

LÓGICAS NÃO-MONOTÔNICAS

58

MARCELINO CAVALCANTE PEQUENO

---

DISSERTAÇÃO SUBMETIDA À COORDENAÇÃO DO  
CURSO DE MESTRADO EM MATEMÁTICA, COMO REQUISITO PARCIAL  
PARA OBTENÇÃO DO GRAU DE MESTRE  
UNIVERSIDADE FEDERAL DO CEARÁ

FORTALEZA - 1985

UFC/BU/BCM 01/07/1998



R827185 Logicas nao-monotonicas /  
C429988  
T510 P479L

Esta Dissertação foi submetida como parte dos requisitos necessários a obtenção do Grau de Mestre em Matemática, outorgado pela Universidade Federal do Ceará, e encontra-se a disposição dos interessados na Biblioteca Central da referida Universidade.

A citação de qualquer trecho desta Tese é permitida, desde que seja feita de conformidade com as normas da ética científica.

Marcelino C. Pequeno

Marcelino Cavalcante Pequeno

DISSERTAÇÃO APROVADA EM 02/07/85

Tarcísio H. C. Pequeno

Tarcísio H. C. Pequeno

Orientador

Guilherme L. A. Ellery

Guilherme L. A. Ellery

José Sérgio A. Sette

José Sérgio A. Sette

Daniel Schwabe

Daniel Schwabe



Orientador:

TARCISO H.C. PEQUENO

## AGRADECIMENTOS

Duas pessoas foram essenciais à realização deste trabalho: o professor e amigo Ellery, à época também coordenador do Curso, que sempre acreditou em mim, e foi um apoio nas horas difíceis. Sua compreensão e ajuda foram de um valor inestimável. E Zulmira Áurea, a Zú, companheira solidária e um eterno incentivo a que eu avançasse em minha tarefa, este trabalho, de uma certa forma, nos pertence.

Na fase final de datilografia foram valiosos os préstimos do Antônio.

A estes especialmente, e a todos que direta ou indiretamente contribuíram ou não, a este trabalho, meus agradecimentos.

## RESUMO

Lógicas não-monotônicas são lógicas nas quais a introdução de novas premissas podem invalidar antigas conclusões. As semânticas restritivas são semânticas nas quais a classe dos modelos de uma sentença é restringida a uma subclasse particular. Apresentamos as semânticas Minimais, Nomeáveis e Iniciais (obtidas por tomar-se apenas os modelos minimais, nomeáveis e iniciais, respectivamente) e analisamos suas propriedades. Estas semânticas são não-monotônicas (a Nomeável apenas em um certo sentido). Várias foram as formalizações para o raciocínio não-monotônico sugeridas em Inteligência Artificial, apresentamos e analisamos as propriedades da circunscrição e do "default" propostas por McCARTHY e REITER, respectivamente.

## ABSTRACT

Non-monotonic logics are logics in which the introduction of new premises can invalidate old conclusions. Restrictive semantics are semantics in which the model class of a sentence is restricted to a particular subclass. We present the Minimal, Namable and Initial semantics (the model class is restricted to minimal, namable and initial models, respectively) and analyse their properties. These semantics are non-monotonic (the namable one just in a special sense). Many formalizations were propounded to the non-monotonic reasoning in Artificial Intelligence, we present and analyse the properties of circumscription and default, proposed by McCARTHY and REITER, respectively.



## SUMÁRIO

	<u>Página</u>
1 - <u>INTRODUÇÃO</u> .....	1
2 - <u>A LÓGICA DE PRIMEIRA ORDEM (LÓGICA CLÁSSICA)</u> .....	10
2.1 - <u>A Linguagem</u> .....	10
2.2 - <u>Modelos de uma linguagem</u> .....	14
2.3 - <u>Semântica da lógica de primeira ordem</u> .....	17
2.4 - <u>Sintaxe da lógica de primeira ordem</u> .....	19
2.5 - <u>Corretude e completude da lógica de primeira ordem</u> .....	20
2.6 - <u>Teoremas da compacidade, enumerabilidade e Löwenheim-Skolem</u> .....	21
2.7 - <u>Teoria de primeira ordem</u> .....	23
2.8 - <u>Tópicos especiais</u> .....	25
2.8.1 - <u>Funções de Skolem</u> .....	25
2.8.2 - <u>A aritmética</u> .....	28
3 - <u>SEMÂNTICAS RESTRITIVAS</u> .....	31
3.1 - <u>Semântica Minimal</u> .....	40
3.1.1 - <u>Modelos Minimais</u> .....	40
3.1.2 - <u>Conseqüência Lógica Minimal</u> .....	47
3.1.3 - <u>Propriedades</u> .....	48
3.1.3.1 - <u>Não-Monotonicidade</u> .....	48
3.1.3.2 - <u>Inexistência de modelos minimais</u> .....	50
3.1.3.3 - <u>Não-Compacidade</u> .....	52
3.1.3.4 - <u>Enumerabilidade dos modelos minimais</u> .....	53
3.1.3.5 - <u>Incompletude</u> .....	55
3.1.4 - <u>Modelos Minimais e Minimalidade Local</u> .....	57
3.2 - <u>Semânticas Nomeáveis</u> .....	59
3.2.1 - <u>Estruturas Nomeáveis</u> .....	59
3.2.2 - <u>Conseqüência Lógica Nomeável</u> .....	62
3.2.3 - <u>W-Lógica</u> .....	67
3.2.4 - <u>Interpretações de Herbrand</u> .....	70
3.2.5 - <u>Propriedades</u> .....	78
3.2.5.1 - <u>Monotonicidade</u> .....	78
3.2.5.2 - <u>Não-Conservação em extensões da linguagem</u> ..	79

	<u>Página</u>
3.2.5.3 - Existência de modelos nomeáveis .....	80
3.2.5.4 - Incompletude .....	82
3.2.5.5 - Enumerabilidade dos modelos nomeáveis .....	83
3.3 - <u>Semântica Inicial</u> .....	83
3.3.1 - Estrutura natural .....	83
3.3.2 - Inicialidade .....	86
3.3.3 - Igualdade natural .....	87
3.3.4 - Modelos Iniciais .....	90
3.3.5 - Conseqüência Lógica Inicial .....	90
3.3.6 - Propriedades .....	91
3.3.6.1 - Minimalidade e Nomeabilidade .....	91
3.3.6.2 - Inicialidade .....	91
3.3.6.3 - Inexistência de modelos iniciais .....	93
4 - <u>INFERÊNCIAS NÃO-MONOTÔNICAS</u> .....	96
4.1 - Circunscrição .....	103
4.1.1 - Circunscrição do domínio .....	103
4.1.1.1 - Inferência minimal .....	106
4.1.2 - Circunscrição de predicados .....	111
4.1.2.1 - Inferência circumspectiva .....	112
4.1.3 - Propriedades .....	119
4.1.3.1 - Não-Monotonicidade .....	119
4.1.3.2 - Possível inconsistência da circunscrição ...	121
4.1.3.3 - Não-Construtividade .....	123
4.1.4 - Relação entre circunscrição e semântica minimal	124
4.2 - <u>"Default"</u> .....	128
4.2.1 - Teoria "default" e suas extensões .....	128
4.2.2 - Teorias "default" arbitrárias .....	135
4.2.3 - Propriedades .....	138
4.2.3.1 - Não-Monotonicidade .....	138
4.2.3.2 - Inexistência de extensões .....	139
4.2.3.3 - Não semi-decidibilidade das extensões .....	140
5 - <u>CONCLUSÕES</u> .....	143
6 - <u>REFERÊNCIAS BIBLIOGRÁFICAS</u> .....	146

LISTA DE ILUSTRAÇÕES

<u>Figura</u>		<u>Página</u>
1	- Exemplos de estruturas de acesso a tipos de dados .....	35
2	- Modelos de um conjunto de sentenças de primeira ordem (versão 1) .....	43
3	- Modelos de um conjunto de sentenças de primeira ordem (versão 2) .....	46



## I - INTRODUÇÃO

Em nosso trabalho são apresentadas algumas semânticas e inferências não-monotônicas, e verificadas suas propriedades.

Por não-monotonicidade entendemos a propriedade de que uma conclusão tirada a partir de um conjunto de fatos, pode ser rejeitada com a adição de novas informações. Assim, semântica não-monotônica é aquela na qual uma consequência lógica de um conjunto de premissas, pode não ser consequência lógica de um conjunto de premissas contendo o primeiro. Inferência não-monotônica é aquela na qual um teorema de um conjunto de axiomas pode deixar de sê-lo com a adição de novos axiomas. A Lógica Clássica é monotônica: "Se  $\sigma$  é uma conclusão tirada a partir de um conjunto de premissas A, então  $\sigma$  é uma conclusão de  $A \cup B$ , para qualquer conjunto de premissas B.

As Lógicas Não-Monotônicas tiveram origem em duas áreas distintas da Ciência da Computação: em Especificação de Tipos de Dados Abstratos e em Inteligência Artificial (IA). [CARVALHO et al, 1980] defronta-se com o problema da caracterização axiomática para um Tipo de Dado. Dada uma estrutura abstrata deseja-se encontrar alguns axiomas tais que os modelos destes axiomas sejam



as estruturas que realizem a estrutura abstrata. A semântica da Lógica Clássica não atende a este propósito e assim recorre-se aos modelos nomeáveis, minimais e iniciais. A semântica restrita a cada um destes modelos é não-monotônica (a semântica nomeável apenas em um certo sentido, conforme veremos no capítulo 3). Por outro lado surgiram em Inteligência Artificial os "sistemas com modos de inferência estendidos\*" implementados para atender necessidades específicas em diferentes áreas. Posteriormente verificou-se que as inferências executadas nestes sistemas eram não-monotônicas, ao mesmo tempo que tentativas de se formalizar o raciocínio do senso comum em um sistema lógico mostraram-se não-monotônicas, [McCARTHY, 1960].

A relação entre a lógica formal e a operação da mente humana não tem sido clara. Por um lado a lógica foi originalmente o estudo das "leis do pensamento", e tem permanecido pelo menos em princípio como uma teoria prescritiva da operação da mente "ideal". Por outro lado muitos lógicos têm estado interessados em fenômenos mentais ordinários "não-ideais". Algumas das diferenças mais notáveis entre as propriedades da lógica formal e os fenômenos mentais, ocorrem em situações que lidam com percepções, ambigüidades, senso comum, causalidade e predição. Uma característica comum destes problemas é que parecem lidar com conhecimento incompleto. Percepção despreza uma miríade de nuances, senso comum ignora uma grande quantidade de exceções, atribuições de cau

---

\* Em inglês, "Extended inferences modes systems".

sa podem estar enganadas, planos para o futuro consideram contingências que talvez nunca se realizem. São os erros aparentemente inevitáveis nestes casos que levam aos problemas mais profundos da análise formal da mente humana.

Alguns estudos destes problemas ocorrem na literatura filosófica. É relevante o trabalho de [RESCHER, 1964]. Em Inteligência Artificial, estudos de percepção, ambigüidade e senso comum têm levado a representações do conhecimento que simulam facetas da operação da mente humana. São simuladas "defaults", tentativas de inferência por apenas uma certa quantidade de tempo, etc. Por exemplo, "robots" para tomarem uma decisão não esperam até poderem provar uma alternativa (que talvez não seja possível), tomam-na baseados em apenas suposições plausíveis; sistemas dedutivos de pergunta e resposta, respondem "não", quando falham em provar o contrário (o Prolog, por exemplo). Ver [MINSKY, 1974], [REITER, 1978], [DOYLE, 1979], [HAYES, 1970, 71 e 73]. A programação requerida nestes casos é denominada de "programação heurística", e sua eficiência obriga o uso de métodos os quais ocasionalmente erram ou falham. A possibilidade de falha implica que as formalizações do raciocínio nestas áreas devem capturar o processo de revisão de conclusões. Isto é, conclusões tiradas devem posteriormente serem verificadas e rejeitadas, se for o caso.

O raciocínio que se faz com conhecimento incompleto é não-monotônico. Deve-se tirar conclusões que, em base de posteriores informações, podem ser rejeitadas. Apresentamos

em seguida alguns problemas cuja solução envolve raciocínio não-monotônico.

O primeiro problema é o conhecido problema dos missionários e canibais. Este exemplo foi retirado de [McCARTHY, 1980].

"Três missionários e três canibais estão à margem de um rio e querem atravessá-lo. Apenas um barco com dois lugares está disponível e o número de canibais não pode ser maior do que o de missionários em nenhuma das margens senão os missionários são comidos. Como atravessar o rio?"

Obviamente o problema é proposto para que se encontre uma estratégia de atravessar o rio através de idas e vindas do barco sem que nenhum missionário seja comido. Imagine propor o problema a alguém que após pensar um pouco sugerisse que seguissem em frente um quilômetro e atravessassem a ponte.

- "Que ponte?", você diz, "nenhuma ponte foi mencionada."

- "Bem. Você também não disse que não existia uma ponte."

Então você modifica o problema para excluir pontes e propõe-no novamente, e a coisa se repete para helicópteros, navios, etc. Com o propósito de fazê-lo entender o "espírito do problema" você mostra-lhe a solução e ele retruca sua solução afirmando que o barco tem um furo ou faltam re-



mos. Depois que você retifica a omissão ele diz que um monstro marinho emerge e destrói o barco. Você busca uma maneira de acabar com isso de uma vez. Então você afirma no problema que não há outra maneira de se atravessar o rio que não seja usando o barco e que nada errado acontece com o barco. Uma pessoa com bom senso não precisaria de tantas restrições para tentar uma solução adequada para o problema.

A circunscrição é uma regra de inferência que formaliza este tipo de raciocínio. Ela permite conjecturar que não existem objetos relevantes em certas categorias, exceto aqueles cujas existências seguem da proposição do problema e do conhecimento do senso comum. A circunscrição permitiria concluir que não existem pontes ou helicópteros mas não que não existem remos. É parte do senso comum que um barco pode ser usado para atravessar um rio, exceto exista alguma coisa errada com ele ou alguma coisa impedindo seu uso. Se os fatos não requerem que alguma coisa impeça o uso do barco, a circunscrição gerará a conjectura que nada existe impedindo seu uso. O preço que pagamos é o de ter que introduzir como entidades em nossa linguagem as "algumas coisas" que possam impedir o uso do barco.

Se o enunciado do problema fosse estendido de modo a mencionar uma ponte não mais a circunscrição do problema poderia permitir a conclusão de que não existem pontes, isto é, uma conclusão tirada de uma coleção de fatos, pode não ser tirada de uma coleção maior. Assim a circunscrição como regra de inferência é não-monotônica.



O segundo exemplo diz respeito ao "problema de quadro"\*, [RAPHAEL, 1971], que surge na representação de situações dinâmicas. O problema origina-se na necessidade de representar-se os aspectos do mundo que permanecem invariantes sob certas mudanças de estado. Por exemplo, mover um objeto em particular, ou acender a luz, não muda a cor de nenhum objeto na situação. Ao representar-se essas situações em lógica clássica é necessário explicitamente representar todos os invariantes sob todas mudanças de estado. Estes são referidos como os "axiomas de quadro" para a situação sendo modelada. Assim, para representar o fato que pintar um objeto não altera as locações de objetos, é requerido no cálculo situacional de [McCARTHY & HAYES, 1969], um "axioma de quadro" como o abaixo:

$$\forall xyzsc \text{ LOCAÇÃO } (x,y,s) \rightarrow \text{ LOCAÇÃO } (x,y, \text{ pintar } (z,c,s))$$

onde  $s$  é um variável de estado,  $x$  e  $z$  são objetos,  $c$  é uma cor, e  $y$  é alguma locação.

O problema é que em geral um vasto número de tais axiomas é requerido. Por exemplo, locações de objetos permanecem invariantes quando luzes são ligadas, quando chove, quando alguém fala, etc. Existe uma grande dificuldade em se achar um conjunto de "axiomas de quadro" adequado para uma situação.

---

\* Em inglês, "frame problem".

Uma solução para o "problema de quadro" é uma representação da situação acoplada com regras de inferências apropriadas, tais que os "axiomas de quadro" não são nem explicitamente representados nem explicitamente usados em um raciocínio sobre a situação. Uma solução para o problema é assumir-se o esquema "default": "Toda ação (de mudança de estado) é assumida deixar toda relação inalterada exceto se for possível deduzir o contrário". Isto simplificaria consideravelmente a representação da situação. Com efeito, somente as informações positivas sobre a situação, no nosso exemplo as ações que alterariam as posições dos objetos, precisariam ser representadas explicitamente. Informações negativas, no nosso exemplo as ações que não mudam as posições dos objetos, não precisam ser representadas, são inferidas por "default". Desde que em geral a quantidade de conhecimento negativo é bem maior que a de conhecimento positivo, existe uma vantagem representacional e computacional considerável em se trabalhar com o "default". Por exemplo, no nosso caso não precisaríamos representar que acender as luzes não altera a posição dos objetos, nem tropejar, nem falar, etc. Desde que informações subseqüentes podem invalidar suposições anteriormente feitas, a inferência por "default" é não-monotônica.

Existem sistemas computacionais com propriedades não-monotônicas desde os primórdios da Inteligência Artificial. Em geral eles foram simplesmente implementados e utilizados sem se atentar para as implicações na lógica que estes siste

mas pudessem ter. Recentemente, na década de 80, houve um grande interesse em desenvolver-se sistemas formais para o raciocínio não-monotônico, originando as Lógicas Não-Monotônicas. No volume 13 do *Jornal de Inteligência Artificial*, de abril de 1980, especialmente dedicado às Lógicas Não-Monotônicas, encontram-se diferentes abordagens de raciocínio não-monotônico desenvolvidas em IA. WINOGRAD apresenta e analisa as propriedades de diversos sistemas não-monotônicos; McCARTHY apresenta a circunscrição; McDERMOTT & DOYLE desenvolvem uma lógica não-monotônica estendendo a lógica de primeira ordem com um operador modal para a consistência; REITER desenvolve uma formalização para o raciocínio de "default", e WEYHRAUCH propõe um sistema no qual uma teoria de primeira ordem é associada com uma "meta-teoria" a qual por sua vez é uma teoria de primeira ordem no seu domínio. O nosso trabalho é uma apresentação e estudo das propriedades das semânticas restritivas (não-monotônicas), e das inferências não-monotônicas de circunscrição (McCARTHY) e "default" (REITER).

O capítulo 2 é uma breve apresentação da lógica de primeira ordem necessária à compreensão do restante da tese. Vemos, ainda, a Aritmética que exerce um papel importante em todo decorrer do trabalho.

No capítulo 3 apresentamos as Semânticas Restritivas: Minimal, Nomeável e Inicial. É feito um estudo das pro



priedades de cada uma destas semânticas. Vemos também uma lógica nomeável em particular: a W-Lógica.

No capítulo 4 apresentam-se as inferências não-monotônicas: circunscrição e "default". As propriedades destas inferências são analisadas e relaciona-se a Semântica Minimal com a circunscrição. Mostra-se que a circunscrição é correta em relação à Semântica Minimal mas não é completa.

Finalmente, no capítulo 5 apresentamos nossas conclusões.



## 2 - A LÓGICA DE PRIMEIRA ORDEM (LÓGICA CLÁSSICA)

Neste capítulo fazemos uma breve exposição da lógica de primeira ordem (ou lógica clássica, ou cálculo de predicados de primeira ordem). O intuito deste capítulo é capacitar o leitor não familiarizado com a lógica matemática, à compreensão do restante do trabalho. Para um maior aprofundamento do assunto recomendamos ao leitor que consulte uma bibliografia especializada. Recomendamos especialmente os livros de [ENDERTON, 1972] e [CHANG-KEISLER, 1973].

Inicialmente apresentamos a linguagem da lógica de primeira ordem, em seguida damos o conceito de estrutura e interpretação de modelos em uma linguagem. Vemos então alguns conceitos relacionados com a semântica da lógica de primeira ordem e então apresentamos o sistema dedutivo desta lógica. São enunciados os principais teoremas da lógica clássica: completude, compacidade, enumerabilidade e Löwenheim-Skolem. Definimos então as teorias de primeira ordem e finalmente apresentamos alguns tópicos especiais que utilizaremos no decorrer do trabalho: as funções de Skolem, cláusulas e uma teoria muito especial: a aritmética.

### 2.1 - A Linguagem

O alfabeto das linguagens de primeira ordem consiste de uma infinidade de símbolos a saber:

### 2.1.1 - Símbolos lógicos

1. Parênteses ( , ).
2. Símbolos Conectivos Sentenciais  $\rightarrow$ ,  $\neg$ .
3. Variáveis (uma para cada número natural):  $V_1$ ,  
 $V_2$ ,  $\dots$ .
4. Quantificador Universal:  $\forall$ .
5. Símbolo de Igualdade (opcional):  $\approx$ .

### 2.1.2 - Parâmetros

1. Símbolos Predicados: Para cada natural  $n$  algum conjunto (possivelmente vazio) de símbolos chamados símbolos predicados  $n$ -ários.
2. Símbolos Funcionais: Para cada natural  $n$ , algum conjunto (possivelmente vazio) de símbolos chamados símbolos funcionais  $n$ -ários.
3. Símbolos Constantes: Algum conjunto (possivelmente vazio) de símbolos.

Uma linguagem de primeira ordem em particular é dada listando-se os parâmetros da linguagem. Assim, uma linguagem denotada por  $L$ , é uma tripla de símbolos predicados, símbolos funcionais e símbolos constantes. Nós denotamos uma linguagem  $L$  por:

$$L = \langle R, F, C \rangle.$$

onde,  $R$  é um conjunto de símbolos predicados;  
 $F$  é um conjunto de símbolos funcionais;  
 $C$  é um conjunto de símbolos constantes.

Cada símbolo predicado  $P \in R$  é uma relação  $n$ -ária para algum inteiro  $n \geq 1$ , dependendo de  $P$ . Cada símbolo funcional  $f \in F$  é uma função  $m$ -ária para algum inteiro  $m \geq 1$ , dependendo de  $f$ . Os símbolos constantes  $c \in C$  podem ser vistos como funções 0-árias.

Os termos da linguagem são definidos recursivamente como abaixo:

- (i) Uma variável é um termo.
- (ii) Um símbolo constante é um termo.
- (iii) Se  $f$  é um símbolo funcional  $m$ -ário e  $t_1, \dots, t_m$  são termos, então  $f(t_1, \dots, t_m)$  é um termo.
- (iv) Uma seqüência de símbolos é um termo sss pode ser mostrado ser um termo através de um número finito de aplicações de (i) - (iii).

As fórmulas atômicas de  $L$  são as expressões da forma abaixo:

- (i)  $t_1 \approx t_2$  é uma fórmula atômica onde  $t_1$  e  $t_2$  são termos de  $L$ .
- (ii) Se  $P$  é um símbolo predicado  $n$ -ário e  $t_1, \dots, t_n$  são termos, então  $P(t_1, \dots, t_n)$  é uma fórmula atômica.



Finalmente, as fórmulas de  $L$  são definidas recursivamente como abaixo:

- (i) Uma fórmula atômica é uma fórmula.
- (ii) Se  $\varphi$  e  $\psi$  são fórmulas, então  $(\varphi \vee \psi)$  e  $(\neg \varphi)$  são fórmulas.
- (iii) Se  $V_1$  é uma variável e  $\varphi$  é uma fórmula, então  $\forall V_1. \varphi$  é uma fórmula.
- (iv) Uma seqüência de símbolos é uma fórmula sss pode ser mostrado ser uma fórmula através de um número finito de aplicações de (i) - (iii).

Observações 2.1:

- 1) Admitiremos como variável de uma linguagem as letras minúsculas:  $x, y, z, t, u, v, \omega, \dots$
- 2) Os outros conectivos sentenciais:  $\wedge$  (e),  $\vee$  (ou),  $\leftrightarrow$  (equivalente) podem ser introduzidos como abreviações (Ver [CHANG-KEISLER, 1973, p. 6]).
- 3) O quantificador existencial é introduzido como uma abreviação definida por:

$$\exists v. \varphi \text{ para } \neg (\forall v. \neg \varphi)$$

- 4) Não definiremos as noções de variáveis livres e ligadas, e substituição de variáveis por termos e/ou variáveis. O leitor encontrará as definições em [ENDERTON, 1972].

- 5) Sentenças são definidas como as fórmulas de uma linguagem sem variáveis livres (Ver [ENDERTON, 1972, p. 75]).
- 6) Se os símbolos de uma linguagem são os padrões como + para a adição,  $\leq$  para a relação de ordem, etc., simplesmente escrevemos

$$L = \langle \leq \rangle, \quad L = \langle +, ., <, 0, 1 \rangle, \text{ etc.}$$

A cardinalidade de uma linguagem  $L$ , denotada por  $\|L\|$ , é definida como

$$\|L\| = \omega \cup |L|$$

onde,  $\omega$  é a cardinalidade enumerável e  $|L|$  é a cardinalidade do conjunto de símbolos parâmetros de  $L$ .

Se  $L$  e  $L'$  são linguagens tais que  $L \subset L'$ , isto é,  $L'$  tem todos os símbolos parâmetros de  $L$  e mais alguns, então  $L'$  é uma expansão de  $L$ , e  $L$  é uma redução de  $L'$ .

## 2.2 - Modelos de uma Linguagem

Um modelo (ou uma estrutura) para uma linguagem  $L$ ,  $W$ , consiste de um conjunto universo ou domínio  $|W|$ , não vazio, no qual, cada símbolo predicado  $n$ -ário  $P$  de  $L$ , corresponde a uma relação  $n$ -ária  $P^W \subset |W|^n$ ; cada símbolo fun-

cional  $m$ -ário  $f$  de  $L$ , corresponde a função  $m$ -ária  $f^W : |W|^m \rightarrow |W|$ ; e cada símbolo constante  $c$  de  $L$ , corresponde a um elemento  $c^W$  de  $|W|$ . A correspondência é dada por uma função de "interpretação"  $\tau$  que transforma os parâmetros de  $L$  em relações, funções e constantes em  $|W|$  apropriadas. Um modelo  $W$  para  $L$  é um par  $\langle |W|, \tau \rangle$ , e escrevemos,

$$W = \langle |W|, \tau \rangle.$$

Quando  $L = \langle p_0, \dots, p_n, f_0, \dots, f_m, c_0, \dots, c_k \rangle$ .

Escrevemos  $W = \langle |W|, p_0^W, \dots, p_n^W, f_0^W, \dots, f_m^W, c_0^W, \dots, c_k^W \rangle$

para os modelos  $W$  de  $L$ .

O cardinal do modelo  $W$  é o cardinal de  $|W|$ .

Dois modelos  $W$  e  $W'$  para  $L$  são isomorfos sss existe uma função injetiva  $h : |W| \rightarrow |W'|$  satisfazendo:

- (i) Para cada relação  $n$ -ária  $p^W$  de  $W$  e a correspondente relação  $p^{W'}$  de  $W'$ ,

$$p^W(x_1, \dots, x_n) \text{ sss } p^{W'}(h(x_1) \dots h(x_n))$$

para todos  $x_1, \dots, x_n$  em  $|W|$ .

- (ii) Para toda função  $m$ -ária  $f^W$  de  $W$  e a correspondente função  $f^{W'}$  de  $W'$



$$h(f^W(x_1, \dots, x_m)) = f^{W'}(h(x_1) \dots h(x_m)),$$

para todos  $x_1, \dots, x_m$  em  $|W|$ .

- (iii) Para toda constante  $c^W$  de  $W$ , e a correspondente constante  $c^{W'}$  de  $W'$ ,

$$h(c^W) = c^{W'}.$$

Se um modelo  $W$  é isomorfo a um modelo  $W'$  denotamos por  $W \approx W'$ .

