjunto Φ que possui f-testemunhas, se Π é uma derivação de φ a partir do conjunto Φ, então existe uma derivação Π' de φ a partir do conjunto Φ tal que $|\Pi'| < \omega$ (onde ω é o primeiro ordinal não-natural).*

Prova: Indução nas regras de inferência. Quanto às regras de primeira-ordem e as regra $(LFP\ FP)$ e $(LFP\ I)$ são triviais pois, dado que a prova das premissas possuem $|\Pi_i| < \omega$ para cada (e temos um número finito delas), então a árvore possui tamanho menor que ω. Seja a regra $(FIN\ \perp)$, então temos:

$$\cfrac{\begin{matrix}\Pi_1 & \Pi_2 & \\ \lambda_{\geq 2} & \lambda_{\geq 3} & \ldots\end{matrix}}{\perp}$$

então, pela hipótese de indução, existem Π'_1, Π'_2, …, tais que:

$$\dfrac{\begin{array}{ccc}\Pi'_1 & \Pi'_2 & \\ \lambda_{\geq 2} & \lambda_{\geq 3} & \dots\end{array}}{\bot}$$

e $|\Pi'_i| < \omega$ para todo natural i. Como Φ contém f-testemunhas logo, para algum j, $\neg\lambda_{\geq j} \in \Phi$. Portanto podemos reescrever a derivação anterior como:

$$\dfrac{\begin{array}{cc}\Pi'_{j-1} & \\ \lambda_{\geq j} & \neg\lambda_{\geq j}\end{array}}{\bot}$$

que possui tamanho menor que ω.

Resta provar para $(LFP-E)$, utilizando a hípótese de indução, temos:

$$\dfrac{\begin{array}{cccccc} & \varphi^0(\vec{t}) & \varphi^1(\vec{t}) & & \varphi^i(\vec{t}) & \\ \Pi & \Pi'_0 & \Pi'_1 & & \Pi'_i & \\ [\mathbf{lfp}_{R,\vec{x}}\ \varphi(R,\vec{x})](\vec{t}) & \sigma & \sigma & \dots & \sigma & \dots\end{array}}{\sigma}\ (LFP\ E)$$

Como Φ possui f-testemunhas, então pelo teorema 5.9 temos uma derivação finita $\Phi \vdash_{LFP_{fin}} \forall x_1 \dots \forall x_n [\mathbf{lfp}_{R,\vec{x}}\ \varphi(R,\vec{x})](x_1,\dots,x_n) \to \varphi^{i^{|\vec{t}|}}(x_1,\dots,x_n)$ e portanto obtemos:

$$\dfrac{\begin{array}{cc}\Pi' & \dfrac{\dfrac{\dfrac{[\mathbf{lfp}_{R,\vec{x}}\ \varphi(R,\vec{x})](x_1,\dots,x_n)^0 \quad \neg\lambda_{\geq i+1}}{\varphi^{i^{|\vec{t}|}}(x_1,\dots,x_n)}\ (LFP-FP)}{[\mathbf{lfp}_{R,\vec{x}}\ \varphi(R,\vec{x})](x_1,\dots,x_n) \to \varphi^{i^{|\vec{t}|}}(x_1,\dots,x_n)}\ (\to I),\text{ eliminando } 0}{\overline{\overline{\forall x_1 \dots \forall x_n([\mathbf{lfp}_{R,\vec{x}}\ \varphi(R,\vec{x})](x_1,\dots,x_n) \to \varphi^{i^{|\vec{t}|}}(x_1,\dots,x_n))}}}\ (\forall I)^n \\ & \vdots \\ [\mathbf{lfp}_{R,\vec{x}}\ \varphi(R,\vec{x})](\vec{t}) & [\mathbf{lfp}_{R,\vec{x}}\ \varphi(R,\vec{x})](\vec{t}) \to \varphi^{i^{|\vec{t}|}}(\vec{t})\end{array}}{\begin{array}{c}\varphi^{i^{|\vec{t}|}}(\vec{t}) \\ \Pi'_{i^{|\vec{t}|}} \\ \sigma\end{array}}$$

que possui tamanho menor que ω (observe que talvez na reescrita acima seja necessário alterar as variáveis que possuem restrições na aplicação das regras $\forall I$ e $\exists E$ na derivação $\Pi'_{i^{|\vec{t}|}}$ por outras que não ocorram nas hipóteses da nova dedução, caso isto aconteça basta substituirmos por variáveis que não ocorrem na prova e obtemos o resultado). $\square$

Corolário 5.2 (Compacidade Restrita) *Para todo conjunto Φ consistente que possui f-testemunhas a seguinte relação é válida:*

$$\Phi \text{ é consistente sse todo } \Phi_0 \subset \Phi \text{ finito é consistente.}$$

Prova: O teorema 5.10 nos indica que podemos dispensar regras infinitárias em derivações para conjuntos com f-testemunhas. Portanto, Φ é consistente sse todo subconjunto finito é consistente. □

Teorema 5.11 *Para todo conjunto consistente Φ_0 existe um conjunto consistente Φ tal que $\Phi_0 \subseteq \Phi$ contém f-testemunhas.*

Prova: Como Φ_0 é consistente significa que para algum i, $\Phi_0 \nvdash_{LFP_{fin}} \lambda_{\geq i}$ (caso contrário provaríamos $\perp$). Então definimos o conjunto Φ como:

$$\Phi := \Phi_0 \cup \{\neg \lambda_{\geq i}\}.$$

Usando 5.4 (a), claramente Φ é consistente. □

Teorema 5.12 *Seja Φ consistente e com um número finito de variáveis livres. Então existe um conjunto Ψ consistente tal que $\Phi \subseteq \Psi$ e Ψ contém testemunhas.*

Prova: Sem perda de generalidade podemos considerar que Φ possui f-testemunhas (pelo teorema 5.11). Pelo Lema 5.2 L^S é contável. Portanto, considere uma lista de todas as fórmulas que estão no escopo de um quantificador existencial: $\exists x_0 \varphi_0, \exists x_1 \varphi_1, \ldots$. Iremos definir indutivamente uma lista de fórmulas $\psi_0, \psi_1, \ldots$, que adicionaremos à Φ, onde cada ψ_n é uma fórmula que contém uma "testemunha" para $\exists x_n \varphi_n$.

Suponha que para todo número natural m tal que $m < n$ a fórmula ψ_m está definida. Como o número de variáveis livre de Φ é finito, somente um número finito de variáveis ocorrem em $\Phi \cup \{\psi_m \mid m < n\} \cup \{\exists x_n \varphi_n\}$. Então considere y_n a variável com menor índice distinta dessas. Definimos a fórmula ψ_n como sendo:

$$\psi_n := (\exists x_n \varphi_n \rightarrow \varphi \tfrac{y_n}{x_n}).$$

Considere agora o seguinte conjunto Ψ em que adicionamos todos os ψ_i's:

$$\Psi := \Phi \cup \{\psi_0, \psi_1, \ldots\}.$$

Como temos que $\Phi \subseteq \Psi$, então Ψ claramente contém testemunhas. Resta-nos provar que Ψ é consistente. Considere então a seguinte seqüência:

$$\Phi_n := \Phi \cup \{\psi_m \mid m < n\}.$$

É fácil notar que $\Phi_0 \subseteq \Phi_1 \subseteq \Phi_2 \ldots$ e $\Psi = \bigcup_{n \in N} \Phi_n$. Ψ possui f-testemunhas já que $\Phi \subseteq \Psi$. Se Ψ for inconsistente, logo para algum conjunto finito $\Gamma \subseteq \Psi$, Γ é inconsistente (pelo corolário 5.2). Portanto se provarmos a consistência de cada Φ_n, provaremos consequentemente a consistência de Ψ (caso contrário, obteríamos um Γ finito tal que $\Gamma \subseteq \Phi_i$, para algum $i \in N$ e Inc Γ).

Iremos provar por indução em n. Para $n = 0$ é válido pois Φ_0 é consistente pela hipótese. Suponha agora que Φ_n é consistente e Φ_{n+1} é inconsistente. Então para todo φ existe um conjunto finito $\Gamma \subseteq \Phi_n$ tal que

$$\begin{array}{c} \Gamma \cup \{\psi_n\} \\ \Pi_1 \\ \varphi \end{array}$$

ou seja, existe a seguinte derivação:

$$\begin{array}{c} \Gamma \cup \{(\neg\exists x_n \varphi_n \vee \varphi_n \frac{y_n}{x_n})\} \\ \Pi_1 \\ \varphi \end{array} .$$

Considere φ uma sentença qualquer. Portanto existem as seguintes derivações:

$$\begin{array}{c} \dfrac{\neg\exists x_n \varphi_n}{(\neg\exists x_n \varphi_n \vee \varphi_n \frac{y_n}{x_n})} \\ \Pi_1 \\ \varphi \end{array}$$

e

$$\dfrac{\exists x_n \varphi_n \qquad \begin{array}{c} \dfrac{\varphi_n \frac{y_n}{x_n}}{(\neg\exists x_n \varphi_n \vee \varphi \frac{y_n}{x_n})} \\ \Pi_1 \\ \varphi \end{array}}{\varphi}\ (\exists\, E), \text{já que } y_n \notin \Gamma \cup \{\exists x_n \varphi_n\} \cup \{\varphi\}.$$

Onde um subconjunto de fórmulas de Γ ocorrem como hipóteses em Π_1. Concluimos que $\Gamma \cup \{\exists x_n \varphi_n\} \vdash_{LFP_{fin}} \varphi$ e $\Gamma \cup \{\neg \exists x_n \varphi_n\} \vdash_{LFP_{fin}} \varphi$. Logo, pelo teorema 5.4(c), Γ é inconsistente, o que é um absurdo. $\square$

Teorema 5.13 *Dado Φ consistente, então existe um conjunto Θ consistente tal que $\Phi \subseteq \Theta$ e Θ é completo para negação.*

Prova: Suponha que Φ é consistente e seja $\varphi_0, \varphi_1, \varphi_2, \ldots$ uma enumeração de fórmulas em L^S de LFP. Pelo teorema 5.11, sem perda de generalidade, suponha que Φ possui f-testemunhas. Logo, para um certo $i \in N$, temos que $\Phi \vdash_{LFP_{fin}} \neg \lambda_{\geq i}$. Definimos indutivamente a seguinte sequência:

$$\Theta_0 := \Phi$$

e

$$\Theta_{n+1} = \begin{cases} \Theta_n \cup \{\varphi_n\} & \text{, se Con } \Theta_n \cup \{\varphi_n\}, \\ \Theta_n & \text{, caso contrário,} \end{cases}$$

e definimos o conjunto

$$\Theta := \bigcup_{n \in N} \Theta_n.$$

Claramente $\Phi \subseteq \Theta_0$. Vemos que todos Θ_n's são consistentes, resta provarmos que Θ é consistente e completo para negação. Suponha que Θ é inconsistente. Logo, como Θ possui f-testemunhas, existe um conjunto finito $\Phi' \subseteq \Theta$ tal Φ' é inconsistente. Portanto, para algum Θ_i temos $\Phi' \subseteq \Theta_i$. Assim Θ_i é inconsistente, absurdo. Logo Θ é consistente. Θ é completo para negação pois dado $\varphi = \varphi_n$ e não $\Theta \vdash_{LFP_{fin}} \neg\varphi$, então Con $\Theta \cup \{\varphi_n\}$ (teorema 5.4.(b)) e portanto Con $\Theta_n \cup \{\varphi\}$. Então $\Theta_{n+1} = \Theta_n \cup \{\varphi\}$. Dessa forma $\varphi \in \Theta$ e portanto $\Theta \vdash_{LFP_{fin}} \varphi$. $\square$

Teorema 5.14 *Seja Φ consistente tal que Φ possui um número finito de variáveis livres. Então Φ é satisfatível.*

Prova: Pelo teoremas 5.13, 5.12 e 5.6. $\square$

Iremos agora considerar um número qualquer de variáveis livres.

Teorema 5.15 *Dado Φ consistente, então Φ é satisfatível.*

Prova: Considere o número de variáveis livres de Φ infinito. Vamos mostrar como reduzir o problema de satisfabilidade de Φ para o teorema 5.14.

Como Φ é consistente, então usando a construção do teorema 5.13, existe um conjunto consistente Ψ tal que $\Phi \subseteq \Psi$ e Ψ é completo para negação e possui f-testemunhas. Consequentemente Ψ possui uma quantidade infinita de variáveis livres. Nosso objetivo é encontrar um modelo para Ψ (e portanto para Φ).

