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

Modelos Minimais e Hierarquia de Expressividade

Francicleber Martins Ferreira

Orientadora: Profa. Ana Teresa de Castro Martins

Fortaleza
23 de Janeiro de 2007

ao meu pai, à minha mãe
e
ao meu avô
Francisco Martins da Rocha
(26 de Novembro de 1912—7 de Maio de 2007—∞)

Agradecimentos

É... acabou. Mais uma fase vencida. Mas foi bom, embora difícil! Fiquei feliz com o resultado. Nesse meio tempo, tive a ajuda inestimável de várias forças. Às quais dedico esta seção.

Agradeço ao pessoal da Biblioteca da Matemática que sempre atendeu com rapidez e eficiência aos meus pedidos de artigos efetuados através do COMUT. Ao CNPq e à FUNCAP pelo apoio financeiro que viabilizou este trabalho através de bolsa de mestrado. Ao Professor Tarcisio H. C. Pequeno, que alocou recursos de projetos de pesquisa sob sua coordenação para que eu pudesse apresentar parte do meu trabalho no Encontro Brasileiro de Lógica de 2006 e no Simpósio Brasileiro de IA também de 2006.

Agradeço ao Professor João Marcos, que sacrificou parte de suas férias para ler minha tese e participar da banca da defesa da minha tese. Ao Professor Paulo Veloso, por ter lido minha tese e ter participado da banca nas condições especiais que permearam os acontecimentos. Ao Professor Marcelino Pequeno pela participação na banca. Também gostaria de agadecer a H. G. Lobo, pelo acompanhamento anímico.

Em especial, quero agradecer à Professora Ana Teresa, minha orientadora, que sempre me ajudou e me motivou nos momentos mais difíceis e me deu ânimo para continuar. A Ana Teresa foi peça importante nesse processo.

Por fim, agradeço aos meus pais, Francisco Ferreira Manço e Maria Lucineide Martins Ferreira, e ao meu avô Francisco Martins da Rocha, aos quais dedico este trabalho.

“CDA— [. . .] Teve satisfação em escrever, esvaziou a alma, está acabado.

[. . .]

Carteiro— [. . .], aqui está o seu volume, não repare os defeitos, ouviu? Esvaziei bastante a alma, tudo não era possível!”

Carlos Drummond de Andrade

Resumo

Neste trabalho, o conceito de Modelo Minimal e seu uso na semântica de certas lógicas são estudados. Nós analisamos o poder expressivo de diversas lógicas que usam o conceito de Modelo Minimal para definir sua relação de satisfação. Os principais teoremas estudados foram o Teorema de Löwenheim-Skolem e o Teorema de Definibilidade de Beth. No Capítulo 1, nós damos algumas motivações e revisamos alguns conceitos básicos de Lógica. No Capítulo 2, nos estudamos a Lógica de Menor Ponto Fixo—LFP. Nós exibimos uma prova de que o Teorema de Beth não vale para LFP. Nós usamos teorias infinitas para provar isso. Utilizando um resultado de Hodkinson para $L^{\omega}_{\omega_1\omega}$, nós mostramos que o Teorema de Beth continua não valendo mesmo para teorias finitas de LFP. Nós continuamos estudando problemas de definibilidade para LFP e demonstramos que, para tipos especiais de definições implícitas formadas por Sistemas Recursivos, que funcionam como definições recursivas em determinados contextos, existe uma definição explícita. Nós promavos ainda que o Teorema de Löwenheim-Skolem Descendente vale para qualquer conjunto de fórmulas de LFP, independentemente de sua cardinalidade. No Capítulo 3, a Circunscrição de McCarthy e as Teorias Circunscritivas Aninhadas de Lifschitz, uma generalização da primeira. Nós abordamos o poder expressivo de Circunscrição e a falha do Teorema de Löwenheim-Skolem Descendente. Nós também investigamos questões de definibilidade no contexto de Circunscrição. Nós encerramos esse capítulo mostrando que as Teorias Circunscritivas Aninhadas possuem poder expressivo comparável com o da Lógica de Segunda-Ordem. No Capítulo 4, nós estendemos uma lógica criada por van Benthem dando origem a duas outras lógicas, a saber, U-MIN e I-MIN. Nós provamos que ambas são equivalentes entre si em poder expressivo e daí em diante chamamos U-MIN de MIN. Nós introduzimos a Lógica Si-MIN de minimalização simultânea e provamos que Si-MIN é equivalente a U-MIN e I-MIN e também à Lógica de Segunda-Ordem. Nós então propomos o o fragmento MIN_{Δ} de MIN, cujo poder expressivo situa-se entre o da Lógica de Segunda-Ordem e o de LFP. No Capítulo 5, nós reunimos nossas conclusões e apontamos trabalhos futuros.

Abstract

In this work, the concept of Minimal Model is studied in connection with its use in the semantics of some logics. We analize the expressive power of several logics which use Minimal Model to define its satisfaction relation. The main theorems we treat here are the Löwenheim-Skolem Theorem and the Beth Definability Theorem. In Chapter 1, we give some motivation on Minimal Elements and review some basic concepts of First-Order Logic. In the Chapter 2, we study the Least Fixed Point Logic—LFP. We gave a proof that the Beth Definability Theorem does not hold for LFP. We use infinite theories to show this. Using a result of Hodkinson for $L^{\omega}_{\omega_1\omega}$, we show that the Beth's Theorem fails even if we restrict ourselves to finite theories also. We continue investigating definability problems for LFP and demonstrate that, for especial kinds of implicit definitions made by Recursive Systems, which works like recursive definitions in specifc contexts, there is an explicit definition. We prove also that the Löwenheim-Skolem Theorem holds for any countable set of LFP-formulas. We extend this result and show that a general version of Downward Löwenheim-Skolem Theorem also holds for sets of LFP-formulas of any cardinality. In the Chapter 3, we investigate the Circumscription of McCarthy and the Nested Abnormality Theories of Lifschiz, a generalization of the former. We discuss about the expressive power of Circumscription and the fail of the Löwenheim-Skolem Theorem. We also investigate definability in the context of Circumscription. We close this chapter prooving that the Nested Abnormality Theories has expressive power comparable to Second-Order Logic's one. In the Chapter 4, we extend a logic of van Benthem giving raise to two other logics, namely, U-MIN and I-MIN. We prove that each of these logics are equivalent to the other in expressive power and thereafter we call U-MIN just MIN. We introduce the Si-MIN Logic of simultaneous minimization and prove that Si-MIN is equivalent to U-MIN and I-MIN and also to Second-Order Logic in expressive power. We propose the MIN_Δ fragment of MIN, whose expressive power lies between those of LFP and Second-Order Logic. In the Chapter 5, we make some conclusions and point futher works.

Sumário

Lista de Definições

Lista de Teoremas

Teoremas

Lemas

Corolários

Capítulo 1

Introdução

A seguir, na Seção 1.1, será feita a apresentação do trabalho, dando motivações e explicando o conteúdo de cada capítulo. Na Seção 1.2, apresentaremos uma revisão dos conceitos básicos de Lógica Matemática.

1.1 Minimalidade

Um dos conceitos mais utilizados nos diversos ramos da matemática é o de relação de ordem, em suas diversas acepções e variações. Ordens parciais, pré-ordens, ordens totais, estritas, densas, completas, etc., são frequentemente encontradas na literatura. Dentre os objetos relacionados por uma ordem, alguns assumem posição de importância conferida por certa propriedade: são os objetos minimais e maximais. A propriedade da qual desfrutam os objetos minimais (respectivamente, maximais), e que lhes dá nome, é a minimalidade (resp. maximalidade).

Como foi observado, quando falamos sobre objetos minimais (ou maximais), estamos subentendendo uma relação de ordem. Definiremos exatamente o que entendemos por relação de ordem. Existem basicamente duas definições iniciais das quais se derivam as demais acrescentando-lhes mais propriedades.

Definição 1.1 (Pré-Ordem) *Seja A um conjunto e $\leq$ uma relação binária sobre A. $\leq$ é uma* pré-ordem *se, e somente se, satisfaz as seguintes propriedades:*

- *para todo $a \in A$, $a \leq a$ (reflexividade);*
- *para todos $a, b, c \in A$, se $a \leq b$ e $b \leq c$, então $a \leq c$ (transitividade).*

Definição 1.2 (Ordem Parcial) *Seja A um conjunto e $\leq$ uma pré-ordem sobre A. $\leq$ é uma* ordem parcial *se, e somente se, satisfaz a seguinte propriedade adicional:*

- *para quaisquer $a, b \in A$, se $a \leq b$ e $b \leq a$, então $a = b$.*

Como podemos observar, uma ordem parcial é uma pré-ordem. Nós a destacamos acima por ser um tipo de pré-ordem que utilizaremos bastante.

A partir de uma ordem parcial ou uma pré-ordem definimos os elementos minimais e maximais.

Definição 1.3 (Elemento Minimal) *Seja A um conjunto, $\leq$ uma pré-ordem sobre A e C um subconjunto de A. Um elemento $c \in C$ é um* elemento minimal de C com relação a $\leq$ *se, e somente se, não existe um elemento $c' \in C$ tal que*

$$c' \leq c \text{ e } c \not\leq c'.$$

O conceito dual de elemento minimal é o de elemento maximal.

Definição 1.4 (Elemento Maximal) *Seja A um conjunto, $\leq$ uma pré-ordem sobre A e C um subconjunto de A. Um elemento $c \in C$ é um* elemento maximal de C com relação a $\leq$ *se, e somente se, não existe um elemento $c' \in C$ tal que*

$$c \leq c' \text{ e } c' \not\leq c.$$

A relação de dualidade entre esses dois conceitos pode ser observada da seguinte forma. Seja A um conjunto, $\leq$ uma pré-ordem sobre A e C um subconjunto de A. A partir de $\leq$, nós definimos a relação $\geq$ tal que, para quaisquer $a, a' \in A$,

$$a \geq a' \text{ se, e somente se } a' \leq a.$$

Imediatamente temos:

Teorema 1.1 (Minimalidade $\times$ Maximalidade) *Seja A um conjunto, $\leq$ uma pré-ordem sobre A, C um subconjunto de A e $\geq$ como definida acima. Um elemento $c \in C$ é minimal de C com relação a $\leq$ se, e somente se, for maximal de C com relação a $\geq$.*

Em geral, uma proposição que envolva os termos "$\leq$" e "minimal" possui uma equivalente em termos de "$\geq$" e "maximal".

Objetos (elementos) minimais e maximais também são bastante utilizados, por exemplo, em Matemática, Ciência da Computação e Lógica.

Em Matemática, mais precisamente em Teoria dos Conjuntos, observamos o aparecimento do termo maximal, por exemplo, na formulação do Lema de Zorn. O Lema de Zorn afirma a existência de elemento maximal em qualquer conjunto parcialmente ordenado não vazio no qual toda cadeia possui um limitante superior. Vejamos as seguintes definições.

Definição 1.5 (Conjunto Parcialmente Ordenado) *Um par $(A, \leq)$ onde A é um conjunto e $\leq$ é uma ordem parcial sobre A é um conjunto parcialmente ordenado.*

Definição 1.6 (Limitantes Superior e Inferior) *Seja A um conjunto, $\leq$ uma pré-ordem sobre A e C um subconjunto de A. Um elemento u (resp. l) de A é um limitante superior (resp. inferior) de C se, e somente se, para todo elemento $c \in C$, $c \leq u$ (resp. $l \leq c$).*

Definição 1.7 (Cadeia) *Seja $(A, \leq)$ um conjunto parcialmente ordenado. Um sub conjunto $C \subseteq A$ é uma cadeia se, e somente se, $(C, \leq_C)$ é um conjunto totalmente ordenado, isto é, $\leq_C$, a restrição de $\leq$ a C, é uma ordem total sobre C.*

Lema 1.2 (Lema de Zorn) *Seja $(A, \leq)$ um conjunto parcialmente ordenado no qual toda cadeia possui um limitante superior. Então A possui um elemento maximal.*

Analogamente, temos um teorema equivalente em termos de elementos minimais.

Lema 1.3 (Lema de Zorn—2ª Versão) *Seja $(A, \leq)$ um conjunto parcialmente ordenado no qual toda cadeia possui um limitante inferior. Então A possui um elemento minimal.*

O Lema de Zorn é de grande relevância em Matemática, principalmente em Teoria dos Conjuntos, sendo equivalente ao Axioma da Escolha, e mesmo em Lógica, onde é utilizado para provar a seguinte versão do Teorema da Completude de Gödel para a Lógica de Primeira-Ordem:

Teorema 1.4 (Teorema da Completude [Gödel, 1929]) *Considere um conjunto de símbolos S (veja Definição 1.11) de cardinalidade qualquer. Seja Γ um conjunto de S-sentenças da lógica de primeira-ordem de cardinalidade qualquer. Seja ϕ uma S-sentença da lógica de primeira-ordem. Então*

$$\Gamma \models \phi \textit{ se, e somente se, } \Gamma \vdash \phi.$$

No Teorema 1.4, a ausência de restrição sobre a cardinalidade do conjunto Γ faz com que seja necessário que valha na teoria de conjuntos subjacente um teorema equivalente ao Lema de Zorn.

Outra utilização de elementos minimais em Matemática se encontra na Teoria das Definições Indutivas. Conforme veremos no Capítulo 2, um conjunto é dito indutivo se coincidir com o *menor* (que é caso particular de minimal) ponto fixo de um operador monótono em $\wp(A) \to \wp(A)$, para algum conjunto A. O Teorema de Knaster-Tarski (veja Teorema A.1) garante a existência de tal menor ponto fixo de funções monótonas sobre reticulados completos (veja Definição A.2). Para ser mais preciso, dada uma função monótona (veja Definição A.3) definida sobre um reticulado completo, Tarski mostra que a subestrutura desse reticulado induzida pelo conjunto dos pontos fixos daquela função monótona é também um reticulado completo [Tar55]. Um reticulado pode ser visto como um conjunto parcialmente ordenado, e, sendo completo, possui menor elemento. No caso dos conjuntos indutivos, a ordem parcial considerada é a relação de subconjunto (ou a sua restrição ao conjunto dos pontos fixos do operador monótono em questão). Mais detalhes sobre reticulados se encontram no Apêndice A e sobre Definições Indutivas no Capítulo 2, Seção 2.1.

Conjuntos indutivos também são utilizados para a criação de lógicas mais expressivas do que a lógica de primeira-ordem. A Lógica de Menor Ponto Fixo (Least Fixed Point Logic—LFP) é uma extensão da lógica de primeira-ordem através do operador de menor ponto fixo, que é utilizado na linguagem de LFP para a formação de predicados interpretados como o menor ponto fixo de certos operadores definidos por fórmulas positivas no domínio de uma estrutura qualquer. Funciona da seguinte forma. Fórmulas (por exemplo, de primeira-ordem) definem operadores sobre o conjunto das partes do domínio de uma estrutura qualquer. Quando a fórmula é positiva o operador definido é monótono. Pelo Teorema de Knaster-Tarski, esse operador possui um menor ponto fixo. Assim, adiciona-se à sintaxe da lógica de primeira-ordem um operador, a saber, **lfp**, para a construção de expressões que são interpretadas como sendo o menor ponto fixo do operador monótono definido por uma fórmula positiva. Isso nos leva a uma lógica, a LFP, que possui poder expressivo maior que o da lógica de primeira-ordem, já que alguns conjuntos indutivos não podem ser definidos nesta última. Mais detalhes sobre a Lógica de Menor Ponto Fixo serão apresentados no Capítulo 2.

Menores pontos fixos também são utilizados em Computação, por exemplo, em Semântica Denotacional de Programas Recursivos [Sch86, Sco82, DSW94]. Strachey e Scott desenvolveram uma semântica para programas através do menor ponto fixo de determinados operadores[1]. A partir de sistemas de equações recursivas (que podem representar, por exemplo, programas recursivos) definimos um conjunto de possíveis interpretações para o programa desejado. Veja o exemplo seguinte:

Exemplo 1.1 *Considere o seguinte programa:*

```
Programa ProdImp
Entrada:  natural x;
Saída:  se x for ímpar, retorna ∏_{i≤x,i ímpar}(i),
se x for par, o programa retorna,
recursivamente, ProdImp(x + 2);

se x = 1, retorna 1
se x é ímpar, retorna x × ProdImp(x − 2)
se x é par, retorna ProdImp(x + 2)
fim do programa.
```

Obviamente, sempre que receber como entrada um número natural par, o programa `ProdImp` *jamais terminará a sua execução, o que significa que a função que o programa* `ProdImp` *computa é uma* função parcial.

O programa `ProdImp` pode ser representado da seguinte forma. Considere uma linguagem formada por símbolos funcionais, símbolos constantes, variáveis e pelo símbolo de igualdade =. Pretende-se que tais símbolos funcionais representem funções parciais e que constantes e variáveis representem elementos em algum domínio previamente definido. O símbolo de igualdade representa o fato de que as expressões à sua esquerda e à sua direita representam o mesmo objeto qualquer que seja a atribuição de variáveis (respeitados os domínios aos quais se aplicam tais variáveis). Dessa forma, é possível, nesta linguagem, haver expressões que não representam (ou referenciam) valor algum. Seja SE um símbolo funcional ternário que, ocorrendo em uma expressão, deverá ter como primeiro argumento uma expressão que represente valores booleanos e como segundo e terceiro argumentos expressões que representem números naturais. Ou seja, sempre que SE ocorrer em uma expressão ocorrerá na forma

$$SE(\alpha, \beta, \gamma). \tag{1.1}$$

[1] Neste texto, o termo "operador" é utilizado como sinônimo de "função," sendo o primeiro mais frequentemente utilizado quando a função tomar como valores objetos de ordem superior, como subconjuntos, funções, conjuntos de predicados, etc.

Se a expressão α representar o valor $Verdadeiro$ (resp. $Falso$), então a Expressão 1.1 representará o mesmo valor que representa a expressão β (resp. γ), caso β (resp. γ) represente algum valor, ou não representará valor algum, se assim também for β (resp. γ). Observe que β ou γ, embora não ambas simultaneamente, podem não representar valor algum e mesmo assim a Expressão 1.1 possuir valor definido. Seja $\acute{I}$ um símbolo funcional unário, cujo argumento deve ser uma expressão que represente números naturais, tal que $\acute{I}(\alpha)$ assumirá valor $Verdadeiro$ caso α represente um número natural ímpar, $Falso$ caso α represente um número natural ímpar, e não assumirá valor algum se assim também for α. Seja UM uma função unária, de valor booleano, tal que $UM(\alpha)$ assume valor $Verdadeiro$ se a expressão α representa o número 1, $Falso$ caso α represente um número natural diferente de 1, e não assumirá valor algum se assim também for α. Sejam $+$ e $-$ funções binárias interpretadas como as operações de soma e subtração de números naturais, exceto pelo fato de que, caso aplicadas a alguma expressão sem valor, assim também o serão. Considere a seguinte equação, onde ProdImp é um símbolo funcional:

$$\text{ProdImp}(x) = SE(UM(x), 1, SE(\acute{I}(x), \text{ProdImp}(x-2), \text{ProdImp}(x+2))). \tag{1.2}$$

A partir de qualquer programa recursivo, pode-se construir uma equação ou um sistema de equações semelhantes à Equação 1.2. Tal sistema de equações é dito recursivo devido ao fato de que os símbolos funcionais que ocorrem do lado esquerdo do símbolo de igualdade também podem ocorrer do lado direito.

A partir do domínio $dom(\text{ProdImp})$ e do contra-domínio $cdom(\text{ProdImp})$ de ProdImp[2], definimos o conjunto

$$dom(\text{ProdImp}) \dashrightarrow cdom(\text{ProdImp}) \tag{1.3}$$

das funções parciais de $dom(\text{ProdImp})$ em $cdom(\text{ProdImp})$. Definimos também a relação de ordem parcial

$$\sqsubseteq_{dom(\text{ProdImp}) \dashrightarrow cdom(\text{ProdImp})}, \tag{1.4}$$

definida como sendo a relação de subconjunto[3], ou seja, se

$$f, g \in dom(\text{ProdImp}) \dashrightarrow cdom(\text{ProdImp}),$$

[2]Observe que ProdImp é um símbolo funcional. No entanto, estamos utilizando as notações $dom(\text{ProdImp})$ e $cdom(\text{ProdImp})$ para denotar o fato de que uma possível interpretação para ProdImp deve ser, obrigatoriamente, uma função parcial de domínio $dom(\text{ProdImp})$ e contra-domínio $cdom(\text{ProdImp})$. Normalmente, os operadores dom e $cdom$, bem como o operador img, são aplicados a funções parciais, ao invés de símbolos funcionais.

[3]Considerando que funções parciais são alguns subconjuntos do produto cartesiano de

então
$$f \sqsubseteq_{dom(\mathrm{ProdImp}) \dashrightarrow cdom(\mathrm{ProdImp})} g$$
se, e somente se,
$$f \subseteq g.$$

Observe, no entanto, que existem diversas, na verdade infinitas, interpretações possíveis para o símbolo funcional ProdImp (isto é, elementos de $dom(\mathrm{ProdImp}) \dashrightarrow cdom(\mathrm{ProdImp})$) que satisfazem a Equação 1.2.

O conjunto parcialmente ordenado
$$\mathfrak{D} = (dom(\mathrm{ProdImp}) \dashrightarrow cdom(\mathrm{ProdImp}), \sqsubseteq_{dom(\mathrm{ProdImp}) \dashrightarrow cdom(\mathrm{ProdImp})}) \tag{1.5}$$
tem como característica principal a propriedade de ser uma *ordem parcial completa* [DSW94, Capítulo 16]. Veja as definições abaixo:

Definição 1.8 (Supremo e Ínfimo) *Seja $(A, \leq)$ um conjunto parcialmente ordenado e C um subconjunto de A. Seja $C^{\geq}$ (resp. $C^{\leq}$) o conjunto dos elementos $a \in A$ tais que, para quaisquer $c \in C$, $a \geq c$ (resp. $a \leq c$). Caso exista, o elemento $s \in A$ que for o menor elemento de $C^{\geq}$, denotado por $\bigsqcup C$, é dito o* supremo *de C em $(A, \leq)$. Também caso exista, o elemento $i \in A$ que for o maior elemento de $C^{\leq}$, denotado por $\bigsqcap C$, é dito o* ínfimo *de C em $(A, \leq)$.*

Definição 1.9 (Ordem Parcial Completa—OPC) *Chamamos um conjunto parcialmente ordenado $(A, \leq)$ de* ordem parcial completa *se, e somente se, qualquer cadeia $C \subseteq A$ possui um supremo em $(A, \leq)$.*

Utilizando o sistema de equações, definimos um operador Φ sobre o conjunto parcialmente ordenado $\mathfrak{D}$ de forma que podemos garantir que todo ponto fixo desse operador é uma solução do sistema de equações, e vice-versa. Seja $\alpha(x)$ a expressão definida como segue:
$$\alpha(x, \mathit{ProdImp}) = \mathit{SE}(\mathit{UM}(x), 1, \mathit{SE}(\acute{I}(x), \mathrm{ProdImp}(x-2), \mathrm{ProdImp}(x+2))). \tag{1.6}$$
Seja $\Phi \in |\mathfrak{D}| \to |\mathfrak{D}|$ definido da seguinte forma:
$$\begin{aligned} \Phi(f) \quad = \quad & \{(a, b) \in dom(\mathrm{ProdImp}) \times cdom(\mathrm{ProdImp}) | \alpha(x, \mathrm{ProdImp}) \\ & \text{representa o valor } b \text{ quando } x \text{ assume o valor } a \text{ e} \\ & \mathrm{ProdImp} \text{ é interpretado como sendo } f\} \end{aligned} \tag{1.7}$$

---

um domínio com um contra-domínio. No caso,
$$dom(\mathrm{ProdImp}) \dashrightarrow cdom(\mathrm{ProdImp}) \subseteq \wp(dom(\mathrm{ProdImp}) \times cdom(\mathrm{ProdImp})).$$

Esse operador possui o atributo especial de ser *contínuo* [DSW94]. Veja definição a seguir.

Definição 1.10 (Operador Contínuo) *Seja $\mathfrak{A} = (A, \sqsubseteq)$ uma ordem parcial completa e $\Phi \in A \to A$. Dizemos que a função Φ é* contínua *se, e somente se, para qualquer cadeia $C \subseteq A$,*

$$\Phi\left(\bigsqcup C\right) = \bigsqcup\{\Phi(c) \in A | c \in C\}.$$

Isso garante a existência de um menor ponto fixo. A Semântica de Strachey-Scott baseia-se em interpretar o sistema de equações recursivas como o menor ponto fixo daquele operador. Novamente se observa o estabelecimento de uma ordem[4] (no caso, entre os pontos fixos do operador em questão) e o uso do elemento minimal (na verdade, do menor elemento).

Em Programação em Lógica, encontramos outra utilização do conceito de minimalidade. Um programa lógico é um conjunto de cláusulas de Horn [Llo87]. A interpretação dos predicados definidos por um programa lógico é dada pelo menor modelo de Herbrand. Isto é, uma ordem sobre os modelos de Herbrand das fórmulas que compõem o programa lógico é definida de forma que dois modelos H_1 e H_2 se relacionam, ou seja, $H_1 \leq H_2$, se a extensão dos predicados definidos pelo programa em H_1 está contida na extensão desses mesmos predicados em H_2. O padrão de cláusulas de Horn das fórmulas que compõem o programa garante a existência de um menor modelo de Herbrand. Esse modelo é a interpretação do programa lógico.

Um dos formalismos baseados em lógica mais estudados em Inteligência Artificial (IA) é a Circunscrição de McCarthy [McC80, McC86]. Usamos Circunscrição a fim de lidar com conhecimento incompleto em problemas de IA da seguinte forma. Com o intuito de representar um determinado *default* em uma teoria, adicionamos a esta um novo predicado, frequentemente chamado *anormal*, e alguns axiomas extras que devem assegurar que o predicado *anormal* contém aqueles elementos que são exceções ao *default*. A Circunscrição escolhe os modelos da teoria em que *abnormal* possui as menores extensões possíveis, isto é, os modelos minimais da teoria. Os modelos minimais dos quais trata a Circunscrição, e que utilizaremos também no

[4] Vale destacar a diferença entre o Teorema de Knaster-Tarski e o teorema de menor ponto fixo utilizado na semântica de Strachey-Scott. Embora ambos utilizem uma ordem parcial, no primeiro esta ordem parcial corresponde a um reticulado completo, o que, junto com a monotonicidade do operador em questão, garante a existência do menor ponto fixo [Tar55]. Já no segundo se exige apenas que a estrutura seja uma ordem parcial completa [DSW94] (um reticulado completo é uma ordem parcial completa, mas nem sempre o contrário é verdade), mas, por outro lado, necessitamos que o operador seja, mais do que monótono, contínuo. O Teorema de Knaster-Tarski é descrito no Apêndice A.

Capítulo 4, estão descritos na Definição 3.6, e a ordem das quais tais modelos são extraídos nas Definições 3.4 e 3.5, no Capítulo 3.

Em [Lif95], Lifschitz estende a Circunscrição através das *Nested Abnormality Theories* (NATs). Enquanto que na Circunscrição de McCarthy apenas teorias (finitas) de primeira-ordem são circunscritas, em NATs, é possível circunscrever um predicado em uma teoria que seja o resultado de uma circunscrição previamente efetuada sobre uma teoria de primeira-ordem, e o resultado dessa segunda circunscrição pode ser novamente circunscrito e assim por diante. Em NATs, axiomas são agrupados em blocos. Cada bloco corresponde a uma circunscrição. Blocos, por sua vez, podem ser agrupados novamente e aninhados em outros blocos. Dessa forma podemos "circunscrever uma circunscrição" previamente feita. Em alguns casos, as NATs se mostram mais adequadas e intuitivas para axiomatizar certos problemas [Lif95]. No Capítulo 3 definiremos precisamente as NATs.

Modelos minimais também estão presentes na lógica MIN(FO) de van Benthem. Em MIN(FO), um operador é adicionado à linguagem a fim de criar novas expressões que serão interpretadas como o menor predicado que satisfaz uma determinada fórmula. Essa fórmula segue o padrão sintático das *condições* PIA [vB05]. Essas condições PIA têm a propriedade de possuir um menor predicado, ou modelo minimal, que as satisfaz (fixadas as interpretações dos demais símbolos). Examinaremos melhor MIN(FO) no Capítulo 4.

Como vimos, objetos minimais são amplamente utilizados em vários ramos da Matemática. Os capítulos que se seguem, e dos quais consiste o presente trabalho, tratam de lógicas que utilizam o conceito de modelo minimal. Tal conceito é utilizado por essas lógicas a fim de lhes conferir aumento ou diferenciação em poder expressivo. O esquema geral é o seguinte. Partimos de uma linguagem básica, por exemplo, a linguagem da lógica de primeira-ordem, e definimos uma nova linguagem $\mathcal{L}$ adicionando-se a seguinte regra ao cálculo de criação das fórmulas bem formadas:

RM Se ϕ é uma fórmula de $\mathcal{L}$ e ι é um conjunto de parâmetros composto por fórmulas e termos de $\mathcal{L}$, então $\mho(\phi, \iota)$ é uma fórmula de $\mathcal{L}$.

O que vai caracterizar as lógicas que estudaremos é o fato de a semântica dessas lógicas utilizar os modelos minimais de ϕ a fim de estipular o valor verdade das fórmulas geradas a partir da regra *RM*. Assim, os modelos de uma fórmula da forma $\mho(\phi, \iota)$ são os elementos do conjunto $\Xi(\mathbb{M}, \iota)$, onde Ξ é uma função que tem como parâmetros a classe $\mathbb{M}$ dos modelos minimais de ϕ e o conjunto ι. Chamaremos operadores que seguem esse esquema de operadores de *minimização*.

Os problemas que abordaremos estão relacionados com o poder expressivo dessas lógicas e a definibilidade de símbolos não-lógicos. No Capítulo 2 introduziremos a Lógica de Menor Ponto Fixo. Exploraremos questões sobre a definibilidade de símbolos não-lógicos. Mostraremos que o Teorema de Beth falha para esta lógica e utilizaremos resultados de Hodkinson [Hod93] sobre a lógica $L^{\omega}_{\omega_1\omega}$ para mostrar que aquele teorema continua falhando mesmo se nos restringirmos a definições implícitas feitas por teorias finitas da Lógica de Menor Ponto Fixo. Posteriormente, realizaremos um estudo de caso sobre a definibilidade de símbolos definidos a partir de certos *sistemas recursivos*, que são certos tipos de definições recursivas em determinado contexto. Veremos que uma restrição do Teorema de Beth vale para símbolos definidos através de sistemas recursivos em teorias de estruturas indutivas. Finalizaremos esse capítulo mostrando que o Teorema de Löwenheim-Skolem Descendente vale para teorias enumeráveis quaisquer da Lógica de Menor Ponto Fixo, bem como sua generalização para teorias de cardinalidade qualquer.

No Capítulo 3, examinaremos a Circunscrição de McCarthy e as *Nested Abnormality Theories* (NATs) de Lifschitz. Após introduzirmos as definições básicas, provaremos alguns teoremas a respeito da expressividade da Circunscrição Paralela de Predicados. Depois falaremos a respeito da falha do Teorema de Löwenheim-Skolem para Circunscrição e exibiremos uma prova da falha desse teorema. Após isso, teceremos algumas observações a respeito da definibilidade do símbolo circunscrito. Estabeleceremos que, quando equivalente a uma teoria de primeira-ordem e ao definir implicitamente o símbolo circunscrito, a circunscrição desse símbolo em uma teoria finita poder ser substituída por, isto é, é logicamente equivalente a, esta teoria finita mais determinada definição explícita do símbolo circunscrito. Por último, apresentaremos as NATs de Lifschitz e mostraremos que, para qualquer teoria finita da lógica de segunda-ordem em um alfabeto S, existe uma NAT em um alfabeto $S' \supset S$, isto é, em um vocabulário estendido, que é uma extensão conservativa da teoria de segunda-ordem inicial. Isso significa que tanto a teoria de segunda-ordem em questão quanto a NAT correspondente provam exatamente os mesmos teoremas escritos no alfabeto S. Mais ainda, a tradução entre fórmulas de segunda-ordem e NATs que construímos para obter tal resultado é tal que tanto o tamanho da NAT desejada quanto o tempo necessário para calculá-la são lineares no tamanho da fórmula de segunda-ordem na forma normal prenex.

No Capítulo 4, introduziremos a lógica MIN(FO) de van Benthem, que utiliza o operador de minimização MIN sobre fórmulas que atendem a certo padrão sintático e que são chamadas de *condições PIA*. van Benthem mostrou que MIN(FO) e a Lógica de Menor Ponto Fixo são equivalentes em poder expressivo, estabelecendo uma ligação entre a Lógica de Menor Ponto Fixo e os operadores de minimização. Na Seção 4.2, nós definiremos as lógicas U-MIN e I-MIN que nós criamos a fim de estender a aplicação do operador MIN a qualquer fórmula, obtendo uma lógica mais expressiva. Enquanto na lógica U-MIN o operador MIN^u realiza uma operação de união nos modelos minimais de uma fórmula, o operador MIN^i de I-MIN realiza uma operação de interseção. Nós mostraremos então que U-MIN e I-MIN são equivalentes em poder expressivo. Adotaremos então o nome MIN para lógica U-MIN. Em seguida, estendemos a lógica MIN a fim de podermos minimizar vários predicados simultaneamente e mostraremos que essa extensão não é acompanhada de aumento de poder expressivo. Nós então provaremos que MIN é equivalente à lógica de segunda-ordem em poder expressivo. Na última seção, exibiremos um fragmento de MIN, a lógica MIN_{Δ}, que é menos expressiva que a lógica de segunda-ordem, mais expressiva que a Lógica de Menor Ponto Fixo e que contém os operadores booleanos e quantificadores de primeira-ordem.

Finalizaremos este trabalho no Capítulo 5 onde reuniremos nossas conclusões e apontaremos direções para possíveis trabalhos futuros.

1.2 Preliminares

Nesta seção serão detalhadas as notações básicas bem como as definições utilizadas no restante do texto. Nós seguiremos a notação de [EFT94]. Assumiremos também que o leitor possui alguma familiaridade com os conceitos fundamentais de lógica e matemática, tais como linguagem formal, lógica de primeira ordem e de segunda ordem e indução matemática. O leitor experiente pode seguir direto para a leitura do Capítulo 2.

1.2.1 Lógicas de Primeira-Ordem e Segunda-Ordem

Inicialmente, faremos uma breve revisão da lógica de primeira ordem. Começaremos com a definição de alfabeto para uma linguagem de primeira ordem.

Definição 1.11 (Alfabeto) *Um* alfabeto para uma linguagem de primeira ordem *é um conjunto contendo os seguintes símbolos:*

1. *um conjunto* $\{v_0, v_1, v_2, \ldots\}$ *de variáveis;*
2. $\neg, \wedge, \vee, \rightarrow, \leftrightarrow$ *(os símbolos lógicos);*
3. $\forall, \exists$ *(os quantificadores universal e existencial);*
4. $\equiv$ *(símbolo de igualdade);*

5. *), (, ,, (símbolos de pontuação);*

6. *(a) para cada $n \geq 1$ um conjunto (possivelmente vazio) de símbolos relacionais n-ários;*

   (b) para cada $n \geq 1$ um conjunto (possivelmente vazio) de símbolos funcionais n-ários;

   (c) um conjunto (possivelmente vazio) de símbolos constantes.

Chamaremos de $\mathcal{A}$ o conjunto dos elementos correspondentes aos itens de 1 a 5 da Definição 1.11, e de S o conjunto dos símbolos de 6. Dado S, definimos o alfabeto $\mathcal{A}_S := \mathcal{A} \cup S$ e chamamos S de *conjunto de símbolos.* A seguir definiremos o *termos* da linguagem de primeira ordem.

Definição 1.12 (Conjunto dos Termos T^S) *O conjunto T^S dos S-*termos *é o conjunto das seqüências de símbolos de S dadas pelo seguinte* cálculo de termos*:*

$\dfrac{}{v_i}$ para cada variável v_i; $\qquad \dfrac{}{c}$ para cada constante $c \in S$;

$\dfrac{t_1, \ldots, t_n}{ft_1 \ldots t_n}$ se $f \in S$ e f n-ário.

As fórmulas da linguagem de primeira-ordem baseada em S são definidas como se segue:

Definição 1.13 (A Linguagem de Primeira-Ordem L^S) *A* linguagem de primeira-ordem baseada em um conjunto de símbolos S *é o conjunto L^S composto pelas S-*fórmulas*, que são seqüências de símbolos de $\mathcal{A}_S$ obtidas através do seguinte* cálculo[5] de fórmulas*:*

[5]Na verdade, as quatro últimas regras mostradas na Definição 1.12 são *esquemas de regras.* As regras propriamente ditas são instância dos esquemas onde ϕ e ψ são seqüências de símbolos do alfabeto. Veremos, no Capítulo 2, que o conjunto de fórmulas L^S é o conjunto indutivamente definido pelo conjunto das regras codificadas pelo cálculo da Definição 1.12.

$\dfrac{}{t_1 \equiv t_2}$ onde t_1 e t_2 são S-termos;

$\dfrac{}{Rt_1 \ldots t_n}$ onde $t_1, \ldots, t_n$ são S-termos e $R \in S$, R n-ário;

$\dfrac{\phi}{\neg\phi}$; $\quad \dfrac{\phi \quad \psi}{(\phi * \psi)}$ para $* \in \{\vee, \wedge, \rightarrow, \leftrightarrow\}$;

$\dfrac{\phi}{\forall v_i \phi}$ para variável cada v_i; $\quad \dfrac{\phi}{\exists v_i \phi}$ para variável cada v_i.

Um fragmento importante da linguagem é aquele formado pelas fórmulas positivas em um determinado símbolo predicativo.

Definição 1.14 (Fórmula Positiva em P) *Uma $S \cup P$-fórmula ϕ é positiva em P se qualquer ocorrência de P em ϕ ocorre dentro do escopo de uma quantidade par de negações (isto é, do conectivo $\neg$).*

Por conveniência utilizaremos as letras $x, y, z, \ldots$, indexadas ou não, para denotar variáveis. Também eliminaremos os parêntesis mais externos de uma fórmula qualquer. Assumiremos ainda certa relação de precedência entre os conectivos e quantificadores a fim de eliminar parêntesis internos. Dessa forma, quantificadores possuem precedência maior que todos os conectivos, seguidos pela negação ($\neg$), depois pela conjunção ($\wedge$) e pela disjunção ($\vee$), que possuem mesmo grau de precedência, e por último a implicação ($\rightarrow$) e a equivalência ($\leftrightarrow$), ambos com a mesma precedência.

Dizemos que a variável x *ocorre livre* na fórmula ϕ se existe uma ocorrência dessa variável fora do escopo de qualquer quantificador da forma $\forall x$ ou $\exists x$. Se x ocorre livre em ϕ dizemos que x é uma *variável livre* de ϕ. Quando escrevemos $\phi(v_1, \ldots, v_n)$, indicamos que as variáveis $v_1, \ldots, v_n$ podem ou não ocorrer livres em ϕ, mas estão destacadas na notação porque são importantes no contexto. Uma S-fórmula que não possui variáveis livres é chamada *S-sentença.*

Denotamos por L_i^S, $i \geq 0$, o subconjunto de L^S composto das fórmulas cujas variáveis livres estão entre $\{v_0, \ldots, v_{i-1}\}$.

Apresentaremos agora a semântica da lógica de primeira ordem. A fim de darmos significado às expressões da linguagem, devemos interpretar os elementos da mesma. A seguir definimos o que é uma *estrutura*.

Definição 1.15 (S-Estrutura) *Uma S-estrutura $\mathfrak{A}$ é um par $(A, \mathfrak{a})$ composto por um conjunto A, chamado de* universo *ou* domínio *de $\mathfrak{A}$ e um mapeamento $\mathfrak{a}$ sobre S que leva:*

- *cada símbolo relacional $R \in S$ n-ário a uma relação $R^{\mathfrak{A}} \subseteq A^n$;*
- *cada símbolo funcional $f \in S$ n-ário a uma função $f^{\mathfrak{A}} : A^n \rightarrow A$;*
- *cada símbolo constante $c \in S$ a um elemento $c^{\mathfrak{A}} \in A$.*

Para simplificar a notação, representaremos uma S-estrutura $\mathfrak{A}$ como uma tupla cujo primeiro elemento é o domínio seguido pelos elementos que compõem a imagem de $\mathfrak{a}$. Por exemplo, se $S = \{R_1, R_2, f, c\}$, então representamos $\mathfrak{A}$ por $(A, R_1^{\mathfrak{A}}, R_2^{\mathfrak{A}}, f^{\mathfrak{A}}, c^{\mathfrak{A}})$. Utilizaremos normalmente caracteres góticos para nomear estruturas e seu correspondente em caractere romano para nomear o domínio.

A seguir definiremos o reduto de uma estrutura e isomorfismo entre estruturas.

Definição 1.16 (S-Reduto $\mathfrak{A}'|_S$ de $\mathfrak{A}'$) *Seja $\mathfrak{A} = (A, \mathfrak{a})$ uma S-estrutura e $\mathfrak{A}' = (A', \mathfrak{a}')$ uma S'-estrutura com $S \subseteq S'$. Dizemos que $\mathfrak{A}$ é um S-*reduto *de $\mathfrak{A}'$ (e nesse caso $\mathfrak{A}'$ é uma S'-*expansão *de $\mathfrak{A}$) se $A = A'$ e $\mathfrak{a}$ e $\mathfrak{a}'$ concordam em S, isto é, para cada elemento $r \in S$, $r^{\mathfrak{A}} = r^{\mathfrak{A}'}$. Escrevemos $\mathfrak{A} = \mathfrak{A}'|_S$.*

Definição 1.17 (Isomorfismo entre Estruturas) *Dadas duas S-estruturas $\mathfrak{A}$ e $\mathfrak{B}$ dizemos que $\mathfrak{A}$* é isomórfica a *$\mathfrak{B}$ (escreve-se $\mathfrak{A} \cong \mathfrak{B}$) se existe uma função bijetiva $\pi : A \rightarrow B$ tal que:*

- *para todo símbolo relacional n-ário $R \in S$ e $a_1, \ldots, a_n \in A$ temos $R^{\mathfrak{A}}a_1 \ldots a_n$ sss[6] $R^{\mathfrak{B}}\pi(a_1) \ldots \pi(a_n)$;*
- *para todo símbolo funcional n-ário $f \in S$ e $a_1, \ldots, a_n \in A$ temos $\pi(f^{\mathfrak{A}}a_1 \ldots a_n) = f^{\mathfrak{B}}\pi(a_1) \ldots \pi(a_n)$;*
- *para todo símbolo constante $c \in S$ temos $\pi(c^{\mathfrak{A}}) = c^{\mathfrak{B}}$.*

Nós chamamos π de isomorfismo entre *$\mathfrak{A}$ e $\mathfrak{B}$.*

A fim de interpretarmos as variáveis livres que possivelmente ocorrerão nas fórmulas, definiremos *assinalamentos de variáveis*.

