Universidade Federal de Uberlândia

Faculdade de Engenharia Elétrica

Verificação Automática de Lógicas Finitas Multivalentes

Tese apresentada por Marcelo Rodrigues de Sousa à Universidade Federal de Uberlândia (UFU) como parte dos requisitos para obtenção do título de Doutor.

Banca Examinadora:

- Antônio Eduardo Costa Pereira, PhD. Orientador (UFU)
- Alexsandro Santos Soares, Dr. (UFG)
- Mamede Lima-Marques, Dr. (UnB)
- Jamil Salem Barbar, Dr. (UFU)
- João Nunes Souza, Dr. (UFU)

Dados Internacionais de Catalogação na Publicação (CIP)

S725v Sousa, Marcelo Rodrigues de, 1964-
Verificação automática de lógicas finitas multivalentes [manuscrito] / Marcelo Rodrigues de Sousa. - 2010.
79 f. : il.

Orientador: Antônio Eduardo Costa Pereira.

Tese (doutorado) – Universidade Federal de Uberlândia, Programa de Pós-Graduação em Engenharia Elétrica.
Inclui bibliografia.

1. Lógica simbólica e matemática - Teses. I. Pereira, Antônio Eduardo Costa. II. Universidade Federal de Uberlândia. Programa de Pós-Graduação em Engenharia Elétrica. IV. Título.

CDU: 510.6

Elaborado pelo Sistema de Bibliotecas da UFU / Setor de Catalogação e Classificação

Verificação Automática de Lógicas Finitas Multivalentes

Marcelo Rodrigues de Sousa

Tese apresentada por Marcelo Rodrigues de Sousa à Universidade Federal de Uberlândia (UFU) como parte dos requisitos para obtenção do título de Doutor.

Antônio Eduardo Costa Pereira, PhD.	Prof. Alexandre Cardozo, Dr.
Orientador	Coordenador do
	Curso de Pós-Graduação

pudet enim me uxori meae optimae, suavissimis liberis virtutem et diligentiam non praestitisse

Marco Túlio Ciceron, "Epístolas, Ó Cartas", Livro XIV, pág. 232

Agradecimentos

Agradeço ao amor da minha vida, Kelbya, aos meus filhos, Joel e Giovanna, a nossa felicidade.

Sumário

Resumo

Nos últimos anos tem crescido o interesse nas denominadas lógicas multivalentes: no ramo da computação, em áreas como prova automática de teoremas, raciocínio aproximado, sistemas multi-agente e verificação de programas; na engenharia elétrica como em circuitos digitais; na área da matemática pura, como em provas de independência ou consistência, na teoria generalizada de conjuntos e estruturas algébricas universais e mesmo na linguística e filosofia.

Nesse contexto, o problema de axiomatização geral de lógicas finitas multivalentes ainda não foi resolvido de forma satisfatória. Para tal, devemos demonstrar os teoremas da correção e completude desses sistemas lógicos, ambos teoremas matematicamente rigorosos. De forma geral, a demostração do "teorema da correção" não pode ser considerada como uma dificuldade pois é direta, bastando uma verificação nos axiomas e regras de inferência. No entanto, é sabido que em geral a demonstração do "teorema da completude" é muito mais sofisticada e particular para cada sistema lógico.

A completude é uma das noções mais importantes na Lógica e nos fundamentos da Matemática. Completude significa a demonstração da possibilidade de obtermos todos os esquemas corretos de inferência através do uso de um sistema formal lógico. Além disso, quando desejamos "construir" ou "desenhar" um sistema lógico multivalente implicitamente estamos procu-

rando por um sistema com um conjunto mínimo de axiomas e regras.

A proposta dessa tese é estabelecer um método algorítmico para demonstração da completude em lógicas finitas multivalentes. Demonstra-se que se uma matriz M é correta para um sistema lógico finito multivalente $\mathcal{L}$ e todas as suas extensões unitárias corretas são repetições da matriz M, então M é uma matriz característica de $\mathcal{L}$. A partir desse resultado, são implementados procedimentos computacionais que demonstram a completude de um sistema lógico finito multivalente. A nova abordagem soluciona de uma forma mais simples e uniforme o problema da axiomatização de lógicas finitas multivalentes.

Abstract

In recent years, there is a growing interest in many-valued logics in many areas such as computer sciences – automated theorem proving, approximate reasoning, multi-agent systems, program verification – electrical engineering and digital circuits, linguistics, mathematics and algebra, philosophy, etc.

In this context, the general problem of finding an axiomatization for finite many-valued logics has not yet been solved satisfactorily. For this, we must demonstrate the correction and completeness theorems of these logic systems, both mathematically rigorous. Soundness theorem is straightforward, however, it is known that the demonstration of the "completeness theorem"is much more sofisticated and unique for each logic system.

Completeness is one of the most important notions in logic and the foundations of mathematics. Completeness means the possibility of getting all correct and reliable schemata of inference by use of logical methods. When one wishes to build or design a many-valued logic system, he or she is implicitly looking for a system with a minimal set of axioms and rules.

The purpose of this thesis is to establish a general algorithmic completeness proof procedure for finite many-valued logics. It is shown that a matrix is characteristic (sound and complete) for a many-valued finite logic system when, all successive correct extensions of this matrix have the same set of tautologies. It is also shown that in order to determine if a matrix is cha-

racteristic, all one has to do is to fetch from all the unit extensions those which are correct and then verify if they are repetitions of the matrix being considered. How to implement a computational procedure to demonstrate the completeness of a many-valued finite logic system is shown from these results. The new approach is a simpler and more uniform solution for the problem of finite many-valued logics axiomatization.

Introdução

Uma teoria científica geralmente apresenta um corpo de conceitos e uma coleção de asserções. Quando questionados sobre o significado de um conceito em uma teoria, frequentemente o explicamos ou o definimos em termos de outros conceitos presentes na teoria. Mais ainda, quando questionados sobre uma verdade ou a razão da crença sobre a verdade de uma asserção, usualmente a justificamos demonstrando que podemos deduzi-la a partir de um conjunto de outras asserções que aceitamos como verdadeiras.

O processo de sucessivamente questionarmos a verdade de definições e deduções pode nos levar a dois caminhos. No primeiro deles, recorremos às argumentações circulares onde há uma sequência fechada de definições; no segundo, finalizamos as explicações em asserções e conceitos tão básicos que podemos garantir a sua veracidade.

Quando o problema é simplesmente entender o significado de um conceito ou a verificação da veracidade de uma proposição, não existe nenhuma objeção básica a esses procedimentos circulares. Todavia, quando é possível iniciar uma teoria científica através de um pequeno número de ideias primitivas e proposições, há um apelo natural às indagações sobre o significado e a verdade das asserções em função dessas poucas ideias primitivas acrescidas a formas típicas de deduções e definições.

Geralmente, as ideias e proposições primitivas são denominadas *axio-*

mas ou *postulados*. E quando os conceitos e proposições de uma teoria são arranjados em concordância às conexões de deducibilidade e definibilidade constituem-se num *sistema axiomático* para a teoria. O sistema axiomático mais famoso é, sem dúvidas, a axiomática de Euclides para a geometria. O seu *Os Elementos* [6] é um tratado matemático e geométrico consistindo em 13 livros escritos em Alexandria por volta de 300AC. Esse tratado engloba uma coleção de definições, axiomas (postulados), teoremas (proposições e construções) e provas matemáticas dos teoremas, cobrindo a geometria, hoje dita Euclidiana, e a da antiga teoria grega dos números elementares. *Os Elementos* é considerado uma obra-prima da aplicação da lógica à matemática.

Existem em *Os Elementos* 10 proposições primitivas, das quais 5 são denominadas noções comuns e 5 são os axiomas ou postulados. A partir desses e algumas definições, 465 teoremas são deduzidos com um rigor lógico exemplar. O mesmo método dedutivo foi posteriormente utilizado por Sir Isaac Newton na física (mecânica), por Lagrange na mecânica analítica, etc. Por todos esses anos, um grande número de sistemas axiomáticos foram criados na matemática e em vários ramos das ciências naturais. Hoje em dia o formalismo dos sistemas axiomáticos foi desenvolvido e aperfeiçoado, todavia é inegável o sucesso e a aceitação universal do *Os Elementos*, assim como a sua contribuição à ciência matemática.

O atual desenvolvimento de sistemas axiomáticos está diretamente relacionado com a descoberta das novas geometrias ditas não-euclidianas. No final do século XIX, Lobachevshi, Bolyai e Gauss independentemente descobriram geometrias consistentes onde o postulado das paralelas foi substituído pela assunção de que "em um plano bi-dimensional, para uma dada linha I e um ponto A, que não está em I, existe mais de uma reta que passa por A e não apresentam intersecção com I". Em 1854, Riemann criou uma outra geo-

metria consistente onde o quinto postulado de Euclides foi falsificado, desta vez, com a assunção da "não existência de retas que passam por A e não apresentam intersecção com l". O recorrente questionamento do chamado quinto postulado de Euclides, o "postulado das paralelas", demonstrou que ele é independente dos demais axiomas da geometria euclidiana, podendo portanto ser substituído consistentemente por outra assunção.

As "geometrias não-euclidianas", de extrema importância na física, influenciaram substancialmente a lógica matemática, possibilitando que no ramo da lógica, ciência tida como morta por Kant, surgissem novos caminhos, novas visões — as lógicas não-clássicas. De acordo com o lógico brasileiro Newton da Costa [4], "o surgimento das geometrias não-euclidianas talvez tenha sido um dos maiores acontecimentos na história da cultura e, até hoje, tais geometrias são motivações heurísticas ou analógicas para a construção das lógicas não-clássicas".

A Lógica e o Teorema da Completude

A lógica pode ser considerada a ciência do raciocínio, aquela que estuda a relação de conseqüência tratando das inferências cujas conclusões têm que ser tão verdadeiras quanto as suas premissas. Nesse cenário podemos considerar várias abordagens da lógica, como exemplos, a lógica dedutiva cujo objetivo consiste no estudo dos princípios lógicos usados no raciocínio dedutivo, matemático; ou a lógica indutiva, aquela que trata das formas de inferência plausíveis. Assim, quando dizemos que tem chovido nos últimos 50 carnavais, não podemos deduzir logicamente que no próximo carnaval irá chover, todavia essa asserção é plausível.

A lógica dedutiva atualmente desenvolve-se no sentido da aproximação

da ciência matemática, fundamentalmente através da disciplina lógica matemática que estuda o tipo de raciocínio desenvolvido por matemáticos, claramente um raciocínio dedutivo. Nesse sentido o método axiomático recebe especial importância, pois na sua vestimenta contemporânea, a lógica é vista como sistema formal dedutivo, edificado sobre uma linguagem formal.

Uma lógica formal é um sistema que provê regras para inferências válidas. Se essas regras e as definições são suficientemente precisas, elas tornam-se regras para manipulação de cadeias de caracteres e símbolos (*strings*) que podem ser realizadas em uma folha de papel. O próximo estágio na construção de uma lógica formal é atribuir significado a essas *strings* e apresentar uma justificativa matemática para a escolha e determinação das regras. Dois teoremas são normalmente apresentados nesse contexto, o primeiro, o "teorema da correção" estabelece que nenhuma dedução incorreta pode ser feita a partir do sistema formal; o segundo, o "teorema da completude" determina que *todas* as deduções corretas podem ser expressas no sistema lógico através de uma combinação das suas regras e definições. Ambos são teoremas matemáticos rigorosos, todavia, em geral, a demonstração do "teorema da correção" normalmente é direta, já a demonstração do famoso "teorema da completude" é muito mais sofisticada.

A completude é uma das noções mais importantes na lógica e nos fundamentos da Matemática. Completude significa a demonstração da possibilidade de obtermos todos os possíveis corretos esquemas de inferência através do uso de um sistema formal lógico. Existem técnicas bem definidas na literatura para a demonstração da completude em sistemas lógicos formais. Podemos citar o método construtivo estabelecido por Kalmár utilizado por Mendelson [18], todavia é fundamental perceber que, mesmo utilizando qualquer método, para cada sistema lógico distinto temos uma demonstração

própria do "teorema da completude".

Atualmente, a lógica apresenta várias ramificações e pode ser classificadas em finitas ou infinitas, clássicas ou não-clássicas etc. O objeto desse trabalho está restrito ao universo das lógicas finitas proposicionais multivalentes, sendo considerados, por exemplo, os sistemas lógicos bivalentes clássicos, as lógicas multivalentes de Łukasiewicz, Post, e Bochvar, assim como muitos outros existentes na literatura, vide [8].

Existem na lógica clássica assim como nos diversos ramos das lógicas não-clássicas abordagens distintas na direção da formulação de um cálculo lógico (*Logic Calculi*). Historicamente o primeiro sistema e ainda o mais estudado e reconhecido é o cálculo lógico do tipo hilbertiano.

Quando surgiram diversas lógicas multivalentes no início do século XX, a questão natural que emergiu foi a determinação de um método geral para a axiomatização de lógicas finitas multivalentes. Somente em 1952, J. B. Rosser e A. R. Turquette apresentaram uma axiomatização para uma extensa classe de lógicas finitas multivalentes [20]. Eles assumiram que o conjunto de conectivos dessas lógicas contem um conectivo binário denotado por $\rightarrow$, um tipo de implicação, e conectivos unários J_s para cada s pertencente ao conjunto dos valores-verdade do sistema lógico. Todavia, a axiomatização obtida é muito complexa em decorrência da generalidade do método desenvolvido por Rosser e Turquette. Em 1994, Anshakov e Rychkov [1] apresentaram um método de construção de lógicas finitas multivalentes através da extensão do cálculo booleano clássico e a demonstração do teorema da completude para esses lógicas.

Hoje, o problema de axiomatização geral de lógicas finitas multivalentes ainda não foi resolvido satisfatoriamente, como afirmado por Siegfried Gottwald em seu livro "A Treatise on Many-Valued Logics" [8]: *"it would be*

quite welcome to have some general and uniform way to axiomatize logical systems of some type".

A proposta dessa tese é o estabelecimento de um novo procedimento para demonstração da completude em lógicas finitas multivalentes: um método algorítmico. As vantagens de tal abordagem serão percebidas no decorrer do trabalho, visto que, para fazer a demonstração da completude de um sistema lógico finito multivalente são necessários apenas alguns segundos para a execução de um programa de computador que automaticamente a constroi. Outro aspecto relevante na nova abordagem proposta é a possibilidade de construir sistemas axiomáticos apropriados no sentido da sua simplicidade.

Quando desejamos "construir" ou "desenhar" um sistema lógico multivalente implicitamente estamos procurando por um sistema com beleza matemática intrínseca. Isso geralmente é obtido através da designação de um conjunto mínimo de axiomas e regras. Esse objetivo tem razão em si mesmo, pois com menos regras e axiomas normalmente a demonstração de propriedades desses sistemas lógicos apresentam menor complexidade.