Temos que para cada termo $t_1, t_2 \in T^S$ ou $\Psi \vdash_{LFP_{fin}} t_1 \equiv t_2$ ou $\Psi \vdash_{LFP_{fin}} \neg t_1 \equiv t_2$, já que Ψ é completo para negação. Portanto, podemos criar classes de equivalência $\bar{t}$ tal que $\bar{t} := \{t' \in T^S \mid \Psi \vdash_{LFP_{fin}} t \equiv t'\}$. Concluimos, pela consistência de Ψ, que há uma quantidade finita de classes de equivalência (caso contrário provaríamos todos $\lambda_{\geq i}$'s). Suponha que tenhamos k classes de equivalência listadas como $\bar{t}_1, \bar{t}_2, \ldots, \bar{t}_k$. Considere $x_{\bar{i}}$ a variável pertencente à $\bar{t}_i$ com menor índice. Seja o conjunto

$$\Phi' := \{\varphi \frac{x_{\bar{1}} \ldots x_{\bar{k}}}{\bar{t}_1 \ldots \bar{t}_k} \mid \varphi \in \Phi\}.$$

Onde a notação $\frac{x_{\bar{1}} \ldots x_{\bar{k}}}{\bar{t}_1 \ldots \bar{t}_k}$ indica que todo termo pertencente à classe $\bar{t}_i$ é substituído por $x_{\bar{i}}$. Portanto Φ' possui uma quantidade finita de variáveis e é consistente já que, pela construção e consistência de Ψ, temos $\Phi' \subseteq \Psi$. Usando o teorema 5.14 concluímos que Φ' é satisfatível. Considere agora que uma estrutura que satisfaz Φ' tem a forma $\mathfrak{I} = (\mathfrak{A}, \beta)$. Para construirmos uma estrutura que satisfaça Ψ basta considerarmos um assinalamento $\beta'(t) := \beta(x_{\bar{i}})$ para todo termo $t \in \bar{t}_i$. Então a estrutura $\mathfrak{I}' = (\mathfrak{A}, \beta')$ satisfaz Ψ. $\square$

5.7 Exemplo de prova em LFP_{fin}

Agora vamos fornecer um exemplo de prova em LFP_{fin}. Temos que a fórmula $\alpha(S,x)$ dada por

$$\forall y(E(y,x) \rightarrow S(y))$$

é associada com um operador F_α tal que dado um conjunto X ele retorna todos os nós a em um grafo tal que todos os nós b os quais possuem uma aresta com a estão em X. Iterando o operador temos que $F_\alpha(\emptyset)$ é o conjunto de nós com grau de entrada zero. $FF_\alpha(\emptyset)$ é o conjunto de nós a's tais que todos os nós b's com arestas $(b,a) \in E$ tem grau de entrada zero, ou seja, todos os nós a's tais que todos os caminhos terminando em a possuem tamanho no máximo um. Portanto a iteração i nos estabelece o conjunto de todos os nós a's tais que todos os caminhos que terminam neles tem tamanho no máximo i. Calculando o ponto-fixo, temos que todos os nós que terminam em a são finitos, e podemos expressar pela seguinte fórmula:

$$\forall z[\mathbf{lfp}_{S,\vec{x}}\ \alpha(S,\vec{x})](z)$$

Portanto tal fórmula testa se um grafo é acíclico. Temos que um grafo com mais de dois vértices que é acíclico não é completo. Definimos que um grafo é completo se para cada dois vértices distintos há uma aresta, ou seja, $\forall x\forall y(\neg x\equiv y\leftrightarrow E(x,y))$. Iremos provar em LFP_{fin} que um grafo acíclico com mais de dois vértices (e com um número finito deles) não pode ser completo:

$$\dfrac{\dfrac{\dfrac{\dfrac{\exists x_1\exists x_2\neg x_1\equiv x_2\wedge\forall z[\mathbf{lfp}_{S,\vec{x}}\ \alpha(S,\vec{x})](z)^0}{\exists x_1\exists x_2\neg x_1\equiv x_2}\qquad\dfrac{\exists x_2\neg v\equiv x_2\quad\Sigma}{\bot}}{\bot}}{\neg\forall x\forall y(\neg x\equiv y\leftrightarrow E(x,y))}\,{,1}}{(\exists x_1\exists x_2\neg x_1\equiv x_2\wedge\forall z[\mathbf{lfp}_{S,\vec{x}}\ \alpha(S,\vec{x})](z))\to\neg\forall x\forall y(\neg x\equiv y\leftrightarrow E(x,y))}\,{,0}$$

onde temos que $\Sigma =$

$$\dfrac{\dfrac{\dfrac{\exists x_1\exists x_2\neg x_1\equiv x_2\wedge\forall z[\mathbf{lfp}_{S,\vec{x}}\ \alpha(S,\vec{x})](z)}{\forall z[\mathbf{lfp}_{S,\vec{x}}\ \alpha(S,\vec{x})](z)}}{[\mathbf{lfp}_{S,\vec{x}}\ \alpha(S,\vec{x})](y)}\qquad\begin{matrix}\Pi_1\\ \bot\end{matrix}\quad\begin{matrix}\Pi_2\\ \bot\end{matrix}\quad\begin{matrix}\Pi_3\\ \bot\end{matrix}\quad\ldots}{\bot}$$

e $\Pi_1 =$

$$\dfrac{\neg y\equiv y\quad\overline{y\equiv y}}{\bot}$$

e $\Pi_2 =$

$$\dfrac{\dfrac{\dfrac{\forall z_1(E(z_1,y)\to\neg y\equiv y)}{E(v,y)\to\neg y\equiv y}\qquad\dfrac{\neg v\equiv y\qquad\dfrac{\forall x\forall y(\neg x\equiv y\leftrightarrow E(x,y)^1)}{\neg v\equiv y\to E(v,y)}}{E(v,y)}}{\neg y\equiv y}\qquad\overline{y\equiv y}}{\bot}$$

e $\Pi_3 =$

$$
\begin{prooftree}
\AxiomC{$\forall z_2(E(z_2,y) \rightarrow \forall z_1(E(z_1,z_2) \rightarrow \neg z_2 \equiv z_2))$}
\doubleLine
\UnaryInfC{$E(v,y) \rightarrow \forall z_1(E(z_1,v) \rightarrow \neg v \equiv v)$}
\AxiomC{$\neg v \equiv y$}
\AxiomC{$\forall x \forall y(\neg x \equiv y \leftrightarrow E(x,y)^1)$}
\doubleLine
\UnaryInfC{$\neg v \equiv y \rightarrow E(v,y)$}
\BinaryInfC{$E(v,y)$}
\BinaryInfC{$\forall z_1(E(z_1,v) \rightarrow \neg v \equiv v)$}
\UnaryInfC{$E(y,v) \rightarrow \neg v \equiv v$}
\AxiomC{Σ_1}
\BinaryInfC{$\neg v \equiv v$}
\AxiomC{}
\UnaryInfC{$v \equiv v$}
\BinaryInfC{$\bot$}
\end{prooftree}
$$

onde $\Sigma_1 =$

$$
\begin{prooftree}
\AxiomC{$\neg v \equiv y$}
\doubleLine
\UnaryInfC{$\neg y \equiv v$}
\AxiomC{$\forall x \forall y(\neg x \equiv y \leftrightarrow E(x,y))^1$}
\doubleLine
\UnaryInfC{$\neg y \equiv v \rightarrow E(y,v)$}
\BinaryInfC{$E(y,v)$}
\end{prooftree}
$$

e $\Pi_i = \ldots$

Notamos que todo Π_i prova $\bot$, já que sempre podemos chegar à derivação de uma fórmula $\neg x \equiv x$ para um termo x. Logo a derivação está correta.

5.8 Eliminação da Regra $(LFP\ PF)$

Dado o sistema LFP_{fin} que mostramos no ínicio do capítulo, podemos eliminar a regra de inferência LFP_{fin} que nos foi útil justamente para a prova da completude. Outro motivo de sua eliminação é que a prova de teoremas de normalização para LFP_{fin} seriam bastante complexas já que seria complicado definir uma redução para $(LFP\ PF)$. Podemos eliminá-la de forma que se existe uma derivação Π de φ a partir de um conjunto Γ em LFP_{fin} então existe uma derivação Π' de φ a partir de um conjunto Γ que não utiliza a regra $(LFP\ PF)$.

Antes de provarmos, precisaremos da seguinte definição:

Definição 5.11 *Definimos a função* $nd(\varphi)$ *como:*

1. $nd(\varphi) = 0$, *se* φ *é atômico;*

2. $nd(\neg\varphi) = nd(\varphi)$;

3. $nd(\varphi * \psi) = max(nd(\varphi), nd(\psi))$, *onde* $* \in \{\rightarrow, \wedge, \vee\}$;

4. $nd(\forall x\varphi) = nd(\varphi)$;

5. $nd(\exists x\varphi) = nd(\varphi)$;

6. $nd([\boldsymbol{lfp}_{R,\vec{x}}\ \varphi(R,\vec{x})](\vec{t})) = 1$.

Vemos que a função $nd(\varphi)$ nos retorna 1 se φ possui fórmula da forma $[\mathbf{lfp}_{R,\vec{x}}\ \psi(R,\vec{x})](\vec{t})$ e 0 caso contrário. Com este fato em mente, vemos que $nd(\varphi) = 0$ em geral quer dizer que φ pertence a FOL. Com base nisso, propomos o seguinte teorema:

Teorema 5.16 *Dada a sequência da definição 4.10 (do capítulo 4) para um determinado φ, temos que existe uma derivação de $\varphi^k(\vec{t}), \neg\lambda_{\geq i+1} \vdash_{LFP_{fin}} \varphi^{i^{|\vec{t}|}}(\vec{t})$ (onde $k > i^{|\vec{t}|}$) usando somente regras pertencentes a FOL.*

Prova: Basta analisar os casos em $nd(\varphi)$.
Caso 1: $nd(\varphi) = 0$

Então temos que $\varphi^k(\vec{t})$ e $\varphi^{i^{|\vec{t}|}}(\vec{t})$ são fórmulas pertencentes a FOL. Como $\varphi^k(\vec{t}), \neg\lambda_{\geq i+1} \models \varphi^{i^{|\vec{t}|}}(\vec{t})$ então pela completude de FOL temos $\varphi^k(\vec{t}), \neg\lambda_{\geq i+1} \vdash \varphi^{i^{|\vec{t}|}}(\vec{t})$ e portanto $\varphi^k(\vec{t}), \neg\lambda_{\geq i+1} \vdash_{LFP_{fin}} \varphi^{i^{|\vec{t}|}}(\vec{t})$.

Caso 2: $nd(\varphi) = 1$

Então toda ocorrência de fórmulas em φ da forma $[\mathbf{lfp}_{R,\vec{x}}\ \psi(R,\vec{x})](\vec{t})$ substituímos por um único Q_i. Como exemplo:

$$[\mathbf{lfp}_{R,\vec{x}}\ \varphi_1(R,\vec{x})](\vec{t_1}) \wedge \sigma \wedge \varphi_2 \rightarrow [\mathbf{lfp}_{R,\vec{x}}\ \varphi_3(R,\vec{x})](\vec{t_2})$$

substituímos por

$$Q_1(\vec{t_1}) \wedge \sigma \wedge \varphi_2 \rightarrow Q_2(\vec{t_2}).$$

Dada uma fórmula φ, vamos denotar essa substituição aplicada em φ como φ_{TFOL}. Então temos que $\varphi^k_{TFOL}, \neg\lambda_{\geq i+1} \vdash_{LFP_{fin}} \varphi^{i^{|\vec{t}|}}_{TFOL}$, ou seja, à uma derivação de $\varphi^{i^{|\vec{t}|}}_{TFOL}$ a partir de $\varphi^k_{TFOL}, \neg\lambda_{\geq i+1}$ utilizando somente regras em FOL. Suponha que seja Π esta derivação. Então, desfazendo as trocas (trocando cada Q_i por sua respectiva fórmula da forma $[\mathbf{lfp}_{R,\vec{x}}\ \psi(R,\vec{x})](\vec{t})$), concluimos portanto que há uma derivação

$$\varphi^k(\vec{t}), \neg\lambda_{\geq i+1} \vdash_{LFP_{fin}} \varphi^{i^{|\vec{t}|}}(\vec{t})$$

utilizando somente regras em FOL. $\square$

Teorema 5.17 *Se $k > i$ então $\varphi^i(\vec{t}) \vdash_{LFP_{fin}} \varphi^k(\vec{t})$.*

Prova: Similar ao teorema 5.16. $\square$

Com estes dois últimos teoremas, podemos descartar a regra $(LFP\ PF)$ do sistema, pois dada uma derivação:

$$\dfrac{\begin{array}{c}\Pi_1\\ [\mathbf{lfp}_{R,\vec{x}}\ \varphi(R,\vec{x})](\vec{t})\end{array} \quad \begin{array}{c}\Pi_2\\ \neg\lambda_{\geq i+1}\end{array}}{\varphi^{i^{|\vec{t}|}}(\vec{t})}\ (LFP\ PF)$$
$$\Sigma$$

Basta trocarmos todas as ocorrências acima por

$$\dfrac{\begin{array}{c}\Pi_1\\ [\mathbf{lfp}_{R,\vec{x}}\ \varphi(R,\vec{x})](\vec{t})\end{array} \quad \begin{array}{c}\varphi^0(\vec{t})\\ \Sigma_0\\ \varphi^{i^{|\vec{t}|}}(\vec{t})\end{array} \quad \ldots \begin{array}{c}\varphi^{i^{|\vec{t}|}}(\vec{t})\\ \Sigma_i\\ \varphi^{i^{|\vec{t}|}}(\vec{t})\end{array} \quad \begin{array}{c}\varphi^{i^{|\vec{t}|}+1}(\vec{t})\\ \Sigma_{i^{|\vec{t}|}+1}\\ \varphi^{i^{|\vec{t}|}}(\vec{t})\end{array} \quad \ldots}{\varphi^{i^{|\vec{t}|}}(\vec{t})}\ (LFP\ E)$$
$$\Sigma \qquad ,$$

onde as derivações Σ_j com $j < i^{|\vec{t}|}$ utilizam somente regras de FOL (pelo teorema 5.17) da mesma forma para $j > i^{|\vec{t}|}$, e usando a prova de $\neg\lambda_{\geq i}$, temos o resultado utilizando o teorema 5.16.

Com este intuito, uma idéia é para em casos de conjuntos com f-testemunhas, além de não usarmos a regra $(LFP\ PF)$, poderíamos restringir a regra $(LFP\ E)$. No exemplo anterior poderíamos escrever apenas:

$$\dfrac{\begin{array}{c}\Pi_1\\ [\mathbf{lfp}_{R,\vec{x}}\ \varphi(R,\vec{x})](\vec{t})\end{array} \quad \begin{array}{c}\varphi^0(\vec{t})\\ \Sigma_0\\ \psi\end{array} \quad \ldots \quad \begin{array}{c}\varphi^{i^{|\vec{t}|}}(\vec{t})\\ \Sigma_i\\ \psi\end{array}}{\psi}\ (LFP\ E)_{res}.$$
$$\Sigma$$

Dado que $\neg\lambda_{\geq i}$ pertence ao conjunto Γ de onde queremos provar. Abordaremos esse assunto no próximo capítulo.