Definição 1.18 (Assinalamento de Variáveis) *Um* assinalamento em *A é uma função $\beta : \{v_0, v_1, v_2, \ldots\} \rightarrow A$ que leva cada variável v_i em um elemento $\beta(v_i) \in A$. Dado um assinalamento β, uma variável x e um elemento $a \in A$, definimos o assinalamento $\beta\frac{a}{x}$ como sendo:*

$$\beta\frac{a}{x}(v_i) = \begin{cases} \beta(v_i) & \textit{se } v_i \neq x \\ a & \textit{se } v_i = x \end{cases}$$

[6]Escrevemos "sss" como abreviação para "se, e somente se."

Uma S-estrutura juntamente com um assinalamento para as variáveis formam uma S-interpretação.

Definição 1.19 (S-Interpretação) *Uma S-interpretação $\mathfrak{I} = (\mathfrak{A}, \beta)$ é um par composto por uma S-estrutura $\mathfrak{A}$ e um assinalamento β em A. Dados uma S-interpretação $\mathfrak{I} = (\mathfrak{A}, \beta)$, $a \in A$ e uma variável x, definimos a interpretação $\mathfrak{I}\frac{a}{x} = (\mathfrak{A}, \beta\frac{a}{x})$.*

Agora que interpretamos variáveis, constantes, funções e relações, podemos associar elementos aos termos e valores de verdade às fórmulas da linguagem. Primeiro trataremos dos termos.

Definição 1.20 (A Interpretação $\mathfrak{I}(t)$ de t por $\mathfrak{I}$) *Dada uma S-interpretação $\mathfrak{I}$, o elemento $\mathfrak{I}(t)$ (ou $t^{\mathfrak{I}}$) $\in A$ é definido para todo S-termo t recursivamente da seguinte forma:*

$$
\begin{array}{lcl}
\mathfrak{I}(x) & = & \beta(x) \\
\mathfrak{I}(c) & = & c^{\mathfrak{A}} \\
\mathfrak{I}(ft_1 \ldots t_n) & = & f^{\mathfrak{A}}\mathfrak{I}(t_1) \ldots \mathfrak{I}(t_n), \textit{ para } f \textit{ } n\textit{-ário.}
\end{array}
$$

Agora estamos aptos a definir a relação de satisfatibilidade entre uma interpretação e uma fórmula.

Definição 1.21 (A Relação de Satisfatibilidade $\models$) *Definimos a relação de satisfatibilidade $\models$ entre S-interpretações e S-fórmulas recursivamente da seguinte forma:*

$$
\begin{array}{lll}
\mathfrak{I} \models Rt_1, \ldots, t_n & \textit{sss} & R^{\mathfrak{A}}\mathfrak{I}(t_1) \ldots \mathfrak{I}(t_n) \\
\mathfrak{I} \models t_1 \equiv t_2 & \textit{sss} & \mathfrak{I}(t_1) = \mathfrak{I}(t_1) \\
\mathfrak{I} \models \neg\phi & \textit{sss} & \textit{não } \mathfrak{I} \models \phi \\
\mathfrak{I} \models (\phi \wedge \psi) & \textit{sss} & \mathfrak{I} \models \phi \textit{ e } \mathfrak{I} \models \psi \\
\mathfrak{I} \models (\phi \vee \psi) & \textit{sss} & \mathfrak{I} \models \phi \textit{ ou } \mathfrak{I} \models \psi \\
\mathfrak{I} \models (\phi \rightarrow \psi) & \textit{sss} & \textit{se } \mathfrak{I} \models \phi \textit{ então } \mathfrak{I} \models \psi \\
\mathfrak{I} \models (\phi \leftrightarrow \psi) & \textit{sss} & \mathfrak{I} \models \phi \textit{ se e somente se } \mathfrak{I} \models \psi \\
\mathfrak{I} \models \forall x\phi & \textit{sss} & \textit{para todo } a \in A,\ \mathfrak{I}\frac{a}{x} \models \phi \\
\mathfrak{I} \models \exists x\phi & \textit{sss} & \textit{existe } a \in A,\ \mathfrak{I}\frac{a}{x} \models \phi.
\end{array}
$$

Definiremos agora a lógica de segunda ordem. Primeiramente definiremos a linguagem da lógica de segunda ordem.

Definição 1.22 (A Linguagem da Lógica de Segunda-Ordem) *Definimos a linguagem da lógica de segunda-ordem estendendo a da lógica de primeira ordem através da adição, para cada natural n, de um conjunto de* variáveis relacionais n-*árias* $\{X_{n0}, X_{n1}, X_{n2}, \ldots\}$ *ao alfabeto e das seguintes regras ao cálculo de formação das fórmulas:*

$$\frac{}{Xt_1, \ldots, t_n} \quad \textit{para cada variável relacional } X \textit{ } n\textit{-ária e } S\textit{-termos } t_1, \ldots, t_n;$$

$$\frac{\phi}{\forall X(\phi)} \quad \textit{para cada variável relacional } X.$$

Estendemos a definição de assinalamento de variáveis para incorporar as variáveis de segunda ordem.

Definição 1.23 (Assinalamento de Segunda-Ordem) *Chamamos de* assinalamento de segunda ordem β em A *uma função sobre o conjunto de todas as variáveis (de primeira e segunda ordem) que leva variáveis de primeira ordem a elementos de* A *e variáveis relacionais* n*-árias de segunda ordem a relações* n*-árias sobre* A *(ou subconjuntos de* A^n*). Dados um assinalamento* β*, uma variável relacional* n*-ária* X *e uma relação* n*-ária* $C \subset A^n$*, definimos* $\beta\frac{C}{X}$ *como sendo o assinalamento que leva* X *a* C *e coincide com* β *nas demais variáveis.*

De forma análoga definimos S*-interpretação de segunda-ordem* para um conjunto de símbolos S como um par $\mathfrak{I} = (\mathfrak{A}, \beta)$ onde $\mathfrak{A}$ é uma S-estrutura e β um assinalamento de segunda ordem. Se X é uma variável relacional e $\mathfrak{I}$ é uma interpretação, então $X^{\mathfrak{I}} = \mathfrak{I}(X) = \beta(X)$. Definimos ainda $\mathfrak{I}\frac{C}{X} = (\mathfrak{A}, \beta\frac{C}{X})$ para uma variável relacional n-ária X e uma relação n-ária C em A. A seguir estendemos a relação de satisfação relacionando interpretações de segunda ordem com fórmulas de segunda ordem.

Definição 1.24 (Satisfatibilidade de Segunda-Ordem) *Define-se relação de satisfação* $\models$ *entre* S*-interpretações e* S*-fórmulas de segunda-ordem a partir da Definição 1.21 adicionando as seguintes cláusulas:*

$$\begin{array}{lll}
\mathfrak{I} \models Xt_1, \ldots, t_n & \textit{sss} & \mathfrak{I}(X)\mathfrak{I}(t_1)\ldots\mathfrak{I}(t_n), \textit{ (ou } (\mathfrak{I}(t_1), \ldots, \mathfrak{I}(t_n)) \in \mathfrak{I}(X)); \\
\mathfrak{I} \models \forall X(\phi) & \textit{sss} & \textit{para todo } C \subseteq A^n \ \mathfrak{I}\frac{C}{X} \models \phi; \\
\mathfrak{I} \models \exists X(\phi) & \textit{sss} & \textit{existe } C \subseteq A^n \ \mathfrak{I}\frac{C}{X} \models \phi;
\end{array}$$

onde X *é uma variável relacional* n*-ária.*

Se $\overline{X} = X_1, \ldots, X_n$ e $\overline{x} = x_1, \ldots, x_n$ são tuplas de variáveis relacionais e de primeira ordem respectivamente, escreveremos $\phi(\overline{X}, \overline{x})$ para destacá-las em um determinado contexto, podendo elas ocorrer ou não na fórmula ϕ. Se $\overline{X}$ contém todas as variáveis relacionais de ϕ, $\overline{x}$ contém todas as variáveis elementares (de primeira ordem) de ϕ, $\overline{U}$ é uma tupla de relações sobre A do mesmo tamanho de $\overline{X}$ e é *compatível* com $\overline{X}$ (isto é, o i-ésimo símbolo relacional de $\overline{X}$ tem a mesma aridade da i-ésima relação de $\overline{U}$) e $\overline{a}$ é uma tupla de elementos de A com o mesmo tamanho de $\overline{x}$, escrevemos

$$\mathfrak{A} \models \phi(\overline{X}, \overline{x})[\overline{U}, \overline{a}]$$

ou

$$(\mathfrak{A}, \overline{U}) \models \phi(\overline{X}, \overline{x})[\overline{a}]$$

para indicar que para qualquer interpretação $\mathfrak{I}$ tal que $\mathfrak{I}(X_i) = U_i$ e $\mathfrak{I}(x_i) = a_i$,

$$\mathfrak{I} \models \phi(\overline{X}, \overline{x}).$$

Se $\mathfrak{I}$ é uma S-interpretação, ϕ é uma S-fórmula e $\mathfrak{I} \models \phi$, dizemos que $\mathfrak{I}$ é um modelo de ϕ. Quando ψ é uma sentença, escrevemos $\mathfrak{A} \models \psi$ para dizer que a S-estrutura $\mathfrak{A}$ é modelo de ϕ, isto é, para toda (ou, equivalentemente, existe uma) S-interpretação $\mathfrak{I} = (\mathfrak{A}, \beta)$, $\mathfrak{I} \models \phi$.

1.2.2 Sistema Lógico

Nos capítulos que seguem, trabalharemos com várias lógicas ou *sistemas lógicos* diferentes. Definiremos agora o que é um sistema lógico (de acordo com a definição em [EFT94]).

Definição 1.25 (Sistema Lógico) *Um* sistema lógico *(ou uma lógica) é um par $\mathcal{L} = (L, \models_{\mathcal{L}})$ onde L é uma função que associa a cada conjunto de símbolos S o conjunto $L(S)$ das S-sentenças de $\mathcal{L}$ e $\models_{\mathcal{L}}$ é uma relação binária com as seguintes condições:*

1. *Se $S_0 \subseteq S_1$ então $L(S_0) \subseteq L(S_1)$;*

2. *Se $\mathfrak{A} \models_{\mathcal{L}} \phi$ então para algum S, $\mathfrak{A}$ é uma S-estrutura e $\phi \in L(S)$;*

3. *(Propriedade do Isomorfismo) Se $\mathfrak{A} \cong \mathfrak{B}$ então $\mathfrak{A} \models_{\mathcal{L}} \phi$ sss $\mathfrak{B} \models_{\mathcal{L}} \phi$;*

4. *(Propriedade do Reduto) Se $S \subseteq S'$, $\phi \in L(S)$ e $\mathfrak{A}$ é uma S'-estrutura, então*
$$\mathfrak{A} \models_{\mathcal{L}} \phi \;\; sss \;\; \mathfrak{A}|_S \models_{\mathcal{L}} \phi.$$

Definição 1.26 (Lógica Booleana) *Uma lógica $\mathcal{L}$ é dita* booleana *(escrevemos* $\mathbf{Boole}(\mathcal{L})$*) se satisfaz as seguintes propriedades:*

- *Dados S e $\phi \in L(S)$ existe $\chi \in L(S)$ tal que para toda S-estrutura $\mathfrak{A}$:*
$$\mathfrak{A} \models_{\mathcal{L}} \phi \;\; sss \;\; \mathfrak{A} \not\models_{\mathcal{L}} \chi$$

Definição 1.27 (Os Conjuntos $Th^{\mathcal{L}}_{S'}(\mathbb{C})$ e $Th^{\mathcal{L}}_{S'}(T)$) *Sejam $\mathcal{L}_1$ e $\mathcal{L}_2$ sistemas lógicos, S e S' conjuntos de símbolos (isto é, alfabetos) tais que $S' \subseteq S$ e $\mathbb{C}$ uma classe de S-estruturas. Escrevemos $Th^{\mathcal{L}_1}_{S'}(\mathbb{C})$ para denotar o conjunto das S'-sentenças de $\mathcal{L}_1$ que são satisfeitas por todas as estruturas em $\mathbb{C}$. Seja T um conjunto de S-sentenças de $\mathcal{L}_1$ e $Mod(T)$ a classe dos S-modelos de T. Escrevemos $Th^{\mathcal{L}_2}_{S'}(T)$ para denotar $Th^{\mathcal{L}_2}_{S'}(Mod(T))$. Quando estiver claro a partir do contexto, eliminaremos os sobrescritos $\mathcal{L}_1$ e $\mathcal{L}_2$ e os subscritos S e S'.*

Definição 1.28 (Classe Δ-Elementar) *Se $\mathbb{C}$ é a classe dos modelos de alguma sentença de primeira-ordem ϕ, então $\mathbb{C}$ é dita* elementar. *Se $\mathbb{C}$ é a classe dos modelos de alguma teoria de primeira-ordem Γ, então $\mathbb{C}$ é dita Δ-elementar.*

Definição 1.29 (Fórmulas Equivalentes) *Sejam $\mathcal{L}_1$ e $\mathcal{L}_2$ sistemas lógicos. Seja $\phi \in \mathcal{L}_1$ e $\psi \in \mathcal{L}_2$. Escrevemos $\phi \equiv \psi$ para afirmar que ϕ e ψ possuem os mesmos modelos.*

Definição 1.30 (Equivalência *modulo* $\mathcal{L}$, $\mathfrak{A} \equiv_{\mathcal{L}} \mathfrak{B}$) *Seja $\mathcal{L}$ um sistema lógico e $\mathfrak{A}$ e $\mathfrak{B}$ S-estruturas. Escrevemos $\mathfrak{A} \equiv_{\mathcal{L}} \mathfrak{B}$ para afirmar que $\mathfrak{A}$ e $\mathfrak{B}$ satisfazem as mesmas $\mathcal{L}$ sentenças. Quando estiver claro pelo contexto o sistema lógico em consideração no momento, omitiremos o subscrito $\mathcal{L}$.*

1.2.3 As classes Π, Σ e Δ

Definiremos agora as classes de sentenças Π^i_n, Σ^i_n e Δ^i_n, para $i, n \in N$.

Definição 1.31 (Hierarquias Π^i_n, Σ^i_n e Δ^i_n) *A classe de fórmulas denotada por $\Pi^i_0 = \Delta^i_0 = \Sigma^i_0$, $i \in N$ é classe das fórmulas equivalentes a alguma fórmula da lógica de i^a-ordem.*

A classe de fórmulas denotada por Π^i_{n+1} é formada pelas fórmulas equivalentes a alguma fórmula da forma $\forall X_1 \forall X_m \phi$, $m \in N$, onde cada X_j, $j \in N$, é uma variável de $i + 1^a$-ordem e ϕ é uma fórmula em Σ^i_n. A classe de fórmulas denotada por Σ^i_{n+1} é formada pelas fórmulas equivalentes a alguma fórmula da forma $\exists X_1 \exists X_m \psi$ $m \in N$, onde cada X_j, $j \in N$, é uma variável de $i + 1^a$-ordem e ψ é uma fórmula em Π^i_n. A classe de fórmulas denotada por Δ^i_{n+1} é a classe das fórmulas que estão em Π^i_{n+1} e em Σ^i_{n+1}.

Capítulo 2

Lógica de Menor Ponto Fixo

Neste capítulo, introduziremos a Lógica de Menor Ponto Fixo. Na Seção 2.1, falaremos sobre operadores monótonos sobre conjuntos e definições indutivas, que formam a base da Lógica de Menor Ponto Fixo (LFP) e de outras lógicas de ponto fixo. Na Seção 2.2 definiremos a Lógica de Menor Ponto Fixo e a Lógica de Menor Ponto Fixo Simultâneo. Na Seção 2.3 mostraremos a falha do Teorema de Beth para LFP no caso geral e provaremos um corolário de um teorema de Hodkinson que mostra que o Teorema de Beth também falha se nos restringirmos apenas a teorias finitas de LFP. Finalmente, na Seção 2.4 mostraremos que o teorema de Löwenheim-Skolem também vale para conjuntos arbitrários contáveis de LFP-sentenças, bem como sua generalização para conjuntos de cardinalidade arbitrária.

2.1 Operadores Monótonos e Definições Indutivas

O estudo das Definições Indutivas teve considerável desenvolvimento nos anos 70 com os trabalhos de Moschovakis (e.g. [Mos74]), Aczel [Acz77] e outros no contexto da Teoria da Recursão Generalizada (veja [Acz77] para referências). Utilizamos indução para definir conjuntos da seguinte forma (veja [Acz77]):

Definição 2.1 (Regras e Conjuntos Fechados) *Uma* regra *em um conjunto A é um par (X, x) onde $X \subseteq A$ e $x \in A$. Seja Υ um conjunto de regras em A. Dizemos que um subconjunto $A' \subseteq A$ é* Υ-fechado *se, para toda regra $(X, x) \in \Upsilon$, se $X \subseteq A'$ então $x \in A'$.*

Definição 2.2 (Conjunto Indutivamente Definido) *O conjunto*

$$I(\Upsilon) = \bigcap \{X | X \text{ é } \Upsilon\text{-fechado}\}$$

é o conjunto indutivamente definido *por* Υ*.*

O conjunto das instâncias dos esquemas de regras das Definições 1.12 e 1.13 definem indutivamente os conjuntos T^S e L^S de S-termos e S-fórmulas, respectivamente.

Definições indutivas estão estreitamente relacionadas com operadores monótonos. Veremos abaixo que o teorema de Knaster-Tarski permite uma caracterização daquelas definições com esses operadores. Para tanto, consideraremos, de agora em diante, para cada conjunto A, o reticulado completo $(\wp(A), \subseteq)$ (veja o Apêndice A, Seção A.1). Seja A um conjunto e $F : \wp(A) \to \wp(A)$ um operador. Considere as seguintes definições (definições análogas para reticulados em geral são encontradas no Apêndice A):

Definição 2.3 (Ponto Fixo) *Um conjunto $X \subseteq A$ é um ponto fixo de F se $X = F(X)$. Um ponto fixo X é dito menor se $X \subseteq Y$ para todo ponto fixo Y de F.*

Definição 2.4 (Operador Monótono) *Um operador $F : \wp(A) \to \wp(A)$ é dito monótono se, para todos $X, Y \subseteq A$, se $X \subseteq Y$, então $F(X) \subseteq F(Y)$.*

Pelo Teorema de Knaster-Tarski (veja Apêndice A), todo operador monótono possui um menor ponto fixo.

Teorema 2.1 (Knaster-Tarski [Tar55]) *Seja $F : \wp(A) \to \wp(A)$ um operador monótono. Então F possui um menor ponto fixo* $\mathbf{lfp}(F) = \bigcap\{X | X = F(X)\}$*.*

Conforme observa Aczel em [Acz77], a todo conjunto de regras Υ corresponde um operador monótono F tal que $\mathbf{lfp}(F) = I(\Upsilon)$. Para tanto, dado um conjunto de regras Υ definimos o operador F tal que

$$F(Y) = \{x \in A | (X, x) \in \Upsilon \text{ e } X \subseteq Y\},$$

e dado um operador monótono F, definimos o conjunto de regras Υ tal que

$$\Upsilon = \{(X, x) | X \subseteq A \text{ e } x \in F(X)\}.$$

Uma outra maneira de encontrar um ponto fixo de um operador monótono é através da seguinte seqüência de conjuntos X^α, definidos para cada ordinal α:

$$\begin{aligned} X^0 &:= \emptyset, \\ X^{\alpha+1} &:= F(X^\alpha), \\ X^\lambda &:= \bigcup_{\mu<\lambda} X^\mu, \text{ para } \lambda \text{ limite.} \end{aligned}$$

Se F for monótono, a seqüência de conjuntos X^α forma uma cadeia crescente e atinge um ponto fixo em algum *ordinal de fechamento* $\mathbf{cl}(F)$ (isto é, $\mathbf{cl}(F)$ é o menor ordinal α tal que $X^\alpha = X^{\alpha+1}$). Nós chamamos o ponto fixo alcançado por essa sequência de conjuntos de *ponto fixo indutivo* de F [Kre02b] e o denotamos por X^∞. O conjunto X^α é o *estágio* α da indução em F.

O seguinte teorema, também de Knaster-Tarski, garante que os pontos fixos menor e indutivo de um operador monótono coincidem.

Teorema 2.2 (Knaster-Traski) *Seja* $F : \wp(A) \to \wp(A)$ *um operador monótono. Então* $\mathbf{lfp}(F) = X^\infty$.

Uma *definição indutiva* é, portanto, uma definição feita através do menor ponto fixo de um operador monótono.

Definição 2.5 *Dado um operador monótono* $F : \wp(A) \to \wp(A)$, *o* conjunto indutivamente definido por F *é o conjunto* $\mathbf{lfp}(F)$

Na próxima seção veremos como utilizar definições indutivas para enriquecer a linguagem da lógica de primeira-ordem e obter uma lógica mais expressiva.

2.2 Lógica de Menor Ponto Fixo

A idéia de utilizar definições indutivas a fim de criar linguagens mais poderosas do que a linguagem da álgebra relacional surgiu no contexto da Teoria dos Bancos de Dados [DG02]. Em [AU79], foi sugerido estender a linguagem da álgebra relacional a fim de incorporar um operador para a construção de relações definidas como o ponto fixo de operadores monótonos. Em [CH82] é introduzida uma linguagem de consulta que restringe a aplicação de tal operador a fórmulas positivas (veja a Definição 1.14). Observe que nesse contexto (o de Banco de Dados) estamos nos limitando a relações finitas, isto é, de extensão finita, o que significa que os modelos considerados na avaliação das consultas escritas, quer seja na linguagem da álgebra relacional quer seja em sua versão com pontos fixos, são modelos finitos. Aqui abordaremos o caso dos modelos quaisquer (tanto finitos como infinitos—veja [DG02, Kre02b]). Veremos como essas idéias podem ser aplicadas a fim de dar origem uma lógica mais poderosa do que a lógica de primeira ordem.

Considere uma S-fórmula $\phi(\overline{Q}, X, \overline{x}, \overline{y})$ de, por exemplo, primeira-ordem para algum alfabeto S, onde $\overline{Q} = Q_1, \ldots, Q_m$ é uma tupla de variáveis relacionais, X é uma variável relacional de aridade n e $\overline{x} = x_1, \ldots, x_n$

e $\overline{y}$ são tuplas de variáveis de primeira ordem. Seja $\mathfrak{A}$ uma S-estrutura, $\overline{\mathbf{Q}} = \mathbf{Q}_1, \ldots, \mathbf{Q}_m$ uma tupla de predicados sobre A que interpretam $\overline{Q}$ (isto é, tal que $\mathbf{Q}_i$ e Q_i têm a mesma aridade) e $\overline{b}$ uma tupla de elementos de A com o mesmo tamanho de $\overline{y}$. Podemos definir o operador $\Phi^{\phi}_{\overline{\mathbf{Q}},\bar{b}} : \wp(A^n) \to \wp(A^n)$ como:

$$\Phi^{\phi}_{\overline{\mathbf{Q}},\bar{b}}(\mathbf{X}) = \{\overline{a} \in A^n | (\mathfrak{A}, \overline{\mathbf{Q}}, \mathbf{X}) \models \phi(\overline{Q}, X, \overline{x}, \overline{y})[\overline{a}, \overline{b}]\},$$

para todo $\mathbf{X} \subseteq A^n$.

A fórmula $\phi(\overline{Q}, X, \overline{x}, \overline{y})$ é positiva em X se todo átomo $X(\overline{t})$ ocorre em $\phi(\overline{Q}, X, \overline{x}, \overline{y})$ dentro do escopo de um número par de negações (assumindo que $\phi(\overline{Q}, X, \overline{x}, \overline{y})$ está escrita apenas com os conectivos $\vee$, $\wedge$ e $\neg$). Quando a fórmula $\phi(\overline{Q}, X, \overline{x}, \overline{y})$ é positiva em X o operador $\Phi^{\phi}_{\overline{\mathbf{Q}},\bar{b}}$ é monótono e, pelo Teorema de Knaster-Tarski, possui um menor ponto fixo $\mathbf{lfp}(\Phi^{\phi}_{\overline{\mathbf{Q}},\bar{b}})$. A Lógica de Menor Ponto Fixo (Least Fixed Point Logic—LFP) é obtida adicionando à linguagem da lógica de primeira-ordem um operador **lfp** para a criação de um novo predicado interpretado como o menor ponto fixo de $\Phi^{\phi}_{\overline{\mathbf{Q}},\bar{b}}$ para cada fórmula $\phi(\overline{Q}, X, \overline{x}, \overline{y})$ positiva em X. Temos a seguinte definição.

Definição 2.6 (A Lógica LFP) *A linguagem de LFP é definida adicionando-se a seguinte regra ao cálculo de formação de formulas da lógica de primeira-ordem (veja Definição 1.13):*

$$\frac{\phi(\overline{Q}, X, \overline{x}, \overline{y})}{[\mathbf{lfp}_{X,\overline{x}}\phi(\overline{Q}, X, \overline{x}, \overline{y})](\overline{t})} ,$$

onde $\phi(\overline{Q}, X, \overline{x}, \overline{y})$ é positiva em X, o tamanho de $\overline{x}$ é igual à aridade de X, e $\overline{t}$ é uma tupla de termos cujo tamanho é igual à aridade de X.

Seja $\psi = [\mathbf{lfp}_{X,\overline{x}}\phi(\overline{Q}, X, \overline{x}, \overline{y})](\overline{t})$ e $\mathfrak{I}$ uma S-interpretação. Nós definimos a relação de satisfação $\models$ envolvendo $\mathfrak{I}$ e ψ da seguinte forma:

$$\mathfrak{I} \models [\mathbf{lfp}_{X,\overline{x}}\phi(\overline{Q}, X, \overline{x}, \overline{y})](\overline{t}) \textit{ sss } \mathfrak{I}(\overline{t}) \in \mathbf{lfp}(\Phi^{\phi}_{\mathfrak{I}(\overline{Q}),\mathfrak{I}(\overline{y})}).$$

Muito tem sido estudado a respeito da Teoria dos Modelos Finitos de LFP e outras lógicas de ponto fixo [Gur84, Lib04, EF95]. Entretanto a Teoria dos Modelos (Finitos e Infinitos) não tem sido tão investigada (veja [DG02, Kre02a, Kre02b] para alguns resultados sobre a Teoria dos Modelos Infinitos de algumas lógicas de ponto fixo).

Vejamos agora alguns exemplos de como utilizar LFP para definir classes de estruturas que não podem ser definidas na lógica de primeira-ordem.

Exemplo 2.1 *Seja $S^{ar} := \{0, \sigma, +, \times\}$ o alfabeto da linguagem da aritmética, composto pela constante zero, a função unária sucessor e as funções binárias soma e multiplicação. Seja ϕ^{ar} a conjunção das seguintes fórmulas:*

- $\neg\exists x(0 = \sigma x)$;
- $\forall x \forall y(\neg(x = y) \to \neg(\sigma x = \sigma y))$ *(injetividade de σ)*;
- $\forall x(x + 0 = x) \wedge \forall x \forall y(x + \sigma y = \sigma(x + y))$;
- $\forall x(x \times 0 = 0) \wedge \forall x \forall y(x \times \sigma y = x + x \times y)$.

Seja $\phi := \forall z[\mathbf{lfp}_{X,x}(x = 0 \vee \exists y(X(y) \wedge x = \sigma y))](z)$. $\phi^{ar} \wedge \phi$ define o modelo padrão da aritmética a menos de isomorfismo.

É fácil ver que o predicado definido por

$$[\mathbf{lfp}_{X,x}(x = 0 \vee \exists y(X(y) \wedge x = \sigma y))](z)$$

é a parte padrão de qualquer modelo de ϕ^{ar}, ou seja, o conjunto dos elementos representados pelos termos $0, \sigma 0, \sigma\sigma 0, \ldots$. Para tanto, seja $\mathfrak{A}$ um modelo de ϕ^{ar} e considere a seqüencia de conjuntos X^α, para cada ordinal α. Por definição $X^0 = \emptyset$. Seja

$$\psi(x) := (x = 0 \vee \exists y(X(y) \wedge x = \sigma y)).$$

Temos que $X^1 = \{a \in A | (\mathfrak{A}, X^0) \models \psi(x)[a]\} = \{0\}$. Analogamente, $X^2 = \{0, \sigma 0\}$, $X^3 = \{0, \sigma 0, \sigma\sigma 0\}$ e assim por diante. Temos então que

$$X^\omega = \{0, \sigma 0, \sigma\sigma 0, \sigma\sigma\sigma 0, \ldots\}.$$

É fácil ver que $X^{\omega+1} = X^\omega$, uma vez que, a cada estágio $X^{\alpha+1}$ da indução, adicionamos o sucessor de cada elemento do estágio anterior e o estágio X^ω é fechado para a função sucessor. Logo $X^\omega = X^\infty$. Mas X^∞ é exatamente a parte padrão de $\mathfrak{A}$. A fórmula ϕ afirma que todos os elementos do modelo são aqueles pertencentes à parte padrão. Logo, se $\mathfrak{A} \models \phi^{ar} \wedge \phi$, então $\mathfrak{A}$ é isomórfico ao modelo padrão da aritmética. Isso mostra que LFP é mais expressiva que lógica de primeira-ordem, uma vez que essa classe de modelos não pode ser definida em primeira-ordem.

Exemplo 2.2 *Seja $S = \{<\}$ onde $<$ é um símbolo relacional binário. Seja $OL(<)$ a conjunção das seguintes sentenças:*

- $\forall x \neg(x < x)$ *(irreflexividade)*;
- $\forall x \forall y \forall z((x < y \wedge y < z) \to x < z)$ *(transitividade)*;
- $\forall x \forall y(x < y \vee x = y \vee y < x)$ *(totalidade ou tricotomia)*;

$OL(<)$ diz que $<$ é uma ordem linear estrita. Considere ainda as seguintes fórmulas:

- $S(x, y) := x < y \wedge \forall z(x < z \rightarrow (z = y \vee y < z))$ *(y é o sucessor de x com respeito a $<$);*
- $SS(x, y) := \exists z(S(x, z) \wedge S(z, y))$*;*
- $Ma(x) := \forall y(y < x \vee y = x)$ *(x é o maior elemento de $<$);*
- $Me(x) := \forall y(x < y \vee x = y)$ *(x é o menor elemento de $<$).*

Definimos a sentença

$$\begin{aligned}\theta(<) \;\; := \;\; & OL(<) \wedge \exists x(Ma(x)) \wedge \forall x(\neg(Ma(x)) \rightarrow \exists y(S(x, y))) \wedge \\ & \exists x(Me(x)) \wedge \forall x(\neg(Me(x)) \rightarrow \exists y(S(y, x))).\end{aligned}$$

$\theta(<)$ diz que $<$ é uma ordem linear estrita com maior e menor elemento e na qual todo elemento, exceto o menor, tem um predecessor e todo elemento, exceto o maior, possui um sucessor (observe que uma ordem com essas características possui um segmento inicial que é uma boa-ordem[1]*).*

Seja

$$\zeta(P, x) := \exists z(Me(z) \wedge S(z, x)) \vee \exists z(P(z) \wedge SS(z, x)).$$

A fórmula

$$\phi(w) := \theta(<) \wedge [\mathbf{lfp}_{P,x}\zeta(P, x)](w)$$

afirma que w é um elemento que ocupa a α-ésima posição no segmento inicial bem ordenado de $<$, para algum ordinal α finito e par. Portanto a sentença $\psi(<) := \exists y(Ma(y) \wedge \phi(y))$ diz que $<$ é uma ordem linear total em um universo com número par de elementos.

Para verificar essa afirmação, seja $\mathfrak{A}$ uma S-estrutura tal que $\mathfrak{A} \models \psi(<)$. Seja $a \in A$ tal que $\mathfrak{A} \models (Ma(x) \wedge \phi(x))[a]$. Então $\mathfrak{A} \models Ma(x)[a]$, $\mathfrak{A} \models \theta$ e $\mathfrak{A} \models [\mathbf{lfp}_{P,x}\zeta(P, x)](y)[a]$. Portanto a é o maior elemento de $<^A$ e $a \in \mathbf{lfp}(\Phi^{\zeta}_{\emptyset,\emptyset})$. Considere os estágios da indução em $\Phi^{\zeta}_{\emptyset,\emptyset}$. Por definição $X^0 = \emptyset$. Temos que

$$X^1 = \{a' \in A | (\mathfrak{A}, X^0) \models \zeta(P, x)[a']\}.$$

[1]Dada uma relação binária $<$, definimos o *campo* de $<$ como sendo o conjunto *campo*$-<= dom(<) \cup img(<)$, onde $dom(<)$ e $img(<)$ são, respectivamente, o domínio e a imagem de $<$. Uma relação de ordem é uma *boa-ordem* se todo subconjunto de *campo*$-<$ possuir um menor elemento com respeito a $<$.

Observa-se então que o único elemento em X^1 é o sucessor do menor elemento de A com relação a $<$, portanto, o segundo elemento nessa ordem. Analogamente, temos

$$X^2 = \{a' \in A | (\mathfrak{A}, X^1) \models \zeta(P, x)[a']\}$$

e, portanto, X^2 contém apenas o sucessor do menor elemento de A com relação $<$ e o sucessor do sucessor deste nessa mesma ordem (ou seja, o segundo e o quarto elemento). Continuando por indução nos ordinais finitos, obtemos que X^ω possui todos os elementos de posição par na ordem $<$. É fácil ver que, por argumento análogo ao utilizado no Exemplo 2.1, que $\mathbf{cl}(\Phi^\zeta_{\emptyset,\emptyset}) \leq \omega$. Decorre que $a \in \mathbf{lfp}(\Phi^\zeta_{\emptyset,\emptyset})$ se, e somente se, a ocupa a α-ésima posição na ordem $<$, para algum ordinal α finito e par. Mas a também é o maior elemento dessa ordem. Portanto, essa ordem deve ser finita e ter comprimento par.

Revisitaremos o Exemplo 2.2 na Seção 4.5. Na próxima seção falaremos sobre uma extensão da Lógica de Menor Ponto Fixo: a Lógica de Menor Ponto Fixo Simultâneo.

2.2.1 Lógica de Menor Ponto Fixo Simultâneo

Trataremos agora de uma extensão da Lógica de Menor Ponto Fixo: a Lógica de Menor Ponto Fixo Simultâneo (Si-LFP). Enquanto que em LFP nós calculamos o menor ponto fixo do operador definido por uma fórmula positiva, em Si-LFP nós calculamos, simultaneamente, os menores pontos fixos de vários operadores que dependem uns dos outros em sua definição. Primeiramente, faremos algumas observações.

Seja $\mathfrak{A}$ uma estrutura. Considere o par

$$\mathfrak{R} := (R = \wp(A^{r_1}) \times \ldots \times \wp(A^{r_n}), \leq),$$

onde

$$(D_1, \ldots, D_n) \leq (D'_1, \ldots, D'_n) \text{ sss } D_i \subseteq D'_i, \text{ para } 1 \leq i \leq n,$$

para $(D_1, \ldots, D_n)$ e $(D'_1, \ldots, D'_n)$ em $\wp(A^{r_1}) \times \ldots \times \wp(A^{r_n})$. $\mathfrak{R}$ é um reticulado completo (veja Apêndice A, Exemplo A.2). Sejam $R_1, \ldots, R_n$ símbolos predicativos e $\phi_1(\overline{x}_1, \overline{y}), \ldots, \phi_n(\overline{x}_n, \overline{y})$ S-fórmulas positivas nas variáveis relacionais $R_1, \ldots, R_n$, de aridade $r_1, \ldots, r_n$, respectivamente. Suponha que as variáveis livres de ϕ_i estão entre $\overline{x}_i, \overline{y}$ e $R_1, \ldots, R_n, \overline{Q}$. Seja $\overline{y} = y_1, \ldots, y_m$. Suponha ainda que o comprimento de $\overline{x}_i$ é r_i. Considere os operadores $F^{\phi_i}_{\overline{b},\overline{\mathbf{Q}}} : \wp(A^{r_1}) \times \ldots \times \wp(A^{r_n}) \to \wp(A^{r_i})$ definidos como

$$F^{\phi_i}_{\overline{b},\overline{\mathbf{Q}}}(X_1, \ldots, X_n) := \{\overline{a} \in A^{r_i} | (\mathfrak{A}, \overline{\mathbf{Q}}, X_1, \ldots, X_n) \models \phi(\overline{x}_i, \overline{y}_i)[\overline{a}, \overline{b}]\},$$

onde o comprimento de $\overline{b}$ é o mesmo de $\overline{y}$, isto é, m.

Seja $\Phi^{\phi_1,\ldots,\phi_n}_{\overline{b},\overline{\mathbf{Q}}} : R \to R$ uma função definida sobre o reticulado $\mathfrak{R}$ como

$$\Phi^{\phi_1,\ldots,\phi_n}_{\overline{b},\overline{\mathbf{Q}}}(D_1,\ldots,D_n) := (F^{\phi_1}_{\overline{b},\overline{\mathbf{Q}}}(D_1,\ldots,D_n),\ldots,F^{\phi_n}_{\overline{b},\overline{\mathbf{Q}}}(D_1,\ldots,D_n)).$$

$\Phi^{\phi_1,\ldots,\phi_n}_{\overline{b},\overline{\mathbf{Q}}}$ é monótona (veja Apêndice A, Exemplo A.2). Logo, pelo Teorema de Knaster-Tarski, $\Phi^{\phi_1,\ldots,\phi_n}_{\overline{b},\overline{\mathbf{Q}}}$ possui um menor ponto fixo $\mathbf{lfp}(\Phi^{\phi_1,\ldots,\phi_n}_{\overline{b},\overline{\mathbf{Q}}})$.

Definimos a sequência X^α de elementos de $\wp(A^{r_1}) \times \ldots \times \wp(A^{r_n})$, para cada ordinal α:

$$\begin{aligned} X^0 &= (\emptyset,\ldots,\emptyset), \\ X^{\alpha+1} &= \Phi^{\phi_1,\ldots,\phi_n}_{\overline{b},\overline{\mathbf{Q}}}(X^\alpha), \\ X^\lambda &= (\bigcup_{\mu<\lambda} X^\mu(1),\ldots,\bigcup_{\mu<\lambda} X^\mu(n)), \end{aligned}$$

onde $X^\alpha(i)$ é o i-ésimo elemento da tupla X^α. Seja $X^\infty = X^\alpha$, onde α é o menor ordinal tal que $X^\alpha = X^{\alpha+1}$.

Novamente pelo Teorema de Knaster-Tarski, temos $X^\infty = \mathbf{lfp}(\Phi^{\phi_1,\ldots,\phi_n}_{\overline{b},\overline{\mathbf{Q}}})$.

A Lógica de Menor Ponto Fixo Simultâneo utiliza esse ponto fixo para construir novos predicados.

Definição 2.7 (A Lógica Si-LFP) *A linguagem de Si-LFP é obtida a partir da de LFP adicionando a seguinte regra para formação de fórmulas. Sejam $\phi_1(\overline{x}_1,\overline{y}_1),\ldots,\phi_n(\overline{x}_n,\overline{y}_n)$ S-fórmulas positivas nas variáveis relacionais $R_1,\ldots,R_n$, de aridade $r_1,\ldots,r_n$, respectivamente. Então a fórmula*

$$[\mathbf{slfp}_{\overline{x}_1,R_1,\ldots,\overline{x}_n,R_n}\phi_1,\ldots,\phi_n](\overline{t})$$

é uma fórmula de S-LFP. A interpretação dessa fórmula é dada da seguinte maneira. Seja $\mathfrak{I} = (\mathfrak{A},\beta)$ uma S-interpretação. Suponha que as variáveis livres de ϕ_i estão entre $\overline{x}_i,\overline{y}$ e $R_1,\ldots,R_n,\overline{Q}$.

Dada uma fórmula $\psi = [\mathbf{slfp}_{\overline{x}_1,R_1,\ldots,\overline{x}_n,R_n}\phi_1,\ldots,\phi_n](\overline{t})$ e uma interpretação $\mathfrak{I}$, definimos a relação de satisfação $\models$ entre ψ e $\mathfrak{I}$ como

$$\mathfrak{I} \models [\mathbf{slfp}_{\overline{x}_1,R_1,\ldots,\overline{x}_n,R_n}\phi_1,\ldots,\phi_n](\overline{t}) \text{ sss } \mathfrak{I}(\overline{t}) \in \mathbf{lfp}(\Phi^{\phi_1,\ldots,\phi_n}_{\mathfrak{I}(\overline{y}),\mathfrak{I}(\overline{Q})})(1).$$

É sabido que a adição de pontos fixos simultâneos não adiciona poder expressivo a LFP. Veja o lema seguinte (para uma prova do lema abaixo veja [AN01], veja também [Kre02b]).

Lema 2.3 *Toda fórmula*

$$\psi = [\mathbf{slfp}_{\overline{x}_1,R_1,\ldots,\overline{x}_n,R_n}\phi_1,\ldots,\phi_n](\overline{t})$$

de Si-LFP é equivalente a

$$\psi' = [\mathbf{slfp}_{\overline{x}_1,R_1,\ldots,\overline{x}_n,R_n}\phi'_1,\ldots,\phi'_{n-1}](\overline{t}),$$

onde ϕ'_i *é obtido a partir* ϕ_i *substituindo todo átomo da forma* $R_n\overline{t}'$ *por* $[\mathbf{lfp}_{R_n,\overline{x}_n}\phi_n](\overline{t}')$.

Iterando o Lema 2.3 obtemos:

Teorema 2.4 *Toda fórmula de Si-LFP possui uma equivalente em LFP.*

Na próxima seção, falaremos sobre definibilidade em LFP. Mostraremos que o Teorema de Beth não vale para LFP e, utilizando um teorema de Hodkinson [Hod93], veremos que esse teorema continua não valendo mesmo se nos restringirmos a definições implícitas feitas por teorias finitas. Comentaremos ainda o fato de que uma restrição do Teorema de Beth vale em LFP para certas definições recursivas.