É importante salientar que, pela abordagem proposta, será permitido analisar o problema da axiomatização no sentido inverso ao usual. Ou seja, dada uma lógica definida por axiomas e regras podemos através do novo procedimento de investigação da completude determinar qual a semântica matricial característica dessa lógica.

Essa tese apresenta dois teoremas originais, através dos quais é estabelecido um novo método absolutamente construtivo para a demonstração da completude em lógicas finitas multivalentes. Para a demonstração desses teoremas e posterior exemplificação do uso do método em questão são necessários os conceitos gerais e a compreensão da lógica formal.

Delineamento desse Trabalho

O texto é subdividido em capítulos como a seguir:

No capítulo 1, é apresentada a lógica de forma geral, algumas de suas ramificações, principalmente as lógicas multivalentes e suas aplicações. Também é mostrado o método de Rosser e Turquette.

No capítulo 2, inicialmente são apresentados os conceitos sintáticos e semânticos fundamentais de uma lógica multivalente finita. Posteriormente, são definidas as matrizes corretas, as extensões corretas de matrizes e então demonstra-se o teorema que estabelece:

> *Seja $\mathcal{L}$ uma lógica com assinatura Σ, com matriz característica finita. M_1 é uma matriz característica para $\mathcal{L}$ se, e somente se, $\mathcal{L}$ é correta para M_1, e para toda extensão correta M_2 de M_1, $Taut(M_1) = Taut(M_2)$,*

informalmente, completude pode ser determinada como um resultado de um procedimento de verificação de tautologias para uma matriz e todas as suas extensões corretas.

Posteriormente, demonstra-se um segundo teorema que estabelece:

> *Seja $\mathcal{L} = (\Sigma, A, R)$ uma Lógica Multivalente Finita com assinatura Σ. Se M é uma matriz correta para $\mathcal{L}$ e todas as extensões unitárias corretas de M são repetições de M, então M é uma matriz característica de $\mathcal{L}$.*

Assim, a completude pode ser obtida através do resultado positivo em um procedimento de verificação da repetição entre uma matriz candidata e suas extensões unitárias corretas. A determinação da repetição entre matrizes é uma simples verificação da existência de um homomorfismo "especial" entre

a matriz candidata à matriz característica e toda suas possíveis extensões unitárias corretas. Esse último teorema induz um algoritmo para a demonstração da completude para sistemas lógicos finitos multivalentes.

No capítulo 3, apresenta-se uma implementação do método algorítmico e uma forma prática de seu uso como consequência dos teoremas fundamentais do capítulo 3. Também serão apresentadas as demonstrações de completude de 1 sistema lógicos proposicional finito, através da implementação do algoritmo na linguagem Prolog. No Apêndice estão todo o código-fonte em Prolog que realiza da demonstração apresentada no capítulo 3.

Finalmente, são apresentadas as conclusões e as possíveis direções futuras desse trabalho.

Capítulo 1

Lógicas Finitas Multivalentes

Nos últimos anos tem crescido o interesse nas denominadas lógicas multivalentes: no ramo da computação, em áreas como prova automática de teoremas, raciocínio aproximado, sistemas multi-agente e verificação de programas; na engenharia elétrica como em circuitos digitais; na área da matemática pura, como em provas de independência ou consistência, na teoria generalizada de conjuntos e estruturas algébricas universais e mesmo na linguística e filosofia.

Como exemplo moderno do uso das lógicas multivalentes temos no livro *Test Pattern Generation using Boolean Proof Engines* escrito por Rolf Drechsler, Stephan Eggersglüß, Görschwin Fey, Daniel Tille e publicado pela Springer-verlag em 2009, o capítulo 6, intitulado *Multiple-Valued Logic*, onde explica-se como trabalhar com restrições no mundo real usando a ferramenta ATPG (*Automatic Test Pattern Generation*). Para utilizar essa ferramenta é insuficiente considerar apenas lógicas booleanas, com os valores-verdade 0 e 1, uma vez que circuitos industriais contém elementos que não apresentam um comportamento "booleano". Assim, no livro, define-se uma lógica de quatro valores-verdade para resolver essas limitações. Há uma tendência de

aumento do uso dessas tecnologias com lógicas multivalentes em aparelhos eletrônicos e computadores.

Para uma melhor compreensão das lógicas multivalentes necessitamos inicialmente dos conceitos fundamentais da lógica que a precede, a lógica clássica.

1.1 A Lógica Clássica

Até o início do século XX, considerava-se a existência de uma única lógica fundada por Aristóteles (384-322 AC). Essa lógica é usualmente denominada lógica clássica. Aristóteles criou a *Teoria do Silogismo* e a axiomatizou. Também, introduziu os conceitos que nos permite lidar com as noções de necessidade, possibilidade e contingência.

Após Aristóteles, a lógica clássica pouco desenvolveu-se. Um marco considerável no desenvolvimento da lógica clássica moderna iniciou-se no século XVII, com Leibniz, momento onde a lógica aproximou-se substancialmente da matemática. Outros importantes precursores da lógica contemporânea foram Boole, De Morgan, Peierce, Schröder e McColl.

Friedrich Ludwig Gottlöb Frege foi certamente sistematizador mais importante da Lógica Clássica, sendo considerado como o fundador da Lógica Clássica Moderna. Pela primeira vez é apresentado no *Begriffsschrift* [7] o cálculo proposicional em sua forma atual, assim como a noção de função proposicional, o uso de quantificadores e a análise lógica de prova por indução matemática. Na história da lógica, O *Analytica Priora* de Aristóteles somente é comparável ao *Begriffsschrift* de Frege.

A lógica clássica, na sua parte mais simples ou elementar, versa essencialmente sobre os chamados conectivos lógicos de negação, conjunção, disjunção,

implicação e bicondicional, sobre os quantificadores existencial e universal e sobre o predicado de igualdade; e sobre algumas de suas extensões como sistemas de teorias de conjuntos.

Usando a base construída na obra de Frege, a lógica clássica adquiriu a sua forma quase definitiva, extensa e consistente nos *Principia Mathematica* de Whitehead e Russell [25]. No seu estado atual, a lógica clássica encerra toda silogística aristotélica.

São considerados, de imediato, como intuitivamente claros em termos pré-formais, o uso de variáveis e símbolos de proposições e predicados, e o significado dos operadores lógicos para: quantificador existencial ($\exists$, "existe"), quantificador universal ($\forall$, "para todo"), negação ($\neg$, "não"), conjunção ($\wedge$, "e"), disjunção($\vee$, "ou"), condicional ($\rightarrow$, "se ... então"), e bicondicional ($\leftrightarrow$, "equivalente" ou "se, e somente se").

1.1.1 As Bases da Lógica Clássica

A lógica clássica é baseada em dois princípios: da extensionalidade e da bivalência. A extensionalidade, na lógica clássica proposicional, estabelece que o valor-verdade de uma asserção depende apenas dos valores-verdade de seus componentes; na lógica de predicados de primeira ordem, ela estabelece que cada conceito é completamente determinado pela classe de todos os objetos que apresentam esse conceito, ou seja, pelo seu domínio.

A consideração de que a toda proposição pode ser atribuído exatamente um de dois valores-lógicos possíveis, verdadeiro e falso, denomina-se "Princípio da Bivalência". Esse princípio determina o escopo e o objeto da aplicação da lógica clássica sendo expresso através de duas leis:

Terceiro Excluído (*tertium non datur*) $\quad p \vee \neg p$

Princípio da Contradição (*principium contradictionis*) $\neg(p \wedge \neg p)$

Dado um entendimento clássico dos conectivos lógicos $\vee$, $\wedge$ e $\neg$, o *Terceiro Excluído* estabelece que a uma proposição p ser atribuído um valor-verdade verdadeiro ou falso, apenas. Já o *Princípio da contradição* determina que uma proposição p não pode em um mesmo momento assumir os valores-verdade verdadeiro e falso. Pode-se também utilizar também o termo *não-contradição* para nominar esse princípio, vide [10].

Uma forma natural e direta de ir além da lógica clássica é a introdução de mais valores-verdade, rejeitando o "Princípio da Bivalência". De forma geral, surgem daí as denominadas lógicas não-clássicas.

1.2 Classificação das Lógicas

Haack [9] considera duas categorias principais de lógicas não-clássicas: as que são apresentadas como complementares da clássica e as lógicas alternativas a ela. As do primeiro tipo seguem os princípios básicos da lógica clássica, todavia ampliam ou complementam o seu escopo. De forma geral, a linguagem clássica é enriquecida com a introdução de novos operadores. São exemplos de lógicas complementares, as lógicas modais, com os operadores modais de possibilidade e necessidade; as lógicas deônticas, com os operadores deônticos proibido, permitido, indiferente e obrigatório; as lógicas temporais, etc.

As lógicas do segundo tipo, também ditas heterodoxas foram concebidas como novas lógicas, destinadas a substituir a lógica clássica em alguns domínios do conhecimento. Como exemplo de lógicas heterodoxas temos aquelas onde não vale a lei reflexiva da identidade, as chamadas lógicas não-reflexivas, por exemplo, a lógica quântica. Em lógicas paracompletas não é válido o *Princípio do Terceiro-Excluído*, ou seja, pode existir uma fórmula

A de tal forma que nem A e nem a negação de A não são teoremas. São lógicas paracompletas as lógicas intuicionistas e as lógicas multivalentes. O *Princípio da Contradição* não é aceito nas lógicas relevantes e nas lógicas paraconsistentes. Existe ainda a lógica nebulosa (*fuzzy logic*) com muitas aplicações na área de engenharia.

1.2.1 As Lógicas Multivalentes

As lógicas multivalentes, como uma parte separada da lógica clássica, foram criadas nos trabalhos de J. Łukasiewicz [15] e E. L. Post [19] no início da década de 20, século XX. Esses autores não foram os primeiros a não assumirem como verdade o "Princípio da Bivalência", mas aqueles que verdadeiramente influenciaram o seu desenvolvimento.

Jan Łukasiewicz introduziu sistemas de lógicas multivalentes em uma tentativa de investigar proposições modais e noções de possibilidade e necessidade intimamente relacionadas com essas proposições.

Os principais sistemas lógicos mutivalentes frequentemente são definidos como famílias de lógicas compreendendo sistemas finitos e infinitos. Gottwald [8] faz uma revisão completa dos sistemas lógicos proposicionais multivalentes, das lógicas multivalentes de primeira ordem e as aplicações dessas lógias. Dentre os sistemas lógicos proposicionais multivalentes podemos listar os mais importantes:

- lógicas de Łukasiewicz;
- lógicas de Gödel;
- *t-Norm* lógicas;
- lógicas trivalentes;

- lógica de 4-valores de Dunn-Belnap's;
- lógicas Produto.

Para exemplificar, apresentamos suscintamente duas famílias de lógicas: Lukasiewicz e Gödel, para maiores esclarecimentos vide [8].

Lukasiewicz logics

Os sistemas Ł$_m$ e Ł$_\infty$ são definidos pelas matrizes lógicas cujos os valores-verdade nos casos finitos são definidos em um conjunto finito de racionais V_m no intervalo real unitário: $V_m = \{k/m-1 \mid 0 \leqslant k \leqslant m-1\}$ e, no caso infinito, os valores-verdade pertencem a $V_m = [0,1] = \{x \in \mathbb{R} \mid 0 \leqslant x \leqslant 1\}$. O valor-verdade 1 é o único valor designado como verdadeiro. São definidas duas conjunções: a conjunção forte (&) e a conjunção fraca ($\wedge$).

Os conectivos desses sistemas são:
$a \,\&\, b = max\{0, u+v-1\}$;
$a \wedge b = min\{u, v\}$;
$\neg a = 1-a$;
$u \to v = min\{1, 1-u+v\}$.

Os conectivos disjunção forte e fraca também são definidos como abreviaturas usando as leis de De Morgan.

Lógicas de Gödel

Os sistemas G_m e G_∞ são definidas pelas matrizes lógicas cujos os valores-verdade nos casos finitos são definidos em um conjunto finito de racionais V_m no intervalo real unitário: $V_m = \{k/m-1 \mid 0 \leqslant k \leqslant m-1\}$ e, no caso infinito, os valores-verdade pertencem a $V_m = [0,1] = \{x \in \mathbb{R} \mid 0 \leqslant x \leqslant 1\}$. O valor-verdade 1 é o único valor designado como verdadeiro.

Os conectivos desses sistemas são:

$a \wedge b = min\{u, v\}$;

$a \vee b = max\{u, v\}$;

$$\text{conectivo } \neg \begin{cases} \neg u = 1, \text{se } u = 0, \\ \neg u = 0, \text{se } u \neq 0. \end{cases}$$

$$\text{conectivo } \rightarrow \begin{cases} u \rightarrow v = 1, \text{se } u \leqslant v, \\ u \rightarrow v = v, \text{se } u > v. \end{cases}$$

As lógicas acima foram definidas somente com seu componente semântico. Naturalmente, surge uma necessidade de axiomatização desses e outros sistemas lógicos multivalentes.

Existem na lógica clássica assim como nos diversos ramos das lógicas não-clássicas diferentes abordagens na direção da formulação de um cálculo lógico (*Logic Calculi*). Historicamente o primeiro sistema e ainda o mais estudado e reconhecido é o cálculo lógico do tipo hilbertiano. Em 1931, Wajsberg [24] apresentou uma axiomatização hilbertiana do cálculo Ł3 de Łukasiewicz. Nessa época, a questão de axiomatização das lógicas multivalentes finitas estava totalmente aberta.

1.3 O trabalho de Rosser e Turquette

Em 1952, J. B. Rosser e A. R. Turquette em [20] apresentam uma axiomatização para uma extensa classe de lógicas finitas multivalentes.

Inicialmente Rosser e Turquette assumem que o conjunto de conectivos das lógicas multivalentes finitas contém um conectivo binário denotado por $\rightarrow$, um tipo de implicação, e conectivos unários J_s para cada s pertencente ao conjunto dos valores-verdade do Sistema Lógico. Os valores-verdade são conjunto finito com uma partição definida por dois subconjuntos denomina-

dos designados (valores-verdade considerados verdadeiros) e não-designados (valores-verdade considerados falsos).

Esses conectivos especiais devem satisfazer duas condições [20]:

RT1 A função de valoração correspondente ao conectivo $\rightarrow$ assume um valor-verdade não-designado exatamente nos casos em que o primeiro argumento é um valor-verdade designado e o segundo argumento é um valor-verdade não-designado.

RT2 A função de valoração correspondente ao conectivo J_s, onde s é um valor-verdade qualquer do sistema lógico, assume um valor-verdade designado exatamente nos casos em que o seu único argumento é s e nos demais casos assumir um valor-verdade não-designado.