6 *Normalização para LFP_{fin} com restrições*

6.1 Introdução

O intuito deste capítulo é fornecer um teorema de normalização para o sistema de dedução natural LFP_{fin} com certas restrições. Abossrdaremos, portanto, uma normalização fraca para a lógica de primeira-ordem clássica com igualdade para, em seguida, apresentar um teorema de normalização para uma versão restrita do sistema dedutivo LFP_{fin}. Quanto à lógica de primeira-ordem, basicamente iremos estender a definição de segmento máximo proposta em MASSI [2] através do conceito de segmento equacional, definido sobre as regras de igualdade.

Quanto às seções temos:

Na seção 6.2 iremos definir alguns conceitos importantes para o desenvolvimento do resto do capítulo.

Abordaremos na seção 6.3 a noção de segmentos máximos, que podem ser vistos como redundâncias em uma árvore de dedução.

Será apresentado na seção 6.4 as reduções necessárias para a eliminação dos segmentos máximos mostrados na seção 6.3.

Na seção 6.6 apresentaremos um teorema de normalização fraca para FOL + regras de igualdade.

E, finalmente, apresentaremos na seção 6.7, um teorema de normalização para LFP_{fin}

com uma restrição: lidaremos apenas com conjunto de hipóteses que possuem f-testemunhas (definido no capítulo 5).

6.2 Definições e Teoremas

Nesta seção apresentaremos algumas definições e teoremas sobre árvores de prova. As definições aqui apresentadas são similares as propostas por MASSI [2] e PRAWITZ [17]. Deixaremos explícito durante a seção as definições e teoremas que adicionamos a mais por conta própria. Para este capítulo, consideraremos fórmulas da forma $\neg\varphi$ como sendo iguais a $\varphi \rightarrow \perp$ (Portanto não utilizaremos as regras $\neg I$ e $\neg E$).

Para o entendimento das seções a seguir, precisaremos dos seguintes conceitos:

Definição 6.1 (Grau de uma Fórmula) *O grau de uma fórmula φ, denotado por $d(\varphi)$, é um número natural, tal que:*

1. $d(\varphi) = 0$, *se φ é uma fórmula atômica;*
2. $d(\varphi_1 \Box \varphi_2) = d(\varphi_1) + d(\varphi_2) + 1$, *onde* $\varphi = (\varphi_1 \Box \varphi_2)$ *e* $\Box \in \{\wedge, \vee, \rightarrow\}$;
3. $d(\forall x \varphi_1) = d(\varphi_1) + 1$, *onde* $\varphi = (\forall x \varphi_1)$.
4. $d(\exists x \varphi_1) = d(\varphi_1) + 1$, *onde* $\varphi = (\exists x \varphi_1)$.

O comprimento em uma derivação Π pode ser visto como a quantidade de fórmulas que ocorrem nela. Iremos formalmente definir como:

Definição 6.2 (Comprimento de uma Derivação) *O comprimento de uma derivação Π (denotaremos por $l(\Pi)$), é um número natural, tal que:*

1. $l(\Pi) = 1$, *se Π é uma fórmula;*
2. $l(\Pi) = l(\Pi_1) + l(\Pi_2) + \ldots + l(\Pi_n) + 1$, *se* $\Pi = \dfrac{\Pi_1 \quad \Pi_2 \quad \ldots \quad \Pi_n}{\varphi}$.

Definição 6.3 *Seja Π a derivação* $\dfrac{\Pi_1 \quad \Pi_2 \quad \ldots \quad \Pi_n}{\varphi}$ *e sejam $\varphi_1, \varphi_2, \ldots, \varphi_n$, respectivamente, as fórmulas finais de $\Pi_1, \Pi_2, \ldots, \Pi_n$. Então, dizemos que $\varphi_1, \varphi_2, \ldots, \varphi_n$ ocorrem imediatamente acima φ, na ordem da esquerda para a direita. e que φ_i ocorre ao lado de φ_j $(i, j \leq n)$.*

Definição 6.4 (Ramo em uma Derivação) *Dada uma derivação* Π, *uma sequência* $\varphi_1, \varphi_2, \ldots, \varphi_n$ *de ocorrências de fórmulas é um ramo, se e somente se:*

1. φ_1 *é uma top-fórmula;*

2. φ_i *ocorre imediatamente acima de* φ_{i+1}, *para cada* $(i < n)$;

3. φ_n *é a fórmula final de* Π.

A partir do conceito de ramo em uma derivação, podemos finalmente definir o conceito de segmento clássico e o que iremos propor nesta dissertação: o segmento equacional.

Definição 6.5 (Segmento de uma Derivação) *Um segmento em uma derivação* Π, *é uma sequência* $\alpha = \varphi_1.\varphi_2.\ldots.\varphi_n$, *de ocorrências consecutivas de fórmulas em um ramo* Π, *tal que possui uma das seguintes formas:*

- *Segmento clássico (Proposto por MASSI [2] e PRAWITZ [17]):*
  1. φ_1 *não é uma conclusão de uma regra* $(\vee E)$ *ou* $(\exists E)$;
  2. φ_i $(i < n)$ *é uma premissa menor de uma regra* $(\vee E)$ *ou* $(\exists E)$;
  3. φ_n *não é uma premissa menor de uma regra* $(\vee E)$ *ou* $(\exists E)$.
- *Segmento equacional (Proposta nesta dissertação):*
  1. φ_1 *é a premissa primeira de uma regra* $EQ2$ *e não é a conclusão de uma regra* $EQ2$.
  2. φ_i $(i < n)$ *é a premissa primeira de uma regra* $EQ2$.
  3. φ_n *é a premissa primeira de uma regra* $EQ2$ *cuja conclusão não é premissa primeira de uma regra* $EQ2$.

Obs.: Na seção seguinte esclareceremos melhor a idéia de segmento máximo equacional.

Definição 6.6 *Duas ocorrências de fórmulas são ditas da mesma forma, se elas são ocorrências da mesma fórmula.*

Definição 6.7 (Grau de um Segmento) *O grau de um segmento (denotado por* $d(\alpha)$*) é o grau de uma fórmula que ocorre em* α *(repare que mesmo em um segmento equacional, todas as fórmulas possuem o mesmo grau).*

Definição 6.8 (Comprimento de um Segmento) *O comprimento de um segmento α (denotado por $l(\alpha)$) é o número de ocorrências de fórmulas em α.*

Definiremos agora, como originária desta dissertação, a função χ que intuitivamente corrige o fato de que, no sistema dedutivo que apresentamos, a premissa e a conclusão de uma regra de igualdade não podem ser iguais (sempre ocorre uma substituição).

Definição 6.9 *A função χ é definida como:*

$$\chi\left(\dfrac{\begin{matrix}\Sigma_1 & \Sigma_2\\ \varphi & t\equiv t'\end{matrix}}{\psi}\right)=\begin{cases}\begin{matrix}\Sigma_1\\ \varphi\end{matrix} & \text{, se } \varphi=\psi\\[2ex] \dfrac{\begin{matrix}\Sigma_1 & \Sigma_2\\ \varphi & t\equiv t'\end{matrix}}{\psi} & \text{, caso contrário.}\end{cases}$$

Observe que a função trivial χ, de acordo com o sistema dedutivo que apresentamos (capitulo 2), pode receber como parâmetro uma derivação inválida (caso $\psi=\varphi$). Seu uso neste capítulo estará justamente em corrigir derivações inválidas retornando derivações corretas de acordo com as regras de reduções que nos forem apresentadas.

Definiremos também, como originária desta dissertação, a função α, que tem por objetivo obter a regra de simetria da igualdade.

Definição 6.10 *A função α é definida como:*

$$\left(\begin{matrix}\Sigma\\ t\equiv t'\end{matrix}\right)^{\alpha}=\begin{cases}\begin{matrix}\Sigma_1\\ t'\equiv t\end{matrix} & \text{Se } \Sigma \text{ tem a forma: } \dfrac{\dfrac{\dfrac{\neg t\equiv t' \quad \begin{matrix}\Sigma_1\\ t'\equiv t\end{matrix}}{\neg t\equiv t} \quad \dfrac{}{t\equiv t}}{\perp}}{t\equiv t'}\\[6ex] \dfrac{\dfrac{\dfrac{\neg t'\equiv t \quad \begin{matrix}\Sigma\\ t\equiv t'\end{matrix}}{\neg t'\equiv t'} \quad \dfrac{}{t'\equiv t'}}{\perp}}{t'\equiv t} & \text{Caso contrário.}\end{cases}$$

Observe que $\begin{matrix}\Sigma\\ t\equiv t'\end{matrix} \equiv_d \left(\left(\begin{matrix}\Sigma\\ t\equiv t'\end{matrix}\right)^{\alpha}\right)^{\alpha}$.

Obs.: Abreviaremos a notação $\left(\begin{matrix}\Sigma\\ t\equiv t'\end{matrix}\right)^{\alpha}$ para $\begin{matrix}\Sigma^{\alpha}\\ t'\equiv t\end{matrix}$.

Também propomos aqui, como adição própria o seguinte teorema:

Teorema 6.1 *Dado o uso de uma regra de igualdade*

$$\frac{\begin{array}{cc}\Pi_1 & \Pi_2\\ \varphi^{t}_{z} & t \equiv t'\end{array}}{\varphi^{t'}_{z}}\ ,$$

temos que a derivação (invertendo a ordem de premissa e conclusão)

$$\frac{\begin{array}{cc}\Pi_3 & \Pi_2^{\alpha}\\ \varphi^{t'}_{z} & t' \equiv t\end{array}}{\varphi^{t}_{z}}\ ,$$

também apresenta um uso correto da regra de igualdade.

Prova: É fácil verificar isto. □

Adotaremos também, em nossa dissertação, a seguinte convenção:

Em sequências da regra de igualdade, iremos denotar as fórmulas das premissas primeiras por $\xi_1, \xi_2, \ldots, \xi_n$. Como exemplo:

$$\frac{\begin{array}{c}\dfrac{\begin{array}{cc} & \Pi_1\\ [\xi_1] & t_1 \equiv t_1'\end{array}}{\xi_2}\\ \vdots\\ \xi_{n-1}\end{array} \qquad \begin{array}{c}\Pi_{n-1}\\ t_{n-1} \equiv t_{n-1}'\end{array}}{\xi_n}\ .$$

Note que dada a derivação do exemplo anterior, temos que

$$\frac{\begin{array}{c}\dfrac{\begin{array}{cc} & \Pi_{n-1}^{\alpha}\\ [\xi_n] & t_{n-1}' \equiv t_{n-1}\end{array}}{\xi_{n-1}}\\ \vdots\\ \xi_2\end{array} \qquad \begin{array}{c}\Pi_1^{\alpha}\\ t_1' \equiv t_1\end{array}}{\xi_1}$$

também é uma derivação, da mesma forma que

$$\dfrac{\dfrac{[\neg\xi_n] \quad \begin{matrix}\Pi^{\alpha}_{n-1} \\ t'_{n-1} \equiv t_{n-1}\end{matrix}}{\begin{matrix}\neg\xi_{n-1} \\ \vdots \\ \neg\xi_2\end{matrix}} \qquad \begin{matrix}\Pi^{\alpha}_{1} \\ t'_1 \equiv t_1\end{matrix}}{\neg\xi_1}$$

também é uma derivação correta.

Definição 6.11 *Dada uma derivação*

$$\begin{matrix} \Sigma \\ \dfrac{\varphi(y)}{\forall x\varphi(x)} \\ \vdots \\ \Pi \end{matrix}$$

Dizemos que y é a variável de aplicação da regra $\forall I$ em questão.
Da mesma forma temos que dada uma derivação

$$\dfrac{\begin{matrix}\Sigma_1 \\ \exists x\varphi(x)\end{matrix} \qquad \begin{matrix}\varphi(y) \\ \Sigma_2 \\ \psi\end{matrix}}{\Pi},$$

dizemos que y é a variável de aplicação da regra $\exists E$ em questão.

Com a definição anterior em mente, só permitiremos que cada variável de aplicação de cada regra $\forall I$ em uma derivação Π ocorra somente acima das premissas da regra em questão, da mesma forma, no caso de $\exists E$ permitiremos que estas variáveis só ocorram acima da premissa menor de cada regra em questão. Tal restrição é semelhante ao conceito de parâmetros puros de [17].

6.3 Segmentos Máximos

Intuitivamente, consideramos como segmento máximo a introdução de uma fórmula seguida por sua eliminação. Abordaremos dois tipos de segmentos máximos: os segmentos máximos clássicos e os segmentos máximos equacionais. Estes últimos seguem da idéia que uma fórmula é redudante se ela é conclusão de uma regra de introdução, depois algum termo dela é trocado através da regra de igualdade e, em seguida, a fórmula resultante é premissa maior de uma regra de eliminação. Como exemplo temos:

$$\dfrac{\dfrac{\begin{matrix}\Sigma_1 & \Sigma_2\\ P(x) & Q(x)\end{matrix}}{P(x)\wedge Q(x)} \quad \begin{matrix}\Sigma_3\\ x\equiv y\end{matrix}}{\dfrac{P(y)\wedge Q(y)}{P(y)}}$$

Notamos que este tipo de redudância não é capturada pelo conceito de segmento máximo clássico, sendo necessário a criação de um novo conceito de segmento máximo que lida com estas fórmulas. Do exemplo anterior poderiamos eliminar tal redundância "subindo" a aplicação da regra de igualdade como a seguir:

$$\dfrac{\dfrac{\dfrac{\begin{matrix}\Sigma_1\\ P(x)\end{matrix} \quad \begin{matrix}\Sigma_3\\ x\equiv y\end{matrix}}{P(y)} \quad \dfrac{\begin{matrix}\Sigma_2\\ Q(x)\end{matrix} \quad \begin{matrix}\Sigma_3\\ x\equiv y\end{matrix}}{Q(y)}}{P(y)\wedge Q(y)}}{P(y)}$$

Agora podemos aplicar uma regra de eliminação de segmento máximo clássico. Observe que a redundância que era oculta pela regra de igualdade agora já pode ser eliminada.