2.3 Definibilidade

Conforme comenta Beth em [Bet53], em [Pad00], Padoa nos chama a atenção para o fato de que, a fim de mostrarmos a independência de certa noção primitiva com relação a outra (por noção primitiva entenda um símbolo da linguagem) em uma determinada teoria Γ, podemos adotar o seguinte procedimento (chamado de Método de Padoa). Para simplificar suponhamos que nossas noções primitivas sejam os símbolos predicativos $P, P_1, \ldots, P_n$. A fim de mostrar que P é independente de $P_1, \ldots, P_n$, devemos exibir dois modelos $\mathfrak{A}$ e $\mathfrak{A}'$ de Γ tais que $\mathfrak{A}$ e $\mathfrak{A}'$ coincidem na interpretação dos símbolos $P_1, \ldots, P_n$ (e em seus domínios, isto é, $A = A'$) e diferem quanto à interpretação de P (ou seja, $P^A \neq P^{A'}$). Daí decorre diretamente que, se P pudesse ser definido a partir de Γ *em termos de* $P_1, \ldots, P_n$ (o que corresponde a P possuir uma definição explícita, veja definição abaixo), usando essa definição chegaríamos à conclusão de que $\mathfrak{A}$ e $\mathfrak{A}'$ também concordam na interpretação de P, o que, por hipótese, é falso. Vamos colocar tudo isso em termos mais precisos. Para tanto introduzimos as definições seguintes.

Definição 2.8 (Definição Implícita em Classe Estruturas) *Seja* $\mathcal{L} = (L, \models_{\mathcal{L}})$ *um sistema lógico. Seja* S *um alfabeto e* P *um símbolo qualquer* *(função, constante ou predicado). Seja* $\Gamma(P) \in L(S \cup P)$ *um conjunto sentenças de* $\mathcal{L}$ *e* $\mathbb{C}$ *uma classe de* S*-estruturas. Dizemos que* $\Gamma(P)$ define implicitamente P em $\mathbb{C}$ *se toda estrutura* $\mathfrak{A}$ *de* $\mathbb{C}$ *pode ser expandida para no máximo uma estrutura* $(\mathfrak{A}, P^A)$ *que seja modelo de* $\Gamma(P)$*. Analogamente,* $\Gamma(P)$ *define implicitamente e fortemente* P *em* $\mathbb{C}$ *se toda estrutura* $\mathfrak{A}$ *de* $\mathbb{C}$ *pode ser expandida para exatamente um modelo de* $\Gamma(P)$*. Quando* $\mathbb{C}$ *é o conjunto unitário* $\{\mathfrak{A}\}$ *escrevemos "*$\Gamma(P)$ *define implicitamente* P *em* $\mathfrak{A}$*" ao invés de "*$\Gamma(P)$ *define implicitamente* P *em* $\mathbb{C}$*." Analogamente para "*$\Gamma(P)$ *define implicitamente e fortemente* P *em* $\mathfrak{A}$*."*

Definição 2.9 (Definição Implícita) *Seja* $\mathcal{L}$ *uma lógica (ou sistema lógico). Seja* S *um alfabeto,* P *um símbolo relacional e* Γ *uma* $S \cup \{P\}$*-teoria de* $\mathcal{L}$*. Dizemos que* Γ define implicitamente P *em* $\mathcal{L}$ *se, para quaisquer dois modelos* $(\mathfrak{A}, \mathbf{P})$ *e* $(\mathfrak{A}, \mathbf{P}')$ *de* Γ*,* $\mathbf{P} = \mathbf{P}'$*.*

Definição 2.10 (Definição Explícita) *Seja* $\mathcal{L}$ *uma lógica (ou sistema lógico). Seja* S *um alfabeto,* P *um símbolo relacional* n*-ário e* Γ *uma* $S \cup \{P\}$*-teoria de* $\mathcal{L}$*. Uma* S*-fórmula* $\phi(\overline{x})$*, onde o símbolo* P *não ocorre e* $\overline{x} = x_1, \ldots, x_n$*, tal que*

$$\Gamma \models_{\mathcal{L}} \forall \overline{x}(P(\overline{x}) \leftrightarrow \phi(\overline{x})),$$

é uma definição explícita de P *com relação a* Γ *e* $\mathcal{L}$[2]*.*

Facilmente se observa que, em lógica de primeira-ordem, se existe uma definição explícita de um símbolo P em relação a uma teoria Γ, então Γ define P implicitamente. O Método de Padoa consiste então em aplicar a contra-positiva dessa implicação e concluir que P não possui uma definição explícita em termos dos demais símbolos em S mostrando que Γ não define P implicitamente. Introduzimos então a seguinte definição.

Definição 2.11 (Propriedade de Padoa) *Uma lógica* $\mathcal{L}$ *possui a* Propriedade de Padoa *se, para toda* $S \cup P$*-teoria* Γ *de* $\mathcal{L}$*, se existe uma definição explícita para* P *com relação a* Γ *e* $\mathcal{L}$[3]*, então* Γ *define implicitamente* P *em* $\mathcal{L}$*.*

Em [Bet53], Beth aborda a questão contrária. Isto é, a de verificar a existência ou não de uma definição explícita de um símbolo quando uma teoria o define implicitamente. Para a lógica de primeira ordem, Beth dá uma resposta positiva a esse problema. Isto é, sempre que uma teoria Γ em lógica

[2]Observe que estamos assumindo que o sistema lógico $\mathcal{L}$ possui os operadores booleanos e o quantificador universal.

[3]Cf. 2.

de primeira-ordem define implicitamente um símbolo P existe uma definição explícita de P com relação a Γ. Definimos então a seguinte propriedade sobre sistemas lógicos.

Definição 2.12 (Propriedade e Teorema de Beth) *Uma lógica $\mathcal{L}$ possui a* Propriedade de Beth *se, para toda $S \cup P$-teoria Γ de $\mathcal{L}$, se Γ define P implicitamente então existe uma definição explícita de P com relação a Γ e $\mathcal{L}$*[4]*. O* Teorema de Beth *para uma lógica $\mathcal{L}$ é o teorema que afirma que $\mathcal{L}$ possui a Propriedade de Beth.*

Como dissemos, Beth mostrou que a Propriedade de Beth vale para a lógica de primeira-ordem. Essa propriedade tem sido mostrada válida para outras lógicas, como a lógica infinitária $L_{\omega_1\omega}$ e várias lógicas modais. Em [Hoo01, Capítulo 2], Hoogland traz uma extensa lista de várias lógicas que possuem a Propriedade de Beth. Na próxima seção, veremos que a Propriedade de Beth não vale para LFP.

2.3.1 A Falha do Teorema de Beth

Nós mostraremos a seguir que o Teorema de Beth falha para LFP com prova de nossa autoria[5]. Para tanto, provaremos que existe uma $S \cup \{P\}$-teoria de LFP que define P implicitamente mas não define P explicitamente. Sejam S^{ar}, ϕ^{ar} e ϕ como no Exemplo 2.1. Seja P um símbolo predicativo unário. Definimos o conjunto dos termos canônicos de S^{ar} como sendo o conjunto

$$T^C := \{\mathbf{n} = \underbrace{\sigma \ldots \sigma}_{n} 0 | n \in N\}.$$

Para cada subconjunto T de T^C, definimos a teoria

$$\Gamma(T) = \{\phi^{ar} \wedge \phi\} \cup \{P(t) | t \in T\} \cup \{\neg P(t) | t \notin T\}.$$

Obviamente $\Gamma(T)$ é satisfatível para qualquer conjunto de termos canônicos T. Vejamos o seguinte lema.

Lema 2.5 *$\Gamma(T)$ define P implicitamente.*

[4]Cf. 2.

[5]Durante todo esse trabalho, citamos diretamente a fonte de todos os resultados que não são de nossa autoria. Quando se tratar de resultados de autoria de desconhecida, também resaltaremos esse fato ou indicaremos onde encontrar uma prova para o mesmo. Todos os demais resultados são de nossa autoria. Mesmo se não estiver explícitamente discriminado com expressões do tipo "esse resultado é de nossa autoria." Normalmente, escrevemos "mostraremos," "provaremos," etc., e não sendo expressamente discriminada outra fonte, o leitor poderá concluir que o resultado é nosso.

Prova. Seja $\mathfrak{A}$ um modelo de $\phi^{ar} \wedge \phi$. Conforme observado anteriormente, $\mathfrak{A}$ é isomórfico ao modelo padrão da aritmética. Segue que $A = \{t^{\mathfrak{A}} | t \in T^C\}$. Seja $T \subseteq T^C$ um conjunto de termos canônicos. Seja $(\mathfrak{A}, \mathbf{P})$ modelo de $\Gamma(T)$. Se t é um termo canônico, então $t^{\mathfrak{A}} \in \mathbf{P}$ se, e somente se, $t \in T$. Sejam $\mathbf{P}$ e $\mathbf{P}'$ tais que $(\mathfrak{A}, \mathbf{P}) \models \Gamma(T)$ e $(\mathfrak{A}, \mathbf{P}') \models \Gamma(T)$. Um termo canônico $t \in T^C$ é tal que $t^{\mathfrak{A}} \in \mathbf{P}$ se, e somente se, $t \in T$ se, e somente se, $t^{\mathfrak{A}} \in \mathbf{P}'$. Logo, $\mathbf{P} = \mathbf{P}'$. ∎

Observe também que se T e T' são conjuntos de termos canônicos tais que $T \neq T'$ e $(\mathfrak{A}, \mathbf{P}) \models \Gamma(T)$ e $(\mathfrak{A}, \mathbf{P}') \models \Gamma(T')$, então $\mathbf{P} \neq \mathbf{P}'$. Portanto, se $\phi(x)$ e $\phi'(x)$ são definições explícitas de P com relação a $\Gamma(T)$ e $\Gamma(T')$, respectivamente, então $\phi(x) \neq \phi'(x)$, pois caso contrário essas duas fórmulas definiriam o mesmo predicado em A. Observe ainda que a quantidade de conjuntos de termos canônicos diferentes é $\aleph_1$. Entretanto, o alfabeto $S^{ar} \cup \{P\}$ é finito, e cada fórmula de LFP é finita. Segue-se então que a cardinalidade do conjunto de definições explícitas é $\aleph_0$. Com essas observações nós provamos o seguinte teorema.

Teorema 2.6 (Falha do Teorema de Beth) *O Teorema de Beth não vale para LFP.*

Prova. Suponha the LFP possui a propriedade de Beth. Então para todos conjuntos de termos canônicos T, $\Gamma(T) \models \forall x(P(x) \leftrightarrow \phi(x))$ para alguma definição explícita $\phi(x)$ de P. Como a cardinalidade do conjunto das partes do conjunto de termos canônicos é $\aleph_1$ e a cardinalidade do conjunto de fórmulas é $\aleph_0$, existem dois conjuntos de termos canônicos T e T' e uma fórmula $\phi(x)$ tais que $T \neq T'$ e $\Gamma(T) \models \forall x(P(x) \leftrightarrow \phi(x))$ e $\Gamma(T') \models \forall x(P(x) \leftrightarrow \phi(x))$. Mas, como vimos, isso implicaria em $T = T'$, chegando a uma contradição. Logo LFP não possui a Propriedade de Beth. ∎

No teorema anterior, as teorias $\Gamma(T)$ são infinitas. Poderíamos nos perguntar se o Teorema de Beth valeria caso nos restringíssemos a teorias finitas. Entretanto, isso não é verdade e podemos utilizar um resultado de Hodkinson em [Hod93] para mostrar isso.

Em [GS96], Gurevich e Shelah introduzem certas estruturas em um determinado vocabulário S^m chamadas *multípedes*. Essas estruturas possuem as seguintes características (entre outras—para a definição dos multípedes e suas propriedades veja [Hod93, GS96, DHK95]):

1. existe uma fórmula μ em lógica de primeira-ordem cujos modelos finitos são exatamente os multípedes de cardinalidade ímpar;

2. todo modelo de μ possui um conjunto de elementos (que chamaremos de *espinha*) linearmente ordenado por uma relação binária $<\in S^m$;

3. todo modelo de μ que possui espinha finita é finito.

Gurevich e Shelah mostraram em [GS96] que não existe uma fórmula $\phi(x, y)$ na lógica infinitária $L^{\omega}_{\omega_1\omega}$ que defina explicitamente uma ordem linear em cada multípede ímpar. Assim temos:

Teorema 2.7 (Gurevich e Shelah) *Não existe fórmula $\phi(x, y)$ de $L^{\omega}_{\omega_1\omega}$ que define explicitamente uma ordem linear na classe dos multípedes finitos e ímpares.*

Entretanto, em [DHK95], Dawar et al. mostraram que existe uma sentença $\lambda(\prec)$ no alfabeto $S^m \cup \{\prec\}$ em lógica de primeira-ordem que define implicitamente uma ordem linear $\prec$ na classe dos multípedes ímpares.

Teorema 2.8 (Dawar et al.) *Existe uma sentença $\lambda(\prec)$ em lógica de primeira-ordem que define implicitamente uma ordem linear $\prec$ na classe dos multípedes finitos ímpares.*

Hodkinson então exibe uma fórmula $\delta(<)$ na lógica $L^{\omega}_{\omega_1\omega}$ que afirma que a ordem $<$ possui tamanho finito. Isso garante que a espinha de cada multípede modelo de μ é finita e, portanto, o próprio multípede modelo de μ é finito. Logo, a fórmula $\mu^+ = \mu \wedge \delta(<)$ possui como modelos exatamente os multípedes ímpares (isto é, define a classe dos multípedes ímpares). Segue-se então que $\mu^+ \wedge \lambda(\prec)$ define implicitamente $\prec$ em $L^{\omega}_{\omega_1\omega}$, mas, pelo Teorema 2.7, $L^{\omega}_{\omega_1\omega}$ não possui definição explícita para $\prec$ com relação a $\mu^+ \wedge \lambda(\prec)$. Daí, Hodkinson conclui que a lógica infinitária $L^{\omega}_{\omega_1\omega}$ não possui a Propriedade de Beth.

Aqui, utilizaremos a estratégia de Hodkinson para provar o mesmo teorema para LFP. Para provarmos que LFP também não possui a Propriedade de Beth, mesmo nos restringindo a teorias finitas, mostraremos que $\delta(<)$ possui equivalente em LFP. Juntando isso ao fato de que LFP, considerando-se apenas modelos finitos, está contida em $L^{\omega}_{\omega_1\omega}$ (no sentido de que para toda fórmula em LFP existe uma em $L^{\omega}_{\omega_1\omega}$ com os mesmos modelos finitos—para demonstrações desse resultado veja [Hod93, EF95]), concluiremos que LFP não possui a Propriedade de Beth.

Corolário 2.9 *A prova de Hodkinson de que o Teorema de Beth falha para $L^{\omega}_{\omega_1\omega}$ pode ser adaptada para mostrar que o mesmo teorema falha para LFP. Mais ainda: existe uma teoria finita de LFP que define um predicado implicitamente mas não existe definição explícita para esse predicado em LFP com relação àquela teoria finita.*

Prova. Considere a seguinte fórmula no alfabeto $S^m \cup \{P\}$:

$$\phi(P, x) := Me(x) \vee \exists y(P(y) \wedge S(y, x)),$$

onde as fórmulas $Me(x)$ e $S(y, x)$ são as mesmas do Exemplo 2.2. Seguindo um argumento semelhante ao do Exemplo 2.2, é possível mostrar que a fórmula $\psi'(y) := [\mathbf{lfp}_{P,x}\phi(P, x)](y)$ define o predicado contendo os elementos que ocupam posição α na ordem $<$, para algum ordinal $\alpha < \omega$, quando $<$ é interpretada como uma ordem linear (não necessariamente sobre todo o domínio) com menor elemento e tal que todo elemento, exceto o menor, tem predecessor e todo elemento, exceto o maior, tem sucessor. Seja

$$\phi''(<) := \forall x \forall y((\textit{campo-}{<}(x) \wedge \textit{campo-}{<}(y)) \rightarrow (x < y \vee y < x)),$$

onde $\textit{campo-}{<}(x) := \exists y(x < y \vee y < x)$. ϕ'' diz que $<$ é total em *campo-*$<$ (o conjunto dos elementos que participam em alguma tupla da relação $<$). Seja OL' igual à sentença OL do Exemplo 2.2 substituindo a subfórmula correspondente ao axioma de totalidade por $\phi''(<)$. Seja θ' igual a θ, do mesmo exemplo, substituindo a subfórmula $OL(<)$ por $OL'(<)$. De forma análoga ao Exemplo 2.2, a sentença

$$\delta'(<) := OL'(<) \wedge \exists w(Ma(w) \wedge [\mathbf{lfp}_{P,x}\phi(P, x)](w))$$

define exatamente as estruturas em que $<$ é uma ordem linear em *campo-*$<$ e *campo-*$<$ é finito (pois o último elemento da ordem pertence ao predicado $[\mathbf{lfp}_{P,x}\phi(P, x)]$ e, portanto, ocupa posição α, para algum ordinal $\alpha < \omega$, na ordem $<$). Considerando agora a sentença $\mu'^{+} := \mu \wedge \delta'(<)$, concluímos que os modelos de μ'^{+} são exatamente os multípedes ímpares cuja espinha é finita (pois *campo-*$<$ é exatamente a espinha do multípede), portanto, $\mu'^{+} \equiv \mu^{+}$. Logo μ'^{+} define a classe dos multípedes ímpares finitos e, pelo Teorema 2.8 [DHK95], $\mu'^{+} \wedge \lambda(\prec)$ define $\prec$ implicitamente (lembre-se que $<$ é ordena a espinha de um multípede, enquanto que $\prec$ ordena todo o domínio do multípede). Como $\mu'^{+} \wedge \lambda(\prec)$ é uma fórmula de LFP, basta mostrarmos que não existe definição explícita em LFP para $<$ com relação a $\mu'^{+} \wedge \lambda(\prec)$. Mas isso decorre imediatamente do fato de que, para toda fórmula de LFP que somente possui modelos finitos, existe uma equivalente em $L^{\omega}_{\omega_1\omega}$, junto com o Teorema 2.7 de Gurevich e Shelah (se existisse uma definição explícita $\nu(x, y)$ para $\prec$, nos bastaria tomar a equivalente a $\nu(x, y)$, digamos, $\nu'(x, y)$, em $L^{\omega}_{\omega_1\omega}$ para contradizer o Teorema 2.7). ∎

Na verdade, Hodkinson utiliza a sentença $\mu^{+} \wedge \lambda(\prec)$ para mostrar que a Propriedade Fraca de Beth também falha para $L^{\omega}_{\omega_1\omega}$. Nós podemos fazer o mesmo para LFP utilizando $\mu'^{+} \wedge \lambda(\prec)$. Vamos primeiramente explicar do que se trata a Propriedade Fraca de Beth. Seja Γ uma $S \cup \{P\}$-teoria de uma lógica $\mathcal{L}$. Seja $Th_S(\Gamma) := \{\phi \in \mathcal{L} | \Gamma \models_{\mathcal{L}} \phi\}$ o conjunto das consequências de Γ no alfabeto S. O conceito de definição implícita exige que, se $(\mathfrak{A}, \mathbf{P})$ é modelo de Γ, então $\mathbf{P}$ é único, ou, em outras palavras, que todo modelo $\mathfrak{A}$ de $Th_S(\Gamma)$ possa ser expandido para *no máximo* uma estrutura $(\mathfrak{A}, \mathbf{P})$ que seja modelo de Γ. Ou seja, não se exige que todo modelo de $Th_S(\Gamma)$ necessariamente se expanda para um modelo de Γ. Uma *definição implícita forte* é uma definição implícita que atende a essa exigência adicional. Nós a definimos abaixo.

Definição 2.13 (Definição Implícita Forte) *Seja $\mathcal{L}$ uma lógica (ou um sistema lógico). Seja S um alfabeto, P um símbolo relacional e Γ uma $S \cup \{P\}$-teoria de $\mathcal{L}$. Dizemos que Γ é uma* definição implícita forte de P *em* $\mathcal{L}$ *se, todo modelo $\mathfrak{A}$ de $Th_S(\Gamma)$ expande para exatamente um modelo $(\mathfrak{A}, \mathbf{P})$ de Γ.*

A Propriedade Fraca de Beth é definida assim:

Definição 2.14 *Uma lógica $\mathcal{L}$ possui a* Propriedade Fraca de Beth *se, para toda $S \cup P$-teoria Γ de $\mathcal{L}$, se Γ é uma definição implícita forte de P então existe uma definição explícita de P com relação a Γ e $\mathcal{L}$. O* Teorema Fraco de Beth *para uma lógica $\mathcal{L}$ é o teorema que afirma que $\mathcal{L}$ possui a Propriedade Fraca de Beth.*

Hodkinson mostra que $\mu^{+} \wedge \lambda(\prec)$ é uma definição forte [Hod93]. Analogamente, $\mu'^{+} \wedge \lambda(\prec)$ é também uma definição forte de $\prec$, pois todo multípede finito ímpar $\mathfrak{M}$ expande para somente uma estrutura $(\mathfrak{M}, \prec^M)$ modelo de $\mu'^{+} \wedge \lambda(\prec)$ (o que garante que $\mathfrak{M} \models Th_{S^m}(\mu'^{+} \wedge \lambda(\prec))$ para todo multípede finito ímpar $\mathfrak{M}$) e os únicos modelos de μ'^{+} são os multípedes finitos ímpares (o que garante que os únicos modelos de $Th_{S^m}(\mu'^{+} \wedge \lambda(\prec))$ são os multípedes finitos ímpares). Ou seja, todo modelo de $Th_{S^m}(\mu'^{+} \wedge \lambda(\prec))$ expande para exatamente um modelo de $\mu'^{+} \wedge \lambda(\prec)$. Logo:

Corolário 2.10 *A prova de Hodkinson de que o Teorema Fraco de Beth falha para $L^{\omega}_{\omega_1\omega}$ pode ser adaptada para mostrar que o mesmo teorema também falha para LFP.*

Nós vimos resultados negativos acerca da validade do Teorema de Beth e Teorema Fraco de Beth para LFP. Na próxima seção veremos exemplos em que é possível obter definição explícita a partir de certos tipos de definição implícita (que chamaremos de *definições recursivas*).

2.3.2 Estudo de Caso: Estruturas Indutivas e Definibilidade de Funções Recursivas

Uma forma bastante comum de definir conjuntos indutivos é através de funções em um determinado domínio [End72]. O esquema é o seguinte: dados um conjunto U, um subconjunto B de U e uma família $\mathbb{F}$ de operações em U, dizemos que um subconjunto $C \subseteq U$ é *indutivo* se $B \subseteq C$ e C é fechado para as operações em $\mathbb{F}$ (isto é, se $f : U^n \to U \in \mathbb{F}$, então se $u_1, \ldots, u_n \in C$ então $f(u_1, \ldots, u_n) \in C$). O conjunto *indutivamente gerado* por U, B e $\mathbb{F}$ é definido como a interseção de todos os conjuntos indutivos. Essa definição poder ser posta em termos da definição de conjunto indutivamente definido por um conjunto de regras (Definição 2.2) da Seção 2.1, considerando-se o conjunto de regras

$$\Upsilon := \{(\{\overline{u}\}, f(\overline{u}))|\overline{u} \in U^n, f \in \mathbb{F} \text{ de aridade } n,\ n \in N\} \cup \{(\emptyset, b)|b \in B\}.$$

Assim, o conjunto indutivamente gerado por U, B e $\mathbb{F}$ é o mesmo conjunto indutivamente definido por Υ. Vejamos o seguinte exemplo:

Exemplo 2.3 *Considere a teoria dos conjuntos ZF. Um dos axiomas de ZF é o axioma do infinito, que atesta a existência de um conjunto infinito. Mais precisamente, o axioma do infinito atesta a existência de um conjunto U que contêm $\emptyset$ e, para todo elemento $x \in U$, $S(x) = x \cup \{x\} \in U$. Assim, podemos definir o conjunto N dos números naturais como sendo o conjunto indutivamente gerado a partir de U e $\{S\}$. Por conveniência, chamaremos $\emptyset$ de 0 quando estivermos falando dos números naturais.*

Considere agora a função + definida sobre o conjunto dos números naturais. Classicamente, se costuma definir a soma através do seguinte sistema de equações:

$$x + 0 = x, \tag{2.1}$$
$$(x + S(y)) = S(x + y). \tag{2.2}$$

Isto é, definimos a soma como sendo a função que satisfaz esse sistema de equações. Observe que o símbolo que representa a função que se deseja definir ocorre do lado esquerdo da segunda equação. Isto é, fazemos uso da própria função para defini-la. Tais tipos de definições são chamadas de definições recursivas.

Uma questão que imediatamente ocorre é a respeito da existência e unicidade da função "definida" recursivamente (isto é, se o sistema de equações recursivas é de fato uma definição). O Teorema da Recursão irá garantir isso, sob certas condições. Considere a seguinte classe conjuntos indutivos.

Definição 2.15 *Seja U um conjunto, $B \subseteq U$ e $\mathbb{F}$ uma família de funções sobre U. Seja A o conjunto indutivamente gerado a partir de U, B e $\mathbb{F}$. Dizemos que A é* livremente gerado *se:*

- $B \cap \mathbf{img}(f) = \emptyset$, *para toda* $f \in \mathbb{F}$,
- f *é injetiva, para toda* $f \in \mathbb{F}$,
- $\mathbf{img}(f) \cap \mathbf{img}(f') = \emptyset$, *para todas* $f, f' \in \mathbb{F}$.

O Teorema da Recursão irá garantir a existência e unicidade de funções definidas recursivamente da seguinte forma.

Teorema 2.11 (Teorema da Recursão) *Seja U um conjunto, $B \subseteq U$ e $\mathbb{F}$ uma família de funções sobre U. Suponha que U, B e $\mathbb{F}$ satisfazem as condições da Definição 2.15. Seja A o conjunto (livremente) gerado a partir de U, B e $\mathbb{F}$. Seja V um conjunto. Seja $\mathbb{G} = \{G_f | f \in \mathbb{F}\} \cup \{G_B\}$ um conjunto de funções onde $G_f : A^{n+n_f} \times V^{n_f} \to V$ e $G_B : A^n \times B \to V$, e n_f é a aridade de f. Então existe uma, e apenas uma, função $h : A^{n+1} \to V$ que satisfaz as seguintes condições:*

- *para cada* $\overline{a} = a_1, \ldots, a_n \in A^n$ *e* $b \in B$,
$$h(\overline{a}, b) = G_B(\overline{a}, b).$$
- *para cada* $\overline{a} = a_1, \ldots, a_n \in A^n$ *e* $\overline{a'} = a'_1, \ldots, a'_{n_f} \in A^{n_f}$ *e* $f \in \mathbb{F}$,
$$h(\overline{a}, f(\overline{a'})) = G_f(\overline{a}, \overline{a'}, h(\overline{a}, a'_1), \ldots, h(\overline{a}, a'_{n_f})).$$

A fim de analisarmos estas definições recursivas, trataremos conjuntos indutivamente gerados como estruturas e definições recursivas como certas fórmulas em determinado sistema lógico (por exemplo, lógica de primeira-ordem).

Definimos abaixo o que chamaremos de estruturas indutivas.

Definição 2.16 *Sejam U um conjunto, $B \subseteq U$ e $\mathbb{F}$ uma família de funções sobre U. Seja A o conjunto indutivamente gerado a partir de U, B e $\mathbb{F}$. Então*
$$\mathfrak{A} = (A, \{b | b \in B\}, \{f|_A | f \in \mathbb{F}\})$$
é uma estrutura indutiva, *composta pelo domínio A, por uma constante para cada elemento $b \in B$ e pela restrição $f|_A$ de f a A, para cada $f \in \mathbb{F}$. Dada uma estrutura indutiva*
$$\mathfrak{A} = (A, \{b | b \in B\}, \{f|_A | f \in \mathbb{F}\}),$$

subentende-se imediatamente um alfabeto $S^{\mathfrak{A}} = \{c_b | b \in B\} \cup \{f | f \in \mathbb{F}\}$[6].

Se uma estrutura indutiva for construída a partir de um conjunto livremente gerado, nós a chamaremos de *estrutura indutiva livremente gerada* (ou simplesmente livremente gerada). De agora em diante, todos os conjuntos indutivamente gerados considerados serão livremente gerados, bem como as respectivas estruturas indutivas.

Exemplo 2.4 *Seja N, o conjunto dos naturais, o conjunto indutivo definido no Exemplo 2.3. N é livremente gerado. A estrutura $\mathfrak{N} = (N, 0, \sigma)$ é uma estrutura indutiva livremente gerada, onde σ é a restrição de S a N.*

Considere novamente as equações 2.1 e 2.2. O Teorema da Recursão garante que só existe uma função $+ : N^2 \rightarrow N$ que satisfaz essas equações. Isso significa que a estrutura $\mathfrak{N}$ pode ser expandida para exatamente uma estrutura $(\mathfrak{N}, +)$ que seja um modelo da fórmula

$$\phi^+ = \forall x((x + 0) = x) \wedge \forall x \forall y(x + \sigma(y) = \sigma(x + y)).$$

Ou seja, ϕ^+ é uma definição implícita na estrutura dos números naturais.

Considere a seguinte definição.

Definição 2.17 *Seja Γ um conjunto de S-sentenças de $\mathcal{L}$ (isto é, uma S-teoria de $\mathcal{L}$). Seja $\Delta(P)$ uma $S \cup \{P\}$-teoria de $\mathcal{L}$. Seja $\mathbb{C} = \{\mathfrak{A} | \mathfrak{A} \models_{\mathcal{L}} \Gamma\}$. Dizemos que $\Delta(P)$ define P implicitamente na teoria Γ se $\Delta(P)$ define implicitamente P em $\mathbb{C}$. Analogamente, $\Delta(P)$ define implicitamente e fortemente P na teoria Γ se $\Delta(P)$ define implicitamente e fortemente P em $\mathbb{C}$.*

De forma geral, definições recursivas como a da fórmula ϕ^+ são definições na estrutura indutiva livremente gerada. Isso é garantido pelo Teorema da Recursão (veja Teorema 2.12 abaixo). Entretanto, nem sempre definições recursivas são definições na teoria de estruturas indutivas em um determinado sistema lógico $\mathcal{L}$. Veja o caso da lógica de primeira-ordem. Se considerarmos a teoria $Th(\mathfrak{N})$ do conjunto dos números naturais com 0 e σ, sabemos que essa teoria possui modelos não-padrões, isto é, modelos não isomórficos a $\mathfrak{N}$. Isso acaba impedindo o uso do Teorema da Recursão a fim de provarmos que ϕ^+ define $+$ em $Th(\mathfrak{N})$. De fato, $Th(\mathfrak{N}) \cup \{\phi^+\}$ não define $+$ implicitamente. Ou seja, sistemas de equações recursivas, como o definido pela fórmula ϕ^+, não são, em geral, definições implícitas na teoria de estruturas indutivas na lógica de primeira ordem.

[6]Estamos utilizando o mesmo símbolo (no caso f) para representar as funções de $\mathbb{F}$ e os símbolos funcionais do alfabeto S criado a partir de $\mathbb{F}$ e B. Fica claro pelo contexto a qual dos dois objetos estamos nos referindo.

Vamos definir precisamente o que chamamos de sistemas de equações recursivas. Vamos trabalhar com uma generalização de sistemas de equações recursivas que chamaremos de *sistemas recursivos*, nos quais, ao invés de definirmos funções por meio de equações, definiremos predicados através de equivalências.

Definição 2.18 *[Sistemas Recursivos] Seja $\mathfrak{A} = (A, \{b|b \in B\}, \{f|_A|f \in \mathbb{F}\})$ uma estrutura indutiva e $S^{\mathfrak{A}} = \{c_b|b \in B\} \cup \{f|f \in \mathbb{F}\}$ o alfabeto associado. Seja S um conjunto de símbolos tal que $S^{\mathfrak{A}} \subseteq S$ e P um símbolo predicativo*[7] *tal que $P \notin S$. Um* sistema recursivo para P baseado em S *é um conjunto de LFP-fórmulas Δ_P, para P $n+1$-ário, definido como*

$$\Delta_P = \{\phi^P_{c_b}|b \in B\} \cup \{\phi^P_f|f \in \mathbb{F}\}$$

e tal que:

- *para cada $b \in B$,*

$$\phi^P_{c_b} = \forall \overline{x}(P(\overline{x}, c_b) \leftrightarrow \psi^P_{c_b}(\overline{x}, c_b)),$$

  onde $\psi^P_{c_b}(\overline{x}, c_b)$ é uma S-fórmula de LFP e $\overline{x} = x_1, \ldots, x_n$.

- *para cada $f \in \mathbb{F}$ de aridade n_f,*

$$\phi^P_f = \forall \overline{x}\overline{y}(P(\overline{x}, f(\overline{y})) \leftrightarrow \psi^P_f(\overline{x}, \overline{y})),$$

  onde $\overline{x} = x_1, \ldots, x_n$, $\overline{y} = y_1, \ldots, y_{n_f}$ e $\psi^P_f(\overline{x}, \overline{y})$ é uma $S \cup \{P\}$-fórmula de LFP, nenhuma das variáveis $\overline{y}$ ocorre ligada em $\psi^P_f(\overline{x}, \overline{y})$ e em todos os átomos da forma $Pt_1 \ldots t_{n+1}$ que ocorrem em $\psi^P_f(\overline{x}, \overline{y})$ temos $t_{n+1} = y_i$ para algum $1 \leq i \leq n_f$.

Provaremos o seguinte teorema de nossa autoria.

Teorema 2.12 *Sejam $\mathfrak{A}$, S e P como na Definição 2.18. Seja $\mathfrak{A}'$ uma S-estrutura que seja uma expansão de $\mathfrak{A}$ e Δ_P um sistema recursivo para P baseado em S. Δ_P define implicitamente e fortemente P em $\mathfrak{A}'$.*

Prova. Seja $V = \wp(A^{n+1})$. Seja $G_B : B \to V$ definida, para cada $f \in \mathbb{F}$, como

$$G_B(b) := \{(\overline{a}, b) \in A^{n+1}|\mathfrak{A}' \models \psi^P_b[\overline{a}]\}.$$

[7]Vamos nos restringir ao caso em que sistemas recursivos estão "definindo" símbolos predicativos a fim de simplificar a notação e as definições. As modificações necessárias na Definição 2.18 a fim de admitir o uso de símbolos funcionais são imediatas.

Seja $G_f : A^{n_f} \times V^{n_f} \to V$ definida como

$$G_f(\overline{a'}, X_1, \ldots, X_{n_f}) := \{(\overline{a}, f(\overline{a'})) \in A^{n+1} | (\mathfrak{A}', \bigcup_{1 \le i \le n_f} X_i) \models \psi_f^P(\overline{x}, \overline{y})[\overline{a}, \overline{a'}]\}.$$

Pelo Teorema da Recursão, existe exatamente uma função $h : A \to V$ tal que

1. para cada $b \in B$,
$$h(b) = G_B(b);$$
2. para cada $\overline{a'} = a'_1, \ldots, a'_{n_f} \in A^{n_f}$ e $f \in \mathbb{F}$,
$$h(f(\overline{a'})) = G_f(\overline{a'}, h(a'_1), \ldots, h(a'_{n_f})).$$

Como $\mathfrak{A}$ é livremente gerado, temos que, para todo $a \in A$,

$$h(a) = \{(\overline{a}, a) \in A^{n+1} | (\overline{a}, a) \in \bigcup \mathbf{img}(h)\},$$

onde $\mathbf{img}(h)$ é a imagem de h. Seja então

$$P^A := \bigcup \mathbf{img}(h).$$

Pela definição de h, $(\mathfrak{A}', P^A)$ é modelo de Δ_P.

Seja P'^A tal que $(\mathfrak{A}', P'^A)$ é modelo de Δ_P. Seja $h' : A \to V$ tal que

$$h'(a) = \{(\overline{a}, a) \in A^{n+1} | (\overline{a}, a) \in P'^A\}.$$

É fácil ver, por indução em $a \in A$, que h' satisfaz os Items 1 e 2 acima. Logo, pelo Teorema da Recursão—que garante a unicidade de h—temos $h' = h$. Logo $P^A = P'^A$. E portanto só existe uma expansão $(\mathfrak{A}', P^A)$ de $\mathfrak{A}'$ que é modelo de Δ_P. Logo Δ_P define implicitamente e fortemente P em $\mathfrak{A}'$. ∎

Seja $\mathfrak{A} = (A, \{b | b \in B\}, \{f | f \in \mathbb{F}\})$ uma estrutura indutiva e $\mathfrak{A}'$ uma S-estrutura que seja uma expansão de $\mathfrak{A}$ para algum alfabeto $S \supset S^{\mathfrak{A}}$. Se A for infinito, sabemos que a teoria $Th^{\mathrm{FO}}(\mathfrak{A})$ não define $\mathfrak{A}$ a menos de isomorfismo. Além disso, como dissemos, em geral, sistemas recursivos de primeira-ordem[8] não são definições implícitas na teoria $Th^{\mathrm{FO}}(\mathfrak{A}')$ de primeira-ordem de $\mathfrak{A}'$. Entretanto, se B e $\mathbb{F}$ forem finitos, a teoria $Th^{\mathrm{LFP}}(\mathfrak{A}')$ define $\mathfrak{A}'$ a menos de isomorfismo. Veja o teorema abaixo.

[8]Sistemas recursivos de primeira-ordem são sistemas recursivos em que as fórmulas ϕ_b^P e ϕ_f^P, para $b \in B$ e $f \in \mathbb{F}$, são de primeira-ordem.

Teorema 2.13 *Seja* $\mathfrak{A} = (A, \{b|b \in B\}, \{f|_A|f \in \mathbb{F}\})$ *uma estrutura indutiva onde* B *e* $\mathbb{F}$ *são finitos. Seja* $\mathfrak{A}'$ *uma* S*-expansão de* $\mathfrak{A}$*. Então* $Th^{\mathrm{LFP}}(\mathfrak{A}')$ *define* $\mathfrak{A}'$ *a menos de isomorfismo.*

Prova. Seja $\phi(X, x)$ a seguinte fórmula:

$$\phi(X,x) := (\bigvee_{b\in B} x = c_b) \vee (\bigvee_{f\in\mathbb{F}} \exists\overline{y^f}((\bigwedge_{1\leq i\leq n_f} X(y_i)) \wedge x = f(\overline{y^f}))),$$

onde $\overline{y^f} = y_1, \ldots, y_{n_f}$ e n_f é a aridade de f. Assim, $\phi(X, x)$ afirma que x é igual a c_b para algum $b \in B$ ou é obtido a partir da aplicação de algum $f \in \mathbb{F}$ a elementos de X. Seja

$$\psi(y) = [\mathbf{lfp}_{X,x}\phi(X,x)](y).$$

$[\mathbf{lfp}_{X,x}\phi(X,x)]$ define o menor predicado que contém os elementos representados por c_b, $b \in B$, e fechado com relação às operações representadas por f, $f \in \mathbb{F}$, ou seja o menor conjunto indutivo. Segue-se, portanto, que

$$\mathfrak{A}' \models \forall y(\psi(y)).$$

Observe ainda que, como $\mathfrak{A}$ é livremente gerada, $\mathfrak{A}'$ satisfaz a sentenças γ da lógica de primeira-ordem que afirma que:

- nenhuma constante c_b está na imagem de alguma função f, $f \in \mathbb{F}$;
- as funções $f \in \mathbb{F}$ são injetivas;
- as imagens das funções em $\mathbb{F}$ são duas a duas disjuntas.

Decorre dessas observações que qualquer S-estrutura $\mathfrak{C}'$ que seja modelo de $Th^{\mathrm{LFP}}(\mathfrak{A}')$ tem como seu $S^{\mathfrak{A}}$-reduto $\mathfrak{C}$ uma estrutura indutiva livremente gerada. Seja então $\mathfrak{C}'$ uma S-estrutura tal que $\mathfrak{C}' \models Th^{\mathrm{LFP}}(\mathfrak{A}')$. Seja $h : A \to C$ uma função definida como:

- para todo $b \in B$, $h(b) = c_b^C$;
- para todos $a_1, \ldots, a_{n_f} \in A$ e $f \in \mathbb{F}$,

$$h(f(a_1, \ldots, a_{n_f})) = f^C(h(a_1, \ldots, a_{n_f})).$$

Pelo Teorema da Recursão, h está bem definida. Por indução em $a \in A$ e em $c \in C$, respectivamente, podemos provar que h é injetiva e sobrejetiva.

Logo, por definição de h, temos que o $S^{\mathfrak{A}}$-reduto $\mathfrak{A}$ de $\mathfrak{A}'$ é isomórfico ao $S^{\mathfrak{A}}$-reduto $\mathfrak{C}$ de $\mathfrak{C}'$ (h é o isomorfismo entre $\mathfrak{A}$ e $\mathfrak{C}$).

Também por indução em $a \in A$ e em $c \in C$, podemos provar que para todo elemento $a \in A$ e $c \in C$ existem $S^{\mathfrak{A}}$-termos t_a e t_c tais que $t_a^{\mathfrak{A}} = a$ e $t_c^{\mathfrak{C}} = c$. Mais ainda é possível mostrar que, para cada elemento $a \in A$, $t_a^{\mathfrak{C}} = h(a)$ (isso decorre diretamente do fato de h ser um isomorfismo entre $\mathfrak{A}$ e $\mathfrak{C}$).

Considere, portanto, um símbolo (por exemplo , predicativo) $P \in S$ n-ário. Temos que, para todos $a_1, \ldots, a_n \in A$,

$$\begin{aligned} P^A a_1 \ldots a_n \quad &\text{sss} \quad P t_{a_1} \ldots t_{a_n} \in Th^{\text{LFP}}(\mathfrak{A}') \\ &\text{sss} \quad P^C t_{a_1}^{\mathfrak{C}} \ldots t_{a_n}^{\mathfrak{C}} \\ &\text{sss} \quad P^C h(a_1) \ldots h(a_n). \end{aligned}$$

Segue-se então que h também é um isomorfismo entre $\mathfrak{A}'$ e $\mathfrak{C}'$. ∎

Como $Th^{\text{LFP}}(\mathfrak{A}')$ define $\mathfrak{A}'$ a menos de isomorfismo se $\mathfrak{A}'$ é uma expansão de uma estrutura indutiva livremente gerada $\mathfrak{A}$ tal que $S^{\mathfrak{A}}$ é finito, obtemos, junto como Teorema 2.12, o seguinte corolário.