Visando a compactação da formulação dos axiomas, alguns "açúcares sintáticos" são necessários. Recursivamente define-se o operador $\ominus$ quaisquer fórmulas bem-formadas $H_1, H_2, \ldots, G$ da linguagem do sistema lógico:

$$\ominus_{i=1}^{0}(H_i, G) =_{def} G$$
$$\ominus_{i=1}^{k+1}(H_i, G) =_{def} H_{k+1} \rightarrow \ominus_{i=1}^{k}(H_i, G)$$

Adicionalmente, para cada conectivo φ no sistema lógico, a sua correspondente função valoração é ver_{φ}.

Então definiu-se Ax_{RT} como o conjunto de todas fórmulas bem-formadas que são uma instância de um dos seguintes esquemas de axiomas:

$\mathbf{Ax}_{RT}$1: $A \rightarrow (B \rightarrow A)$;

$\mathbf{Ax}_{RT}$2: $(A \rightarrow (B \rightarrow C)) \rightarrow (B \rightarrow (A \rightarrow C))$;

$\mathbf{Ax}_{RT}$3: $(A \rightarrow B)) \rightarrow ((B \rightarrow C) \rightarrow (A \rightarrow C))$;

$\mathbf{Ax}_{RT}$4: $(J_s(A) \to (J_s(A) \to B)) \to (J_s(A) \to B)$, para todo s valor-verdade do Sistema Lógico;

$\mathbf{Ax}_{RT}$5: $\ominus_{i=1}^{m}(J_{\frac{i-1}{m-1}}(A) \to B, B)$;

$\mathbf{Ax}_{RT}$6: $J_s(s)$, para todo s valor-verdade do Sistema Lógico;

$\mathbf{Ax}_{RT}$7: $J_t(A) \to A$, para todo t valor-verdade designado do Sistema Lógico;

$\mathbf{Ax}_{RT}$8: $\ominus_{i=1}^{n}(J_{s_i}(A_i), J_t(\varphi(A_1, \ldots, A_n)))$,
para todo conectivo n-árioφ, s_i valor-verdade do Sistema Lógico e para o valor-verdade $t = ver_{\varphi}(s_1, \ldots, s_n)$.

Como única regra de inferência, *Modus-Ponens*:

$$\frac{A,\ A \to B}{B}$$

que permite inferir uma fórmula bem-formada B dadas duas fórmulas bem-formadas A e $A \to B$.

Rosser e Turquette [20] demonstram os teoremas :

correção Suponha que o sistema lógico proposicional multivalente S satisfaz as condições **RT1** e **RT2**. Então o cálculo lógico K_{RT}^{n} é correto com respeito ao sistema S, i. e., cada fórmula bem-formada derivável em K_{RT}^{n} é uma S-tautologia.

completude Suponha que o sistema lógico proposicional multivalente S satisfaz as condições **RT1** e **RT2**. Então o cálculo lógico K_{RT}^{n} é completo com respeito ao sistema S, i. e., cada S-tautologia é derivável em K_{RT}^{n}.

As demonstrações desses dois teoremas ocupam **10 páginas** do manuscrito, a correção é uma demonstração mais simples, todavia a completude é não-trivial.

O método ROSSER-TURQUETTE é válido para uma grande quantidade de sistemas lógicos finitos multivalentes, particularmente para os sistemas lógicos multivalentes conhecidos. Sua maior restrição de uso são as necessidades de um conectivo $\rightarrow$ bem comportado e toda uma classe de conectivos J_t, para todo valor-verdade t no sistema lógico.

Posteriormente, Anshakov e Rychkov [1] apresentaram um novo método geral e efetivo para construir um cálculo lógico de tipo "quasi-Hilbert", que é completo com respeito a L_n-validade, para qualquer lógica multivalente finita. Todavia, como estabelecido por Siegfried Gottwald em seu livro "A Treatise on Many-Valued Logics" [8]:

> *"it would be quite welcome to have some general and uniform way to axiomatize logical systems of some type".*

Esse é o problema central dessas abordagens. Grzegorz Malinowski, no capítulo *"Manu-valued logic and its philosophy"* do livro *"Handbook of the history of logic (Volume 8)"* [16], afirma que as axiomáticas definidas por Rosser e Turquette são muito "complicadas" em relação aos sistemas axiomáticos construídos de forma "artesanal" para lógicas particulares. Existem na literatura inúmeros trabalhos cujo conteúdo basicamente é definir uma lógica com uma semântica desejável e construir um sistema axiomático para essa lógica que seja correto e completo, normalmente essas axiomáticas são definidas de forma a conter o menor número possível de regras e axiomas, e que esses tenham o menor tamanho possível.

No capítulo 3 são apresentados os conceitos sintáticos e semânticos fundamentais de lógicas multivalentes. São conceituadas e definidas as matrizes corretas, as suas extensões corretas e então demonstramos dois teoremas: o primeiro, determina que a completude pode ser determinada como um resultado de um procedimento de verificação de tautologias para uma matriz

e todas as suas extensões corretas; o segundo, determina que a completude pode ser demostrada como um resultado positivo de um procedimento de verificação da repetição entre uma matriz candidata e suas extensões unitárias corretas. Através desse último teorema temos um algoritmo para a demonstração da completude em sistemas lógicos finitos multivalentes.

Capítulo 2

Uma Nova Proposta de Demonstração da Completude

Esse capítulo faz o tratamento formal das lógicas multivalentes. São demonstrados os dois resultados (teoremas) desse trabalho. Inicialmente, são apresentadas as definições e a linguagem matemática necessárias.

2.1 Introdução

Primeiramente, definiremos uma teoria axiomática $\mathcal{L}$ para o cálculo proposicional clássico. Como vocabulário, os símbolos de $\mathcal{L}$ são $\neg$, $\rightarrow$, (,) e letras P com índices inteiros: $\mathcal{A}, P_2 \ldots$ As letras P_i são denominadas proposições atômicas e os símbolos $\neg$ e $\rightarrow$ são os conectivos primitivos. Definimos então uma fórmula bem formada (fbf):

1. todas as letras proposicionais são fbf's;
2. se $\mathcal{A}$ e $\mathcal{B}$ são fbf's, então $(\neg\mathcal{A})$ e$(\mathcal{A} \rightarrow \mathcal{B})$ são fbf's.

Sistema axiomático – lógica proposicional clássica (Mendelson)[18]:

1. Se $\mathcal{A}$, $\mathcal{B}$ e $\mathcal{C}$ são fbf's de $\mathcal{L}$, então definimos os esquemas de axiomas e regras de inferência de $\mathcal{L}$:

   (Esquema Ax. 1) $(\mathcal{A} \to (\mathcal{B} \to \mathcal{A}))$

   (Esquema Ax. 2) $(\mathcal{A} \to (\mathcal{B} \to \mathcal{C})) \to ((\mathcal{A} \to \mathcal{B}) \to (\mathcal{A} \to \mathcal{C})))$

   (Esquema Ax. 3) $(((\neg\mathcal{B}) \to (\neg\mathcal{A})) \to (((\neg\mathcal{B}) \to \mathcal{A}) \to B))$

2. Regra de inferência – *Modus Ponens (MP)*: $\mathcal{B}$ é uma consequência direta de $\mathcal{A}$ e $(\mathcal{A} \to \mathcal{B})$.

$\mathcal{L}$ apresenta um conjunto infinito de axiomas através dos três esquemas de axiomas (Axioma 1, Axioma 2 e Axioma 3), onde cada um desses esquemas representa um número infinito de axiomas. Em $\mathcal{L}$, facilmente podemos verificar se uma dada fbf é ou não um axioma.

Como abreviaturas, definimos os conectivos: $(\mathcal{A} \wedge \mathcal{B})$ para $\neg(\mathcal{A} \to (\neg\mathcal{B}))$, $(\mathcal{A} \vee \mathcal{B})$ para $((\neg\mathcal{A}) \to \mathcal{B})$ e $(\mathcal{A} \leftrightarrow \mathcal{B})$ para $(\mathcal{A} \to \mathcal{B}) \wedge (\mathcal{B} \to \mathcal{A})$.

Uma prova em $\mathcal{L}$ é uma sequência $A_1, A_2 \ldots, A_k$ de fbf's tal que, para cada i, ou A_i é um axioma de $\mathcal{L}$, ou A_i é uma consequência direta de duas fbf's precedentes utilizando a regra de inferência *Modus Ponens*. Um teorema de $\mathcal{L}$ é uma fbf A de $\mathcal{L}$ se existe uma prova na qual a última fbf é A.

Como exemplo, demonstraremos que a fórmula bem formada $(\mathcal{A} \to \mathcal{A})$ é um teorema. Demonstração:

1.	$\mathcal{A} \to ((\mathcal{A} \to \mathcal{A}) \to \mathcal{A})$	(Instância do Esquema Ax. 1)
2.	$(\mathcal{A} \to ((\mathcal{A} \to \mathcal{A}) \to \mathcal{A})) \to ((\mathcal{A} \to (\mathcal{A} \to \mathcal{A})) \to (\mathcal{A} \to \mathcal{A}))$	(Instância do Esquema Ax. 2)
3.	$(\mathcal{A} \to (\mathcal{A} \to \mathcal{A})) \to (\mathcal{A} \to \mathcal{A})$	$1, 2$ *e* MP
4.	$\mathcal{A} \to (\mathcal{A} \to \mathcal{A})$	(Esquema Ax. 1)
5.	$\mathcal{A} \to \mathcal{A}$	$4, 3$ *e* MP

Aqui, surge naturalmente uma questão: "Quais são todas as possíveis semânticas adequadas para essa lógica?"

Podem existir matrizes características de dois, três ou mais valores-verdade para esse sistema axiomático. No livro *Introduction to Mathematical Logic*, Elliot Mendelson [18] demonstra que existe uma matriz característica para esse sistema lógico definido acima basedo em dois conectivos, $\rightarrow$ e $\neg$, denotada por M^C:

$\rightarrow$	T	F
T	T	F
F	T	T

$\neg$	
T	F
F	T

M^C é a menor semântica matricial característica para o sistema axiomático acima descrito. Ou seja, o sistema axiomático é bivalente. Trata-se da lógica clássica proposicional.

Se analisarmos apenas o teorema $P \rightarrow P$, questionando-nos sobre quais são as possíveis matrizes que o verificam, teremos como conclusão que ele será validado por todas matrizes do conectivo $\rightarrow$ onde os elementos da diagonal principal apresentam valores designados (considerados verdadeiros).

É elementar notar que cada axioma ou regra de inferência induz uma restrição quanto às possíveis matrizes que os verificam. Por exemplo, existe apenas 1 (uma) semântica de matrizes com dois valores que verificam os axiomas 1 a 3 e onde a regra de inferência *Modus Ponens* é preservada. Isso pode ser mostrado através de um procedimento algorítmico, basta construir todas as 64 possibilidades de matrizes dos conectivos $\rightarrow$ e $\neg$ com 2 valores-verdades, verdadeiro e falso, e então verificar em quais matrizes os axiomas são tautologias e a regra de inferência é preservada. O fato do resultado apresentar apenas 1 (uma) semântica matricial com essa propriedade (as matrizes clássicas dos conectivos $\rightarrow$ e $\neg$) é plenamente esperado, pois a semântica matricial

é correta em relação ao sistema axiomático.

Outro fato plenamente esperado, podemos verificar que dentre todas as extensões de M^C através da adição de um único valor-verdade, designado ou não, se tomarmos apenas aquelas nas quais os axiomas de $\mathcal{L}$ são tautologias e a regra *Modus Ponens* é preservada, teremos um total de 18 extensões com uma característica peculiar, todas são repetições da matriz clássica.

2.2 A Lógica Multivalente $\mathcal{L}^{\mathcal{I}}$

Nessa seção será analisado o sistema axiomático $\mathcal{L}^{\mathcal{I}}$ definido por:

Sejam $\mathcal{A}$, $\mathcal{B}$ e $\mathcal{C}$ fbf's de $\mathcal{L}^{\mathcal{I}}$.

1. Esquema de axiomas:

   (Esquema Ax. 1) $(\mathcal{A} \to (\mathcal{B} \to \mathcal{A}))$

   (Esquema Ax. 2) $(\mathcal{A} \to (\mathcal{B} \to \mathcal{C})) \to ((\mathcal{A} \to \mathcal{B}) \to (\mathcal{A} \to \mathcal{C})))$

   (Esquema Ax. 3) $(\neg\neg\mathcal{A} \to \neg\mathcal{B}) \to ((\neg\neg\mathcal{A} \to \mathcal{B}) \to \neg\mathcal{A})$

   (Esquema Ax. 4) $(\neg\neg(\mathcal{A} \to \mathcal{B}) \to (\mathcal{A} \to \mathcal{B}))$

2. Regra de inferência – *Modus Ponens (MP)*: $\mathcal{B}$ é uma consequência direta de $\mathcal{A}$ e $(\mathcal{A} \to \mathcal{B})$.

Naturalmente, surgem algumas questões:

Existe uma semântica finita adequada para $\mathcal{L}^{\mathcal{I}}$?

Qual é uma matriz característica de $\mathcal{L}^{\mathcal{I}}$?

Temos que $\mathcal{L}^{\mathcal{I}}$ apresenta 4 esquemas de axiomas (ou, simplesmente axiomas) e *Modus Ponens*. Todos esses axiomas são tautologias clássicas. Será M^C uma matriz característica de $\mathcal{L}^{\mathcal{I}}$?

Inicialmente, façamos uma análise das possíveis semânticas matriciais bivalentes. Existem $2^6 = 64$ matrizes bivalentes com os conectivos $\rightarrow$ e $\neg$. Dessas, somente duas verificam todos os $\mathcal{L}^{\mathcal{I}}$-axiomas e onde *modus ponens* é preservada:

1. | $\rightarrow$ | T | F |
   |---|---|---|
   | T | T | T |
   | F | T | T |

$\neg$	
T	T
F	T

   onde todas as fbf's são tautologias, e

2. | $\rightarrow$ | T | F |
   |---|---|---|
   | T | T | F |
   | F | T | T |

$\neg$	
T	F
F	T

   a matriz clássica (M^C) para $\rightarrow$ e $\neg$.

É um fato que ambas matrizes não são características de $\mathcal{L}^{\mathcal{I}}$. Adicionando 1 (um) valor-verdade (designado ou não), existem $3^6 = 729$ matrizes para $\rightarrow$ e $\neg$ que estendem M^C. Dessas, 19 são matrizes trivalentes que verificam todos os $\mathcal{L}^{\mathcal{I}}$-axiomas e preservam *modus ponens*. No entanto, 18 dessas matrizes são repetições da matriz clássica, e existe uma matriz distinta de todas as demais, denotada por M^I que apresenta um valor-verdade não-designado adicional:

	$\neg$
T	F
F	T
I	**F**

$\rightarrow$	T	F	**I**
T	T	F	**F**
F	T	T	**T**
I	**T**	**T**	**T**

Pode-se notar que I é um valor-verdade "novo", distinto de T e F. Isso **sugere** que M^C não é uma matriz característica para $\mathcal{L}^{\mathcal{I}}$ (é correta, mas o é completa). Temos então que M^I é uma matriz candidata a ser característica da lógica $\mathcal{L}^{\mathcal{I}}$

Novamente, construímos extensões da matriz M^I com a adição de um novo valor-verdade, designado ou não. Nesse caso, existem $4^8 (= 65536)$

matrizes de 4 valores-verdade possíveis para $\mathcal{L}^I$ que estendem M^I. Mostra-se através de programas computacionais que apenas 39 dessas matrizes verificam todos os $\mathcal{L}^I$-axiomas e onde *modus ponens* é preservada.