Um modelo  $B$  é um submodelo de um modelo  $W$  se  $|B| \subset |W|$  e:

- (i) Cada relação  $n$ -ária  $p^B$  de  $B$  é a restrição para  $|B|$  da correspondente relação  $p^W$  de  $W$ , isto é,  $p^B = p^W \cap (|B|)^n$ .
- (ii) Cada função  $m$ -ária  $f^B$  de  $B$  é a restrição para  $|B|$  da correspondente função  $f^W$  de  $W$ , isto é,  $f^B = f^W|_{|B|^m}$ .
- (iii) Cada constante de  $B$  é a correspondente constante de  $W$ , isto é,  $c^B = c^W$ .

Usamos  $B \subset W$  para denotar que  $B$  é um submodelo de  $W$ . Dize

mos que  $W$  é uma extensão de  $B$  se  $B$  é um submodelo de  $W$ .

Observação 2.2. Se  $P$  é um predicado  $n$ -ário em uma linguagem  $L$  e  $W$  é um modelo de  $L$ , dizemos que a extensão do predicado  $P$  em  $W$  é o conjunto das  $n$ -uplas de  $|W|$  que pertencem à interpretação de  $P$  em  $W$ , isto é,

$$\text{Extensão de } P \text{ em } W = \{(x_1, \dots, x_n) \in P^W\}$$

### 2.3. Semântica da lógica de primeira ordem

Entendemos por semântica de uma lógica as noções relacionadas com o valor de verdade de uma fórmula. Em lógica de primeira ordem, decidir-se o valor de verdade de uma fórmula em um modelo não é um problema simples. Por exemplo, não existe maneira de decidir-se se uma dada sentença de  $L = \langle +, \cdot, S, 0 \rangle$  é verdadeira ou falsa no modelo padrão da aritmética  $W = \langle N, +, \cdot, S, 0 \rangle$  (onde  $S$  é a função sucessor). Para um estudo detalhado sobre o problema do valor de verdade em lógica de primeira ordem indicamos [CHANG-KEISLER, 1973, p. 26]. Aqui nós não definiremos a noção de valor de verdade.

Usamos a notação  $W \models \sigma$  para denotar que a sentença  $\sigma$  é verdadeira no modelo  $W$  de  $L$ , ou que,  $\sigma$  é válida em  $W$ ;  $W$  satisfaz  $\sigma$ ,  $\sigma$  é satisfeita em  $W$ ,  $W$  é um modelo de  $\sigma$ . Quan

do não é o caso que  $\sigma$  é válida em  $W$ , dizemos que  $\sigma$  é  $\sigma$  falsa em  $W$ , ou que  $\sigma$  falha em  $W$ , ou que  $W$  é um modelo de  $\neg \sigma$ . Dado um conjunto  $\Sigma$  de sentenças, dizemos que  $W$  é um modelo de  $\Sigma$  sss  $W$  é um modelo para cada  $\sigma \in \Sigma$ . Denotamos por  $W \models \Sigma$ . Uma sentença  $\sigma$  que é válida em todo modelo de  $L$  é chamada válida. Denotamos por  $\models \sigma$ . Uma sentença, ou um conjunto de sentenças é satisfatível sss tem no mínimo um modelo.

Uma sentença  $\varphi$  é conseqüência lógica de uma outra sentença  $\sigma$ , em símbolos  $\sigma \models \varphi$ , sss todo modelo de  $\sigma$  é um modelo de  $\varphi$ . Uma sentença  $\varphi$  é conseqüência lógica de um conjunto de sentenças  $\Sigma$ , em símbolos,  $\Sigma \models \varphi$ , sss todo modelo de  $\Sigma$  é um modelo de  $\varphi$ . Duas sentenças  $\sigma$  e  $\varphi$  são logicamente equivalentes, em símbolos  $\sigma \models \varphi$  e  $\varphi \models \sigma$ .

Dois modelos  $W$  e  $B$  são elementarmente equivalentes sss toda sentença que é verdadeira em  $W$  é verdadeira em  $B$ , e vice-versa.

$W$  é um submodelo elementar de  $B$  sss  $W$  é um submodelo de  $B$  e  $W$  é elementarmente equivalente a  $B$ . Denotaremos por  $W < B$ . Nesse caso  $B$  é uma extensão elementar de  $W$ .



## 2.4 - Sintaxe da Lógica de Primeira Ordem

A sintaxe de uma lógica compreende a "gramática" da lógica (a definição de fórmula, por exemplo, é uma noção sintática) e as noções relacionadas com o sistema dedutivo da lógica. O sistema dedutivo consiste de um conjunto de fórmulas chamadas de axiomas lógicos, e um conjunto de regras de inferências. O sistema dedutivo não é único. Há infinitas maneiras de se escolher os axiomas lógicos e as regras de inferência ([CHANG-KEISLER, 1973, p. 24] e [ENDERTON, 1972, p. 101] apresentam sistemas dedutivos distintos para a lógica de primeira ordem).

Seja  $\Sigma$  um conjunto de sentenças e  $\varphi$  uma sentença, uma prova ou dedução de  $\varphi$  a partir de  $\Sigma$  é uma seqüência  $\langle \alpha_0, \dots, \alpha_n \rangle$  de fórmulas, tal que  $\alpha_n = \varphi$  e para cada  $i \leq n$  ou

- (i)  $\alpha_i$  está em  $\Gamma$ , ou
- (ii)  $\alpha_i$  é um axioma lógico, ou
- (iii)  $\alpha_i$  é inferida de um subconjunto de  $\{\alpha_0, \dots, \alpha_{i-1}\}$  a partir de alguma regra de inferência.

$\varphi$  é um teorema de  $\Sigma$ , denotado por  $\Sigma \vdash \varphi$ , se existe uma prova a partir de  $\Sigma$  de  $\varphi$ . Se  $\varphi$  é um teorema de  $\emptyset$ , dizemos que  $\varphi$  é um teorema da linguagem e denotamos por  $\vdash \varphi$ .  $\Sigma$  é inconsistente sss toda fórmula da linguagem é um teorema



de  $\Sigma$ . Caso contrário,  $\Sigma$  é consistente. Uma sentença  $\sigma$  é consistente sss  $\{\sigma\}$  é.  $\Sigma$  é maximal consistente em uma linguagem  $L$  sss  $\Sigma$  é consistente e nenhum conjunto de sentenças (de  $L$ ) propriamente contendo  $\Sigma$  é consistente.

## 2.5. Corretude e completude da Lógica de Primeira Ordem

Um sistema dedutivo para a lógica de primeira ordem pode ser arbitrariamente escolhido mas é importante que se escolha um correto e completo.

Os teoremas abaixo mostram que isto é possível. As demonstrações são encontradas em (CHANG-KIESLER, 1973] e [ENDERTON, 1972].

Teorema da corretude. Existe um sistema dedutivo para a lógica de primeira ordem tal que:

- a) Se  $\Sigma \vdash \varphi$ , então  $\Sigma \models \varphi$  ou equivalentemente,
- b) Se  $\Sigma$  é satisfatível então  $\Sigma$  é consistente.

Onde  $\Sigma$  é um conjunto de sentenças e  $\varphi$  é uma sentença.

Teorema da completude (GÖDEL, 1930). Existe um sistema dedutivo para a lógica de primeira ordem tal que:

- a) Se  $\Sigma \models \varphi$ , então  $\Sigma \vdash \varphi$ , ou equivalentemente,  
 b) Se  $\Sigma$  é consistente, então  $\Sigma$  é satisfatível.

Onde  $\Sigma$  é um conjunto de sentenças e  $\varphi$  é uma sentença.

## 2.6 - Teoremas da Compacidade, Enumerabilidade e Löwenheim-Skolem

### Teorema da compacidade

- (i) Se  $\Sigma \models \varphi$ , então para algum conjunto finito  $\Sigma_0 \subset \Sigma$ ,  
 $\Sigma_0 \models \varphi$ , ou equivalentemente,  
 (ii) Se um conjunto de sentenças é finitamente satisfatível (todo subconjunto finito é satisfatível), então o conjunto é satisfatível.

Onde  $\Sigma$  é um conjunto de sentenças (possivelmente infinito) e  $\varphi$  é uma sentença.

### Teorema de Enumerabilidade

Em uma linguagem razoável, o conjunto de sentenças válidas é recursivamente enumerável.

Por uma linguagem razoável entendemos uma na qual o conjunto de parâmetros é recursivamente enumerável e tal que as duas relações

$$\{(P,n) : P \text{ é um símbolo predicado } n\text{-ário}\}$$

$$\{(f,m) : f \text{ é um símbolo funcional } m\text{-ário}\}$$

são decidíveis. Uma linguagem com um número finito de parâmetros é razoável.

Corolário. Seja  $\Sigma$  um conjunto decidível de fórmulas em uma linguagem razoável.

- (i) o conjunto dos teoremas de  $\Sigma$  é recursivamente enumerável.
- (ii) O conjunto das conseqüências lógicas de  $\Sigma$  é recursivamente enumerável.

As partes (i) e (ii) referem-se ao mesmo conjunto (pela completude). Este corolário inclui o próprio teorema da enumerabilidade, no qual  $\Sigma = \emptyset$ .

#### Teorema de Löwenheim-Skolem

- (i) Seja  $\Sigma$  um conjunto satisfatível de fórmulas em uma linguagem enumerável. Então  $\Sigma$  pode ser satisfeito em algum modelo enumerável.

(ii) Seja  $\Sigma$  um conjunto satisfatível de fórmulas em uma linguagem de cardinalidade  $\chi$ . Então  $\Sigma$  é satisfeito em algum modelo de cardinalidade  $\leq \chi$ .

((i) é um caso particular de (ii) em que  $\chi = \omega$ ).

### Teorema de Löwenheim-Skolem-Tarski

Seja  $\Sigma$  um conjunto de fórmulas em uma linguagem de cardinalidade  $\chi$  e suponha-se que  $\Sigma$  pode ser satisfeito em algum modelo infinito. Então, para cada cardinal  $\lambda \geq \chi$ , exis um modelo de cardinalidade  $\lambda$  no qual  $\Sigma$  é satisfeito.

Estes teoremas por serem teoremas fundamentais da lógica de primeira ordem são demonstrados em qualquer livro de lógica matemática, particularmente em [ENDERTON, 1972]. [CHANG-KEISLER, 1973] apresenta três demonstrações para compactidade e para Löwenheim-Skolem.

### 2.7 - Teoria de Primeira Ordem

Uma teoria é um conjunto de sentenças fechado em relação a implicação lógica. Isto é,  $T$  é uma teoria sss  $T$  é um conjunto de sentenças tal que para qualquer sentença  $\sigma$  da linguagem,

se  $T \models \sigma$ , então  $\sigma \in T$ .



Desde que teorias são conjuntos de sentenças, podemos aplicar os conceitos de

- modelo de uma teoria,
- teoria consistente,
- teoria satisfatível.

Um conjunto de axiomas de uma teoria  $T$  é um conjunto de sentenças cujas conseqüências lógicas são  $T$ .

Para uma classe  $C$  de modelos de uma linguagem, a teoria de  $C$ ,  $\text{Th } C$ , é definida por

$$\text{Th } C = \{ \sigma; \sigma \text{ é verdadeira em todo membro de } C \}$$

Para um conjunto  $\Sigma$  de sentenças, seja  $\text{Mod } \Sigma$  a classe de todos modelos de  $\Sigma$ .  $\text{Th Mod } \Sigma$  é o conjunto de todas sentenças logicamente implicadas por  $\Sigma$ . Chame este conjunto de conjunto das conseqüências de  $\Sigma$ ,  $\text{Cn}(\Sigma)$ . Assim,

$$\text{Cn}(\Sigma) = \{ \sigma; \Sigma \models \sigma \} = \text{Th Mod } \Sigma.$$

(Usualmente abreviamos  $\text{Th Mod } \Sigma$  para  $\text{Th}(\Sigma)$  assim,  $\text{Cn}(\Sigma) = \text{Th}(\Sigma)$ .)

Seja  $\text{th}(\Sigma)$  o conjunto dos teoremas de  $\Sigma$ , isto é,

$$\text{th}(\Sigma) = \{ \sigma; \Sigma \vdash \sigma \}$$

Assim, pela completude, fórmula lógica válida em  $\Sigma$

forma normal prenexa.

$$\text{Th}(\Sigma) = \text{th}(\Sigma).$$

Uma teoria  $T$  é completa sss é maximal consistente, isto é,  $\sigma \in T$  ou  $\neg \sigma \in T$ .

Se  $T$  e  $T'$  são teorias e  $T \subset T'$ , então dizemos que  $T$  é uma subteoria de  $T'$  e que  $T'$  é uma extensão ou superteor  
ria de  $T$ .

## 2.8 - Tópicos especiais

### 2.8.1 - Funções de Skolem

A forma normal prenexa de uma fórmula é uma na qual a matriz não contém quantificadores e o prefixo é uma seqüên-  
cia de quantificadores. Isto é, para algum  $n \geq 0$

$$Q_{1x_1}, \dots, Q_{nx_n} \alpha,$$

onde  $Q_i$  são quantificadores e  $\alpha$  é uma fórmula sem quanti-  
ficadores.

Pode-se provar ([ENDERTON, 1972, p. 150]) que para toda fórm

mula podemos achar uma fórmula logicamente equivalente na forma normal prenexa.

A forma normal conjuntiva de uma fórmula é uma seqüência de conjunções em que os "conjuntos" (as partes da conjunção) são seqüência de disjunções, isto é,

$$(\alpha_1 \vee \dots \vee \alpha_n) \wedge \dots \wedge (\beta_1 \vee \dots \vee \beta_m)$$

onde  $\alpha_i, \beta_i$  são ou fórmulas atômicas ou negações de fórmulas atômicas.

Pode-se provar ([CHANG-LEE, 1973]) que para toda fórmula existe uma fórmula logicamente equivalente em forma normal conjuntiva.

Seja uma fórmula  $F$  já em forma normal prenexa  $Q_1 x_1 \dots Q_n x_n M$ , onde  $M$  é uma forma normal conjuntiva. Suponha  $Q_r$  um quantificador existencial no prefixo  $Q_1 x_1 \dots Q_n x_n$ ,  $1 \leq r \leq n$ . Se não aparece quantificadores universais antes que  $Q_r$ , escolhemos uma nova constante  $c$  diferente das outras constantes ocorrendo na linguagem de  $F$ , substituímos todos  $x_r$  que aparecem em  $M$  por  $c$ , e retiramos  $Q_r x_r$  do prefixo. Se  $Q_{s_1}, \dots, Q_{s_m}$  são todos os quantificadores universais que aparecem antes de  $Q_r$ ,  $1 \leq s_1 \leq s_2 \leq \dots \leq s_m < r$ , escolhermos um novo símbolo funcional  $f$  diferente dos da linguagem, substituímos todas  $x_r$  por  $f(x_{s_1}, x_{s_2}, \dots, x_{s_m})$ , e retiramos  $Q_r x_r$  do prefixo. Depois que o processo acima é aplicado a todos os quantificadores existentes no prefixo, a última fórmula obtida é a forma padrão de Skolem. As constantes e funções introduzidas são chamadas de funções de

Skolem.

Exemplo: Se  $\alpha = \exists x \forall y \forall z \exists m \forall v \exists w P(x, y, z, m, v, w)$

a forma padrão de Skolem de  $\alpha$  é

$$\forall y \forall z \forall v P(a, y, z, f(y, z), v, g(y, z, v))$$

onde  $a, f$  e  $g$  não aparecem na linguagem de  $\alpha$ .

Uma cláusula é uma disjunção de literais (fórmulas atômicas ou negação de fórmula atômicas). Um conjunto  $S$  de cláusulas é visto como a conjunção de todas cláusulas em  $S$ , onde toda variável em  $S$  é considerada ligada a um quantificador universal. Assim, uma forma padrão de Skolem pode ser representada por um conjunto de cláusulas.

A linguagem  $L$  expandida pelas funções de Skolem é chamada uma expansão de Skolem de  $L$ .

Pode-se provar ([CHANG-LEE, 1973, p. 48]) que se  $S$  é um conjunto de cláusulas que representa a forma padrão de Skolem de uma fórmula  $F$ , então  $F$  é inconsistente sss  $S$  é inconsistente.

Um tipo especial de conjunto de cláusulas são as que têm no máximo um literal positivo (sentença atômica) por cláusula, são as chamadas "cláusulas de Hörn". Existem três tipos de cláusula de Hörn distintos:



- (1)  $\square$  a cláusula vazia, não contendo literais e denotando o valor "falso".
- (2)  $\bar{B}_1 \vee \dots \vee \bar{B}_n$  uma cláusula consistindo de nenhum literal positivo e  $n \geq 1$  literais negativos (negação de uma sentença atômica).
- (3)  $A \vee \bar{B}_1 \vee \dots \vee \bar{B}_n$  uma cláusula consistindo de exatamente um literal positivo e  $n \geq 0$  literais negativos.

### 2.8.2 - A Aritmética

Seja  $L = \langle +, ., S, 0 \rangle$  e seja  $W = \langle N, +, ., S, 0 \rangle$  um modelo para  $L$ , onde  $+$ ,  $.$ ,  $0$  têm as interpretações ordinárias e  $S$  é a função sucessor. A aritmética é a teoria do modelo  $W$  ( $\text{Th } W$ ), ou seja, o conjunto das fórmulas válidas neste modelo.  $W$  é dito o modelo padrão da aritmética.

GÖDEL em 1931 demonstrou que a aritmética é incompleta e indecidível.

Incompletude da aritmética. Não existe um algoritmo por meio do qual podemos derivar o conjunto de todas as sentenças aritméticas verdadeiras.

Indecidibilidade da aritmética. Não existe um algoritmo por meio do qual podemos decidir (em um número finito de passos)

para toda sentença aritmética, se a sentença é verdadeira ou falsa.

Uma demonstração destes resultados pode ser encontrada em [ENDERTON, 1972].

Muitas foram as tentativas de se axiomatizar aritmética, a mais notável é a de Peano que apresentamos abaixo.

Os axiomas de Peano para a aritmética são os seguintes:

- (1)  $\forall x \quad Sx \neq 0$
- (2)  $\forall xy \quad (Sx \approx Sy \rightarrow x \approx y)$
- (3)  $\forall x \quad x + 0 \approx x$
- (4)  $\forall xy \quad x + Sy \approx S(x + y)$
- (5)  $\forall x \quad x \cdot 0 \approx 0$
- (6)  $\forall xy \quad x \cdot Sy \approx (x \cdot y) + x$

e finalmente, para cada fórmula  $\varphi(x)$  onde  $x$  ocorre em  $\varphi$  o axioma:

$$(7\varphi) \quad \varphi(0) \wedge (\forall x(\varphi(x) \rightarrow \varphi(Sx))) \rightarrow \forall x \varphi(x)$$

Axioma (1) afirma que "0 não tem predecessor" (2) afirma que  $S$  é injetiva, (3) e (4) são a definição recursiva da soma (+) em termos de zero (0) e sucessor (S). (5) e (6) são a definição recursiva do produto em termos de zero (0), sucessor (S) e soma (+). Todos os axiomas (7 $\varphi$ ), um para cada  $\varphi$ , é chamado o "axioma esquema da indução". Este axioma

permite para toda propriedade a qual é definida por uma fórmula aritmética a aplicação do processo de indução.

A aritmética de Peano é a teoria dos axiomas de Peano, isto é, o conjunto de todas conseqüências lógicas destes axiomas. A aritmética de Peano é claramente enumerável. O teorema da incompletude da aritmética mostra que não é completa. Isto é, a aritmética é uma extensão própria da aritmética de Peano.

CHURCH mostrou que a aritmética de Peano é indecidível (não é o mesmo que a indecidibilidade da aritmética uma vez que a aritmética de Peano não coincide com a aritmética). ROSSER mostrou ainda mais: a indecidibilidade essencial da aritmética (nome este dado por TARSKI à propriedade pela qual não somente a aritmética de Peano, mas toda sua extensão consistente é indecidível).



### 3 - SEMÂNTICAS RESTRITIVAS

O objetivo deste capítulo é apresentar algumas semânticas restritivas. Focalizaremos três maneiras de se restringir a semântica: a Semântica Minimal, a Semântica Nomeável e a Semântica Inicial. Como a restrição à semântica pertença ao escopo da lógica, as semânticas restritivas têm despertado interesse em várias áreas do conhecimento, notadamente na Matemática, em Álgebra e Fundamentos. No nosso caso enfatizaremos os efeitos destas semânticas em Ciência da Computação e em particular em Inteligência Artificial.

A lógica é dividida em duas partes: semântica e a sintaxe. A semântica compreende a noção de valor de verdade e tudo a isto relacionado: as interpretações, os modelos, as validades, as conseqüências lógicas etc. A sintaxe abrange a "gramática" da lógica, se uma determinada expressão é fórmula ou não, o sistema dedutivo, os teoremas, consistência, a teoria de prova e deduções, etc.

Aquí vale um parênteses: a Lógica Clássica ou de primeira ordem (ou o cálculo de predicados), é o nosso padrão tudo o que fazemos é em relação a esta Lógica, inclusive todo nosso estudo de propriedades e teoremas é em relação a ela.



Como o próprio nome diz as semânticas restritivas agem de modo a restringirem a semântica da Lógica Clássica, no sentido que os modelos de um conjunto de sentenças é uma subclasse da classe dos modelos deste conjunto em Lógica Clássica. Na Semântica Minimal são os modelos minimais, na Nomeável os modelos nomeáveis e na Inicial, os modelos iniciais. O principal ponto é a noção de consequência lógica: em Lógica Clássica uma sentença  $\sigma$  é consequência lógica de um conjunto de sentenças  $\Sigma$ , e escreve-se  $\Sigma \models \sigma$ , sss todo modelo de  $\Sigma$  é também modelo de  $\sigma$ . Esta noção nas semânticas restritivas será parafraseada da seguinte maneira: Uma sentença  $\sigma$  é consequência lógica minimal (ou nomeável) (ou inicial) de um conjunto de sentenças, e escreve-se respectivamente  $\Sigma \models_m \sigma$  (ou  $\Sigma \models_n \sigma$ ) (ou  $\Sigma \models_i \sigma$ ), sss todo modelo minimal (ou nomeável) (ou inicial) é também modelo de  $\sigma$ .

O raciocínio restritivo aparece muitas vezes em matemática. Por exemplo, quando um matemático anuncia os seguintes axiomas:

$$(i) \quad \forall x . Sx \neq 0$$

$$(ii) \quad \forall x \forall y (Sx \approx Sy \rightarrow x \approx y)$$

$$(iii) \quad \forall y (y \neq 0 \rightarrow \exists x . y \approx Sx)$$

$$(iv.1) \quad \forall x . Sx \neq x$$

$$(iv.2) \quad \forall x . SSx \neq x$$

...

$$(iv.n) \quad \forall x . S^n x \neq x$$

...

Sua intenção (interpretando-se 0 como "zero" e S como a função sucessor) é caracterizar os números naturais. Em Lógica Clássica ele não consegue pois os números naturais realmente são modelo destes axiomas, mas não são os únicos. Qualquer estrutura que seja uma cópia dos números naturais seguido de um número arbitrário de cópias dos inteiros é modelo destes axiomas (Ver [ENDERTON, 1972, p. 178]). Não é possível argumentar que a inclusão de novos axiomas poderia excluir os modelos não desejados, uma vez que se pode provar que o conjunto de axiomas dado é completo. O único modelo restrito destes axiomas é realmente os naturais.

Um outro exemplo é a aritmética. Um desafio enfrentado pelos matemáticos deste século foi o de axiomatizar a aritmética usando a Lógica Clássica. Muitas foram as tentativas feitas todas com insucesso até que GÖDEL em 1931 pôs ponto final à questão quando provou o Teorema da Incompletude da Aritmética, provavelmente o resultado mais notável da matemática neste século. A mais notável axiomatização conseguida foi a de PEANO e, como não poderia deixar de ser, a "Aritmética de Peano" é um subconjunto próprio da aritmética padrão. Veremos mais adiante que a aritmética padrão é axiomatizável em semânticas restritas.

Inúmeras definições em Matemática trazem a condição de fechamento: "Nada mais faz parte de nossa definição", ou seja, os únicos elementos que pertencem à classe definida são aqueles que satisfazem explicitamente às propriedades enunciadas. Este é justamente o efeito das semânticas restritas



tivas, notadamente da semântica minimal. A condição de fechamento não é expressável em lógica de primeira ordem.

Em computação as utilizações das semânticas restritas são inúmeras. Em especificação de tipos de dados faz-se mister (Ver [PEQUENO, 1981]). Primeiro, modelos para tipo de dados são enumeráveis (de fato finitos se levarmos em consideração limitação de máquinas). Segundo, são nomeáveis. A única maneira possível de se obter novos valores é através das operações dadas aos valores iniciais (isto conforme veremos adiante corresponde a dar um nome a cada elemento do modelo). No entanto a restrição que leva a considerar apenas modelos nomeáveis não é suficiente. Basicamente estruturas de dados são conjuntos (finitos) de células de memória organizados de maneira a serem acessíveis desde um menor subconjunto, os pontos de entrada. Por exemplo considere

$$C \xrightarrow{F} a_1 \rightarrow \dots \rightarrow a_n \xrightarrow{F} d \mathcal{R}_F$$

onde a informação é armazenada em cada ponto  $a_i$  desta estrutura e  $C$  é o ponto de entrada. Para se acessar a informação em um ponto particular atravessa-se a estrutura começando-se em  $C$  e segue-se os  $F$ -ligamentos. Se  $C$  for denotado por um símbolo constante então os pontos acessíveis são aqueles de nome  $F \dots F(C)$ .

Tais estruturas de tamanho arbitrário, possivelmente junto com o "limite" finito

$$C \xrightarrow{F} a_1 \xrightarrow{F} \dots \xrightarrow{F} a_n \xrightarrow{F} \dots d_F$$

formam a classe das listas lineares singularmente ligadas.

Desejamos definir a relação de acessibilidade por ligações  $F$ ,  $ac$ , para esta estrutura como sendo o fecho reflexivo transitivo do grafo de  $F$ .

A sentença

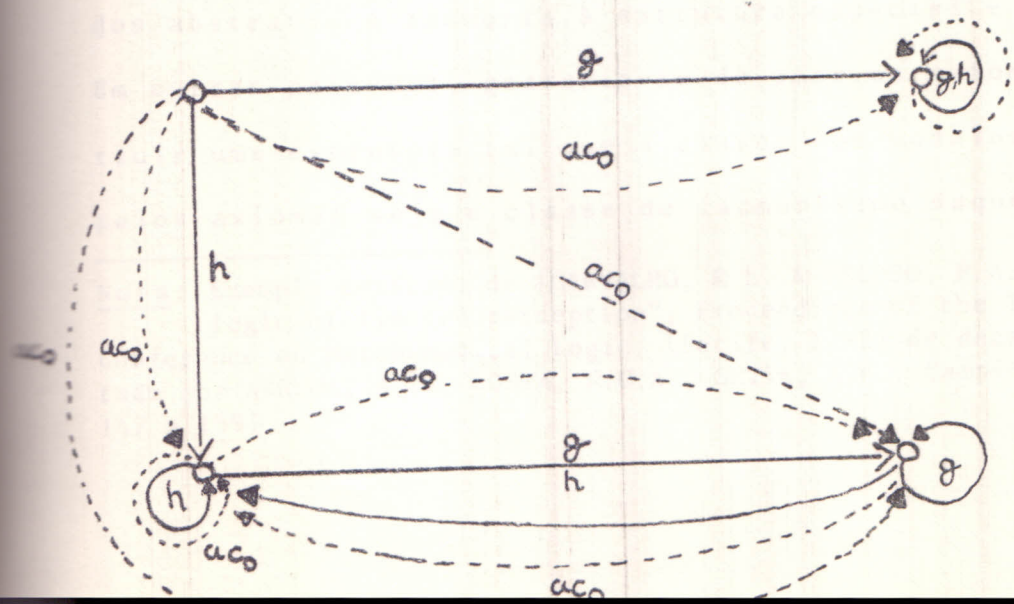
$$(1) \quad \forall x \forall y (ac(x,y) \leftrightarrow (x = C \wedge y = C) \vee (F(x) = y) \vee \exists z (ac(x,z) \wedge F(z) = y))$$

define  $ac$  com a pretendida interpretação.