Corolário 2.14 *Seja $\mathfrak{A} = (A, \{b | b \in B\}, \{f|_A | f \in \mathbb{F}\})$ uma estrutura indutiva onde B e $\mathbb{F}$ são finitos. Seja $\mathfrak{A}'$ uma S-expansão de $\mathfrak{A}$. Seja ainda Δ_P um sistema recursivo para P baseado em S. Então Δ_P define implicitamente e fortemente P em $Th^{\text{LFP}}(\mathfrak{A}')$.*

Em face da falha do Teorema de Beth para LFP, poderíamos nos questionar sobre a existência de definição explícita para predicados definidos através sistemas recursivos. Encerramos esta seção com o seguinte resultado, uma restrição do teorema de Beth para sistemas recursivos.

Teorema 2.15 (Definição Explícita para Sistemas Recursivos) *Seja $\mathfrak{A} = (A, \{b | b \in B\}, \{f|_A | f \in \mathbb{F}\})$ uma estrutura indutiva onde B e $\mathbb{F}$ são finitos. Seja $\mathfrak{A}'$ uma S-expansão de $\mathfrak{A}$. Seja ainda Δ_P um sistema recursivo para P $n+1$-ário baseado em S. Existe uma definição explícita de P para $Th^{\text{LFP}}(\mathfrak{A}') \cup \Delta_P$ em LFP.*

Prova. Um problema que enfrentamos a fim de encontrar uma definição explícita para P é que em sua definição, em termos do sistema recursivo Δ_P, P ocorre tanto positivamente quanto negativamente nas fórmulas ϕ_f^P, para $f \in \mathbb{F}$. Para contornar esse problema, construiremos, simultaneamente, o predicado P e o seu complemento $\widetilde{P}$. Seja

$$\begin{aligned}\xi'(P,\overline{x},x_{n+1}) \quad := \quad & (\bigvee_{b\in B}\psi_b^P(\overline{x},c_b)\wedge(x_{n+1}=c_b))\vee \\ & (\bigvee_{f\in\mathbb{F}}\exists\overline{y^f}(\psi_f^P(\overline{x},\overline{y})\wedge x_{n+1}=f(\overline{y^f}))).\end{aligned}$$

A fórmula $\xi'(P,\overline{x},x_{n+1})$ afirma que $(\overline{x},x_{n+1})$ é da forma $(\overline{x},c_b)$ onde $\overline{x}$ é obtido a partir de $\psi_b^P(\overline{x},c_b)$, ou da forma $(\overline{x},f(\overline{y}))$, para algum $\overline{y}$, e tal que $\overline{x}$ é obtido a partir de $\psi_f^P(\overline{x},\overline{y})$. Seja $\xi(P,\widetilde{P},\overline{x},x_{n+1})$ obtido a partir de $\xi'(P,\overline{x},x_{n+1})$ substituindo toda ocorrência negativa de P em $\xi'(P,\overline{x},x_{n+1})$ por $\widetilde{P}$. Seja

$$\begin{aligned}\widetilde{\xi'}(P,\overline{x},x_{n+1}) \quad := \quad & (\bigvee_{b\in B}\neg\psi_b^P(\overline{x},c_b)\wedge(x_{n+1}=c_b))\vee \\ & (\bigvee_{f\in\mathbb{F}}\exists\overline{y^f}(\neg\psi_f^P(\overline{x},\overline{y})\wedge x_{n+1}=f(\overline{y^f}))).\end{aligned}$$

A fórmula $\widetilde{\xi'}(P,\overline{x},x_{n+1})$ afirma que $(\overline{x},x_{n+1})$ é da forma $(\overline{x},c_b)$ onde $\overline{x}$ é obtido a partir de $\neg\psi_b^P(\overline{x},c_b)$, ou da forma $(\overline{x},f(\overline{y}))$, para algum $\overline{y}$, e tal que $\overline{x}$ é obtido a partir de $\neg\psi_f^P(\overline{x},\overline{y})$. Seja $\widetilde{\xi}(P,\widetilde{P},\overline{x},x_{n+1})$ obtido a partir de $\widetilde{\xi'}(P,\overline{x},x_{n+1})$ substituindo toda ocorrência negativa de P em $\widetilde{\xi'}(P,\overline{x},x_{n+1})$ por $\widetilde{P}$. Sejam $\xi(X,\widetilde{X},\overline{x},x_{n+1})$ e $\widetilde{\xi}(X,\widetilde{X},\overline{x},x_{n+1})$ obtidos a partir de $\xi(P,\widetilde{P},\overline{x},x_{n+1})$ e $\widetilde{\xi}(P,\widetilde{P},\overline{x},x_{n+1})$ substituindo P por X e $\widetilde{P}$ por $\widetilde{X}$.

Seja

$$\zeta'(\overline{y}) := [\mathbf{slfp}_{\overline{x},x_{n+1},X,\overline{x},x_{n+1},\widetilde{X}}\xi,\widetilde{\xi}](\overline{y}).$$

Seja $(\mathfrak{A}',\mathbf{P})$ modelo de $Th^{\mathrm{LFP}}(\mathfrak{A}')\cup\Delta_P$. Por indução nos elementos da estrutura indutiva livremente gerada $\mathfrak{A}$, facilmente se observa, utilizando a definição de Δ_P, que $(\mathbf{P},\widetilde{\mathbf{P}})$ é ponto fixo de $\Phi_{\emptyset,\emptyset}^{\xi,\widetilde{\xi}}$, onde $\widetilde{\mathbf{P}}$ é o complemento de $\mathbf{P}$. Mais ainda, é possível mostrar, utilizando novamente a definição de Δ_P, que $(\mathbf{P},\widetilde{\mathbf{P}})$ está contido em todos os pontos fixos de $\Phi_{\emptyset,\emptyset}^{\xi,\widetilde{\xi}}$. Logo

$$(\mathbf{P},\widetilde{\mathbf{P}}) = \mathbf{lfp}(\Phi_{\emptyset,\emptyset}^{\xi,\widetilde{\xi}}).$$

Portanto, $\zeta'(\overline{y})$ é uma definição explícita para P em Si-LFP. Mas para toda fórmula de Si-LFP existe uma equivalente em LFP no mesmo alfabeto—veja Lema 2.3. Logo existe uma fórmula $\zeta(\overline{y})$ em LFP equivalente a $\zeta'(\overline{y})$. Portanto existe definição explícita de P para $Th^{\mathrm{LFP}}(\mathfrak{A}')\cup\Delta_P$ em LFP. ∎

2.4 Löwenheim-Skolem

Nesta seção avançaremos um pouco mais no estudo da teoria dos modelos de LFP. Nós mostraremos que o Teorema de Löwenheim-Skolem também vale para conjuntos contáveis arbitrários de LFP-sentenças. O Teorema de Löwenheim-Skolem possui várias formas (isto é, existem vários teoremas que levam este nome), uma delas é a seguinte:

Definição 2.19 (Teorema de Löwenheim-Skolem) *Dizemos que o Teorema de Löwenheim-Skolem vale para a lógica $\mathcal{L}$ se, para toda sentença satisfatível $\phi \in \mathcal{L}$, ϕ possui um modelo contável.*

O Teorema de Löwenheim-Skolem é bastante útil para, por exemplo, provarmos que uma lógica $\mathcal{L}_1$ possui uma fórmula que não tem equivalente em $\mathcal{L}_2$: basta exibir uma fórmula de $\mathcal{L}_1$ que não possua modelos contáveis e utilizar o Teorema de Löwenheim-Skolem para $\mathcal{L}_2$ (claro, se este for o caso). Este artifício será utilizado na Seção 4.5 para mostrar que a lógica MIN_Δ é menos expressiva que a lógica de segunda-ordem.

Como é bem sabido, o Teorema de Löwenheim-Skolem vale para a lógica de primeira-ordem. Também é verdade que conjuntos satisfatíveis contáveis de fórmulas quaiquer em lógica de primeira-ordem possuem modelos contáveis. Na verdade, a seguinte generalização desse teorema pode ser provado para a lógica de primeira-ordem (veja [EFT94]).

Teorema 2.16 (Teorema de Löwenheim-Skolem Descendente) *Seja Γ um conjunto satisfatível de fórmulas da lógica de primeira-ordem, então Γ possui um modelo de cardinalidade não maior do que a cardinalidade de Γ.*

Em [Grä02, Flu99], Grädel e Flum, respectivamente, exibem provas de que o Teorema de Löwenheim-Skolem, como apresentado na Definição 2.19, vale para LFP. Na verdade eles mostram algo um pouco mais forte. Eles mostram que se ϕ é uma sentença de LFP e $\mathfrak{A}$ é um modelo de ϕ, então existe uma subestrutura $\mathfrak{A}'$ contável de $\mathfrak{A}$ que também é modelo de ϕ. Nós mostraremos algo semelhante, substituindo ϕ por um conjunto contável satisfatível de LFP-sentenças quaisquer, usando parte das provas de Grädel e Flum. Antes, vamos introduzir a seguinte definição.

Definição 2.20 (LFP-subestrutura) *Sejam $\mathfrak{A}$ uma S-estrutura. Uma S-subestrutura $\mathfrak{B} \subseteq \mathfrak{A}$ é uma LFP-subestrutura de $\mathfrak{A}$ (escreve-se $\mathfrak{B} \subseteq_{\mathrm{LFP}} \mathfrak{A}$) se, e somente se, para toda S-fórmula $\phi(\overline{x})$ de LFP sem variáveis relacionais livres e cujas variáveis livres de primeira-ordem estão entre $\overline{x} = x_1, \ldots, x_n$, e para quaisquer $b_1, \ldots, b_n \in B$, temos*

$$\mathfrak{A} \models \phi(\overline{x})[b_1, \ldots, b_n] \text{ sss } \mathfrak{B} \models \phi(\overline{x})[b_1, \ldots, b_n].$$

Nós apresentaremos uma prova de nossa autoria de que, se S e um conjunto de símbolos contável, então toda S-estrutura tem uma LFP-subestrutura contável. Seja $\mathfrak{A}$ uma S-estrutura. Nós definiremos os conjuntos $\mathbb{S} = \{S_i | i \in N\}$, onde cada S_i e um conjuntos de símbolos, e $\Psi = \{\psi_{lm}(X, \overline{x}, \overline{y}) | l, m \in N\}$, onde cada $\psi_{lm}(X, \overline{x}, \overline{y})$ é uma LFP-fórmula, por indução simultânea como segue:

1. $S_0 = S$;

2. $\{\psi_{0m} | m \in N\}$ é uma enumeração do cojunto das S_0-fórmulas de LFP com no máximo uma variável relacional livre X e tal que o comprimento de $\overline{x}$ é igual à aridade de X e, para cada $m \in N$, as váriáveis livres de primira-ordem de $\{\psi_{0m} | m \in N\}$ estão entre $\overline{x}, \overline{y}$;

3. $S_{i+1} = S_i \cup \{<_i\} \cup \{T_{ij} | j \in N\}$ onde cada T_{ij} é um novo símbolo relacional $v'_{ij} + v_{ij} + 1$-ário, onde v'_{ij} é a aridade de X e v_{ij} e o comprimento de $\overline{y}$ em $\psi_{lm}(X, \overline{x}, \overline{y})$;

4. $\{\psi_{(l+1)m}(X, \overline{x}, \overline{y}) | m \in N\}$ e uma enumeração do conjunto das $S(l+1)$-fórmulas de LFP com no máximo uma variável relacional livre X e tal que o comprimento de $\overline{x}$ é igual à aridade de X e, para cada $m \in N$, as váriáveis livres de primira-ordem de $\{\psi_{(l+1)m} | m \in N\}$ estão entre $\overline{x}, \overline{y}$;.

Nós definimos os conjuntos $\mathbf{S}_i^A$ de relações sobre A que serao atribuídas aos símbolos de S_i, $i \geq 0$, da seguinte forma:

1. $\mathbf{S}_0^A = \{R^A | R \in S\}$;

2. $\mathbf{S}_{i+1}^A = \mathbf{S}_i^A \cup \{\prec_i^A\} \cup \{\mathbf{T}_{ij}^A | j \in N\}$, onde $\prec_i^A$ é uma boa-ordem[9] sobre A de comprimento α_i definida como:

$$\alpha_i = \left(\bigcup_{\overline{a'} \in A, j \in N} \mathbf{cl}(\Phi_{\emptyset, \overline{a'}}^{\psi_{ij}}) \right) + 1,$$

e $\mathbf{T}_{ij}^A(a\overline{a'a''})$ sss a é o γ-ésimo elemento de $\prec_i^A$ e a'' pertence ao γ-ésimo estágio da indução em $\Phi_{\emptyset, \overline{a'}}^{\psi_{ij}}$, para $a \in A$, $\overline{a'} \in A^{v_{ij}}$ e $\overline{a''} \in A^{v'_{ij}}$. Note que essa relação codifica os estágios da indução sobre ψ_{ij}.

[9]Nós estamos assumindo como hipótese o Teorema da Boa-Ordenação (que equivale ao Axioma da Escolha), junto com ZF, nos Lemas 2.21 e 2.19.

Seja $S' = \bigcup_{i \in N, i>0} S_i$ um conjunto de símbolos e $\mathbf{S}'^A = \bigcup_{i \in N, i>0} \mathbf{S}_i^A$ um conjunto de relações sobre A interpretando S'. Seja $\mathfrak{A}' = (\mathfrak{A}, \mathbf{S}'^A)$ uma $S \cup S'$-estrutura. Seja $\mathfrak{B}' = (\mathfrak{B}, \mathbf{S}'^B)$ uma subestrutura elementar contável de $\mathfrak{A}'$ fechada sob fórmulas existenciais de LFP, isto é, para toda $S \cup S'$-formula $\exists x \phi(x, y_1, \ldots, y_n)$ de LFP sem variáveis relacionais livres e quaisquer $b_1, \ldots, b_n \in B$, existe $b \in B$ tal que, se $\mathfrak{A}' \models \exists x \phi(x, y_1, \ldots, y_n)[b_1, \ldots, b_n]$, então $\mathfrak{A}' \models \phi(x, y_1, \ldots, y_n)[b, b_1, \ldots, b_n]$.

Nós provaremos que $\mathfrak{B}'$ é uma LFP-subestrutura de $\mathfrak{A}'$. Para tanto o lema seguinte nos ajudará.

Lema 2.17 *Seja $\mathfrak{A}$ uma S-estrutura e $\psi(X, \overline{x}, \overline{y})$ uma S-fórmula, para algum conjunto de símbolos S, tal que o comprimento de $\overline{y}$ é v e o comprimento de $\overline{x}$ é igual à aridade v' de X. Seja $<$ uma boa-ordem sobre A e $\mathbf{T}$ uma relação $v + v' + 1$-ária sobre A. Seja η a fórmula seguinte (veja [Grä02]):*

$$\eta = \forall z \forall \overline{y} \forall \overline{x} (Tz\overline{y}\overline{x} \leftrightarrow (\exists w (w < z) \wedge \psi(\exists w (w < z \wedge Tw\overline{y}\_), \overline{x}, \overline{y}))),$$

onde $\psi(\exists w (w < z \wedge Tw\overline{y}\_), \overline{x}, \overline{y})$ é obtido a partir de $\psi(X, \overline{x}, \overline{y})$ substituindo $X(\overline{t'})$ por $\exists w (w < z \wedge Tw\overline{y}\overline{t'})$ para toda tupla de termos t'. Nesse caso,

$$(\mathfrak{A}, \mathbf{T}, <) \models \eta$$

sss o α-ésimo estágio da indução sobre $\Phi^{\psi}_{\emptyset, \overline{a'}}$ for

$$X^{\alpha} = \{\overline{a''} \in A^{v'} | (\mathfrak{A}, \mathbf{T}, <) \models \exists w (w < z) \wedge \psi(\exists w (w < z \wedge Tw\overline{y}\_), \overline{x}, \overline{y})[a\overline{a'}\overline{a''}]\},$$

onde, na equação acima, a, a', a'' são atribuídos a $z, \overline{y}, \overline{x}$, respectivamente, e a é o α-ésimo elemento na ordem $<$.

Prova. O lema pode ser facilmente demonstrado através de indução transfinita sobre os ordinais α e a definição dos estágios da indução sobre $\psi(X, \overline{x}, \overline{y})$ junto com o fato de que $<$ é uma boa-ordem. Basta notar que

$$\begin{aligned} X^0 &= \emptyset, \\ X^{\alpha} &= \Phi^{\psi}_{\emptyset, \overline{a'}}(\bigcup_{\mu' < \alpha} X^{\mu'}). \end{aligned}$$

∎

Nós obtemos imediatamente o seguinte corolário.

Corolário 2.18 *Sejam $\mathfrak{A}$, $<$, T e $\psi(X, \overline{x}, \overline{Y})$ como no Lema 2.17. As seguintes afirmações são equivalentes:*

1. *o comprimento de* $<$ *é maior que ou igual a* $\mathbf{cl}(\Phi^{\psi}_{\emptyset,\overline{a'}})$*;*

2. $(\mathfrak{A}, \mathbf{T}, <) \models \forall z \forall \overline{y} \forall \overline{x}(Tz\overline{yx} \leftrightarrow \psi(T, \overline{x}, \overline{y}))$*;*

3. *para cada tupla* $\overline{a'} \in A^v$*,*

$$\mathbf{lfp}(\Phi^{\psi}_{\emptyset,\overline{a'}}) = \{\overline{a''} \in A^{v'} | \textit{para algum } a \in A, \mathbf{T}(a\overline{a'a''})\}. \tag{2.3}$$

Prova. É facil ver que o Lema 2.17 implica que, se $(\mathfrak{A}, \mathbf{T}, <) \models \eta$, então

$$\{\overline{a''} \in A^{v'} | \text{para algum } a \in A, \mathbf{T}(a\overline{a'a''})\} \subseteq \mathbf{lfp}(\Phi^{\psi}_{\emptyset,\overline{a'}}). \tag{2.4}$$

Alem disso, se o comprimento de $<$ for maior que ou igual a $\mathbf{cl}(\Phi^{\psi}_{\emptyset,\overline{a'}})$, então a Equação 2.3 será verdadeira. Portanto o Item 1 implica o Item 3. A implicação do Item 3 no Item 1 é imediata. Note que a fórmula

$$\forall z \forall \overline{y} \forall \overline{x}(Tz\overline{yx} \leftrightarrow \psi(T, \overline{x}, \overline{y}))$$

afirma que para todo $\overline{a'} \in A^v$, o conjunto

$$\{\overline{a''} \in A^{v'} | \text{para algum } a \in A, \mathbf{T}(a\overline{a'a''})\}$$

é um ponto fixo de $\Phi^{\psi}_{\emptyset,\overline{a'}}$. Segue, apartir da inclusão 2.4, que o Item 2 implica o Item 1. ∎

Provaremos agora o nosso principal resultado dessa seção.

Lema 2.19 (LFP-Subestrutura Contável) $\mathfrak{B}' \subseteq_{\text{LFP}} \mathfrak{A}'$.

Prova. Seja o *grau* $\mathbf{gr}(\phi(\overline{y}))$ de $\phi(\overline{y})$ o par $\langle h, m \rangle$ onde h e o número de ocorrências de operadores de ponto fixo e m e o número de ocorrências de outros conectivos e quantificadores da LFP-fórmula $\phi(\overline{y})$, e tal que $\mathbf{gr}(\alpha) = \langle h', m' \rangle < \langle h'', m'' \rangle = \mathbf{gr}(\beta)$ se, e somente se, $h'' > h'$, ou $h'' = h'$ e $m'' > m'$. Nós provaremos, através de indução sobre o grau $\mathbf{gr}(\phi(\overline{y}))$ de uma $S \cup S'$-fórmula de $\phi(\overline{y})$ de LFP sem variáveis relacionais livres e tal que as variáveis (de primeira-ordem) de $\phi(\overline{y})$ estejam entre $\overline{y} = y_0, \dots, b_{n-1}$, que

$$\mathfrak{A}' \models \phi(\overline{y})[b_0, \dots, b_{n-1}] \text{ iff } \mathfrak{B}' \models \phi(\overline{y})[b_0, \dots, b_{n-1}] \tag{2.5}$$

para quaisquer $b_0, \dots, b_{n-1} \in B$. Na base da indução, quando $\mathbf{gr}(\phi(\overline{y})) = \langle 0, 0 \rangle$, a fórmula $\phi(\overline{y})$ não possui operadores de ponto fixo, portanto, é uma fórmula da lógica de primeira-ordem (na verdade, uma fórmula atômica). Como $\mathfrak{B}'$ é, por definição, uma subestrutura elementar de $\mathfrak{A}$, a Equivalência

2.5 vale. Seja $\langle h, m\rangle$ um par qualquer de naturais e suponha que a Equivalência 2.5 vale para toda $S\cup S'$ fórmula $\phi(\overline{y})$ de LFP com grau menor ou igual a $\langle h, m\rangle$ como Hipótese Indutiva. Nós mostraremos a seguir que a Equivalência 2.5 vale para todas as $S\cup S'$-fórmulas de LFP com grau $\langle h, m+1\rangle$ ou $\langle h+1, l\rangle$, para qualquer natural l. O caso difícil é $\phi(\overline{x}) = [\mathbf{lfp}_{X,\overline{x}}\psi](\overline{t})$. Nesse caso, ψ é uma das $\psi_{ij}(X, \overline{x}, \overline{y})$. Considere a relação $\mathbf{T}_{ij}^{A}$ definida acima. Conforme foi mencionado, $\mathbf{T}_{ij}^{A}$ codifica os estágios da indução sobre $\Phi_{\emptyset,\overline{a}}^{\psi_{ij}}$ para cada $\overline{a} \in A^{v'_{ij}}$, portanto, pelo Lema 2.17, obtemos

$$\mathfrak{A}' \models \eta_{ij},$$

onde

$$\eta_{ij} = \forall z \forall \overline{y} \forall \overline{x}(T_{ij} z \overline{y} \overline{x} \leftrightarrow (\exists w(w <_i z) \wedge \psi_{ij}(\exists w(w <_i z \wedge T_{ij} w \overline{y}\_), \overline{x}, \overline{y}))),$$

e

$$\psi_{ij}(\exists w(w <_i z \wedge T_{ij} w \overline{y}\_), \overline{x}, \overline{y})$$

é obtido a partir de $\psi_{ij}(X, \overline{x}, \overline{y})$ substituindo $X(\overline{t'})$ por $\exists w(w <_i z \wedge T_{ij} w \overline{y} \overline{t'})$ para toda tupla de termos $\overline{t'}$. Note que η_{ij} possui grau menor que o grau de $\phi(\overline{y})$.. Decorre, por Hipótese Indutiva, que $\mathfrak{B}' \models \eta_{ij}$. Como $\mathfrak{B}'$ é uma subestrura de $\mathfrak{A}'$, $\prec_i^B$ é também uma boa-ordem. Mais ainda, como o comprimento de α_i de $<_i$ é maior ou igual a $\mathbf{cl}(\Phi_{\emptyset,\overline{a'}}^{\psi_{ij}})$, pelo Corolário 2.18, $\mathfrak{A}' \models \forall z \forall \overline{y} \forall \overline{x}(T z \overline{y} \overline{x} \leftrightarrow \psi(T, \overline{x}, \overline{y}))$, e pela Hipótese Indutiva, $\mathfrak{B}' \models \forall z \forall \overline{y} \forall \overline{x}(T z \overline{y} \overline{x} \leftrightarrow \psi(T, \overline{x}, \overline{y}))$. Portanto, $\mathbf{T}_{ij}^{B}$ codifica os estágios da indução sobre $\widehat{\Phi}_{\emptyset,\overline{b}}^{\psi_{ij}}$, onde $\widehat{\Phi}_{\emptyset,\overline{b}}^{\psi_{ij}}$ é o operador monónoto definido por $\psi_{ij}(X, \overline{x}, \overline{y})$ sobre $B^{v'_{ij}+v_{ij}+1}$ e B é o domínio de $\mathfrak{B}'$. Novamente pelo Corolário 2.18, para qualquer tupla de termos $\overline{t}$ com variáveis entre $\overline{y}$, qualquer tupla $\overline{a}$ de elementos de A com o mesmo comprimento de $\overline{y}$, e qualquer tupla $\overline{b}$ de elementos de B também com o mesmo comprimento de $\overline{y}$, nós temos:

$(*)$ $\overline{t}^{\mathfrak{A}'}[\overline{a}] \in \mathbf{lfp}(\Phi_{\emptyset,\overline{a}}^{\psi_{ij}})$ sss $\mathfrak{A}' \models \exists z T_{ij} z \overline{y} \overline{t}[\overline{a}]$,

$(**)$ $\overline{t}^{\mathfrak{B}'}[\overline{b}] \in \mathbf{lfp}(\widehat{\Phi}_{\emptyset,\overline{b}}^{\psi_{ij}})$ sss $\mathfrak{B}' \models \exists z T_{ij} z \overline{y} \overline{t}[\overline{b}]$.

Segue, para cada $\overline{b} \in B^n$, que

$$\begin{aligned}
\mathfrak{A}' \models \phi(\overline{y})[\overline{b}] \quad & \text{sss } \mathfrak{A}' \models [\mathbf{lfp}_{X,\overline{x}}\psi_{ij}(X,\overline{x},\overline{y})](\overline{t})[\overline{b}] \\
& \text{sss } \overline{t}^{\mathfrak{A}'}[\overline{b}] \in \mathbf{lfp}(\Phi^{\psi_{ij}}_{\emptyset,\overline{b}}) \\
& \text{sss (por } (*)) \ \mathfrak{A}' \models \exists z T_{ij} z\overline{y}\overline{t}[\overline{b}] \\
& \text{sss (já que } \mathfrak{B}' \subseteq_{\mathrm{FO}} \mathfrak{A}') \ \mathfrak{B}' \models \exists z T_{ij} z\overline{y}\overline{t}[\overline{b}] \\
& \text{sss (por } (**)) \ \overline{t}^{\mathfrak{B}'}[\overline{b}] \in \mathbf{lfp}(\widehat{\Phi}^{\psi_{ij}}_{\emptyset,\overline{b}}) \\
& \text{sss } \mathfrak{B}' \models [\mathbf{lfp}_{X,\overline{x}}\psi_{ij}(X,\overline{x},\overline{y})](\overline{t})[\overline{b}] \\
& \text{sss } \mathfrak{B}' \models \phi(\overline{y})[\overline{b}].
\end{aligned}$$

■

Imediatamente obtemos $\mathfrak{B} \subseteq_{\mathrm{LFP}} \mathfrak{A}$ e $\mathfrak{B}$ é contável. Seja Γ um conjunto satisfatível e contável de LFP-fórmulas. Seja $\mathfrak{A}$ um modelo de Γ. Pelo Lema 2.19, existe $\mathfrak{B} \subseteq \mathfrak{A}$ contável. Segue que $\mathfrak{B} \models \Gamma$. Portanto, temos:

Teorema 2.20 (Löwenheim-Skolem para LFP-Teorias Contáveis) *Todo conjunto contável satisfatível de fórmulas de LFP que possui um modelo infinito possui um modelo contável.*

Podemos generalizar o Teorema 2.20 para obter um análogo ao Teorema de Löwenheim-Skolem Descendente. Basta substituirmos os conjuntos $\mathbb{S}$ e Ψ por $\mathbb{S}^{|S|}$ e $\Psi^{|S|}$ definidos abaixo. Seja S um conjunto de símbolos. A Aritmética Cardinal nos informa que a cardinalidade do conjunto das S-fórmulas é também menor que ou igual a $|S|$ (isto é, se $|S|$ for infinito), pois as fórmulas de LFP são seqüências finitas de símbolos. Definimos os conjuntos $\mathbb{S}^{|S|} = \{S_i^{|S|} | i \in N\}$ e $\Psi^{|S|} = \{\Psi_i^{|S|} | i \in N\}$ por indução simultânea (de forma análoga à anterior) como:

1 $S_0^{|S|} = S$;

2 $\Psi_l^{|S|}$ é o conjunto das S_l-fórmulas de LFP que possuem no máximo uma variável relacional livre X, e tais que o comprimento $\overline{x}$ é igual à aridade de X e as variáveis de primeira-ordem de $\psi_{l\alpha}(X, \overline{x}, \overline{y})$ estão entre $\overline{x}, \overline{y}$. Considere uma boa-ordenação $\{\psi_{l\alpha}(X, \overline{x}, \overline{y}) | \alpha \leq |\Psi_l^{|S|}|\}$ das fórmulas em $\Psi_l^{|S|}$ (observe que, se a cardinalidade de S_l é menor ou igual a $|S|$, o conjunto $\Psi_l^{|S|}$ tem cardinalidade menor ou igual a $|S|$);

3 $S_{i+1}^{|S|} = S_i^{|S|} \cup \{<_i\} \cup \{T_{i\alpha} | \alpha \leq |\Psi_i^{|S|}|\}$ onde cada $T_{i\alpha}$ é um novo símbolo relacional $u + v + 1$-ário, onde a aridade de X é u e o comprimento de $\overline{y}$ é v em $\psi_{i\alpha}(X, \overline{x}, \overline{y})$ (observe que S_{i+1} terá cardinalidade menor ou igual a $|S|$ se a cardinalidade de $\Psi_i^{|S|}$ for menor ou igual a $|S|$).

É fácil mostrar por indução que os conjuntos S_i^κ e Ψ_i^κ têm cardinalidade menor ou igual a κ, para $i \in N$. O mesmo para $\bigcup \mathbb{S}^\kappa$ e $\bigcup \Psi^\kappa$.

Seja $\mathfrak{A}$ uma S-estrutura de cardinalidade maior ou igual a κ. Nós definimos então os conjuntos $\mathbf{S}_i^A$, $i \in N$, de relações sobre A, que interpretam S_i, como:

- $\mathbf{S}_0^A = \{R^A | R \in S\}$;
- $\mathbf{S}_{i+1}^A = \mathbf{S}_i^A \cup \{\prec_i^A\} \cup \{\mathbf{T}_{i\alpha}^A | \alpha \in \kappa\}$, onde $\prec_i^A$ é uma boa-ordem sobre os elementos de A de comprimento α_i definido como:

  $$\alpha_i = \left(\bigcup_{\overline{a} \in A, \alpha \in \kappa} \mathbf{cl}(\Phi_{\emptyset,\overline{a}}^{\psi_{i\alpha}}) \right) + 1,$$

  e $\mathbf{T}_{i\alpha}^A(a\overline{b}\overline{c})$ se, e somente se, a é o γ-ésimo elemento na ordem $\prec_i^A$ e $\overline{c}$ pertence ao γ-ésimo estágio da indução sobre o operador $\Phi_{\emptyset,\overline{b}}^{\psi_{i\alpha}}$. Observe que $\mathbf{T}_{i\alpha}^A$ codifica os estágios da indução em $\Phi_{\emptyset,\overline{b}}^{\psi_{i\alpha}}$.

Observe que, como $|A| \geq \kappa$, a cardinalidade de $\{\mathbf{cl}(\Phi_{\emptyset,\overline{a}}^{\psi_{i\alpha}}) | \overline{a} \in A, \alpha \in \kappa\}$ é menor ou igual a $|A|$, e $\mathbf{cl}(\Phi_{\emptyset,\overline{a}}^{\psi_{i\alpha}}) < |A|^+$, para quaisquer $\overline{a} \in A$ e $\alpha \in \kappa$. Novamente, a Aritmética Cardinal mostra que $\alpha_i < |A|^+$, para todo $i \in N$, e portanto a boa-ordem $\prec_i^A$ existe[10], para todo $i \in N$.

Analogamente ao caso anterior, seja $S' = \bigcup \mathbb{S}$ um conjunto de símbolos e $\mathbf{S}'^A = \bigcup_{i \in N, i > 0} \mathbf{S}_i^A$ um conjunto de relações sobre A. Seja $\mathfrak{A}' = (\mathfrak{A}, \mathbf{S}'^A)$ uma $S \cup S'$-estrutura. Seja $\mathfrak{B}' = (\mathfrak{B}, \mathbf{S}'^B)$, onde $\mathfrak{B}$ é uma S-estrutura, uma subestrutura elementar de $\mathfrak{A}'$ de cardinalidade κ e fechada para fórmulas existenciais de LFP.

Temos o seguinte lema:

Lema 2.21 $\mathfrak{B}' \subseteq_{\mathrm{LFP}} \mathfrak{A}'$.

Prova. Análoga à do Lema 2.19. ∎

Novamente, temos $\mathfrak{A} \subseteq_{\mathrm{LFP}} \mathfrak{B}$, e, portanto:

Teorema 2.22 *Todo conjunto satisfatível Γ de LFP-fórmulas tem um modelo de cardinalidade menor ou igual a $|\Gamma|$.*

[10] Cf. 9.

Assim encerramos este capítulo. No próximo capítulo nós estudaremos a Circunscrição de McCarthy e as NATs (Nested Abnormality Theories) de Lifschitz. Falaremos sobre a falha do Teorema de Löwenheim-Skolem para Circunscrição e mostraremos que existe uma tradução linear de fórmulas da lógica de segunda-ordem para NATs. Essa tradução garante que, para toda teoria na lógica de segunda-ordem, existe uma NAT que é uma extensão conservativa dessa teoria de segunda-ordem. Examinaremos a validade e teceremos alguns comentários acerca de uma afirmação de Doyle em [Doy85] sobre da definibilidade em Circunscrição e provaremos alguns resultados relacionados ao assunto.

Capítulo 3

Circunscrição

Neste capítulo, abordaremos o conceito de modelo minimal e Circunscrição de Predicados. Modelos minimais, que formam a base da semântica da Circunscrição, serão definidos na Seção 3.2. Na Seção 3.1 definiremos a Circunscrição. Na Seção 3.3, provaremos alguns teoremas a respeito do poder expressivo da Circunscrição. Apresentaremos uma prova de que o teorema de Löwenheim-Skolem "para baixo" não vale para Circunscrição na Seção 3.4. Na Seção 3.5, examinaremos a validade de uma afirmação feita por Doyle a respeito de definibilidade em Circunscrição. Na Seção 3.6, apresentaremos uma extensão de Circunscrição, chamada NAT, proposta por Lifschitz em [Lif95] na qual é possível realizar circunscrições aninhadas. Provaremos que, para qualquer teoria de segunda-ordem Γ, existe uma NAT que é uma extensão conservativa de Γ. Para tanto exibiremos uma tradução entre lógica de segunda-ordem e NAT. O tempo para computar essa tradução e o tamanho da NAT resultante são lineares no tamanho da fórmula de segunda-ordem na forma normal prenex.

3.1 Circunscrição de Predicados

O raciocínio é uma atividade amplamente estudada pela Inteligência Artificial (IA). Em particular, os formalismos lógicos são bastante utilizados a fim de se obter sistemas capazes raciocinar. Por "raciocinar" entendemos o ato de tirar conclusões a partir de um conjunto de premissas. Essas conclusões podem servir, por exemplo, para nos guiar em nossas decisões e ações ou simplesmente para explicitar relações entre conceitos descritos, ou parcialmente descritos, pelo nosso conjunto de premissas.

Em muitos problemas de IA, como acontece no dia-a-dia de um ser humano, precisamos desenvolver raciocínio a partir de um conjunto de premissas incompleto, no sentido de que esse conjunto de premissas, que chamaremos também de base de conhecimento, não é capaz de responder às perguntas necessárias ou, em outras palavras, não determina a veracidade ou a falsidade de certas proposições. No entanto precisamos decidir que ações tomar baseados nesse conhecimento incompleto. Ou seja, precisamos fazer inferências que vão além do que se pode deduzir a partir das informações das quais dispomos. E como não se pode garantir a corretude dessas inferências, podemos nos encontrar em uma situação na qual descobrimos que uma inferência feita previamente é falsa. Isso pode ocorrer se aumentarmos nossa base de conhecimento. Portanto, uma lógica adequada para lidar com raciocínio sobre conhecimento incompleto deve ter a propriedade da não monotonicidade.

A necessidade de tratar problemas desse tipo em IA motivou o estudo e criação de lógicas não monotônicas. Entre elas encontramos as lógicas *default*, dentre as quais se destaca a Lógica *Default* de Reiter [Rei80], algumas lógicas modais não monotônicas, como a de McDermont e Doyle [MD80], entre outras. Um dos formalismos baseados em lógica mais estudados é a Circunscrição.

A Circunscrição foi desenvolvida por McCarthy a fim resolver o Problema da Qualificação [McC80], que é o problema de se fornecer uma representação lógica (axiomática) do mundo, ou de uma situação particular, a fim de que uma máquina inteligente possa trabalhar sobre esses dados e inferir que decisões tomar. A idéia da Circunscrição é que, ao nos deparamos com a descrição de uma cena, imediatamente concluímos que os objetos que participam de fato da cena, ou as propriedades das quais eles desfrutam, são exatamente aqueles que constam na descrição. Na *Circunscrição de Predicados*, McCarthy tenta atacar esse problema da seguinte forma. Ao circunscrever um predicado P em uma teoria T, pretende-se obter uma nova teoria T' na qual os objetos que possuem a propriedade P são apenas aqueles que necessitam possuir a propriedade P para que T seja satisfeita. Ou seja, pretende-se que os modelos de T' sejam os modelos nos quais o predicado P possui as suas menores extensões possíveis, isto é, os modelos minimais de T.

Em [McC80], McCarthy introduz a Circunscrição de Predicados como uma esquema de fórmulas de primeira-ordem. Em [McC86], é apresentada a *Circunscrição de Fórmula*, uma generalização da Circunscrição de Predicados, dessa vez em lógica de segunda-ordem, que permite a circunscrição de uma fórmula (na verdade do predicado definido por uma fórmula) e que certos objetos (constantes, funções, predicados) possam variar. Outra generalização é a *Circunscrição Paralela*, na qual várias fórmulas são circunscritas simultaneamente em uma teoria. Aqui, trataremos da Circunscrição de Predicados Paralela (CPP) que é a restrição da Circunscrição Paralela para o caso em que a fórmula circunscrita é um átomo da forma $P(\overline{x})$. Veja [Lif94] boa introdução à Circunscrição e outras referências.

A Circunscrição de Predicados Paralela é definida em termos de uma tradução para a lógica de segunda-ordem. Nós a definimos abaixo.

Definição 3.1 (Definição de Circunscrição) *Sejam* $\overline{P} = P_1, \ldots, P_n$ *e* $\overline{Z} = Z_1, \ldots, Z_m$ *tuplas de símbolos predicativos em um alfabeto* S *onde a aridade de cada* P_i *é* k_i, $1 \leq i \leq n$, *e* $\overline{p} = P'_1, \ldots, P'_n$ *e* $\overline{z} = Z'_1, \ldots, Z'_m$ *tuplas de variáveis de segunda ordem onde cada* P'_i *possui a mesma aridade de* P_i, $1 \leq i \leq n$ *e cada* Z'_i *possui a mesma aridade de* Z_i, $1 \leq i \leq m$. *A* circunscrição de $\overline{P}$ em $\alpha(\overline{P}, \overline{Z})$ variando $\overline{Z}$, $Circ[\alpha(\overline{P}, \overline{Z}); \overline{P}; \overline{Z}]$, *é definida como a seguinte fórmula da lógica de segunda-ordem:*

$$\alpha(\overline{P}, \overline{Z}) \wedge \forall \overline{P'} \forall \overline{Z'} (\alpha(\overline{P'}, \overline{Z'}) \rightarrow \neg(\overline{P'} < \overline{P})), \tag{3.1}$$

onde $\overline{P'} < \overline{P}$ *é definido como*

$$\overline{P'} \leq \overline{P} \wedge \neg \overline{P'} = \overline{P},$$

e $\overline{P'} \leq \overline{P}$ *e* $\overline{P'} = \overline{P}$ *são definidos como*

$$\bigwedge_{1 \leq i \leq n} \forall x_1, \ldots, x_{k_i} (P'(x_1, \ldots, x_{k_i}) \rightarrow P(x_1, \ldots, x_{k_i}))$$

e

$$\bigwedge_{1 \leq i \leq n} \forall x_1, \ldots, x_{k_i} (P'(x_1, \ldots, x_{k_i}) \leftrightarrow P(x_1, \ldots, x_{k_i})),$$

respectivamente. Os predicados em $\overline{P}$ *são os* predicados circunscritos*, os predicados em* $\overline{Z}$ *são os* predicados variantes *e os demais predicados que ocorrem em* α *são os* predicados fixos. *Quando a tupla de símbolos* $\overline{Z}$ *for vazia, escrevemos* $Circ[\alpha; \overline{P}]$. *Na nossa linguagem admitiremos a presença de símbolos proposicionais. Desta forma, será possível circunscrevê-los e variá-los em uma circunscrição. Para tanto, dados símbolos* p *e* p', *definimos* $p' < p$ *como*

$$p' < p := (p' \rightarrow p) \wedge \neg(p' \leftrightarrow p)$$

e $\forall p(\psi(p))$ *é o mesmo que* $\psi(true) \wedge \psi(false)$.

Utilizamos Circunscrição para fazer inferências não monotônicas através do que Lifschitz chama de Teorias Circunscritivas. Uma Teoria Circunscritiva é um par $\langle T, \Delta \rangle$ onde T é um conjunto de sentenças da lógica de primeira-ordem e Δ é um conjunto de *regras de circunscrição*. Uma regra de circunscrição é uma expressão do tipo

$$\textbf{circ } P \textbf{ var } \overline{Z},$$

onde $\overline{Z}$ é uma tupla de objetos (símbolos relacionais, funcionais e constantes) e P é um predicado. Dada uma teoria circunscritiva $\langle T, \Delta \rangle$, definimos a relação de conseqüência $\models_{\text{Circ}}$ da seguinte forma:

$$\langle T, \Delta \rangle \models_{\text{Circ}} \phi \text{ sss } \left(\bigwedge_{\textbf{circ } P \textbf{ var } \overline{Z} \in \Delta} Circ[T; P; \overline{Z}] \right) \models \phi$$

Na próxima seção examinaremos o conceito de modelo minimal e sua relação com Circunscrição.

3.2 Modelos Minimais

Modelos Minimais formam a base da semântica da Circunscrição. Nesta seção, apresentaremos as definições a respeito de modelos minimais que serão utilizadas ao longo do texto.

Primeiramente, vamos relembrar certos conceitos de ordem comuns na Teoria dos Conjuntos.

Definição 3.2 *Uma relação binária $\leq$ em um conjunto D é uma* pré-ordem *se for transitiva e reflexiva.*

Definição 3.3 *Seja $\leq$ uma pré-ordem em D e $C \subseteq D$. Um elemento $c \in C$ é um* elemento minimal de C com relação a $\leq$ *se, e somente se, para todo elemento $c' \in C$, se $c' \leq c$ então $c \leq c'$.*

Definiremos agora a relação de ordem, na verdade uma pré-ordem, da qual extrairemos nossos modelos minimais. Essa relação ordena estruturas (veja [Lif94]).