Como resultado do programa, é fato que todas as 39 matrizes de 4 valores-verdade são repetições de M^I, i. e., todas essas matrizes apresentam o mesmo conjunto de tautologias. Ou seja, quando tentou-se construir matrizes com 4 valores-verdade, todas elas se comportaram como a matriz M^I de 3 valores-verdade, isso nos faz crer intuitivamente que M^I é uma matriz característica para $\mathcal{L}^I$. Em 1995, Sette e Carnielli [3] demostraram que "M^I é uma matriz característica para $\mathcal{L}^I$".

Investigar e precisamente explicar a razão desses fenômenos ocorrerem é o propósito principal deste trabalho. Isto é intimamente relacionado com o problema da determinação de um cálculo lógico $\mathcal{C}$ que provê uma adequada axiomatização para uma dada lógica proposicional multivalentes S. Esta tese introduz um novo caminho para demonstrar a completude de lógicas finitas multivalentes.

A partir desse marco, o texto tornar-se-á mais rigoroso quanto às definições de lógica multivalente visando uma demonstração formal de todos os lemas e teoremas.

2.3 Definições Básicas

Definição 1 (Lógicas)

1. Uma **assinatura** Σ é um conjunto finito $\{\phi_1, \ldots, \phi_n\}$ de símbolos funcionais que representam conectivos lógicos ou constantes. Define-se s_i a aridade de ϕ_i para todo i, $1 \leqslant i \leqslant n$.

2. Seja $P = \{p_0, \ldots, p_n, \ldots\}$ o conjunto de variáveis proposicionais. En-

tão, para uma assinatura dada Σ, Fbf(Σ) denota o conjunto de fórmulas proposicionais bem formadas construídas na forma usual a partir do conjunto P de variáveis proposicionais e os conectivos ou constantes em Σ. Também **subfórmulas** apresentam uma definição usual, vide [18].

3. Uma **substituição** ρ é uma função de P em Fbf(Σ). Se $\mathcal{F}$ é uma fórmula e ρ: $P \to$ Fbf(Σ) uma substituição, então $\rho(\mathcal{F})$ é o resultado de simultaneamente aplicar-se a substituição ρ nas variáveis proposicionais em $\mathcal{F}$.

4. Uma **Lógica** $\mathcal{L}$ consiste em uma assinatura Σ, um conjunto de axiomas A e um conjunto de regras R. Axiomas e regras apresentam definições usuais, vide [18].

5. Uma **prova** em uma lógica $\mathcal{L} = (\Sigma, A, R)$ é uma sequência finita de fórmulas $\mathcal{F}_1, \ldots, \mathcal{F}_n$ em Fbf (Σ) tal que para cada i, no domínio $1 \leqslant i \leqslant n$:

   - $\mathcal{F}_i$ é uma instância de um axioma, i.e., existe uma $\mathcal{G} \in A$ e uma substituição ρ com $\rho(\mathcal{G}) = \mathcal{F}_i$ ou
   - $\mathcal{F}_i$ é obtida através da uma instância de uma regra em R a partir das fórmulas que ocorreram previamente na prova, i.e., existe uma regra em R

     $$\frac{\mathcal{G}_1,\ldots,\mathcal{G}_n}{\mathcal{H}}$$

     e uma substituição ρ tal que $\{\rho(\mathcal{G}_1), \ldots, \rho(\mathcal{G}_n)\} \subseteq \{\mathcal{F}_1, \ldots, \mathcal{F}_{i-1}\}$ e $\rho(\mathcal{H}) = \mathcal{F}_i$.

6. $\mathcal{F}_n$ é **derivável** em uma lógica $\mathcal{L}$ se $\mathcal{F}_1, \ldots, \mathcal{F}_n$ é uma prova em $\mathcal{L}$.

Após a definição dos aspectos sintáticos de uma lógica, introduziremos os conceitos semânticos.

Definição 2 (Matrizes) Uma **Matriz** $M = (T, f_1, \ldots, f_m, D)$ para uma assinatura dada $\Sigma = \{\phi_1, \ldots, \phi_n\}$ é uma álgebra finita de assinatura Σ onde:

- $T = \{a_1, \ldots, a_k\}$ é um universo finito e
- $f_1, \ldots, f_m$ são funções tais que f_i e ϕ_i apresentam a mesma aridade (s_i) para todo i, $f_i \colon T^{s_i} \to T$, $1 \leqslant i \leqslant m$ e
- um conjunto D de valores-verdade designados, $D \subseteq T$.

Definição 3 (Tautologias)

1. Cada fórmula $\mathcal{F} \in Fbf(\Sigma)$, na qual r é o maior índice de uma variável p_r que ocorre em $\mathcal{F}$, induz na forma usual uma valoração, i. e., uma função $e_M^{\mathcal{F}} \colon T^r \to T$.
2. Uma fórmula $\mathcal{F}$ é uma **tautologia** em uma matriz $M = (T, f_1, \ldots, f_m, D)$ se a função $e_M^{\mathcal{F}}$ retorna somente valores-verdade designados, i. e., para todo $c_1, \ldots, c_r \in T$, $e_M^{\mathcal{F}}(c_1, \ldots, c_r) \in D$. Nesse caso, $\mathcal{F}$ é simplesmente denominado uma M-tautologia.
3. O conjunto de todas as M-tautologias é denotado por $\mathbf{Taut(M)}$.

Definição 4 (Homomorfismos)
Sejam $M_1 = (T_1, f_1^1, \ldots, f_m^1, D_1)$ e $M_2 = (T_2, f_1^2, \ldots, f_m^2, D_2)$ duas matrizes de mesma assinatura Σ. Uma função $h \colon T_1 \to T_2$ é um **homomorfismo** de M_1 em M_2 se

- para qualquer i, $1 \leqslant i \leqslant m$ e qualquer $c_1, \ldots, c_{s_i} \in T_1$
$$h(f_i^1(c_1, \ldots, c_{s_i})) = f_i^2(h(c_1), \ldots, h(c_{s_i}))$$
onde s_i é a aridade de f_i e

- $h(D_1) \subseteq D_2$.

Definição 5 (Repetições)
Sejam $M_1 = (T_1, f_1^1, \ldots, f_m^1, D_1)$ e $M_2 = (T_2, f_1^2, \ldots, f_m^2, D_2)$ duas matrizes de mesma assinatura Σ. M_1 é uma repetição de M_2 se, e somente se, existe uma função $h\colon T_1 \to T_2$ tal que:

- $h(T_1) = T_2$;
- para qualquer a, $a \in D_1$ se, e somente se, $h(a) \in D_2$;
- para qualquer i, $1 \leqslant i \leqslant m$ e qualquer $c_1, \ldots, c_{s_i} \in T_1$
$$h(f_i^1(c_1, \ldots, c_{s_i})) = f_i^2(h(c_1), \ldots, h(c_{s_i}))$$
onde s_i é a aridade de f_i;
- $h(a) = a$ para todo $a \in T_2$.

Definição 6 (Correção e Completude)
Seja $\mathcal{L} = (\Sigma, A, R)$ uma lógica e M uma matriz para Σ.

1. $\mathcal{L}$ é **correto** em M se toda instância de um axioma em A é uma M-tautologia e para qualquer regra
$$\frac{B_1,\ldots,B_n}{C}$$
em R e uma substituição ρ tal que $\rho(B_1), \ldots, \rho(B_n)$ são M-tautologias também $\rho(C)$ é uma M-tautologia, ou simplesmente, R é preservada em M.
2. $\mathcal{L}$ é **completa** em M se toda M-tautologia é derivável em $\mathcal{L}$.

Se $\mathcal{L}$ é correta em M então podemos verificar de forma direta que toda fórmula derivável em $\mathcal{L}$ é uma M-tautologia.

Definição 7 (Matriz Característica)

M é uma **matriz característica** para $\mathcal{L}$ se $\mathcal{L}$ é correta e completa em M, vide [5].

As definições acima serão exemplificadas no decorrer desse capítulo. Um sistema lógico proposicional trivalente denominado P^1 [21], sempre que possível, será utilizado para ilustrar todas definições e resultados. Esse sistema proposicional P^1 foi criado pelo professor Antônio Mário Antunes Sette e tem como característica peculiar ser a mais simples sublógica paraconsistente do cálculo proposicional. A seguir apresentamos formalmente a lógica proposicional paraconsistente P^1.

Definição 8 (Lógica P^1)

A lógica P^1 é definida como (Σ, A, R), onde

- $\Sigma = \{\rightarrow, \neg\}$
- A é um conjunto de esquemas de axiomas $A1$, $A2$, $A3$, $A4$ e $A5$:

  A1 - $p_0 \rightarrow (p_1 \rightarrow p_0)$

  A2 - $(p_0 \rightarrow (p_1 \rightarrow p_2)) \rightarrow ((p_0 \rightarrow p_1) \rightarrow (p_0 \rightarrow p_2))$

  A3 - $(\neg p_0 \rightarrow \neg p_1) \rightarrow ((\neg p_0 \rightarrow \neg\neg p_1) \rightarrow p_0)$

  A4 - $\neg(p_0 \rightarrow \neg\neg p_0) \rightarrow p_0$

  A5 - $(p_0 \rightarrow p_1) \rightarrow \neg\neg(p_0 \rightarrow p_1)$

- R é um conjunto de regras de inferência, um conjunto unitário com:

  Modus Ponens - $\dfrac{A,\ A \rightarrow B}{B}$

Uma Matriz M_{P^1} para a lógica P^1 é definida como:
$M_{P^1} = (\{T_0, T_1, F\}, f_{\rightarrow}, f_{\neg}, \{T_0, T_1\})$, onde:

$f_\rightarrow$	T_0	T_1	F
T_0	T_0	T_0	F
T_1	T_0	T_0	F
F	T_0	T_0	T_0

$f_\neg$	
T_0	F
T_1	T_0
F	T_0

.

M_{P^1} é uma **matriz característica** para P^1, como demonstrado por Sette [21].

Definição 9 (Extensões)
Sejam $M_1 = (T_1, f_1^1, \ldots, f_m^1, D_1)$ e $M_2 = (T_2, f_1^2, \ldots, f_m^2, D_2)$ duas matrizes de mesma assinatura Σ. M_2 é uma **extensão** de M_1 se

- $T_1 \subseteq T_2$ e
- $D_1 \subseteq D_2$ e
- para todo i as funções f_i^1 e f_i^2 coincidem em $T_1^{s_i}$, onde s_i é a aridade de f_i^1 e f_i^2.

Ambos, $D_1 = D_2$ ou $D_1 \subset D_2$ são permitidos. Na formalização algébrica, M_1 é uma subálgebra de M_2.

Definição 10 (Extensões Corretas)
Seja $\mathcal{L}$ uma lógica com assinatura Σ, sejam M_1 e M_2 duas matrizes de mesma assinatura Σ. M_2 é uma **extensão correta de M_1 com respeito a $\mathcal{L}$** se

- $\mathcal{L}$ é correta para M_1, e
- $\mathcal{L}$ é correta para M_2, e
- M_2 é uma extensão de M_1.

Por exemplo, seja $M_c = (\{T_0, F\}, f_\rightarrow, f_\neg, \{T_0\})$ uma matriz para a lógica P^1, onde:

$f_\rightarrow$	T_0	F
T_0	T_0	F
F	T_0	T_0

$f_\neg$	
T_0	F
F	T_0

.

É imediato que M_{P^1} é uma extensão de M_c. Uma vez que a lógica P^1 é correta para $\mathbf{M_c}$ e M_{P^1}, por definição, M_{P^1} é uma extensão correta de M_c.

Definição 11 (Elementos Marcados)

Seja $M = (T, f_1, \ldots, f_m, D)$ uma matriz para uma assinatura dada Σ. Um **Elemento Designado Marcado** de M é um elemento qualquer x^d tal que $x^d \in D$. Se $T - D \neq \emptyset$, similarmente, um **Elemento Não-designado Marcado** de M é um elemento qualquer x^{nd} tal que $x^{nd} \in (T - D)$.

Definição 12 (Extensões Relativas)

Sejam $M_1 = (T_1, f_1^1, \ldots, f_m^1, D_1)$ e $M_2 = (T_2, f_1^2, \ldots, f_m^2, D_2)$ duas matrizes de mesma assinatura Σ, onde $T_1 \cap T_2 = \emptyset$. Seja x_1^d um elemento designado marcado de M_1, e se $(T_1 - D_1) \neq \emptyset$, seja x_1^{nd} um elemento não-designado de M_1. Então $M_{12} = (T_{12}, f_1^{12}, \ldots, f_m^{12}, D_{12})$ de mesma assinatura Σ é uma **extensão de M_1 relativa a M_2** se:

- $T_{12} = T_1 \cup T_2$, e
- $D_{12} = D_1 \cup D_2$, e
- Para todo i, $1 \leqslant i \leqslant m$,

$$f_i^{12}(x_1, \ldots, x_{s_i}) = \begin{cases} f_i^1(x_1, \ldots, x_{s_i}), & \text{se } (x_1, \ldots, x_{s_i}) \in T_1^{s_i}; \\ f_i^2(x_1, \ldots, x_{s_i}), & \text{se } (x_1, \ldots, x_{s_i}) \in T_2^{s_i}; \\ f_i^1(x_1', \ldots, x_{s_i}'), & \text{caso contrário,} \\ & \text{onde, para todo j, } 1 \leqslant \mathrm{j} \leqslant \mathrm{s_i}: \\ & x_j' = \begin{cases} x_j, & \text{se } x_j \in T_1; \\ x_1^d, & \text{se } x_j \in D_2; \\ x_1^{nd}, & \text{se } x_j \in (T_2 - D_2). \end{cases} \end{cases}$$

Por exemplo, seja $M_{cP^1} = (\{T_0, F, T_0^\star, T_1^\star, F^\star\}, f_\rightarrow, f_\neg, \{T_0, T_0^\star, T_1^\star\})$ uma matriz que é uma extensão de M_c relativa a M_{P^1} para P^1, onde:

$f_\rightarrow$	T_0	F	$T_0^\star$	$T_1^\star$	$F^\star$
T_0	T_0	F	T_0	T_0	F
F	T_0	T_0	T_0	T_0	T_0
$T_0^\star$	T_0	F	$T_0^\star$	$T_0^\star$	$F^\star$
$T_1^\star$	T_0	F	$T_0^\star$	$T_0^\star$	$F^\star$
$F^\star$	T_0	T_0	$T_0^\star$	$T_0^\star$	$T_0^\star$

	$f_\neg$
T_0	F
F	T_0
$T_0^\star$	$F^\star$
$T_1^\star$	$T_0^\star$
$F^\star$	$T_0^\star$

.