Apresentaremos esses dois conceitos nas duas próximas subseções:

6.3.1 Segmento Máximo Clássico

Iremos agora apresentar a noção de segmento máximo clássico que iremos adotar, proposta em [2]: Um segmento $\sigma = \varphi_1, \varphi_2, \ldots, \varphi_n$, é um segmento máximo clássico quando:

(a) $n = 1$

1. $\varphi_1 (= \varphi_n)$ é a conclusão de uma regra de introdução ou então da regra do absurdo intuicionista e é também uma premissa maior de uma regra de eliminação;
2. $\varphi_1 (= \varphi_n)$ é a conclusão da regra do absurdo clássico e é também uma premissa maior de uma regra de eliminação;
3. $\varphi_1 (= \varphi_n)$ é a conclusão da regra do absurdo clássico e é também uma premissa menor da regra $(\rightarrow E)$, tal que sua premissa maior é uma hipótese da forma $(\varphi_1 \rightarrow \perp)$;

(b) $n > 1$

1. φ_n é a conclusão da regra $(\vee E)$ ou $(\exists E)$ e é também uma premissa maior de uma regra de eliminação;

2. φ_n é a conclusão da regra $(\vee E)$ ou $(\exists E)$ e é também uma premissa menor de uma aplicação da regra $(\rightarrow E)$, tal que sua premissa maior é uma hipótese da forma $(\varphi_n \rightarrow \perp)$.

Nesta dissertação diferenciaremos dois tipos de graus: o grau clássico, determinado por segmentos máximos clássicos e o grau equacional, determinado por segmento máximos equacionais (a seguir).

Definição 6.12 *O grau clássico de uma derivação* Π *será denotado por* $d_c(\Pi)$ *e definido como:* $d_c(\Pi) = max\{d_c(\sigma)\ :\ \sigma$ *é um segmento máximo do tipo* $(a.1)$*,* $(a.2)$ *ou* $(b.1)\}$*.*

Certamente se Π não contém segmentos máximos e/ou não contém segmentos máximos do tipo $(a.1)$, $(a.2)$ ou $(b.1)$, então $d_c(\Pi) = 0$.

6.3.2 Segmento Máximo Equacional

Introduziremos agora, como contribuição nossa, o conceito de segmento máximo equacional, que, como haviamos explicado, lida com as redundâncias ocultadas pelo uso da regra de igualdade.

Um segmento $\sigma = \varphi_1, \varphi_2, \ldots, \varphi_n$ é um segmento máximo equacional quando:

(α)

1. φ_1 é a conclusão de uma regra de introdução ou da regra do absurdo intucionista e é também premissa primeira da regra $EQ2$.

2. φ_1 é a conclusão da regra do absurdo clássico e é também premissa primeira da regra $EQ2$.

3. φ_1 $(n = 1)$ é a conclusão da regra do absurdo clássico e é também premissa menor da regra $(\rightarrow E)$, tal que sua premissa maior é da forma $(\varphi_1 \rightarrow \perp)$ do seguinte modo:

$$\dfrac{\begin{matrix}\Pi\\ \varphi_1\end{matrix} \qquad \dfrac{\begin{matrix}\dfrac{\neg\xi_1 \quad \begin{matrix}\Pi_1\\ t_1\equiv t_1^{'}\end{matrix}}{\neg\xi_2}\\ \vdots\\ \neg\xi_k\end{matrix} \qquad \begin{matrix}\Pi_k\\ t_k\equiv t_k^{'}\end{matrix}}{\neg\varphi_1}}{\perp}$$

(β) 1. φ_n (n=1) é a conclusão da regra ($\vee E$) ou ($\exists E$) e é também premissa primeira da regra $EQ2$.

2. φ_n (n=1) é a conclusão da regra ($\vee E$) ou ($\exists E$) e é também premissa menor de aplicação da regra ($\rightarrow E$), tal que sua premissa maior é da forma ($\varphi_n \rightarrow\perp$) da seguinte forma:

$$\dfrac{\dfrac{\begin{matrix}\Pi\\ (\sigma\vee\tau)\end{matrix} \quad \begin{matrix}\Pi_1^{'}\\ \varphi_n\end{matrix} \quad \begin{matrix}\Pi_2^{''}\\ \varphi_n\end{matrix}}{\varphi_n} \qquad \dfrac{\begin{matrix}\dfrac{\neg\xi_1 \quad \begin{matrix}\Pi_1\\ t_1\equiv t_1^{'}\end{matrix}}{\neg\xi_2}\\ \vdots\\ \neg\xi_k\end{matrix} \qquad \begin{matrix}\Pi_k\\ t_k\equiv t_k^{'}\end{matrix}}{\neg\varphi_n}}{\perp}$$

no caso em que φ_n é a conclusão da regra ($\vee E$) ou

$$\dfrac{\dfrac{\begin{matrix}\Pi\\ \exists x\psi\end{matrix} \quad \begin{matrix}\Pi'\\ \varphi_n\end{matrix}}{\varphi_n} \qquad \dfrac{\begin{matrix}\dfrac{\neg\xi_1 \quad \begin{matrix}\Pi_1\\ t_1\equiv t_1^{'}\end{matrix}}{\neg\xi_2}\\ \vdots\\ \neg\xi_k\end{matrix} \qquad \begin{matrix}\Pi_k\\ t_k\equiv t_k^{'}\end{matrix}}{\neg\varphi_n}}{\perp}$$

no caso em que φ_n é a conclusão da regra ($\exists E$).

Da mesma forma que definimos o grau de um segmento máximo clássico, iremos agora definir o grau de um segmento equacional:

Definição 6.13 *O grau equacional de uma derivação será denotado por $d_e(\Pi)$ e definido como:*

$$d_e(\Pi) = max\{d_e(\Pi)\ :\ \Pi \textit{ é um segmento máximo equacional de tipo } (\alpha.1), (\alpha.2) \textit{ ou } (\beta.1)\}$$

Observe que para os segmentos máximos clássicos usaremos as letras de nosso alfabeto (a e b), já para segmento máximos equacionais utilizaremos letras gregas (α e β).

Da mesma forma anterior, se Π não contém segmentos máximos equacionais e/ou se não contém segmentos máximos equacionais do tipo (α.1), (α.2) ou (β.1), então $d_e(\Pi) = 0$.

Porém temos a necessidade de unificar o que será o grau de uma derivação como um todo, fazemos isso da seguinte forma:

Definição 6.14 *O grau de uma derivação Π é definido como:*

$$d(\Pi) = max\{d_c(\Pi), d_e(\Pi)\}.$$

Com base nisto, podemos definir nosso conceito de quando uma derivação é normal:

Definição 6.15 (Derivação Normal) *Uma derivação Π é chamada de normal se e somente se Π não contém segmentos máximos (tanto do tipo clássico como os de igualdade).*

6.4 Reduções

Veremos nesta seção como remover os dois tipos de segmentos máximos de uma derivação, estabelecendo reduções para cada tipo de segmento máximo.

Podemos dividir as reduçoes em: reduções clássicas e reduções de igualdade.

Ambas as reduções são subdivididas em três tipos: reduções operacionais, reduções permutativas e reduções do absurdo.

6.4.1 Reduções Clássicas

As reduções clássicas são como as encontradas em MASSI [2] e PRAWITZ [17], com exceção da regra do absurdo. Elas são divididas em reduções clássicas operacionais e permutativas.

Reduções Clássicas Operacionais

1. $\wedge$ - redução

$$\dfrac{\dfrac{\begin{array}{cc}\Pi_1 & \Pi_2\\ \varphi_1 & \varphi_2\end{array}}{\varphi_1\wedge\varphi_2}}{\varphi_1} \quad \text{por} \quad \begin{array}{c}\Pi_i\\ \varphi_i\end{array} \quad (i=1,2)$$

2. $\vee$ - redução

$$\dfrac{\dfrac{\begin{array}{c}\Pi\\ \varphi_i\end{array}}{\varphi_1\vee\varphi_2} \quad \begin{array}{c}\varphi_1\\ \Pi_1\\ \psi\end{array} \quad \begin{array}{c}\varphi_2\\ \Pi_2\\ \psi\end{array}}{\psi} \quad \text{por} \quad \begin{array}{c}\Pi\\ \varphi_i\\ \Pi_i\\ \psi\end{array} \quad (i=1,2)$$

3. $\rightarrow$ - redução

$$\dfrac{\begin{array}{c}\Pi_1\\ \varphi\end{array} \quad \dfrac{\begin{array}{c}\varphi\\ \Pi_2\\ \psi\end{array}}{\varphi\rightarrow\psi}}{\psi} \quad \text{por} \quad \begin{array}{c}\Pi_1\\ \varphi\\ \Pi_2\\ \psi\end{array}$$

4. $\forall$ - redução

$$\dfrac{\dfrac{\begin{array}{c}\Pi\\ \varphi(y)\end{array}}{\forall x\varphi(x)}}{\varphi(t)} \quad \text{por} \quad \begin{array}{c}\Pi^t_y\\ \varphi(t)\end{array}$$

5. $\exists$ - redução

$$\dfrac{\dfrac{\begin{array}{c}\Pi_1\\ \varphi(t)\end{array}}{\exists x\varphi(x)} \quad \begin{array}{c}[\varphi(y)]\\ \Pi\\ \psi\end{array}}{\psi} \quad \text{por} \quad \begin{array}{c}\Pi_1\\ \varphi(t)\\ \Pi^t_y\\ \psi\end{array}$$

Obs.: Repare que as reduções clássicas operacionais não contribuem para o aumento do grau da derivação em questão.

Reduções Clássicas Permutativas

1. $\vee E$ - redução

$$\dfrac{\dfrac{\dfrac{\Pi_1}{\varphi_1\vee\varphi_2} \quad \begin{array}{c}\Pi_2\\ \psi\end{array} \quad \begin{array}{c}\Pi_3\\ \psi\end{array}}{\psi} \quad \Pi_4}{\sigma} \quad \text{por} \quad \dfrac{\begin{array}{c}\Pi_1\\ \varphi_1\vee\varphi_2\end{array} \quad \dfrac{\begin{array}{c}\Pi_2\\ \psi\end{array} \quad \Pi_4}{\sigma} \quad \dfrac{\begin{array}{c}\Pi_3\\ \psi\end{array} \quad \Pi_4}{\sigma}}{\sigma}$$

Onde ψ é a premissa maior de uma regra de eliminação e Π_4 pode ocorrer a esquerda.

2. $\exists E$ - redução

$$\dfrac{\dfrac{\begin{matrix}\Pi_1\\ \exists x\varphi(x)\end{matrix} \quad \begin{matrix}\Pi_2\\ \psi\end{matrix}}{\psi} \quad \Pi_3}{\sigma} \quad \text{por} \quad \dfrac{\begin{matrix}\Pi_1\\ \exists x\varphi(x)\end{matrix} \quad \dfrac{\begin{matrix}\Pi_2\\ \psi\end{matrix} \quad \Pi_3}{\sigma}}{\sigma}$$

Onde ψ é a premissa maior de uma regra de eliminação e Π_3 pode ocorrer a esquerda.

Reduções Clássicas do Absurdo

1.

$$\dfrac{\dfrac{\begin{matrix}[\neg\varphi]^i\\ \Pi\\ \bot\end{matrix}}{\varphi}\,{}^{,i} \quad [\neg\varphi]^k}{\bot} \quad \text{por} \quad \begin{matrix}[\neg\varphi]^i\\ \Pi\\ \bot\end{matrix}$$

2.

$$\dfrac{\dfrac{\begin{matrix}\Pi\\ \exists x\psi(x)\end{matrix} \quad \begin{matrix}\Pi'\\ \varphi\end{matrix}}{\varphi} \quad [\neg\varphi]}{\bot} \quad \text{por} \quad \dfrac{\begin{matrix}\Pi\\ \exists x\psi(x)\end{matrix} \quad \dfrac{\begin{matrix}\Pi'\\ \varphi\end{matrix} \quad [\neg\varphi]}{\bot}}{\bot}$$

3.

$$\dfrac{\dfrac{\begin{matrix}\Pi_1\\ \psi_1 \vee \psi_2\end{matrix} \quad \begin{matrix}\Pi_2\\ \varphi\end{matrix} \quad \begin{matrix}\Pi_3\\ \varphi\end{matrix}}{\varphi} \quad [\neg\varphi]}{\bot} \quad \text{por} \quad \dfrac{\begin{matrix}\Pi_1\\ \psi_1 \vee \psi_2\end{matrix} \quad \dfrac{\begin{matrix}\Pi_2\\ \varphi\end{matrix} \quad [\neg\varphi]}{\bot} \quad \dfrac{\begin{matrix}\Pi_3\\ \varphi\end{matrix} \quad [\neg\varphi]}{\bot}}{\bot}$$

4.

$$\dfrac{\dfrac{\begin{matrix}[\neg\psi]^k\\ \Pi\\ \bot\end{matrix}}{\psi}\,{}^{,k} \quad \Sigma}{\varphi} \quad \text{por} \quad \dfrac{\begin{matrix}\dfrac{\dfrac{\dfrac{[\psi]^k \quad \Sigma}{\varphi} \quad [\neg\varphi]^i}{\bot}}{\neg\psi}\,{}^{,k}\\ \Pi^0\\ \bot\end{matrix}}{\varphi}\,{}^{,i}$$

onde:

- ψ é premissa maior de uma regra de eliminação;
- Σ pode ocorrer à esquerda de ψ;
- Π^0 é obtido de Π, substituindo todas as partes da forma

$$\dfrac{\begin{matrix}\Pi_2\\ \psi\end{matrix} \quad [\neg\psi]^k}{\bot} \quad \text{por} \quad \dfrac{\dfrac{\begin{matrix}\Pi_2\\ \psi\end{matrix} \quad \Sigma}{\varphi} \quad [\neg\varphi]^i}{\bot}$$

e

$$\dfrac{\begin{matrix}\Pi_3\\ \sigma\end{matrix} \qquad \dfrac{\begin{matrix}\dfrac{[\neg\xi_1]^k \quad \begin{matrix}\Sigma_1\\ t_1 \equiv t_1'\end{matrix}}{\neg\xi_2}\\ \vdots\\ \neg\xi_k\end{matrix} \qquad \begin{matrix}\Sigma_k\\ t_k \equiv t_k'\end{matrix}}{\neg\sigma}}{\bot} \qquad \text{por} \qquad \dfrac{\dfrac{\dfrac{\begin{matrix}\dfrac{\begin{matrix}\Pi_3\\ \sigma\end{matrix} \quad \begin{matrix}{\Sigma_k}^{\alpha}\\ t_k' \equiv t_k\end{matrix}}{\xi_k}\\ \vdots\\ \xi_2\end{matrix} \qquad \begin{matrix}\Sigma_1^{\alpha}\\ t_1' \equiv t_1\end{matrix}}{\xi_1} \qquad \Sigma}{\varphi} \qquad [\neg\varphi]^i}{\bot}$$

Onde $[\xi_1]^k = [\psi]^k$.