Definição 3.4 *Seja S um alfabeto e sejam $\overline{P} = P_1, \ldots, P_n$ e $\overline{Z} = Z_1, \ldots, Z_m$ uma tupla de predicados de S e uma tupla de variáveis predicativas, respectivamente. Definimos a relação $\leq^{\overline{P};\overline{Z}}$ sobre as S-estruturas da seguinte forma: $\mathfrak{A} \leq^{\overline{P};\overline{Z}} \mathfrak{A}'$ se, e somente se,*

1. *$A = A'$, ($\mathfrak{A}$ e $\mathfrak{A}'$ possuem o mesmo domínio);*
2. *$R^{\mathfrak{A}} = R^{\mathfrak{A}'}$, para $R \notin \{\overline{P}, \overline{Z}\}$;*
3. *$P_i^{\mathfrak{A}} \subseteq P_i^{\mathfrak{A}'}$, para $1 \leq i \leq n$.*

Aqui, estenderemos a definição de $\leq^{\overline{P};\overline{Z}}$ para S-interpretações. Essa extensão será amplamente utilizada no Capítulo 4.

Definição 3.5 *Duas S-interpretações $\mathfrak{I} = (\mathfrak{A}, \beta)$ e $\mathfrak{I}' = (\mathfrak{A}', \beta')$ são tais que $\mathfrak{I} \leq^{\overline{P};\overline{Z}} \mathfrak{I}'$ se, e somente se, as condições $1, 2, 3$ da Definição 3.4 são satisfeitas bem como a seguinte condição adicional:*

4 $\beta = \beta'$.

Os modelos minimais dos quais trataremos são definidos da seguinte forma.

Definição 3.6 *Dada uma classe $\mathbb{C}$ de $S \cup \overline{P}, \overline{Z}$-interpretações, uma interpretação $\mathfrak{I} \in \mathbb{C}$ é um* modelo $\overline{P}; \overline{Z}$-minimal de $\mathbb{C}$ *se não existe um modelo $\mathfrak{I}' \in \mathbb{C}$ diferente de $\mathfrak{I}$ tal que $\mathfrak{I}' \leq^{\overline{P};\overline{Z}} \mathfrak{I}$. Se ϕ é uma fórmula, $\mathfrak{I}$ é um modelo $\overline{P}; \overline{Z}$-minimal de ϕ se $\mathfrak{I}$ é um modelo $\overline{P}; \overline{Z}$-minimal de $Mod(\phi)$, onde $Mod(\phi)$ é a classe dos modelos de ϕ. Se $\overline{P}$ for composto apenas pelo predicado P, nós dizemos que um modelo $\mathfrak{I}$ $P; \overline{Z}$-minimal de ϕ é não vazio se $P^{\mathfrak{I}} \neq \emptyset$.*

Como é amplamente sabido, os modelos de $Circ[\alpha; \overline{P}; \overline{Z}]$ são exatamente os modelos de α minimais com relação a $\leq^{\overline{P};\overline{Z}}$ (veja [Lif94]).

Teorema 3.1 *Seja α uma S-sentença de primeira-ordem. Uma S-estrutura $\mathfrak{A}$ é modelo de $Circ[\alpha; \overline{P}; \overline{Z}]$ se, e somente se, $\mathfrak{A}$ é modelo de α minimal com relação a $\leq^{\overline{P},\overline{Z}}$.*

Na próxima seção, examinaremos o poder expressivo da Circunscrição.

3.3 Expressividade da Circunscrição

Nesta Seção examinaremos o poder expressivo da CPP. Mostraremos sua relação com a lógicas de primeira-ordem e segunda ordem. Começaremos definindo precisamente a linguagem da CPP.

Definição 3.7 *A linguagem $\mathcal{L}_C$ da CPP é definida como sendo o conjunto da sentenças da lógica de primeira-ordem junto com as sentenças da lógica de segunda ordem da forma $Circ[\alpha; \overline{P}; \overline{Z}]$ para alguma sentença de primeira-ordem α. Se S é um alfabeto então $\mathcal{L}_C(S)$ é o fragmento de $\mathcal{L}_C$ cujas sentenças tanto de primeira-ordem quanto de segunda ordem são escritas no alfabeto S.*

Mostraremos agora que a CPP é mais expressiva que a lógica de primeira-ordem.

Exemplo 3.1 *Seja $\alpha(R)$ a sentença*

$$\forall x(R(x,x)) \wedge \forall x \forall y \forall z((R(x,y) \wedge E(y,z)) \rightarrow R(x,z)).$$

Em um modelo $\mathfrak{A}$ qualquer, $\alpha(P)$ diz que $R^{\mathfrak{A}}$ contém o fecho reflexivo e transitivo de $E^{\mathfrak{A}}$. Seja $\phi = Circ[\alpha(R), R]$. Então em um modelo $\mathfrak{A}$ qualquer, ϕ diz que $R^{\mathfrak{A}}$ é a menor relação que contém o fecho reflexivo e transitivo de $E^{\mathfrak{A}}$, portanto, é o próprio fecho reflexivo e transitivo de $E^{\mathfrak{A}}$.

Com um fácil argumento utilizando o Teorema da Compacidade para a lógica de primeira-ordem, mostraremos que a classe dos modelos da sentença ϕ do exemplo anterior não é sequer Δ-elementar.

Lema 3.2 *Não existe um conjunto de $S \cup \{E, R\}$-fórmulas Γ em lógica de primeira-ordem tal que, para qualquer modelo $\mathfrak{A}$ de Γ, $R^{\mathfrak{A}}$ seja o fecho transitivo e reflexivo de $E^{\mathfrak{A}}$.*

Prova. Suponha que tal conjunto Γ exista. Seja Δ tal que

$$\begin{aligned} \Delta \;=\; & \Gamma \cup \{R(x,y), x \neq y, \neg E(x,y), \neg \exists v_1(E(x,v_1) \wedge E(v_1,y)), \\ & \neg \exists v_1 \exists v_2(E(x,v_1) \wedge E(v_1,v_2) \wedge E(v_2,y)), \ldots\}. \end{aligned}$$

As fórmulas adicionadas a Γ dizem que os elementos x e y são diferentes e que y não pode ser alcançado a partir de x através de E num número finito de passos. Observe que cada subconjunto finito de Δ é satisfatível, logo, pela Compacidade da lógica de primeira-ordem, o conjunto Δ é satisfatível. Mas, nesse caso, existe um modelo $\mathfrak{A}$ de Γ para o qual $R^{\mathfrak{A}}$ não é o fecho transitivo e reflexivo de $E^{\mathfrak{A}}$ (os elementos $x^{\mathfrak{A}}$ e $y^{\mathfrak{A}}$ não pertencem ao fecho transitivo e reflexivo de $E^{\mathfrak{A}}$). Absurdo! ∎

Assim obtemos:

Teorema 3.3 *$FO < \mathcal{L}_C$ em poder expressivo.*

A CPP possui, entretanto, limites de expressividade. Mostraremos agora que a CPP não é sistema lógico booleano, uma vez que não é fechada para a negação.

Lema 3.4 *Existem sentenças $\phi \in \mathcal{L}_C$ para as quais não existem sentenças $\psi \in \mathcal{L}_C$ tais que $\models \neg\phi \leftrightarrow \psi$.*

Prova. Seja ϕ a circunscrição do Exemplo 3.1. Suponha que existe $\psi \in \mathcal{L}_C$ tal que $\models \neg\phi \leftrightarrow \psi$. Se ψ fosse primeira-ordem, então ϕ seria equivalente a uma sentença de primeira-ordem, e pelo Lema 3.2 isso não é possível. Suponha então que $\psi = Circ[\beta; \overline{P}; \overline{Z}]$ para alguma sentença de primeira-ordem β. Se $E, R \notin \overline{P}$, então ψ é equivalente a uma sentença de primeira-ordem, o que, como já vimos, não é possível. Como quaisquer duas estruturas $\mathfrak{A} = (A, E^A, R^A)$ e $\mathfrak{A}' = (A, E^A, R'^A)$ tais que R^A e R'^A não são o fecho reflexivo e transitivo de E^A e $R^A \subset R'^A$, são modelos de $\neg\phi$, então $R \notin \overline{P}$. Analogamente, como quaisquer duas estruturas $\mathfrak{A} = (A, E^A, R^A)$ e $\mathfrak{A}' = (A, E'^A, R^A)$ tais que R^A não é o fecho reflexivo e transitivo de E^A nem de E'^A, e $E^A \subset E'^A$ são modelos de $\neg\phi$, então $E \notin \overline{P}$. Logo $E, R \notin \overline{P}$. Absurdo! ∎

Como podemos observar a partir da definição do operador $Circ$, uma sentença $\phi \in \mathcal{L}_C$ é equivalente a uma sentença em Π_1^1, pois temos

$$Circ[\alpha; \overline{P}; \overline{Z}]$$

$$=$$

$$\alpha(\overline{P}, \overline{Z}) \wedge \forall \overline{P'} \forall \overline{Z'} (\neg\alpha(\overline{P'}, \overline{Z'}) \vee \neg(\overline{P'} < \overline{P}))$$

$$\equiv$$

$$\forall \overline{P'} \forall \overline{Z'} (\alpha(\overline{P}, \overline{Z}) \wedge (\neg\alpha(\overline{P'}, \overline{Z'}) \vee \neg(\overline{P'} < \overline{P})).$$

Nós mostraremos que, dada uma S-sentença ψ de segunda-ordem em Π_1^1, existe $\phi \in \mathcal{L}_C$ em um alfabeto estendido $S \cup S'$ tal que $\phi \equiv \psi \wedge p_1$, onde p_1 é um símbolo proposicional em S'. Isso significa que ϕ e ψ provam os mesmos teoremas escritos no alfabeto S. Na Seção 3.6, nós generalizaremos este resultado para NATs. Para tanto, vamos precisar do seguinte lema, que é a base de ambos os resultados.

Lema 3.5 *Seja $\phi(\overline{X})$ uma S-fórmula and $p_1, \ldots, p_n$ uma tupla de letras proposicionais que não ocorrem em $\overline{X}$. Então*

$$Circ[(\neg\phi(\overline{X}) \vee p_1) \wedge p_2 \wedge \ldots \wedge p_n; p_1; \overline{X}] \wedge p_1 \equiv \forall \overline{X} \phi(\overline{X}) \wedge p_1 \wedge \ldots \wedge p_n.$$

Prova. Nós temos a seguinte cadeia de equivalências.

$$Circ[(\neg\phi(\overline{X}) \vee p_1) \wedge p_2 \wedge \ldots \wedge p_n; p_1; \overline{X}] \wedge p_1$$

$$=$$

$$(\neg\phi(X) \vee p_1) \wedge p_2 \wedge \ldots \wedge p_n \wedge \forall p' \forall \overline{X'} (((\neg\phi(\overline{X'}) \vee p') \wedge p_2 \wedge \ldots \wedge p_n) \rightarrow$$

$$\neg(p' < p_1)) \wedge p_1$$
$$\equiv$$
$$p_2 \wedge \ldots \wedge p_n \wedge \forall p' \forall \overline{X'}(((\neg\phi(\overline{X'}) \vee p') \wedge p_2 \wedge \ldots \wedge p_n) \rightarrow \neg(p' < p_1)) \wedge p_1$$
$$\equiv, \text{ já que } p' \text{ toma valores em } \{\mathit{true}, \mathit{false}\},$$
$$p_2 \wedge \ldots \wedge p_n \wedge \forall \overline{X'}(((\neg\phi(\overline{X'}) \vee \mathit{true}) \wedge p_2 \wedge \ldots \wedge p_n) \rightarrow \neg(\mathit{true} < p_1))\wedge$$
$$\forall \overline{X'}(((\neg\phi(\overline{X'}) \vee \mathit{false}) \wedge p_2 \wedge \ldots \wedge p_n) \rightarrow \neg(\mathit{false} < p_1)) \wedge p_1$$
$$\equiv, \text{ já que } \neg(\mathit{true} < p_1) \text{ é sempre verdadeiro},$$
$$p_2 \wedge \ldots \wedge p_n \wedge \forall \overline{X'}(((\neg\phi(\overline{X'}) \vee \mathit{false}) \wedge p_2 \wedge \ldots \wedge p_n) \rightarrow \neg(\mathit{false} < p_1)) \wedge p_1$$
$$\equiv$$
$$p_2 \wedge \ldots \wedge p_n \wedge \forall \overline{X'}(\neg((\neg\phi(\overline{X'}) \vee \mathit{false}) \wedge p_2 \wedge \ldots \wedge p_n) \vee \neg(\mathit{false} < p_1)) \wedge p_1$$
$$\equiv$$
$$p_2 \wedge \ldots \wedge p_n \wedge \forall \overline{X'}(\neg(\neg\phi(\overline{X'}) \wedge p_2 \wedge \ldots \wedge p_n)) \wedge p_1$$
$$\equiv$$
$$p_2 \wedge \ldots \wedge p_n \wedge \forall \overline{X'}(\phi(\overline{X'}) \vee \neg(p_2 \wedge \ldots \wedge p_n)) \wedge p_1$$
$$\equiv$$
$$\forall \overline{X} \phi(\overline{X}) \wedge p_1 \wedge p_2 \wedge \ldots \wedge p_n.$$

∎

Observe que as variáveis relacionais $\overline{X}$ ocorrem livres em $Circ[(\neg\phi(\overline{X}) \vee p_1) \wedge p_2 \wedge \ldots \wedge p_n; p_1; \overline{X}] \wedge p_1$, mas o valor dessas variáveis em uma interpretação $\mathfrak{I}$ não importa para a veracidade de $\mathfrak{I} \models Circ[(\neg\phi(\overline{X}) \vee p_1) \wedge p_2 \wedge \ldots \wedge p_n; p_1; \overline{X}] \wedge p_1$. Isto é, $Circ[(\neg\phi(\overline{X}) \vee p_1) \wedge p_2 \wedge \ldots \wedge p_n; p_1; \overline{X}] \wedge p_1$ é uma $S \cup \{\overline{X}\} \cup \{p_1, \ldots, p_n\}$-fórmula de segunda-ordem equivalente a uma $S \cup \{p_1, \ldots, p_n\}$-fórmula de segunda-ordem.

Do Lema 3.5 temos imediatamente:

Teorema 3.6 *Para toda S-sentença ϕ de Π^1_1 existe uma sentença ψ de $\mathcal{L}_C$ cujo fecho dedutivo é uma extensão conservativa do fecho dedutivo de ϕ.*

Em [dKK89], de Kleer e Konolige nos ensinam a eliminar os predicados fixos de uma CPP, isto é, é mostrado que toda sentença $\phi \in \mathcal{L}_C$ é equivalente, módulo símbolo predicativos adicionais, a uma fórmula $\psi \in \mathcal{L}_C$, em um alfabeto possivelmente maior, na qual todo predicado é circunscrito ou variante. Em [CEG92] Cadoli *et al.* exibe um método para a eliminação dos predicados variantes na avaliação da relação de conseqüência lógica $\models$. Isto é, com o intuito de estabelecer $\phi \models \psi$, onde $\phi \in \mathcal{L}_C$ e ψ é de primeira-ordem, é possível encontrar uma sentença $\phi' \in \mathcal{L}_C$, onde ϕ' é escrita em um alfabeto aumentado e não ocorrem símbolos predicativos variantes em ϕ' (ver Definição 3.1), e uma sentença θ de primeira-ordem (na verdade, proposicional, veja [CEG92]) também no alfabeto aumentado, tal que $\phi \models \psi$ se, e somente se, $\phi' \models \theta$.

Na próxima seção, comentaremos a falha do Teorema de Löwenheim-Skolem para $\mathcal{L}_C$.

3.4 A Falha do Teorema de Löwenheim-Skolem

A primeira publicação na qual encontramos um estudo mais aprofundado do poder expressivo de Circunscrição é em [Sch87]. Schlipf mostra que vários cardinais podem ser definidos[1] por Circunscrição. Em particular, Schlipf mostra que existe uma circunscrição que somente possui modelos incontáveis[2]. O Lema 3.5 também garante que o Teorema de Löwenheim-Skolem não vale para Circunscrição, pois é fácil construir uma fórmula em Π^1_1 que possui apenas modelos incontáveis. No entanto, o Lema 3.5 utiliza o fato da circunscrição poder admitir predicados variáveis. Abaixo, apresentaremos uma prova de que o teorema de Löwenheim-Skolem não vale[3]

[1]O conceito de definição usado por Schlipf é particular e está definido em [Sch87]. Para nós, o importante é que, quando Schlipf mostra que vários cardinais podem ser definidos, isso significa que existem circunscrições tais que todos os seus modelos têm cardinalidade igual ao cardinal definido (isto é, o cardinal κ pode ser definido por Circunscrição se existe uma circunscrição $\phi \in \mathcal{L}_C$ tal que todos os seus modelos têm cardinalidade κ).

[2]O exemplo de Schlipf usa predicados variáveis—para ser exato um certo predicado U (veja [Sch87, Exemplo 2.6]) está variando—mas o mesmo exemplo pode ser adaptado para provar que existe uma circunscrição sem objetos variantes que só possui modelos incontáveis—basta retirar o predicado U da lista de objetos variáveis na circunscrição citada.

[3]É importante frisar que a Circunscrição da qual estamos tratando aqui é aquela definida em lógica de segunda-ordem nos moldes da que é apresentada em [McC86]. No caso da Circunscrição de primeira-ordem, como apresentada em [McC80], é evidente que o Teorema de Löwenheim-Skolem vale. Esse teorema também vale para a lógica de primeira-

para Circunscrição mesmo sem objetos variáveis. Nosso contra exemplo utiliza um alfabeto menor que o utilizado em [Sch87].

Antes de provarmos esse teorema, vamos relembrar os seguintes conceitos da teoria dos conjuntos.

Seja P um conjunto e $<$ uma relação binária em P. $<$ é uma *ordem estrita* se $<$ for transitiva ($a < b$ e $b < c$ implica $a < c$) e assimétrica ($a < b$ implica não $b < a$). Uma ordem estrita é *linear* (ou *total*) se, para todo $a, b \in P$, $a < b$,ou $a = b$, ou $b < a$. Uma ordem estrita é *densa* se, sempre que $a < b$, existe c tal que $a < c$ e $c < b$. Seja $<$ uma ordem estrita (ou simplesmente uma ordem) em P e $A \subset P$. Um elemento p é um *limitante superior* de A se $a < p$ ou $a = p$ para todo $a \in A$. p é o *supremo* se p é o menor dos limitantes superiores, com respeito a $<$. Uma ordem linear densa é *completa* se todo subconjunto não vazio que possuir um limitante superior tiver um supremo.

O seguinte teorema será útil na prova do teorema 3.12. Para uma prova do Teorema 3.7 indicamos [HJ99].

Teorema 3.7 *Sejam* $(P, <)$ *e* $(L, \prec)$ *ordens lineares, densas, contáveis, sem extremos. Então* $(P, <)$ *e* $(L, \prec)$ *são isomorfas.*

Agora estamos prontos para provar o teorema 3.12.

Considere a seguinte lista de axiomas:

- $OL(<) := \forall x \forall y (x < y \to \neg y < x) \land \forall x \forall y \forall z (x < y \land y < z \to x < z) \land \forall x \forall y (x < y \lor x = y \lor y < x) \land \forall x \forall y (x < y \to \exists z (x < z \land z < y)) \land \neg \exists x \forall y (x < y \lor x = y)$,

  “$<$ é uma ordem linear (total), densa, que não possui um menor elemento. ”

- $B(R) := \exists x \forall y (R(y) \to (y < x \lor y = x))$,

  “R possui um limitante superior.”

- $LUB(R) := \exists x (\forall y (R(y) \to (y < x \lor y = x)) \land \forall y (\forall z (R(z) \to (z < y \lor z = y)) \to (x < y \lor x = y))$,

  “R possui um supremo. ”

- $D(R) := B(R) \to LUB(R)$,

  “se R possui um limitante superior, então R possui um supremo.” Chamaremos esse axioma de *propriedade de Dedekind.*

---

ordem com semântica minimal, na qual os modelos considerados são minimais com respeito à relação de subestrutura [Peq85, Proposição 3.4], ao passo que os modelos considerados em CPP são minimais com relação a $\leq^{\overline{P};\overline{Z}}$.

- $DC(R) := \forall x(R(x) \to \forall y(y < x \to R(y)))$,

  "se a pertence a R então todo elemento menor do que a também pertence a R."

- $NE(R) := \exists x R(x)$,

  "R é não vazio. "

A seguinte seqüência de lemas culmina com a prova do Teorema 3.12. Seja

$$A(R) := OL(<) \wedge DC(R) \wedge NE(R) \wedge \neg D(R).$$

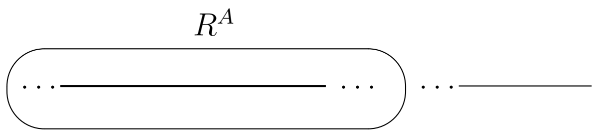
Seja $\phi := Circ[A(R); R]$. Podemos reescrever ϕ como

$$A(R) \wedge \forall R'((DC(R') \wedge NE(R') \wedge R' \subset R) \to D(R')).$$

Intuitivamente ϕ afirma que R é uma ordem densa, não vazia e minimal (na verdade mínima) em não possuir a propriedade de Dedekind. Isso implica que qualquer subconjunto próprio de R possui a propriedade de Dedekind. Isso fará de R uma ordem completa, e, portanto, não será contável.

Lema 3.8 *ϕ é satisfatível.*

Prova. Um possível modelo $\mathfrak{A}$ de ϕ tem a seguinte aparência.



Pense no conjunto dos números reais à esquerda ordenados usualmente (simbolizados pelo conjunto R^A) e alguns elementos extras à direita formando uma cadeia descendente infinita, mas cujos elementos são todos maiores que os elementos de R com respeito à relação de ordem que interpreta $<$ em $\mathfrak{A}$. Isso garante que R^A não possui supremo. Então, R^A não terá a propriedade de Dedekind, no entanto, qualquer subconjunto próprio de R que satisfizer a propriedade DC terá a propriedade de Dedekind. ∎

Seja $\mathfrak{A} = (U, <^A, R^A)$ um modelo de ϕ. Seja $\mathfrak{A}'$ o $\{<\}$-reduto de $\mathfrak{A}$. Seja $\mathfrak{B} = (R^A, <^R)$ a subestrutura de $\mathfrak{A}'$ induzida por R^A.

Lema 3.9 $\mathfrak{B}$ *é um conjunto ordenado denso, infinito e sem extremos.*

Prova. R^A não é vazio uma vez que $\mathfrak{A} \models NE(R)$. Se R fosse finito, o maior elemento de R seria um supremo em $\mathfrak{A}$ o que não é possível pois $\mathfrak{A} \models \neg D(R)$. Pelo mesmo motivo R^A não possui um extremo superior, e como $\mathfrak{A} \models DC(R)$ e $\mathfrak{A}$ não possui menor elemento (veja $OL(<)$), então R^A não possui um extremo inferior. Como $<^A$ é densa e $\mathfrak{A} \models DC(R)$, $<^R$ é densa. ∎

Lema 3.10 $\mathfrak{B}$ *é completa.*

Prova. Seja $C \subset R^A$ que possua um limitante superior em R^A. Seja $C' = \{x \in R^A \mid x < c \text{ ou } x = c \text{ para algum } c \in C\}$. Obviamente todo limitante superior de C também é limitante superior de C' e vice-versa. Observe que C' é um subconjunto próprio de R^A. Então, como $\mathfrak{A} \models \phi$, C' possui um supremo em $\mathfrak{A}$. Mas $\mathfrak{A} \models DC(R)$ e existe um limitante superior de C' em R^A, logo C' possui um supremo em R^A, portanto C possui um supremo em R^A. Logo, $\mathfrak{B}$ é completa. ∎

Lema 3.11 $\mathfrak{B}$*, e portanto também* $\mathfrak{A}$*, é incontável.*

Prova. Suponha $\mathfrak{B}$ contável. Pelo Teorema 3.7, $\mathfrak{B}$ é isomorfo a, por exemplo, $(\mathbf{Q}, <)$, a ordem dos números racionais. Mas $\mathfrak{B}$ é completo e $(\mathbf{Q}, <)$ não. Absurdo! ∎

Teorema 3.12 *Existe uma circunscrição $Circ[A(R); R]$ sem objetos variáveis e num alfabeto $\{R, <\}$, onde R é um predicado unário e $<$ um predicado binário, que não possui modelos enumeráveis.*

Prova. Como vimos, a sentença ϕ possui apenas modelos não enumeráveis. ∎

Continuaremos a análise do poder expressivo de Circunscrição na próxima seção. Investigaremos algumas questões relativas à definibilidade.

3.5 Definibilidade em Circunscrição

Nesta seção, estudaremos a definibilidade de um símbolo predicativo através da Circunscrição. Isto é, estudaremos os casos em que uma circunscrição define o predicado circunscrito.

Em [Doy85] é levantada a questão de quando uma circunscrição define implicitamente o predicado circunscrito. Doyle examina o problema no contexto da Circunscrição de primeira-ordem [McC80]. Podemos transferir o problema para o contexto da Circunscrição de segunda ordem como apresentada aqui. De agora em diante, assumiremos que apenas um predicado é circunscrito em cada circunscrição a fim de simplificar a exposição do problema. Vejamos o seguinte lema a respeito dos casos em que a Circunscrição define o predicado circunscrito.

Lema 3.13 *Seja $\alpha(P, \overline{Z})$ uma $S \cup \{P\} \cup \{\overline{Z}\}$-sentença de primeira-ordem. $Circ[\alpha(P, \overline{Z}); P; \overline{Z}]$ define implicitamente o símbolo predicativo P se, e somente se, para toda S-estrutura $\mathfrak{A}$ existe no máximo uma $S \cup \{P\} \cup \{\overline{Z}\}$-expansão $\mathfrak{A}'$ que é modelo de $\alpha(P, \overline{Z})$ minimal com relação a $\leq^{P;\overline{Z}}$.*

Prova. Direto da definição de definição implícita. ∎

Doyle lança a seguinte questão: quando a circunscrição define implicitamente o predicado circunscrito, será possível substituir o esquema de circunscrição[4] pela definição explícita do predicado circunscrito (que, pelo Teorema de Beth, sempre existe em primeira-ordem) junto com a fórmula circunscrita?

Podemos reformular essa questão para a Circunscrição de segunda ordem: quando $Circ[\alpha(P, \overline{Z}); P; \overline{Z}]$ é equivalente a uma teoria de primeira-ordem e define implicitamente o predicado P, será que $Circ[\alpha(P, \overline{Z}); P; \overline{Z}]$ é equivalente α mais a definição explícita do predicado P?

Em termos precisos, a questão de Doyle é: se $Circ[\alpha(P, \overline{Z}); P; \overline{Z}]$ é equivalente a uma teoria de primeira-ordem e

$$Circ[\alpha(P, \overline{Z}); P; \overline{Z}] \models \forall \overline{x}(P(\overline{x}) \leftrightarrow \phi(\overline{x})),$$

onde P não ocorre em $\phi(\overline{x})$, será que temos

$$Circ[\alpha(P, \overline{Z}); P; \overline{Z}] \quad \equiv \quad \alpha(P, \overline{Z}) \wedge \forall \overline{x}(P(\overline{x}) \leftrightarrow \phi(\overline{x}))? \tag{3.2}$$

Doyle afirma que isso não é verdade. No entanto, o contra-exemplo que ele apresenta não é realmente um contra exemplo. Abaixo, exibiremos um contra-exemplo real, demonstrando a resposta de Doyle.

Exemplo 3.2 *Seja $\phi(<)$ uma fórmula de primeira-ordem afirmando que $<$ é uma ordem linear na qual todo elemento, exceto o menor, possui um predecessor e todo elemento, exceto o maior, possui um sucessor (entretanto, $\phi(<)$ não determina se $<$ possui maior ou menor elemento). Seja c uma constante. Seja $\psi(c)$ uma fórmula de primeira-ordem que afirma que se $<$ possui um menor elemento então c é o menor elemento de $<$. Seja P um*

[4] Doyle trabalha com a Circunscrição de Primeira-Ordem, na qual a circunscrição de um predicado em uma teoria é representada por um esquema de sentenças de primeira-ordem.

predicado unário. Seja $\theta(P)$ a fórmula de primeira-ordem que afirma que P é não vazio, P possui um maior elemento[5] *com relação a $<$ e P é "fechado para baixo" com relação a $<$ (isto é, se $P(a)$ e $b < a$ então $P(b)$). Observe que em todo modelo $(\mathfrak{A}, \mathbf{P})$ de $Circ[\phi(<) \wedge \psi(c) \wedge \theta(P); P]$ a ordem $<$ possui menor elemento (caso contrário seria sempre possível escolher uma interpretação para P menor, pois bastaria escolher um subconjunto próprio de P "fechado para baixo"). Analogamente, se $<$ possui um menor elemento em uma estrutura $\mathfrak{A}$, então existe um modelo $(\mathfrak{A}, \mathbf{P})$ de $Circ[\phi(<) \wedge \psi(c) \wedge \theta(P); P]$. Facilmente se observa que $Circ[\phi(<) \wedge \psi(c) \wedge \theta(P); P]$ é equivalente a*

$$\begin{aligned}\Gamma = \{&\phi(<) \wedge \psi(c) \wedge \theta(P),\\ &\text{"}< \textit{ possui menor elemento}\text{"},\\ &\forall x(P(x) \leftrightarrow \text{"}x \textit{ é o menor elemento de } <\text{"})\}.\end{aligned}$$

E fácil ver que

$$\Gamma \models \forall x(P(x) \leftrightarrow x = c \vee x < c).$$

Entretanto,

$$Circ[\phi(<) \wedge \psi(c) \wedge \theta(P); P] \not\equiv \phi(<) \wedge \psi(c) \wedge \theta(P) \wedge \forall x(P(x) \leftrightarrow x = c \vee x < c).$$

(É fácil ver que existe um modelo de

$$\phi(<) \wedge \psi(c) \wedge \theta(P) \wedge \forall x(P(x) \leftrightarrow x = c \vee x < c)$$

no qual a ordem $<$ não tem menor elemento.)

Mesmo em face dessa resposta negativa, para a classe das sentenças bem-fundadas e em circunscrições sem objetos variantes nós provamos que a equivalência vale. Abaixo definimos a classe das sentenças bem fundadas (veja [Lif94]).

Definição 3.8 *Seja $\alpha(P, \overline{Z})$ uma sentença da lógica de primeira-ordem. Nós dizemos que $\alpha(P, \overline{Z})$ é* bem-fundada com relação a $\leq^{P;\overline{Z}}$ *se, e somente se, para todo modelo $\mathfrak{A}$ de $\alpha(P, \overline{Z})$ existe um modelo $\mathfrak{B}$ de $\alpha(P, \overline{Z})$ $P; \overline{Z}$-minimal e tal que $\mathfrak{A} \leq^{P;\overline{Z}} \mathfrak{B}$.*

Teorema 3.14 *Se $\alpha(P)$ é bem-fundada com relação a $\leq^{P}$, a circunscrição $Circ[\alpha(P); P]$ é equivalente a uma teoria de primeira-ordem e define implicitamente o símbolo predicativo P, então $Circ[\alpha(P); P]$ é equivalente a $\alpha(P)$ mais a definição explícita de P.*

[5] Com essa condição, esse contra-exemplo se aplica também à Circunscrição de primeira-ordem, conforme Doyle pretendia.

Prova. Seja $\alpha(P)$ como no enunciado do teorema. Seja $\phi(\overline{x})$ tal que

$$Circ[\alpha(P); P] \models \forall \overline{x}(P(\overline{x}) \leftrightarrow \phi(\overline{x})).$$

Obviamente todo modelo de $Circ[\alpha; P]$ é modelo de

$$\Gamma := \alpha(P) \wedge \forall \overline{x}(P(\overline{x}) \leftrightarrow \phi(\overline{x}))).$$

Seja $\mathfrak{A}$ modelo de Γ. Como $\mathfrak{A} \models \alpha$ e $\alpha(P)$ é bem-fundada, existe um modelo $\mathfrak{A}' \leq^P \mathfrak{A}$ P-minimal de $\alpha(P)$. Logo $\mathfrak{A}' \models Circ[\alpha(P); P]$. Mas nesse caso $\mathfrak{A}' \models \Gamma$ e, como Γ define P explicitamente, e portanto também implicitamente, temos $P^{\mathfrak{A}'} = P^{\mathfrak{A}}$ e portanto $\mathfrak{A}$ também é minimal com relação a $\leq^P$, logo $\mathfrak{A} \models Circ[\alpha(P); P]$. ∎

Vamos relembrar o Exemplo 3.2. Nós exibimos uma fórmula

$$\alpha(P) := \phi(<) \wedge \psi(c) \wedge \theta(P)$$

tal que $Circ[\alpha(P); P]$ define P implicitamente e tal que

$$Circ[\alpha(P); P] \models \forall x(P(x) \leftrightarrow \phi(x)),$$

onde P não ocorre em $\phi(x)$, mas

$$Circ[\alpha(P); P] \not\equiv \alpha(P) \wedge \forall x(P(x) \leftrightarrow \phi(x)).$$

Agora, chamamos atenção ao seguinte fato. Uma teoria pode admitir várias definições explícitas para um símbolo definido implicitamente. Se ao invés da definição $\phi(x)$ utilizarmos a definição $\phi'(x) := (x = c)$, nós temos

$$Circ[\alpha(P); P] \equiv \alpha(P) \wedge \forall x(P(x) \leftrightarrow \phi'(x)).$$

Fica a pergunta: será que, quando $Circ[\alpha(P, \overline{Z}); P; \overline{Z}]$ define P implicitamente, existe uma definição explícita $\phi(\overline{x})$ tal que a equivalência 3.2 seja satisfeita? Nós vimos no Teorema 3.14 que se $\alpha(P)$ for bem-fundada e a circunscrição não envolver objetos variantes, então qualquer definição explícita $\phi(\overline{x})$ para P satisfaz a equivalência 3.2. Mostraremos abaixo que a resposta a essa pergunta é afirmativa para qualquer fórmula $\alpha(P, \overline{Z})$.

Primeiramente, vamos demonstrar o seguinte lema a respeito de Circunscrição quando esta pode ser escrita em primeira-ordem, isto é, quando uma circunscrição é equivalente a uma teoria de primeira-ordem.

Lema 3.15 *Seja $\alpha(\overline{P}, \overline{Z})$ uma $S \cup \{\overline{P}\} \cup \{\overline{Z}\}$-fórmula de primeira-ordem, $\overline{P} = P_1, \ldots, P_n$ e $\overline{Z} = Z_1, \ldots, Z_m$ tuplas de símbolos relacionais. Se*

$$Mod(Circ[\alpha(\overline{P}, \overline{Z}); \overline{P}; \overline{Z}])$$

é Δ-elementar, então $Mod(Circ[\alpha(\overline{P}, \overline{Z}); \overline{P}; \overline{Z}])$ é elementar.

Prova. Suponha que existe $Circ[\alpha(\overline{P},\overline{Z});\overline{P};\overline{Z}]$ tal que

$$Circ[\alpha(\overline{P},\overline{Z});\overline{P};\overline{Z}] \equiv Th^{\mathrm{FO}}(Circ[\alpha(\overline{P},\overline{Z});\overline{P};\overline{Z}])$$

e $Th^{\mathrm{FO}}(Circ[\alpha(\overline{P},\overline{Z});\overline{P};\overline{Z}])$ não é equivalente a nenhuma fórmula ϕ de primeira-ordem. Sejam $\overline{P'} = P'_1,\ldots,P'_n$ e $\overline{Z'} = Z'_1,\ldots,Z'_m$ tuplas de símbolos relacionais tais que a aridade de P_i é igual à de P'_i e a aridade de Z'_i é igual à de Z_i, $1 \leq i \leq n$. Considere a teoria

$$\Theta := Th^{\mathrm{FO}}(Circ[\alpha;\overline{P};\overline{Z}]) \cup \{\alpha(\overline{P'},\overline{Z'})\}.$$

Obviamente, para qualquer $S \cup \{\overline{P},\overline{Z},\overline{P'},\overline{Z'}\}$-estrutura

$$\mathfrak{C} := (\mathfrak{A},\overline{\mathbf{P}},\overline{\mathbf{Z}},\overline{\mathbf{P'}},\overline{\mathbf{Z'}}),$$

temos que

$$\mathfrak{C} \models \Theta \quad \text{sss} \quad (\mathfrak{A},\overline{\mathbf{P}},\overline{\mathbf{Z}}) \models Circ[\alpha(\overline{P},\overline{Z});\overline{P};\overline{Z}] \text{ e } (\mathfrak{A},\overline{\mathbf{P'}},\overline{\mathbf{Z'}}) \models \alpha(\overline{P'},\overline{Z'}),$$

onde $\overline{\mathbf{P}}$ e $\overline{\mathbf{P'}}$ são tuplas de predicados em A que interpretam $\overline{P}$ e $\overline{P'}$. Considere a teoria $\Theta' := \Theta \cup \{P' \subsetneq P\}$. Obviamente, Θ' é inconsistente, pois, caso contrário, haveria um modelo $\mathfrak{C} = (\mathfrak{A},\overline{\mathbf{P}},\overline{\mathbf{Z}},\overline{\mathbf{P'}},\overline{\mathbf{Z'}})$ de Θ' e nesse caso $(\mathfrak{A},\overline{\mathbf{P'}},\overline{\mathbf{Z'}}) \lneq^{\overline{P};\overline{Z}} (\mathfrak{A},\overline{\mathbf{P}},\overline{\mathbf{Z}})$, contrariando $(\mathfrak{A},\overline{\mathbf{P}},\overline{\mathbf{Z}}) \models Circ[\alpha(\overline{P},\overline{Z});\overline{P};\overline{Z}]$. Por Compacidade da lógica de primeira-ordem, seja Θ_0 um subconjunto finito de Θ tal que $\Theta_0 \cup \{P' \subsetneq P\}$ é inconsistente. Seja

$$C^{\alpha}_{\overline{P};\overline{Z}} := \bigwedge\{\Theta_0 \cap Th^{\mathrm{FO}}(Circ[\alpha(\overline{P},\overline{Z});\overline{P};\overline{Z}])\}$$

uma fórmula de primeira-ordem. Por definição de $C^{\alpha}_{\overline{P};\overline{Z}}$, todo modelo $\overline{P};\overline{Z}$-minimal de $\alpha(\overline{P},\overline{Z})$ é modelo de $\alpha(\overline{P},\overline{Z}) \wedge C^{\alpha}_{\overline{P};\overline{Z}}$. Seja $(\mathfrak{A},\overline{\mathbf{P}},\overline{\mathbf{Z}})$ um modelo de $\alpha(\overline{P},\overline{Z}) \wedge C^{\alpha}_{\overline{P};\overline{Z}}$. Suponha que $(\mathfrak{A},\overline{\mathbf{P}},\overline{\mathbf{Z}})$ não é modelo $\overline{P};\overline{Z}$-minimal de $\alpha(\overline{P},\overline{Z})$. Então existe um modelo $(\mathfrak{A},\overline{\mathbf{P'}},\overline{\mathbf{Z'}})$ de $\alpha(\overline{P},\overline{Z})$ tal que

$$(\mathfrak{A},\overline{\mathbf{P'}},\overline{\mathbf{Z'}}) \lneq^{\overline{P};\overline{Z}} (\mathfrak{A},\overline{\mathbf{P}},\overline{\mathbf{Z}}).$$

Mas nesse caso $\mathfrak{C} := (\mathfrak{A},\overline{\mathbf{P}},\overline{\mathbf{Z}},\overline{\mathbf{P'}},\overline{\mathbf{Z'}})$ é modelo de $\Theta_0 \cup \{P' \subsetneq P\}$. Absurdo. Portanto, $(\mathfrak{A},\overline{\mathbf{P}},\overline{\mathbf{Z}})$ é modelo $\overline{P};\overline{Z}$-minimal de $\alpha(\overline{P},\overline{Z})$. Isso significa que

$$Circ[\alpha(\overline{P},\overline{Z});\overline{P};\overline{Z}] \equiv \alpha(\overline{P},\overline{Z}) \wedge C^{\alpha}_{\overline{P};\overline{Z}}.$$

Logo, $Mod(Circ[\alpha(\overline{P},\overline{Z});\overline{P};\overline{Z}])$ é elementar. ∎

Agora, mostraremos o principal teorema desta seção.