Note que M_{cP^1} é uma extensão de M_c relativa a $M_{P^1}^\star$, onde $M_{P^1}^\star$ é a matriz $M_{P^1} = (\{T_0, T_1, F\}, f_\rightarrow, f_\neg, \{T_0, T_1\})$ resultado da concatenação do símbolo $\star$ a todos os elementos de $\{T_0, T_1, F\}$, $x_c^d = T_0$ e $x_c^{nd} = F$.

2.4 Teoremas

Com as definições acima, podemos passar às demonstrações dos teoremas fundamentais. Entretanto, para torná-las claras alguns lemas são necessários.

Lema 1 Sejam $M_1 = (T_1, f_1^1, \ldots, f_m^1, D_1)$ e $M_2 = (T_2, f_1^2, \ldots, f_m^2, D_2)$ duas matrizes de mesma assinatura $\Sigma = \{\phi_1, \ldots, \phi_m\}$, onde $T_1 \cap T_2 = \emptyset$. Seja $M_{12} = (T_{12}, f_1^{12}, \ldots, f_m^{12}, D_{12})$ uma extensão de M_1 relativa a M_2. Se a lógica $\mathcal{L}$ é correta para M_1 e M_2, então $\mathcal{L}$ é correta para M_{12}.

Demonstração 1 *Suponha que $\mathcal{L} = (\Sigma, A, R)$ não é correta para M_{12}. Então, são duas as possibilidades: (a) existe uma fórmula $\varphi \in A$ tal que φ não é uma M_{12}-tautologia ou (b) existe uma regra $r \in R$ que não é preservada em M_{12}.*

Primeira possibilidade (a)

Por definição, existe uma valoração $e_{M_{12}}^\varphi \colon T_{12}^r \rightarrow T_{12}$, onde r

é o maior índice de uma variável p_r que ocorre em φ, tal que $e^{\varphi}_{M_{12}}(c_1,\ldots,c_r) \notin D_{12}$. Sem perda de generalidade, φ tem uma forma $\phi_i^{12}(\mathcal{F}_1,\ldots,\mathcal{F}_{s_i})$, para algum i, $1 \leqslant i \leqslant m$, onde $\mathcal{F}_1,\ldots,\mathcal{F}_{s_i}$ são subfórmulas de φ e para todo j, $1 \leqslant j \leqslant s_i \leqslant r$, $e^{\varphi}_{M_{12}}(\mathcal{F}_j) \in \{c_1,\ldots,c_r\}$ e $e^{\varphi}_{M_{12}}(c_1,\ldots,c_r) = f_i^{12}(e^{\varphi}_{M_{12}}(\mathcal{F}_1),\ldots,e^{\varphi}_{M_{12}}(\mathcal{F}_{s_i}))$.
Agora, existem três possibilidades:

1. *se $\{e^{\varphi}_{M_{12}}(\mathcal{F}_1),\ldots,e^{\varphi}_{M_{12}}(\mathcal{F}_{s_i})\} \subseteq T_1$, como M_{12} é uma extensão de M_1 relativa a M_2, segue que*

$$f_i^{12}(e^{\varphi}_{M_{12}}(\mathcal{F}_1),\ldots,e^{\varphi}_{M_{12}}(\mathcal{F}_{s_i})) = f_i^{1}(e^{\varphi}_{M_{12}}(\mathcal{F}_1),\ldots,e^{\varphi}_{M_{12}}(\mathcal{F}_{s_i})),$$

   então $e^{\varphi}_{M_{12}}(c_1,\ldots,c_r) = e^{\varphi}_{M_1}(c_1,\ldots,c_r) \notin D_{12}$, pois $D_1 \subseteq D_{12}$, por consequência $e^{\varphi}_{M_1}(c_1,\ldots,c_r) \notin D_1$ e φ não é uma M_1-tautologia, uma contradição visto que $\mathcal{L}$ é correta para M_1;

2. *se $\{e^{\varphi}_{M_{12}}(\mathcal{F}_1),\ldots,e^{\varphi}_{M_{12}}(\mathcal{F}_{s_i})\} \subseteq T_2$, como M_{12} é uma extensão de M_1 relativa a M_2, segue que*

$$f_i^{12}(e^{\varphi}_{M_{12}}(\mathcal{F}_1),\ldots,e^{\varphi}_{M_{12}}(\mathcal{F}_{s_i})) = f_i^{2}(e^{\varphi}_{M_{12}}(\mathcal{F}_1),\ldots,e^{\varphi}_{M_{12}}(\mathcal{F}_{s_i})),$$

   então $e^{\varphi}_{M_{12}}(c_1,\ldots,c_r) = e^{\varphi}_{M_2}(c_1,\ldots,c_r) \notin D_{12}$, pois $D_2 \subseteq D_{12}$, por consequência $e^{\varphi}_{M_2}(c_1,\ldots,c_r) \notin D_2$ e φ não é uma M_2-tautologia, uma contradição visto que $\mathcal{L}$ é correta para M_2;

3. *finalmente, se $\{c_1,\ldots,c_r\} \subseteq T_{12}$, $\{c_1,\ldots,c_{s_i}\} \not\subseteq T_1$ e $\{c_1,\ldots,c_r\} \not\subseteq T_2$, como M_{12} é uma extensão de M_1 relativa a M_2, segue que*

$$f_i^{12}(e^{\varphi}_{M_{12}}(\mathcal{F}_1),\ldots,e^{\varphi}_{M_{12}}(\mathcal{F}_{s_i})) = f_i^{1}(e^{\varphi}_{M_{12}}(\mathcal{F}_1)',\ldots,e^{\varphi}_{M_{12}}(\mathcal{F}_{s_i})'),$$

   onde

$$\{e^{\varphi}_{M_{12}}(\mathcal{F}_1)',\ldots,e^{\varphi}_{M_{12}}(\mathcal{F}_{s_i})'\} \in T_1,$$

então $e^{\varphi}_{M_{12}}(c_1, \ldots, c_r) = e^{\varphi}_{M_1}(c_1, \ldots, c_r) \notin D_{12}$, *como* $D_1 \subseteq D_{12}$, *por consequência* $e^{\varphi}_{M_1}(c_1, \ldots, c_r) \notin D_1$ *e* φ *não é uma* M_1*-tautologia, uma contradição visto que* $\mathcal{L}$ *é correta para* M_1.

Segunda possibilidade (b)

Existe uma regra $r \in R$, $r = \frac{\mathcal{G}_1, \ldots, \mathcal{G}_n}{\mathcal{H}}$, *que não é preservada por* M_{12}. *Por definição,* $\mathcal{G}_1, \ldots, \mathcal{G}_n$ *são* M_{12}*-tautologias, todavia* $\mathcal{H}$ *não é uma* M_{12}*-tautologia. Como existe uma valoração* $e^{\mathcal{H}}_{M_{12}} \colon T^r_{12} \to T_{12}$, *onde* r *é o maior índice de uma variável* p_r *que ocorre em* $\mathcal{G}_1, \ldots, \mathcal{G}_n$ *e* $\mathcal{H}$, *tal que* $e^{\mathcal{H}}_{M_{12}}(c_1, \ldots, c_r) \notin D_{12}$. *Sem perda de generalidade,* $\mathcal{H}$ *tem uma forma* $\phi^{12}_i(\mathcal{F}_1, \ldots, \mathcal{F}_{s_i})$, *para algum* i, $1 \leqslant i \leqslant m$, *onde* $\mathcal{F}_1, \ldots, \mathcal{F}_{s_i}$ *são subfórmulas de* φ *e para todo* j, $1 \leqslant j \leqslant s_i$, $e^{\mathcal{H}}_{M_{12}}(\mathcal{F}_j) \in \{c_1, \ldots, c_r\}$ *e* $e^{\mathcal{H}}_{M_{12}}(c_1, \ldots, c_r) = f^{12}_i(e^{\mathcal{H}}_{M_{12}}(\mathcal{F}_1), \ldots, e^{\mathcal{H}}_{M_{12}}(\mathcal{F}_{s_i}))$. *Mutatis mutandis, o mesmo argumento utilizado no item (a) demonstra que uma contradição ocorre. Dessa forma,* $\mathcal{L} = (\Sigma, A, R)$ *é correta para* M_{12}. ■

Definição 13 (Extensões Corretas Relativas)

Seja $M_1 = (T_1, f^1_1, \ldots, f^1_m, D_1)$ e $M_2 = (T_2, f^2_1, \ldots, f^2_m, D_2)$ duas matrizes de mesma assinatura $\Sigma = \{\phi_1, \ldots, \phi_m\}$, onde $T_1 \cap T_2 = \emptyset$. Seja $M_{12} = (T_{12}, f^{12}_1, \ldots, f^{12}_m, D_{12})$ uma extensão de M_1 relativa a M_2. Se uma lógica $\mathcal{L} = (\Sigma, A, R)$ é correta para M_1 e M_2, pelo lema 1, M_{12} é uma **extensão correta de** M_1 **relativa a** M_2.

Lema 2 Seja $\mathcal{L}$ uma lógica de assinatura Σ. Se M_1 é uma matriz característica para $\mathcal{L}$, e M_2 é uma extensão correta de M_1, então Taut(M_1) =Taut(M_2).

Demonstração 2 *A inclusão* $Taut(M_1) \supseteq Taut(M_2)$ *trivialmente é verificada. Para mostrar que a inclusão reversa também é verdadeira, seja* $A \in Taut(M_1)$. *Como* $\mathcal{L}$ *é completa para* M_1, *existe uma prova* $B_1, \ldots, B_n$ *onde* $B_n = A$. *Como* $\mathcal{L}$ *é correta para* M_2, *segue diretamente que* $A \in Taut(M_2)$. ■

Lema 3 Seja $\mathcal{L}$ uma lógica com assinatura Σ. Se exisir uma matriz M_1 tal que $\mathcal{L}$ é correta para M_1, e Taut(M_1) =Taut(M_2), para toda extensão correta M_2 de M_1. Então M_1 é uma matriz característica para $\mathcal{L}$.

Demonstração 3 *Suponhamos que* $M_1 = (T_1, f_1^1, \ldots, f_m^1, D_1)$ *não é uma matriz característica para* $\mathcal{L}$*, ou seja,* $\mathcal{L}$ *é correta, mas não é completa para* M_1*, como um argumento do tipo* reductio ad absurdum. *Seja* $M_c = (T_c, f_1^c, \ldots, f_m^c, D_c)$ *uma matriz característica para* $\mathcal{L}$*, onde, sem perda de generalidade,* $T_c \cap T_1 = \emptyset$.

Seja $M_{1c} = (T_{1c}, f_1^{1c}, \ldots, f_m^{1c}, D_{1c})$ *uma extensão correta de* M_1 *relativa a* M_c. *Pelo lema 1,* M_{1c} *é uma extensão correta de* M_1 *e* M_c*, por hipótese, Taut*(M_1) =*Taut*(M_{1c})*, pelo lema 2, Taut*(M_c) =*Taut*(M_{1c})*, segue que Taut*(M_1) =*Taut*(M_c)*, i.e.,* M_1 *é uma matriz característica para* $\mathcal{L}$*, uma contradição.* ■

O próximo passo é demonstrar que "se uma dada matriz M é correta para $\mathcal{L}$ e todas as extensões corretas de M apresentam o mesmo conjunto de tautologias que M, então M é uma matriz característica para $\mathcal{L}$. Esse resultado nos fornece uma nova abordagem sobre a completude de sistemas lógicos finitos multivalentes.

É importante salientar que o problema de determinar se duas matrizes apresentam o mesmo conjunto de tautologias foi totalmente resolvido de forma independente por Łos [14] e Kalicki [11] [13].

Teorema 4 Seja $\mathcal{L}$ uma lógica com assinatura Σ, com matriz característica finita. M_1 é uma matriz característica para $\mathcal{L}$ se, e somente se, $\mathcal{L}$ é correta para M_1, e para toda extensão correta M_2 de M_1, Taut(M_1) =Taut(M_2).

Demonstração 4 *Pela definição 7 e lema 2, segue diretamente que: se* M_1 *é uma matriz característica para* $\mathcal{L}$*, então* $\mathcal{L}$ *é correta para* M_1*, e para todas as*

extensões corretas M_2 *de* M_1, *Taut*(M_1) =*Taut*(M_2). *Pelo lema 3, o inverso é válido.* ■

O teorema 4 determina que para demonstrar se uma matriz é característica de um sistema lógico finito multivalente deve-se verificar de forma exaustiva todas as sucessivas extensões corretas dessa matriz, mostrando que todas elas apresentam o mesmo conjunto de tautologias (isto é, todas essas matrizes são equivalentes).

O que foi estabelecido pelo teorema 4 não é um resultado computacionalmente efetivo, pois não há como gerar e testar todas as extensões corretas de uma dada matriz. Todavia, trata-se de uma caracterização do problema de demonstrar a completude para uma lógica finita multivalente qualquer.

2.4.1 Completude Automática

Objetiva-se encontrar um método algorítmico efetivo para demonstrar a completude lógicas finitas multivalentes. Para definirmos esse método necessitamos inicialmente da definição de um limitador, ou mesmo uma espécie de ponto-fixo na busca pelas extensões corretas da matriz considerada. Para tal, definimos as extensões corretas com apenas um valor-verdade adicional, ou simplesmente, as extensões unitárias corretas:

Definição 14 (Extensões Unitárias Corretas)
Seja $\mathcal{L}$ uma lógica com assinatura $\Sigma = \{f_1, \ldots, f_m\}$, sejam as matrizes $M_1 = (T_1, f_1, \ldots, f_m, D_1)$ e $M_2 = (T_2, f_1, \ldots, f_m, D_2)$. Então, M_2 é uma **extensão unitária correta de** M_1 **com respeito a** $\mathcal{L}$ se

- $\mathcal{L}$ é correta para M_1, e
- $\mathcal{L}$ é correta para M_2, e

- M_2 é uma extensão de M_1, e
- existe um único elemento $a \in T_2$ tal que $T_2 = T_1 \cup \{a\}$.

Com essa definição, podemos enunciar e demonstrar um teorema que permite construir um algoritmo para provar a completude de lógicas finitas multivalentes.

Teorema 5 Seja $\mathcal{L} = (\Sigma, A, R)$ uma lógica multivalente finita com assinatura Σ. Se M é uma matriz correta para $\mathcal{L}$ e todas as extensões unitárias corretas de M são repetições de M, então M é uma matriz característica de $\mathcal{L}$.