A primeira (última) substituição evita o surgimento de um segmento máximo clássico (equacional) de grau maior que ψ (ambos podem surgir com o uso da regra $(\rightarrow I)$).

6.4.2 Reduções Equacionais

As reduções equacionais que propomos nesta dissertação eliminam segmentos máximos equacionais tornando evidente segmentos máximos clássicos ocultados pela regra da igualdade. Elas são divididas, assim como as reduções clássicas, em três subtipos: Operacionais, Permutativas e do Absurdo.

Reduções Equacionais Operacionais

1. $\wedge I$:

$$\dfrac{\dfrac{\begin{matrix}\Pi_1\\ \varphi\frac{t}{z}\end{matrix} \quad \begin{matrix}\Pi_2\\ \psi\frac{t}{z}\end{matrix}}{(\varphi \wedge \psi)\frac{t}{z}} \quad \begin{matrix}\Pi_3\\ t \equiv t'\end{matrix}}{(\varphi \wedge \psi)\frac{t'}{z}} \quad \text{por} \quad \dfrac{\chi\left(\dfrac{\begin{matrix}\Pi_1\\ \varphi\frac{t}{z}\end{matrix} \quad \begin{matrix}\Pi_3\\ t \equiv t'\end{matrix}}{\varphi\frac{t'}{z}}\right) \quad \chi\left(\dfrac{\begin{matrix}\Pi_2\\ \psi\frac{t}{z}\end{matrix} \quad \begin{matrix}\Pi_3\\ t \equiv t'\end{matrix}}{\psi\frac{t'}{z}}\right)}{(\varphi \wedge \psi)\frac{t'}{z}}$$

2. $\vee I$:

$$\dfrac{\dfrac{\begin{matrix}\Pi_i\\ \varphi_i\frac{t}{z}\end{matrix}}{(\varphi_1 \vee \varphi_2)\frac{t}{z}} \quad \begin{matrix}\Pi_3\\ t \equiv t'\end{matrix}}{(\varphi_1 \vee \varphi_2)\frac{t'}{z}} \quad \text{por} \quad \dfrac{\chi\left(\dfrac{\begin{matrix}\Pi_i\\ \varphi_i\frac{t}{z}\end{matrix} \quad \begin{matrix}\Pi_3\\ t \equiv t'\end{matrix}}{\varphi_i\frac{t'}{z}}\right)}{(\varphi_1 \vee \varphi_2)\frac{t'}{z}} \quad \text{, onde } i \in \{1,2\}$$

3. $\to I$:

$$\dfrac{\dfrac{\begin{array}{c}[\varphi\frac{t}{z}]\\ \Pi_1\\ \psi\frac{t}{z}\end{array}}{(\varphi\to\psi)\frac{t}{z}} \quad \begin{array}{c}\Pi_2\\ t\equiv t'\end{array}}{(\varphi\to\psi)\frac{t'}{z}} \quad \text{por} \quad \dfrac{\chi\left(\dfrac{\begin{array}{c}\chi\left(\dfrac{\varphi\frac{t'}{z}\quad \begin{array}{c}\Pi_2{}^{\alpha}\\ t'\equiv t\end{array}}{[\varphi\frac{t}{z}]}\right)\\ \Pi_1\\ \psi\frac{t}{z}\end{array}\qquad \begin{array}{c}\Pi_2{}^{\alpha}\\ t\equiv t'\end{array}}{\psi\frac{t'}{z}}\right)}{(\varphi\to\psi)\frac{t'}{z}}$$

Onde $\psi \neq \perp$

$$\dfrac{\dfrac{\begin{array}{c}[\varphi\frac{t}{z}]\\ \Pi_1\\ \perp\end{array}}{\neg\varphi\frac{t}{z}} \quad \begin{array}{c}\Pi_2\\ t\equiv t'\end{array}}{\neg\varphi\frac{t'}{z}} \quad \text{por} \quad \dfrac{\begin{array}{c}\dfrac{\varphi\frac{t'}{z}\quad \begin{array}{c}\Pi_2\\ t'\equiv t\end{array}}{[\varphi\frac{t}{z}]}\\ \Pi_1\\ \perp\end{array}}{\neg\varphi\frac{t'}{z}}$$

Caso $\psi = \perp$

4. $\perp_c$:

$$\dfrac{\dfrac{\begin{array}{c}[\neg\varphi\frac{t}{z}]\\ \Pi_1\\ \perp\end{array}}{\varphi\frac{t}{z}} \quad \begin{array}{c}\Pi_2\\ t\equiv t'\end{array}}{\varphi\frac{t'}{z}} \quad \text{por} \quad \dfrac{\begin{array}{c}\dfrac{\neg\varphi\frac{t'}{z}\quad \begin{array}{c}\Pi_2\\ t'\equiv t\end{array}}{[\neg\varphi\frac{t}{z}]}\\ \Pi_1\\ \perp\end{array}}{\varphi\frac{t'}{z}}$$

5. $\forall I$:

$$\dfrac{\dfrac{\begin{array}{c}\Pi_1\\ \varphi\frac{t}{z}\end{array}}{\forall x\varphi\frac{t}{z}\frac{x}{y}} \quad \begin{array}{c}\Pi_2\\ t\equiv t'\end{array}}{\forall x\varphi\frac{t'}{z}\frac{x}{y}} \quad \text{por} \quad \dfrac{\dfrac{\begin{array}{c}\Pi_1\\ \varphi\frac{t}{z}\end{array}\quad \begin{array}{c}\Pi_2\\ t\equiv t'\end{array}}{\varphi\frac{t'}{z}}}{\forall x\varphi\frac{t'}{z}\frac{x}{y}}$$

6. $\exists I$:

$$\dfrac{\dfrac{\begin{array}{c}\Pi_1\\ \varphi\frac{t}{z}\frac{t''}{x}\end{array}}{\exists x\varphi\frac{t}{z}} \quad \begin{array}{c}\Pi_2\\ t\equiv t'\end{array}}{\exists x\varphi\frac{t'}{z}} \quad \text{por} \quad \dfrac{\dfrac{\begin{array}{c}\Pi_1\\ \varphi\frac{t}{z}\frac{t''}{x}\end{array}\quad \begin{array}{c}\Pi_2\\ t\equiv t'\end{array}}{\varphi\frac{t'}{z}\frac{t''}{x}}}{\exists x\varphi\frac{t'}{z}}$$

Obs.: Repare que as reduções equacionais operacionais não contribuem para o aumento do grau da derivação em questão.

Reduções Equacionais Permutativas

1.

$$\dfrac{\dfrac{\begin{matrix}\Pi_1\\ \varphi_1\vee\varphi_2\end{matrix} \quad \begin{matrix}[\varphi_1]\\ \Pi_2\\ \psi^t_z\end{matrix} \quad \begin{matrix}[\varphi_2]\\ \Pi_3\\ \psi^t_z\end{matrix}}{\psi^t_z} \quad \begin{matrix}\Pi_4\\ t\equiv t'\end{matrix}}{\psi^{t'}_z} \quad \text{por} \quad \dfrac{\begin{matrix}\Pi_1\\ \varphi_1\vee\varphi_2\end{matrix} \quad \dfrac{\begin{matrix}[\varphi_1]\\ \Pi_2\\ \psi^t_z\end{matrix} \quad \begin{matrix}\Pi_4\\ t\equiv t'\end{matrix}}{\psi^{t'}_z} \quad \dfrac{\begin{matrix}[\varphi_2]\\ \Pi_3\\ \psi^t_z\end{matrix} \quad \begin{matrix}\Pi_4\\ t\equiv t'\end{matrix}}{\psi^{t'}_z}}{\psi^{t'}_z}$$

2.

$$\dfrac{\dfrac{\begin{matrix}\Pi_1\\ \exists x\varphi(x)\end{matrix} \quad \begin{matrix}[\varphi(y)]\\ \Pi_2\\ \psi^t_z\end{matrix}}{\psi^t_z} \quad \begin{matrix}\Pi_3\\ t\equiv t'\end{matrix}}{\psi^{t'}_z} \quad \text{por} \quad \dfrac{\begin{matrix}\Pi_1\\ \exists x\varphi(x)\end{matrix} \quad \dfrac{\begin{matrix}[\varphi(y)]\\ \Pi_2\\ \psi^t_z\end{matrix} \quad \begin{matrix}\Pi_3\\ t\equiv t'\end{matrix}}{\psi^{t'}_z}}{\psi^{t'}_z}$$

Reduções Equacionais do Absurdo

1.

$$\dfrac{\dfrac{\begin{matrix}[\neg\varphi]^i\\ \Pi\\ \bot\end{matrix}}{\varphi}\,{}_{,i} \quad \dfrac{\begin{matrix}\dfrac{[\neg\xi_1] \quad \begin{matrix}\Pi_1\\ t_1\equiv t_1'\end{matrix}}{\neg\xi_2}\\ \vdots\\ \neg\xi_k\end{matrix} \quad \begin{matrix}\Pi_k\\ t_k\equiv t_k'\end{matrix}}{\neg\varphi}}{\bot} \quad \text{por} \quad \begin{matrix}\dfrac{\begin{matrix}\dfrac{[\neg\xi_1] \quad \begin{matrix}\Pi_1\\ t_1\equiv t_1'\end{matrix}}{\neg\xi_2}\\ \vdots\\ \neg\xi_k\end{matrix} \quad \begin{matrix}\Pi_k\\ t_k\equiv t_k'\end{matrix}}{[\neg\varphi]^i}\\ \Pi\\ \bot\end{matrix}$$

2.

$$\dfrac{\dfrac{\begin{matrix}\Pi\\ \exists x\psi(x)\end{matrix} \quad \begin{matrix}\Pi'\\ \varphi\end{matrix}}{\varphi} \quad \dfrac{\begin{matrix}\dfrac{\neg\xi_1 \quad \begin{matrix}\Pi_k\\ t_1\equiv t_1'\end{matrix}}{\neg\xi_2}\\ \vdots\\ \neg\xi_1\end{matrix} \quad \begin{matrix}\Pi_k\\ t_k\equiv t_k'\end{matrix}}{\neg\varphi}}{\bot} \quad \text{por} \quad \dfrac{\begin{matrix}\Pi\\ \exists x\psi(x)\end{matrix} \quad \dfrac{\begin{matrix}\Pi'\\ \varphi\end{matrix} \quad \dfrac{\begin{matrix}\dfrac{\neg\xi_1 \quad \begin{matrix}\Pi_1\\ t_1\equiv t_1'\end{matrix}}{\neg\xi_2}\\ \vdots\\ \neg\xi_k\end{matrix} \quad \begin{matrix}\Pi_k\\ t_k\equiv t_k'\end{matrix}}{\neg\varphi}}{\bot}}{\bot}$$

3.

$$\dfrac{\dfrac{\begin{matrix}\Pi_1\\ \psi_1\vee\psi_2\end{matrix} \quad \begin{matrix}\Pi_2\\ \varphi\end{matrix} \quad \begin{matrix}\Pi_3\\ \varphi\end{matrix}}{\varphi} \quad \Sigma}{\bot} \quad \text{por} \quad \dfrac{\begin{matrix}\Pi_1\\ \psi_1\vee\psi_2\end{matrix} \quad \dfrac{\begin{matrix}\Pi_2\\ \varphi\end{matrix} \quad \Sigma}{\bot} \quad \dfrac{\begin{matrix}\Pi_3\\ \varphi\end{matrix} \quad \Sigma}{\bot}}{\bot}$$

$$\text{onde } \Sigma = \dfrac{\begin{array}{c}\dfrac{\neg\xi_1 \quad \begin{array}{c}\Sigma_1\\ t_1 \equiv t_1^{'}\end{array}}{\neg\xi_2}\\ \vdots \\ \neg\xi_k\end{array} \qquad \begin{array}{c}\Sigma_k\\ t_k \equiv t_k^{'}\end{array}}{\neg\varphi}$$

6.5 Redutibilidade

Com as reduções apresentadas na seção anterior, podemos falar em obter uma derivação normal a partir de uma sequência de reduções.

Definição 6.16 (Redução Imediata) *Dada uma derivação Π e Π', dizemos que Π' é uma redução imediata de Π se Π' é obtida aplicando somente uma das reduções apresentadas na seção anterior.*

O que nos possibilita definir:

Definição 6.17 (Sequência de Redução) *Dizemos que $\Pi_1, \ldots, \Pi_n$ é uma seqûência de redução se e somente se Π_{i+1} é uma redução imediata de Π_i (para $0 \leq i < n$).*

Definição 6.18 *Dizemos que uma derivação Π se reduz a Π^* (denotaremos por $\Pi \rightarrow \Pi^*$) se existe uma sequência de redução $\Pi_1, \ldots, \Pi_n$, onde $n \geq 1$, tal que $\Pi_1 = \Pi$ e $\Pi = \Pi^*$.*

e, finalmente, podemos falar em obter normalização com:

Definição 6.19 (Derivação Normalizável) *Dizemos que uma derivação Π é normalizável se existe uma sequência $\Pi_1, \ldots, \Pi_n$, tal que $\Pi_1 = \Pi$ e Π_n é uma derivação normal.*

6.6 Normalização Fraca para FOL + Regras da Igualdade

Nesta seção iremos provar o teorema de normalização fraca para a lógica de primeira ordem com a igualdade. Tal prova serve como base para a prova do teorema de normalização fraca para LFP_{fin} com restrições.