Teorema 3.16 *Seja $\alpha(P,\overline{Z})$ uma $S \cup \{P,\overline{Z}\}$-sentença em lógica de primeira-ordem tal que $Circ[\alpha(P,\overline{Z})]$ é equivalente a uma teoria de primeira-ordem Γ e P é definido implicitamente por $Circ[\alpha(P,\overline{Z})]$. Então existe uma definição explícita $\phi(\overline{x})$ de P—onde, obviamente, P não ocorre—tal que*

$$Circ[\alpha(P,\overline{Z})] \equiv \alpha(P,\overline{Z}) \wedge \forall\overline{x}(P(\overline{x}) \leftrightarrow \phi(\overline{x})).$$

Prova. Pelo Teorema 3.15, existe uma $S \cup \{P,\overline{Z}\}$-sentença de primeira-ordem $\gamma(P,\overline{Z})$ tal que $Circ[\alpha(P,\overline{Z})] \equiv \gamma(P,\overline{Z})$. Como $Circ[\alpha(P,\overline{Z})]$ define implicitamente P, pelo Teorema de Beth existe uma definição explícita $\psi(\overline{x})$ de P tal que

$$\gamma(P,\overline{Z}) \quad \models \quad \forall\overline{x}(P(\overline{x}) \leftrightarrow \psi(\overline{x})). \tag{3.3}$$

Seja $\gamma' := \gamma(\psi,\overline{Z})$ uma $S \cup \{\overline{Z}\}$-sentença de primeira-ordem, onde $\gamma(\psi,\overline{Z})$ é obtido a partir de $\gamma(P,\overline{Z})$ substituindo átomos da forma $P(\overline{t})$ por $\psi(\overline{t})$. Se $\mathfrak{A}$ é um $S \cup \{\overline{Z}\}$-modelo de γ', então existe um predicado $\mathbf{P}$ tal que $(\mathfrak{A},\mathbf{P})$ é modelo de $\gamma(P,\overline{Z})$—basta fazer $\mathbf{P} := \{\overline{a} \in A | \mathfrak{A} \models \psi(\overline{x})[\overline{a}]\}$. Por outro lado, se $(\mathfrak{A},\mathbf{P}) \models \gamma(P,\overline{Z})$, então $\mathfrak{A} \models \gamma'$—por 3.3. Logo

$$(\mathfrak{A},\mathbf{P}) \models \gamma(P,\overline{Z}) \quad \text{sss} \quad \mathfrak{A} \models \gamma' \text{ e } \mathbf{P} := \{\overline{a} \in A | \mathfrak{A} \models \psi(\overline{x})[\overline{a}]\}. \tag{3.4}$$

Seja $\phi(\overline{x}) := \gamma' \wedge \psi(\overline{x})$. Como $\gamma(P,\overline{Z}) \models \gamma'$—veja 3.4—então

$$\gamma(P,\overline{Z}) \models \alpha(P,\overline{Z}) \wedge \forall\overline{x}(P(\overline{x}) \leftrightarrow \phi(\overline{x})).$$

Seja $(\mathfrak{A},\mathbf{P})$ um modelo de $\alpha(P,\overline{Z}) \wedge \forall\overline{x}(P(\overline{x}) \leftrightarrow \phi(\overline{x}))$. Se $\mathbf{P} = \emptyset$, então $(\mathfrak{A},\mathbf{P})$ é modelo $P;\overline{Z}$-minimal de $\alpha(P,\overline{Z})$. Se $\mathbf{P} \neq \emptyset$, então existe $\overline{a} \in A^n$ tal que $(\mathfrak{A},\mathbf{P}) \models P(\overline{x})[\overline{a}]$. Mas como $(\mathfrak{A},\mathbf{P}) \models \forall\overline{x}(P(\overline{x}) \leftrightarrow \phi(\overline{x}))$, então existe $\overline{a} \in A^n$ tal que $(\mathfrak{A},\mathbf{P}) \models \gamma' \wedge \psi(\overline{x})$. Logo $(\mathfrak{A},\mathbf{P}) \models \gamma'$ e, como P não ocorre em γ', $\mathfrak{A} \models \gamma'$. Utilizando novamente o fato de que $(\mathfrak{A},\mathbf{P}) \models \forall\overline{x}(P(\overline{x}) \leftrightarrow \phi(\overline{x}))$, temos $\mathbf{P} := \{\overline{a} \in A | \mathfrak{A} \models \psi(\overline{x})[\overline{a}]\}$. Mas como $\mathfrak{A} \models \gamma'$, então $\mathbf{P} := \{\overline{a} \in A | \mathfrak{A} \models \phi(\overline{x})[\overline{a}]\} = \{\overline{a} \in A | \mathfrak{A} \models \psi(\overline{x})[\overline{a}]\}$. A partir de 3.4 temos que $(\mathfrak{A},\mathbf{P})$ é modelo de $\gamma(P,\overline{Z})$ e portanto modelo $P;\overline{Z}$-minimal de $\alpha(P,\overline{Z})$. Portanto

$$Circ[\alpha(P,\overline{Z})] \equiv \alpha(P,\overline{Z}) \wedge \forall\overline{x}(P(\overline{x}) \leftrightarrow \phi(\overline{x})).$$

∎

Na próxima seção, investigaremos o poder expressivo das NATs de Lifschitz, que são generalizações da Circunscrição.

3.6 Nested Abnormality Theories—NATs

Em [Lif95], Lifschitz introduz as *Nested Abnormality Theories* (NATs) como um sistema adequado para representação do conhecimento. Em uma NAT é possível circunscrever blocos de axiomas que podem ser agrupados em novos blocos e circunscritos novamente. Lifschitz afirma que as NATs são formas mais intuitivas de exprimir informações e que em muitos casos parecem oferecer resultados mais satisfatórios que a Circunscrição ou *Prioritized Circumscription.*

Definiremos abaixo as NATs.

Definição 3.9 (NATs) *Dado um conjunto de símbolos S, o conjunto Blocos de blocos é definido indutivamente como o menor conjunto tal que, para qualquer k, se $B_1, \ldots, B_n$ são blocos ou $S \cup \{Ab\}$-fórmulas da lógica de primeira-ordem, então*

$$\{\overline{C}; B_1, \ldots, B_n\}$$

é um bloco, onde Ab é um símbolo relacional de aridade k e $\overline{C} := C_1, \ldots, C_m$ é uma tupla de símbolos relacionais.

Uma NAT T é um conjunto de blocos.

Conforme observa Lifschitz em [Lif95], se B_i e B_j são fórmulas no bloco $\{\overline{C}; B_1, \ldots, B_n\}$ e Ab ocorre em ambas, então Ab ocorre com a mesma aridade.

A semântica das NATs é definida em termos da lógica de segunda-ordem. Nós definimos o operador σ que converte NATs em teorias de segunda-ordem. Os modelos dessas NATs são portanto os modelos dessas teorias de segunda-ordem.

Definição 3.10 (A Função σ) *Seja T uma NAT. Então*

$$\sigma(T) := \{\sigma(B) | B \in T\}.$$

Definimos a função σ sobre blocos $B = \{\overline{C}; B_1, \ldots, B_n\}$ como

$$\sigma(B) := \exists Ab(Circ[\sigma(B_1) \wedge \ldots \wedge \sigma(B_n); Ab; \overline{C}]),$$

e definimos σ sobre fórmulas da lógica de primeira-ordem como seu fecho universal.

Nós escrevemos $\{\overline{C}; P; B_1, \ldots, B_n\}$ como uma abreviação[6] para

$$\{\overline{C}, P; B_1, \ldots, B_n, P \subseteq Ab\}.$$

Nós temos o seguinte.

[6] Em [Lif95], é usada a notação $\{\overline{C}, minP; B_1, \ldots, B_n\}$. Nós adotamos um notação diferente para evitar confusão com o operador MIN da lógica MIN (veja Capítulo 4).

Lema 3.17 *Seja* $B := \{\overline{C}; P; B_1, \ldots, B_n\}$ *um bloco. Se Ab não ocorre livre em* $\sigma(B_i)$*, para* $1 \leq i \leq n$*, então* $\sigma(B)$ *é equivalente a*

$$Circ[\sigma(B_1) \wedge \ldots \wedge \sigma(B_n); P; \overline{C}].$$

Prova. A prova é análoga à apresenta em [Lif95], onde este lema é provado para o caso em que cada B_i é uma sentença de primeira-ordem. ∎

Em [CEG05], Cadoli et al. investigaram a complexidade computacional da avaliação das relações de conseqüência lógica e satisfação em NATs proposicionais. Cadoli et al. estabeleceram limites superiores e inferiores para a complexidade desses problemas traduzindo NATs proposicionais para Fórmulas Booleanas Quantificadas (QBFs) e estabelecendo a validade de QBFs em termos de conseqüência lógica a partir de NATs.

Aqui, estamos interessados no poder expressivo de NATs de primeira-ordem. Nós provaremos que, para cada sentença ϕ da lógica de segunda-ordem em um alfabeto S, existe uma NAT em um alfabeto estendido $S \cup S'$ que possui exatamente os mesmos modelos de $\phi \wedge p_i \wedge \ldots \wedge p_n$, onde $p_1, \ldots, p_n$ são novos símbolos proposicionais. Isto é, para cada S-sentença de segunda-ordem existe uma NAT em uma linguagem estendida que prova exatamente as mesmas S-sentenças que ϕ prova.

Nossa tradução da lógica de segunda-ordem em NATs será baseada no Lema 3.5. A partir do Lema 3.5 podemos provar um caso particular do nosso resultado. Segue-se diretamente do Lema 3.5 que, para toda S-sentença da lógica de segunda-ordem em Π^1_1, existe uma NAT sem blocos aninhados que possui os mesmos modelos de $\phi \wedge p_1$, onde p_1 é um símbolo proposicional que não ocorre em ϕ.

Corolário 3.18 *Se* $\phi = \forall \overline{X} \psi$ *é uma* S*-sentença em* Π^1_1 *onde* ψ *é de primeira-ordem, então* $T := \{\{\overline{X}; p_1; \neg\psi \vee p_1\}, p_1\}$ *é uma NAT no alfabeto* $S \cup \{\overline{X}\} \cup \{p_1\}$ *equivalente a* $\phi \wedge p_1$*.*

Prova. Pela Definição 3.10, $\sigma(T) = \sigma(\{\overline{X}; p_1; \neg\psi \vee p_1\}) \wedge \sigma(p_1)$. Pelo Lema 3.17, $\sigma(\{\overline{X}; p_1; \neg\psi \vee p_1\}) \wedge \sigma(p_1) = Circ[\neg\psi \vee p_1; p_1; \overline{X}] \wedge p_1 \equiv$, (pelo Lema 3.5), $\phi \wedge p_1$. ∎

Vejamos o seguinte exemplo.

Exemplo 3.3 *Seja* $S := \{P, s, 0\}$*, onde* P *é um símbolo relacional unário,* s *é um símbolo funcional unário e* 0 *é um símbolo de constante. Seja*

$$\psi(X) = (X(0) \wedge \forall x(X(x) \rightarrow X(s(x)))) \rightarrow \forall y(P(y) \rightarrow X(y)).$$

Essa fórmula afirma que, se 0 *pertence a* X *e* X *é fechado sobre* s*, então* X *contém* P*. Seja* $\phi := \forall X \psi(X)$*.* ϕ *afirma que* P *é o menor conjunto ao qual* 0 *pertence e é fechado para* s*. Seja* $T = \{\{X; p_1; \neg\psi(X) \vee p_1\}, p_1\}$ *uma NAT. Seguindo a prova do Corolário 3.18 temos*

$$\sigma(T) = Circ[\neg\psi(X) \vee p_1; p_1; X] \wedge p_1$$

$$=$$

$$(\neg\psi(X) \vee p_1) \wedge \forall p' \forall X' ((\neg\psi(X') \vee p'_1) \rightarrow \neg(p' < p_1)).$$

Observe que $\sigma(T)$ *é uma fórmula onde* X *ocorre livre. Nós podemos chamá-la de uma* $S \cup \{X\}$*-fórmula.*

Agora, provaremos o caso geral. Nosso método funcionará da seguinte forma. Seja $\psi := \forall \overline{X} \phi$ uma fórmula da lógica de segunda-ordem em forma normal prenex. Para representarmos esta fórmula utilizando o operador $Circ$, nós aplicamos o Lema 3.5 e obtemos

$$\forall \overline{X} \phi \wedge p_1 \wedge \ldots \wedge p_n \equiv Circ[(\neg\phi(\overline{X}) \vee p_1) \wedge p_2 \wedge \ldots \wedge p_n; p_1; \overline{X}] \wedge p_1.$$

Isto é, precisamos circunscrever $(\neg\phi(\overline{X}) \vee p_1) \wedge p_2 \wedge \ldots \wedge p_n$. Suponha que $\phi = \exists \overline{Y} \theta$. Nesse caso, a fórmula que precisamos circunscrever é equivalente a $\forall \overline{Y} (\neg\theta \vee p_1) \wedge p_2 \wedge \ldots \wedge p_n$. Ou seja, nós temos outra fórmula que começa com quantificador universal para a qual podemos aplicar novamente o Lema 3.5 e escrevê-la em termos do operador $Circ$. Como o aninhamento de blocos em uma NAT corresponde ao aninhamento de operadores $Circ$ (veja Definição 3.10), nós podemos representar uma sentença de segunda-ordem através de uma NAT. Observe que, neste processo, precisamos lidar com símbolos proposicionais adicionais e alguns símbolos relacionais ficam livres, embora o valor desses símbolos relacionais em um modelo não importe para estabelecer se esse modelo satisfaz ou não a NAT resultante. Portanto, precisamos estender a linguagem a fim de incluirmos esses símbolos relacionais e proposicionais.

Primeiramente, iremos introduzir a função ρ definida sobre certas fórmulas de segunda ordem. Essa função será utilizada para construir uma fórmula de primeira-ordem que será utilizada para formar a NAT desejada.

Definição 3.11 (A Função ρ) *Seja* $\overline{X}_i := X_{i1}, \ldots, X_{in_i}$ *uma lista de variáveis relacionais para algum* n_i*,* $Q_1\overline{X}_1 \ldots Q_n\overline{X}_n\psi$ *uma fórmula de segunda-ordem possivelmente contendo símbolos proposicionais onde*

$$Q_j\overline{X}_j := Q_jX_{j1} \ldots Q_{n_j}X_{n_j}$$

e ψ pode conter quantificadores de segunda-ordem, e sejam $p_1, \ldots, p_j$ símbolos proposicionais que não ocorrem em ψ. Nós definimos

$$\rho(Q_1\overline{X}_1 \ldots Q_m\overline{X}_m\psi \wedge p_1 \wedge \ldots \wedge p_j) = \widetilde{Q_2}\overline{X}_2 \ldots \widetilde{Q_m}\overline{X}_m(\neg\psi \vee p_1) \wedge p_2 \wedge \ldots \wedge p_j,$$

onde $\widetilde{Q} = \forall$ se $Q = \exists$ e $\widetilde{Q} = \exists$ se $Q = \forall$.

Provaremos alguns lemas sobre ρ.

Lema 3.19 $\rho(Q_1\overline{X}_1 \ldots Q_m\overline{X}_m\psi \wedge p_1 \wedge \ldots \wedge p_j) \equiv (\neg(Q_2\overline{X}_2 \ldots Q_m\overline{X}_m\psi) \vee p_1) \wedge p_2 \wedge \ldots \wedge p_j$.

Prova.

$$\rho(Q_1\overline{X}_1 \ldots Q_m\overline{X}_m\psi \wedge p_1 \wedge \ldots \wedge p_j) = \widetilde{Q_2}\overline{X}_2 \ldots \widetilde{Q_m}\overline{X}_m(\neg\psi \vee p_1) \wedge p_2 \wedge \ldots \wedge p_j$$

$$\equiv \text{, já que } p_1 \text{ não é ligado pelos quantificadores } \widetilde{Q_2} \ldots \widetilde{Q_m},$$
$$((\widetilde{Q_2}\overline{X}_2 \ldots \widetilde{Q_m}\overline{X}_m\neg\psi) \vee p_1) \wedge p_2 \wedge \ldots \wedge p_j$$
$$\equiv$$
$$(\neg(Q_2\overline{X}_2 \ldots Q_m\overline{X}_m\psi) \vee p_1) \wedge p_2 \wedge \ldots \wedge p_j.$$

∎

O próximo lema mostra o resultado da aplicação da função ρ várias vezes a uma fórmula.

Lema 3.20 *Seja $\phi = Q_1\overline{X}_1 \ldots Q_m\overline{X}_m\psi \wedge p_1 \wedge \ldots \wedge p_m$ onde $Q_1 = \forall$, ψ possivelmente contém quantificadores de segunda-ordem e $p_1, \ldots, p_m$ são símbolos proposicionais que não ocorrem em ϕ. Para $l \leq m$, nós temos*

$$\rho^l(\phi) = Q_{l+1}\overline{X}_{l+1} \ldots Q_m\overline{X}_m\psi' \wedge p_{l+1} \wedge \ldots \wedge p_m,$$

se l é par, e

$$\rho^l(\phi) = \widetilde{Q}_{l+1}\overline{X}_{l+1} \ldots \widetilde{Q}_m\overline{X}_m\psi' \wedge p_{l+1} \wedge \ldots \wedge p_m,$$

se l é ímpar. Considere $\rho^0(\phi) = \phi$.

Prova. Para $l = 0$ o lema vale. Suponha que o lema valha para um l qualquer. Observe que $\rho^{l+1}(\phi) = \rho(\rho^l(\phi))$. Se l é par, então $l+1$ é ímpar e, pela Hipótese Indutiva,

$$\rho(\rho^l(\phi)) = \rho(Q_{l+1}\overline{X}_{l+1} \ldots Q_m\overline{X}_m\psi'' \wedge p_{l+1} \wedge \ldots \wedge p_m)$$

$$=$$

$$\widetilde{Q_{l+2}}\overline{X}_{l+2}\ldots\widetilde{Q_m}\overline{X}_m(\neg\psi''\vee p_{l+1})\wedge p_{l+2}\wedge\ldots\wedge p_m.$$

Se l é ímpar, então $l+1$ é par e, pela Hipótese Indutiva,

$$\rho(\rho^l(\phi))=\rho(\widetilde{Q_{l+1}}\overline{X}_{l+1}\ldots\widetilde{Q_m}\overline{X}_m\psi''\wedge p_{l+1}\wedge\ldots\wedge p_m)$$

$$=$$

$$\widetilde{\widetilde{Q_{l+2}}}\overline{X}_{l+2}\ldots\widetilde{\widetilde{Q_m}}\overline{X}_m(\neg\psi''\vee p_{l+1})\wedge p_{l+2}\wedge\ldots\wedge p_m$$

$$=$$

$$Q_{l+2}\overline{X}_{l+2}\ldots Q_m\overline{X}_m(\neg\psi''\vee p_{l+1})\wedge p_{l+2}\wedge\ldots\wedge p_m.$$

∎

Segue imediatamente do Lema 3.20, após aplicarmos ρ em uma fórmula ϕ da forma exigida pela Definição 3.11, a fórmula resultante $\rho(\phi)$ é também uma fórmula neste formato e cujo primeiro bloco de quantificadores é do mesmo tipo do primeiro bloco de quantificadores de ϕ, isto é, universal ou existencial. Assim nós temos o seguinte corolário.

Corolário 3.21 *Seja $\phi = Q_1\overline{X}_1\ldots Q_m\overline{X}_m\psi$ onde Q_{i+1} é o dual de Q_i e ψ pode conter quantificadores de segunda-ordem. Se*

$$\rho^l(\phi)=Q'_{l+1}\overline{X}_{l+1}\ldots Q'_m\overline{X}_m\psi'\wedge p_{l+1}\wedge\ldots\wedge p_m,$$

para $l\leq m$, então $Q'_{l+1}=Q_1$.

Nós definiremos agora uma notação para simplificar a prova dos próximos lemas e teoremas. Dadas uma tupla de tuplas de variáveis relacionais $\overline{X}_1,\ldots,\overline{X}_m$ e uma tupla de símbolos proposicionais $p_1,\ldots,p_m$, nós definimos as fórmulas C_j de segunda-ordem, $0<j\leq m$, como

$$C_j(\phi)=Circ[\phi;p_j;\overline{X}_j]\wedge p_j.$$

O próximo lema é o lema principal dessa seção. O Lema 3.22 relaciona a aplicação da função ρ com a aplicação do operador $Circ$ utilizando as fórmulas C_j. Enquanto a função ρ remove quantificadores de segunda-ordem, o operador $Circ$ os introduz novamente. Isso explica como o aninhamento de blocos em uma NAT simula a aplicação e a alternância de quantificadores de segunda-ordem.

Lema 3.22 *Para quaisquer números naturais* m *e* l *tais que* $m \leq l \leq 1$ *e toda fórmula de segunda-ordem* $\phi = Q_1\overline{X}_1 \ldots Q_l\overline{X}_l\psi \wedge p_1 \wedge \ldots \wedge p_m$*, onde* $Q_1 = \forall$*,* Q_{i+1} *é o dual de* Q_i *e* ψ *possivelmente com quantificadores de segunda-ordem,* $C_1(\ldots C_l(\rho^l(\phi))\ldots)$ *é uma* $S \cup \{\overline{X}_1, \ldots, \overline{X}_l\}$*-fórmula equivalente a* ϕ*.*

Prova. Por indução em l. Para $l = 1$ nós temos $\phi = \forall\overline{X}_1\psi \wedge p_1 \wedge \ldots \wedge p_m$ e, pela Definição 3.11,

$$C_1(\rho(\phi)) = C_1((\neg\psi \vee p_1) \wedge p_2 \wedge \ldots \wedge p_m).$$

Pela definição de C_i,

$$C_1((\neg\psi \vee p_1) \wedge p_2 \wedge \ldots \wedge p_m) = Circ[(\neg\psi \vee p_1) \wedge p_2 \wedge \ldots \wedge p_m; p_1; \overline{X}_1] \wedge p_1.$$

Pelo Lema 3.5, $Circ[(\neg\psi \vee p_1) \wedge p_2 \wedge \ldots \wedge p_m; p_1; \overline{X}_1] \wedge p_1 \equiv \forall\overline{X}_1\psi \wedge p_1 \wedge \ldots \wedge p_m = \phi$. Suponha que o lema valha para um l qualquer. Nós precisamos provar o lema para

$$C_1(\ldots C_{l+1}(\rho^{l+1}(\phi))) = C_1(\ldots C_{l+1}(\rho(\rho^l(\phi)))).$$

Pelo Lema 3.20 e pelo Corolário 3.21, $\rho^l(\phi) = Q'_{l+1}\overline{X}_{l+1}\psi' \wedge p_{l+1} \wedge \ldots \wedge p_m$ para algum ψ' onde $Q'_{l+1} = \forall$. Logo

$$C_1(\ldots C_{l+1}(\rho(\rho^l(\phi)))) = C_1(\ldots C_l(C_{l+1}(\rho(Q'_{l+1}\overline{X}_{l+1}\psi' \wedge p_{l+1} \wedge \ldots \wedge p_m)))).$$

Pelo Lema 3.19,

$$\begin{gathered} \rho(Q'_{l+1}\overline{X}_{l+1}\psi' \wedge p_{l+1} \wedge \ldots \wedge p_m) \\ \equiv \\ (\neg\psi' \vee p_{l+1}) \wedge p_{l+2} \wedge \ldots \wedge p_m, \end{gathered}$$

portanto

$$\begin{gathered} C_1(\ldots C_l(C_{l+1}(\rho(Q'_{l+1}\overline{X}_{l+1}\psi' \wedge p_{l+1} \wedge \ldots \wedge p_m)))\ldots) \\ \equiv \\ C_1(\ldots C_l(C_{l+1}((\neg\psi' \vee p_{l+1}) \wedge p_{l+2} \wedge \ldots \wedge p_m))). \end{gathered}$$

Pelo Lema 3.5,

$$\begin{gathered} C_{l+1}((\neg\psi' \vee p_{l+1}) \wedge p_{l+2} \wedge \ldots \wedge p_m) \\ \equiv \\ \forall\overline{X}_{l+1}\psi' \wedge p_{l+1} \wedge p_{l+2} \wedge \ldots \wedge p_m \\ = \\ \rho^l(\phi). \end{gathered}$$

Assim

$$C_1(\dots C_l(C_{l+1}((\neg\psi' \vee p_{l+1}) \wedge p_{l+2} \wedge \dots \wedge p_m)))$$
$$\equiv$$
$$C_1(\dots C_l(\rho^l(\phi))),$$

que, pela Hipótese Indutiva, é equivalente a ϕ. ∎

Nós exibiremos agora a tradução δ de fórmulas de segunda-ordem em NATs.

Definição 3.12 (A Tradução $\boldsymbol{\delta}$) *Seja $\phi = Q_1\overline{X}_1 \dots Q_m\overline{X}_m\psi$ uma S-sentença de segunda-ordem em forma normal prenex onde ψ é de primeira-ordem, $Q_1 = \forall$ e Q_{i+1} é o dual de Q_i. Nós definimos os blocos B_i^ϕ, $m \geq i \geq 0$, como*

$$B_m^\phi = \{\overline{X}_m; p_m; \rho^m(\phi \wedge p_1 \wedge \dots \wedge p_m)\}$$

e

$$B_{j-1}^\phi = \{\overline{X}_{j-1}; p_{j-1}; B_j^\phi, p_j\},$$

onde $p_1, \dots, p_m$ são símbolos proposicionais que não ocorrem em ϕ. A função de tradução que mapeia ϕ em uma $S \cup \{\overline{X}_1, \dots, \overline{X}_m\} \cup \{p_1, \dots, p_m\}$-NAT é definida como

$$\delta(\phi) = \{B_1^\phi, p_1\}.$$

O seguinte lema mostra o principal resultado desta seção: para toda sentença ϕ de segunda-ordem existe uma NAT que é uma extensão conservativa de ϕ.

Teorema 3.23 $\sigma(\delta(\phi)) \equiv \phi \wedge p_1 \wedge \dots \wedge p_m$.

Prova. Nós mostraremos que

$$\sigma(B_i^\phi) \wedge p_i = C_i(C_{i+1}(\dots C_m(\rho^m(\phi \wedge p_1 \wedge \dots \wedge p_m)))),$$

para cada $m \geq i > 0$. Para $i = m$,

$$\sigma(B_m^\phi) \wedge p_m$$
$$\equiv \text{ (Pelo Lema 3.17)}$$
$$Circ[\rho^m(\phi \wedge p_1 \wedge \dots \wedge p_m); p_m; \overline{X}_m] \wedge p_m$$
$$=$$
$$C_m(\rho^m(\phi \wedge p_1 \wedge \dots \wedge p_m)).$$

Suponha que o lema valha para um i arbitrário. Nós temos $\sigma(B^{\phi}_{i-1}) \wedge p_{i-1} = \sigma(\{\overline{X}_{i-1}; p_{i-1}; B^{\phi}_{i}, p_i\}) \wedge p_{i-1}$ e, pela Definição 3.10,

$$\sigma(\{\overline{X}_{i-1}; p_{i-1}; B^{\phi}_{i}, p_i\}) \wedge p_{i-1} \equiv Circ[\sigma(B^{\phi}_{i}) \wedge p_i; p_{i-1}; \overline{X}_{i-1}] \wedge p_{i-1}$$

$$\equiv \text{ (pela Hipótese Indutiva)}$$

$$Circ[C_i(\ldots C_m(\rho^m(\phi \wedge p_1 \wedge \ldots \wedge p_m))); p_{i-1}; \overline{X}_{i-1}] \wedge p_{i-1}$$

$$=$$

$$C_{i-1}(C_i(\ldots C_m(\rho^m(\phi \wedge p_1 \wedge \ldots \wedge p_m)))).$$

É fácil ver que $\sigma(\delta(\phi)) \equiv C_1(\ldots C_m(\rho^m(\phi \wedge p_1 \wedge \ldots \wedge p_m)))$ que, pelo Lema 4.10, é equivalente a $\phi \wedge p_1 \wedge \ldots \wedge p_m$. ∎

O Teorema 3.23 mostra que para toda S-sentença ϕ de segunda-ordem existe uma NAT $T = \sigma(\delta(\phi))$ que é uma extensão conservativa de ϕ, pois $T \equiv \phi \wedge p_1 \wedge \ldots \wedge p_m$ e $p_1 \wedge \ldots \wedge p_m$ não ocorre em ϕ. Logo $Mod(\phi) = \{\mathfrak{A}|_S | \mathfrak{A} \in Mod(T)\}$, isto é, a classe formada pelos redutos dos modelos de T ao alfabeto S é igual à classe dos modelos de ϕ. Portanto, ϕ e T implicam logicamente as mesmas S-sentenças.

Capítulo 4

A Lógica MIN

Neste capítulo apresentaremos a lógica MIN. MIN extende a lógica MIN(FO) apresentada em [vB05] e que definiremos na Seção 4.1. Na Seção 4.2, definiremos as lógicaS U-MIN e I-MIN e mostraremos que elas são equivalentes em poder expressivo. Daí em diante chamaremos U-MIN de MIN. Na Seção 4.3, definiremos então a Lógica Si-MIN de minimização simultânea e mostraremos que essa extensão não aumenta o poder expressivo de MIN, e com isso mostraremos que a lógica MIN possui o mesmo poder expressivo da lógica de segunda ordem. Mostraremos, na Seção 4.4, que a lógica MIN está relacionada com uma extensão da lógica de primeira-ordem através de quantificadores minimais. Finalizaremos este capítulo na Seção 4.5 exibindo um fragmento de MIN que se situa entre a Lógica de Menor Ponto Fixo e a lógica MIN em poder expressivo.

4.1 A Lógica MIN(FO)

Em [vB05], van Benthem estuda algumas fórmulas da lógica de primeira-ordem que possuem a *Propriedade da Interseção.* Definiremos abaixo essa propriedade.

Definição 4.1 (Propriedade da Interseção) *Se ϕ é uma $S \cup \{P\}$-fórmula de primeira-ordem então ϕ possui a Propriedade da Interseção sss para qualquer classe de modelos $\{(\mathfrak{I}, \mathbf{P}_i) | i \in I\}$ de ϕ, onde $\mathfrak{I}$ é uma S-interpretação, I é um conjunto de índices e cada $\mathbf{P}_i$ é um predicado em A, $(\mathfrak{I}, \bigcap_{i \in I} \mathbf{P}_i)$ é modelo de ϕ. (Quando $\{\mathbf{P}_i | i \in I\}$ é vazio, $\bigcap_{i \in I} \mathbf{P}_i$ é interpretado como A, o universo de $\mathfrak{I}$.*

O principal resultado de [vB05] é demonstrar que a classe das fórmulas de primeira-ordem que possuem a Propriedade da Interseção é exatamente a classe das fórmulas equivalentes a alguma *condição-PIA*. As condições PIA (de *Positive Implies Atom*) são fórmulas compostas por uma implicação quantificada universalmente, onde o antecedente da implicação é uma fórmula positiva em um determinado símbolo predicativo P e o consequente é o átomo $P(\overline{x})$. Definiremos as condições PIA abaixo.

Definição 4.2 (Condições PIA) *Uma $S \cup \{P\}$-fórmula de primeira-ordem é uma condição PIA sss for da forma*

$$\forall \overline{x}(\phi(P, \overline{Q}, \overline{x}) \rightarrow P(\overline{x})),$$

onde $\overline{x}$ é uma tupla de variáveis cujo tamanho é igual à aridade de P, $\overline{Q}$ é uma tupla de símbolos predicativos (variáveis relacionais ou símbolos relacionais em S) e $\phi(P)$ é uma fórmula positiva em P.

van Benthem estabelece o seguinte teorema.

Teorema 4.1 (van Benthem [vB05]) *Seja ψ uma fórmula de primeira-ordem. ψ possui a Propriedade da Interseção sss ψ é equivalente a uma condição PIA.*

A Propriedade da Interseção confere a uma $S \cup \{P\}$-fórmula $\phi(P)$ que a possua a propriedade de, dada uma S-interpretação $\mathfrak{I}$, existir exatamente um modelo $(\mathfrak{I}, \mathbf{P})$ P-minimal de $\phi(P)$. van Benthem então sugere adicionarmos à linguagem da lógica de primeira-ordem um operador para a construção de novos predicados levando-se em consideração o predicado minimal (no caso mínimo) que satisfaz uma dada condição PIA. van Benthem define a lógica MIN(FO) da seguinte forma.

Definição 4.3 (A Lógica MIN(FO)) *A linguagem da lógica MIN(FO) e o conjunto das* condições PIA estendidas *é definido simultaneamente por indução como sendo os menores conjuntos tais que:*

1. *toda S-fórmula de primeira-ordem é uma S-fórmula de MIN(FO),*

2. *toda condição PIA é uma condição PIA estendida,*

3. *se ϕ e ψ são S-fórmulas de MIN(FO), então $(\phi \wedge \psi)$, $(\phi \vee \psi)$, $(\phi \rightarrow \psi)$, $(\phi \leftrightarrow \psi)$, $\neg\phi$, $\forall x(\phi)$ e $\exists x(\phi)$ são fórmulas de MIN(FO),*

4. *se a $S \cup \{P\}$-fórmula $\phi(P, \overline{Q})$ é uma condição PIA estendida, então $[MINP \bullet \phi(P, \overline{Q})](\overline{t})$ é uma S-fórmula de MIN(FO), onde $\overline{t}$ é uma tupla de termos cujo tamanho é igual à aridade de P,*

5. se $\phi(P, \overline{Q}, \overline{x})$ é uma fórmula de MIN(FO) positiva em P, então

$$\forall \overline{x}(\phi(P, \overline{Q}, \overline{x}) \to P(\overline{x}))$$

é uma condição PIA. (Uma fórmula do tipo $[MINR \bullet \phi(R, \overline{Q})](\bar{t})$ é positiva em P, para $P \neq R$, se $\phi(R, \overline{Q})$ é positiva em P.)

A relação de satisfação envolvendo uma S-interpretação $\mathfrak{I}$ e uma S-fórmula de MIN(FO) da forma $[MINP \bullet \phi(P, \overline{Q})](\bar{t})$ é definida da seguinte forma: se $\mathbf{P}^m$ é o predicado em A tal que $(\mathfrak{I}, \mathbf{P}^m)$ é modelo P-minimal de $\phi(P, \overline{Q})$, então

$$\mathfrak{I} \models [MINP \bullet \phi(P, \overline{Q})](\bar{t}) \text{ sss } \mathfrak{I}(\bar{t}) \in \mathbf{P}^m.$$

Conforme demonstrado em [vB05], MIN(FO) é equivalente a LFP em poder expressivo.

Teorema 4.2 (van Benthem) *$MIN(FO) = LFP$ em poder expressivo.*

O fato de o operador MIN ser aplicado apenas a condições PIA estendidas garante que o modelo minimal existe e é único. Nós propomos a generalização de permitir a aplicação do operador MIN a qualquer fórmula, fazendo uma modificação na definição da relação de satisfação para o caso das fórmulas que utilizam o operador MIN.

4.2 As Lógicas U-MIN e I-MIN

Como vimos na seção anterior, o operador MIN da lógica MIN(FO) é somente aplicado a condições PIA estendidas. Nós iremos generalizar a lógica MIN(FO) permitindo a aplicação do operador MIN a qualquer fórmula. Entretanto, uma $S \cup \{P\}$-fórmula qualquer $\phi(P)$ pode possuir vários modelos minimais $(\mathfrak{I}, \mathbf{P})$ para uma mesma S-interpretação $\mathfrak{I}$. Veja o seguinte exemplo.

Exemplo 4.1 *Seja $S = \{c_a, c_b\}$ um conjunto de símbolos e $\mathfrak{I}$ uma S-interpretação com domínio $A := \{a, b\}$ tal que $\mathfrak{I}(c_a) = a$ e $\mathfrak{I}(c_b) = b$. Seja $\phi := P(c_a) \vee P(c_b)$ uma $S \cup \{P\}$-fórmula. Existem duas expansões de $\mathfrak{I}$ que são modelos de ϕ: uma $S \cup \{P\}$-interpretação $(\mathfrak{I}, \mathbf{P})$ tal que $\mathbf{P} := \{a\}$ e outra tal que $\mathbf{P} := \{b\}$.*

A fim de dar interpretação para o operador MIN nós realizamos duas operações sobre os modelos minimais de uma dada fórmula, levando a duas lógicas: a lógica U-MIN, que toma a união dos modelos minimais, e I-MIN, que utiliza a interseção dos modelos minimais. Nós definimos as duas lógicas abaixo.

Definição 4.4 (A Lógica U-MIN) *A linguagem da lógica U-MIN estende a linguagem da lógica de primeira ordem adicionando a seguinte regra ao cálculo de fórmulas:*

$$\frac{\phi(P)}{[MIN^u P \bullet \phi(P)](\bar{t})} \ ,$$

onde P é uma variável relacional de aridade k, $\bar{t}$ é uma tupla de termos da mesma aridade de P, e $\phi(P)$ é uma $S \cup \{P\}$-fórmula de U-MIN.

Seja $\mathfrak{I} = (\mathfrak{A}, \beta)$ uma S-interpretação. Então $\mathfrak{I} \models [MIN^u P \bullet \phi(P)](\bar{t})$ se, e somente se, existe um predicado $\mathbf{P}$ sobre A, onde a aridade de $\mathbf{P}$ é k, tal que $(\mathfrak{I}, \mathbf{P})$ é modelo P-minimal de $\phi(P)$ e $\mathfrak{I}(\bar{t}) \in \mathbf{P}$.

Se considerarmos a interseção dos predicados minimais obtemos a lógica I-MIN.

Definição 4.5 (A Lógica I-MIN) *A linguagem da lógica I-MIN estende a linguagem da lógica de primeira ordem adicionando a seguinte regra ao cálculo de fórmulas:*

$$\frac{\phi(P)}{[MIN^i P \bullet \phi(P)](\bar{t})} \ ,$$

onde P é uma variável relacional de aridade k, $\bar{t}$ é uma tupla de termos da mesma aridade de P, e $\phi(P)$ é uma $S \cup \{P\}$-fórmula de I-MIN.

Seja $\mathfrak{I} = (\mathfrak{A}, \beta)$ uma S-interpretação. Então $\mathfrak{I} \models [MIN^i P \bullet \phi(P)](\bar{t})$ se, e somente se, existe modelo P-minimal $(\mathfrak{I}, \mathbf{P})$ de $\phi(P)$ e para todo predicado $\mathbf{P}$ sobre A, onde a aridade de $\mathbf{P}$ é k, tal que $(\mathfrak{I}, \mathbf{P})$ é modelo P-minimal de $\phi(P)$, $\mathfrak{I}(\bar{t}) \in \mathbf{P}$.

As variáveis relacionais que compõem a tupla $\overline{P}$ em $[MIN^u P_i.\overline{P} \bullet \phi](\bar{t})$ e $[MIN^i P_i.\overline{P} \bullet \phi](\bar{t})$ são chamadas de *variáveis de minimização* ou *variáveis minimizadas*.

As lógicas U-MIN e I-MIN estão intimamente relacionadas. De fato, nós provamos que elas são equivalentes em poder expressivo.

Teorema 4.3 *Toda fórmula ϕ de I-MIN é equivalente a uma fórmula ψ de U-MIN e vice-versa.*

Prova. Mostraremos que toda fórmula ϕ em I-MIN é equivalente a alguma ψ de U-MIN por indução em ϕ. O caso difícil é quando

$$\phi := [MIN^i P \bullet \alpha(P)](\bar{t}).$$

Por Hipótese Indutiva, existe uma fórmula $\alpha'(P)$ de U-MIN equivalente a $\alpha(P)$. Seja

$$\psi := \exists y[MIN^u P \bullet \alpha'(P)](y) \wedge \neg\exists z[MIN^u P \bullet \alpha'(P) \wedge \neg P(\bar{t})](z).$$

Afirmamos que ϕ e ψ são equivalentes. Seja $\mathfrak{I}$ uma interpretação. Dividiremos a prova em dois casos: (i) $\alpha(P)$ não possui modelos P-minimais com relação a $\mathfrak{I}$, isto é, não existe um predicado $\mathbf{P}$ sobre A, tal que $(\mathfrak{I}, \mathbf{P})$ é modelo P-minimal de $\alpha(P)$. Como $\alpha'(P)$ é equivalente a $\alpha(P)$, $\alpha'(P)$ também não possui modelos P-minimais com relação a $\mathfrak{I}$. Nesse caso $\mathfrak{I} \not\models \phi$ e $\mathfrak{I} \not\models \psi$, logo vale a equivalência. (ii) $\alpha(P)$ possui modelos minimais com relação a $\mathfrak{I}$. Se $(\mathfrak{I}, \emptyset)$ for modelo P-minimal de $\alpha(P)$, então $(\mathfrak{I}, \emptyset) \not\models \phi$ e $\mathfrak{I} \not\models \psi$. Caso contrário, $\mathfrak{I} \models \phi$ sss, para todo modelo P-minimal $(\mathfrak{I}, \mathbf{P})$ de $\alpha(P)$, $\mathfrak{I}(\bar{t}) \in \mathbf{P}$ sss não existe modelo P-minimal $(\mathfrak{I}, \mathbf{P})$ de $\alpha(P)$ tal que $\mathfrak{I}(\bar{t}) \notin \mathbf{P}$ sss $\alpha'(P) \wedge \neg P(\bar{t})$ não possui modelos minimais sss $\mathfrak{I} \not\models \psi$. Logo vale a equivalência.

Para mostrar que toda fórmula de U-MIN possui equivalente em I-MIN, seja

$$\phi := [MIN^u P \bullet \alpha(P)](\bar{t}).$$

Por Hipótese Indutiva, existe uma fórmula $\alpha'(P)$ de U-MIN equivalente a $\alpha(P)$. Seja

$$\psi := \exists \overline{y}[MIN^i P' \bullet [MIN^i P \bullet P \subseteq P' \wedge \alpha'(P)](\bar{t})](\overline{y}).$$

Se $\mathfrak{I} \models \phi$, então existe um modelo P-minimal $(\mathfrak{I}, \mathbf{P})$ de $\alpha(P)$ tal que $\mathfrak{I}(\bar{t}) \in \mathbf{P}$. Seja $\mathbf{P}' = \mathbf{P}$. Então $(\mathfrak{I}, \mathbf{P}', \mathbf{P})$ é modelo P-minimal de $P \subseteq P' \wedge \alpha'(P)$, e como $\mathfrak{I}(\bar{t}) \in \mathbf{P}$, temos que

$$(\mathfrak{I}, \mathbf{P}') \models [MIN^i P \bullet P \subseteq P' \wedge \alpha'(P)](\bar{t}).$$

Observe ainda que $(\mathfrak{I}, \mathbf{P}')$ é modelo P'-minimal de

$$[MIN^i P \bullet P \subseteq P' \wedge \alpha'(P)](\bar{t})$$

pois, caso contrário, $(\mathfrak{I}, \mathbf{P})$ não seria modelo P-minimal de $\alpha(P)$. Portanto $[MIN^i P \bullet P \subseteq P' \wedge \alpha'(P)](\bar{t})$ possui modelos P'-minimais. Seja $(\mathfrak{I}, \mathbf{P}')$ um modelo P'-minimal qualquer de $[MIN^i P \bullet P \subseteq P' \wedge \alpha'(P)](\bar{t})$. Como

$$(\mathfrak{I}, \mathbf{P}') \models [MIN^i P \bullet P \subseteq P' \wedge \alpha'(P)](\bar{t}),$$

temos, para todo modelo P-minimal $(\mathfrak{I}, \mathbf{P}', \mathbf{P})$ de $P \subseteq P' \wedge \alpha'(P)$, que $\mathbf{P} \subseteq \mathbf{P}'$ e $\mathfrak{I}(\bar{t}) \in \mathbf{P}$, portanto temos $\mathfrak{I}(\bar{t}) \in \mathbf{P}'$. Isto é, $\mathfrak{I}(\bar{t}) \in \mathbf{P}'$ para todo modelo minimal $(\mathfrak{I}, \mathbf{P}')$ de $[MIN^i P \bullet P \subseteq P' \wedge \alpha'(P)](\bar{t})$, o que significa que o predicado definido por

$$[MIN^i P' \bullet [MIN^i P \bullet P \subseteq P' \wedge \alpha'(P)](\bar{t})](\overline{y})$$

não é vazio. Logo

$$\mathfrak{I} \models \exists \overline{y}[MIN^i P' \bullet [MIN^i P \bullet P \subseteq P' \wedge \alpha'(P)](\bar{t})](\overline{y}),$$

isto é, $\mathfrak{I} \models \psi$. Suponha agora que $\mathfrak{I} \models \psi$. Isso significa que existe um elemento que pertence a $\mathbf{P}'$, para todo modelo P'-minimal $(\mathfrak{I}, \mathbf{P}')$ de $[MIN^i P \bullet P \subseteq P' \wedge \alpha'(P)](\bar{t})$. Portanto existe $(\mathfrak{I}, \mathbf{P}')$ que é modelo P'-minimal de $[MIN^i P \bullet P \subseteq P' \wedge \alpha'(P)](\bar{t})$. Nesse caso, $\mathfrak{I}(\bar{t}) \in \mathbf{P}$ para todo modelo P-minimal $(\mathfrak{I}, \mathbf{P}', \mathbf{P})$ de $P \subseteq P' \wedge \alpha'(P)$. Portanto existe modelo P-minimal $(\mathfrak{I}, \mathbf{P}', \mathbf{P})$ de $P \subseteq P' \wedge \alpha'(P)$. Mas, nesse caso, $(\mathfrak{I}, \mathbf{P})$ é modelo P-minimal de $\alpha'(P)$—pois, caso contrário, $(\mathfrak{I}, \mathbf{P}')$ não seria modelo P'-minimal de $[MIN^i P \bullet P \subseteq P' \wedge \alpha'(P)](\bar{t})$. Logo $\mathfrak{I} \models \phi$. ∎

Sendo U-MIN e I-MIN equivalentes, de agora em diante utilizaremos como padrão a lógica U-MIN, chamando-a apenas de MIN, e eliminaremos o sobrescrito ^{u} do operador MIN. Quando necessário utilizaremos o operador MIN^i, sendo, nesse caso, entendido como sendo seu equivalente em U-MIN.