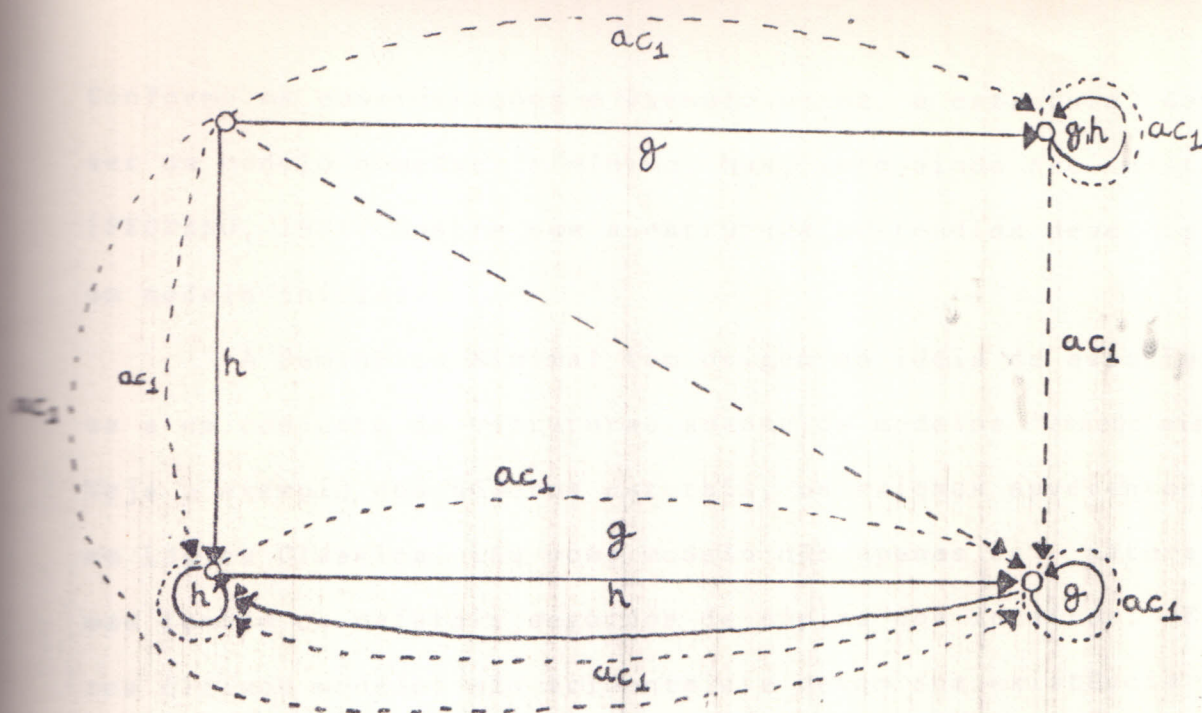
Agora considere o caso de termos duas funções  $g$  e  $h$  e o ponto de entrada  $C$ . A sentença natural análoga à (1) é

$$(2) \quad \forall x \forall y (ac(x,y) \leftrightarrow (x = C \wedge y = C) \vee (g(x) = y \vee h(x) = y) \vee \exists z (ac(x,z) \wedge (g(z) = y \vee h(z) = y)))$$

Mas as duas estruturas abaixo satisfazem (2)







A primeira interpretação é a que se deseja, onde  $ac_0 \subseteq ac_1$ .  $ac_0$  é a estrutura mínima satisfazendo a sentença (2).

O problema básico de caracterização de tipos de dados abstratos é como associar (univocamente) uma estrutura especial a um conjunto de axiomas, que pretende definir um certo tipo de dados abstratos, de tal forma que qualquer estrutura que reconhecemos como uma realização do tipo de dados abstratos é isomorfa à estrutura especial e vice-versa. Em outras palavras, gostaria-se de usar os axiomas para construir uma estrutura tal que a classe dos modelos pretendidos pelos axiomas seja a classe de isomorfismo daquela estrutura.

Nota: Exemplo retirado de [CARVALHO, R.L. & VELOSO, P.A.S., "Towards a logic of limited perception", Proceedings of the Third Brazilian Conference on Mathematical Logic. (Recife, 17-22 de dezembro, 1979) editado por ARRUDA, A.I., COSTA, N.C.A., SETTE, A.M., Campinas, 1980, pg. 147 a 159].

Conforme as considerações e exemplo acima, a estrutura deve ser um modelo nomeável e mínimo. Mas, isto ainda não basta. [PEQUENO, 1981] mostra que a estrutura pretendida deve ser um modelo inicial.

A Semântica Minimal tem origem na idéia de associar-se a um conjunto de estruturas apenas os modelos "essenciais". Veja o exemplo dos números naturais, os axiomas apresentados em Lógica Clássica, têm como modelo não apenas os naturais mas também os naturais seguidos de cópias dos inteiros. Estes últimos modelos são acidentais e devem sua existência ao fraco poder de caracterização da Lógica Clássica. Mas como caracterizar os modelos essenciais? Observe que, dado um conjunto de sentenças para que uma estrutura seja um modelo do conjunto, basta que possua os elementos exigidos pelas sentenças em suas relações, podendo ter outras. Assim, uma maneira de se capturar os modelos essenciais é exigir que o modelo possua apenas os elementos exigidos pelas sentenças em suas relações. Assim, se tivermos uma cadeia crescente de modelos (cada modelo contido no seguinte) para o conjunto de sentenças, o modelo essencial é aquele que tem as menores relações na cadeia. Isto é, nenhuma subestrutura desse modelo é modelo das sentenças. Esta é a definição de Modelo Minimal.

A Semântica Minimal tem um maior poder de caracterização do que a Lógica Clássica e é não-monotônica: se  $\sigma$  é consequência lógica minimal de um conjunto de sentenças  $A$ , e  $A \subseteq B$ , em geral,  $\sigma$  não é consequência lógica minimal de  $B$ .



A não-monotonicidade da Semântica Minimal deve-se ao fato de que se  $A$  e  $B$  são conjunto de sentenças e  $A \subseteq B$ , se  $M$  é um modelo minimal de  $B$  não necessariamente  $M$  é um modelo minimal de  $A$  pois  $M$  pode ter alguns elementos em suas relações exigidos pelas sentenças de  $B$  que não estão em  $A$ .

Modelos nomeáveis são estruturas nas quais todo elemento pertencente à estrutura tem "nome". Por "nome" de um elemento entendemos um termo da linguagem livre de variáveis que denota este elemento. Estes modelos têm a interessante característica de podermos referir-nos explicitamente a todos os seus elementos através de uma sentença. Em um modelo qualquer há alguns elementos que sabemos de suas existências, mas não podemos exibí-los. São referidos apenas através de sentenças quantificadas. Por exemplo, se a sentença  $\exists x . Px$  é verdadeira em um modelo arbitrário, sabemos que existe um elemento do domínio no predicado  $P$ , mas, em geral, não se sabe qual. Se o modelo for nomeável existirá um termo livre de variáveis  $\alpha$  pertencente a  $P$ . A Semântica Nomeável também tem um maior poder de caracterização em relação a Lógica Clássica, e é, ao contrário das Semânticas Minimal e Inicial, monotônica, embora apresente uma outra espécie de não-monotonicidade que chamamos de "não-conservação em extensões da linguagem".

Modelos iniciais são os mais restritivos e foram desenvolvidos em especificação de tipos de dados abstratos, [PEQUENO, 1981], com o intuito de resolver a questão da caracterização de um tipo de dados. É um caso particular dos ante-

riores no sentido que um modelo inicial é minimal e nomeável. Até mais que isso, é mínimo e nomeável. Mas não apenas isso. É uma interpretação de Herbrand mínima (no Universo de Herbrand) em que se excluem os elementos "iguais". Dois elementos são "iguais" se exibem as mesmas propriedades até o ponto em que a linguagem consegue expressar. A Semântica Inicial goza das propriedades das semânticas anteriores e de outras mais, inclusive a inicialidade.

Na seção 1 apresentamos a Semântica Minimal: Formalização, definições, exemplos e propriedades. E ainda os conceitos de modelo mínimo e modelo minimal local.

Na seção 2 damos o mesmo tratamento à Semântica Nomeável. Apresentamos as Interpretações de Herbrand e fazemos um estudo de uma lógica nomeável particular: a W-Lógica. W-Lógica é a lógica de primeira ordem da linguagem  $L = \langle +, \cdot, S, 0 \rangle$  em que adicionamos uma regra de inferência infinita, a Regra W, e aceitamos deduções infinitas.

Na seção 3 apresentamos a Semântica Inicial. Damos os conceitos de estrutura natural e igualdade natural e então definimos os modelos iniciais. Apresentamos algumas propriedades, inclusive a inicialidade.



### 3.1 - Semântica Minimal

#### 3.1.1 - Modelos Minimais

A definição abaixo afirma que um elemento (ou uma  $n$ -upla) pertence a uma relação de um modelo minimal de um conjunto de sentenças, se, e somente se, for exigido pelas sentenças.

Definição 3.1 - Seja  $\Gamma$  um conjunto de sentenças de uma linguagem  $L$  e  $M$  uma estrutura para a linguagem,  $M$  é um modelo minimal de  $\Gamma$ , se, e somente se:

- (1)  $M \models \sigma$  para cada sentença  $\sigma \in \Gamma$ ;
- (2) Para toda subestrutura própria  $N$  de  $M$ ,  $N \not\models \sigma$ , para algum  $\sigma \in \Gamma$ .

Exemplo 3.1. Seja  $\Gamma = \emptyset$ . Desde que toda estrutura é modelo de  $\emptyset$  os modelos minimais de  $\Gamma$  são todas estruturas cujo domínio consiste de apenas um elemento.

Exemplo 3.2. Seja  $L = \langle P, a, b \rangle$ , onde  $P$  é um símbolo predicado unário e,  $a$  e  $b$  são símbolos constantes. A sentença

$\Gamma = Pa \vee Pb \wedge a \neq b$  tem dois modelos minimais

$$M_1 = \langle |M_1| = \{a, b\}, P^{M_1} = \{a\}, a^{M_1} = a, b^{M_1} = b \rangle$$

$$M_2 = \langle |M_2| = \{a, b\}, P^{M_2} = \{b\}, a^{M_2} = a, b^{M_2} = b \rangle$$

Exemplo 3.3. Seja  $L = \langle S, 0 \rangle$ , onde  $S$  é um símbolo funcional unário e  $0$  é um símbolo constante. Seja  $\Gamma$  o conjunto das sentenças abaixo:

- (i)  $\forall x . S(x) \neq 0$
- (ii)  $\forall xy (S(x) \approx S(y) \rightarrow x \approx y)$
- (iii)  $\forall y (y \neq 0 \rightarrow \exists x . y \approx S(x))$
- (iv.1)  $\forall x . S(x) \neq x$
- (iv.2)  $\forall x . S(S(x)) \neq x$
- ...
- (iv.n)  $\forall x . S^n(x) \neq x$
- ...

$W = \langle N, S, 0 \rangle$  onde

$N$  é o conjunto dos números naturais;

$S$  é a função sucessor usual; e

$0$  é o número "zero"

é o único modelo minimal de  $\Gamma$ .

Em uma linguagem  $L = \langle P, F, C \rangle$  os modelos para um conjunto de sentenças de  $L$ , formam um reticulado em que um modelo "está contido" em um outro modelo  $B$  ( $A$  é uma subestrutura de  $A$ ) sss

1) se  $|A| \subseteq |B|$

Então,

- a)  $P^A$  é a restrição de  $P^B$  a  $|A|$ , para todo símbolo predicado  $P$  de  $L$ ;
- b)  $F^A$  é a restrição de  $F^B$  a  $|A|$ , para todo símbolo funcional  $F$  de  $L$ ;
- c)  $C^A = C^B$  para todo símbolo constante  $C$  de  $L$ .

2) Se  $|A| = |B|$

Então,

- a)  $C^A = C^B$ , para todo símbolo constante  $C$  de  $L$ ;
- b)  $F^A = F^B$ , para todo símbolo funcional  $F$  de  $L$ ;
- c)  $P^A \subseteq P^B$  para todo símbolo predicado  $P$  de  $L$ ;  
 ( $P^A \subseteq P^B$  sss o conjunto das  $n$ -uplas que pertencem a  $P^A$   
 (a extensão de  $P$  em  $A$ ), está contido no conjunto das  
 uplas que pertencem a  $P^B$  (a extensão de  $P$  em  $B$ ).

A figura seguinte ilustra o reticulado dos modelos de um conjunto de sentenças  $\Gamma$ .

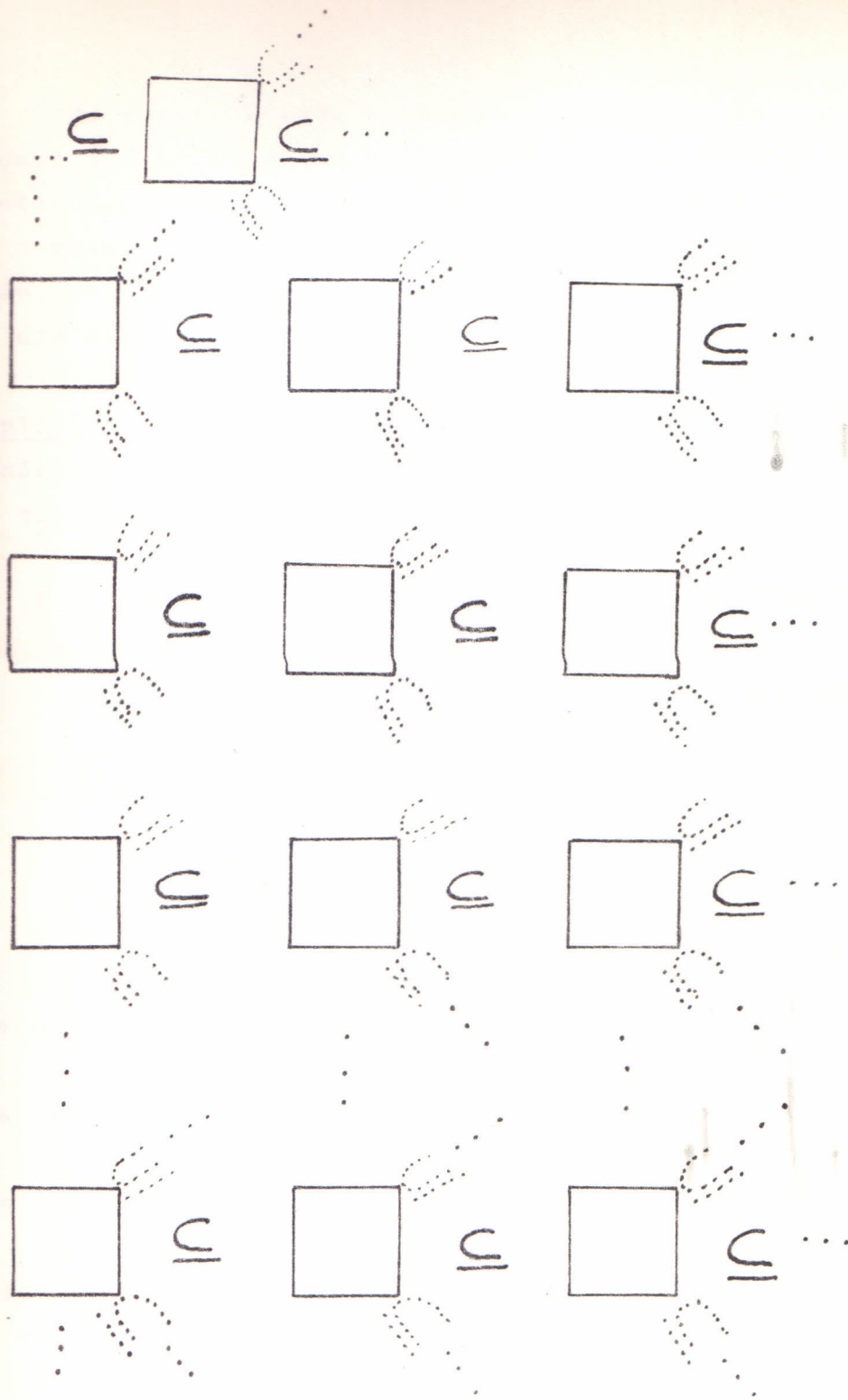


Figura 2



Na figura 2 cada retângulo representa um modelo do conjunto de sentenças  $\Gamma$ ; Modelos em cadeias horizontais diferentes podem ter um mesmo domínio ou não; o símbolo  $\subseteq$  indica a possibilidade de um modelo "estar contido" em outro ou não. Os modelos minimais de  $\Gamma$  são os modelos iniciais de cada uma das cadeias.

Exemplo 3.4 Seja  $L = \langle P, C \rangle$  onde  $P$  é um símbolo predicado unário e  $C$  é o conjunto dos símbolos constantes  $[a_1, a_2, \dots, a_n]$ .

Seja  $\Gamma = (Pa_1 \vee Pa_2 \vee \dots \vee Pa_n) \wedge (a_i \neq a_j, \text{ para todo } i, j = 1, \dots, n, i \neq j)$   
 onde a segunda parte da conjunção é uma abreviatura de  $a_1 \neq a_2 \wedge a_1 \neq a_3 \wedge \dots \wedge a_1 \neq a_n \wedge a_2 \neq a_3 \wedge a_2 \neq a_4 \wedge \dots \wedge a_2 \neq a_n$   
 $\vdots$   
 $\vdots$   
 $a_{n-1} \neq a_n$

Tome  $D = [a_1, a_2, \dots, a_n]$

e as estruturas  $M_i, i=1, \dots, n$ .

$M_i = \langle D, P^{M_i} = \{a_i\}, a_j^{M_i} = a_j, j=1, \dots, n \rangle$

Então  $M_i, i=1, \dots, n$  são os modelos minimais de  $\Gamma$ . Observe que os modelos  $M_i$  possuem o mesmo domínio.

Nem sempre acontece como no exemplo acima. Para algumas sentenças podem existir cadeias de modelos que não têm limites inferiores. Quando todas as cadeias são assim, não há modelo minimal.

Exemplo 3.5. Seja  $L = \langle S \rangle$ ,  $S$  um símbolo predicado binário e  $\Gamma$  o conjunto das sentenças abaixo:

- (i)  $\forall x . \exists y . S(x, y)$
- (ii)  $\exists y . \forall x . \neg S(x, y)$
- (iii)  $\forall xyz . (S(x, y) \wedge S(x, z) \rightarrow y \approx z)$
- (iv)  $\forall xyz . (S(y, x) \wedge S(z, x) \rightarrow y \approx z)$

Tome  $D_k = \{x \in \mathbb{N}; x \geq k\}$ ;  $k = 1, 2, 3, \dots\}$  e  $S$  interpretado como a função sucessor usual, isto é,  $S(x, y)$  significa que "y é o sucessor de x".

As estruturas  $M_k = \langle D_k, S \rangle$  são modelos de  $\Gamma$  e

$$M_1 \supset M_2 \supset \dots M_k \supset M_{k+1} \supset \dots$$

Portanto  $\Gamma$  não tem modelo minimal.

Observação 3.1. Um melhor diagrama para representar o reticulado dos modelos de um conjunto de sentenças  $\Gamma$  é o seguinte:

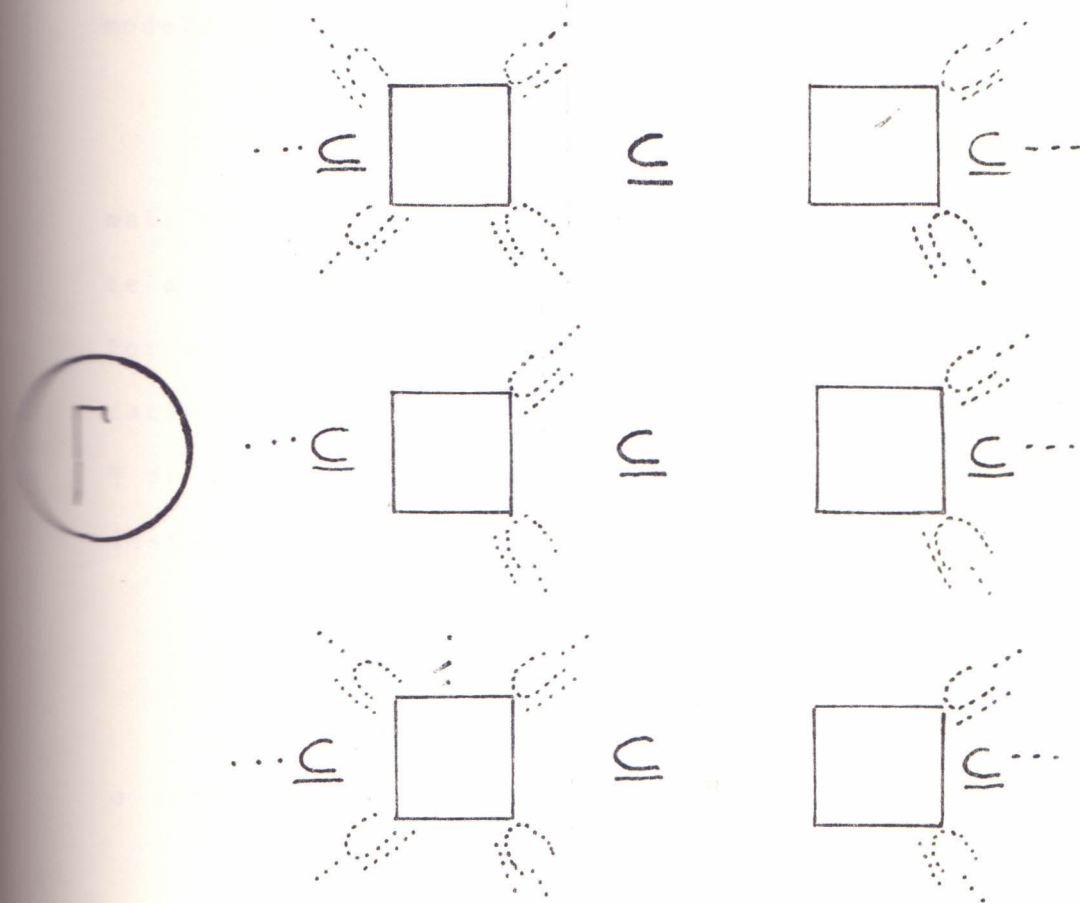


Figura 3

(Sempre que houver uma cadeia com limite inferior, este é um modelo minimal para  $\Gamma$ .)



### 3.1.2. Conseqüência Lógica Minimal

Definição 3.2. Uma sentença  $\sigma$  é conseqüência lógica minimal de um conjunto de sentenças  $\Sigma$ , em símbolos  $\Sigma \models_m \sigma$ , sss todo modelo minimal de  $\Sigma$  é também modelo de  $\sigma$ .

Portanto  $\Sigma \models \sigma$  implica que  $\Sigma \models_m \sigma$ .

Introduzimos os conceitos de satisfatibilidade minimal, teoria minimal, axiomatização minimal, etc., naturalmente a partir dos respectivos conceitos da lógica clássica. Por exemplo, um conjunto de sentenças é minimalmente satisfável se possui modelo minimal. Teoria minimal de conjunto  $T$  de axiomas,  $\text{Thm}$ , é o conjunto das conseqüências minimais deste, conjunto, isto é:

$$\text{Thm}(\Sigma) = \{\sigma; \Sigma \models_m \sigma\}$$

e assim por diante.

Exemplo 3.6. Seja  $\sigma = \forall x \forall y x \approx y$ . Então  $\emptyset \models_m \sigma$ . Pois conforme vimos no exemplo 3.1 da seção anterior, os modelos minimais de  $\emptyset$  são todas estruturas cujo domínio tem apenas um elemento, e a sentença  $\sigma$  é verdadeira em tais estruturas.

Exemplo 3.7. Seja  $\Gamma = Pa \vee Pb \wedge a \neq b$

Então  $\Gamma \cup \{Pa\} \models_m \neg Pb$   
 $\Gamma \cup \{Pb\} \models_m \neg Pa$

(Veja exemplo 3.2)

Exemplo 3.8. Seja  $A_0$  o conjunto dos axiomas de Peano sem a indução (Ver capítulo 2). O modelo padrão da aritmética é o único modelo minimal de  $A_0$ . Portanto, se  $\sigma$  é uma sentença aritmética verdadeira,  $A_0 \models_m \sigma$ . Isto é,  $A_0$  é um conjunto de axiomas para a aritmética. Portanto a aritmética é minimalmente axiomatizável.

Exemplo 3.9. Seja  $\Gamma$  o conjunto das sentenças de (i) a (iv) do exemplo 3.5. Então  $\Sigma \models_m \sigma$  para qualquer sentença  $\sigma$  da linguagem (observe que  $\Sigma$  é minimalmente insatisfável mas não insatisfável).

Este exemplo ilustra que se  $\Sigma \models_m \sigma$  não implica que  $\Sigma \models \sigma$  como era de se esperar.

### 3.1.3. Propriedades

#### 3.1.3.1. Não-Monotonicidade

Em lógica de primeira ordem a função  $F$  que associa a um conjunto de sentenças suas conseqüências lógicas é uma função monotônica no sentido que se  $A \subseteq B$  então  $F(A) = \text{CN}(A) \subseteq \text{CN}(B) = F(B)$ . ( $\text{CN}(X)$  é o conjunto das conseqüências lógicas do conjunto de sentenças  $X$ ). Com efeito, uma sentença  $q$  é conseqüência lógica de um conjunto de sentenças

A em símbolos  $A \models q$ , se  $q$  é verdadeira em todos modelos de  $A$ . Se  $M$  é um modelo de  $B$  e  $A \subseteq B$ , então  $M$  é um modelo de  $A$  (isto é,  $K(B) \subseteq K(A)$ , onde  $K(X)$  é a classe dos modelos do conjunto de sentenças  $X$ ). Assim, se  $A \subseteq B$  e  $A \models q$  então  $B \models q$ .

O mesmo não acontece com a semântica minimal. Se  $A \subseteq B$  não temos que  $K_m(B) \subseteq K_m(A)$  ( $K_m(X)$  é a classe dos modelos minimais de  $X$ ). As classes dos modelos minimais de  $A$  e  $B$  não guardam esta relação de ordem, podem ser classes distintas. Assim, na semântica minimal ao adicionarmos novos axiomas podemos perder algumas conclusões tiradas anteriormente.

Proposição 3.1. A semântica minimal é não-monotônica, isto é, se  $A$  e  $B$  são conjuntos de sentenças e  $A \subseteq B$ , o conjunto das conseqüências lógicas minimais de  $A$ , pode não estar contido no conjunto das conseqüências lógicas minimais de  $B$ , em símbolos,  $CNm(A) \not\subseteq CNm(B)$  ( $CNm(X)$  é o conjunto das conseqüências lógicas minimais de  $X$ ).

Exemplo 3.10. Seja  $\Gamma = \emptyset$ .  $\Gamma \models_m (\forall x)(\forall y) x \approx y$  (ver exemplo 3.6).

Seja  $\Sigma = \{\exists x . \exists y . x \neq y\}$ . Os modelos minimais de  $\Sigma$  são as estruturas cujo domínio têm dois elementos. Portanto  $\Sigma \not\models_m \forall x . \forall y . x \approx y$ .