Lema 6.1 *Dada uma derivação* Π *tal que* $\Gamma \vdash \varphi$ *e* $d(\Pi) = n$*, de forma que todo segmento máximo de grau* n *que contribui para o grau de* Π *é do tipo* $(a.1)$*. Então,* Π *se reduz a uma derivação* Π' *de* $\Delta \subseteq \Gamma \vdash \varphi$*, tal que* $d(\Pi') < d(\Pi)$*.*

Prova: Similar a [17]. $\square$

Lema 6.2 *Dada uma derivação* Π *tal que* $d(\Pi) = 0$*, então* Π *se reduz a uma derivação normal* Π'*.*

Prova: Se Π é normal, então o lema é válido com $\Pi' \equiv_d \Pi$.

Se Π não é normal, então basta estabelecermos como valor de indução:

$$k = \text{a soma dos comprimentos dos segmentos máximos de } \Pi.$$

Basta escolhermos, na derivação Π, um segmento máximo α (tanto clássico como equacional), tal que não exista outro segmento máximo acima dele, nem que contenha (ou ocorra acima de) uma fórmula que ocorre ao lado da última fórmula de α. Podermos ver facilmente que este segmento é um segmento máximo do absurdo (podendo ser clássico ou equacional).

Considere Π_1 uma redução de Π que elimina este segmento máximo. Vemos então, a partir das reduções do absurdo(tanto clássicas quanto equacionais), que o valor de indução de Π_1 é menor que o valor de indução de Π. O resultado então é imediato. $\square$

O corolário a seguir nos fala sobre a conservação da regra introdutória de um segmento do tipo $\alpha.1$ ou $\alpha.2$ sob a redução aplicada nele.

Corolário 6.1 *Seja* Π *uma derivação de* $\Gamma \vdash \varphi$ *onde* $d_e(\Pi) = k$*, tal que* $r(\Pi)$ *é a regra EQ2, cuja premissa primeira* ψ *é da forma:*

- ψ *é a última ocorrência de fórmula de todos segmentos máximos equacionais que contribui para* $d_e(\Pi) = k$ *e é do tipo* $\alpha.1$ *ou* $\alpha.2$*.*

Então Π *se reduz a uma derivação* Π' *de* $\Delta \subseteq \Gamma \vdash \varphi$ *onde* $d(\Pi') \leq d(\Pi)$ *e* $d_e(\Pi') < d_e(\Pi)$

tal que:

$$
Se\ \Pi = \cfrac{\cfrac{\cfrac{\begin{array}{cc}\Sigma & \Pi_1 \\ \xi_1 & t_1 \equiv t_1^{'}\end{array}}{\xi_2} \\ \vdots \\ \xi_{n-1} \qquad \begin{array}{c}\Pi_{n-1} \\ t_{n-1} \equiv t_{n-1}^{'}\end{array}}{\psi} \qquad \begin{array}{c}\Pi_n \\ t_n \equiv t_n^{'}\end{array}}{\varphi}
$$

então $r\begin{pmatrix}\Sigma \\ \xi_1\end{pmatrix} = r(\Pi')$.

Prova: Basta aplicarmos um número suficiente de vezes as reduções para as regras em $\alpha.1$ ou em $\alpha.2$. Note que cada redução destes tipos conserva a regra que introduz o segmento.

□

Teorema 6.2 *Seja* Π *uma derivação de* $\Gamma \vdash \varphi$ *onde* $d(\Pi) = n$ *e todos os segmentos máximos que contribuem para* $d(\Pi) = n$ *são da forma* $a.1$ *ou* $\alpha.1$.

Então Π *se reduz a* Π' *tal que* $d(\Pi') \leq d(\Pi)$ *e* $d_e(\Pi') < d_e(\Pi)$ *e todo segmento (se houver) que contribui para o caso de* $d(\Pi') = n$ *é da forma* $a.1$.

Prova: Basta recursivamente escolhermos um segmento máximo equacional que contribui para $d_e(\Pi) = n$ de tal forma que não ocorra um segmento máximo acima da última fórmula deste (nem ao lado, nem acima da fórmula ao lado desta) que contribua para $d_e(\Pi) = n$. Utilizando o corolário 6.1 diversas vezes para eliminar os segmentos do tipo $\alpha.1$ que contribuem para o grau equacional de Π, obtemos o resultado. Dessa forma, os únicos segmentos máximos (se existirem) que contribuem para $d(\Pi) = n$ são da forma $a.1$.

□

Lema 6.3 *Seja* Π *uma derivação de* $\Gamma \vdash \varphi$ *onde* $r(\Pi) = EQ2$ *e a premissa primeira* ψ *tal que:*

1. ψ *é a última ocorrência de fórmula de todos os segmentos máximos de* Π.

2. $d(\Pi) > 0$

Então, Π se reduz a uma derivação Π' de $\Delta \subseteq \Gamma \vdash \varphi$, tal que $d(\Pi') < d(\Pi)$.

Prova: Indução no comprimento de Π:

- Se o segmento é da forma $\alpha.1$ ou $\alpha.2$ basta aplicarmos um número suficiente de vezes a respectiva regra de redução. É fácil ver que não há formação de novos segmentos máximos cujo grau seja maior que $d(\Pi)$.

- Se o segmento é da forma $\beta.1$ e,

$$\Pi = \dfrac{\dfrac{\begin{matrix}\Pi_1\\ \psi_1 \vee \psi_2\end{matrix} \quad \begin{matrix}\Pi_2\\ \psi\end{matrix} \quad \begin{matrix}\Pi_3\\ \psi\end{matrix}}{\psi} \quad \begin{matrix}\Pi_4\\ t \equiv t'\end{matrix}}{\varphi}$$

Π se reduz a Π^*:

$$\Pi^* = \dfrac{\begin{matrix}\Pi_1\\ \psi_1 \vee \psi_2\end{matrix} \quad \dfrac{\begin{matrix}\Pi_2\\ \psi\end{matrix} \quad \begin{matrix}\Pi_4\\ t \equiv t'\end{matrix}}{\varphi} \quad \dfrac{\begin{matrix}\Pi_3\\ \psi\end{matrix} \quad \begin{matrix}\Pi_4\\ t \equiv t'\end{matrix}}{\varphi}}{\varphi}$$

Se $d(\Pi^*) < d(\Pi)$, então $\Pi' \equiv_d \Pi^*$.

Se $d(\Pi^*) = d(\Pi)$, então aplicando a hipótese de indução nas sub-árvores de Π determinadas pela premissas menores acima, temos:

$$\dfrac{\begin{matrix}\Pi_i\\ \psi\end{matrix} \quad \begin{matrix}\Pi_4\\ t \equiv t'\end{matrix}}{\varphi} \Rightarrow \begin{matrix}\Pi_i\\ \varphi\end{matrix} \;\; (i \in \{2,3\})$$

tal que $d(\Pi_i) < d(\Pi)$.
Podemos então considerar Π' como

$$\Pi' = \dfrac{\begin{matrix}\Pi_1\\ \psi_1 \vee \psi_2\text{‘}\end{matrix} \quad \begin{matrix}\Pi_2\\ \psi\end{matrix} \quad \begin{matrix}\Pi_3\\ \psi\end{matrix}}{\varphi}$$

logo, $d(\Pi') < d(\Pi)$.
O caso de

$$\Pi = \dfrac{\dfrac{\begin{matrix}\Pi_1\\ \exists x \varphi_1\end{matrix} \quad \begin{matrix}\Pi_2\\ \psi\end{matrix}}{\psi} \quad \begin{matrix}\Pi_3\\ t \equiv t'\end{matrix}}{\varphi}$$

é similar.

$\square$

Lema 6.4 *Seja* Π *uma derivação de* $\Gamma \vdash \varphi$*, tal que:*

1. $r(\Pi)$ *é uma regra de eliminação, cuja premissa maior* ψ *é a última ocorrência de fórmula de todos os segmentos máximos de* Π*;*
2. $d(\Pi) > 0$

Então, Π *se reduz a uma derivação* Π' *de* $\Delta \subseteq \Gamma \vdash \varphi$*, tal que* $d(\Pi') < d(\Pi)$.

Prova: Por indução no comprimento de Π.

1. Se ψ é do tipo $a.1$ o resultado segue diretamente das reduções.
2. Se ψ é do tipo $b.1$ segue como em MASSI [2].
3. Se ψ é do tipo $a.2$, então

$$\Pi = \cfrac{\cfrac{\begin{array}{c}[\neg\psi]^k \\ \Pi \\ \bot\end{array}}{\psi} \quad \Sigma}{\varphi}\,{,k}$$

se reduz a

$$\Pi^* = \cfrac{\begin{array}{c}\cfrac{\cfrac{\cfrac{[\psi]^j \quad \Sigma}{\varphi} \quad [\neg\varphi]^i}{\bot}}{[\neg\psi]^k}\,{,j} \\ \Pi^0 \\ \bot\end{array}}{\varphi}\,{,i}$$

onde todas as partes de Π, da forma

$$\cfrac{\begin{array}{c}\Pi_2 \\ \psi\end{array} \quad [\neg\psi]^k}{\bot}$$

são substituídas por

$$\cfrac{\cfrac{\begin{array}{c}\Pi_2 \\ \psi\end{array} \quad \Sigma}{\varphi} \quad [\neg\varphi]^i}{\bot}$$

e todas as partes de Π, da forma

$$\dfrac{\begin{array}{c}\Pi_3\\ \xi_n\end{array} \quad \dfrac{\begin{array}{c}\dfrac{[\neg\psi]^k \quad \begin{array}{c}\Sigma\\ t_1\equiv t_1'\end{array}}{\neg\xi_2}\\ \vdots\\ \neg\xi_{n-1}\end{array} \quad \begin{array}{c}\Pi_{n-1}\\ t_{n-1}\equiv t_{n-1}'\end{array}}{\neg\xi_n}}{\bot}$$

foram substituídas por

$$\dfrac{\dfrac{\dfrac{\begin{array}{c}\dfrac{\begin{array}{c}\Pi_3\\ \xi_n\end{array} \quad \begin{array}{c}\Pi^\alpha_{n-1}\\ t_{n-1}'\equiv t_{n-1}\end{array}}{\xi_{n-1}}\\ \vdots\\ \xi_2\end{array} \quad \begin{array}{c}\Pi^\alpha_1\\ t_1'\equiv t_1\end{array}}{\psi} \quad \Sigma}{\varphi} \quad [\neg\varphi]^i}{\bot}$$

Os únicos segmentos máximos que podem ocorrer em $\Pi*$ são todos do tipo $a.1$, $\alpha.1$, $b.2$ e $\beta.2$. Os dois primeiros são da forma do teorema 6.2, portanto aplicando tal lema temos uma derivação Π' em que os segmentos máximos que ocorrem são do tipo $a.1$, $b.2$ e $\beta.2$, aplicando o lema 6.1 obtemos o resultado.

Lema 6.5 *Seja* Π *uma derivação de* $\Gamma \vdash \varphi$*, então* Π *se reduz a uma derivação normal* Π' *de* $\Delta \subseteq \Gamma \vdash \varphi$*.*

Prova: Por indução no par ordenado (α, β), onde α é o grau de Π e β é o seu comprimento.

1. Se $r(\Pi)$ é uma introdução ou a regra do absurdo, então o resultado segue da hipótese indutiva.

2. Se $r(\Pi)$ é uma eliminação, ou uma regra da igualdade, então Π é da forma

$$\Pi \equiv \dfrac{\begin{array}{c}\Pi_1\\ \varphi_1\end{array} \quad \begin{array}{c}\Pi_2\\ \varphi_2\end{array} \quad \ldots \quad \begin{array}{c}\Pi_n\\ \varphi_n\end{array}}{\varphi}$$

Usando a hipótese indutiva, cada Π_i $(1 \le i \le n)$ se reduz a uma derivação normal

Π'_i. Seja Π a seguinte derivação:

$$\Pi \equiv \cfrac{\begin{matrix}\Pi'_1 & \Pi'_2 & & \Pi'_n \\ \varphi_1 & \varphi_2 & \dots & \varphi_n\end{matrix}}{\varphi}$$

Se Π é normal, então consideramos $\Pi' \equiv_d \Pi$. Se Π não é normal, então Π' é obtida de Π usando lema 6.2 (caso $d(\Pi)$=0) ou lema 6.4 (caso $d(\Pi) > 0$ e $r(\Pi)$ é uma regra de eliminação) ou aplicando 6.3 (caso $d(\Pi) > 0$ e $r(\Pi)$ é uma regra de igualdade). O resultado segue da hipótese indutiva com respeito a α.

6.7 Normalização para LFP_{fin} com restrições

Nesta seção apresentaremos um teorema de normalização para LFP_{fin} com uma restrição: iremos lidar com conjuntos que possuam f-testemunhas. Como vimos anteriormente, a prova de uma fórmula φ a partir de um conjunto com f-testemunhas possui uma versão finita (ou seja, sem o uso de regras infinitárias), isto é obtido atráves do uso regra ($LFP\ PF$). Porém, como visto, podemos eliminar a regra ($LFP\ PF$) e utilizar somente a regra ($LFP\ E$) como única regra infinitária adotada. A partir disto, podemos obter um sistema finitário de prova para conjuntos com f-testemunhas se restringirmos as regras ($LFP\ E$) e ($LFP\ I$). Com estes conceitos, o objetivo desta seção é fornecer uma base para em um trabalho futuro provarmos um teorema de normalização para LFP_{fin} sem esta restrição.

Lembramos que devido a utilizarmos somente conjuntos com f-testemunhas nas provas a seguir, não iremos utilizar a regra infinitária ($FIN- \perp$): conforme foi provado no capítulo anterior, em conjuntos dessa forma a regra pode ser eliminada.