Demonstração 5 *Sem perda de generalidade, seja $M = M_n$, onde n é o número de valores-verdade de M_n. Suponhamos por hipótese do absurdo que M_n não é matriz característica de $\mathcal{L}$. Seja M_c a matriz finita característica de $\mathcal{L}$. Podemos construir todas as matrizes $M_{cn} = (T_{cn}, \Sigma, D_{cn})$ que são extensões corretas relativas de M_c em relação a M_n, como M_{cn} é extensão correta de M_c, é fato que M_{cn} é matriz característica de $\mathcal{L}$. Sem perda de generalidade consideremos uma matriz M_{cn} qualquer. Temos duas possibilidades:*
(1) Se M_{cn} é uma repetição de M_n, seque que M_n é matriz característica de $\mathcal{L}$, uma contradição.
(2) Se M_{cn} não é uma repetição de M_n, então temos que M_{cn} é uma matriz característica de $\mathcal{L}$ com pelo menos $n+1$ valores-verdade distintos. Para fixar ideias, seja v um valor-verdade correspondente a algum desses (n+1) valores-verdade distintos. A partir de M_n criamos as extensões unitárias denotadas por M_{nv}, com (n+1) valores-verdade como a seguir. Temos $M_n = (T_n, \Sigma, D_1)$ e $M_{nv} = (T_n \cup \{v\}, \Sigma, D_2)$, onde $D_2 = D_1 \cup \{v\}$. Seja x_n^d um elemento designado marcado de M_n e x_n^{nd} um elemento não-designado marcado de M_n.

Seja $\Sigma = f_1, \ldots, f_m$, *para todo* i, $1 \leqslant i \leqslant m$,

$$f_i^{nv}(x_1, \ldots, x_{s_i}) = \begin{cases} x_n^d, & se \quad f_i^{nv}(x_1, \ldots, x_{s_i}) \in D_{cn}; \\ x_n^{nd}, & se \quad f_i^{nv}(x_1, \ldots, x_{s_i}) \notin D_{cn}. \end{cases}$$

Por construção, analogamente ao lema 1, $\mathcal{L}$ *é correta para* M_{nv}, *com (n+1) valores-verdades distintos. Todavia, como hipótese temos que todas as extensões unitárias corretas de* M_n *são repetições de* M_n, *dessa forma, como* M_{nv} *é uma extensão unitária de* M_n, M_{nv} *é uma repetição de* M_n, *isto signfica que* M_{nv} *não apresenta (n+1) valores-verdade distintos, uma contradição.*

Por fim, concluímos que M_n *é matriz característica de* $\mathcal{L}$. ■

Esse teorema possibilita a criação de um procedimento computacional para a solução do problema da completude. Para determinar se uma matriz correta é característica de um sistema lógico proposicional multivalente basta determinar dentre todas as suas extensões unitárias aquelas que também são corretas. Posteriormente, fazer uma verificação de que todas essas extensões corretas unitárias são repetições da matriz candidata, o que é significa determinar que todas essas matrizes apresentam o mesmo conjunto de tautogias. Isso se dá através da verificação de um homomorfismo "especial" entre as matrizes consideradas, de acordo com o conceito de repetição (definição 5). No próximo capítulo são apresentados uma implementação do método de completude automática proposto na linguagem PROLOG e um exemplo de demonstração de completude do sistema lógico proposicional multivalente.

Capítulo 3

Provador da Completude de Lógicas Finitas Multivalentes

Nesse caítulo são apresentadas uma implementação do método de completude automática proposto na linguagem PROLOG e a demonstração de completude do sistema lógico proposicional trivalente Ł3 de Łukasiewicz que não é funcionalmente completa.

O método de demonstração da completude dos sistemas lógicos finitos multivalentes é construtivo e algorítmico. Para exemplificar e tonar prático o método foram implementadas essas demonstrações usando o método proposto nessa tese, os programas foram implementados em Prolog e estão totalmente listados no Apêndice.

3.1 Implementação do Método

(Passo 1) Seja `TVals` uma lista contendo os valores-verdade considerados. Para a verificação dos axiomas e regras de inferência necessita-se das valorações possíveis a eles aplicáveis. Por exemplo, para determinar valorizações

de 3 e 2 valores-verdade, respectivamente, temos o seguinte código Prolog:

```
triplesSetDet(TVals,TripList) :-
  findall([X,Y,Z],
          ( member(X,TVals),
            member(Y,TVals),
            member(Z,TVals) ),
            TriplesList),
  retractall(triples_set(_)),
  assert(triples_set(TripList)).

pairsSetDet(TVals,PairsList) :-
  findall([X,Y],
          ( member(X,TVals),
            member(Y,TVals) ),
            PairsList),
  retractall(pairs_set(_)),
  assert(pairs_set(PairsList)).
```

Ou seja, `pairsSetDet([t,i,f],PairList)` produz como resposta `PairList=[[t,t],[t,i],[t,f],[i,t],[i,i],[i,f],[f,t],[f,i],[f,f]]`.

(Passo 2) Nesse, são criadas todas as extensões corretas unitárias possíveis para a matriz original através da insersão de um novo valor-verdade, considerando esse valor designado ou não-designado. O predicado `extendedMatricesDet(TVals, Dist, NDist, Trip, Pair)` implementa essa funcionalidade. No exemplo abaixo, temos o predicado `extendedMatricesDet/5` adaptado para determinar as extensões corretas de uma semântica matricial trivalente, com dois conectivos (sendo 1 unário e o outro binário).

```
extendedMatricesDet(TVals,Dist,NDist,Triples,Pairs) :-
  member(A,TVals),
  member(B,TVals),
  member(C,TVals),
  member(D,TVals),
  member(E,TVals),
  member(F,TVals),
  member(G,TVals),
  member(H,TVals),
  truthTableVerification(A,B,C,D,E,F,G,H,
                         Dist,NDist,Triples,Pairs,TVals),
  fail.
extendedMatricesDet(_,_,_,_,_,_).
```

(Passo 3) Uma vez criada uma extensão unitária da matriz original, deve-se exaustivamente verificar se os axiomas são taulogogia e a regra de inferência é preservada na matriz estendida (predicado `truthTableVerification/11`. Por exemplo:

```
truthTableVerification(A,B,C,D,E,F,G,H,Dist,NDist,Triples,Pairs,TVals) :-
    extendImp(A,B,C,D,E,F,G,ImpTT),
    extendNeg(H,NegTT),
    axiom1(Pairs,ImpTT,Dist),
    axiom2(Triples,ImpTT,Dist),
    axiom3(Pairs,ImpTT,NegTT,Dist),
    axiom4(TVals,ImpTT,NegTT,Dist),
    modusPonens(Pairs,ImpTT,Dist), !,
    saveMatrices(ImpTT,NegTT,Dist,NDist).
```

Aqui são verificados os 4 axiomas e a regra de inferência *Modus Ponens*. As matrizes que são extensões unitárias corretas são também armazenadas.

A verificação dos axiomas é feita da forma:

```
/* axiom 1 a => (b => a) */
axiom1([[X,Y]|T],ImpTT,Dist) :-
    member([Y,X,A],ImpTT),
    member([X,A,Result],ImpTT),
    member(Result,Dist), !,
    axiom1(T,ImpTT,Dist).
axiom1([],_,_).

/* axiom 2 (a => b) => ((b => c) => (a => c)) */
axiom2([[A,B,C]|T],ImpTT,Dist) :-
    member([A,B,I1],ImpTT),
    member([B,C,I2],ImpTT),
    member([A,C,I3],ImpTT),
    member([I2,I3,I4],ImpTT),
    member([I1,I4,Result],ImpTT),
    member(Result,Dist), !,
    axiom2(T,ImpTT,Dist).
axiom2([],_,_).

/* axioma 3  (~a => ~b) => (b => a) */
axiom3([[A,B]|T],ImpTT,NegTT,Dist) :-
    member([A,NA],NegTT),
    member([B,NB],NegTT),
    member([NA,NB,I1],ImpTT),
    member([B,A,I2],ImpTT),
    member([I1,I2,Result],ImpTT),
    member(Result,Dist), !,
    axiom3(T,ImpTT,NegTT,Dist).
axiom3([],_,_,_).
```

```
/* axioma 4  ((p => ~p) => p) => p */
axiom4([A|T],ImpTT,NegTT,Dist) :-
    member([A,NA],NegTT),
    member([A,NA,I1],ImpTT),
    member([I1,A,I2],ImpTT),
    member([I2,A,Result],ImpTT),
    member(Result,Dist), !,
    axiom4(T,ImpTT,NegTT,Dist).
axiom4([],_,_,_).
```

A verificação de que *Modus Ponens* é preservada na matriz é implementada como:

```
modusPonens([[A,B]|T],ImpTT,Dist) :-
    member(A,Dist),
    member([A,B,R],ImpTT),
    member(R,Dist), !,
    member(B,Dist), !,
    modusPonens(T,ImpTT,Dist).
modusPonens([_|T],ImpTT,Dist) :- !,
    modusPonens(T,ImpTT,Dist).
modusPonens([ ],_ImpTT,_Dist).
```

(Passo 4) Basta verificar se as matrizes corretas são ou não uma repetição da matriz original. Para cada extensão unitária deve-se verificar se existe um homomorfismo entre ela e a matriz original (definição de repetição 5).

```
repetitionVerify(Function,Pairs,TVals,ImpTT1,NegTT1) :-
    imptt(ImpTT2),
    negtt(NegTT2),
    repetitionVerifyNeg(Function,TVals,NegTT1,NegTT2),
    repetitionVerifyImp(Function,Pairs,ImpTT1,ImpTT2),!.

repetitionVerifyImp(Function,[[A,B]|Ps],ImpTT1,ImpTT2) :-
    member([A,B,R1],ImpTT1),
    applyFunction(Function,R1,HR1),
    applyFunction(Function,A,HA),
    applyFunction(Function,B,HB),
    member([HA,HB,R2],ImpTT2),
    HR1 = R2, !,
    repetitionVerifyImp(Function,Ps,ImpTT1,ImpTT2).
repetitionVerifyImp(_,[],_,_).

repetitionVerifyNeg(Function,[A|Vs],NegTT1,NegTT2) :-
    member([A,R1],NegTT1),
    applyFunction(Function,R1,HR1),
    applyFunction(Function,A,HA),
    member([HA,R2],NegTT2),
    HR1 = R2, !,
    repetitionVerifyNeg(Function,Vs,NegTT1,NegTT2).
repetitionVerifyNeg(_,[],_,_).
```

```
applyFunction(Function,R,FR) :-
    member([R,FR],Function),!.
```

Como exemplo, a demonstração da completude da lógica Ł3.

3.2 Lógica Trivalente Ł3 — Łukasiewicz[15]

3.2.1 Sistema Axiomático Ł3

Conectivos: $\{ \rightarrow, \neg \}$

axioma 1 $(a \rightarrow (b \rightarrow a))$

axioma 2 $(a \rightarrow b) \rightarrow ((b \rightarrow c) \rightarrow (a \rightarrow c))$

axioma 3 $(\neg a \rightarrow \neg b) \rightarrow (b \rightarrow a)$

axioma 4 $((a \rightarrow \neg a) \rightarrow a) \rightarrow a$

Modus Pones $\frac{a \quad a \rightarrow b}{b}$

Matriz Candidata a Característica

	$\neg$
t	f
i	i
f	t

$\rightarrow$	t	i	f
t	t	i	f
i	t	t	i
f	t	t	t

3.2.2 Demonstração da Completude

(1) Determinação das Matrizes estendidas corretas e verificação de repetitividade

Considerando os dois conectivos: $\rightarrow$ (binário) e $\neg$ (unário), existem 65.536 ($4^8 = 65.536$) extensões da suas matrizes apresentadas. O primeiro passo

na demonstração da completude é determinar quais dessas extensões são extensões corretas em relação à axiomática acima. Posteriormente, verifica-se se cada uma dessas extensões corretas é uma repetição das matrizes clássicas. O resultado dessa verificação é listado a seguir (total de 42 matrizes corretas).

1. Valores Designados = {t}
   Valores Não-designados = {w, i, f}

	¬
t	f
i	i
f	t
w	i

→	t	i	f	w
t	t	i	f	i
i	t	t	i	t
f	t	t	t	t
w	t	t	i	t

É repetição. $h\colon \{t,i,f,w\} \to \{t,i,f\}$ onde
$\left\{ \; h(w) = i,\ h(t) = t,\ h(i) = i,\ h(f) = f \right.$

2. Valores Designados = {t}
   Valores Não-designados = {w, i, f}

	¬
t	f
i	i
f	t
w	w

→	t	i	f	w
t	t	i	f	i
i	t	t	i	t
f	t	t	t	t
w	t	t	i	t

É repetição. $h\colon \{t,i,f,w\} \to \{t,i,f\}$ onde
$\left\{ \; h(w) = i,\ h(t) = t,\ h(i) = i,\ h(f) = f \right.$

3. Valores Designados = {t}
   Valores Não-designados = {w, i, f}

	¬
t	f
i	i
f	t
w	i

→	t	i	f	w
t	t	i	f	i
i	t	t	i	t
f	t	t	t	t
w	t	t	w	t

É repetição. $h\colon \{t,i,f,w\} \to \{t,i,f\}$ onde
$\left\{ \; h(w) = i,\ h(t) = t,\ h(i) = i,\ h(f) = f \right.$

4. Valores Designados = {t}
   Valores Não-designados = {w, i, f}

	¬
t	f
i	i
f	t
w	w

→	t	i	f	w
t	t	i	f	i
i	t	t	i	t
f	t	t	t	t
w	t	t	w	t

É repetição. $h\colon \{t,i,f,w\} \to \{t,i,f\}$ onde
$\left\{ \; h(w) = i,\ h(t) = t,\ h(i) = i,\ h(f) = f \right.$

5. Valores Designados = {t}
   Valores Não-designados = {w, i, f}

	¬
t	f
i	i
f	t
w	t

→	t	i	f	w
t	t	i	f	f
i	t	t	i	i
f	t	t	t	t
w	t	t	t	t

É repetição. $h\colon \{t,i,f,w\} \to \{t,i,f\}$ onde
$\left\{ \; h(w) = f,\ h(t) = t,\ h(i) = i,\ h(f) = f \right.$

6. Valores Designados = {t}
   Valores Não-designados = {w, i, f}

	¬
t	f
i	i
f	t
w	i

→	t	i	f	w
t	t	i	f	w
i	t	t	i	t
f	t	t	t	t
w	t	t	i	t

É repetição. $h\colon \{t,i,f,w\} \to \{t,i,f\}$ onde
$\left\{ \; h(w) = i,\ h(t) = t,\ h(i) = i,\ h(f) = f \right.$

7. Valores Designados = {t}
   Valores Não-designados = {w, i, f}

	¬
t	f
i	i
f	t
w	w

→	t	i	f	w
t	t	i	f	w
i	t	t	i	t
f	t	t	t	t
w	t	t	i	t

É repetição. $h\colon \{t,i,f,w\} \to \{t,i,f\}$ onde
$\left\{ \; h(w) = i,\ h(t) = t,\ h(i) = i,\ h(f) = f \right.$