Exemplo 3.11. Seja  $L = \langle P, a, b \rangle$   $P$  um símbolo funcional e  $a$  e  $b$  símbolos constantes; e  $\Gamma = Pa \vee Pb \wedge a \neq b$ . Temos que

$$\Gamma \cup \{Pa\} \models_m \neg Pb$$



Mas

$$\Gamma \cup \{Pa\} \cup \{Pb\} \not\models_m \neg Pb$$

(ver exemplo 3.7)

A monotonicidade da lógica de primeira ordem exerce um importante papel na teoria de prova desta lógica. É esta propriedade que permite uma teoria de prova local. Local no sentido que, se  $A$  é um conjunto de sentenças e se determinamos uma prova de uma sentença  $\sigma$  usando apenas um subconjunto  $A'$  de  $A$  então sabemos que  $\sigma$  é um teorema (ver capítulo 2) de  $A$ . Não precisamos necessariamente levar em conta todas as sentenças de  $A$  para determinarmos uma prova de  $\sigma$ . Esta localidade implicará que os teoremas de um conjunto de sentenças em lógica de primeira ordem, se não são decidíveis, são pelo menos semidecidíveis. É possível uma enumeração recursiva deles. O mesmo não acontece com as lógicas não-monotônicas. Qualquer teoria de prova para estas lógicas tem inerentemente algum processo não semi-decidível.

### 3.1.3.2. Inexistência de modelos minimais

A satisfatibilidade minimal é uma propriedade mais forte que a satisfatibilidade da lógica de primeira ordem. Existem conjuntos de sentenças satisfatíveis em lógica de

primeira ordem mas não minimalmente satisfatíveis. Uma característica importante da lógica de primeira ordem é a relação entre consistência e satisfatibilidade (ver capítulo 2). Se um conjunto de sentenças é consistente (a partir dele não se pode provar uma contradição), então é satisfatível (existe um modelo para o conjunto). Mesmo ao trabalharmos com sistemas dedutivos mais adequados à semântica minimal, como ao adicionarmos a regra de circunscrição (ver capítulo 4) ao sistema dedutivo de primeira ordem, teremos conjunto de sentenças consistentes (circunscriptivamente) mas não minimalmente satisfatíveis, embora valha a recíproca: Todo conjunto de sentenças minimalmente satisfatível é consistente (circunscriptivamente), (Ver capítulo 4).

Proposição 3.2. Um conjunto de sentenças satisfatível pode não ser minimalmente satisfatível.

Exemplo 3.12. Seja  $L = \emptyset$  e  $\Gamma$  o conjunto infinito dos axiomas abaixo:

$$\lambda_2: \exists x_1, x_2 \quad x_1 \neq x_2$$

$$\lambda_3: x_1, x_2, x_3 \quad x_1 \neq x_2 \wedge x_1 \neq x_3 \wedge x_2 \neq x_3$$

⋮  
⋮  
⋮

$\lambda_n$  afirma que o modelo tem no mínimo (n elementos). Os modelos com exatamente n elementos são os modelos minimais de  $\Gamma_0 = \{\lambda_2, \lambda_3, \dots, \lambda_n\}$ . Uma estrutura de L é modelo de  $\Gamma = \{\lambda_2, \lambda_3, \dots\}$  sss for um conjunto infinito. Todo conjunto infinito possui subconjunto próprio infinito. Assim,  $\Gamma$  não tem modelo minimal.

Exemplo 3.13. Ver exemplo 3.5.

### 3.1.3.3. Não-Compacidade

Em lógica clássica se  $\Sigma \models \sigma$  e  $\Sigma$  é infinito, então existe um subconjunto finito de  $\Sigma$ ,  $\Sigma_0$  tal que  $\Sigma_0 \models \sigma$ . Ou por outra, um conjunto de sentenças é satisfatível sss é finitamente satisfatível (todo subconjunto finito é satisfatível). A compacidade não é gozada pela semântica minimal conforme proposição abaixo.

Proposição 3.3. Um conjunto  $\Gamma$  de sentenças finitamente minimalmente satisfatível pode não ser minimalmente satisfatível.

Exemplo 3.14. Ver exemplo 3.12.

---

Nota: Exemplo 3.12 sugerido por P.A.S. Veloso em conversações pessoais.



#### 3.1.3.4. Enumerabilidade dos modelos minimais

Em lógica de primeira ordem, se a linguagem tem cardinalidade  $\chi$  e se um conjunto de sentenças  $\Gamma$  tem um modelo, então  $\Gamma$  tem um modelo de cardinalidade menor ou igual a  $\chi$  (Teorema de Löwenhein-Skolem). Daí, se a linguagem é enumerável  $\Gamma$  tem um modelo enumerável. Por outro lado, se  $\Gamma$  tem um modelo infinito, então  $\Gamma$  tem um modelo de todas cardinalidades maiores (ou igual) à cardinalidade da linguagem (Teorema de Löwenhein-Skolem-Tarski). Ver capítulo 2. A proposição abaixo afirma que se  $M$  é um modelo minimal para um conjunto de sentenças em uma linguagem de cardinalidade  $\chi$ , então  $M$  tem cardinalidade menor ou igual a  $\chi$ . Se a linguagem é enumerável todo modelo minimal de  $\Gamma$  (se existir) é enumerável. A demonstração é uma versão adaptada da demonstração do Teorema de Löwenhein-Skolem, que apela às funções de Skolem. O leitor encontrará uma breve explanação sobre essas funções no capítulo 2.

Proposição 3.4. Em uma linguagem  $L$  de cardinalidade  $\chi$  todo modelo minimal tem cardinalidade menor ou igual a  $\chi$ .

Demonstração: Dado um modelo para a linguagem  $W$ , mostraremos que ele possui um submodelo de cardinalidade menor ou igual a  $\chi$  elementarmente equivalente (ver capítulo 2).

Seja  $S$  um subconjunto do domínio de  $W$ ,  $S \subseteq |W|$ , contendo todas as constantes de  $A$  e de cardinalidade  $\lambda \leq \chi$ . Para toda fórmula  $\varphi$  de  $L$  tome as funções de Skolem de  $\varphi$  em  $W$ . Seja  $\bar{S}$  o fechamento de  $S$  sob todas funções de Skolem ( $\bar{S} \subseteq |W|$ ). Seja  $B$  o submodelo de  $W$  com domínio  $\bar{S}$ ,  $|B| = \bar{S}$ . Mostraremos que  $B$  é elementarmente equivalente a  $W$ . Basta mostrar que para toda fórmula  $\varphi$  de  $L$  nas variáveis  $V_1, \dots, V_n$  e para todo elemento  $S_1, \dots, S_n \in \bar{S}$ , temos que

$$(*) \quad B \models \varphi[S_1, \dots, S_n] \text{ sss } W \models \varphi[S_1, \dots, S_n]$$

Como  $\bar{S}$  é fechado sob todas funções de Skolem, para todas fórmulas  $\exists V \varphi(V, V_1, \dots, V_n)$  em  $V_1, \dots, V_n$  e para todo  $S_1, \dots, S_n \in \bar{S}$ ,

$$(**) \text{ se } W \models \exists x \varphi(x, S_1, \dots, S_n), \text{ então existe } S \in \bar{S} \text{ tal que } B \models \varphi(S, S_1, \dots, S_n).$$

Mostraremos (\*) por indução nas fórmulas  $\varphi$ .

A base da indução, as fórmulas atômicas, e as fórmulas com conectivos sentenciais são mostrados facilmente. O problema são as fórmulas quantificadas.

Se  $W \models \exists v_1 \varphi [S_2, \dots, S_n]$ , por (\*\*), existe um  $S_1 \in \overline{S}$  tal qual que  $W \models \varphi [S_1, \dots, S_n]$ .

Então, por indução,  $B \models \varphi [S_1, \dots, S_n]$  e

$$W \models \exists v_1 \varphi [a_2, \dots, a_n] \text{ . cqd.}$$

Portanto  $B$  é elementarmente equivalente a  $W$  e observe que  $|B| \leq |S| \cup \|L\| = \lambda + \aleph$ .

**Corolário 3.1** (Enumerabilidade). Em uma linguagem enumerável, todo modelo minimal é enumerável.

Vemos assim que o Teorema de Löwenheim-Skolem-Tarski não é válido para semânticas minimais.

### 3.1.3.5. Incompletude

Uma propriedade fundamental da Lógica Clássica é sua completude. Podemos construir um sistema dedutivo, finitário e efetivo tal que  $\Sigma \models \sigma$  sss  $\Sigma \vdash \sigma$ , onde  $\Sigma \vdash \sigma$  significa que  $\sigma$  é um teorema de  $\Sigma$ , ou seja, existe uma prova a partir de  $\Sigma$  de  $\sigma$  (Ver capítulo 2).

Uma prova é um algoritmo como é normalmente definido, uma seqüência de passos finita e cada passo pode ser realizado sem brilhantes "flashes" de raciocínio. Algo que uma máquina possa executar. Assim, se  $\Sigma$  é um conjunto infinito de sentenças, na prova de  $\sigma$  você não pode usar todas sentenças de  $\Sigma$ . O Teorema da Compacidade (ver capítulo 2) que pode



ser provado usando-se o sistema dedutivo de primeira ordem, garante a existência de um conjunto finito  $\Sigma_0 \subseteq \Sigma$  tal que  $\Sigma_0 \models \sigma$ . Por outro lado, é trivialmente decidível se uma seqüência de sentenças é uma prova. Assim, o conjunto das sentenças válidas (deduzidas a partir do  $\emptyset$  (ver capítulo 2)) é recursivamente enumerável (enumeramos todas seqüências finitas de sentenças e decidimos se é uma prova (a partir do  $\emptyset$ ) ou não. Se não, nós a descartamos. Se for, colocamos sua última sentença na lista de validades). O Teorema da enumerabilidade (ver capítulo 2) afirma que o conjunto das sentenças válidas é recursivamente enumerável. Segue-se que o conjunto das conseqüências lógicas de  $\Sigma$ ,  $CN(\Sigma)$ , é semi-decidível. Assim os Teoremas da Compacidade e Enumerabilidade são condições necessárias para que o sistema dedutivo seja completo. É fácil ver que também são suficientes. Ver [ENDERTON, 1972, p. 101]. Já vimos que em semântica minimal não vale o Teorema da Compacidade (e o próximo exemplo mostra que não vale a Enumerabilidade). Portanto não podemos ter um sistema dedutivo, finito e efetivo, completo. A relação de conseqüência lógica minimal vista como uma regra de inferência é infinitária.

Proposição 3.5 (Incompletude da Lógica Minimal). Não existe um algoritmo por meio do qual podemos obter precisamente o conjunto de todas conseqüências minimais de um dado conjunto de sentenças.

Exemplo 3.15. Seja  $L = \langle +, \cdot, S, 0 \rangle$  e  $A_0$  o conjunto dos axiomas de Peano sem a indução (ver capítulo 2). Já vimos que o modelo padrão da aritmética  $W = \langle \mathbb{N}, +, \cdot, S, 0 \rangle$  com as interpretações usuais, é o único modelo minimal de  $A_0$ . Assim,  $A_0 \models \sigma$  sss  $W \models \sigma$  ( $\sigma$  é verdadeiro em  $W$ ). Portanto, se tivéssemos um tal algoritmo poderíamos, por meio dele, derivar todas sentenças aritméticas verdadeiras, o que é uma contradição com a incompletude da aritmética (ver capítulo 2).

Segue-se que o conjunto das conseqüências minimais de um conjunto de sentenças não é efetivamente enumerável, portanto nem ao menos semi-decidível.

#### 3.1.4. Modelos Mínimos e Minimalidade Local

Apresentamos abaixo dois conceitos similares ao conceito de minimalidade. O primeiro é mais restrito e o segundo é mais abrangente. Estes conceitos serão mencionados nos capítulos subseqüentes.

Definição 3.3. Em uma linguagem  $L = \langle R, F, C \rangle$ . Seja  $K(L) = \{W_i, i \in I\}$  uma família não-vazia de modelo para  $L$  com o mesmo domínio,  $|W|$ , e mesmas funções. Diz-se que  $W_i \subseteq W_j$  sss para todo predicado  $P \in R$ ,  $P^{W_i} \subseteq P^{W_j}$  {a extensão de  $P$  em  $W_i$  está contida na extensão de  $P$  em  $W_j$ }. Definimos a interseção  $\bigcap_{i \in I} W_i$  como o modelo  $W$  para  $L$  com o mesmo domínio e funções que  $W_i$  e tal que para todo  $P \in R$ ,

$$P^W = \bigcap_{i \in I} P^{W_i}. \quad \underline{W \text{ é o modelo mínimo para } L \text{ no domínio } |W|}.$$

Em uma linguagem  $L$  seja  $M(\Gamma)$  a classe dos modelos de  $\Gamma$  no domínio  $|W|$ , todos com as mesmas funções. Se a interseção dos modelos desta classe for modelo de  $\Gamma$ , então é chamada de modelo mínimo de  $\Gamma$  no domínio  $|W|$ . É fácil ver que há sentenças que não possuem modelo mínimo e que, todo modelo mínimo é um modelo minimal, mas não vice-versa.

Exemplo 3.16. Seja  $L = \langle P, a, b \rangle$ ;  $P$  um símbolo predicado e  $a, b$  símbolos constantes. A sentença  $\Gamma = Pa \vee Pb \wedge a \neq b$  tem dois modelos minimais, ver exemplo 3.2,

$$M_1 = \langle |M_1| = \{a, b\}, P^{M_1} = \{a\}, a^{M_1} = a, b^{M_1} = b \rangle$$

$$M_2 = \langle |M_2| = \{a, b\}, P^{M_2} = \{b\}, a^{M_2} = a, b^{M_2} = b \rangle$$

mas não tem modelo mínimo. Observe que  $P^{M_1} \cap M_2 = \emptyset$

Definição 3.4. Seja  $M$  e  $N$  modelos de um conjunto de sentenças  $\Gamma$ . Dizemos que  $M$  é submodelo de  $N$  em  $P$ , e escrevemos  $M \leq_P N$ . Se  $M$  e  $N$  têm o mesmo domínio, todo símbolo predicado em  $\Gamma$  exceto  $P$  têm as mesmas extensões em  $M$  e  $N$ , mas a extensão de  $P$  em  $M$  está contida na sua extensão em  $N$ .



Definição 3.5. Um modelo  $M$  de  $\Gamma$  é chamado modelo minimal em  $P$  sss para todo modelo  $M'$  de  $\Gamma$ ,

$$M' \leq_P M \text{ somente se } M' = M.$$

É fácil ver que

Proposição 3.6.  $M$  é um modelo minimal de  $\Gamma$  sss  $M$  é minimal em  $P$  para todo símbolo predicado  $p$  de  $\Gamma$ .

### 3.2. Semânticas Nomeáveis

#### 3.2.1. Estruturas Nomeáveis

Definição 3.6. Considere uma linguagem de 1ª ordem onde  $C$ ,  $F$  e  $R$  são respectivamente os conjuntos dos símbolos constantes funcionais e relacionais. Seja  $T$  o conjunto de todos os termos sem variáveis de  $L$ . Assumiremos  $C$  não vazio, de modo que  $T$  também não será.

Considere a estrutura  $W = \langle A, C^W, F^W, R^W \rangle$  para  $L$ . Cada  $t \in T$  denota um elemento  $t^A$  no domínio de  $M$ ; definimos a função (denotação)  $d: T \rightarrow A$  que a cada termo  $t$  associa o elemento de  $A$  denotado por ele. Seja  $T^A$  a imagem desta função.

Diremos que W é nomeável (por T) sss  $T^A = A$ , isto é, a função de denotação é sobrejetiva. Portanto, uma estrutura nomeável, ou T-estrutura como às vezes é chamada, é tal que todo elemento é denotado por um termo livre de variáveis o qual é um nome para ele.

Por outro lado, se definirmos a parte acessível (funcionalmente) de A como o menor subconjunto  $F(A)$  de A tal que  $C \subseteq F(A)$  e  $F(A)$  é fechado sob as operações em F. Ou seja,  $F(A)$  é o menor conjunto tal que:

(i)  $C \subseteq F(A)$ ;

(ii) Se  $a_1, \dots, a_n \in F(A)$  e  $f \in F$  é de aridade n então  $F^W(a_1, \dots, a_n) \in F(A)$ .

$F(A)$  é o domínio da subestrutura de W que consiste daqueles elementos de W que são designados por termos livres de variáveis de L, e ainda mais, esta subestrutura está incluída em qualquer outra subestrutura de W, ou seja, é a subestrutura mínima de W. Portanto, uma estrutura W é nomeável sss  $A = F(A)$  onde A é o domínio de W.

Exemplo 3.17. No exemplo 3.3 da seção anterior já vimos que o único modelo minimal dos axiomas (i) a (iv...) são os naturais. Observe que na linguagem com  $\langle 0, S, +, . \rangle$  os naturais também são o único modelo nomeável destes axiomas.

Exemplo 3.18. A aritmética padrão é o único modelo minimal e observe que também é o único modelo nomeável, para os axiomas de Peano.

Mais uma vez vemos a minimalidade e a nomeabilidade cumprindo o mesmo papel. Mas minimalidade e nomeabilidade não são o mesmo conceito. O exemplo abaixo ilustra este fato.

Exemplo 3.19. No exemplo 3.5 vimos que o conjunto  $\Gamma$  de sentenças não possui modelo minimal ( $M_k \supset M_{k+1}$ ,  $i = 1, \dots, +\infty$ , onde  $M_k$  são modelos de  $\Gamma$ ). Mas todos os modelos  $M_k$  são modelos nomeáveis de  $\Gamma$ .

Exemplo 3.20. Seja  $L = \langle a \rangle$  onde  $a$  é um símbolo constante. A sentença  $\sigma = \exists x . x \neq a$  não possui modelo nomeável em  $L$ . Observe que  $\sigma$  é satisfatível.

Observação 3.2. Nomeabilidade é um conceito que depende diretamente da linguagem. Ao afirmarmos que um modelo é nomeável nossa afirmação só estará completa se dissermos também a linguagem que estamos usando. Por exemplo, a sentença  $\sigma$  do exemplo anterior tem um modelo nomeável na linguagem  $\langle a, b \rangle$ ;  $a, b$  símbolos constantes.

Exemplo 3.21. Seja  $L = \langle +, \cdot, S, 0, a \rangle$ ; uma expansão da linguagem da aritmética,  $\Gamma$  o conjunto dos axiomas de Peano sem a indução e  $\Sigma$  o conjunto das sentenças

$$\{a \neq 0, a \neq S0, a \neq SS0, \dots\}$$



Então  $\Gamma \cup \Sigma$  não tem modelo nomeável em  $L$ . Observe que a aritmética padrão não é modelo de  $\Gamma \cup \Sigma$ , mas todo modelo não padrão da aritmética pode ser (desde que estendido apropriadamente).

### 3.2.2. Conseqüência Lógica Nomeável

Definição 3.7. Em uma linguagem  $L$ , uma sentença da linguagem  $\sigma$  é conseqüência lógica nomeável de um conjunto de sentenças da linguagem,  $\Sigma$ , em símbolos,  $\Sigma \models_n \sigma$ , sss todo modelo nomeável (em  $L$ ) de  $\Sigma$  é também modelo de  $\sigma$ .

Portanto,  $\Sigma \models \sigma$  implica que  $\Sigma \models_n \sigma$ , mas  $\Sigma \models_n \sigma$  não implica  $\Sigma \models \sigma$ . E  $\Sigma \models_m \sigma$  implica  $\Sigma \models_n \sigma$ ? ou vice-versa? O leitor é convidado a mostrar que nenhuma das implicações é válida.

Analogamente como fizemos em relação à minimalidade introduzimos os conceitos de satisfatibilidade nomeável, teoria nomeável, axiomatização nomeável etc. em contrapartida aos respectivos conceitos da lógica clássica.

Claramente entendemos por teoria nomeável de um conjunto  $\Sigma$  de sentenças,  $\text{Th}_n(\Sigma)$ , o conjunto das conseqüências nomeáveis de  $\Sigma$ , em símbolos

$$\text{Th}_n(\Sigma) = \{\sigma, \Sigma \models_n \sigma\}.$$

Em [VELOSO, 83], de onde extraímos o conceito abaixo, o leitor encontra uma variedade de resultados concernentes

às teorias nomeáveis.

Adotaremos a seguinte notação:  $L(V)$  é o conjunto de todas fórmulas da linguagem  $L$  tendo no máximo  $V$  como variável livre;  $\varphi(V)$  é uma fórmula da linguagem  $L$ , sendo  $V$  sua única variável livre;  $\varphi(t)$  é a substituição da variável  $V$  pelo termo  $t \in T$ . Lembre-se que  $T$  é o conjunto de todos os termos sem variáveis de  $L$ .

- Um conjunto  $\Sigma$  de sentenças de uma linguagem  $L$  é uma teoria  $T$ -completa sss sempre que o conjunto  $\{\varphi(t); t \in T\} \subseteq \Sigma$ , então  $\forall V. \varphi(V) \in \Sigma$ , para toda  $\varphi(V) \in L(V)$ .

Gostaria agora de demonstrar uma proposição, mesmo que para isso tenha que fazer algumas definições e enumerar alguns teoremas, a título de ilustrar os resultados que podem ser obtidos com a nomeabilidade, e principalmente pela importância do resultado em assuntos correlatos, como a  $W$ -Lógica (seção 3.2.3).

O leitor poderá passar para a próxima seção sem prejuízo de compreensão. Assinalaremos as partes dispensáveis com traços verticais.

O ponto inicial de nossa discussão é a noção de um conjunto  $\Gamma(X)$  de fórmulas de  $L$  nas variáveis (livres)  $x_1, \dots, x_n$ .  $\Gamma$  é um conjunto de fórmulas de  $L$  nas variáveis (livres)  $x_1, \dots, x_n$ , em símbolos,  $\Sigma = \Sigma(x_1, \dots, x_n)$ , sss  $x_1, \dots, x_n$  são variáveis individuais distintas e toda fórmula  $\gamma$  em  $\Gamma$  contém no máximo as variáveis  $x_1, \dots, x_n$  livres.

Se  $\Gamma = \Gamma(x_1, \dots, x_n)$  então a notação

$$\mu \models \Gamma [a_1, \dots, a_n]$$

significa que para todo  $\gamma \in \Gamma$ , a seqüência  $a_1, \dots, a_n$  satisfaz  $\gamma$  em  $\mu$ ; neste caso dizemos que  $a_1, \dots, a_n$  satisfaz ou realiza,  $\Gamma$  em  $\mu$ .

Seja  $\Gamma$  um conjunto de fórmulas nas variáveis  $x_1, \dots, x_n$  e seja  $\mu$  modelo para  $L$ . Dizemos que  $\mu$  realiza  $\Gamma$  sss alguma  $n$ -upla de elementos de  $|\mu|$ , satisfaz  $\Gamma$  em  $\mu$ . Dizemos que  $\mu$  omite  $\Gamma$  sss  $\mu$  não realiza  $\Gamma$ .

Por um tipo  $\Gamma(x_1, \dots, x_n)$  nas variáveis  $x_1, \dots, x_n$  entendemos um conjunto de fórmulas de  $L$  nestas variáveis, maximal consistente.

Dizemos que  $\Gamma(x_1, \dots, x_n)$  é consistente com uma teoria  $\Sigma$  sss  $\Sigma$  tem um modelo que satisfaz  $\Gamma$ .

Seja  $\Gamma = \Gamma(x_1, \dots, x_n)$  um conjunto de fórmulas de  $L$ . Uma teoria  $\Sigma$  em  $L$  localmente realiza  $\Gamma$  sss existe uma fórmula  $\varphi(x_1, \dots, x_n)$  em  $L$  tal que:

- (i)  $\varphi$  é consistente com  $\Sigma$ .
- (ii) Para todo  $\gamma \in \Gamma$ ,  $\Sigma \models \varphi \rightarrow \gamma$ .

Isto é, toda  $n$ -upla em um modelo de  $\Sigma$  que satisfaz  $\varphi$  realiza  $\Gamma$ .

$\Sigma$  localmente omite  $\Gamma$  sss  $\Sigma$  não localmente realiza  $\Gamma$ . Assim,  $\Sigma$  localmente omite  $\Gamma$  sss para toda fórmula



$\varphi(x_1, \dots, x_n)$  que é consistente com  $\Sigma$  existe  $\gamma \in \Gamma$  tal que  $\varphi \wedge \neg \gamma$  é consistente com  $\Sigma$ .

Se  $\Sigma$  é uma teoria completa temos que  $\Sigma$  localmente omitir  $\Gamma$  é condição necessária para  $\Sigma$  omitir  $\Gamma$ , conforme proposição abaixo.

Proposição 3.7. Seja  $\Sigma$  uma teoria completa em  $L$  e seja  $\Gamma = \Gamma(x_1, \dots, x_n)$  um conjunto de fórmulas de  $L$ . Se  $\Sigma$  tem um modelo que omite  $\Gamma$ , então  $\Sigma$  localmente omite  $\Gamma$ .

O "teorema dos tipos omitidos" é a inversa da proposição acima e vale para teorias consistentes arbitrárias em uma linguagem enumerável.

Teorema 3.1 (TIPOS OMITIDOS). Seja  $\Sigma$  uma teoria consistente em uma linguagem enumerável  $L$  e seja  $\Gamma = \Gamma(x_1, \dots, x_n)$  um conjunto de fórmulas de  $L$ . Se  $\Sigma$  localmente omite  $\Gamma$ , então  $\Sigma$  tem um modelo enumerável que omite  $\Gamma$ .

A prova deste teorema assim como a da proposição anterior é encontrada em [CHANG-KEISLER, 1973, p. 79].

Agora podemos enunciar nossa proposição que afirma que  $T$ -completude é condição necessária para  $T$ -satisfatibilidade, e demonstrá-la.

Proposição 3.8. Se  $\Sigma$  é uma teoria  $T$ -completa em uma linguagem enumerável  $L$ , então  $\Sigma$  tem um modelo nomeável (em  $L$ ). Em outros, termos

$\Sigma$  é  $T$ -completo  $\Rightarrow \Sigma$  é  $T$ -satisfatível.

Demonstração: Primeiramente tomamos uma enumeração  $t_1, t_2, \dots$ , para o conjunto  $T$ , os termos sem variáveis da linguagem (é possível, pois  $L$  é enumerável). Para demonstrarmos nossa proposição utilizamos o lema.

Lema 3.1: Seja  $\Gamma(x)$  o conjunto de fórmulas  $\{x \neq t_1, x \neq t_2, \dots\}$  onde  $t_1, t_2, \dots$  é uma enumeração de  $T$ . Se  $\Sigma$  é um conjunto satisfatível de sentenças tal que  $\Sigma$  omite  $\Gamma(x)$  então  $\Sigma$  tem um modelo nomeável.

Demonstração: trivial.

Portanto, pelo lema 3.1, precisamos apenas mostrar que  $\Sigma$  omite  $\Gamma(x)$ . Suponha  $\varphi(x)$  consistente com  $\Sigma$ .

Então  $\Sigma \models \forall x \neg \varphi(x)$  falha. Pela  $T$ -completude existe um termo  $t \in T$  tal que não  $\Sigma \models \neg \varphi(t)$ .

Portanto  $\varphi(t)$  é consistente com  $\Sigma$ , então  $\varphi(x) \wedge \neg x \neq t$  é consistente com  $\Sigma$ . Assim  $\Sigma$  localmente omite  $\Gamma(x)$ , pelo "teorema dos tipos omitidos",  $\Sigma$  omite  $\Gamma(x)$  e a demonstração está completa.

Estreitamente relacionado com os conceitos apresentados está o que chamamos de  $W$ -Lógica, uma lógica de primeira ordem na qual adicionamos uma regra de inferência infinita, a regra- $W$ , e permitimos provas infinitas.

### 3.2.3 - W-Lógica

A W-Lógica exerce um papel importante na teoria dos números ou aritmética. A incompletude da aritmética demonstrada por Gödel em 1931 sempre foi um incômodo aos matemáticos. Ao adicionarmos a regra-W ao sistema dedutivo da lógica de primeira ordem, teremos a importante propriedade, trivialmente demonstrada da completude da aritmética. Intuitivamente a regra-W é uma regra de inferência infinita que afirma que os únicos elementos no domínio de um modelo são os números naturais.