Quanto as regras apresentadas no capítulo 5, e dado que provaremos a partir de um conjunto com uma f-testemunha $\lambda_{\leq i}$, exigiremos as seguintes restrições:

1. A partir da regra ($LFP\ I$), definimos $(LFP\ I)_{res}$ como:

$$\frac{\varphi^j(\vec{t})}{[\mathbf{lfp}_{R,\vec{x}}\ \varphi(R,\vec{x})](\vec{t})}\ \text{, com } j \leq i^{|\vec{t}|}$$

2. A partir da regra $(LFP\ E)$, definimos $(LFP\ E)_{res}$ como:

$$\dfrac{\begin{matrix} \Pi \\ [\mathbf{lfp}_{R,\vec{x}}\ \varphi(R,\vec{x})](\vec{t}) \end{matrix} \quad \begin{matrix} \varphi^0(\vec{t}) \\ \Pi_0 \\ \psi \end{matrix} \quad \begin{matrix} \varphi^1(\vec{t}) \\ \Pi_1 \\ \psi \end{matrix} \quad \ldots \quad \begin{matrix} \varphi^{i^{|\vec{t}|}}(\vec{t}) \\ \Pi_{i^{|\vec{t}|}} \\ \psi \end{matrix}}{\psi}$$

Quanto a definição de grau de uma fórmula, estenderemos a definição anterior incluindo a seguinte regra:

$$d([\mathbf{lfp}_{R,\vec{x}}\ \varphi(R,\vec{x})](\vec{t})) = max\{\varphi^0(\vec{t}), \varphi^1\vec{(t)}, \ldots \varphi^{i^{|t|}}(\vec{t})\} + 1$$

Para a prova da normalização, e dada a prova a partir de um conjunto Γ com uma f-testemunha $\lambda_{\leq i}$, precisaremos alterar as seguintes definições de segmento máximo clássico e equacional (conforme seção 6.3 deste capítulo):

1. Quanto a definição de segmento máximo clássico:

   $b.1$ φ_n é a conclusão da regra $(\vee E)$ ou $(\exists E)$ ou $(LFP\ E)_{res}$ e é, ao mesmo tempo, premissa maior de uma regra de eliminação;

   $b.2$ φ_n é a conclusão da regra $(\vee E)$ ou $(\exists E)$ ou $(LFP\ E)_{res}$ e é, ao mesmo tempo, premissa menor de uma aplicação da regra $(\rightarrow E)$, cuja premissa maior é uma top-fórmula da forma $(\varphi_n \rightarrow \perp)$.

2. Quanto a definição de segmento máximo equacional:

   $\beta.1$ φ_n (n=1) é a conclusão da regra $(\vee E)$ ou $(\exists E)$ ou $(LFP\ E)_{res}$ e é, ao mesmo tempo, premissa primeira da regra $EQ2$.

   $\beta.2$ φ_n (n=1) é a conclusão da regra $(\vee E)$ ou $(\exists E)$ ou $(LFP\ E)_{res}$ e é, ao mesmo tempo, premissa menor de aplicação da regra $(\rightarrow E)$, cuja premissa maior é da forma $(\varphi_n \rightarrow \perp)$ do seguinte modo:

$$\dfrac{\dfrac{\begin{matrix} \Pi \\ (\sigma \vee \tau) \end{matrix} \quad \begin{matrix} \Pi_1' \\ \varphi_n \end{matrix} \quad \begin{matrix} \Pi_2'' \\ \varphi_n \end{matrix}}{\varphi_n} \qquad \dfrac{\begin{matrix} \dfrac{\neg\xi_1 \quad \begin{matrix} \Pi_1 \\ t_1 \equiv t_1' \end{matrix}}{\neg\xi_2} \\ \vdots \\ \neg\xi_k \end{matrix} \qquad \begin{matrix} \Pi_k \\ t_k \equiv t_k' \end{matrix}}{\neg\varphi_n}}{\perp}$$

no caso em que φ_n é a conclusão da regra $(\vee E)$ ou

$$\dfrac{\dfrac{\begin{matrix}\Pi\\ \exists x\psi\end{matrix} \quad \begin{matrix}\Pi'\\ \varphi_n\end{matrix}}{\varphi_n} \qquad \dfrac{\begin{matrix}\dfrac{\neg\xi_1 \quad \begin{matrix}\Pi_1\\ t_1 \equiv t_1'\end{matrix}}{\neg\xi_2}\\ \vdots\\ \neg\xi_k\end{matrix} \qquad \begin{matrix}\Pi_k\\ t_k \equiv t_k'\end{matrix}}{\neg\varphi_n}}{\bot}$$

no caso em que φ_n é a conclusão da regra $(\exists E)$, ou ainda

$$\dfrac{\dfrac{\begin{matrix}\Pi\\ [\mathbf{lfp}_{R,\vec{x}}\ \varphi(R,\vec{x})](\vec{t})\end{matrix} \quad \begin{matrix}\varphi^0(\vec{t})\\ \Pi_0\\ \varphi_n\end{matrix} \quad \begin{matrix}\varphi^1(\vec{t})\\ \Pi_1\\ \varphi_n\end{matrix} \quad \ldots \quad \begin{matrix}\varphi^{i^{|\vec{t}|}}(\vec{t})\\ \Pi_{i^{|\vec{t}|}}\\ \varphi_n\end{matrix}}{\varphi_n} \qquad \dfrac{\begin{matrix}\dfrac{\neg\xi_1 \quad \begin{matrix}\Pi_1\\ t_1 \equiv t_1'\end{matrix}}{\neg\xi_2}\\ \vdots\\ \neg\xi_k\end{matrix} \qquad \begin{matrix}\Pi_k\\ t_k \equiv t_k'\end{matrix}}{\neg\varphi_n}}{\bot}$$

no caso em que φ_n é a conclusão da regra $(LFP\ E)_{res}$.

Para a prova da normalização, incluiremos as seguintes reduções na lista anterior de reduções clássicas:

1. Nas reduções clássicas operacionais, adicionamos a seguinte regra:

$$\dfrac{\dfrac{\begin{matrix}\Pi\\ \varphi^j(\vec{t})\end{matrix}}{[\mathbf{lfp}_{R,\vec{x}}\ \varphi(R,\vec{x})](\vec{t})}\,(LFP\ I)_{res} \quad \begin{matrix}[\varphi^0(\vec{t})]\\ \Pi_0\\ \psi\end{matrix} \quad \begin{matrix}[\varphi^1(\vec{t})]\\ \Pi_1\\ \psi\end{matrix} \quad \ldots \quad \begin{matrix}[\varphi^{i^{|\vec{t}|}}(\vec{t})]\\ \Pi_{i^{|\vec{t}|}}\\ \psi\end{matrix}}{\psi}\,(LFP\ E)_{res}$$

por

$$\begin{matrix}\Pi\\ [\varphi^j(\vec{t})]\\ \Pi_j\\ \psi\end{matrix}$$

2. Quanto as reduções clássicas permutativas, incluímos:

$$\dfrac{\dfrac{\begin{matrix}\Pi\\ [\mathbf{lfp}_{R,\vec{x}}\ \varphi(R,\vec{x})](\vec{t})\end{matrix} \quad \begin{matrix}[\varphi^0(\vec{t})]\\ \Pi_0\\ \psi\end{matrix} \quad \begin{matrix}[\varphi^1(\vec{t})]\\ \Pi_1\\ \psi\end{matrix} \quad \ldots \quad \begin{matrix}[\varphi^{i^{|\vec{t}|}}(\vec{t})]\\ \Pi_{i^{|\vec{t}|}}\\ \psi\end{matrix}}{\psi} \qquad \Sigma}{\psi'}$$

por

$$\dfrac{\begin{matrix}\Pi\\ [\mathbf{lfp}_{R,\vec{x}}\ \varphi(R,\vec{x})](\vec{t})\end{matrix} \quad \dfrac{\begin{matrix}[\varphi^0(\vec{t})]\\ \Pi_0\\ \psi\end{matrix} \quad \Sigma}{\psi'} \quad \dfrac{\begin{matrix}[\varphi^1(\vec{t})]\\ \Pi_1\\ \psi\end{matrix} \quad \Sigma}{\psi'} \quad \dots \quad \dfrac{\begin{matrix}[\varphi^{i^{|\vec{t}|}}(\vec{t})]\\ \Pi_{i^{|\vec{t}|}}\\ \psi\end{matrix} \quad \Sigma}{\psi'}}{\psi'}$$

onde ψ é uma premissa maior de uma regra de eliminação e Σ pode ocorrer a esquerda.

3. E, finalmente, quanto as reduções clássicas do absurdo, adicionamos:

$$\dfrac{\dfrac{\begin{matrix}\Pi\\ [\mathbf{lfp}_{R,\vec{x}}\ \varphi(R,\vec{x})](\vec{t})\end{matrix} \quad \begin{matrix}[\varphi^0(\vec{t})]\\ \Pi_0\\ \psi\end{matrix} \quad \begin{matrix}[\varphi^1(\vec{t})]\\ \Pi_1\\ \psi\end{matrix} \quad \dots \quad \begin{matrix}[\varphi^{i^{|\vec{t}|}}(\vec{t})]\\ \Pi_{i^{|\vec{t}|}}\\ \psi\end{matrix}}{\psi} \qquad [\neg\psi]}{\bot}$$

por

$$\dfrac{\begin{matrix}\Pi\\ [\mathbf{lfp}_{R,\vec{x}}\ \varphi(R,\vec{x})](\vec{t})\end{matrix} \quad \dfrac{\begin{matrix}[\varphi^0(\vec{t})]\\ \Pi_0\\ \psi\end{matrix} \quad [\neg\psi]}{\bot} \quad \dfrac{\begin{matrix}[\varphi^1(\vec{t})]\\ \Pi_1\\ \psi\end{matrix} \quad [\neg\psi]}{\bot} \quad \dots \quad \dfrac{\begin{matrix}[\varphi^{i^{|\vec{t}|}}(\vec{t})]\\ \Pi_{i^{|\vec{t}|}}\\ \psi\end{matrix} \quad [\neg\psi]}{\bot}}{\bot}$$

.

Para as reduções equacionais, adicionaremos as seguintes reduções:

1. Quanto as reduções equacionais operacionais, incluimos:

$$\dfrac{\dfrac{\begin{matrix}\Pi_0\\ \varphi^j(\vec{t})\frac{t'}{x}\end{matrix}}{[\mathbf{lfp}_{R,\vec{x}}\ \varphi(R,\vec{x})](\vec{t})\frac{t'}{x}}\ (LFP\ I)_{res} \quad \begin{matrix}\Pi_1\\ t'\equiv t''\end{matrix}}{[\mathbf{lfp}_{R,\vec{x}}\ \varphi(R,\vec{x})](\vec{t})\frac{t''}{x}}$$

por

$$\dfrac{\dfrac{\begin{matrix}\Pi_0\\ \varphi^j(\vec{t})\frac{t'}{x}\end{matrix} \quad \begin{matrix}\Pi_1\\ t'\equiv t''\end{matrix}}{\varphi^j(\vec{t})\frac{t''}{x}}\ (LFP\ I)_{res}}{[\mathbf{lfp}_{R,\vec{x}}\ \varphi(R,\vec{x})](\vec{t})\frac{t''}{x}}$$

2. Para as reduções equacionais permutativas, adicionamos:

$$\dfrac{\dfrac{\begin{array}{c}\Pi\\ [\mathbf{lfp}_{R,\vec{x}}\ \varphi(R,\vec{x})](\vec{t})\end{array} \quad \begin{array}{c}[\varphi^{0}(\vec{t})]\\ \Pi_0\\ \psi\frac{t'}{x}\end{array} \quad \begin{array}{c}[\varphi^{1}(\vec{t})]\\ \Pi_1\\ \psi\frac{t'}{x}\end{array} \quad \dots \quad \begin{array}{c}[\varphi^{i^{|\vec{t}|}}(\vec{t})]\\ \Pi_{i^{|\vec{t}|}}\\ \psi\frac{t'}{x}\end{array}}{\psi\frac{t'}{x}} \qquad \begin{array}{c}\Sigma\\ t' \equiv t''\end{array}}{\psi\frac{t''}{x}}$$

por

$$\dfrac{\begin{array}{c}\Pi\\ [\mathbf{lfp}_{R,\vec{x}}\ \varphi(R,\vec{x})](\vec{t})\end{array} \quad \dfrac{\begin{array}{c}[\varphi^{0}(\vec{t})]\\ \Pi_0\\ \psi\frac{t'}{x}\end{array} \quad \begin{array}{c}\Sigma\\ t' \equiv t''\end{array}}{\psi\frac{t''}{x}} \quad \dfrac{\begin{array}{c}[\varphi^{1}(\vec{t})]\\ \Pi_1\\ \psi\frac{t'}{x}\end{array} \quad \begin{array}{c}\Sigma\\ t' \equiv t''\end{array}}{\psi\frac{t''}{x}} \quad \dots \quad \dfrac{\begin{array}{c}[\varphi^{i^{|\vec{t}|}}(\vec{t})]\\ \Pi_{i^{|\vec{t}|}}\\ \psi\frac{t'}{x}\end{array} \quad \begin{array}{c}\Sigma\\ t' \equiv t''\end{array}}{\psi\frac{t''}{x}}}{\psi\frac{t''}{x}}$$

3. E, finalmente, para as reduções equacionais do absurdo adicionamos:

$$\dfrac{\dfrac{\begin{array}{c}\Pi\\ [\mathbf{lfp}_{R,\vec{x}}\ \varphi(R,\vec{x})](\vec{t})\end{array} \quad \begin{array}{c}[\varphi^{0}(\vec{t})]\\ \Pi_0\\ \psi\end{array} \quad \begin{array}{c}[\varphi^{1}(\vec{t})]\\ \Pi_1\\ \psi\end{array} \quad \dots \quad \begin{array}{c}[\varphi^{i^{|\vec{t}|}}(\vec{t})]\\ \Pi_{i^{|\vec{t}|}}\\ \psi\end{array}}{\psi} \qquad \dfrac{\begin{array}{c}\dfrac{\neg\xi_1 \quad \begin{array}{c}\Pi_1\\ t_1 \equiv t_1^{'}\end{array}}{\neg\xi_2}\\ \vdots\\ \neg\xi_k\end{array} \quad \begin{array}{c}\Pi_k\\ t_k \equiv t_k^{'}\end{array}}{\neg\psi}}{\bot}$$

por

$$\dfrac{\begin{array}{c}\Pi\\ [\mathbf{lfp}_{R,\vec{x}}\ \varphi(R,\vec{x})](\vec{t})\end{array} \quad \dfrac{\begin{array}{c}[\varphi^{0}(\vec{t})]\\ \Pi_0\\ \psi\end{array} \quad \Sigma}{\bot} \quad \dfrac{\begin{array}{c}[\varphi^{1}(\vec{t})]\\ \Pi_1\\ \psi\end{array} \quad \Sigma}{\bot} \quad \dots \quad \dfrac{\begin{array}{c}[\varphi^{i^{|\vec{t}|}}(\vec{t})]\\ \Pi_{i^{|\vec{t}|}}\\ \psi\end{array} \quad \Sigma}{\bot}}{\bot}$$

. onde

$$\Sigma = \dfrac{\begin{array}{c}\dfrac{\neg\xi_1 \quad \begin{array}{c}\Pi_1\\ t_1 \equiv t_1^{'}\end{array}}{\neg\xi_2}\\ \vdots\\ \neg\xi_k\end{array} \quad \begin{array}{c}\Pi_k\\ t_k \equiv t_k^{'}\end{array}}{\neg\psi}$$

A prova de normalização para este caso restrito segue como similar a que apresentamos neste capítulo para FOL com as regras de igualdade.