Na próxima seção, nós apresentaremos uma extensão da lógica MIN na qual nós permitimos a minimização de vários predicados simultaneamente.

4.3 Minimização Simultânea

Em [vB05] é sugerida a idéia estender MIN(FO) a fim de permitirmos que vários predicados sejam minimizados simultaneamente. Nós tornamos essa idéia precisa para MIN. Vejamos a definição da lógica Si-MIN de minimização simultânea abaixo.

Definição 4.6 (A Lógica Si-MIN) *A lógica Si-MIN estende a lógica de primeira-ordem adicionando a seguinte regra ao cálculo de formação das fórmulas:*

$$\frac{\phi(\overline{P})}{[MIN^u P_i.\overline{P} \bullet \phi](\bar{t})},$$

onde $\overline{P} = P_1, \dots, P_n$, a aridade de P_i é k_i, $\bar{t}$ é uma tupla de termos com comprimento igual à aridade de P_i, e $\phi(\overline{P})$ é uma $S \cup \{\overline{P}\}$-fórmula de Si-MIN.

Seja $\mathfrak{I} = (\mathfrak{A}, \beta)$ uma interpretação. Então $\mathfrak{I} \models [MIN^u P_i.\overline{P} \bullet \phi](\bar{t})$ se, e somente se, existe uma tupla de predicados $\overline{\mathbf{P}} = \mathbf{P}_1, \dots, \mathbf{P}_n$, onde a aridade de $\mathbf{P}_i$ é k_i, tal que $(\mathfrak{I}, \overline{\mathbf{P}})$ é modelo $\overline{P}$-minimal de $\phi(\overline{P})$ e $\mathfrak{I}(\bar{t}) \in \mathbf{P}_i$.

Quando utilizamos o operador MIN em, por exemplo, $[MIN\ P_i.\overline{P} \bullet \phi](\bar{t})$, estamos minimizando simultaneamente vários predicados, representados pelo vetor de variáveis relacionais $\overline{P}$. No entanto, é possível limitar a utilização do operador MIN a apenas um predicado, sem que isso diminua a expressividade de Si-MIN. De fato, mostraremos que Si-MIN é equivalente a MIN em poder expressivo. Antes de demonstrar isso, vamos provar os seguintes lemas.

Lema 4.4 *Sejam P e P' símbolos predicativos com aridade k e $k+1$, respectivamente. Seja $\phi(y,P)$ uma $S \cup \{P\}$-fórmula de MIN, e $\mathfrak{I} = (\mathfrak{A}, \beta)$ uma S-interpretação tal que $\mathfrak{I}(y) = a$. Seja $\mathbf{P}$ um predicado da mesma aridade de P. Então $(\mathfrak{I}, \mathbf{P})$ é modelo de $\phi(y,P)$ se, e somente se, $(\mathfrak{I}, \mathbf{P} \times \{a\})$ é modelo de $\phi(y, P'\_y)$, onde $\phi(y, P'\_y)$ é obtido a partir de $\phi(y,P)$ substituindo átomos da forma $P(\bar{t})$ por $P'(\bar{t}, y)$.*

Prova. Por indução em $\phi(y,P)$.

—$\phi(y,P) = R(\bar{t})$. Se $R \neq P$, temos $\phi(y, P'\_y) = R(\bar{t})$. Logo $(\mathfrak{I}, \mathbf{P}) \models \phi(y,P)$ se, e somente se, $\mathfrak{I} \models R(\bar{t})$ se, e somente se, $(\mathfrak{I}, \mathbf{P} \times \{a\}) \models \phi(y, P'\_y)$.

Se $R = P$, então $\phi(y, P'\_y) = P'(\bar{t}, y)$. Logo $(\mathfrak{I}, \mathbf{P}) \models \phi(y,P)$ se, e somente se, $(\mathfrak{I}, \mathbf{P}) \models P(\bar{t})$ se, e somente se, $\mathfrak{I}(\bar{t}) \in \mathbf{P}$ se, e somente se, $(\mathfrak{I}(\bar{t}), a) \in \mathbf{P} \times \{a\}$ se, e somente se, $(\mathfrak{I}, \mathbf{P} \times \{a\}) \models P'(\bar{t}, y)$ se, e somente se $(\mathfrak{I}, \mathbf{P} \times \{a\}) \models \phi(y, P'\_y)$.

—$\phi(y,P) = \gamma(y,P) \vee \theta(y,P)$. Nesse caso

$$\phi(y, P'\_y) = \gamma(y, P'\_y) \vee \theta(y, P'\_y).$$

Logo $(\mathfrak{I}, \mathbf{P}) \models \phi(y,P)$ se, e somente se, $(\mathfrak{I}, \mathbf{P}) \models \gamma(y,P)$ ou $(\mathfrak{I}, \mathbf{P}) \models \theta(y,P)$ se, e somente se, pela hipótese indutiva,

$$(\mathfrak{I}, \mathbf{P} \times \{a\}) \models \gamma(y, P'\_y) \text{ ou } (\mathfrak{I}, \mathbf{P} \times \{a\}) \models \theta(y, P'\_y)$$

se, e somente se,

$$(\mathfrak{I}, \mathbf{P} \times \{a\}) \models \gamma(y, P'\_y) \vee \theta(y, P'\_y)$$

se, e somente se, $(\mathfrak{I}, \mathbf{P} \times \{a\}) \models \phi(y, P'\_y)$.

—$\phi(y,P) = \neg\gamma(y,P)$. Nesse caso $\phi(y, P'\_y) = \neg\gamma(y, P'\_y)$. Logo $(\mathfrak{I}, \mathbf{P}) \models \phi(y,P)$ se, e somente se, $(\mathfrak{I}, \mathbf{P}) \not\models \gamma(y,P)$ se, e somente se, pela hipótese indutiva, $(\mathfrak{I}, \mathbf{P} \times \{a\}) \not\models \gamma(y, P'\_y)$ se, e somente se, $(\mathfrak{I}, \mathbf{P} \times \{a\}) \models \phi(y, P'\_y)$.

—$\phi(y,P) = \exists x \gamma(y,P)$. Nesse caso $\phi(y, P'\_y) = \exists x \gamma(y, P'\_y)$. Logo $(\mathfrak{I}, \mathbf{P}) \models \phi(y,P)$ se, e somente se, existe $b \in A$ tal que $(\mathfrak{I}, \mathbf{P})\frac{b}{x} \models \gamma(y,P)$ se, e somente se, pela hipótese indutiva, existe $b \in A$ tal que

$$(\mathfrak{I}, \mathbf{P} \times \{a\})\frac{b}{x} \models \gamma(y, P'\_y)$$

se, e somente se, $(\mathfrak{I}, \mathbf{P} \times \{a\}) \models \phi(y, P'\_y)$.

—$\phi(y, P) = [MIN\ R_i.\overline{R} \bullet \gamma(y, P)](\bar{t})$. Assim

$$\phi(y, P'\_y) = [MIN\ R_i.\overline{R} \bullet \gamma(y, P'\_y)](\bar{t}).$$

Logo $(\mathfrak{I}, \mathbf{P}) \models \phi(y, P)$ se, e somente se, existe uma tupla de predicados $\overline{\mathbf{R}}$ tal que $(\mathfrak{I}, \overline{\mathbf{R}}, \mathbf{P})$ é modelo $\overline{R}$-minimal de $\gamma(y, P)$ e $\bar{t} \in \mathbf{R}_i$, se, e somente se, pela hipótese indutiva, $(\mathfrak{I}, \overline{\mathbf{R}}, \mathbf{P} \times \{a\})$ é modelo $\overline{R}$-minimal de $\gamma(y, P\_y)$ e $\bar{t} \in \mathbf{R}_i$, se, e somente se,

$$(\mathfrak{I}, \mathbf{P} \times \{a\}) \models [MIN\ R_i.\overline{R} \bullet \gamma(y, P'\_y)](\bar{t})$$

se, e somente se, $(\mathfrak{I}, \mathbf{P} \times \{a\}) \models \phi(y, P'\_y)$. ∎

Corolário 4.5 *Seja $\overline{P}$ uma tupla de símbolos predicativos, P'_i um símbolo predicativo de aridade $k_i + 1$, $\phi(y, \overline{P})$ uma fórmula, $\mathfrak{I}$ uma interpretação tal que $\mathfrak{I}(y) = a$, e $\overline{\mathbf{P}}$ uma tupla de predicados. Então $(\mathfrak{I}, \overline{\mathbf{P}})$ é modelo $\overline{P}$-minimal de $\phi(y, \overline{P})$ se, e somente se, $(\mathfrak{I}, \mathbf{P}_1, \ldots, \mathbf{P}_i \times \{a\}, \ldots, \mathbf{P}_n)$ é modelo $P_1, \ldots, P'_i, \ldots, P_n$-minimal de $\phi(y, P_1, \ldots, P'_{i}\_y, \ldots, P_n)$.*

Lema 4.6 *Seja $\overline{P}$ uma tupla de símbolos predicativos, P'_j um símbolo predicativo de aridade $k_j + 1$. A fórmula $[MIN\ P_i.\overline{P} \bullet \phi(y, \overline{P})](\bar{t})$ é equivalente a*

$$[MIN\ P_i.P_1, \ldots, P'_j, \ldots, P_n \bullet \phi(y, P_1, \ldots, P'_{j}\_y, \ldots, P_n)](\bar{t}),$$

se $i \neq j$, e a

$$[MIN\ P'_j.P_1, \ldots, P'_j, \ldots, P_n \bullet \phi(y, P_1, \ldots, P'_{j}\_y, \ldots, P_n)](\bar{t}y)$$

se $i = j$.

Prova. Seja $\mathfrak{I}$ uma interpretação tal que $\mathfrak{I}(y) = a$. Se $i = j$, então $\mathfrak{I} \models [MIN\ P_i.\overline{P} \bullet \phi(y, \overline{P})](\bar{t})$ se, e somente se, existe uma tupla de predicados $\overline{\mathbf{P}}$ tal que $(\mathfrak{I}, \overline{\mathbf{P}})$ é modelo $\overline{P}$-minimal de $\phi(y, \overline{P})$ e $\mathfrak{I}(\bar{t}) \in \mathbf{P}_j$, se, e somente se, pelo Corolário 4.5, $(\mathfrak{I}, \mathbf{P}_1, \ldots, \mathbf{P}_j \times \{a\}, \ldots, \mathbf{P}_n)$ é modelo $P_1, \ldots, P'_i, \ldots, P_n$-minimal de $\phi(y, P_1, \ldots, P'_{j}\_y, \ldots, P_n)$ e $(\mathfrak{I}(\bar{t}), a) \in \mathbf{P}_j \times \{a\}$, se, e somente se,

$$\mathfrak{I} \models [MIN\ P'_j.P_1, \ldots, P'_j, \ldots, P_n \bullet \phi(y, P_1, \ldots, P_{j}\_y, \ldots, P_n)](\bar{t}y).$$

Se $i \neq j$ a prova é análoga. ∎

Corolário 4.7 *Seja $\overline{P}$ uma tupla de símbolos predicativos, P'_j um símbolo predicativo de aridade $k_j + 1$ e y uma variável que não ocorre em $\phi(\overline{P})$. Então a fórmula $[MIN\ P_i.\overline{P} \bullet \phi(\overline{P})](\overline{t})$ é equivalente a*

$$Qy([MIN\ P_i.P_1, \ldots, P'_j, \ldots, P_n \bullet \phi(P_1, \ldots, P'_{j\_}y, \ldots, P_n)](\overline{t})),$$

se $i \neq j$, e a

$$Qy([MIN\ P'_j.P_1, \ldots, P'_j, \ldots, P_n \bullet \phi(P_1, \ldots, P'_{j\_}y, \ldots, P_n)](\overline{t}y))$$

se $i = j$, onde $Q = \exists$ ou $Q = \forall$.

O Lema 4.6 nos permite adicionar parâmetros aos predicados minimizados, aumentando assim a aridade desses predicados.

Lema 4.8 *Seja ϕ uma fórmula de Si-MIN. Então ϕ é equivalente a uma fórmula ψ com as mesmas variáveis livres e na qual todos os predicados minimizados têm a mesma aridade.*

Prova. Basta utilizar o Lema 4.6 um número finito de vezes para aumentar a aridade dos predicados minimizados de menor aridade utilizando variáveis que não ocorrem em ϕ e quantificar essas variáveis existencialmente ou universalmente de forma análoga à feita no Corolário 4.7. ∎

Observe que, se fixarmos um natural l, toda fórmula de Si-MIN é equivalente a uma fórmula de primeira-ordem com relação a modelos de cardinalidade menor ou igual a l.

Lema 4.9 *Seja ϕ uma fórmula de U-MIN e l um natural. Então existe uma fórmula de primeira ordem $\phi^{\leq l}$ tal que, ϕ e $\phi^{\leq l}$ possuem os mesmos modelos de cardinalidade menor ou igual a l.*

Prova. É fato da lógica de primeira-ordem que qualquer classe de estruturas finitas que é finita a menos de isomorfismos é definida por uma fórmula de primeira-ordem. ∎

Vamos agora demonstrar o lema principal desta seção.

Lema 4.10 *Seja $\phi(\overline{P})$ uma $S \cup \{\overline{P}\}$-fórmula de Si-MIN na qual todos os símbolos predicativos $\overline{P} = P_1, \ldots, P_n$ têm a mesma aridade k. Seja P' um novo símbolo predicativo de aridade $k+1$, $x_1, \ldots, x_n$ variáveis que não ocorrem em $\phi(\overline{P})$, $\mathfrak{I}$ uma S-interpretação de cardinalidade maior ou igual a n e* *tal que $\mathfrak{I}(x_i) = a_i$ e $a_i \neq a_j$ para $0 \leq i \neq j \leq n$. Seja ainda $\overline{\mathbf{P}}$ uma tupla de predicados. Então*

$$(\mathfrak{I}, \overline{\mathbf{P}}) \models \phi(\overline{P}) \text{ sss } (\mathfrak{I}, \bigcup_{1 \leq i \leq n} (\mathbf{P}_i \times \{a_i\})) \models \phi(P'\_x_1, \ldots, P'\_x_n),$$

onde $\phi(P'\_x_1, \ldots, P'\_x_n)$ é obtida a partir de $\phi(\overline{P})$ substituindo átomos da forma $P_i(\bar{t})$ por $P'(\bar{t}, x_i)$.

Prova. Por indução em $\phi(\overline{P})$.

—$\phi(\overline{P}) = R(\bar{t})$. Se $R \neq P_i$, $1 \leq i \leq n$, então $\phi(P'\_x_1, \ldots, P'\_x_n) = R(\bar{t})$ e portanto vale o teorema. Se $R = P_i$ para algum $0 \leq i \leq n$, então $\mathfrak{I} \models \phi(\overline{P})$ se, e somente se, $\mathfrak{I} \models P_i(\bar{t})$ se, e somente se, $\mathfrak{I}(\bar{t}) \in \mathbf{P}_i$ se, e somente se, $(\mathfrak{I}(\bar{t}), a_i) \in \bigcup_{1 \leq i \leq n}(\mathbf{P}_i \times \{a_i\})$ se, e somente se,

$$(\mathfrak{I}, \bigcup_{1 \leq i \leq n} (\mathbf{P}_i \times \{a_i\})) \models P'(\bar{t}, x_i)$$

se, e somente se,

$$(\mathfrak{I}, \bigcup_{1 \leq i \leq n} (\mathbf{P}_i \times \{a_i\})) \models \phi(P'\_x_1, \ldots, P'\_x_n).$$

—$\phi(\overline{P}) = \gamma(\overline{P}) \vee \theta(\overline{P})$. Logo $\mathfrak{I} \models \phi(\overline{P})$ se, e somente se, $\mathfrak{I} \models \gamma(\overline{P})$ ou $\mathfrak{I} \models \theta(\overline{P})$ se, e somente se, pela hipótese indutiva,

$$(\mathfrak{I}, \bigcup_{1 \leq i \leq n} (\mathbf{P}_i \times \{a_i\})) \models \gamma(P'\_x_1, \ldots, P'\_x_n)$$

ou

$$(\mathfrak{I}, \bigcup_{1 \leq i \leq n} (\mathbf{P}_i \times \{a_i\})) \models \theta(P'\_x_1, \ldots, P'\_x_n)$$

se, e somente se,

$$(\mathfrak{I}, \bigcup_{1 \leq i \leq n} (\mathbf{P}_i \times \{a_i\})) \models \phi(P'\_x_1, \ldots, P'\_x_n).$$

—$\phi(\overline{P}) = \neg\gamma(\overline{P})$. Logo $\mathfrak{I} \models \phi(\overline{P})$ se, e somente se, $\mathfrak{I} \models \neg\gamma(\overline{P})$ se, e somente se, $\mathfrak{I} \not\models \gamma(\overline{P})$ se, e somente se, pela hipótese indutiva,

$$(\mathfrak{I}, \bigcup_{1 \leq i \leq n} (\mathbf{P}_i \times \{a_i\})) \not\models \gamma(P'\_x_1, \ldots, P'\_x_n)$$

se, e somente se,

$$(\mathfrak{I}, \bigcup_{1\leq i\leq n} (\mathbf{P}_i \times \{a_i\})) \models \neg\gamma(P'\_x_1, \ldots, P'\_x_n)$$

se, e somente se,

$$(\mathfrak{I}, \bigcup_{1\leq i\leq n} (\mathbf{P}_i \times \{a_i\})) \models \phi(P'\_x_1, \ldots, P'\_x_n).$$

—$\phi(\overline{P}) = \exists y\gamma(\overline{P})$. Logo $\mathfrak{I} \models \phi(\overline{P})$ se, e somente se, $\mathfrak{I} \models \exists y\gamma(\overline{P})$ se, e somente se, existe $b \in A$ tal que $\mathfrak{I}\frac{b}{x} \models \gamma(\overline{P})$ se, e somente se, pela hipótese indutiva, existe $b \in A$ tal que

$$(\mathfrak{I}, \bigcup_{1\leq i\leq n} (\mathbf{P}_i \times \{a_i\}))\tfrac{b}{x} \models \gamma(P'\_x_1, \ldots, P'\_x_n)$$

se, e somente se,

$$(\mathfrak{I}, \bigcup_{1\leq i\leq n} (\mathbf{P}_i \times \{a_i\})) \models \exists y\gamma(P'\_x_1, \ldots, P'\_x_n)$$

se, e somente se,

$$(\mathfrak{I}, \bigcup_{1\leq i\leq n} (\mathbf{P}_i \times \{a_i\})) \models \phi(P'\_x_1, \ldots, P'\_x_n).$$

—$\phi(\overline{P}) = [MIN\ R_i.\overline{R} \bullet \gamma(\overline{R}, \overline{P})](\overline{t})$. Logo $\mathfrak{I} \models \phi(\overline{P})$ se, e somente se, $\mathfrak{I} \models [MIN\ R_i.\overline{R} \bullet \gamma(\overline{P})](\overline{t})$ se, e somente se, existe uma tupla de predicados $\overline{\mathbf{R}}$ tal que $(\mathfrak{I}, \overline{R})$ é modelo $\overline{R}$-minimal de $\gamma(\overline{R}, \overline{P})$ e $\mathfrak{I}(\overline{t}) \in \mathbf{R}_i$, se, e somente se, pela hipótese indutiva, $(\mathfrak{I}, \overline{\mathbf{R}}, \bigcup_{1\leq i\leq n}(\mathbf{P}_i \times \{a_i\}))$ é modelo $\overline{R}$-minimal de $\gamma(\overline{R}, P'\_x_1, \ldots, P'\_x_n)$ e $\overline{t} \in \mathbf{R}_i$, se, e somente se,

$$(\mathfrak{I}, \bigcup_{1\leq i\leq n} (\mathbf{P}_i \times \{a_i\})) \models [MIN\ R_i.\overline{R} \bullet \gamma(\overline{R}, P'\_x_1, \ldots, P'\_x_n)](\overline{t})$$

se, e somente se,

$$(\mathfrak{I}, \bigcup_{1\leq i\leq n} (\mathbf{P}_i \times \{a_i\})) \models \phi(P'\_x_1, \ldots, P'\_x_n).$$

∎

Teorema 4.11 *Seja ϕ uma fórmula de U-MIN. Então ϕ é equivalente a uma fórmula ψ onde todo operador MIN somente ocorre na forma $[MIN\ P'.P' \bullet \gamma](\overline{t})$, onde P' é um símbolo predicativo.*

Prova. A prova é novamente por indução em ϕ. Pelo Corolário 4.7 podemos assumir que todas as variáveis de minimização de cada operador MIN têm a mesma aridade. Novamente o caso difícil da indução é quando $\phi = [MIN\ \ P_i.\overline{P} \bullet \gamma(\overline{P})](\bar{t})$. Pela hipótese indutiva existe uma fórmula $\gamma'(\overline{P})$ equivalente a $\gamma(\overline{P})$ com as condições desejadas. Seja P' um símbolo predicativo de aridade $k+1$. Seja ψ a fórmula

$$\phi^{\leq n-1} \vee \exists x_1 \dots \exists x_n (\bigwedge_{1 \leq i \neq j \leq n} (x_i \neq x_j) \wedge$$

$$[MINP'.P' \bullet \gamma'(P'\_x_1, \dots, P'\_x_n)](\bar{t}x_i)),$$

onde $x_1, \dots, x_n$ não ocorrem em $\gamma(\overline{P})$. Seja $\mathfrak{I}$ uma interpretação. Se a cardinalidade de $\mathfrak{I}$ é menor do que n, então ϕ é equivalente a ψ pelo Lema 4.9. Seja a cardinalidade de $\mathfrak{I}$ maior ou igual a n. Sejam $a_1, \dots, a_n \in A$. Se $\mathfrak{I} \models \phi$, então $\mathfrak{I}\frac{a_1,\dots,a_n}{x_1,\dots,x_n} \models \phi$. Logo, existe uma tupla de predicados $\overline{\mathbf{P}}$ tal que $(\mathfrak{I}\frac{a_1,\dots,a_n}{x_1,\dots,x_n}, \overline{\mathbf{P}})$ é modelo $\overline{P}$-minimal de $\gamma'(\overline{P})$ e $\mathfrak{I}(t) \in \mathbf{P}_i$. Mas pelo Lema 4.10, $(\mathfrak{I}\frac{a_1,\dots,a_n}{x_1,\dots,x_n}, \bigcup_{1 \leq i \leq n} (\mathbf{P}_i \times \{a_i\}))$ é modelo P'-minimal de $\gamma'(P'\_x_1, \dots, P'\_x_n)$, e $(\mathfrak{I}(\bar{t}), a_i) \in \bigcup_{1 \leq i \leq n} (\mathbf{P}_i \times \{a_i\})$. Logo

$$\begin{aligned} \mathfrak{I} \ \models\ & \exists x_1 \dots \exists x_n (\bigwedge_{1 \leq i \neq j \leq n} (x_i \neq x_j) \wedge \\ & [MINP'.P' \bullet \gamma'(P'\_x_1, \dots, P'\_x_n)](\bar{t}x_i)), \end{aligned}$$

logo $\mathfrak{I} \models \psi$. Se $\mathfrak{I} \models \psi$, então temos dois casos: 1) se $\mathfrak{I} \models \phi^{\leq n-1}$ é fácil ver que $\mathfrak{I} \models \phi$, 2) se $\mathfrak{I} \models \exists x_1 \dots \exists x_n (\bigwedge_{1 \leq i \neq j \leq n} (x_i \neq x_j) \wedge [MINP'.P' \bullet \gamma'(P'\_x_1, \dots, P'\_x_n)](\bar{t}x_i))$, então existem $a_1, \dots, a_n \in A$ tais que $a_i \neq a_j$ para $1 \leq i \neq j \leq n$ e

$$\mathfrak{I}\frac{a_1,\dots,a_n}{x_1,\dots,x_n} \models [MINP'.P' \bullet \gamma'(P'\_x_1, \dots, P'\_x_n)](\bar{t}x_i)),$$

ou seja, existe uma tupla de predicados $\overline{\mathbf{P}}$ tal que

$$(\mathfrak{I}\frac{a_1,\dots,a_n}{x_1,\dots,x_n}, \bigcup_{1 \leq i \leq n} (\mathbf{P}_i \times \{a_i\}))$$

é modelo P'-minimal de $\gamma'(P'\_x_1, \dots, P'\_x_n)$ e $(\mathfrak{I}(\bar{t}), a_i) \in \mathbf{P}_i$. Logo, pelo Lema 4.10, $(\mathfrak{I}\frac{a_1,\dots,a_n}{x_1,\dots,x_n}, \overline{\mathbf{P}})$ é modelo $\overline{P}$-minimal de $\gamma'(\overline{P})$, e $\mathfrak{I}\frac{a_1,\dots,a_n}{x_1,\dots,x_n}(\bar{t}) \in \mathbf{P}_i$. Logo $\mathfrak{I}\frac{a_1,\dots,a_n}{x_1,\dots,x_n} \models \phi$, e como as variáveis $x_1, \dots, x_n$ não ocorrem em ϕ, $\mathfrak{I} \models \phi$. ∎

Como vimos, as lógicas Si-MIN e MIN são equivalentes em poder expressivo. Iremos mostrar agora que o operador MIN confere grande poder expressivo à lógica MIN. De fato, mostraremos que MIN é equivalente à lógica de segunda-ordem em poder expressivo. Para tanto, utilizaremos o seguinte lema[1].

Lema 4.12 *Seja $\phi(P)$ uma $S \cup \{P\}$-fórmula de MIN. Seja $\widetilde{P}$ um símbolo predicativo com a mesma aridade n de P. Para toda $S \cup \{P\}$-interpretação $(\mathfrak{I}, \mathbf{P})$ temos*

$$(\mathfrak{I}, \mathbf{P}) \models \phi(P)$$

se, e somente se,

$$(\mathfrak{I}, \mathbf{P}, \widetilde{\mathbf{P}}) \text{ é modelo } P, \widetilde{P}\text{-minimal de } \phi(P) \wedge \forall\overline{x}(\widetilde{P}(\overline{x}) \leftrightarrow \neg P(\overline{x})),$$

onde $\widetilde{\mathbf{P}} = A^n - \mathbf{P}$.

Prova. Se $(\mathfrak{I}, \mathbf{P}, \widetilde{\mathbf{P}})$ é modelo $P, \widetilde{P}$-minimal de $\phi(P) \wedge \forall\overline{x}(\widetilde{P}(\overline{x}) \leftrightarrow \neg P(\overline{x}))$, então, em particular, $(\mathfrak{I}, \mathbf{P}, \widetilde{\mathbf{P}}) \models \phi(P) \wedge \forall\overline{x}(\widetilde{P}(\overline{x}) \leftrightarrow \neg P(\overline{x}))$. Logo $(\mathfrak{I}, \mathbf{P}) \models \phi(P)$.

Suponha agora que $(\mathfrak{I}, \mathbf{P}) \models \phi(P)$. Seja $\widetilde{\mathbf{P}} = A^n - \mathbf{P}$. Obviamente, $(\mathfrak{I}, \mathbf{P}, \widetilde{\mathbf{P}}) \models \phi(P) \wedge \forall\overline{x}(\widetilde{P}(\overline{x}) \leftrightarrow \neg P(\overline{x}))$. Seja $(\mathfrak{I}, \mathbf{P}', \widetilde{\mathbf{P}'})$ modelo de $\phi(P) \wedge \forall\overline{x}(\widetilde{P}(\overline{x}) \leftrightarrow \neg P(\overline{x}))$ tal que $(\mathfrak{I}, \mathbf{P}', \widetilde{\mathbf{P}'}) \leq^{P,\widetilde{P}} (\mathfrak{I}, \mathbf{P}, \widetilde{\mathbf{P}})$. Nesse caso, $\widetilde{\mathbf{P}'} = A^n - \mathbf{P}'$. Além disso, $\mathbf{P}' \subseteq \mathbf{P}$, logo $(A^n - \mathbf{P}) \subseteq (A^n - \mathbf{P}')$, portanto, $\widetilde{\mathbf{P}} \subseteq \widetilde{\mathbf{P}'}$. Mas também $\widetilde{\mathbf{P}'} \subseteq \widetilde{\mathbf{P}}$, logo $\widetilde{\mathbf{P}} = \widetilde{\mathbf{P}'}$ e $\mathbf{P}' = \mathbf{P}$. Logo $(\mathfrak{I}, \mathbf{P}, \widetilde{\mathbf{P}})$ é modelo $P, \widetilde{P}$-minimal de $\phi(P) \wedge \forall\overline{x}(\widetilde{P}(\overline{x}) \leftrightarrow \neg P(\overline{x}))$. ∎

O seguinte lema mostra que toda fórmula de segunda-ordem possui equivalente em MIN e vice-versa.

Teorema 4.13 *MIN é equivalente à lógica de segunda-ordem em poder expressivo.*

Prova. Primeiro mostraremos que toda fórmula de MIN possui equivalente em segunda-ordem. Para tanto, precederemos por indução na fórmula ϕ de MIN. O caso difícil é quando a fórmula $\phi = [MIN P \bullet \psi(P)](\bar{t})$. Por Hipótese Indutiva, existe uma fórmula de segunda-ordem $\psi'(P)$ equivalente a $\psi(P)$. Seja $\phi' = \exists P(Circ[\psi'(P); P] \wedge P(\bar{t}))$. Temos que $\mathfrak{I} \models \phi'$ sss

$$\mathfrak{I} \models \exists P(Circ[\psi'(P); P] \wedge P(\bar{t}))$$

sss existe um predicado $\mathbf{P}$ sobre A tal que

$$(\mathfrak{I}, \mathbf{P}) \models Circ[\psi'(P); P] \wedge P(\bar{t})$$

[1] O Lema 4.12 é baseia-se num resultado de [dKK89] para a eliminação dos predicados fixos de uma circunscrição

sss existe um predicado $\mathbf{P}$ sobre A tal que

$$(\mathfrak{I}, \mathbf{P}) \models Circ[\psi'(P); P] \text{ e } (\mathfrak{I}, \mathbf{P}) \models P(\bar{t})$$

sss existe um predicado $\mathbf{P}$ sobre A tal que $(\mathfrak{I}, \mathbf{P})$ é modelo P-minimal de $\psi'(P)$ e $\mathfrak{I}(\bar{t}) \in \mathbf{P}$ sss, por Hipótese Indutiva, existe um predicado $\mathbf{P}$ sobre A tal que $(\mathfrak{I}, \mathbf{P})$ é modelo P-minimal de $\psi(P)$ e $\mathfrak{I}(\bar{t}) \in \mathbf{P}$ sss

$$\mathfrak{I} \models [MINP \bullet \psi(P)](\bar{t}) \text{ sss } \mathfrak{I} \models \phi.$$

Seja agora ϕ' uma fórmula de segunda-ordem. Vamos provar por indução em ϕ' que toda fórmula de segunda-ordem possui equivalente em MIN. O caso difícil é quando $\phi' = \exists P \psi'(P)$ para alguma fórmula $\psi'(P)$ de segunda-ordem. Por Hipótese Indutiva, existe uma fórmula $\psi(P)$ de MIN que é equivalente a $\psi'(P)$. Seja

$$\phi = \psi(\emptyset) \vee \exists \overline{y}[MINP.P, \widetilde{P} \bullet \psi(P) \wedge \forall \overline{x}(\widetilde{P}(\overline{x}) \leftrightarrow \neg P(\overline{x}))](\overline{y}).$$

Seja $\mathfrak{I}$ uma interpretação. Então $\mathfrak{I} \models \phi'$ sss $\mathfrak{I} \models \exists P \psi'(P)$ sss existe um predicado $\mathbf{P}$ sobre A tal que $(\mathfrak{I}, \mathbf{P}) \models \psi'(P)$ sss, por Hipótese Indutiva, $(\mathfrak{I}, \mathbf{P}) \models \psi(P)$. Dividimos a prova em dois casos: i) se $\mathbf{P} = \emptyset$, então $(\mathfrak{I}, \mathbf{P}) \models \psi(P)$ sss $(\mathfrak{I}, \mathbf{P}) \models \psi(\emptyset)$ sss $(\mathfrak{I}, \mathbf{P}) \models \phi$; ii) se $\mathbf{P} \neq \emptyset$, então $(\mathfrak{I}, \mathbf{P}) \models \psi(P)$ sss existe $a \in A^n$ tal que $a \in \mathbf{P}$ e $(\mathfrak{I}, \mathbf{P}) \models \psi(P)$ sss, pelo Lema 4.12, existe $a \in A^n$ tal que $a \in \mathbf{P}$ e $(\mathfrak{I}, \mathbf{P}, \widetilde{\mathbf{P}})$ é modelo $P, \widetilde{P}$-minimal de

$$\psi(P) \wedge \forall \overline{x}(\widetilde{P}(\overline{x}) \leftrightarrow \neg P(\overline{x}))$$

sss existe $a \in A^n$ tal que

$$\mathfrak{I}^{\overline{y}}_{a} \models [MINP.P, \widetilde{P} \bullet \psi(P) \wedge \forall \overline{x}(\widetilde{P}(\overline{x}) \leftrightarrow \neg P(\overline{x}))](\overline{y})$$

sss

$$\mathfrak{I} \models \exists \overline{y}[MINP.P, \widetilde{P} \bullet \psi(P) \wedge \forall \overline{x}(\widetilde{P}(\overline{x}) \leftrightarrow \neg P(\overline{x}))](\overline{y})$$

sss $(\mathfrak{I}, \mathbf{P}) \models \phi$. Por i) e ii), $\mathfrak{I} \models \phi'$ sss $\mathfrak{I} \models \phi$. ∎

Na próxima seção, falaremos sobre certos quantificadores, que chamamos de *quantificadores minimais*, e sua relação com a lógica MIN.

4.4 Quantificadores Minimais

Se observarmos a definição da relação de satisfação para a lógica MIN, veremos que está implícita no operador MIN uma quantificação de segunda-ordem. De fato, para estabelecer essa relação entre uma interpretação $\mathfrak{I}$ e fórmulas da forma $[MINP \bullet \phi](\bar{t})$, verificamos se "existe um predicado **P** ...." Essa quantificação implícita quantifica apenas predicados minimais de ϕ, isto é, predicados que formam modelos minimais $(\mathfrak{I}, \mathbf{P})$ de ϕ. Isso faz com que a lógica MIN esteja estreitamente relacionada com uma certa extensão da lógica de primeira-ordem através de *quantificadores minimais.*

Definição 4.7 (A Lógica MQ(FO)) *A lógica MQ(FO) é a extensão da lógica de primeira-ordem através dos quantificadores minimais* $\exists^{\psi(P)}$ *(lê-se "existe um predicado minimal* **P** *de* $\psi(P)$ *tal que ...") e* $\forall^{\psi(P)}$ *(lê-se "para todo predicado minimal* **P** *de* $\psi(P)$ *...") para cada fórmula* $\psi(P)$ *de MQ(FO). Nós definimos a linguagem de MQ(FO) adicionando a seguinte regra ao cálculo de formação das fórmulas bem formadas:*

- *se* $\psi(P)$ *e* $\phi(P)$ *são S-fórmulas de MQ(FO), então*

$$\exists^{\psi(P)} P\phi(P) \text{ e } \forall^{\psi(P)} P\phi(P)$$

  são S-fórmulas de MQ(FO).

A relação de satisfação entre uma S-interpretação $\mathfrak{I}$ *e uma S-fórmula da forma* $\exists^{\psi(P)} P\phi(P)$ *é definida como* $\mathfrak{I} \models \exists^{\psi(P)} P\phi(P)$ *sss existe um predicado* **P** *sobre A tal que* $(\mathfrak{I}, \mathbf{P})$ *é um modelo P-minimal de* $\psi(P)$ *e* $(\mathfrak{I}, \mathbf{P}) \models \phi(P)$*. No caso de uma S-fórmula da forma* $\forall^{\psi(P)} P\phi(P)$ *temos* $\mathfrak{I} \models \forall^{\psi(P)} P\phi(P)$ *sss existe um predicado* **P** *sobre A tal que* $(\mathfrak{I}, \mathbf{P})$ *seja um modelo P-minimal de* $\psi(P)$ *e para todo modelo P-minimal* $(\mathfrak{I}, \mathbf{P})$ *de* $\psi(P)$*,* $(\mathfrak{I}, \mathbf{P}) \models \phi(P)$*.*

Esses quantificadores minimais podem ser definidos em lógica de segunda-ordem. Logo MQ(FO) está contida na lógica de segunda-ordem em poder expressivo. Mostramos isso no seguinte lema.

Lema 4.14 *Para toda fórmula* θ *de MQ(FO) existe uma forma* θ' *de segunda-ordem equivalente.*

Prova. Procedemos por indução em θ os casos difíceis sendo quando $\theta = \exists^{\psi(P)} P\phi(P)$ e $\theta = \forall^{\psi(P)} P\phi(P)$ para $\phi(P)$ e $\psi(P)$ em MQ(FO). Por Hipótese Indutiva, existe $\phi'(P)$ equivalente a $\phi(P)$ e $\psi'(P)$ equivalente a $\psi(P)$. Suponha $\theta = \exists^{\psi(P)} P\phi(P)$. Seja

$$\theta' := \exists P(Circ[\psi'(P); P] \wedge \phi'(P))$$

e $\mathfrak{I}$ uma interpretação. $\mathfrak{I} \models \exists^{\psi(P)} P\phi(P)$ sss existe um predicado **P** sobre A tal que $(\mathfrak{I}, \mathbf{P})$ é modelo P-minimal de $\psi(P)$ e $(\mathfrak{I}, \mathbf{P}) \models \phi(P)$ sss, por Hipótese Indutiva, existe um predicado **P** sobre A tal que $(\mathfrak{I}, \mathbf{P})$ é modelo P-minimal de $\psi'(P)$ e $(\mathfrak{I}, \mathbf{P}) \models \phi'(P)$ sss existe um predicado $\mathbf{P}$ sobre A tal que $(\mathfrak{I}, \mathbf{P})$ é modelo de $Circ[\psi(P); P]$ e $(\mathfrak{I}, \mathbf{P}) \models \phi'(P)$ sss

$$\mathfrak{I} \models \exists P(Circ[\psi'(P); P] \wedge \phi'(P))$$

sss $\mathfrak{I} \models \theta'$.

Suponha $\theta = \forall^{\psi(P)} P\phi(P)$. Seja

$$\theta' := \exists P(Circ[\psi'(P); P]) \wedge \forall P(Circ[\psi'(P); P] \rightarrow \phi'(P)).$$

$\mathfrak{I} \models \forall^{\psi(P)} P\phi(P)$ sss existe um predicado $\mathbf{P}$ tal que $(\mathfrak{I}, \mathbf{P})$ é modelo P-minimal de $\psi(P)$ e para todo modelo P-minimal $(\mathfrak{I}, \mathbf{P})$ de $\psi(P)$, $(\mathfrak{I}, \mathbf{P}) \models \phi(P)$ sss, por Hipótese Indutiva, existe um predicado $\mathbf{P}$ tal que $(\mathfrak{I}, \mathbf{P})$ é modelo P-minimal de $\psi'(P)$ e para todo modelo P-minimal $(\mathfrak{I}, \mathbf{P})$ de $\psi'(P)$, $(\mathfrak{I}, \mathbf{P}) \models \phi'(P)$ sss existe um predicado $\mathbf{P}$ tal que $(\mathfrak{I}, \mathbf{P}) \models Circ[\psi'(P); P]$ e para todo modelo $(\mathfrak{I}, \mathbf{P})$ de $Circ[\psi'(P); P]$, $(\mathfrak{I}, \mathbf{P}) \models \phi'(P)$ sss

$$\mathfrak{I} \models \exists P(Circ[\psi'(P); P]) \wedge \forall P(Circ[\psi'(P); P] \rightarrow \phi'(P))$$

sss $\mathfrak{I} \models \theta'$. ∎

Por outro lado, é possível mostrar que MIN está contida em MQ(FO). Basta observar que $[MIN^u P \bullet \phi(P)](\bar{t})$ é equivalente a $\exists^{\phi(P)} P(P(\bar{t}))$ e que $[MIN^i P \bullet \phi(P)](\bar{t})$ é equivalente a $\forall^{\phi(P)} P(P(\bar{t}))$. Assim obtemos:

Teorema 4.15 *MIN = MQ(FO) = SO em poder expressivo.*

Na próxima seção, analisaremos um fragmento da lógica MIN cujo poder expressivo está entre o da Lógica de Menor Ponto Fixo e o da lógica de segunda-ordem.

4.5 O Fragmento MIN$_\Delta$

Como vimos, a forma como lidamos com modelos minimais em MIN através do operador MIN e em MQ(FO) através dos quantificadores minimais nos levam a lógicas com poder expressivo equivalente ao da lógica de segunda-ordem. Algo semelhante acontece com as NATs de Lifschitz através do aninhamento de blocos. Isso mostra que essas operações de minimização são bastante poderosas, no que diz respeito ao poder expressivo. Isso nos motiva a procurar por fragmentos dessas lógicas que possuam menor poder expressivo. Isso pode ser feito, por exemplo, restringindo a aplicação do operador

MIN. A lógica MIN(FO) de van Benthem é um caso de restrição do operador MIN a condições PIA estendidas[2]. A seguir, exibiremos um fragmento da lógica MIN que utiliza o operador MIN em sua sintaxe, embora de forma restrita.