A linguagem da W-Lógica é a linguagem da aritmética  $L = \langle S, +, \cdot, 0 \rangle$ ; os modelos nomeáveis em  $L$  são chamados de W-modelos. Se abreviamos  $1 = S0$ ,  $2 = SS0$ ,  $3 = SSS0$ , ... os W-modelos são as estruturas  $W$  com domínio  $|W| = \{0, 1, 2, 3, \dots\}$ . Em vista do exposto na demonstração da proposição 3.8 podemos definir os W-modelos como sendo os modelos de  $L$  que omitem o conjunto  $\{x \neq 0, x \neq 1, x \neq 2, \dots\}$

A regra-W é uma regra infinita de inferência que de  $\varphi(0), \varphi(1), \varphi(2), \dots$  permite inferir-se  $\forall V. \varphi(V)$ , para toda fórmula  $\varphi(V) \in L(V)$ . Em símbolos:

$$\text{Regra-W} \quad \frac{\{\varphi(0), \varphi(1), \varphi(2), \dots\}}{\forall V. \varphi(V)}$$

Observe que a regra-W atua de modo a restringir o domínio do modelo à sua parte nomeável. A completude da



W-lógica, proposição 3.9, decorre deste fato. Podemos agora definir o que chamamos de W-Lógica.

Definição 3.8. W-Lógica é a lógica de primeira ordem da linguagem  $L = \langle S, +, \cdot, 0 \rangle$  na qual introduzimos a regra-W de inferência e as provas podem ser infinitamente longas.

Uma teoria  $A$  é W-consistente sss não existe uma fórmula  $\varphi(V) \in L(V)$  tal que

$$\{\varphi(0), \varphi(1), \varphi(2), \dots\} \subseteq A$$

e

$$\exists x . \neg \varphi(x) \in A$$

Uma teoria  $A$  é W-completa sss para toda fórmula  $\varphi(V) \in L(V)$ , se  $\{\varphi(0), \varphi(1), \varphi(2), \dots\} \subseteq A$  então  $\forall V . \varphi(V) \in A$ . Observe-se que W-completude é uma particularização da T-completude para a linguagem da aritmética.

Segue-se da proposição 3.8 que

Corolário 3.2. Seja  $A$  uma teoria consistente em  $L$ , se  $A$  é W-completa então  $A$  tem um W-modelo.

Demonstração. A linguagem  $L$  é enumerável e se  $A$  é uma teoria W-completa, então  $A$  é T-completa. Portanto, pela proposição 3.8  $A$  tem um modelo nomeável em  $L$ , um W-modelo.

É fácil ver que a W-Lógica é correta. Se uma teoria  $T$  é W-satisfatível (tem um W-modelo), então  $T$  é W-consistente. A proposição abaixo afirma que a W-lógica é completa.

Proposição 3.9 (Teorema da W-completude). Uma teoria  $T$  em  $L$  é consistente em W-Lógica sss  $T$  é W-satisfatível.

Demonstração:

Seja  $T'$  o conjunto de todas sentenças de  $L$  deriváveis de  $T$  em W-Lógica.  $T$  é consistente em W-Lógica sss  $T'$  é consistente em  $L$

$\Rightarrow$ )  $T'$  é W-completa, portanto se  $T'$  é consistente em  $L$  pelo corolário 3.2,  $T'$  tem um W-modelo e este modelo é um W-modelo de  $T$ .

$\Leftarrow$ ) Se  $T$  tem um W-modelo então  $T$  é W-consistente (corretude da W-Lógica) e portanto  $T'$  é consistente em  $L$ .

Assim as conseqüências semânticas e os teoremas da W-Lógica de um conjunto  $\Sigma$  de sentenças coincidem. Como conseqüência a aritmética é completa em W-Lógica. Os teoremas dos axiomas de Peano em W-Lógica são a aritmética padrão.

A W-Lógica pode ser naturalmente generalizada a outras linguagens enumeráveis, onde os W-modelos são substituídos pelos modelos nomeáveis e a regra-W é generalizada pela regra-T.

Regra-T:  $\frac{\{\varphi(t); t \in T\}}{\forall v. \varphi(v)}$ , Para todo  $\varphi(v) \in L(v)$ .

W-consistência e W-completude são generalizadas respectivamente por T-consistência e T-completude. E a W-Lógica Generalizada é claramente correta e a proposição abaixo afirma a completude. A demonstração é análoga à demonstração da proposição 3.9.

Proposição 3.10. Uma teoria T em uma linguagem enumerável L é consistente em W-Lógica Generalizada sss T é nomeavelmente satisfatível.

#### 3.2.4. Interpretações de Herbrand

Antes de apresentarmos as propriedades da semântica nomeável será útil estudarmos uma classe especial de estruturas nomeáveis, as estruturas de Herbrand.

Seja uma linguagem enumerável  $L = \langle F, R, C \rangle$ ;

F: conjunto dos símbolos funcionais;

R: conjunto dos símbolos predicados;

C: conjunto dos símbolos constantes;

$C \neq \emptyset$ . O universo de Herbrand de L,  $H(L)$  é o conjunto de todos os termos livres de variáveis de L, ou seja,  $H(L) = T$ .

Base de Herbrand de L,  $B(H)$ , é o conjunto de todas fórmulas atômicas  $P(t_1, \dots, t_n)$  onde  $P \in R$  e  $t_1, \dots, t_n \in H(L)$ .



Uma Interpretação de Herbrand de L, ou H-interpretação, é uma interpretação I de L em H(L) tal que:

- (1) I associa a toda constante de C a si mesma.
- (2) Se  $f \in F$  e  $t_1, \dots, t_n \in H(L)$ , I associa a f uma função que transforma  $(t_1, \dots, t_n)$  (um elemento em  $H(L)^n$ ) em  $f(t_1, \dots, t_n)$  (um elemento em H(L)).

Seja  $B(H) = \{A_1, A_2, \dots, A_n, \dots\}$  a base de Herbrand de L. Uma H-interpretação I pode ser convenientemente representada por um conjunto.

$$I = \{m_1, m_2, \dots, m_n, \dots\}$$

Se  $m_j$  é  $A_j$ , a  $A_j$  é atribuído "verdadeiro", se não a  $A_j$  é atribuído "falso".

Exemplo 3.22. Seja  $L = \langle F, R, C \rangle$  com

$$F = \{f\}$$

$$R = \{P, Q\}$$

$$C = \{a\}$$

Neste caso

$$H(L) = \{a, f(a), f(f(a)) \dots\}$$

$$BH(L) = \{P(a), Q(a), P(f(a)), Q(f(a)), \dots\}$$

e as interpretações abaixo são H-interpretações:

$$I_1 = \{P(a), Q(a), P(f(a)), Q(f(a)), \dots\}$$

$$I_2 = \{\neg P(a), \neg Q(a), \neg P(f(a)), \neg Q(f(a)), \dots\}$$

$$I_3 = \{P(a), \neg Q(a), P(f(a)), \neg Q(f(a)), \dots\}$$

Interpretações de Herbrand exercem um papel central em prova automática de teoremas baseado em dois resultados:

Teorema A: Para toda fórmula  $\varphi$  existe uma sentença em forma clausal, a forma padrão de  $\varphi$ , tal que  $\varphi$  é satisfatível sss  $S$  é satisfatível.

(Ver capítulo 2, Funções de Skolem).

Teorema B: Um conjunto de sentenças em forma clausal tem um modelo sss tem um modelo de Herbrand.

(Este teorema é uma versão do teorema de Löwenheim-Skolem, uma demonstração é encontrada em [CHANG-LEE, 1973, p. 55])

Interpretação de Herbrand são estruturas nomeáveis que de uma certa forma, incluem todas as outras estruturas nomeáveis da linguagem, conforme a proposição 3.11 a seguir.

Lema 3.2. Seja  $W$  uma estrutura nomeável para uma linguagem  $L = \langle F, R, C \rangle$  e  $S$  uma estrutura para a mesma linguagem  $L$ . Existe no máximo um homomorfismo de  $W$  em  $S$ , que será sobre sss  $S$  é nomeável.

Demonstração. Se  $\emptyset$  é um homomorfismo de  $W$  em  $S$ , então

$$\text{Para todo } c \in C \quad \emptyset(c^W) = c^S$$

Para todo  $f \in F$  de aridade  $n$  e para toda  $n$ -upla

$$\langle a_1, \dots, a_n \rangle \text{ de elementos de } |W|,$$

$$\emptyset(f^W(a_1, \dots, a_n)) = f^S(\emptyset(a_1) \dots \emptyset(a_n))$$

Para todo  $P \in R$  de aridade  $n$ , se

$$\langle a_1, \dots, a_n \rangle \in P^W \text{ então } \langle \emptyset(a_1), \dots, \emptyset(a_n) \rangle \in P^S$$

Suponhamos que exista um outro homomorfismo  $\emptyset'$  de  $W$  para  $S$ .

Quero mostrar que para todo  $a \in |W|$ ,  $\emptyset(a) = \emptyset'(a)$ .

Mas como  $W$  é nomeável, se  $a \in |W|$  então

$$\text{ou } a = c^W$$

$$\text{ou } a = f(a_1, \dots, a_n) \text{ onde}$$

$$a_1, \dots, a_n \in |W|$$

$$\text{Se } a = c^W, \emptyset(a) = c^S = \emptyset'(a)$$

Se  $a = f^W(a_1, \dots, a_n)$ , supondo-se

$$\emptyset(a_i) = \emptyset'(a_i), \quad i = 1, \dots, n,$$

temos que

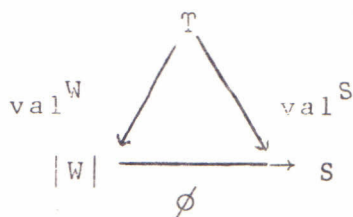
$$\emptyset'(a) = \emptyset'(f^W(a_1, \dots, a_n))$$



$$\begin{aligned}
&= f^S(\varphi'(a_1), \dots, \varphi'(a_n)); && \varphi' \text{ é homomorf.} \\
&= f^S(\varphi(a_1), \dots, \varphi(a_n)); && \text{Hip. de Indução} \\
&= \varphi(f^W(a_1, \dots, a_n)); && \varphi \text{ é homomorf.} \\
&= \varphi(a)
\end{aligned}$$

Portanto, se existir um homomorfismo de  $W$  para  $S$ , é único.

Que  $\varphi$  é sobre sss  $S$  é nomeável pode ser deduzido do diagrama comutativo



onde  $T$  é o conjunto de termos livres de variáveis e  $\text{val}^W$ ,  $\text{val}^S$  são funções de valoração dos termos de  $W$  e  $S$  respectivamente.

Corolário 3.3. Se  $W$  é uma estrutura nomeável e  $\equiv$  é uma relação de congruência em  $W$ , então  $W/\equiv$  também é nomeável.

Proposição 3.11. Seja  $W$  uma estrutura nomeável para  $L = \langle F, R, C \rangle$ . Existe uma interpretação de Herbrand única  $\mathfrak{H}(W)$  tal que  $W \cong \mathfrak{H}(W)/K(\varphi)$ , onde  $K(\varphi)$  é o kernel de homomorfismo único  $\varphi$  de  $\mathfrak{H}(W)$  em  $W$ .

Demonstração: Seja  $\emptyset$  uma função tal que:

(i) Para todo  $c \in C$

$$\emptyset(c) = c^W;$$

(ii) Para todo  $f \in F$

$$\emptyset(f(h_1, \dots, h_n)) = f^W(\emptyset(h_1), \dots, \emptyset(h_n)),$$

para  $h_1, \dots, h_n \in H(L)$ .

Uma vez que a interpretação de Herbrand e  $W$  são nomeáveis,  $\emptyset$  é único e sobre (ver demonstração Lema 3.2). Usamos  $\emptyset$  para definir as relações de  $H(W)$ .

Para todo  $P \in R$  e para todo  $h_1, \dots, h_n \in H(L)$ .

$$\langle h_1, \dots, h_n \rangle \in P^{H(W)} \text{ sss } \langle \emptyset(h_1), \dots, \emptyset(h_n) \rangle \in P^W$$

Observe que isto faz de  $\emptyset$  um homomorfismo forte.

Basta agora provar que o diagrama a seguir comuta e que  $\varphi$  é um isomorfismo.

$$\begin{array}{ccc} H(W) & \xrightarrow{\text{nat}(K(\emptyset))} & H(W)/K(\emptyset) \\ \emptyset \downarrow & \swarrow & \\ W & \xleftarrow{\varphi} & \end{array}$$

(i) O diagrama comuta:

$\varphi$  é definido por

$$\varphi([h]) = \emptyset(h)$$

Observe que  $\varphi$  está bem definido,

se  $[h_1] = [h_2]$  então  $\emptyset(h_1) = \emptyset(h_2)$   
pois  $K(\emptyset)$  é o kernel de  $\emptyset$ .

(ii)  $\varphi$  é homomorfismo:

Para todo  $c \in C$ ,

$$\begin{aligned} \varphi(c^{H(W)/K(\emptyset)}) &= \varphi([c^{H(W)}]) \\ &= \emptyset(c^{H(W)}) \\ &= c^W \end{aligned}$$

Para todo  $f \in F$ ,

$$\begin{aligned} \varphi(f^{H(W)/K(\emptyset)}([h_1], \dots, [h_n])) &= \varphi([f^{H(W)}(h_1, \dots, h_n)]) \\ &= \emptyset(f^{H(W)}(h_1, \dots, h_n)) \\ &= f^W(\emptyset(h_1), \dots, \emptyset(h_n)) \\ &= f^W(\varphi([h_1]), \dots, \varphi([h_n])) \end{aligned}$$



Para todo  $P \in R$

$$\langle [h_1], \dots, [h_n] \rangle \in P^{H(W)/K(\emptyset)}$$

$$\text{sss } \langle h_1, \dots, h_n \rangle \in P^{H(W)}$$

$$\text{sss } \langle \emptyset(h_1), \dots, \emptyset(h_n) \rangle \in P^W$$

$$\text{sss } \langle \varphi([h_1]), \dots, \varphi([h_n]) \rangle \in P^W$$

(iii)  $\varphi$  é um para um:

Sejam  $h_1, h_2$  tais que

$$\varphi([h_1]) = \varphi([h_2])$$

então

$$\emptyset(h_1) = \emptyset(h_2)$$

$$\text{e } [h_1] = [h_2]$$

Finalmente como  $W$  e  $H(W)/K(\emptyset)$  são nomeáveis (Lema 3.2),  $\varphi$  é sobre.

A estrutura  $H(W)$  é chamada a interpretação de Herbrand induzida por  $W$ .

Observação 3.3. Lema 3.2 e proposição 3.11 assim como suas demonstrações foram extraídas de [PEQUENO, 1981].

## 3.2.5. Propriedades

## 3.2.5.1. Monotonicidade

A semântica nomeável é monotônica segundo a definição de monotonicidade apresentada no item 3.1.1, isto é:

Proposição 3.12. Se  $A$  e  $B$  são dois conjuntos de sentenças em uma linguagem  $L$  e  $A \subseteq B$  então

$$CN_n(A) \subseteq CN_n(B)$$

onde  $CN_n(X) = \{\sigma; X \models_n \sigma\}$

Demonstração : Sejam  $\eta(L)$  a classe das estruturas nomeáveis de  $L$ ;

$M(X)$  a classe dos modelos de um conjunto  $X$  de sentenças de  $L$  e

$\eta(X)$  a classe dos modelos nomeáveis de  $X$ .

Teremos:

$$\eta(A) = M(A) \cap \eta(L)$$

e

$$\eta(B) = M(B) \cap \eta(L)$$

E, pela monotonicidade da lógica clássica

$$M(B) \subseteq M(A)$$

e portanto

$$\eta(B) \subseteq \eta(A).$$

A semântica nomeável apresenta uma outra espécie de não-monotonicidade, que melhor seria chamá-la de "não-conservação das conseqüências nomeáveis em extensões da linguagem". Em lógica clássica se se estende a linguagem, as conseqüências lógicas de um conjunto de sentenças permanecem as mesmas, inclusive a técnica de se expandir a linguagem (para se obter modelos com propriedades desejadas) baseia-se nesta característica, sendo empregada na prova de teoremas fundamentais como a completude, Löwenheim-skolem, etc. Isto não é possível em semântica nomeável conforme a proposição abaixo.

#### 3.2.5.2. Não-conservação em extensões da linguagem

Proposição 3.13. Sejam  $L$  uma linguagem,  $\Sigma$  um conjunto de sentenças de  $L$  e  $L'$  uma expansão de  $L$ . O conjunto das conseqüências nomeáveis de  $\Sigma$  em  $L$  não coincide com o conjunto de conseqüências nomeáveis de  $\Sigma$  em  $L'$ , isto é:



$$CN'_n(\Sigma) \neq CN_n(\Sigma), \quad \text{onde}$$

$$CN'_n(X) = \{\sigma, X \models \sigma \text{ (em } L')\}$$

Exemplo 3.23. Seja  $L = \langle a \rangle$  e  $L' = \langle a, b \rangle$ ;  $a, b$  constantes

Em  $L$ ,

$$\emptyset \models_n \forall x \ x \approx a$$

Mas em  $L'$

$$\emptyset \models_n \forall x \ x \approx a$$

O leitor é convidado a mostrar que na realidade

$$CN'_n(\Sigma) \subseteq CN_n(\Sigma)$$

### 3.2.5.3. Existência de modelos nomeáveis

Já vimos que existem sentenças satisfatíveis que não possuem modelo minimal, analogamente existem sentenças satisfatíveis que não têm modelo nomeável.

Exemplo 3.24. Seja  $L = \langle a \rangle$ ;  $a$  uma constante, e a sentença

$$\sigma : \exists x \ x \neq a.$$

É fácil ver que  $\sigma$  é satisfatível mas, em  $L$ , não existe modelo nomeável de  $\sigma$ .

Em semânticas nomeáveis o caso não é tão drástico. Se uma sentença é satisfatível pode ser que não exista modelo nomeável na linguagem, mas sempre há uma expansão da linguagem  $L'$  que admite um modelo nomeável para a sentença.

Proposição 3.14. Seja  $L$  uma linguagem. Um conjunto de sentenças satisfatíveis (de  $L$ ) é nomeavelmente satisfatível em alguma expansão  $L'$  da linguagem.

Demonstração 1. Reportemo-nos à demonstração da proposição 3.4 bastando observar que o modelo  $B$  é nomeável na linguagem  $L' = L \cup \{F\varphi : \varphi = \exists x \varphi, \text{ uma fórmula de } L\}$ ;  $\{F_\varphi\}$  são as funções de Skolem de  $L$  (Ver capítulo 2).

Demonstração 2: (Via modelos de Herbrand). Na seção anterior o teorema A nos afirma que uma sentença é satisfatível sss sua forma clausal padrão é satisfatível; e o teorema B afirma que uma sentença em forma clausal tem um modelo sss tem um modelo de Herbrand, mas todo modelo de Herbrand é nomeável (na linguagem estendida).

Esta proposição é mais uma versão do teorema de Löwenhein-Skolem e é a base da atual teoria de prova automática de teoremas.

#### 3.2.5.4. Incompletude

A semântica nomeável compartilha da infinitariedade da semântica minimal, isto é, também não será possível construir uma teoria de prova (finita e efetiva) completa.

Proposição 3.15. (Incompletude da Lógica Nomeável). Não existe um algoritmo por meio do qual podemos obter precisamente o conjunto de todas conseqüências nomeáveis de um dado conjunto de sentenças.

Exemplo 3.23. Ver exemplo 3.15 lembrando que o modelo padrão da aritmética é o único modelo nomeável dos axiomas de Peano sem a indução na linguagem  $L = \langle +, \cdot, S, 0 \rangle$ .

Observação 3.4. Vimos no item 3.2.3 que podemos obter um sistema dedutivo completo para a semântica nomeável (em linguagens enumeráveis naturalmente) ao adicionarmos a regra-T ao cálculo dedutivo de primeira ordem. A regra-T é uma regra de inferência infinita.

Concluimos que o conjunto das conseqüências nomeáveis de um conjunto de sentenças não é semi-decidível.



### 3.2.5.5. Enumerabilidade dos modelos nomeáveis

É perfeitamente claro que a cardinalidade dos modelos nomeáveis de uma dada linguagem é menor ou igual à cardinalidade desta linguagem, portanto se a linguagem é enumerável também serão seus modelos nomeáveis.

## 3.3. Semântica Inicial

### 3.3.1. Estrutura natural

Definição 3.9. Seja  $\Gamma$  um conjunto de sentenças em uma linguagem  $L = \langle R, F, C \rangle$ . A estrutura natural de  $\Gamma$ ,  $H(\Gamma)$ , é a interpretação de Herbrand para  $L$  na qual as relações são definidas como a seguir.

Para todo  $P \in R$ , para todo  $h_1, \dots, h_n \in H(L)$ ,  
tem-se

$$\langle h_1, \dots, h_n \rangle \in P^{H(\Gamma)} \text{ sss } \Gamma \models_n P(h_1, \dots, h_n)$$

Intuitivamente, esta definição diz que as relações em  $H(\Gamma)$  são o mínimo permitido pelas sentenças de  $\Gamma$ . Mostraremos que de fato, quando  $H(\Gamma) \models \Gamma$ , então  $H(\Gamma)$  é o modelo mínimo no universo de Herbrand para  $\Gamma$ .

Proposição 3.16. Seja  $L = \langle R, F, C \rangle$  e  $\Gamma$  um conjunto de sentenças de  $L$  e  $H(L)$  o universo de Herbrand para  $L$ . A estrutura natural de  $\Gamma$  é o modelo mínimo no universo de Herbrand para  $\Gamma$ , em símbolos,

$$H(\Gamma) = \bigcap \text{Mod}_{H(L)}(\Gamma)$$

Demonstração  $\bigcap \text{Mod}_{H(L)}(\Gamma) \subseteq H(\Gamma)$

Seja  $P \in R$  e  $h = \langle h_1, \dots, h_n \rangle \in H(L)^n$

Queremos mostrar que

$$P \in \bigcap \text{Mod}_{H(L)}(\Gamma) \subseteq P^{H(\Gamma)}, \text{ isto é,}$$

se  $h \in \bigcap \text{Mod}_{H(L)}(\Gamma)$ , então  $h \in P^{H(\Gamma)}$

Suponha  $h \in P^H$  para todo  $H \in \text{Mod}_H(\Gamma)$

e seja  $W$  um modelo nomeável de  $\Gamma$ , pela proposição 3.11, existe uma interpretação de Herbrand isomorfa a  $W$ ,  $H(W)$ .

Assim,

$$H(W) \models \Gamma$$

e

$$h \in P^{H(W)} \quad (\text{pois } H(W) \in \text{Mod}_{H(L)}(\Gamma))$$

logo

$$H(W) \models P(h)$$

e

$$W \models P(h) \quad (\text{pois } W \approx H(W))$$

Portanto  $\Gamma \models_n P(h)$

e por conseguinte  $h \in H(\Gamma)$ .

A demonstração de que  $H(\Gamma) \subseteq \cap \text{Mod}_H(\Gamma)$  é deixada ao leitor (pode ser encontrada em [CARVALHO et al, 1980]).

Exemplo 3.24. Sejam  $L = \langle \langle, S, 0 \rangle$  e  $\Gamma$  o conjunto das sentenças abaixo:

1.  $\forall y \quad (y \approx 0 \rightarrow \exists x \ y \approx sx)$
2.  $\forall x \forall y \quad (x < sy \leftrightarrow x \leq y)$
3.  $\forall x \quad x \not\approx 0$
4.  $\forall x \forall y \quad (x < y \vee x \approx y \vee y < x)$
5.  $\forall x \forall y \quad (x < y \rightarrow y \not\approx x)$
6.  $\forall x \forall y \forall z \quad (x < y \rightarrow y < z \rightarrow x < z)$

$H(\Gamma) = (N, <, S, 0)$  com as interpretações usuais e com as abreviações:

$$1 = S0$$

$$2 = SS0$$

$$3 = SSS0$$

...



## 3.3.2. Inicialidade

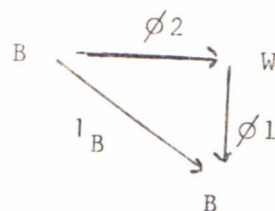
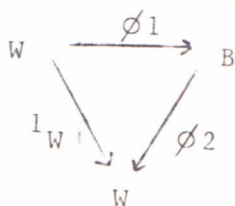
Definição 3.10. Dizemos que uma estrutura  $W$  é inicial em uma classe de estruturas se ela pertence à classe e se para toda estrutura  $B$  da classe existe um homomorfismo único de  $W$  para  $B$ .

Há uma razão muito especial para utilizarmos a noção de inicialidade. Ela possui a interessante propriedade de poder caracterizar estruturas a menos de isomorfismo.

Proposição 3.17. Se duas estruturas (nomeáveis)  $W$  e  $B$  são iniciais da mesma classe, então são isomorfas.

Demonstração. Seja  $\phi_1$  o homomorfismo único de  $W$  para  $B$  e  $\phi_2$  o homomorfismo único de  $B$  para  $W$  (lema 3.2). Mostraremos que  $\phi_1$  é de fato um isomorfismo de  $W$  para  $B$ .

Consideremos os seguintes diagramas comutativos:



onde  $l_W$  e  $l_B$  são os homomorfismos identidade em  $W$  e  $B$ , respectivamente. Pela inicialidade de  $W$ , tem-se que  $\emptyset_2 \cdot \emptyset_1 = l_W$  (uma vez que  $l_W: W \rightarrow W$  deve ser único). De forma análoga  $\emptyset_1 \cdot \emptyset_2 = l_B$ . Então,  $\emptyset_1$  (e  $\emptyset_2$ ) é um isomorfismo.

Proposição 3.18.  $H(\Gamma)$  é inicial na classe de todos os modelos nomeáveis de  $H(\Gamma)$ .

Demonstração. Seja  $W$  um modelo nomeável de  $\Gamma$  e  $H(W)$  a interpretação de Herbrand induzida por  $W$ . Sabemos que há um único homomorfismo (forte) de  $H(W)$  em  $W$ . Agora seja  $i$  a função identidade de  $H(L)$  em  $H(L)$ . Não há dúvidas de que  $i$  é um homomorfismo de  $H(\Gamma)$  de  $H(W)$  com respeito às constantes e funções. Para ver que  $i$  é um homomorfismo (fraco) com respeito às relações, lembre que  $H(\Gamma) \subseteq H(W)$  e portanto cada relação de  $H(\Gamma)$  está contida na relação correspondente de  $H(W)$ . Logo  $i \cdot \emptyset$  é um homomorfismo (fraco) de  $H(\Gamma)$  em  $W$  e, pelo lema 3.2, ele é único e sobre.

### 3.3.3. Igualdade natural

Em  $H(\Gamma)$  a única igualdade é a puramente sintática, isto é, dois elementos de  $H(L)$  são iguais em  $H(\Gamma)$  apenas se eles são o mesmo termo. De fato, gostaríamos de considerar dois elementos como iguais se eles exibem as mesmas proprie-

dades até o ponto que a linguagem consiga expressar. Em outras palavras, deveríamos considerá-los iguais se eles não podem ser distinguidos pelos predicados que temos na linguagem.

Dois objetos possuindo as mesmas propriedades expressáveis em uma linguagem significa que eles ocorrem nos mesmos lugares e nos mesmos predicados da linguagem. Mais precisamente essa igualdade pode ser definida como segue-se. Para uma dada linguagem  $L$ , seja

$$\bar{\Theta} = \text{axiomas da igualdade para } L + \left( \bigwedge_{P \in R} \bigwedge_{j=1}^n \forall x_1, \dots, \forall x_n \right.$$

$$P(x_1, \dots, x_n) [x_j/x] \leftrightarrow P(x_1, \dots, x_n) [x_j/y]$$

$$\left. \rightarrow x \approx y \right)$$

onde  $n$  é a aridade de  $P$ .