8. Valores Designados = {t}
   Valores Não-designados = {w, i, f}

	¬
t	f
i	i
f	t
w	i

→	t	i	f	w
t	t	i	f	w
i	t	t	i	t
f	t	t	t	t
w	t	t	w	t

É repetição. $h\colon \{t,i,f,w\} \to \{t,i,f\}$ onde
$\left\{ \; h(w) = i,\ h(t) = t,\ h(i) = i,\ h(f) = f \right.$

9. Valores Designados = {t}
   Valores Não-designados = {w, i, f}

	¬
t	f
i	i
f	t
w	w

→	t	i	f	w
t	t	i	f	w
i	t	t	i	t
f	t	t	t	t
w	t	t	w	t

É repetição. $h\colon \{t,i,f,w\} \to \{t,i,f\}$ onde
$\left\{ \; h(w) = i,\ h(t) = t,\ h(i) = i,\ h(f) = f \right.$

10. Valores Designados = {t}
    Valores Não-designados = {w, i, f}

	¬
t	f
i	i
f	t
w	t

→	t	i	f	w
t	t	i	f	w
i	t	t	i	i
f	t	t	t	t
w	t	t	t	t

É repetição. $h\colon \{t,i,f,w\} \to \{t,i,f\}$ onde
$\left\{ \; h(w) = f,\ h(t) = t,\ h(i) = i,\ h(f) = f \right.$

11. Valores Designados = {w, t}
    Valores Não-designados = {i, f}

	¬
t	f
i	i
f	t
w	f

→	t	i	f	w
t	t	i	f	t
i	t	t	i	t
f	t	t	t	t
w	t	i	f	t

É repetição. $h\colon \{t,i,f,w\} \to \{t,i,f\}$ onde
$\left\{ \; h(w) = t,\ h(t) = t,\ h(i) = i,\ h(f) = f \right.$

12. Valores Designados = {w, t}
    Valores Não-designados = {i, f}

	¬
t	f
i	i
f	t
w	f

→	t	i	f	w
t	t	i	f	t
i	t	t	i	t
f	t	t	t	t
w	t	i	f	w

É repetição. $h\colon \{t,i,f,w\} \to \{t,i,f\}$ onde
$\left\{ \; h(w) = t,\ h(t) = t,\ h(i) = i,\ h(f) = f \right.$

13. Valores Designados = {w, t}
Valores Não-designados = {i, f}

	¬
t	f
i	i
f	t
w	f

→	t	i	f	w
t	t	i	f	t
i	t	t	i	t
f	t	t	t	t
w	w	i	f	t

É repetição. h: $\{t, i, f, w\} \to \{t, i, f\}$ onde
$\left\{ \; h(w) = t,\ h(t) = t,\ h(i) = i,\ h(f) = f \right.$

14. Valores Designados = {w, t}
Valores Não-designados = {i, f}

	¬
t	f
i	i
f	t
w	f

→	t	i	f	w
t	t	i	f	t
i	t	t	i	t
f	t	t	t	t
w	w	i	f	w

É repetição. h: $\{t, i, f, w\} \to \{t, i, f\}$ onde
$\left\{ \; h(w) = t,\ h(t) = t,\ h(i) = i,\ h(f) = f \right.$

15. Valores Designados = {w, t}
Valores Não-designados = {i, f}

	¬
t	f
i	i
f	t
w	f

→	t	i	f	w
t	t	i	f	t
i	t	t	i	t
f	t	t	t	w
w	t	i	f	t

É repetição. h: $\{t, i, f, w\} \to \{t, i, f\}$ onde
$\left\{ \; h(w) = t,\ h(t) = t,\ h(i) = i,\ h(f) = f \right.$

16. Valores Designados = {w, t}
Valores Não-designados = {i, f}

	¬
t	f
i	i
f	t
w	f

→	t	i	f	w
t	t	i	f	t
i	t	t	i	t
f	t	t	t	w
w	t	i	f	w

É repetição. h: $\{t, i, f, w\} \to \{t, i, f\}$ onde
$\left\{ \; h(w) = t,\ h(t) = t,\ h(i) = i,\ h(f) = f \right.$

17. Valores Designados = {w, t}
Valores Não-designados = {i, f}

	¬
t	f
i	i
f	t
w	f

→	t	i	f	w
t	t	i	f	t
i	t	t	i	t
f	t	t	t	w
w	w	i	f	t

É repetição. h: $\{t, i, f, w\} \to \{t, i, f\}$ onde
$\left\{ \; h(w) = t,\ h(t) = t,\ h(i) = i,\ h(f) = f \right.$

18. Valores Designados = {w, t}
Valores Não-designados = {i, f}

	¬
t	f
i	i
f	t
w	f

→	t	i	f	w
t	t	i	f	t
i	t	t	i	t
f	t	t	t	w
w	w	i	f	w

É repetição. h: $\{t, i, f, w\} \to \{t, i, f\}$ onde
$\left\{ \; h(w) = t,\ h(t) = t,\ h(i) = i,\ h(f) = f \right.$

19. Valores Designados = {w, t}
Valores Não-designados = {i, f}

	¬
t	f
i	i
f	t
w	f

→	t	i	f	w
t	t	i	f	t
i	t	t	i	w
f	t	t	t	t
w	t	i	f	t

É repetição. h: $\{t, i, f, w\} \to \{t, i, f\}$ onde
$\left\{ \; h(w) = t,\ h(t) = t,\ h(i) = i,\ h(f) = f \right.$

20. Valores Designados = {w, t}
Valores Não-designados = {i, f}

	¬
t	f
i	i
f	t
w	f

→	t	i	f	w
t	t	i	f	t
i	t	t	i	w
f	t	t	t	t
w	t	i	f	w

É repetição. h: $\{t, i, f, w\} \to \{t, i, f\}$ onde
$\left\{ \; h(w) = t,\ h(t) = t,\ h(i) = i,\ h(f) = f \right.$

21. Valores Designados = {w, t}
Valores Não-designados = {i, f}

	¬
t	f
i	i
f	t
w	f

→	t	i	f	w
t	t	i	f	t
i	t	t	i	w
f	t	t	t	t
w	w	i	f	t

É repetição. h: $\{t, i, f, w\} \to \{t, i, f\}$ onde
$\left\{ \; h(w) = t,\ h(t) = t,\ h(i) = i,\ h(f) = f \right.$

22. Valores Designados = {w, t}
Valores Não-designados = {i, f}

	¬
t	f
i	i
f	t
w	f

→	t	i	f	w
t	t	i	f	t
i	t	t	i	w
f	t	t	t	t
w	w	i	f	w

É repetição. h: $\{t, i, f, w\} \to \{t, i, f\}$ onde
$\left\{ \; h(w) = t,\ h(t) = t,\ h(i) = i,\ h(f) = f \right.$

23. Valores Designados = {w, t}
Valores Não-designados = {i, f}

	¬
t	f
i	i
f	t
w	f

→	t	i	f	w
t	t	i	f	t
i	t	t	i	w
f	t	t	t	w
w	t	i	f	t

É repetição. h: $\{t, i, f, w\} \to \{t, i, f\}$ onde
$\left\{ \; h(w) = t,\ h(t) = t,\ h(i) = i,\ h(f) = f \right.$

24. Valores Designados = {w, t}
Valores Não-designados = {i, f}

	¬
t	f
i	i
f	t
w	f

→	t	i	f	w
t	t	i	f	t
i	t	t	i	w
f	t	t	t	w
w	t	i	f	w

É repetição. h: $\{t, i, f, w\} \to \{t, i, f\}$ onde
$\left\{ \; h(w) = t,\ h(t) = t,\ h(i) = i,\ h(f) = f \right.$

25. Valores Designados = {w, t}
Valores Não-designados = {i, f}

	¬
t	f
i	i
f	t
w	f

→	t	i	f	w
t	t	i	f	t
i	t	t	i	w
f	t	t	t	w
w	w	i	f	t

É repetição. h: $\{t, i, f, w\} \to \{t, i, f\}$ onde
$\left\{ \; h(w) = t,\ h(t) = t,\ h(i) = i,\ h(f) = f \right.$

26. Valores Designados = {w, t}
Valores Não-designados = {i, f}

	¬
t	f
i	i
f	t
w	f

→	t	i	f	w
t	t	i	f	t
i	t	t	i	w
f	t	t	t	w
w	w	i	f	w

É repetição. h: $\{t, i, f, w\} \to \{t, i, f\}$ onde
$\left\{ \; h(w) = t,\ h(t) = t,\ h(i) = i,\ h(f) = f \right.$

27. Valores Designados = {w, t}
Valores Não-designados = {i, f}

	¬
t	f
i	i
f	t
w	f

→	t	i	f	w
t	t	i	f	w
i	t	t	i	t
f	t	t	t	t
w	t	i	f	t

É repetição. h: $\{t, i, f, w\} \to \{t, i, f\}$ onde
$\left\{ \; h(w) = t,\ h(t) = t,\ h(i) = i,\ h(f) = f \right.$

28. Valores Designados = {w, t}
Valores Não-designados = {i, f}

	¬
t	f
i	i
f	t
w	f

→	t	i	f	w
t	t	i	f	w
i	t	t	i	t
f	t	t	t	t
w	t	i	f	w

É repetição. h: $\{t, i, f, w\} \to \{t, i, f\}$ onde
$\left\{ \; h(w) = t,\ h(t) = t,\ h(i) = i,\ h(f) = f \right.$

29. Valores Designados = {w, t}
Valores Não-designados = {i, f}

	¬
t	f
i	i
f	t
w	f

→	t	i	f	w
t	t	i	f	w
i	t	t	i	t
f	t	t	t	t
w	w	i	f	t

É repetição. h: $\{t, i, f, w\} \to \{t, i, f\}$ onde
$\left\{ \; h(w) = t,\ h(t) = t,\ h(i) = i,\ h(f) = f \right.$

30. Valores Designados = {w, t}
Valores Não-designados = {i, f}

	¬
t	f
i	i
f	t
w	f

→	t	i	f	w
t	t	i	f	w
i	t	t	i	t
f	t	t	t	t
w	w	i	f	w

É repetição. h: $\{t, i, f, w\} \to \{t, i, f\}$ onde
$\left\{ \; h(w) = t,\ h(t) = t,\ h(i) = i,\ h(f) = f \right.$

31. Valores Designados = {w, t}
Valores Não-designados = {i, f}

	¬
t	f
i	i
f	t
w	f

→	t	i	f	w
t	t	i	f	w
i	t	t	i	t
f	t	t	t	w
w	t	i	f	t

É repetição. h: $\{t, i, f, w\} \to \{t, i, f\}$ onde
$\left\{ \; h(w) = t,\ h(t) = t,\ h(i) = i,\ h(f) = f \right.$

32. Valores Designados = {w, t}
Valores Não-designados = {i, f}

	¬
t	f
i	i
f	t
w	f

→	t	i	f	w
t	t	i	f	w
i	t	t	i	t
f	t	t	t	w
w	t	i	f	w

É repetição. h: $\{t, i, f, w\} \to \{t, i, f\}$ onde
$\left\{ \; h(w) = t,\ h(t) = t,\ h(i) = i,\ h(f) = f \right.$

33. Valores Designados = {w, t}
Valores Não-designados = {i, f}

	¬
t	f
i	i
f	t
w	f

→	t	i	f	w
t	t	i	f	w
i	t	t	i	t
f	t	t	t	w
w	w	i	f	t

É repetição. h: $\{t, i, f, w\} \to \{t, i, f\}$ onde
$\left\{ \; h(w) = t,\ h(t) = t,\ h(i) = i,\ h(f) = f \right.$

34. Valores Designados = {w, t}
Valores Não-designados = {i, f}

	¬
t	f
i	i
f	t
w	f

→	t	i	f	w
t	t	i	f	w
i	t	t	i	t
f	t	t	t	w
w	w	i	f	w

É repetição. h: $\{t, i, f, w\} \to \{t, i, f\}$ onde
$\left\{ \; h(w) = t,\ h(t) = t,\ h(i) = i,\ h(f) = f \right.$

35. Valores Designados = {w, t}
Valores Não-designados = {i, f}

	¬
t	f
i	i
f	t
w	f

→	t	i	f	w
t	t	i	f	w
i	t	t	i	w
f	t	t	t	t
w	t	i	f	t

É repetição. h: $\{t, i, f, w\} \to \{t, i, f\}$ onde
$\left\{ \; h(w) = t,\ h(t) = t,\ h(i) = i,\ h(f) = f \right.$

36. Valores Designados = {w, t}
Valores Não-designados = {i, f}

	¬
t	f
i	i
f	t
w	f

→	t	i	f	w
t	t	i	f	w
i	t	t	i	w
f	t	t	t	t
w	t	i	f	w

É repetição. h: $\{t, i, f, w\} \to \{t, i, f\}$ onde
$\left\{ \; h(w) = t,\ h(t) = t,\ h(i) = i,\ h(f) = f \right.$

37. Valores Designados = {w, t}
Valores Não-designados = {i, f}

	¬
t	f
i	i
f	t
w	f

→	t	i	f	w
t	t	i	f	w
i	t	t	i	w
f	t	t	t	t
w	w	i	f	t

É repetição. $h\colon \{t,i,f,w\} \to \{t,i,f\}$ onde
$\left\{ \; h(w)=t,\ h(t)=t,\ h(i)=i,\ h(f)=f \right.$

38. Valores Designados = {w, t}
Valores Não-designados = {i, f}

	¬
t	f
i	i
f	t
w	f

→	t	i	f	w
t	t	i	f	w
i	t	t	i	w
f	t	t	t	t
w	w	i	f	w

É repetição. $h\colon \{t,i,f,w\} \to \{t,i,f\}$ onde
$\left\{ \; h(w)=t,\ h(t)=t,\ h(i)=i,\ h(f)=f \right.$

39. Valores Designados = {w, t}
Valores Não-designados = {i, f}

	¬
t	f
i	i
f	t
w	f

→	t	i	f	w
t	t	i	f	w
i	t	t	i	w
f	t	t	t	w
w	t	i	f	t

É repetição. $h\colon \{t,i,f,w\} \to \{t,i,f\}$ onde
$\left\{ \; h(w)=t,\ h(t)=t,\ h(i)=i,\ h(f)=f \right.$

40. Valores Designados = {w, t}
Valores Não-designados = {i, f}

	¬
t	f
i	i
f	t
w	f

→	t	i	f	w
t	t	i	f	w
i	t	t	i	w
f	t	t	t	w
w	t	i	f	w

É repetição. $h\colon \{t,i,f,w\} \to \{t,i,f\}$ onde
$\left\{ \; h(w)=t,\ h(t)=t,\ h(i)=i,\ h(f)=f \right.$

41. Valores Designados = {w, t}
Valores Não-designados = {i, f}

	¬
t	f
i	i
f	t
w	f

→	t	i	f	w
t	t	i	f	w
i	t	t	i	w
f	t	t	t	w
w	w	i	f	t

É repetição. $h\colon \{t,i,f,w\} \to \{t,i,f\}$ onde
$\left\{ \; h(w)=t,\ h(t)=t,\ h(i)=i,\ h(f)=f \right.$

42. Valores Designados = {w, t}
Valores Não-designados = {i, f}

	¬
t	f
i	i
f	t
w	f

→	t	i	f	w
t	t	i	f	w
i	t	t	i	w
f	t	t	t	w
w	w	i	f	w

É repetição. $h\colon \{t,i,f,w\} \to \{t,i,f\}$ onde
$\left\{ \; h(w)=t,\ h(t)=t,\ h(i)=i,\ h(f)=f \right.$

(2) Conclusão

Foram examinadas 65.536 matrizes tetravalentes, dessas apenas 42 são extensões corretas tetravalentes da matriz candidata. Foi verificado que todas elas são repetições da matriz candidata. Pelo teorema 5, conclui-se que a axiomática de Łukasiewicz é completa. Ou seja, a matriz

	$\neg$
t	f
i	i
f	t

$\rightarrow$	t	i	f
t	t	i	f
i	t	t	i
f	t	t	t

é característica do sistema axiomático Ł3 definido por:

axioma 1 $(a \rightarrow (b \rightarrow a))$

axioma 2 $(a \rightarrow b) \rightarrow ((b \rightarrow c) \rightarrow (a \rightarrow c))$

axioma 3 $(\neg a \rightarrow \neg b) \rightarrow (b \rightarrow a)$

axioma 4 $((a \rightarrow \neg a) \rightarrow a) \rightarrow a$

Modus Pones $\dfrac{a \quad a \rightarrow b}{b}$

Conclusão

O resultado apresentado no capítulo 4 demonstra a praticidade da utilização do método algorítmico proposto nesse trabalho para demonstrar a completude de lógicas finitas multivalentes.