Seja $\subseteq_{\mathrm{MIN(FO)}}$ uma relação entre estruturas definidas de forma análoga a $\subseteq_{\mathrm{LFP}}$—observe que, sendo MIN(FO) e Lógica de Menor Ponto Fixo equivalentes em poder expressivo, essas relações são iguais. Utilizaremos a seguinte propriedade para restringir a aplicação do operador MIN.

Definição 4.8 *Uma S-fórmula $\phi(\overline{X}, P, \overline{y})$ de MIN com as variáveis relacionais $\overline{X} := X_1, \ldots, X_n$ e P, onde P tem aridade m, tem a propriedade χ (escreve-se $\chi(\phi)$) com relação a P sss, para quaisquer estruturas $\mathfrak{A}$ e $\mathfrak{B}$, todas as tuplas de predicados $\overline{\mathbf{X}}^A := \mathbf{X}_1^A, \ldots, \mathbf{X}_n^A$ sobre A e que interpretam $\overline{X}$, todas as tuplas de predicados $\overline{\mathbf{X}}^B := \mathbf{X}_1^B, \ldots, \mathbf{X}_n^B$ sobre B e que interpretam $\overline{X}$ tal que $\mathfrak{B}' = (\mathfrak{B}, \overline{\mathbf{X}}^B) \subseteq_{\mathrm{MIN(FO)}} (\mathfrak{A}, \overline{\mathbf{X}}^A) = \mathfrak{A}'$ (ou, alternativamente, para todas $S \cup \{\overline{X}\}$-estruturas $\mathfrak{A}' = (\mathfrak{B}, \overline{\mathbf{X}}^B)$ e $\mathfrak{B}' = (\mathfrak{A}, \overline{\mathbf{X}}^A)$ tais que $(\mathfrak{B}, \overline{\mathbf{X}}^B) \subseteq_{\mathrm{MIN(FO)}} (\mathfrak{A}, \overline{\mathbf{X}}^A)$) e todos os assinalamentos β sobre B, temos:*

1. *se $\mathfrak{I} = ((\mathfrak{A}', \mathbf{P}), \beta)$ é modelo P-minimal de $\phi(\overline{X}, P, \overline{y})$, então $\mathfrak{I}' = ((\mathfrak{B}', \mathbf{P} \cap B^m), \beta)$ é modelo P-minimal de $\phi(\overline{X}, P, \overline{y})$;*

2. *se $\mathfrak{I}' = ((\mathfrak{B}', \mathbf{P}'), \beta)$ é modelo P-minimal de $\phi(\overline{X}, P, \overline{y})$, então existe um predicado $\mathbf{P}$ sobre A tal que $\mathbf{P}' = \mathbf{P} \cap B^m$ e $\mathfrak{I} = ((\mathfrak{A}', \mathbf{P}), \beta)$ é modelo P-minimal de $\phi(\overline{X}, P, \overline{y})$.*

O fragmento MIN_Δ é obtido restringindo-se a aplicação do operador MIN apenas a fórmulas que possuam a propriedade χ.

Definição 4.9 (A Lógica $\mathbf{MIN}_\Delta$) *A lógica MIN_Δ é a extensão da lógica de primeira-ordem obtida adicionando a seguinte regra ao cálculo de formação das fórmulas:*

$$\frac{\phi(\overline{Q}, P, \overline{y})}{[MIN P \bullet \phi(\overline{Q}, P, \overline{y})](\overline{t})}\ , \ \textit{se}\ \chi(\phi).$$

Nós provaremos inicialmente que a lógica MIN(FO) está contida em MIN_Δ em poder expressivo.

Teorema 4.16 *Toda condição PIA estendida $\forall \overline{x}(\psi(\overline{X}, P, \overline{x}, \overline{y}) \rightarrow P(\overline{x}))$ possui a propriedade χ com relação a P.*

[2]Vale ressaltar que MIN(FO) [vB05] surgiu antes de MIN [FM06].

Prova. Seja $\phi(\overline{X}, P, \overline{y}) = \forall\overline{x}(\psi(\overline{X}, P, \overline{x}, \overline{y}) \to P(\overline{x}))$ uma condição PIA estendida. Sejam $\mathfrak{A}'$ e $\mathfrak{B}'$ $S \cup \{\overline{X}\}$-estruturas tais que $\mathfrak{A}' \subseteq_{\text{MIN(FO)}} \mathfrak{B}'$ e β um assinalamento em B. Sejam $((\mathfrak{A}', \mathbf{P}), \beta)$ e $((\mathfrak{B}', \mathbf{P}'), \beta)$ modelos P-minimais de $\phi(\overline{X}, P, \overline{y})$. Assim temos:

$$\mathbf{P} = \{\overline{a} \in A | \mathfrak{A}' \models [MINP \bullet \phi(\overline{Q}, P, \overline{y})](\overline{z})[\mathfrak{I}(\overline{y}), \overline{a}]\}.$$

Analogamente,

$$\mathbf{P}' = \{\overline{b} \in B | \mathfrak{B}' \models [MINP \bullet \phi(\overline{Q}, P, \overline{y})](\overline{z})[\mathfrak{I}'(\overline{y}), \overline{b}]\}.$$

($\overline{a}$ e $\overline{b}$ estão interpretando $\overline{z}$ acima.) Como $\mathfrak{A}' \subseteq_{\text{MIN(FO)}} \mathfrak{B}'$, temos, para todo $\overline{b} \in B^n$,

$$\mathfrak{B}' \models [MINP \bullet \phi(\overline{Q}, P, \overline{y})](\overline{z})[\mathfrak{I}'(\overline{y}), \overline{b}]$$

sss

$$\mathfrak{A}' \models [MINP \bullet \phi(\overline{Q}, P, \overline{y})](\overline{z})[\mathfrak{I}'(\overline{y}), \overline{b}].$$

Logo $\mathbf{P}' = \mathbf{P} \cap B^n$. ∎

Entretanto, MIN_Δ é mais expressiva que MIN(FO). Para provar isso, nós mostraremos que EVEN, a classe das estruturas pares no alfabeto vazio, é definível em MIN_Δ. Como a Lógica de Menor Ponto Fixo possui a 0–1 *law* [Lib04], a Lógica de Menor Ponto Fixo e, pelo Teorema 4.2, também MIN(FO) não definem EVEN.

Teorema 4.17 *MIN_Δ define EVEN.*

Prova. Considere a fórmula $\psi(<)$ do Exemplo 2.2. Conforme demonstramos, um modelo $(\mathfrak{A}, \prec)$ de $\psi(<)$ é uma estrutura cujo domínio têm cardinalidade par e $\prec$ é uma ordem linear sobre A. A fórmula

$$\lambda := \exists x_1 x_2 [MIN < \bullet \psi(<)](x_1 x_2)$$

define a classe das estruturas de cardinalidade par. Para mostrar isso, seja $\mathfrak{A}$ uma estrutura. $\mathfrak{A} \models \phi$ sss existem $a_1, a_2 \in A$ tais que

$$\mathfrak{A} \models [MIN < \bullet \psi(<)](x_1 x_2)[a_1, a_2]$$

sss existe uma relação binária $\prec$ sobre A tal que $(\mathfrak{A}, \prec)$ é um modelo $<$-minimal de $\psi(<)$ e $(a_1, a_2) \in \prec$. Mas $(\mathfrak{A}, \prec)$ é modelo de $\psi(<)$ sss $\mathfrak{A}$ for uma estrutura de cardinalidade par e $\prec$ for uma ordem linear (total) sobre A. Mas isso significa que todo modelo de $\phi(<)$ é $<$-minimal, pois a relação $\prec$ não pode ser diminuída sem perder a totalidade. Portanto, existem $a_1, a_2 \in A$ tais que

$$\mathfrak{A} \models [MIN < \bullet\psi(<)](x_1x_2)[a_1, a_2]$$

sss existe uma relação binária $\prec$ tal que $(\mathfrak{A}, \prec)$ é um modelo de $\psi(<)$ e $(a_1, a_2) \in \prec$ sss $\mathfrak{A}$ for uma estrutura de cardinalidade par.

Mostraremos agora que λ pertence a MIN_Δ. Como LFP = MIN(FO) $\subseteq \mathrm{MIN}_\Delta$, $\psi(<) \in \mathrm{MIN}_\Delta$. Portanto, basta mostrar que $\psi(<)$ possui a propriedade χ com relação a $<$. Sejam $\mathfrak{A}$ e $\mathfrak{B}$ estruturas no mesmo alfabeto e tais que $\mathfrak{A} \subseteq_{\mathrm{MIN(FO)}} \mathfrak{B}$. Se $(\mathfrak{A}, \prec^A) \models \psi(<)$ para alguma relação binária $\prec^A$ sobre A, então $\mathfrak{A}$ é uma estrutura finita e portanto pode ser definida a menos de isomorfismo por uma fórmula de MIN(FO)—na verdade, mesmo por uma fórmula de primeira-ordem. Portanto, a única MIN(FO)-subestrutura de $\mathfrak{A}$ é a própria $\mathfrak{A}$. Logo $\mathfrak{A} = \mathfrak{B}$ e assim o Item 1 da Definição 4.8 é satisfeito. Por outro lado, se $(\mathfrak{B}, \prec^B) \models \psi(<)$, então $\mathfrak{B}$ é finita e, analogamente, $\mathfrak{B} = \mathfrak{A}$, portanto o Item 2 da Definição 4.8 é satisfeito. Logo $\psi(<)$ possui a propriedade χ com relação a $<$. Dessa forma, $\lambda \in \mathrm{MIN}_\Delta$. ∎

Nós podemos provar entretanto, que MIN_Δ é menos expressiva que a lógica de segunda-ordem. Para isso, mostraremos que o Teorema de Löwenheim-Skolem vale para MIN_Δ. Seja $\subseteq_{\mathrm{MIN}_\Delta}$ definida de forma análoga a $\subseteq_{\mathrm{LFP}}$, substituindo LFP por MIN_Δ na Definição 2.20. Nós provaremos o seguinte lema.

Lema 4.18 *Seja $\mathfrak{A}$ uma S-estrutura infinita. Então existe uma S-estrutura $\mathfrak{B}$ contável que é uma MIN_Δ-subestrutura de $\mathfrak{A}$, isto é, $\mathfrak{A} \subseteq_{\mathrm{MIN}_\Delta} \mathfrak{B}$.*

Prova. Seja $\mathfrak{A}$ uma S-estrutura infinita e $\mathfrak{B}$ uma MIN(FO)-subestrutura de $\mathfrak{A}$ fechada sobre MIN_Δ-fórmulas existenciais, isto é, para cada S-fórmula $\exists x\phi(y_1, \ldots, y_n, x)$ de MIN_Δ sem variáveis relacionais livres, e para quaisquer $b_1, \ldots, b_n \in B$, existe $b \in B$ tal que, se $\mathfrak{A}' \models \exists x\phi(y_1, \ldots, y_n, x)[b_1, \ldots, b_n]$, então $\mathfrak{A}' \models \phi(y_1, \ldots, y_n, x)[b_1, \ldots, b_n, b]$. Por indução em uma fórmula de ϕ de MIN_Δ mostraremos que

$$\mathfrak{A} \models \phi[\bar{b}] \text{ sss } \mathfrak{A} \models \phi[\bar{b}],$$

todo $\bar{b} = b_1, \ldots, b_n \in B^n$. A Hipótese Indutiva é utilizada no caso dos conectivos e quantificadores. Para o caso do operador MIN, nós utilizamos a propriedade χ. Seja $\phi = [MINP\bullet\psi](\bar{t})$. Se $\mathfrak{A} \models \phi$, então existe um modelo P-minimal $\mathfrak{I} = ((\mathfrak{A}, \mathbf{P}), \beta)$ de ϕ, com $\beta(y_i) = b_i$ para cada variável livre y_i de ϕ, tal que $\mathfrak{I}(\bar{t}) \in \mathbf{P}$. Como ϕ é uma fórmula de MIN_Δ, ψ possui a propriedade χ com relação a P. Pela Definição 4.8, Item 1, $\mathfrak{I}' = ((\mathfrak{B}, \mathbf{P} \cap B^n), \beta)$ é um modelo P-minimal de ψ, portanto $\mathfrak{B} \models \phi[\bar{b}]$. Por outro lado, se $\mathfrak{B} \models \phi[\bar{b}]$, então existe um modelo P-minimal $\mathfrak{I}' = ((\mathfrak{B}, \mathbf{P}'), \beta)$ de ψ, onde $\beta(y_i) = b_i$ para cada variável livre y_i de ϕ, tal que $\mathfrak{I}(\bar{t}) \in \mathbf{P}'$. Pela Definição 4.8, Ítem 2, existe um predicado $\mathbf{P}$ sobre A tal que $\mathbf{P}' = \mathbf{P} \cap B^n$ e $\mathfrak{I} = ((\mathfrak{A}, \mathbf{P}), \beta)$ é modelo P-minimal de ψ, logo $\mathfrak{A} \models \phi[\bar{b}]$. ∎

Portanto temos:

Teorema 4.19 *Todo conjunto contável e satisfatível de fórmulas de* MIN_Δ *possui um modelo contável.*

Isso mostra que MIN_Δ é um fragmento de MIN que possui os operadores booleanos e os quantificadores de primeira-ordem e possui poder expressivo intermediário entre a Lógica de Menor Ponto Fixo e a lógica de segunda-ordem.

Capítulo 5

Conclusão e Trabalhos Futuros

Neste trabalho, foram examinadas lógicas obtidas a partir da lógica de primeira-ordem através de certos operadores introduzidos na linguagem e que têm como principal característica a utilização de modelos minimais em suas interpretações. Chamaremos tais operadores de *operadores de minimização.* O foco de nossas atenções foi o estudo do poder expressivo dessas lógicas. As lógicas que examinamos se encaixam no padrão de sistema lógico como apresentado na Definição 1.25. Esses sistemas lógicos possuem semântica baseada em estruturas matemáticas (veja Definição 1.15). Portanto, o estudo do poder expressivo de um sistema lógico (como é o caso das lógicas que aqui foram tratadas) corresponde ao estudo das classes de estruturas definíveis/exprimíveis[1] pelas sentenças ou teorias desse sistema lógico. A natureza comum (isto é, de estrutura matemática) dos modelos considerados facilita também a comparação entre o poder expressivo das lógicas tratadas.

A questão da definibilidade, no sentido de Beth, também é examinada em alguns casos (veja, principalmente, Capítulo 2 e Seção 3.5). Questões acerca da definibilidade de símbolos não lógicos são casos especiais de problemas de expressividade. Considere a seguinte definição:

Definição 5.1 *Seja $S \cup \{P\}$ um alfabeto e $\mathbb{C}$ uma classe de $S \cup \{P\}$-estruturas. Dizemos que $\mathbb{C}$ é P-definida se qualquer S-estrutura pode ser expandida para no máximo uma estrutura em $\mathbb{C}$.*

Dessa forma, questões sobre a definibilidade, no sentido de Beth, de símbolos não lógicos podem ser examinadas como questões de expressividade, por se

[1] Por "classe de estrutura exprimível por uma lógica," entenda-se as classes de modelos de sentenças ou teorias dessa lógica. Na literatura, usa-se bastante o termo "definível" ao invés de "exprimível." Como estamos lidando com definibilidade de símbolos não lógicos, no sentido de Beth, convém prestar atenção ao contexto.

tratarem de problemas sobre as classes de modelos P-definidos que uma lógica é capaz de expressar.

Como vimos, as lógicas estudadas aqui são obtidas através da adição de operadores sintáticos de caráter relacional cuja interpretação depende dos modelos minimais (ou predicados minimais) de uma determinada fórmula à qual tal operador é aplicado. Isso se verifica na lógica MIN(FO), através do operador MIN aplicado a condições PIA; em U-MIN e I-MIN através dos operadores MIN^u e MIN^i aplicados a fórmulas quaisquer; analogamente em Si-MIN com a minimização simultânea; em MIN_Δ com a aplicação do operador MIN a fórmulas com a propriedade χ; em Circunscrição com o operador $Circ$ e em NATs através dos blocos. Embora a Lógica de Menor Ponto Fixo não utilize explicitamente em sua definição os modelos minimais, van Benthem demonstrou sua equivalência expressiva com MIN(FO) [vB05].

Começamos então o estudo do poder expressivo dessas lógicas com a Lógica de Menor Ponto Fixo. A primeira questão que abordamos foi o Teorema de Beth. Nós mostramos que o Teorema de Beth não vale para Lógica de Menor Ponto Fixo. A técnica utilizada no Teorema 2.6 pode ser utilizada para mostrar que esse teorema falha para qualquer extensão da lógica de primeira-ordem capaz de definir o modelo padrão da aritmética a menos de isomorfismo e que possua linguagem contável. Entretanto nossa construção utiliza teorias infinitas, o que nos leva a considerar uma restrição do mesmo a teorias finitas. Utilizando um Teorema de Hodkinson sobre a lógica infinitária $L^{\omega}_{\omega_1\omega}$, provamos, através do Corolário 2.9, que o Teorema de Beth falha mesmo se nos restringirmos a definições implícitas feitas por teorias finitas[2]. Na verdade, provamos que o Teorema Fraco de Beth também falha (ver Corolário 2.10). Seguimos a investigação sobre definições na Lógica de Menor Ponto Fixo com um estudo de caso a respeito de definições feitas pelo que chamamos de sistemas recursivos (Seção 2.3.2). Mostramos que definições implícitas realizadas por sistemas recursivos sempre possuem definição explícita.

Cabe aqui uma observação. Nota-se, na literatura que trata do assunto, certo intercâmbio entre os termos "definição indutiva" e "definição recursiva" (analogamente com os termos "indução" e "recursão"). No entanto, baseado nos textos sobre definições indutivas, onde é estudado o conceito de "conjunto indutivamente gerado por um operador monótono", sugerimos que uma razoável interpretação de "definição indutiva" é considerar como tal toda definição cujo *definiendum* é identificado com o menor ponto fixo de um operador monótono, isto é, o *definiens* deve ser entendido como tal ponto

[2]É interessante contrastar com a lógica de segunda-ordem na qual, segundo argumento semelhante ao do Teorema 2.6, o Teorema de Beth falha no caso geral mas vale na sua restrição a teorias finitas.

fixo. Já no caso do termo "definição recursiva," e por entender recursividade como algo ligado à auto-referência, sugerimos que "definição recursiva" é qualquer equivalência do tipo

$$\textit{símbolo} \leftrightarrow \textit{expressão},$$

na qual o *símbolo* pode ocorrer em *expressão*, e com a propriedade adicional de definir implicitamente *símbolo*. Essa última condição é importante, uma vez que nem toda equivalência daquele tipo define *símbolo*. Embora definição implícita seja uma condição semântica, é possível fazer restrições sintáticas a fim de garanti-la. Nesse sentido foi que incluímos os *sistemas recursivos* da Seção 2.3.2. Observe que estamos falando não só do objeto definido, mas principalmente da definição enquanto objeto de estudo. Daí a potencial importância desses conceitos em uma Teoria das Definições[3].

Uma ferramenta importante quando se estuda o poder expressivo de uma lógica é o Teorema de Löwenheim-Skolem. Esse teorema afirma que qualquer classe de estruturas definível por sentenças de uma lógica na qual este teorema valha possui um modelo contável. A validade desse teorema para uma lógica implica certo "limite expressivo" dessa lógica, uma vez que atesta a sua incapacidade de definir classes de estruturas onde todas as estruturas são incontáveis. Esboços[4] de provas de tal teorema para a Lógica de Menor Ponto Fixo podem ser encontrados em [Grä02, Flu99], onde se mostra que fórmulas da Lógica de Menor Ponto Fixo sempre possuem modelos contáveis. Aqui, nós mostramos que mesmo teorias contáveis (infinitas) de sentenças da Lógica de Menor Ponto Fixo possuem modelo contável(veja Teorema 2.20). Mais que isso, mostramos que a generalização de Teorema de Löwenheim-Skolem para teorias de qualquer cardinalidade também vale (veja Teorema 2.22).

Continuamos nosso trabalho no Capítulo 3, onde estudamos a Circunscrição de McCarthy. Em Circunscrição, o uso de modelos minimais se torna explícito através do operador *Circ*. Na Seção 3.3, provamos alguns lemas a respeito do poder expressivo da Circunscrição. Chamamos atenção para o Lema 3.4. Nesse lema, mostramos que a Circunscrição não é fechada paras as operações booleanas, isto é, não é um sistema lógico booleano, no sentido da Definição 1.26. Mostramos no Exemplo 3.1 que o fecho reflexivo e transitivo de uma relação binária qualquer pode ser definido implicitamente através

[3]No entanto, se essas sugestões notacionais que fizemos introduzirem uniformidade e eliminarem ambiguidades, seu valor já se justifica.

[4]Vale observar que tais esboços tratam do caso em que a fórmula em questão possui apenas um operador de ponto fixo, ficando a generalização para fórmulas quaisquer a cargo do leitor.

de uma circunscrição. Isso também pode ser feito[5] na Lógica de Menor Ponto Fixo. Mas, por definição, a Lógica de Menor Ponto Fixo é um sistema lógico booleano e, portanto, consegue definir o complemento dessa classe de modelos. O Lema 3.4 mostra que o mesmo não acontece com a Circunscrição. Logo, existem fórmulas na Lógica de Menor Ponto Fixo sem equivalente em Circunscrição. Mesmo com essa falta de expressividade, a Circunscrição também é capaz de definir classes de estruturas que a Lógica de Menor Ponto Fixo não consegue. Para provar isso, recorremos ao supracitado Teorema de Lowenheim-Skolem. No Lema 3.5, mostramos que para toda sentença em Π^1_1 existe uma circunscrição que é uma extensão conservativa de tal sentença. Isso garante que o Teorema de Löwenheim-Skolem falha para Circunscrição e, portanto, existe um circunscrição que define uma classe de estruturas nas quais o domínio é incontável. O mesmo não acontece com a Lógica de Menor Ponto Fixo, como provamos através do Teorema 2.20.

Como citamos no início da Seção 3.4, a falha do Teorema de Löwenheim-Skolem para Circunscrição foi primeiramente provada em [Sch87]. Nessa mesma seção nós exibimos uma elegante prova alternativa, utilizando raciocínio diverso daquele apresentado em [Sch87].

Na Seção 3.5, fazemos novas considerações a respeito de definibilidade, agora no contexto da Circunscrição. Nos exibimos um contra-exemplo correto para uma conjectura proposta por Doyle em [Doy85]. A questão se refere à possibilidade de substituir a circunscrição de uma teoria finita pela teoria finita mais a definição explícita do símbolo circunscrito—claro, quando a circunscrição definir implicitamente tal símbolo. Em termos formais, Doyle afirma:

$$Circ_{\mathrm{FO}}[\alpha; P] \not\equiv \alpha \wedge \forall \overline{x}(P(\overline{x}) \leftrightarrow \phi(\overline{x})),$$

onde $Circ_{\mathrm{FO}}$ é o esquema de Circunscrição de primeira-ordem e P não ocorre em $\phi(\overline{x})$. Embora tal afirmação esteja correta, o contra-exemplo de Doyle não funciona. Nós reformulamos a questão para o caso da Circunscrição de segunda-ordem e exibimos um contra-exemplo correto para a questão que funciona em ambos os casos (de primeira-ordem e segunda-ordem). Fomos além. De fato, não é qualquer definição $\phi(\overline{x})$ que satisfaz a equivalência contestada por Doyle. No entanto, mostramos que, sempre que uma circunscrição de segunda-ordem é equivalente a uma teoria de primeira-ordem e define implicitamente o símbolo circunscrito, existe uma definição explícita

[5] A sentença

$$\forall vw(R(v,w) \leftrightarrow [\mathbf{lfp}_{X,xy}(x = y) \vee \exists z(X(x,z) \wedge E(z,y))](v,w)),$$

define implicitamente (e explicitamente) o símbolo R que é o fecho reflexivo e transitivo de E.

$\phi'(\overline{x})$ para a qual a equivalência acima se verifica, isto é:

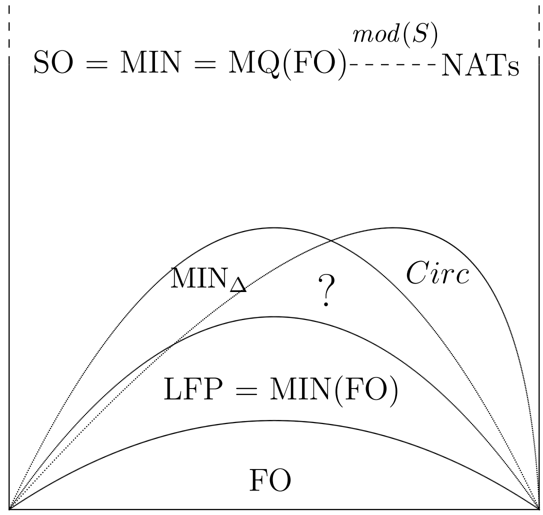
$$Circ[\alpha; P; \overline{Z}] \equiv \alpha \wedge \forall \overline{x}(P(\overline{x}) \leftrightarrow \phi'(\overline{x})).$$

Para tanto, utilizamos o Lema 3.15 que mostra que, se a classe dos modelos de uma circunscrição (possivelmente com objetos variáveis) for Δ-elementar, tal classe será elementar.

No restante do Capítulo 3, nos dedicamos a explorar o poder expressivo das NATs de Lifschitz. Por definição, cada NAT é equivalente a uma teoria de segunda-ordem. Nós lidamos com o problema contrário. Isto é, investigamos quando é que uma teoria de segunda-ordem possui equivalente em NATs. As restrições sintáticas inerentes à definições das NATs dificultam a questão. Por exemplo, não é possível negar blocos explicitamente, ou seja, não existe um elemento sintático que simbolize explicitamente a negação de um bloco. Ao invés de estabelecer equivalência entre NATs e lógica de segunda-ordem, estabelecemos equivalência *módulo* símbolos adicionais. Nós provamos que, para toda sentença em lógica de segunda-ordem, existe uma NAT que é uma extensão conservativa de tal sentença de segunda-ordem. Nós utilizamos o aninhamento de blocos para simular a alternância e o aninhamento de quantificadores de segunda-ordem. Devido a limitações impostas pelas restrições sintáticas que citamos, alguns símbolos novos são introduzidos. Entretanto, fica estabelecido que para qualquer sentença ϕ de segunda-ordem em um alfabeto S, existe uma NAT T em um alfabeto $S' \supseteq S$ tal que ϕ e T implicam logicamente as mesmas S-sentenças. Isso demonstra que as NATs possuem grande poder expressivo (comparável ao da lógica de segunda-ordem). Tanto a NAT T quanto o tempo necessário para obtê-la são lineares no tamanho da fórmula ϕ da lógica de segunda-ordem.

O último capítulo deste trabalho foi dedicado ao estudo das lógicas da família MIN. As lógicas dessa família têm como principal característica o operador MIN para a construção de predicados a partir dos predicados (modelos) minimais de uma determinada fórmula à qual o operador é aplicado. A primeira lógica dessa família é a lógica MIN(FO) de van Benthem. Nós então estendemos essa lógica permitindo a aplicação do operador MIN a qualquer fórmula. Obtemos as lógicas U-MIN e I-MIN. Provamos sua equivalência expressiva e chamamos a lógica U-MIN de MIN. Introduzimos a minimização simultânea e obsevamos que não há acréscimo de poder expressivo. Finalmente, provamos que a lógica MIN é equivalente à lógica de segunda-ordem em poder expressivo. Continuando a investigação a respeito de operadores de minimização, introduzimos os quantificadores minimais. Explicitamos a relação da lógica MQ(FO) de quantificadores minimais com a lógica MIN. Mostramos que novamente obtemos uma lógica de poder expressivo igual ao da lógica de segunda-ordem. A fim de encontrarmos lógicas com poder expressivo intermediário entre o da Lógica de Menor Ponto Fixo e lógica de segunda-ordem, introduzimos o fragmento MIN_Δ. Mostramos que esse fragmento contém a Lógica de Menor Ponto Fixo, mas é mais expressiva do que esta, sendo, no entanto, menos expressiva do que a lógica de segunda-ordem. Além disso, é um sistema lógico booleano que possui os quantificadores de primeira-ordem.

Nós estabelecemos o seguinte mapa representando a expressividade das lógicas investigadas:



onde NATs são equivalentes a fórmulas de segunda-ordem *módulo* símbolos adicionais S', isto é, dada uma S-sentença ϕ, existe uma $S \cup S'$-NAT T tal que $Mod(\phi) = \{\mathfrak{A}|_S | \mathfrak{A} \in Mod(T)\}$, onde $\mathfrak{A}|_S$ é o reduto da $S \cup S'$-estrutura $\mathfrak{A}$ a S.

Além desses resultados listados acima, este estudo comprova que operadores de minimização são bastante poderosos se comparados à quantificação de segunda-ordem. No entanto, restrições impostas à aplicação desses operadores podem levar a lógica de poder expressivo intermediário, como é o caso de MIN_Δ. Outros níveis de expressividade são aqueles definidos pela quantidade de operadores MIN aninhados em uma fórmula. É preciso estabelecer a relação dessa hierarquia com a hierarquia Π, Δ, Σ de fórmulas de segunda-ordem. Como sugestão de trabalho futuro, apontamos a investigação acerca de tais restrições, principalmente restrições sintáticas, a fim de adicionar novos níveis de expressividade ao mapa acima. De forma análoga à forma que os operadores de ponto fixo têm sido estudados no âmbito da Teoria dos Modelos Finitos, principalmente no campo da Complexidade Descritiva

[EF95, Lib04], no intuito de obter novas lógicas capazes de capturar classes de complexidade computacional, os operadores de minimização podem também ser investigados com esse intuito.

Apêndice A

Apêndice

A.1 Reticulados

A seguir, definimos reticulado, reticulado completo e função monótona definida sobre um reticulado. Além disso, apresentamos dois exemplos de reticulados e enunciamos o Teorema de Knaster-Tarski. Esse teorema forma a base das definições indutivas e, consequentemente, da Lógica de Menor Ponto Fixo.

Definição A.1 (Reticulado) *Um par* $(A, \sqsubseteq)$ *onde* A *é um conjunto e* $\sqsubseteq$ *é uma relação de ordem parcial (reflexiva, transitiva e anti-simétrica) é um* reticulado *se, para todos* $a, b \in A$*, existem um supremo com relação a* $\sqsubseteq$ *(denotado por* $\bigsqcup\{a, b\}$*) e um ínfimo com relação a* $\sqsubseteq$ *(denotado por* $\bigsqcap\{a, b\}$*) de* $\{a, b\}$*.*

Definição A.2 (Reticulado Completo) *Se* $(A, \sqsubseteq)$ *é um reticulado tal que qualquer subconjunto* $B \subseteq A$ *possui um ínfimo (denotado por* $\bigsqcap B$*) e um supremo (denotado por* $\bigsqcup B$*), então* $(A, \sqsubseteq)$ *é um* reticulado completo*. Os elementos* $\bigsqcup A$ *e* $\bigsqcap A$ *são chamados* maior *elemento e* menor *elemento de* $(A, \sqsubseteq)$ *e denotados por* $\top$ *e* $\bot$*, respectivamente.*

Seja $F : A \to A$ uma função definida sobre A. Um ponto fixo de F é um $a \in A$ tal que $F(a) = a$. O Teorema de Knaster-Tarski assegura que certas funções chamadas monótonas definidas sobre o domínio de um reticulado possuem menor e maior pontos fixos. Vejamos as definições.

Definição A.3 (Função Monótona) *Dado um reticulado* $(A, \sqsubseteq)$*, uma função* $F : A \to A$ *é* monótona *(ou* preserva ordem*) se, para quaisquer* $a, b \in A$ *tais que* $a \sqsubseteq b$*,* $F(a) \sqsubseteq F(b)$*.*

Teorema A.1 (Knaster-Tarski) *Seja $(A, \sqsubseteq)$ um reticulado completo. Seja $F : A \to A$ uma função monótona com relação a $\sqsubseteq$.*

(a) F possui um menor ponto fixo (denotado por $\mathbf{lfp}(F)$) e um maior ponto fixo (denotado por $\mathbf{gfp}(F)$) tais que:

$$\mathbf{lfp} = \bigsqcap\{a \in A | F(a) \sqsubseteq a\}$$

e

$$\mathbf{gfp} = \bigsqcup\{a \in A | a \sqsubseteq F(a)\};$$

(b) considere a seguinte seqüência F^α, para α ordinal:

$$\begin{aligned} F^0 &= \perp, \\ F^{\alpha+1} &= F(F^\alpha), \\ F^\lambda &= \bigsqcup\{F^\mu | \mu < \lambda\}, \textit{ para } \lambda \textit{ limite.} \end{aligned}$$

Como F é monótona, essa sequência atinge um ponto fixo em algum ordinal. Seja $\mathbf{cl}(F)$ o menor ordinal α tal que $F^\alpha = F^{\alpha+1}$. Então

$$F^{\mathbf{cl}(F)} = \mathbf{lfp}(F).$$

Em [Tar55], Tarski mostra algo um pouco mais forte. Tarski mostra que o par $(P, \sqsubseteq_P)$, onde $P = \{a \in A | F(a) = a\}$ (isto é, P é o conjunto dos pontos fixos de A com relação à função monótona F) e $\sqsubseteq_P$ é a restrição de $\sqsubseteq$ a P, forma um reticulado completo.

A seguir, recordaremos alguns exemplos de reticulados completos.

Exemplo A.1 *Seja C um conjunto. Então $(\wp(C), \subseteq)$ é um reticulado completo, onde, para qualquer $B \subseteq \wp(C)$,*

$$\bigsqcup B = \bigcup B$$

e

$$\bigsqcap B = \bigcap B.$$

Exemplo A.2 *Sejam $C_1, \ldots, C_n$ conjuntos. Seja $C = \wp(C_1) \times \ldots \times \wp(C_n)$. Dado $c \in C$, denotamos por $c(i)$, $1 \leq i \leq n$, o elemento de C_i que ocupa a i-ésima posição da tupla c. Seja $\leq$ uma relação binária sobre C definida como*

$$(D_1, \ldots, D_n) \leq (D'_1, \ldots, D'_n) \textit{ sss } D_i \subseteq D'_i, \textit{ para } 1 \leq i \leq n.$$

Então $(C, \leq)$ *é um reticulado completo. Para demonstrar isso, basta considerar, para cada subconjunto* $B \subseteq C$*, os elementos*

$$\bigsqcup B = (\bigcup_{b \in B} b(1), \ldots, \bigcup_{b \in B} b(n))$$

e

$$\bigsqcap B = (\bigcap_{b \in B} b(1), \ldots, \bigcap_{b \in B} b(n)).$$

Sejam $F_i : C \to \wp(C_i)$*, para* $1 \leq i \leq n$*, tais que, se* $c_1 \leq c_2$*, então* $F_i(c_1) \subseteq F_i(c_2)$*. Seja* $F : C \to C$ *definida como*

$$F(c) = (F_1(c), \ldots, F_n(c)).$$

Obviamente, F *é monótona com relação a* $\leq$*, pois, dados* $c_1, c_2 \in C$ *tais que* $c_1 \leq c_2$*,* $F_i(c_1) \leq F_i(c_2)$*, para* $1 \leq i \leq n$*, logo* $F(c_1) \leq F(c_2)$*.*

Referências Bibliográficas

[Acz77] P. Aczel. An introduction to inductive definitions. In J. Barwise, editor, *Handbook of Mathematical Logic*, pages 739–782. North-Holland, Amsterdam, 1977.

[AN01] A. Arnold and D. Niwiński. *Rudiments of μ-calculus.* North Holland, 2001.

[AU79] Alfred V. Aho and Jeffrey D. Ullman. Universality of data retrieval languages. In *POPL '79: Proceedings of the 6th ACM SIGACT-SIGPLAN symposium on Principles of programming languages*, pages 110–119, New York, NY, USA, 1979. ACM Press.

[Bet53] E. W. Beth. On Padoa's method in the theory of definitions. *Indag. Math.*, 15, 1953.

[CEG92] Marco Cadoli, Thomas Eiter, and Georg Gottlob. An efficient method for eliminating varying predicates from a circumscription. *Artificial Intelligence*, 54(2):397–410, 1992.

[CEG05] Marco Cadoli, Thomas Eiter, and Georg Gottlob. Complexity of propositional nested circumscription and nested abnormality theories. *ACM Trans. Comput. Logic*, 6(2):232–272, 2005.

[CH82] Ashok K. Chandra and David Harel. Structure and complexity of relational queries. *Journal of Computer Systems and Sciences*, 25(1):99–128, 1982.

[DG02] A. Dawar and Y. Gurevich. Fixed-point logics. *Bulletin of Symbolic Logic*, 8(1):65–88, 2002.

[DHK95] Anuj Dawar, Lauri Hella, and Phokion Kolaitis. Implicit definability and infinitary logic in finite model theory. In Z. Fülöp and F. Gécseg, editors, *Proceedings 22nd Int'l Coll. on Automata,* *Languages, and Programming, ICALP'95, Szeged, Hungary, 10–14 July 1995*, volume 944, pages 624–635. Springer-Verlag, Berlin, 1995.

[dKK89] Johan de Kleer and Kurt Konolige. Eliminating the fixed predicates from a circumscription. *Artif. Intell.*, 39(3):391–398, 1989.

[Doy85] Jon Doyle. Circumscription and Implicit Definability. *Journal of Automated Reasoning*, 1(4):391–405, 1985.

[DSW94] M. Davis, R. Sigal, and E. J. Weyuker. *Computability, complexity, and languages: fundamentals of theoretical computer science.* Academic Press Professional, Inc., San Diego, CA, USA, 1994.

[EF95] H.-D. Ebbinghaus and J. Flum. *Finite Model Theory.* Springer-Verlag, 1995. ISBN 3-540-60149-X.

[EFT94] H.-D. Ebbinghaus, J. Flum, and W. Thomas. *Mathematical Logic.* Springer-Verlag, New York, NY, 1994.

[End72] Herbert B. Enderton. *A Mathematical Introduction to Logic.* Academic Press, San Diego, California, 1972.

[Flu99] Jörg Flum. On the (infinite) model theory of fixed-point logics. In X. Caicedo and C. Montenegro, editors, *Models, algebras and proofs*, number 2003 in Lecture Notes in Pure and Applied Mathematics, pages 67–75. Marcel Dekker, 1999.

[FM06] Francicleber M. Ferreira and Ana Teresa Martins. The Predicate-Minimizing Logic MIN. In J.S. Sichman et al., editor, *Lecture Notes in Artificial Intelligence: Proceedings of IBERAMIA/SBIA 2006*, volume 4140, pages 581–591. Springer-Verlag, Berlin, 2006.

[Grä02] Erich Grädel. Guarded fixed point logics and the monadic theory of countable trees. *Theoretical Computer Science*, 288(1):129–152, 2002.

[GS96] Yuri Gurevich and Saharon Shelah. On finite rigid structures. *J. Symb. Log.*, 61(2):549–562, 1996.

[Gur84] Y. Gurevich. Toward logics tailored for computational complexity. In *Computation and Proof Theory*, volume 1104 of *Springer Lecture Notes in Mathematics*, pages 175–216. Springer-Verlag, 1984.

[HJ99] Karel Hrbacek and Thomas Jech. *Introduction to Set Theory*. Marcel Dekker, 1999.

[Hod93] Ian M. Hodkinson. Finite variable logics. *Bulletin of the EATCS*, 51:111–140, 1993.

[Hoo01] Eva Hoogland. *Definability and Interpolation : Model-theoretic investigations*. PhD thesis, Universiteit van Amsterdam: Institute for Logic, Language and Computation, 2001.

[Kre02a] S. Kreutzer. Expressive equivalence of least and inflationary fixed point logic, 2002.

[Kre02b] Stephan Kreutzer. *Pure and Applied Fixed-Point Logics*. PhD thesis, Von der Fakultät für Mathematik, Informatik und Naturwissenschaften der Rheinisch-Westfälischen Technischen Hochschule Aachen, 2002.

[Lib04] Leonid Libkin. *Elements of Finite Model Theory*. Springer, 2004.

[Lif94] V. Lifschitz. Circumscription. In D. M. Gabbay, C. J. Hogger, and J. A. Robinson, editors, *Handbook of Logic in Artificial Intelligence and Logic Programming-Nonmonotonic Reasoning and Uncertain Reasoning(Volume 3)*, pages 297–352. Clarendon Press, Oxford, 1994.

[Lif95] Vladimir Lifschitz. Nested abnormality theories. *Artif. Intell.*, 74(2):351–365, 1995.

[Llo87] John W. Lloyd. *Foundations of Logic Programming, 2nd Edition*. Springer, 1987.

[McC80] John L. McCarthy. Circumscription - a form of non-monotonic reasoning. *Artif. Intell.*, 13(1-2):27–39, 1980.

[McC86] John McCarthy. Applications of circumscription to formalizing common-sense knowledge. *Artif. Intell.*, 28(1), 1986.

[MD80] Drew V. McDermott and Jon Doyle. Non-monotonic logic I. *Artif. Intell.*, 13(1-2):41–72, 1980.

[Mos74] Y. N. Moschovakis. *Elementary Induction on Abstract Structures*. North Holland, 1974.

[Pad00] A. Padoa. Essai d'une théorie algébrique des nombres entiers, precedé d'une introduction logique a une théorie deductive qelconque. In *Bibliothèque Du Congrès Int. de Philos.*, volume 3, pages 118–123, 1900.

[Peq85] Marcelino Cavalcante Pequeno. Lógicas Não-Monotônicas. Master's thesis, Universidade Federal do Ceará, 1985.

[Rei80] Raymond Reiter. A logic for default reasoning. *Artif. Intell.*, 13(1-2):81–132, 1980.

[Sch86] David A. Schmidt. *Denotational Semantics.* Allyn and Bacon, Massachusetts, 1986.

[Sch87] J. Schlipf. Decidability and definability with circumscription. *Annals of Pure and Applied Logic*, 35(2):173–191, 1987.

[Sco82] Dana S. Scott. Domains for denotational semantics. In *Proceedings of the 9th Colloquium on Automata, Languages and Programming*, pages 577–613, London, UK, 1982. Springer-Verlag.

[Tar55] Alfred Tarski. A lattice-theoretical fixpoint theorem and its applications. *Pacific Journal of Mathematics*, 5:285–309, 1955.

[vB05] J. van Benthem. Minimal predicates, fixed-points, and definability. *J. Symbolic Logic*, 70(3):696–712, 2005.

Índice Remissivo