Para um modelo nomeável  $W$  de  $\Gamma$ , definimos os axiomas de particularização para  $W$  como

$$\Delta(W) = \{h_1 \approx h_2 / h_1^W \approx h_2^W; h_1, h_2 \in H(L)\}$$

Finalmente a igualdade em  $W$ ,  $\equiv [W]$ , é definida como

Para todo  $a, a' \in |W|$ ,

$$a \equiv [W] a' \text{ sss. } \Gamma \cup \bar{\Theta} \cup \Delta(W) \vdash a \approx a',$$

para algum  $h, h' \in H(L)$  tais que  $h^W \approx a$  e  $h'^W \approx a'$ .

Sendo  $\equiv [W]$  uma relação de congruência (isto decorre diretamente da congruência da  $\approx$ ), temos uma projeção natural forte na estrutura quociente  $W / \equiv [W]$ . A próxima proposição que está demonstrada em [PEQUENO, 1981, p. 24], mostra que tais identificações são consistentes.

Proposição 3.19. Sejam  $W_1$  e  $W_2$  modelos nomeáveis de  $\Gamma$ ,  $\emptyset$  um homomorfismo de  $W_1$  em  $W_2$ . Então existe um único homomorfismo  $\bar{\emptyset}$  tal que o seguinte diagrama comuta

$$\begin{array}{ccc}
 W_1 & \xrightarrow{\text{nat}(\equiv[W_1])} & W_1 / \equiv[W_1] \\
 \emptyset \downarrow & & \downarrow \bar{\emptyset} \\
 W_2 & \xrightarrow{\text{nat}(\equiv[W_2])} & W_2 / \equiv[W_2]
 \end{array}$$

$\bar{\emptyset}$  será um homomorfismo forte sss  $\emptyset$  o for).

Chamaremos a relação  $\equiv [W]$  a igualdade natural em  $W$ .



### 3.3.4. Modelos Iniciais

Definição 3.11. Seja  $\Gamma$  um conjunto de sentenças satisfeitas na estrutura natural, isto é,  $H(\Gamma) \models \Gamma$ . Dizemos que uma estrutura  $W$  é o modelo inicial de  $\Gamma$  ou que  $W$  satisfaz inicialmente  $\Gamma$  se e somente se  $W$  é isomorfo a  $H(\Gamma)/\equiv [H(\Gamma)]$ . Nesse caso a denotaremos por  $I(\Gamma)$ .

### 3.3.5. Conseqüência Lógica Inicial

Definição 3.12. Uma sentença  $\sigma$  é conseqüência lógica inicial de um conjunto de sentenças  $\Sigma$ , em símbolos  $\Sigma \models_I \sigma$  sss todo modelo inicial de  $\Sigma$  é também modelo de  $\sigma$ .

Mais uma vez analogamente ao que fazemos para as semânticas minimais e nomeáveis, introduzimos os conceitos de satisfatibilidade inicial, teoria inicial, axiomatização inicial etc. a partir dos respectivos conceitos de lógica clássica.

### 3.3.6. Propriedades

#### 3.3.6.1. Minimalidade e Nomeabilidade

Em uma linguagem  $L$ , seja  $\Gamma$  um conjunto de sentenças satisfeitas na estrutura natural  $H(\Gamma)$ .  $H(\Gamma)$  é nomeável (pois é uma particular interpretação de Herbrand) e é também um modelo mínimo no universo de Herbrand para  $\Gamma$  (conforme proposição 4.1). Portanto, o modelo inicial de  $\Gamma$ ,  $I(\Gamma)$  é nomeável e mínimo (na classe dos quocientes dos modelos nomeáveis de  $\Gamma$  pela sua correspondente igualdade natural). Logo  $I(\Gamma)$  é nomeável e minimal. Como consequência a semântica inicial herda as propriedades das duas semânticas apresentadas anteriormente. Assim, a semântica inicial é não-monotônica, não-constante em extensões da linguagem, não vale a compacidade, é infinitária etc.

#### 3.3.6.2. Inicialidade

Proposição 3.20.  $H(\Gamma)/\equiv [H(\Gamma)]$  é inicial na classe dos quocientes dos modelos nomeáveis de  $\Gamma$  pela sua correspondente igualdade natural, com homomorfismos fracos.

Demonstração. Seja  $W$  um modelo nomeável de  $\Gamma$ . Pela aplicação da proposição 4.3 obtemos o seguinte diagrama:

$$\begin{array}{ccc}
 H(W) & \xrightarrow{\text{nat } (\equiv[H(W)])} & H(W)/\equiv[H(W)] \\
 \emptyset \downarrow & & \downarrow \bar{\emptyset} \\
 W & \xrightarrow{\text{nat } (\equiv[W])} & W/\equiv[W]
 \end{array}$$

onde  $\bar{\emptyset}$  é um homomorfismo forte pois  $\emptyset$  o é.

Aplicando-se novamente a proposição 4.3 obtemos o diagrama

$$\begin{array}{ccc}
 H(\Gamma) & \xrightarrow{\text{nat } (\equiv[H(\Gamma)])} & H(\Gamma)/\equiv[H(\Gamma)] \\
 i \downarrow & & \downarrow \psi \\
 H(W) & \xrightarrow{\text{nat } (\equiv[W])} & H(W)/\equiv[H(W)]
 \end{array}$$

$i$  é homomorfismo fraco e, portanto,  $\psi$  também o é. Pela combinação dos dois diagramas concluímos que  $\bar{\emptyset} \cdot \psi$  é um homomorfismo (fraco) de  $H(\Gamma)/\equiv[H(\Gamma)]$  sobre  $W/\equiv[W]$ . Desde que  $H(\Gamma) \equiv [H(\Gamma)]$  é nomeável esse homomorfismo é único (conforme lema 3.2). E isto completa a demonstração.

## 3.3.6.3. Inexistência de modelos iniciais

Segue-se da proposição 3.16 que a propriedade pela qual  $\cap \text{Mod}_{\mathcal{H}(L)}(\Gamma)$  é um modelo de  $\Gamma$  é um requisito básico para a existência de modelos iniciais de  $\Gamma$ . [PEQUENO, 1981] chama, esta propriedade de propriedade da interseção por interpretações de Herbrand, e conjuntos de sentenças com esta propriedade de inicialmente consistentes. Evidentemente existem sentenças que não gozam desta propriedade.

Exemplo 3.25 Seja  $L = \langle P, Q, a \rangle$   $p, Q$  símbolos predicados e  $a$  símbolo constante.

Seja  $\Gamma = Pa \vee Qa$ .

Então,

$$W_1 = \langle |\{a\}|, P^{W_1} = \{a\}, Q^{W_1} = \emptyset, a^{W_1} = a \rangle$$

e

$$W_2 = \langle |\{a\}|, P^{W_2} = \emptyset, Q^{W_2} = \{a\}, a^{W_2} = a \rangle$$

são modelos de  $\Gamma$ .



Mas

$$W = W_1 \cap W_2 = \langle \{a\}, P^W = \emptyset, Q^W = \emptyset, a^W = a \rangle$$

não é modelo de  $\Gamma$ .

Há, entretanto, uma condição suficiente para que uma sentença goze desta propriedade.

Proposição 3.20. Se uma sentença é equivalente a um conjunto de cláusulas de Horn, então ela goza da propriedade de interseção por interpretações de Herbrand.

Demonstração. Seja  $\Gamma$  uma sentença equivalente a um conjunto de cláusulas de Horn,  $A$ . Suponha que  $\cap \text{Mod}_{H(L)}(\Gamma)$  não é modelo de  $A$ . Então existe uma "ground" instância  $C\sigma$  de uma cláusula  $C \in A$ , falsificada por  $\cap \text{Mod}_{H(L)}(\Gamma)$  (Ver capítulo 2). Se  $C$  é uma cláusula de Horn do tipo (3) então

$$C\sigma = B \vee \neg C_1 \vee \dots \vee \neg C_n, n \geq 0, B \notin \cap \text{Mod}_{H(L)}(\Gamma)$$

e

$$C_1, \dots, C_n \in \cap \text{Mod}_{H(L)}(\Gamma).$$

Portanto para algum  $H \in \cap \text{Mod}_{H(L)}(\Gamma)$ ,  $B \notin H$  e

$C_1, \dots, C_n \in H$ . Assim  $C$  é falso em  $H$  em contradição a suposição que  $H \in \text{Mod}_{H(L)}(\Gamma)$  e que  $\Gamma$  é equivalente a  $A$ .

Se  $C$  é uma cláusula de Horn do tipo (2) então

$$C\sigma = \bigwedge C_1 \vee \dots \vee \bigwedge C_n, n > 0,$$

$C_1, \dots, C_n \in \text{Mod}_{H(L)}(\Gamma)$ .

Portanto para todo  $H \in \text{Mod}_{H(L)}(\Gamma)$ ,  $C_1, \dots, C_n \in H$ . Assim  $C$  é falso em  $H$ , contradição.

Em [CARVALHO et al, 1980] encontramos uma demonstração que a recíproca desta proposição não é verdadeira. O problema da caracterização sintática de sentenças que possuem a propriedade de interseção por interpretação de Herbrand permanece aberto.

Nota: A demonstração da proposição 3.20 foi extraída de [ENDEM & KOWALSKI, 1976].

#### 4 - INFERÊNCIAS NÃO-MONOTÔNICAS

O principal objetivo deste capítulo é apresentar algumas formas de inferências não-monotônicas. Várias formas de inferência não-monotônica têm sido propostas ultimamente em Inteligência Artificial, deteremo-nos em duas delas: a circunscrição proposta por McCARTHY em 1980 e o "default" proposto por REITER no mesmo ano. A primeira está intimamente relacionada com a semântica minimal e a segunda ilustra muito bem os conceitos e mecanismos envolvidos em tais formas de inferências, além de ser amplamente divulgada.

Inferências não-monotônicas são inferências nas quais a introdução de novas premissas pode invalidar conclusões tiradas anteriormente. A monotonicidade é uma propriedade básica da lógica clássica. Se uma sentença  $q$  é um teorema de um conjunto de premissas  $A$  e  $A \subseteq B$ , então  $q$  é teorema de  $B$ . Com efeito uma prova a partir das premissas  $A$  é uma seqüência de sentenças cada uma das quais é, ou uma premissa ou um axioma lógico ou segue de um subconjunto das sentenças que ocorreram anteriormente na prova por uma regra de inferência. Portanto, uma prova a partir de  $A$  é também uma prova a partir de  $B$ .

Sistema de raciocínio em Inteligência Artificial apelam constantemente a inferências não-monotônicas tanto os

sistemas que tentam simular o senso comum como aqueles que devem planejar e raciocinar no mundo real, pois estes sistemas normalmente trabalham com informações incompletas e são obrigados a tirar conclusões, mesmo que não tenham provas, mas que pareçam "plausíveis". Este é o caso de um "robot" que locomove-se ao local onde se espera que seu carro esteja, em vez de ficar imobilizado pensando sobre as infinitas circunstâncias nas quais o carro poderia não mais estar lá. De um ponto de vista formal não existe maneira de se provar que o carro ainda está lá, mas, em todo caso, ele deve agir e assim o faz baseado unicamente em suas suposições plausíveis. Assim, é de se esperar que de vez em quando algumas dessas suposições estejam erradas originando conclusões indevidas. Conclusões sancionadas por inferências não-monotônicas são melhor vistas como crenças que podem ser modificadas ou rejeitadas por subseqüentes observações. O sistema deve ser, portanto, preparado para rever suas crenças.

O "problema da qualificação" imediatamente surge na representação do conhecimento do senso comum. Aparentemente, a fim de se representar completamente as condições necessárias para o desempenho de uma ação, um número impraticável e implausível de qualificações teria que ser incluído nas sentenças que as expressam. Por exemplo, o uso de um barco para atravessar um rio requer, se o barco é um bote de remos, que os remos e as torleteiras estejam presentes e que se encaixem. Muitas outras qualificações podem ser tomadas, fazendo as regras para o uso de um bote de remos impraticáveis e



98  
ainda assim alguém poderia pensar em um requerimento adicional ainda não incluído.

A circunscrição [McCARTHY, 1980], é uma regra de inferência não-monotônica que permite a uma pessoa ou a um programa postular certas conclusões a partir da suposição que "os objetos que se pode determinar que tenham certas propriedades são as únicas que as têm". Por exemplo, podíamos postular que um bote pode ser usado para atravessar um rio a menos que "alguma coisa" o impeça. A circunscrição podia ser usada para conjecturar que as únicas entidades que podem impedir o uso de um bote são aquelas cuja existência segue-se dos fatos que temos em mãos. Se não faltam remos, ou outra circunstância impedindo o uso não é deduzível, então conclui-se que o bote pode ser usado. A correção desta conclusão dependerá de termos levado em conta todos os fatores relevantes quando fizemos a circunscrição.

O resultado de se aplicar a circunscrição a uma coleção de fatos A é uma sentença-esquema que afirma que as únicas uplas satisfazendo um predicado  $P(x_1, \dots, x_n)$  são aquelas cuja satisfação a P segue-se das sentenças de A. Desde que somando-se mais sentenças a A poderia fazer P aplicável a mais uplas, a circunscrição é não-monotônica, conclusões derivadas de circunscrições são baseadas na conjectura que A inclui todos os fatos relevantes e que os objetos, cuja existência segue-se de A são todos os objetos relevantes.

Sistemas Inteligentes usualmente tiram conclusões baseadas em padrões de inferência da forma "na falta de qual-

quer informação em contrário assuma ...". Padrões de raciocínio como este representam uma forma de plausíveis inferências e são tipicamente requeridos quando conclusões devem ser tiradas a despeito da falta de total conhecimento da situação. Tais padrões de inferência correspondem ao que chamamos de raciocínio por "default" e foi formalizado por REITER em 1980. A principal questão é o que significa "falta de informação em contrário", [REITER, 1980] formaliza da seguinte maneira: "se é consistente assumir-se uma sentença  $\alpha$ , então assumo  $\alpha$ ". Mas ainda permanece um problema de interpretação. Exatamente o que significa o requerimento de consistência associado com um "default"? Consistente com que? Este é o principal ponto em definir-se uma lógica para o raciocínio por "default". Na seção 4.2 analisamos cuidadosamente a formalização de [REITER, 1980]. Uma boa interpretação intuitiva é que requer-se consistência com respeito a todos os fatos de primeira ordem a respeito da situação, as sim como com respeito a todas outras crenças sancionadas por todos os outros "defaults".

Como exemplo do uso da inferência por "default" considere um sistema dedutivo de "pergunta e resposta" no qual consta um banco de dados representando o catálogo de uma companhia aérea. Seja a pergunta "O voo 263 da VASP conecta Fortaleza ao Rio?" O sistema tipicamente tentaria provar CONECTA (V263, FORT, RIO) a partir do banco de dados e algum conhecimento geral sobre linhas aéreas. Se a prova suceder-se o sistema responderia "sim". Caso contrário, ele respon-

deria "não". A falha em achar a prova sancionou a inferência. Mas da falha em provar-se CONECTA (V263, FORT, RIO) não se pode concluir ordinariamente que:

] CONECTA (V263, FORT, RIO).

O sistema raciocinou com o seguinte "default": "É consistente afirmar que VÔO 263 não conecta Fortaleza ao Rio, então VÔO 263 não conecta Fortaleza ao Rio".

De uma vez que uma conclusão sancionada por "default" pode mais tarde ser rejeitada, a inferência por "default" é não-monotônica. Por exemplo, se subseqüentemente fosse descoberto que o VÔO 263 conecta Fortaleza ao Rio, não mais seria consistente afirmar que "o VÔO 263 não conecta..." e a conclusão teria que ser rejeitada.

[McDERMOTT & DOYLE, 1980] propõem uma lógica estendida pela inclusão de um símbolo modal  $M$  representando consistência. Assim a fórmula  $M\alpha$  é lida como uma afirmativa de que  $\alpha$  é consistente. Esta abordagem segue uma tradição da lógica modal e tem paralelo com o uso do operador modal "tempo" para axiomatização de fatos envolvendo seqüências temporais.

Do ponto de vista da construção de sistemas de raciocínio efetivos existe um problema significativo compartilhado por todas estas abordagens. Contrariamente à lógica de primeira ordem cada uma destas extensões requer a prova de um ponto-fixo não construtivo. McDERMOTT e DOYLE usam a noção



de ponto-fixo explicitamente, definindo "provabilidade" em termos de um ponto-fixo (Ver [McDERMOTT & DOYLE, 1980]). REITER similarmente define a noção de uma "extensão" de uma teoria "default", com uma teoria de prova apelando para um teste de satisfatibilidade envolvendo todo o corpo de axiomas (Ver [REITER, 1980]). McCARTHY não discute provas em detalhe, mas em sua formalização em termos de complementação minimal e inferência circumscriptiva existe a noção de minimalidade a qual cumpre o mesmo papel que o ponto-fixo em McDERMOTT & DOYLE. Ver seções 4.1 e 4.2.

Existe uma clara razão para este problema. Cada um destes formalismos provê alguma coisa equivalente a uma inferência não-monotônica da forma: "Se A é consistente com o que é conhecido, então infira B". Em lógica de primeira ordem clássica, não é decidível se uma fórmula arbitrária é um teorema mas é claramente decidível se uma seqüência de sentenças é uma prova.

Uma vez que uma regra de inferência do tipo acima é introduzida, não mais é possível demonstrar em geral que alguma coisa é uma prova, desde que um passo da prova envolve decidir a satisfatibilidade de uma fórmula arbitrária. Intuitivamente estamos tentando prover um passo do raciocínio da forma: "Se você não conhece X, então Y", onde conhecer X é definido em termos de provabilidade, e portanto, não existe uma maneira de decidir se conhecemos X ou não.

[BOBROW & WINOGRAD, 1977] propõem um sistema que implementa um tipo de raciocínio que depende da quantidade (e



qualidade) de computação despendida em uma dedução. A questão de se saber se  $\alpha$  é consistente é convertida em perguntar-se se  $\alpha$  ou  $\neg \alpha$  podem ser concluída na base de alguma computação limitada. Isto provê uma maneira efetiva de se fazer inferências não-monotônicas, desde que decidir a consistência é substituído por avaliar alguma coisa diretamente computável.

Uma classe de sistemas chamados de "Sistemas Mantenedores da Verdade" \*o "TMS", tem sido desenvolvida por DOYLE entre outros (Ver [DOYLE, 1979]). Um "TMS" é um sistema formal de "restrições" entre os objetos representando os teoremas de uma teoria de primeira ordem. As regras de inferência da lógica podem ser usadas para gerar novas "restrições" tendo a forma: "Se você acredita em A, então você deve acreditar em B". Em um momento, o TMS terá um conjunto finito de teoremas e restrições. Pode então ser efetivamente implementada a regra de inferência não-monotônica: "Se A é consistente, então B" desde que consistência é julgada com respeito ao conjunto corrente de teoremas, e não com respeito a tudo provável a partir dele. A adição de uma nova restrição pode fazer um julgamento prévio de consistência inválido, e isto ocasionará o reajustamento de todo o corpo de crenças. O objetivo em se construir um TMS é fazer este processo consistente e computacionalmente praticável. Os algoritmos para o teste de consistência são complexos pois o conjunto inclui tanto as inferências não-monotônicas quanto os teoremas de primeira ordem.

---

\* Em inglês, "Truth Maintenance Systems".

Na seção 1 apresentamos a circunscrição de McCARTHY em suas duas formalizações: circunscrição de domínios e circunscrição de predicados. Fazemos a formalização, damos exemplos, estudamos suas propriedades e analisamos a relação entre semântica minimal e circunscrição.

Na seção 2 apresentamos a inferência por "default" de REITER, formalização, exemplos e propriedades.

#### 4.1 - Circunscrição

##### 4.1.1 - Circunscrição do domínio

Circunscrição é uma regra de inferência formalizada que pode ser utilizada juntamente com as regras de inferência da lógica de primeira ordem. Circunscrição do domínio conjectura que as entidades conhecidas são todas que existem. O resultado de aplicar-se a circunscrição do domínio a uma coleção de fatos  $A$  é a sentença-esquema que afirma que os únicos elementos que existem são aqueles cuja existência segue-se das sentenças de  $A$ . Adicionando-se novas sentenças a  $A$  pode-se obrigar a existência de mais elementos, assim a circunscrição do domínio é uma inferência não-monotônica. A circunscrição do domínio é um caso particular da circunscrição de predicados.

Definição 4.1. Seja  $A$  uma sentença da linguagem  $L$  e  $\Phi$  uma expressão predicativa (um símbolo predicado ou uma  $\lambda$ -expressão). A relativização de  $A$  com respeito a  $\Phi$ , representada por  $A^\Phi$ , é a sentença formada substituindo-se os quantificadores universais  $\forall x$  de  $A$  por  $\forall x. \Phi(x) \rightarrow$ , e os quantificadores existenciais  $\exists x$  de  $A$  por  $\exists x. \Phi(x) \wedge$ .

Exemplo 4.1. Seja  $L = \langle P, f, a, b \rangle$ ,  $P$  símbolo predicado,  $f$  símbolo funcional e  $a, b$  símbolos constantes. Seja  $A$  a conjunção das sentenças 1 a 3 abaixo.

1.  $\forall x. (P(x) \rightarrow P(f(x)))$
2.  $f(a) \approx b$
3.  $f(b) \approx a$

A relativização de  $A$  com respeito a  $\Phi(x) = (P(x) \rightarrow x \approx a \vee x \approx b)$  é a conjunção das sentenças

- 1 $^\Phi$ .  $\forall x. ((x \approx a \vee x \approx b) \rightarrow (P(x) \rightarrow P(f(x))))$
- 2 $^\Phi$ .  $f(a) \approx b$
- 3 $^\Phi$ .  $f(b) \approx a$

Lema 4.1. Se  $A$  é uma sentença prenexa puramente universal, então  $A \vdash A^\Phi$ .

Demonstração: Trivial.

Definição 4.2. Se  $\Phi$  é uma expressão predicativa em uma linguagem L, o fechamento de  $\Phi$  em L representado por Axiom( $\Phi$ ), é a conjunção das sentenças  $\Phi(c)$  para cada constante c de L, e das sentenças  $\forall \bar{x}(\Phi(\bar{x}) \rightarrow \Phi(f(\bar{x})))$  para cada símbolo funcional de aridade n e  $\bar{x} = (x_1, \dots, x_n)$ .

Exemplo 4.2. Sejam L e  $\Phi$  como no exemplo anterior. Então:

$$\begin{aligned} \text{Axiom}(\Phi) = & (a \approx a \vee a \approx b) \wedge (b \approx a \vee b \approx b) \wedge \\ & \wedge \forall x . (x \approx a \vee x \approx b \rightarrow f(x) \approx a \vee f(x) \approx b) \end{aligned}$$

Definição 4.3 Sejam A uma sentença e  $\Phi$  uma expressão predicativa em uma linguagem L, a circunscrição do domínio de A com respeito a  $\Phi$  é a sentença

$$\text{Axiom}(\Phi) \wedge A^\Phi \rightarrow \forall x . \Phi(x) .$$

Exemplo 4.3. No exemplo 4.1 a circunscrição do domínio de A com respeito a  $\Phi$  é a sentença:

$$\begin{aligned} \text{Axiom}(\Phi) \{ & (a \approx a \vee a \approx b) \wedge (b \approx a \vee b \approx b) \wedge \\ A^\Phi & \left\{ \begin{array}{l} \wedge \forall x . (x \approx a \vee x \approx b \rightarrow f(x) \approx a \vee f(x) \approx b) \wedge \\ \wedge \forall x . ((x \approx a \vee x \approx b \rightarrow (P(x) \rightarrow P(f(x)))) \wedge \\ \wedge f(a) \approx b \wedge f(b) \approx a \rightarrow \forall x . (x \approx a \vee x \approx b) \end{array} \right. \end{aligned}$$



Observe que  $A \vdash \text{Axiom}(\Phi) \wedge A^{\Phi}$ , e portanto de  $A$  união a circunscrição do domínio de  $A$  segue-se  $\forall x . (x \approx a \vee x \approx b)$ .

#### 4.1.1.1 Inferência minimal

Definição 4.4. Seja  $A$  uma sentença de uma linguagem  $L$ . Define  $\Omega(A)$  como o conjunto das circunscrições do domínio de  $A$  com respeito a toda expressão predicativa  $\Phi$ , em símbolos:

$$\Omega(A) = \{ \text{Axiom}(\Phi) \wedge A^{\Phi} \rightarrow \forall x . \Phi(x) \}$$

onde  $\Phi$  percorre todas expressões predicativas de  $L$ .

A complementação minimal de  $A$ , escrito  $\text{CM}(A)$ , é a teoria cujos axiomas são os elementos do conjunto  $\Omega(A) \cup \{A\}$ . Em símbolos,

$$\text{CM}(A) = \{ \sigma \in L; \Omega(A) \cup \{A\} \vdash \sigma \}$$

Uma sentença  $B \in L$  é minimalmente inferida de  $A$ , representado por  $A \vdash_m B$ , quando  $B$  pertence à complementação minimal de  $A$ , em símbolos  $\vdash_{\text{CM}(A)} B$ .

Exemplo 4.4. Dos exemplos da seção anterior depreende-se que

$$\forall x(P(x) \rightarrow P(f(x))) \wedge f(a) \approx b \wedge f(b) \approx a \vdash_m \forall x(x \approx a \vee x \approx b)$$

Exemplo 4.5. Seja  $L = \langle P, a, b, c \rangle$  e  $A$  a sentença  $P(a) \wedge P(b) \wedge P(c)$ . Mostraremos que

$$A \vdash_m \forall x(P(x) \rightarrow x \approx a \vee x \approx b \vee x \approx c)$$

Tome  $\Phi(x) = (P(x) \rightarrow x \approx a \vee x \approx b \vee x \approx c)$

Então,

$$\begin{aligned} \text{Axiom}(\Phi) &= (P(a) \rightarrow a \approx a \vee a \approx b \vee a \approx c) \wedge \\ &\quad \wedge (P(b) \rightarrow b \approx a \vee b \approx b \vee b \approx c) \wedge \\ &\quad \wedge (P(c) \rightarrow c \approx a \vee c \approx b \vee c \approx c). \end{aligned}$$

$$A(\Phi) = A.$$

E,

A circunscrição do domínio de  $A$  com respeito a  $\Phi$  é

$$\text{Axiom}(\Phi) \wedge A^\Phi \rightarrow \forall x(P(x) \rightarrow x \approx a \vee x \approx b \vee x \approx c)$$

Observe que  $\text{Axiom}(\Phi)$  é uma tautologia ( $\emptyset \vdash \text{Axiom}(\Phi)$ ).

Portanto

$$A \vdash \text{Axiom}(\Phi) \wedge A^\Phi$$

E

$$\vdash_{CM(A)} \forall x (P(x) \rightarrow x \approx a \vee x \approx b \vee x \approx c)$$

logo

$$P(a) \wedge P(b) \wedge P(c) \vdash_m \forall x (P(x) \rightarrow x \approx a \vee x \approx b \vee x \approx c)$$

Exemplo 4.6. Seja  $L = \langle P \rangle$  e  $A = \exists x . P(x)$ .

$$\text{Axiom}(\emptyset) = \emptyset$$

$$A^{\emptyset} = \exists x . \emptyset(x) \wedge P(x)$$

$$\Omega(A) = \{\exists x . \emptyset(x) \wedge P(x) \rightarrow \forall x . \emptyset(x)\}$$

$$\text{Tome } \emptyset(x) = P(x)$$

$$\text{Então, } \Omega(A) \vdash \exists x . P(x) \rightarrow \forall x . P(x)$$

$$\text{e portanto } \vdash_{CM(A)} \forall x . P(x)$$

$$\text{logo, } \exists x . P(x) \vdash_m \forall x . P(x).$$

Exemplo 4.7. A aritmética

Sejam  $L = \langle +, \cdot, S, 0 \rangle$  e os axiomas (1) a (6) como no capítulo 2.