Teorema 6.3 (Normalização Fraca para LFP_{fin} restrito) *Seja* Π *uma derivação em* LFP_{fin} *restrito, então* Π *se reduz a uma derivação normal.*

7 *Conclusão e Trabalhos Futuros*

7.1 Considerações sobre os Resultados da Normalização e as Linguagens de Consulta em Bancos de Dados Relacionais

O procedimento de normalização é um processo de reescrita que transforma uma derivação em outra através de passos de redução. Estes passos de redução objetivam eliminar todas as fórmulas máximas, isto é, fórmulas que representam algum tipo de redundância. O produto final deste processo é uma derivação em forma normal.

A definição precisa destas provas normais depende das propriedades que queremos obter delas. Normalmente é requerido, pelo menos, o princípio da subfórmula que diz que todas as fórmulas que aparecem em uma derivação normal são ou subfórmulas das hipóteses não descartadas na prova ou subfórmulas da conclusão[1]. Como consequência importante do princípio da subfórmula temos a possibilidade de construir provadores automáticos de teoremas de uma maneira mais eficiente já que todas as fórmulas que aparecem em uma derivação normal são previsíveis.

Nós seguimos o procedimento de normalização adotado por Prawitz [17] e Massi [2] para as regras de dedução natural clássicas para o caso dos conectivos e quantificadores. Veja também [22] para detalhes adicionais. Para nossas regras de igualdade, reflexividade e substituição, nós definimos passos de redução de forma a levar todas as regras de igualdade para o topo da dedução, ou imediatamente abaixo de regras de eliminação. A motivação para tais reduções permutativas é que a regra de substituição pode esconder segmentos máximos.

[1]Exceto para hipóteses descartadas por aplicações da regra de *Redução ao Absurdo* e para ocorrências de $\perp$ que ocorrem imediatamente abaixo de tais hipóteses.

O procedimento de normalização para todo o sistema *LFP* combina todos esses passos de redução mais novos para lidar com *FIN* $\perp$, *LFP I*, *LFP E* e *LFP PF*. Para simplificar o processo, nos assumimos que sempre sabemos o limite máximo de cardinalidade dos modelos finitos que estamos lidando em *LFP*. A contrapartida sintática desta limitação é representada pelo fato de que, em todas as derivações de $LFP \vdash_{LFP_{fin}} \varphi$ sob consideração no procedimento de normalização, existe um n tal que $\neg\lambda_{\geq n}$ pertence a Γ. Com esta restrição, as regras infinitárias não são mais necessárias, e precisaremos apenas de uma versão finitária de *LFP E*, a $(LFP\ E)_{res}$, mais a $(LFP\ I)_{res}$ além das regra clássicas já mencionadas. Além disso, os novos passos de redução para $(LFP\ E)_{res}$, $(LFP\ I)_{res}$ são similares aos definidos para regras de primeira-ordem clássica. Neste sistema finitário é possível provar o teorema da normalização para *LFP* no sentido de que, se $\Gamma \cup \{\neg\lambda_{\geq n}\}$, então existe uma prova normal de φ de $\Gamma \cup \{\neg\lambda_{\geq n}\}$ tal que o princípio da subfórmula vale.

Como dissemos anteriormente, a importância de obter o princípio da subfórmula é possibilidade de construção de um provador de teoremas para *LFP*. Esse provador de teoremas poderia ser útil, por exemplo, para computar consultas em banco de dados relacionais.

Para ilustrar esta situação, considere a linguagem de consulta *DATALOG* e suas relações com *LFP*. Um programa *DATALOG* é um conjunto de regras, possivelmente recursivas, da forma $P(\bar{x})$:- $\alpha_1(\bar{x},\bar{y}),\ldots,\alpha_m(\bar{x},\bar{y})$. Por *DATALOG*$\neg$ queremos dizer uma extensão de *DATALOG* onde as fórmulas atômicas negadas só podem aparecer no escopo '$\alpha_1(\bar{x},\bar{y}),\ldots,\alpha_m(\bar{x},\bar{y})$' das regras. É conhecido que $\exists$LFP = DATALOG$_\neg$ [14], isto é, toda consulta *DATALOG*$\neg$ pode ser traduzida para $\exists$*LFP*, o fragmento existencial de *LFP*, e vice-versa.

Agora iremos investigar um resultado de decidibilidade sobre $\exists$*LFP*. Seja φ uma sentença $\exists$*LFP*. Já que toda fórmula *LFP* é equivalente à uma fórmula $\mathscr{L}_{\infty\omega}$[2] [14], podemos provar que podemos transformar uma $\exists$LFP-formula φ para uma equivalente φ' em $\mathscr{L}_{\infty\omega}$ que contém somente conjunções finitas, disjunções finitas, uma disjunção infinita contável e quantificação existencial, com a negação aplicada somete a fórmulas atômicas. Para provar que $\vdash_{\exists LFP} \varphi$, podemos alternativamente provar que $\neg\varphi$ é insatisfatível. Além disso, podemos transformas $\neg\varphi$ em uma fórmula $\neg\varphi'$ de $\mathscr{L}_{\infty\omega}$ que contém somente con-

[2] $\mathscr{L}_{\infty\omega}$ estende *FOL* com conjunções e disjunções infinitárias.

junções finitas, disjunções finitas, uma conjunção infinita e quantificação universal, com a negação aplicada somente a fórmulas atômicas. Portanto, sentença resultante $\neg\varphi'$ é uma conjunção, possivelmente infinita, de sentenças universais finitas de FOL. Portanto, $\neg\varphi'$ é equivalente a uma Teoria T infinita de FOL com sentenças universais.

É conhecido que satisfabilidade de fórmulas universais são preservadas sob subestruturas [6]. Segue que, em um vocabulário relacional, se T é um possível conjunto infinito de sentenças universais de FOL, então T é satisfatível se e somente se existe um número natural n tal que T tem um modelo de cardinalidade n. Então, T é satisfatível se e somente se é finitamente satisfatível. De outro modo, pelo teorema da compacidade de FOL, T é insatisfatível se e somente se existe um subconjunto finito $T_0 \subseteq T$ tal que T_0 é insatisfatível. Segue que T é finitamente insatisfatível se e somente se existe um subconjunto finito $T_0 \subseteq T$ que é finitamente insatisfatível. Como precisamos somente verificar os modelos de cardinalidade n, a insatisfabilidade de T, e portanto a validade de $\varphi \in \exists LFP$ é semi-decidível. De fato, se $T = \{\varphi_1, \varphi_2, \ldots\}$, basta procedermos verificando se

$$\bigwedge_{0 \leq i \leq m} \varphi_i$$

tem um modelo de cardinalidade n, para cada $m \in N$. Como $T_0 \subseteq \{\varphi_1, \ldots, \varphi_h\}$ para algum $h \in N$, segue que, se T é finitamente insatisfatível, então este procedimento realmente pára e, como conseqûência, o conjunto de teoremas de $\exists LFP$ é recursivamente enumerável. Isto parece indicar que existe uma prova finita para $\vdash_{\exists LFP} \varphi$ em nosso cálculo LFP_{fin}, numa versão finitária de nosso sistema LFP.

Agradecimento: Agradecemos a Francicleber Martins Ferreira por suas contribuições nesta parte.

7.2 Conclusões

Nesta dissertação, introduzimos um sistema de dedução infinitário para a lógica de menor ponto-fixo (LFP) que é correto e completo em relação à LFP em modelos finitos. Este cálculo é uma extensão de um sistema infinitário para a lógica de primeira-ordem em modelos finitos. À primeira vista parece ser impossível obter um sistema dedutivo para FOL_{fin} e LFP_{fin} já que validade sobre modelos finitos em FOL não é recursivamente enumerável. O truque aqui foi relaxar a noção de prova formal permitindo o uso de regras infinitárias.

Podemos comparar nosso trabalho com o sistema definido por Compton [3] para o fragmento existencial de LFP. Nossa regra de introdução e eliminação no estilo de dedução natural para o novo operador aqui definido, o operador **lfp**, está estreitamente relacionada com as regras de introdução à esquerda e à direita para definições indutivas no cálculo de seqüentes de Compton. Entretanto, nós estamos interessados no uso da LFP quanto ao escopo dos modelos finitos, diferentemente do trabalho de Compton, nós introduzimos duas regras para lidar com a cardinalizadade do domínio, a regra $FIN\ \perp$ usada no contexto de FOL_{fin}, e a regra $LFP\ PF$, com regras adicionais para lidar com o quantificador universal.

Estamos particularmente interessado na LFP em modelos finitos devido a sua importância no escopo da ciência da computação teórica. Como fato, LFP captura a classe de complexidade $PTIME$ sobre a classe de estruturas finitas com ordenação. Já que LFP é tradicionalmente definida no escopo da Teoria dos Modelos Finitos (FMT em inglês), as definições de nosso sistema dedutivo infinitário para LFP abre caminhos alternativos de provar resultados já obtidos em Teoria dos Modelos Finitos e também novos resultados em ponto de vista da teoria da prova, como teoremas de normalização e seus corolários (como o príncipio da subfórmula, por exemplo). Além disso, com algumas restrições, este sistema dedutivo pode ser usada como provador de teorema para computar consultas em bancos de dados relacionais.

Como trabalho futuro, pretendemos estender a normalização para LFP_{fin} sem restrições. Intuitivamente esta prova poderia fornecer um princípio da subfórmula de forma que as fórmulas que aparecem na prova possuíriam subfórmulas das hipóteses e/ou da conclusão e/ou da sequência $\lambda_{\geq n}$. Também pretendemos investigar mais resultados que obteríamos em teoria dos modelos finitos.

Referências Bibliográficas

[1] B.A.Trakhtenbrot. *The Impossibility of an Algorithm for the Decision Problem for Finite Models*, volume 70. Doklady Academii Nauk SSSR, 1950.

[2] C.D.B.Massi. *Provas de Normalização para a Lógica Clássica*. Departamento de Filosofia, Universidade Estadual de Campinas, 1990.

[3] K. COMPTON. A Deductive System for Existential Least Fixpoint Logic. *Journal of Logic and Computation*, 3(2):197–213, 2002.

[4] D. Van Dalen. *Logic and Structure*. Springer Verlag, 2 edition, 1989.

[5] H.D. Ebbinghaus and J. Flum. *Finite Model Theory*. Springer-Verlag, 1995.

[6] H.D. Ebbinghaus, J. Flum, and W. Thomas. *Mathematical Logic*. Springer-Verlag, second edition, 1994.

[7] R. Fagin. *Generalized First-order Spectra and Polynomial-time Recognizable Sets*, volume 7. R. Karp, 1974.

[8] R. Fraïssé. *Sur Quelques Classifications des Systèmes de Relations*, volume 1. 1954.

[9] G. Gentzen. Die widerspruchsfreiheit der reinen zahlentheorie. *Mathematische Annalen*, 112:493–565, 1936. Tradução em [21], 132-170.

[10] G. Gentzen. Beweisbarkeit und unbeweisbarkeit von anfangsfällen der transfiniten induktion in der reinen zahlentheorie. *Mathematische Annalen*, 119:140–161, 1943. Tradução em Gentzen [21], 287-311.

[11] K. Hrbacek and T.J. Jech. *Introduction to Set Theory*. Marcel Dekker, 1999.

[12] N. Immerman. Relational queries computable in polynomial time (extended abstract). *In ACM Symposium on Theory of Computation*, pages 147–152, 1982.

[13] N. Immerman. Upper and lower bounds for first order expressability. *Journal of Computer and System Sciences*, 25:287–311, 1982.

[14] L. Libkin. *Elements of Finite Model Theory*. Springer, 2004.

[15] E.G.K. López-Escobar and Ï.M.L. D'Ottaviano. *A regra ω: passado, presente e futuro*. Centro de Lógica, Epistemologia e História da Ciência, Universidade Estadual de Caminas, 1987.

[16] M.Y.Vardi. The complexity of relational query languages. pages 137–146, 1982.

[17] D. Prawitz. *Natural Deduction: A Proof-Theoretical Study*, volume 3. Stockholm Studies in Philosophy, ALMQVIST AND WIKSELL, 1965.

[18] D. A. Schmidt. *Denotational Semantics: a Methodology for Language Development.* Allyn and Bacon, 1986.

[19] K. Schütte. Beweistheoretische erfassung der unendliche induktion in der zahlentheorie. *Mathemastiche Annalen*, 122:369–389, 1950.

[20] D. Scott. Domains for denotational semantics. *LNCS 140: Proc. 9th ICALP*, pages 577–613, 1982.

[21] M. E. Szabo, editor. *The Collected Papers of Gerhard Gentzen.* North-Holland, 1969.

[22] A.S. Troelstra and H. Schwichtenberg. *Basic Proof Theory.* Cambridge University Press, 1996.