- (1)  $\forall x \quad (0 \neq Sx)$   
 (2)  $\forall xy \quad (Sx \approx Sy \rightarrow x \approx y)$   
 (3)  $\forall x \quad (x + 0 \approx x)$   
 (4)  $\forall xy \quad (x + Sy \approx S(x + y))$   
 (5)  $\forall x \quad (x \cdot 0 \approx 0)$   
 (6)  $\forall xy \quad (x \cdot Sy \approx (xy) + x)$

Neste caso

$$\begin{aligned} \text{Axiom}(\phi) = & \phi(0) \wedge \forall x(\phi(x) \rightarrow \phi(Sx)) \wedge \\ & \wedge \forall xy (\phi(x) \wedge \phi(y) \rightarrow \phi(x + y)) \wedge \\ & \wedge \forall xy (\phi(x) \wedge \phi(y) \rightarrow \phi(x \cdot y)) \end{aligned}$$

E circunscrevendo-se o domínio da sentença identicamente verdadeira,  $A \equiv V$  teríamos

$$\text{Axiom}(\phi) \rightarrow \forall x \cdot \phi(x)$$

que é uma versão mais fraca do axioma esquema da indução. Mas, podíamos reescrever novos axiomas na linguagem  $\langle A, P, S, 0 \rangle$  onde substituiríamos os símbolos funcionais  $+$ ,  $\cdot$  pelos símbolos predicados  $A$  e  $P$ , respectivamente, e interpretando-se

$$A(x, y, z) \quad \text{como} \quad "x + y \approx z" \quad \text{e}$$

$$P(x, y, z) \quad \text{como} \quad "x \cdot y \approx z"$$



Dessa maneira os axiomas 3, 4, 5 e 6 se escreveriam:

$$(3') \quad \forall x . A(x, 0, x)$$

$$(4') \quad \forall x y z u (A(x, Sy, z) \wedge A(x, y, u) \rightarrow z \approx Su)$$

$$(5') \quad \forall x P(x, 0, 0)$$

$$(6') \quad \forall x y z u (P(x, Sy, z) \wedge P(x, y, u) \rightarrow A(v, x, u))$$

Agora,

$$\text{Axiom}(\phi) = \phi(0) \wedge (\phi(x) \rightarrow \phi(Sx))$$

E a circunscricao de  $A \equiv V$  é a seguinte

$$\phi(0) \wedge \forall x(\phi(x) \rightarrow \phi(Sx)) \rightarrow \forall x \phi(x)$$

que é o axioma esquema da indução.

Portanto, se  $\Gamma$  é a conjunção dos axiomas 1, 2, 3 a 6, então,  $\Gamma \cup \text{CM}(A)$  é a aritmética de Peano. De fato, a idéia por trás da noção de "completamentação minimal" é "completar" qualquer conjunto de axiomas da maneira que o axioma-esquema da indução completa os axiomas de Peano.

## 4.1.2. Circunscrição de predicados

A circunscrição dos predicados é uma regra de conjectura que assume que as entidades que satisfazem um dado predicado são as únicas exigidas por uma determinada coleção de sentenças.

O resultado da aplicação da circunscrição a uma coleção de sentenças  $A$  é uma sentença-esquema que afirma que as únicas  $n$ -uplas satisfazendo um símbolo predicado  $P(x_1, \dots, x_n)$  são aquelas exigidas pelas sentenças de  $A$ . De uma vez que somando-se mais sentenças a  $A$  pode-se fazer  $P$  aplicável a mais uplas, a circunscrição de predicados não é uma regra monotônica. As conclusões derivadas de uma circunscrição são conjecturas baseadas em que  $A$  inclui todos os fatos relevantes e que os objetos cuja existência segue de  $A$  são todos os objetos relevantes. Circunscrição de predicados generaliza a circunscrição dos domínios:

Definição 4.5. Seja  $A$  uma sentença contendo um símbolo predicado  $P(x_1, \dots, x_n)$ , ou mais sucintamente  $P(\bar{x})$ . Escrevemos  $A(\phi)$  para o resultado de substituir-se todas ocorrências de  $P$  em  $A$  pela expressão predicativa  $\phi$ . A circunscrição de  $P$  em  $A(P)$  é a sentença-esquema:

$$(A(\phi) \wedge \forall \bar{x}(\phi(\bar{x}) \rightarrow P(\bar{x}))) \rightarrow \forall \bar{x}(P\bar{x}) \rightarrow \phi(\bar{x})$$

Observe que esta sentença-esquema contém um parâmetro predicativo  $\phi$  o qual podemos substituir por uma expressão predicativa arbitrária. E, como ela é uma implicação, temos que se tivermos o antecedente, a conjunção, poderemos concluir o conseqüente. A primeira parte da conjunção,  $A(\phi)$ , expressa a suposição que  $\phi$  satisfaz as condições satisfeitas por  $P$ . A segunda parte,  $\forall x(\phi(x) \rightarrow P(x))$ , por sua vez, expressa a suposição que  $\phi \subseteq P$ , isto é, as  $n$ -uplas que satisfazem  $\phi$  são um subconjunto das  $n$ -uplas que satisfazem  $P$ . O conseqüente expressa o inverso, isto é,  $P \subseteq \phi$ . Portanto, podemos concluir que, neste caso,  $\phi$  e  $P$  coincidem. Em suma, a sentença-esquema da circunscrição de  $P$  em  $A(P)$  expressa o fato que as únicas  $n$ -uplas que satisfazem  $P$  são as que são obrigadas pela sentença  $A$ .

#### 4.1.2.1. Inferência circunsriptiva

Definição 4.6. Uma sentença  $B$  é inferida circunsriptivamente de uma sentença  $A$ , representado por  $A \vdash_P B$ , se  $B$  pode ser obtida (por dedução) a partir de  $A$  e da circunscrição de algum símbolo predicado  $P$  em  $A$ .

Exemplo 4.8. Seja uma sentença  $A = P(a) \wedge P(b) \wedge P(c)$

Circunscrevendo-se  $P$  em  $A$  temos o esquema:

$$(\Phi(a) \wedge \Phi(b) \wedge \Phi(c) \wedge \forall x(\Phi(x) \rightarrow P(x))) \rightarrow \forall x(P(x) \rightarrow \Phi(x))$$

Tome  $\Phi(x) = (x \approx a \vee x \approx b \vee x \approx c)$  e

substitua no esquema, obtemos:

$$\begin{aligned} & ((a \approx a \vee a \approx b \vee a \approx c) \wedge (b \approx a \vee b \approx b \vee b \approx c) \wedge \\ & \wedge (c \approx a \vee c \approx b \vee c \approx c) \wedge \forall x(x \approx a \vee x \approx b \vee x \approx c \rightarrow \\ & \rightarrow P(x))) \rightarrow \forall x(P(x) \rightarrow x \approx a \vee x \approx b \vee x \approx c) \end{aligned}$$

que simplifica-se para

$$\forall x(x \approx a \vee x \approx b \vee x \approx c \rightarrow P(x)) \rightarrow \forall x(P(x) \rightarrow x \approx a \vee x \approx b \vee x \approx c)$$

De A segue-se o antecedente desta implicação, e isto nos permite concluir

$$\forall x(P(x) \rightarrow x \approx a \vee x \approx b \vee x \approx c) \quad (1)$$

ou melhor

$$P(a) \vee P(b) \vee P(c) \vdash_P \forall x(P(x) \rightarrow x \approx a \vee x \approx b \vee x \approx c)$$

(1) afirma que os únicos elementos em P são a, b e c, justamente os elementos que A requer estarem em P.



Exemplo 4.9. Seja uma sentença

$$A = P(a) \vee P(b)$$

Circunscrevendo-se  $P$  em  $A$  temos o esquema:

$$(\Phi(a) \vee \Phi(b) \wedge \forall x(\Phi(x) \rightarrow P(x))) \rightarrow \forall x(P(x) \rightarrow \Phi(x))$$

Tome  $\Phi(x) = x \approx a$

Então, teremos:

$$((a \approx a \vee b \approx a) \wedge \forall x(x \approx a \rightarrow P(x))) \rightarrow \forall x(P(x) \rightarrow x \approx a)$$

que simplifica-se para

$$P(a) \rightarrow \forall x(P(x) \rightarrow x \approx a) \quad (2)$$

Agora tome  $\Phi(x) = x \approx b$ .

Teremos:

$$((a \approx b \vee b \approx b) \wedge \forall x(x \approx b \rightarrow P(x))) \rightarrow \forall x(P(x) \rightarrow x \approx b)$$

e simplificando, obtemos

$$P(b) \rightarrow \forall x(P(x) \rightarrow x \approx b) \quad (3)$$

$$A \cup \{(1)\} \cup \{(2)\} \vdash \forall x(P(x) \rightarrow x \approx a) \vee \forall x(P(x) \rightarrow x \approx b)$$

e portanto,

$$P(a) \vee P(b) \vdash_P \forall x(P(x) \rightarrow x \approx a) \vee \forall x(P(x) \rightarrow x \approx b)$$

Exemplo 4.10. Seja  $S$  uma relação, por exemplo, binária e  $\bar{S}$  uma outra relação (binária) tal que:

$$\forall xy . S(x,y) \rightarrow \bar{S}(x,y); \quad (4)$$

$$\forall xyz \bar{S}(x,y) \wedge \bar{S}(y,z) \rightarrow \bar{S}(x,z); \quad (5)$$

(5) nos diz que  $S$  é transitiva.

Podemos usar a circunscrição de predicados para definirmos  $\bar{S}$  como o fecho transitivo de  $S$ .

Com efeito, circunscrevendo-se  $\bar{S}$  em (4) e (5) obtemos o esquema:

$$\begin{aligned} & (\forall xy . S(x,y) \rightarrow \phi(x,y) \wedge \\ & \wedge \forall xyz (\phi(x,y) \wedge \phi(y,z) \rightarrow \phi(x,z)) \wedge \\ & \wedge \forall xy (\phi(x,y) \rightarrow \bar{S}(x,y))) \rightarrow \\ & \rightarrow \forall xy (\bar{S}(x,y) \rightarrow \phi(x,y)) \end{aligned}$$

Tome  $\phi(x,y) = S(x,y) \vee \exists t(\bar{S}(x,t) \wedge S(t,y))$

e substitua no esquema acima, teremos

$$\begin{aligned}
& (\forall xy (S(x,y) \rightarrow S(x,y) \vee \exists t(\bar{S}(x,t) \wedge S(t,y))) \wedge \\
& \wedge \forall xyz (S(x,y) \vee \exists t(\bar{S}(x,t) \wedge S(t,y)) \wedge S(y,z) \vee \exists t(\bar{S}(y,t) \wedge S(t,z)) \rightarrow \\
& \rightarrow S(x,z) \vee \exists t(\bar{S}(x,t) \wedge S(t,z))) \wedge \\
& \wedge \forall xy (S(x,y) \vee \exists t(\bar{S}(x,t) \wedge S(t,y)) \rightarrow \bar{S}(x,y)) \rightarrow \\
& \rightarrow \forall xy (\bar{S}(x,y) \rightarrow S(x,y) \vee \exists t(\bar{S}(x,t) \wedge S(t,y)))
\end{aligned}$$

De (4) e (5) segue-se o antecedente desta implicação, o que nos deixa concluir que

$$\forall xy (\bar{S}(x,y) \rightarrow S(x,y) \vee \exists t(\bar{S}(x,t) \wedge S(t,y))) \quad (6)$$

É esta é justamente a definição de  $\bar{S}$  como o fecho transitivo de  $S$ .

Em suma,

$$\begin{aligned}
& \forall xy. S(x,y) \rightarrow \bar{S}(x,y) \wedge \\
& \wedge \forall xyz (\bar{S}(x,y) \wedge \bar{S}(y,z) \rightarrow \bar{S}(x,z)) \quad |-\bar{S} \\
& |-\bar{S} \forall xy (\bar{S}(x,y) \rightarrow S(x,y) \vee \exists t(\bar{S}(x,t) \wedge S(t,y))).
\end{aligned}$$

Exemplo 4.11. A aritmética. Adicione à linguagem da aritmética, o predicado  $T$  e junte aos axiomas de Peano (sentenças 1 a 6 do exemplo 4.7) o axioma

$$\forall x. T(x)$$

(\*)

De (\*) segue-se a sentença

$$T(0) \wedge \forall x (T(x) \rightarrow T(S(x))) \quad (7)$$

Circunscrevendo-se o predicado T em (7) temos:

$$\begin{aligned} & \Phi(0) \wedge \forall x (\Phi(x) \rightarrow \Phi(S(x))) \wedge \\ & \wedge (\forall x (\Phi(x) \rightarrow T(x)) \rightarrow \forall x (T(x) \rightarrow \Phi(x))) \end{aligned} \quad (8)$$

Usando-se (\*); (8) simplifica-se para

$$\Phi(0) \wedge \forall x (\Phi(x) \rightarrow \Phi(S(x)) \rightarrow \forall x. \Phi(x)) \quad (9)$$

(9) é justamente o axioma-esquema da indução.

Dessa maneira temos adicionados aos axiomas de Peano, o axioma de indução e com isto construído a Aritmética de Peano.

A circunscrição de predicados generaliza a circunscrição dos domínios. Seja uma sentença A relativando-se A com respeito a um novo predicado T e circunscrevendo-se T em  $A^T \wedge \text{Axiom}(T)$  obtemos

$$\text{Axiom}(\Phi) \wedge A^\Phi \wedge \forall x (\Phi(x) \rightarrow T(x)) \rightarrow \forall x (T(x) \rightarrow \Phi(x)) \quad (10)$$

Adicione o axioma  $\forall x. T(x)$  (11)



Simplificando-se (10) obtém-se

$$\text{Axiom}(\Phi) \wedge A^{\Phi} \rightarrow \forall x . \Phi(x) \quad (12)$$

(12) é a circunscrição do domínio de A com respeito a  $\Phi$ .

Exemplo 4.12. Seja A a sentença  $\exists x . P(x)$ .

Relativando-se A com respeito ao predicado T obtemos

$$\exists x(T(x) \wedge P(x)) \quad (13)$$

Circunscrevendo-se T em (13) obtemos

$$\exists x(\Phi(x) \wedge P(x)) \wedge \forall x(\Phi(x) \rightarrow T(x)) \rightarrow \forall x(T(x) \rightarrow \Phi(x)) \quad (14)$$

Adicionando-se o axioma  $\forall x . T(x)$  e simplificando-se (14) temos

$$\exists x(\Phi(x) \wedge P(x)) \rightarrow \forall x . \Phi(x) \quad (15)$$

(15) é a circunscrição do domínio da sentença

$A = \exists x . P(x)$  com respeito a  $\Phi$ .

### 4.1.3. Propriedades

A circunscrição (tanto de domínios quanto de predicados) é uma regra de conjectura que pode ser usada ao lado das regras de inferência da lógica de primeira ordem. O sistema dedutivo da lógica fica assim com maior poder de prova. O maior poder de prova da lógica estendida por circunscrição acarretará na não-monotonicidade e na não-construtividade desta lógica. Além disso, ocasionalmente a circunscrição poderá ser inconsistente.

#### 4.1.3.1. Não-Monotonicidade

A circunscrição dos domínios de uma sentença  $A$  conjectura que as entidades "conhecidas" são todas as existentes. Desde que somando-se mais sentenças a  $A$  pode-se aumentar o número de entidades "conhecidas", a circunscrição dos domínios de  $A$  não é monotônica.

Proposição 4.1. A inferência minimal é uma inferência não-monotônica.

Exemplo 4.13. Some-se à sentença  $A$  dos exemplos 4.1 a 4.4, a sentença

$$B = \exists x (x \neq a \wedge x \neq b)$$

É óbvio que

$$\{A\} \cup \{B\} \not\vdash_m \forall x(x \approx a \vee x \approx b)$$

Exemplo 4.14. Expandindo-se a linguagem  $L$  do exemplo 4.5 a  $L' = L \cup \{d\}$ ;  $d$  um símbolo constante e somando-se à sentença  $A$ ,  $B = P(d)$  temos que

$$\{A\} \cup \{B\} \not\vdash_m \forall x(P(x) \rightarrow x \approx a \vee x \approx b \vee x \approx c)$$

A sentença-esquema que resulta da aplicação da circunscrição de predicados de uma sentença  $A$ , afirma que as únicas uplas a satisfazem um predicado  $P$ , são as "exigidas" pela sentença  $A$ . Desde que somando-se mais sentenças a  $A$  pode-se fazer que  $P$  aplique-se a mais uplas, a circunscrição de predicados de  $A$  é não-monotônica.

Proposição 4.2. A inferência circunscritiva é uma inferência não-monotônica.

Exemplo 4.15. Somando-se à sentença  $A$  do exemplo 4.8 a sentença  $B = P(d)$  temos que

$$\{A\} \cup \{B\} \not\vdash_p \forall x(P(x) \rightarrow x \approx a \vee x \approx b \vee x \approx c)$$

Exemplo 4.16. Do exemplo 4.9 conclui-se que

$$(P(a) \vee P(b)) \wedge P(a) \not\vdash_P \neg P(b)$$

Mas, claramente

$$(P(a) \vee P(b)) \wedge P(a) \wedge P(b) \not\vdash_P \neg P(b)$$

#### 4.1.3.2. Possível inconsistência da circunscrição

Lembre-se que uma sentença é inconsistente, se a partir dela é possível deduzir-se uma contradição. Como o poder de dedução foi aumentado, torna-se mais fácil deduzir-se uma contradição e por conseguinte algumas sentenças anteriormente (na lógica de primeira ordem padrão) consistentes, tornam-se inconsistentes com o novo sistema dedutivo.

Proposição 4.3. Um conjunto consistente de sentenças pode ter uma complementação minimal inconsistente.

Exemplo 4.17. Em  $L = \langle S \rangle$ ;  $S$  um símbolo predicado, seja  $A$  a conjunção das sentenças 1 a 4 abaixo:

1.  $\forall x. \exists y. S(x, y)$
2.  $\exists y. \forall x. \neg S(x, y)$
3.  $\forall xyz (S(x, y) \wedge S(x, z) \rightarrow y \approx z)$
4.  $\forall xyz (S(y, x) \wedge S(z, x) \rightarrow y \approx z)$



Circunscreva o domínio da sentença A com relação a

$$\Phi(t) = \exists u. s(u, t)$$

teremos

$$\text{Axiom}(\Phi) \wedge A^\Phi \rightarrow \forall t. \exists u. S(u, t);$$

$$\text{Axiom}(\Phi) = \emptyset \quad (\text{vazio})$$

e

$$1^\Phi = \forall x (\exists u. S(u, x) \rightarrow \exists y (\exists u. S(u, y) \wedge S(x, y)))$$

$$2^\Phi = \exists y (\exists u. S(u, y) \wedge \forall x (\exists u. S(u, x) \rightarrow \neg S(x, y)))$$

Não é difícil ver que  $A \vdash 1^\Phi \wedge 2^\Phi$ .

Observe ainda que como 3 e 4 são sentenças prenexas puramente universais  $3 \vdash 3^\Phi$  e  $4 \vdash 4^\Phi$  (Lema 4.1)

Assim  $A \vdash A^\Phi$

E,  $\vdash_{\text{CM}(A)} \forall x. \exists u. S(u, t)$

Esta sentença contradiz à sentença 2, portanto,  $\text{CM}(A)$  é inconsistente.

Corolário 4.1. A circunscrição de predicados de um conjunto de sentenças pode ser inconsistente. Observe que A não tem modelo minimal, conforme exemplo 3.5.

Demonstração. A circunscrição de predicados generaliza a circunscrição dos domínios.

---

Nota: Exemplo retirado de {DAVIS, 1980}.

#### 4.1.3.3. Não-Construtividade

Seja uma sentença  $\sigma$  que se quer provar a partir de um conjunto  $\Sigma$  de premissas em um sistema dedutivo estendido com a regra de circunscrição. Fazem-se as deduções ordinárias até que chega um ponto em que decide-se que deve-se usar a circunscrição. Primeiro, tem-se que escolher-se uma sentença e o predicado que será circunscrito nela. Mas circunscrever em relação a que sentença predicativa? (Qual o  $\Phi$  conveniente?) Não há maneira direta automática de se responder a esta questão. Isto requer uma heurística especial por parte de quem está efetuando a prova.

A circunscrição é uma sentença-esquema e sentenças-esquemas não são automaticamente manipuladas. Mesmo o esquema da indução matemática, que é um esquema-fixo quando é utilizado em prova automática de teoremas, requer intervenção humana ou heurística especial. Com mais razão a circunscrição, que requer que novos esquemas sejam produzidos de acordo com a necessidade, não terá utilização automática. A maneira mais direta de se utilizar a circunscrição é em programas inteligentes que representam a maioria dos seus conhecimentos por sentenças da lógica de primeira ordem. O programa algumas vezes circunscreveria alguns predicados. Claramente o programa teria que incluir uma heurística "dependente do domínio" para decidir que circunscrições fazer e quando retorná-las.

#### 4.1.4. Relação entre circunscrição e semântica minimal

A idéia intuitiva da circunscrição é que os elementos "conhecidos", ou os que satisfazem um certo predicado P, em uma sentença A, são todos os elementos existentes. Isto equivale em termos semânticos a restringir a classe dos modelos de A àqueles modelos que contêm apenas elementos "exigidos" por A. Esta é a mesma idéia por trás dos modelos minimais. Os teoremas a seguir mostram que a circunscrição é correta em relação à semântica minimal. Assim, de uma certa forma, mas não exatamente, o efeito de se restringir a semântica aos modelos minimais equivaleria em termos sintáticos a se adicionar uma regra de inferência como a circunscrição ao sistema dedutivo (na realidade teria que ser algo mais forte, uma vez que o sistema dedutivo estendido por circunscrição não é completo em relação à semântica minimal).

Proposição 4.4. Se uma sentença  $\sigma$  é inferida minimalmente de uma sentença A então  $\sigma$  é consequência lógica minimal de A. Em símbolos,

se  $A \vdash_m \sigma$  então  $A \models_m \sigma$ .

Demonstração. - Seja M um modelo minimal de A. Mostramos que M é modelo de  $\Omega(A)$ .

Com efeito, seja  $\Phi(x)$  uma expressão predicativa. É suficiente verificar que

$$M \models \text{Axiom}(\Phi) \wedge A^\Phi \rightarrow \forall x . \Phi(x).$$

Assim, suponha que

$$M \models \text{Axiom}(\Phi) \wedge A^\Phi$$

Seja  $|M|$  o domínio do modelo  $M$ .

$$\text{Seja } |M_0| = \{m \in |M|; M \models \Phi(m)\}$$

Seja  $N$  a estrutura induzida por  $M$  de domínio  $|M_0|$ . Então (como se pode mostrar por indução no tamanho de  $A$ ),  $N \models A$ . Desde que  $N$  é subestrutura de  $M$  e  $M$  é modelo minimal de  $A$ ,  $M = N$ . Então,  $|M| = |M_0|$ , logo  $M \models \forall x . \Phi(x)$ .

Portanto,  $M \models \text{CM}(A)$ .

Se  $A \vdash_m \sigma$

então  $\text{CM}(A) \vdash \sigma$

e  $M \models \sigma$  (pela completude da lógica clássica)

Isto completa a prova.

E a recíproca é verdadeira?

Se  $A \models_m \sigma$  então  $A \vdash_m \sigma$ ? O exemplo abaixo nos mostra que não.



Exemplo 4.18. Vimos na seção 3.1.2 que o único modelo minimal dos axiomas de Peano (sem a indução) é o modelo padrão da Aritmética. No exemplo 4.7 vimos que o resultado de se circunscrever o domínio destes axiomas é o axioma-esquema da indução. Isto nos dá a Aritmética de Peano que é incompleta, isto é, existe sentença verdadeira na aritmética padrão que não é dedutível a partir dos axiomas de Peano (com a indução). Portanto, existe uma sentença  $\sigma$  tal que

$$A \models_m \sigma \text{ mas } A \not\vdash_m \sigma$$

onde  $A$  é o conjunto dos axiomas de Peano, sem a indução.

Podemos enunciar uma recíproca mais fraca para o teorema.

Proposição 4.5. Suponha que

- 1) Todo modelo de uma sentença  $A$  tem submodelo minimal.
- 2) Toda vez que uma sentença  $B$  é verdadeira em tal submodelo minimal, também é válida no modelo original.

Então,  $A \models_m B$  implica que  $A \vdash_m B$ ; (de fato,  $A \vdash B$ ).

Demonstração. Seja  $K$  um modelo de  $A$ . Seja  $M$  um submodelo minimal de  $K$ . Por hipótese, temos que  $M \models B$ , e pela suposição (2),  $K \models B$ . Portanto,  $A \models B$  e pela completude da lógica de primeira ordem,  $A \vdash B$ .

---

Nota: Proposições 4.4 e 4.5, assim como suas demonstrações, foram extraídas de [DAVIS, 1980].

Observação 4.2. A suposição (2) é automaticamente satisfeita se B é uma sentença prenexa existencial pura.

A Circunscrição de Predicados está relacionada com a noção de minimalidade local. Se P é um símbolo predicado de uma sentença A, então ao circunscrever-se P em A temos que uma upla satisfaz ao predicado P se isto segue-se da sentença A. Se M é um modelo minimal em P, então as uplas pertencentes a P em M, são tão poucas quanto possível, isto é, uma upla pertence a P em M se isto segue-se de A.

Proposição 4.6. Se uma sentença  $\sigma$  é inferida de uma sentença A através da circunscrição de um predicado P em A, então  $\sigma$  é consequência lógica minimal em P de A.

Demonstração. Segue-se diretamente do lema 4.2.

Lema 4.2. Qualquer instância da circunscrição de um predicado P em uma sentença A é verdadeira em todos modelos minimais em P de A, isto é, é consequência lógica minimal em P de A.

Demonstração. A circunscrição de P em A é a sentença-esquema.

$$A(\Phi) \wedge \forall \bar{x}(\Phi(\bar{x}) \rightarrow P(\bar{x})) \rightarrow \forall \bar{x}(P(\bar{x}) \rightarrow \Phi(\bar{x}))$$

Seja M um modelo minimal em P de A. Tome  $\Phi = P'$ , onde

$P'$  é um predicado que satisfaz o lado esquerdo do esquema da circunscrição.  $P$  é uma extensão de  $P'$  e se o lado direito da circunscrição não fosse satisfeito,  $P$  seria uma extensão própria de  $P'$ . Mas, neste caso, seja  $M'$  um submodelo de  $M$  tal que  $M'$  concorda com  $M$  em todos predicados exceto em  $P$ , e em  $P$  concorda com  $P'$ .  $M'$  é modelo de  $A$  e  $M'$  é submodelo próprio de  $M$ . Isto contradiria a assumida minimalidade em  $P$  de  $M$ .

#### 4.2. "Default"

##### 4.2.1. Teoria "default" e suas extensões

Seja a linguagem  $L = \langle R, F, C \rangle$ . Um "default" é qualquer expressão da forma

$$\alpha(\bar{x}) : \frac{MB_1(\bar{x}), \dots, MB_m(\bar{x})}{w(\bar{x})}$$

onde  $\alpha(\bar{x})$ ,  $B_1(\bar{x})$ , ...,  $B_m(\bar{x})$  são fórmulas de  $L$  cujas variáveis livres estão entre as  $\bar{x} = x_1, \dots, x_n$ .  $\alpha(\bar{x})$  é chamado o pré-requisito do default,  $B_1(\bar{x})$ , ...,  $B_m(\bar{x})$  são as suposições do default e  $w(\bar{x})$  é o seu conseqüente. Um "default" é fechado se todas as fórmulas  $\alpha$ ,  $B_1$ , ...,  $B_m$  são sentenças.

Uma teoria "default" é um par  $(D, W)$  onde  $D$  é um conjunto de "defaults" e  $W$  é um conjunto de sentenças. Tanto  $D$  como  $W$  podem ser conjuntos infinitos. Uma teoria "default" é fechada sss todo "default" de  $D$  é fechado.

Nota: A proposição 4.6 e sua demonstração foram extraídas de [McCARTHY, 1980].



Inicialmente definiremos o conceito de extensão para teorias "defaults" fechadas depois generalizaremos esta noção para teorias "default" arbitrárias. A idéia intuitiva que se quer capturar é que o conjunto de "defaults" induz uma extensão do conjunto de sentenças de primeira ordem  $W$ , que é incompleto. A extensão não é única. Qualquer extensão é interpretada como um conjunto aceitável de crenças referente ao contexto incompletamente especificado por  $W$ . É razoável se esperar que se  $E$  é uma tal extensão então:

- 1)  $E$  contém  $W$ ,  $W \subseteq E$ .
- 2)  $E$  é dedutivamente fechado, isto é,  $th(E) = E$ ; onde  $th(E) = \{\sigma \in L, E \vdash \sigma\}$ .
- 3) Suponha que  $(\alpha : MB_1, \dots, MB_m / w)$  é um "default" de  $D$ , se  $\alpha \in E$  e  $\bigwedge B_1, \dots, \bigwedge B_m, \bigwedge w \notin E$  (isto é,  $B_1, \dots, B_m, w$  é consistente com  $E$ ), então  $w \in E$ . Assim, se  $\alpha$  é acreditado, e se cada uma das suposições  $B_1, \dots, B_m$ , assim como o conseqüente  $w$ , pode ser consistentemente acreditado, então  $w$  é acreditado.

Isto motiva a definição abaixo:

Definição 4.7.  $\Delta = (D, W)$  uma teoria "default" fechada, tal que todo "default" de  $D$  tem a forma  $(\alpha : MB_1, \dots, MB_m / w)$  onde  $\alpha, B_1, \dots, B_m, w$  são sentenças da linguagem  $L$ . Para qualquer conjunto de sentenças de  $L$ ,  $S$ , seja  $\Gamma(S)$  o menor conjunto satisfazendo as seguintes propriedades:



$$D1) \quad W \subseteq \Gamma(S);$$

$$D2) \quad \text{th}(\Gamma(S)) = \Gamma(S)$$

$$D3) \quad \text{se } (\alpha: MB_1, \dots, MB_m/w) \in D \text{ e } \alpha \in \Gamma(S), \\ \text{e } \neg B_1, \dots, \neg B_m, \neg w \notin S \text{ então } w \in \Gamma(S).$$

Um conjunto de sentenças de L, E é uma extensão para  $\Delta$  sss  $\Gamma(E) = E$ , isto é, E é um ponto fixo do operador  $\Gamma$ .

O próximo teorema nos dá uma caracterização mais intuitiva das extensões. A demonstração é encontrada em [REITER, 1980].

Teorema 4.1. Seja E um conjunto de sentenças de L e seja  $\Delta = (D, W)$  uma teoria "default" fechada. Defina

$$E_0 = W$$

e para cada  $i \geq 0$

$$E_{i+1} = \text{th}(E_i) \cup \{w_i \alpha : \frac{MB_1, \dots, MB_m}{w} \in D$$

onde  $\alpha \in E_i$  e  $\neg B_1, \dots, \neg B_m, \neg w \notin E_i\}$

Então E é uma extensão para  $\Delta$  sss

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

Observação 4.3. Note a ocorrência de E na definição de  $E_{i+1}$ .

Exemplo 4.19. Seja  $D = \left\{ \frac{:MA}{A}, \frac{:MB}{B}, \frac{:MC}{C} \right\}$  e

$$W = \{B \rightarrow (\neg A \wedge \neg C)\}.$$

$\Delta = (D, W)$  tem duas extensões:

$$E_1 = \text{th}(W \cup \{A, C\})$$

e

$$E_2 = \text{th}(W \cup \{B\}).$$

Exemplo 4.20. Seja  $D = \left\{ \frac{:MC}{\neg D}, \frac{:MD}{\neg E}, \frac{:ME}{\neg F} \right\}$ ;  $W = \emptyset$

$\Delta = (D, W)$  tem uma extensão

$$E = \text{th}(\{\neg D, \neg F\})$$

Exemplo 4.21 Seja  $D = \left\{ \frac{:MC}{\neg D}, \frac{:MD}{\neg C} \right\}$ ,  $W = \emptyset$

$\Delta = (D, W)$  tem duas extensões

$$E_1 = \text{th}(\{\neg C\})$$

e

$$E_2 = \text{th}(\{\neg D\}).$$

Exemplo 4.22 Seja

$$D = \left\{ \frac{:MA}{A}, \frac{B:MC}{C}, \frac{D \wedge A:ME}{E}, \frac{C \wedge E: M \neg A, M(D \vee A)}{F} \right\}$$

$$W = \{B, C \rightarrow (D \vee A), (A \wedge C) \rightarrow \neg E\}$$

$\Delta = (D, W)$  tem três extensões:

$$E_1 = \text{th}(W \cup \{A, C\})$$

$$E_2 = \text{th}(W \cup \{A, E\})$$

e

$$E_3 = \text{th}(W \cup \{C, E, F\}).$$

Exemplo 4.23 Seja

$$D = \left\{ \frac{A : M \exists x.P(x)}{\exists x.P(x)}, \frac{:MA}{A}, \frac{:M \neg A}{\neg A} \right\}$$

$$W = \emptyset$$

$\Delta = (D, W)$  tem duas extensões:

$$E_1 = \text{th}(\{\neg A\})$$

e

$$E_2 = \text{th}(\{A, \exists x.P(x)\}).$$

Nota: Estes exemplos foram retirados de [REITER, 1980].

Dizemos que uma extensão de uma teoria "default" fechada é inconsistente se ela é o conjunto de todas sentenças da linguagem. Assim, do teorema 4.1 podemos concluir que:

Corolário 4.2. Uma teoria "default" fechada tem uma extensão inconsistente sss  $W$  é inconsistente.

Corolário 4.3. Se uma teoria "default" fechada tem uma extensão inconsistente, então esta é sua única extensão.

Observação 4.4. Na versão original da definição 4.7 em [REITER, 1980], em D3, não está incluído o caso de  $\perp_w$  também não pertencer a S. Neste caso, poderíamos ter uma extensão inconsistente para uma teoria "default" fechada, caso tivéssemos "defaults" contraditórios.

Exemplo 4.24. Seja  $D = \left\{ \frac{:MA}{w}, \frac{:MB}{\perp_w} \right\}$ ;  $W = \emptyset$

Se  $E$  é uma extensão de  $\Delta = (D, W)$  então  $w, \perp_w \in E$  e  $E$  é uma extensão inconsistente.

Uma teoria "default" fechada é inconsistente se ela tem uma extensão inconsistente; é consistente se possui uma extensão consistente. Pelo corolário 2 podemos concluir que se uma teoria "default" fechada  $(D, W)$  é consistente, então  $W$  é consistente.



Proposição 4.7. (Minimalidade das Extensões). Se  $E$  e  $F$  são extensões para uma teoria "default" fechada  $(D, W)$  e se  $E \subseteq F$ , então  $E = F$ .

Demonstração Pelo teorema 4.1  $E = \bigcup_{i=0}^{\infty} E_i$ ,  $F = \bigcup_{i=0}^{\infty} F_i$ . Nós indutivamente provamos que  $F_i \subseteq E_i$  para todo  $i \geq 0$ , assim  $F \subseteq E$  e portanto  $F = E$ .

Trivialmente  $F_i \subseteq E_i$ . Assuma  $F_i \subseteq E_i$ , e considere  $w \in F_{i+1}$ . Se  $w \in \text{th}(F_i)$  então desde que  $F_i \subseteq E_i$ ,  $w \in \text{th}(E_i) \subseteq E_{i+1}$ . Senão, existe um "default"  $(\alpha : MB_1, \dots, MB_m/w) \in D$  onde  $\alpha \in F_i$  e  $\neg B_1, \dots, \neg B_m, \neg w \notin F$ . Desde que  $F_i \subseteq E_i$  e  $E \subseteq F$  temos que  $\alpha \in E_i$  e  $\neg B_1, \dots, \neg B_m, \neg w \notin E$ . Portanto,  $w \in E_{i+1}$ .

Definição 4.8. Seja  $\Delta = (D, W)$  uma teoria "default" fechada e  $E$  uma extensão para  $\Delta$ . O conjunto de "defaults" geradores de  $E$  com respeito a  $\Delta$  é definido como sendo:

$$\text{GD}(E, \Delta) = \left\{ \frac{\alpha : MB_1, \dots, MB_m}{w} \in D; \alpha \in E \text{ e } \neg B_1, \dots, \neg B_m, \neg w \notin E \right\}.$$

Se  $D$  é um conjunto de "defaults" (não necessariamente fechados) então

$$\text{CONSEQUENTES (D)} = \{w(\bar{x}); \frac{\alpha(\bar{x}) : MB_1(\bar{x}), \dots, MB_m(\bar{x})}{w(\bar{x})} \in D\},$$

isto é, CONSEQUENTES (D) é o conjunto dos consequentes dos "defaults" de D.

A próxima proposição justifica a terminologia de "defaults" geradores. Uma demonstração é encontrada em [REITER, 1980].

Proposição 4.8. Seja E uma extensão para uma teoria "defaults"  $\Delta = (D, W)$ .

Então,

$$E = \text{th} (W \cup \text{CONSEQUENTES (GD(E, \Delta))}).$$

#### 4.2.2. Teorias "default" arbitrárias

A tarefa agora é generalizar o conceito de uma extensão de uma teoria default fechada a teorias default arbitrárias. Intuitivamente queremos interpretar o default aberto  $\alpha(\bar{x}) : MB_1(\bar{x}), \dots, MB_m(\bar{x})/w(\bar{x})$  como afirmando: "Para todos indivíduos  $x_1, \dots, x_m$ , se  $\alpha(\bar{x})$  é acreditado e cada um dos  $B_1(\bar{x}), \dots, B_m(\bar{x})$  pode ser consistentemente acreditado então pode-se acreditar em  $w(\bar{x})$ ". O problema inicial então é especificar exatamente quem são os indivíduos de uma teo-

ria default. Os elementos nomeáveis da linguagem certamente devem ser considerados indivíduos. Mas, existem ainda os não nomeáveis que são definidos via quantificadores. Lembre-se que uma teoria default é um par  $(D,W)$ ;  $D$  um conjunto de defaults e  $W$  um conjunto de sentenças de primeira ordem. Substituímos, então, todas fórmulas de  $W$  e todos os conseqüentes dos defaults de  $D$  por suas formas Skolemizadas (ver capítulo 2). Seja  $S$ , e o conjunto das funções de Skolem introduzidas. Assim, consideramos como indivíduos da teoria default os elementos nomeáveis da linguagem expandida pelas funções de  $S$ . E as variáveis livres de um default aberto percorrem o conjunto desses indivíduos.

Uma teoria default  $(D,W)$  está na forma skolemizada sss os conseqüentes de todos os defaults de  $D$  e todas fórmulas de  $W$  estão na forma skolemizada. Por exemplo, seja

$$\Delta = \left( \frac{\exists y \forall z P(x,y,z) : M \exists y . Q(x,y)}{\exists y \forall z \forall w R(x,y,z,w)} ; \exists x . P_x \right)$$

A forma skolemizada de  $\Delta$  é a teoria abaixo:

$$\left( \frac{\exists y . \forall z P(x,y,z) : M \exists y Q(x,y)}{R(x, f(x), z, g(x,z))} ; P \alpha \right)$$

onde  $f$ ,  $g$  e  $\alpha$  são funções de skolem ( $\alpha$  é uma função de Skolem 0-ária, uma constante). Observe que a "parte superior" do default fica inalterada.

Seja  $T$  o conjunto dos termos livres de variáveis da linguagem de uma teoria default na forma skolemizada,  $\Delta=(D,W)$ , expandida pelas funções de Skolem. Defina

$$\text{DEFAULTS-FECHADOS } (\Delta) = \{\delta(t_1, \dots, t_m); \delta(\bar{x}) \in D \text{ e } t_1, \dots, t_m \in T\}$$

As extensões para uma teoria default arbitrária podem agora ser definidas:

Definição 4.9. E é uma extensão para uma teoria default  $\Delta = (D, W)$  sss E é uma extensão para a teoria default fechada,  $\text{FECHAMENTO } (\Delta) = (\text{DEFAULTS-FECHADOS } (\Delta), W)$ .

Vários pontos devem ser observados nesta definição:

- 1) O conjunto  $\text{DEFAULTS-FECHADOS } (\Delta)$  é em geral infinito enumerável. Assim, a teoria default fechada  $\text{FECHAMENTO } (\Delta)$  tem um número infinito enumerável de defaults. Em nossa definição de extensão para uma teoria default fechada não há nenhuma suposição quanto ao número de defaults de uma teoria. Portanto, não há nenhum problema com a extensão da teoria  $\text{FECHAMENTO } (\Delta)$ .
- 2) Inicialmente temos uma teoria default em uma linguagem  $L = \langle R, F, C \rangle$ . Ao skolemizarmos a teoria, introduzimos novos símbolos funcionais, as funções de Skolem S, e  $\text{FECHAMENTO } (\Delta)$  é uma teoria default fechada na linguagem expandida  $L' = \langle R, FUS, C \rangle$ . Assim as extensões para  $\Delta$  consistem de sentenças da linguagem expandida  $L'$  e não da linguagem original L.



### 4.2.3. Propriedades

#### 4.2.3.1. Não-Monotonicidade

O raciocínio por "default" corresponde ao processo de derivar-se conclusões baseado sobre padrões de inferência da forma "Na falta de alguma informação em contrário assuma...". Portanto, o raciocínio por "default" não é monotônico. Caso se adicione nova informação que contrarie uma conclusão tirada anteriormente, esta conclusão não mais vale. Por exemplo, considere a teoria consistindo de apenas um "default" :  $MA/B$ . Assim, B pode ser assumido. Adicionando-se a sentença  $\neg A$  a esta teoria, não mais se pode assumir B. Assim, qualquer Lógica que formalize o raciocínio por "default" é não-monotônica. Em geral as teorias "defaults" são não-monotônicas no sentido que se  $\Delta = (D, W)$  é uma teoria "default" com uma extensão E,  $D'$  é um conjunto de "defaults" e  $W'$  é um conjunto de sentenças, então a teoria "default"  $\Delta' = (D \cup D', W \cup W')$  pode não ter uma extensão  $E'$ , tal que  $E \subseteq E'$ .

Exemplo 4.25. Seja

$$D = \left\{ \frac{:MC}{\neg D}, \frac{:MD}{\neg E}, \frac{:ME}{\neg F} \right\}; \quad W = \emptyset$$

$\Delta = (D, W)$  tem apenas uma extensão

$$E = \text{th} (\ulcorner D, \urcorner F)$$

Seja  $W' = \{\ulcorner C\}$

Então,  $\Delta' = (D, W')$  tem também apenas uma extensão

$$E' = \text{th} (\ulcorner C, \urcorner F)$$

$$E \not\subseteq E'.$$

#### 4.2.3.2. Inexistência de extensões

Proposição 4.9. Existem teorias "defaults" que não possuem extensões.

Exemplo 4.26. Seja  $D = \left\{ \frac{\ulcorner MA}{\urcorner A} \right\}$ ;  $W = \emptyset$

Esta teoria não tem extensão.

## 4.2.3.3. Não semi-decidibilidade das extensões

Em lógica de primeira ordem dada uma sentença arbitrária não é decidível se ela é um teorema de um dado conjunto de axiomas ou não. Mas dada uma seqüência de sentenças é (trivialmente) decidível se esta seqüência é uma prova. Assim, podemos recursivamente enumerar os teoremas de um dado conjunto de axiomas (semi-decidibilidade). Podemos ver os elementos das extensões de uma teoria "default" como os "teoremas" desta teoria. Naturalmente levanta-se a questão:

"Dada uma teoria "default" e uma sentença arbitrária da linguagem B, B é um teorema desta teoria? (ou por outra, existe uma extensão para esta teoria que contenha B?)"

Não podemos esperar decidir esta questão, uma vez que em lógica de primeira ordem não é decidível, mas é pelo menos como na lógica de primeira ordem, semi-decidível? As teorias "defaults" são não-monotônicas e portanto não temos uma teoria de prova local, a localidade da teoria de prova exerce um papel importante para que esta questão seja semi-decidível em lógica de primeira ordem (ver capítulo 2 e capítulo 3 seção 3.1.3). Temos um bom motivo para acreditarmos que esta questão não é nem ao menos semi-decidível. É isto que nos confirma a proposição abaixo:

Proposição 4.10. A união das extensões de uma teoria "default" arbitrária não é recursivamente enumerável.

Demonstração. Em uma linguagem  $L$ , dada uma teoria "default"  $\Delta = (D, W)$  seja  $B(\Delta) = \cup E$  onde a união é tomada sobre todas extensões  $E$  de  $\Delta$ . Seja  $w_1, w_2, \dots$  uma enumeração recursiva das sentenças de  $L$ . Para  $i = 1, 2, \dots$  seja  $\Delta_i$  a teoria "default"  $(\{ :Mw_i/w_i \}, \emptyset)$  ( $\Delta_i$  tem apenas um único "default" e nenhuma sentença de primeira ordem).  $\Delta_i$  tem claramente uma única extensão  $E_i$ .  $E_i$  e portanto  $B(\Delta_i)$  é vazia se  $w_i$  é insatisfatível, e é  $th(w_i)$  se  $w_i$  é satisfatível. Portanto  $\bigcup_{i=1}^{\infty} E_i = \bigcup_{i=1}^{\infty} B(\Delta_i)$  é o conjunto das sentenças satisfatíveis de  $L$ .

Argumentamos que não existe uma enumeração recursiva de  $B(\Delta)$  para teoria "defaults" arbitrárias. Pois, se existisse, então existiria uma função recursiva  $f(\Delta, i)$  definida sobre as teorias "defaults" e os inteiros não negativos  $i$ , tal que a imagem de  $f(\Delta, i)$  seria  $B(\Delta)$ . Em particular, imagem de  $f(\Delta_j, i)$  seria  $B(\Delta_j)$ . Sabe-se então que, (ver [ROGERS, 1967] que existiria uma função recursiva  $g(i)$  definida sobre os inteiros não negativos tal que a imagem de  $g$  seria  $\bigcup_{j=1}^{\infty} B(\Delta_j)$ , isto é,  $\bigcup_{j=1}^{\infty} B(\Delta_j)$ , o conjunto das fórmulas satisfatíveis de  $L$ , seria recursivamente enumerável, uma contradição (ver capítulo 2).

[REITER, 1980] comenta este resultado "... qualquer teoria de prova para teorias "default" devem de algum modo apelar a algum processo inerentemente não semi-decidível.



Este resultado extremamente pessimista força a conclusão que qualquer tratamento computacional de "defaults" deve necessariamente ter um componente heurístico e, algumas vezes, tirará conclusões errôneas."

No presente trabalho apresentamos algumas semânticas e inferências não-monotônicas. Nosso objetivo principal foi analisar as propriedades destes sistemas lógicos não ordinários.

Um fato relevante é a não-construtividade essencial dos sistemas não-monotônicos. As semânticas restritivas são incompletas. Não há meios de se construir um sistema efetivo e finito que derive as conseqüências lógicas restritas de um conjunto de sentenças arbitrário. As inferências não-monotônicas provêm alguma coisa equivalente a uma regra de inferência da forma: "se P é consistente com o que é conhecido, então inferimos Q". Uma vez que uma regra de inferência do tipo acima é adicionada não é possível demonstrar em geral que alguma coisa é uma prova, desde que um passo da prova envolve decidir-se a satisfatibilidade de uma fórmula arbitrária. Assim qualquer implementação computacional dos sistemas lógicos não-monotônicos requererá um componente heurístico e, algumas vezes, cometer-se-á enganos.

Uma comparação entre as diversas abordagens não-monotônicas seria útil para a compreensão e conhecimento destes sistemas.

A lógica não-monotônica de [McDERMOTT & DOYLE, 1980] parece ser uma generalização da lógica de "default". Mas em geral a relação entre as duas é bastante complexa. [REITER,

1980b] apresenta um pequeno estudo comparativo entre as duas. McCARTHY em um Addendum à edição do Jornal de Inteligência Artificial, V. 13, nº 1 e 2, 1980 (citado anteriormente) faz uma ligeira comparação entre a circunscrição, a lógica de McDERMOTT e DOYLE e a lógica de "default". Conclui que as duas generalizam a circunscrição, mas nos casos de aplicação prática em que se usam as duas lógicas, a circunscrição também poderia ser usada. Uma comparação efetiva entre as diversas abordagens ainda está por se fazer.

Os formalismos não monotônicos desempenham um papel importante em Inteligência Artificial na construção de sistemas com modos de inferência estendidos. O futuro progresso neste campo dependerá do desenvolvimento de conhecimentos práticos e teóricos. O conhecimento prático capacitará os pesquisadores, com a experiência adquirida, à construção de sistemas de raciocínio mais e mais complexos. Nisto podemos contar com o desenvolvimento da tecnologia de computadores digitais, que permitirá a implementação de sistemas com complexidade crescente.

Muitos pontos no formalismo e implementação de sistemas não-monotônicos continuam em aberto. O componente heurístico necessário à implementação desses formalismos e o processo de revisão de conclusões, precisam ser melhor entendidos e desenvolvidos. Gostaríamos de ver as inferências não-monotônicas atuando de modo a restringir a classe dos modelos de uma teoria de primeira ordem, a um conjunto de modelos que satisfazem todas fórmulas derivadas no sistema de inferência estendido, e somente essas. E,



inversamente, o conjunto das fórmulas inferidas são as fórmulas satisfeitas pelos modelos do conjunto. Existe este conjunto? Como caracterizá-lo? Para a circunscrição este conjunto seria a classe dos modelos minimais, mas não exatamente. Conforme vimos, existem teorias cujas fórmulas satisfeitas pelos modelos minimais da teoria, não são todas derivadas pelo sistema de inferência estendido com a circunscrição. O problema para os outros formalismos permanece totalmente em aberto.

[McDERMOTT & DOYLE, 1980] levanta algumas questões de natureza matemática. "Existem interpretações das lógicas não-monotônicas dentro da lógica clássica? Existem conexões entre as lógicas não-monotônicas e as lógicas com declarações infinitas? Existe uma interpretação topológica das lógicas não-monotônicas em analogia com a interpretação topológica do cálculo intuicionista?" E ainda algumas questões mais especulativas: "A revisão de conclusões executadas pelos programas de Inteligência Artificial pode ser vista como uma versão microscópica do processo de mudanças das teorias científicas. Podem as idéias capturadas nas lógicas não-monotônicas serem usadas para descrever o processo geral das descobertas científicas, ou do comportamento pragmático em geral? Como são as semânticas holísticas das lógicas não-monotônicas relacionadas com mudanças de significado? Quanto custa a suspensão do juízo? Pode a lógica não-monotônica ser usada efetivamente para descrever e raciocinar sobre ações, comandos, e causalidade?"



6 - REFERÊNCIAS BIBLIOGRÁFICAS

- BOBROW, D.G. & WINOGRAD, T. An overview of KRL, a Knowledge Representation Language, Cognitive Science 1 (1), p. 3-46, 1977.
- CARVALHO, R.L., MAIBAUM, T.S.E., PEQUENO, T.H.C., PEREDA, A. A. e VELOSO, P.A.S. A model theoretic approach to the teory of abstract data types an data structures. Research Report CS-8022, University of Waterloo, Waterloo, Ontario, Canada, Abril, 1980.
- CHANG, C.C. & KEISLER, H.J. Model Teory, North-Holland, 1973.
- CHANG, C.L. & LEE, R.C.T. Simbolic Logic and Mechanical Teorem Proving. New York, Academic Press, 1973.
- CHURCH, A. An Unsolvable Problem of Elementary Number Theory. Amer. J. Math. 58, p. 345-363, 1936.
- DAVIS, M. Notes on the mathematics of non-monotonic reasoning. Artificial Intelligence, V. 13, Nº 1 e 2, 1980.
- DOYLE, J. A Truth Maintenance System. Artificial Intelligence, V. 12, p. 231-272, 1979.
- ENDEM, M.H.V. & KOWALSKI, R.M. The semantics of predicate logic as a programming language. Journal of ACM, Vol. 23, nº 4, 1976.
- ENDERTON, H.B. A mathematical introduction to logic. Academic Press, 1972.
- GÖDEL, K. Die Vollständigkeit der Axiome der Axiome des logischen Funktionenkalküls, Monatsh. Math. Phys. V. 37, p. 349-360, 1930.
- GÖDEL, K. Über formal unentscheidbare Sätze der Principia Mathematica und vermandter Systeme I, Monastsh. Moth. Phys. V. 38, p. 173-198, 1931.
- HAYES, P.J. Robotologic. Machine Intelligence, V. 5, p. 533-554, 1970.
- HAYES, P.J. A logic of actions. Machine Intelligence, V. 6, p. 495-520, 1971.

- HAYES, P.J. The frame problem and related problems in artificial intelligence. Artificial and Human Thinking, 1973.
- HERMES, H. Enumerability, Decidability, Computability. Springer-Verlag, 1965.
- McCARTHY, J. Programs with common sense. Proceedings of the Teddington Conference on the Mechanization of Thought Process, 1960.
- McCARTHY, J. Circumscription - a form of non-monotonic reasoning. Artificial Intelligence, V. 13, nºs 1 e 2, 1980.
- McCARTHY, J. & HAYES, P.J. Some philosophical problems from the standpoint of Artificial Intelligence. Machine Intelligence, V. 4, 1969.
- McDERMOTT, D. Nonmonotonic Logic II: Nonmonotonic Modal Theories. Journal of ACM, V. 29, nº 1, 1982.
- McDERMOTT, D. & DOYLE, J. Non-Monotonic Logic I. Artificial Intelligence, V. 13, nºs 1 e 2, 1980.
- MEDEIROS, J.H.N. Lógicas Não-Monotônicas - Aplicações. Tese de Mestrado, Departamento de Informática, PUC-RJ, 1982.
- MINSKY, M. A framework for representing knowledge. MIT Artificial Intelligence Laboratory, AI Memo 306, 1974.
- NEWELL, A. & SIMON, H. GPS a program that simulates human thought. E.A. Feigenbaum and J. Feldman (Eds.). Computers and Thought, McGraw-Hill, 1963.
- PEQUENO, T.H.C. Uma descrição formal dos processos de especificação e implementação de tipos abstratos de dados. Tese de Doutorado, Departamento de Informática, PUC-RJ, 1981.
- RAPHAEL, B. The frame problem in problem-solving systems. N.V. Findler and B. Meltzer (Eds.), Artificial Intelligence and Heuristic Programming, Edinburg University Press, 1971.
- REITER, R. A Logic for Default Reasoning. Artificial Intelligence, V. 13, nºs 1 e 2, 1980.
- REITER, R. Some results on defaults logics. Technical Report, Department of Computer Science, Univ. of British Columbia, 1980b.

- ROGERS, H. Theory of Recursive Functions and Effective Computability. McGraw-Hill, 1967.
- VELOSO, P.A.S. Nomable Models and Programing. Departamento de Informática, PUC-RJ, 1979.
- VELOSO, P.A.S. & CARVALHO, R.L. Towards a logic of limited perception. Proceedings of the third Brazilian Conference on Mathematical Logic, Recife, 1979.
- WEYRAUCH, R., W. Prolegomena to a Theory of Mechanized Formal Reasoning. Artificial Intelligence. V. 13, nºs 1 e 2, 1980.
- WINOGRAD, T. Extended Inference Modes in Reasoning by Computer Systems. Artificial Intelligence, V. 13, nºs 1 e 2, 1980.