O método tem como característica ser uniforme para os sistemas lógicos multivalentes, assim um programa para verificar a completude de uma lógica pode ser adaptado para uma outra lógica. No apêndice, os vários programas que fazem a demonstração da completude das lógicas proposicionais clássicas definidas por Kleene, Mendelson, Hilbert e Ackerman, e, as lógicas trivalentes de Łukasiewicz e Setter apresentam uma estrutura comum, o que ilustra e confirma essa característica de uniformidade. Os programas foram desenvolvidos em Prolog ISO.

A abordagem desenvolvida nesse trabalho é inovadora, não há na literatura nenhum trabalho similar. São dois os teoremas fundamentais que possibilitaram o método algorítmico de demonstração da completude:

1. Seja $\mathcal{L}$ uma lógica com assinatura Σ, com matriz característica finita. M_1 é uma matriz característica para $\mathcal{L}$ se, e somente se, $\mathcal{L}$ é correta para M_1, e para toda extensão correta M_2 de M_1, $\mathrm{Taut}(M_1) = \mathrm{Taut}(M_2)$.

2. Seja $\mathcal{L} = (\Sigma, A, R)$ uma lógica multivalente finita com assinatura Σ. Se M_n uma matriz correta para $\mathcal{L}$, onde n é o número de valores-verdade

de M_n, e todas as extenões unitárias corretas de M_n são repetições de M_n, então M_n é matriz característica de $\mathcal{L}$.

O primeiro determina que para demonstrar se uma matriz é característica de um sistema lógico finito multivalente deve-se verificar de forma exaustiva todas as sucessivas extensões corretas dessa matriz, mostrando que todas elas apresentam o mesmo conjunto de tautologias. Todavia, O que foi estabelecido pelo teorema 4 não é um resultado computacionalmente efetivo, pois não há como gerar e testar todas as extensões corretas de uma dada matriz. Trata-se de uma caracterização do problema de demonstrar a completude para uma lógica finita multivalente qualquer.

O segundo, possibilita a criação de um procedimento computacional para a solução do problema da completude. Assim, para determinar se uma matriz correta é característica basta buscar dentre todas as suas extensões aquelas que também são corretas. Posteriormente, fazer uma verificação de que todas essas extensões corretas são repetições da matriz candidata, o que é significa determinar que todas essas matrizes apresentam o mesmo conjunto de tautologias. Isso se dá através da verificação de um homomorfismo "especial" entre as matrizes consideradas.

Uma outra possibilidade do uso do método algorítmico proposto é determinar a semântica matricial para sublógicas proposicionais. Assim, dado um sistema lógico $\mathcal{L} = (\Sigma, A, R)$, onde A é um conjunto de axiomas e R é um conjunto de regras de inferência, podemos construir um procedimento exaustivo para a determinação de uma possível semântica multivalente finita. Para tal, inicia-se a busca por matrizes com dois valores-verdade, um designado, o outro não-designado. Determina-se dentre essas matrizes quais são corretas e então aplica-se o procedimento de verificação de completude. Caso não se verifique o segundo teorema, então passa-se à verificação de matrizes com três

valores-verdade e assim sucessivamente. Essa possibilidade, juntamente com o desenvolvimento de um ambiente de *software* iterativo e amigável para a aplicação do métdodo algorítmico proposto serão alvo de próximos trabalhos.

Apêndice A

Implementação do Método

Seguem os códigos implementados em **Prolog** para a demonstração da completude da lógica proposicional multivalente Ł3 de Łukasiewicz.

```
:- dynamic      triples_set/1, pairs_set/1,
                count/1, r/4,
                h/3.

truthValues([t,i,f,w],[t],[i,f],w).

triplesSetDet(TVals,TriplesList) :-
    findall([X,Y,Z],
            ( member(X,TVals),
              member(Y,TVals),
              member(Z,TVals) ),
              TriplesList),
    retractall(triples_set(_)),
    assert(triples_set(TriplesList)).
```

```
pairsSetDet(TVals,PairsList) :-
    findall([X,Y],
           ( member(X,TVals),
             member(Y,TVals) ),
             PairsList),
    retractall(pairs_set(_)),
    assert(pairs_set(PairsList)).


main :-
    initialize,
    truthValues(TVals,Dist,NDist,NewVal),
    triplesSetDet(TVals,Triples),
    pairsSetDet(TVals,Pairs),
    extendedMatricesDet(TVals,Dist,[NewVal|NDist],Triples,Pairs),
    extendedMatricesDet(TVals,[NewVal|Dist],NDist,Triples,Pairs),
    printResultsLatexFormat,
    !.

initialize :-
    retractall(r(_,_,_,_)),
    retractall(h(_,_,_)),
    !.
```

```
extendedMatricesDet(TVals,Dist,NDist,Triples,Pairs) :-
    member(A,TVals),
    member(B,TVals),
    member(C,TVals),
    member(D,TVals),
    member(E,TVals),
    member(F,TVals),
    member(G,TVals),
    member(H,TVals),
    truthTableVerification(A,B,C,D,E,F,G,H,Dist,NDist,
                           Triples,Pairs,TVals),
    fail.
extendedMatricesDet(_,_,_,_,_).


truthTableVerification(A,B,C,D,E,F,G,H,Dist,NDist,
                       Triples,Pairs,TVals) :-
    extendImp(A,B,C,D,E,F,G,ImpTT),
    extendNeg(H,NegTT),
    axiom1(Pairs,ImpTT,Dist),
    axiom2(Triples,ImpTT,Dist),
    axiom3(Pairs,ImpTT,NegTT,Dist),
    axiom4(TVals,ImpTT,NegTT,Dist),
    modusPonens(Pairs,ImpTT,Dist),
    !,
    saveMatrices(ImpTT,NegTT,Dist,NDist),
    !.
```

```
saveMatrices(ImpTT,NegTT,Dist,NDist) :-
    assert(r(ImpTT,NegTT,Dist,NDist)),
    !.


saveRepetitions(ImpTT,NegTT,Function) :-
    assert(h(ImpTT,NegTT,Function)),
    !.


imptt([[t,t,t],[t,i,i],[t,f,f],
       [i,t,t],[i,i,t],[i,f,i],
       [f,t,t],[f,i,t],[f,f,t]]).
negtt([[t,f],[i,i],[f,t]]).


extendImp(A,B,C,D,E,F,G,
          [[t,t,t],[t,i,i],[t,f,f],[t,w,A],
           [i,t,t],[i,i,t],[i,f,i],[i,w,B],
           [f,t,t],[f,i,t],[f,f,t],[f,w,C],
           [w,t,D],[w,i,E],[w,f,F],[w,w,G]]).

extendNeg(H,[[t,f],[i,i],[f,t],[w,H]]).
```

```
/* axiom 1 a => (b => a) */
axiom1([[X,Y]|T],ImpTT,Dist) :-
    member([Y,X,A],ImpTT),
    member([X,A,Result],ImpTT),
    member(Result,Dist),
    !,
    axiom1(T,ImpTT,Dist).
axiom1([],_,_) :- !.
```

```
/* axiom 2 (a => b) => ((b => c) => (a => c)) */
axiom2([[A,B,C]|T],ImpTT,Dist) :-
    member([A,B,I1],ImpTT),
    member([B,C,I2],ImpTT),
    member([A,C,I3],ImpTT),
    member([I2,I3,I4],ImpTT),
    member([I1,I4,Result],ImpTT),
    member(Result,Dist),
    !,
    axiom2(T,ImpTT,Dist).
axiom2([],_,_) :- !.
```

```
/* axioma 3  (~a => ~b) => (b => a) */
axiom3([[A,B]|T],ImpTT,NegTT,Dist) :-
    member([A,NA],NegTT),
    member([B,NB],NegTT),
    member([NA,NB,I1],ImpTT),
    member([B,A,I2],ImpTT),
    member([I1,I2,Result],ImpTT),
    member(Result,Dist),
    !,
    axiom3(T,ImpTT,NegTT,Dist).
axiom3([],_,_,_) :- !.



/* axioma 4  ((p => ~p) => p) => p */
axiom4([A|T],ImpTT,NegTT,Dist) :-
    member([A,NA],NegTT),
    member([A,NA,I1],ImpTT),
    member([I1,A,I2],ImpTT),
    member([I2,A,Result],ImpTT),
    member(Result,Dist),
    !,
    axiom4(T,ImpTT,NegTT,Dist).
axiom4([],_,_,_) :- !.
```

```
modusPonens([[A,B]|T],ImpTT,Dist) :-
    member(A,Dist),
    member([A,B,R],ImpTT),
    member(R,Dist),
    !,
    member(B,Dist),
    modusPonens(T,ImpTT,Dist),
    !.
modusPonens([_|T],ImpTT,Dist) :-
    !,
    modusPonens([_|T],ImpTT,Dist),
    !.
modusPonens([ ],_ImpTT,_Dist) :- !.

printResultsLatexFormat :-
    tell('c:\\tese2010\\progtese\\luka4.tex'),
    write('\\section*{Lógica Proposicional - {\\L}ukasiewicz
          - {\\L}3}'),nl,nl,
    write('\\noindent Conectivos: \\{ $\\rightarrow$,
          $\\neg$ \\}'),nl,nl,
    write('\\begin{description}'),nl,
    write('\\item[axioma 1] $(a \\rightarrow (b
          \\rightarrow a))$'),nl,
    write('\\item[axioma 2] $(a \\rightarrow b)
           \\rightarrow ((b
           \\rightarrow c) \\rightarrow (a
           \\rightarrow c))$'),nl,
```

```
    write('\\item[axioma 3] $(\\neg a \\rightarrow \\neg b)
           \\rightarrow (b \\rightarrow a)$'),nl,
    write('\\item[axioma 4] $((a \\rightarrow \\neg a)
           \\rightarrow a) \\rightarrow a$'),nl,
    write('\\item[{\\em Modus Pones}] {\\Large $\\frac{a \\ \\ a
           \\rightarrow b}{b}$}'),nl,
    write('\\end{description}'),nl,nl,
    write('\\noindent {\\bf Determinação das Matrizes
          extendidas corretas e verificação de repetitividade}'),
    nl,nl,
    write('\\begin{enumerate}'),nl,
    printLatexResults,nl,
    write('\\end{enumerate}'),nl,
    told.

printLatexResults :-
    r(ImpTT,NegTT,Dist,NDist),
    printLatexMain(ImpTT,NegTT,Dist,NDist),
    fail.
printLatexResults :- !.

printLatexMain(ImpTT,NegTT,Dist,NDist) :-
    truthValues(TVals,D,ND,AdVal),
    append(D,ND,TV0),
    write('\\item Valores Distinguidos = \\{'),write_list(Dist),
    write('\\} \\hspace{0.4cm} '),
    write('Valores Não-Distinguidos = \\{'),write_list(NDist),
```

```
    write('\\}'),nl,nl,
    printLatexTruthTable_1('$\\neg$',NegTT),nl,
    printLatexTruthTable_2('$\\rightarrow$',TVals,ImpTT),nl,
    write('\\end{center}'),nl,
    repetitionLatex(ImpTT,NegTT,TVals,TV0,AdVal),
    !.

repetitionLatex(ImpTT,NegTT,TVals,TV,AdVal) :-
    function(TVals,AdVal,Function),
    pairs_set(Pairs),
    repetitionVerify(Function,Pairs,TVals,ImpTT,NegTT),
    printLatexRepetion(Function,TVals,TV),
    nl,nl,
    !.
repetitionLatex(_,_,_,_) :-
    write('{\\bf Não há repetição!}'),
    nl,!.

function(TVals,AdVal,[[AdVal,Val]|Identidy]) :-
    delete(TVals,AdVal,TV),
    member(Val,TV),
    identidyList(TV,Identidy).

identidyList([A|As],[[A,A]|R]):-
    identidyList(As,R),
    !.
identidyList([],[]):- !.
```

```
repetitionVerify(Function,Pairs,TVals,ImpTT1,NegTT1) :-
    imptt(ImpTT2),
    negtt(NegTT2),
    repetitionVerifyNeg(Function,TVals,NegTT1,NegTT2),
    repetitionVerifyImp(Function,Pairs,ImpTT1,ImpTT2),
    !.


repetitionVerifyImp(Function,[[A,B]|Ps],ImpTT1,ImpTT2) :-
    member([A,B,R1],ImpTT1),
    applyFunction(Function,R1,HR1),
    applyFunction(Function,A,HA),
    applyFunction(Function,B,HB),
    member([HA,HB,R2],ImpTT2),
    HR1 = R2,
    !,
    repetitionVerifyImp(Function,Ps,ImpTT1,ImpTT2),
    !.
repetitionVerifyImp(_,[],_,_) :- !.


repetitionVerifyNeg(Function,[A|Vs],NegTT1,NegTT2) :-
    member([A,R1],NegTT1),
    applyFunction(Function,R1,HR1),
    applyFunction(Function,A,HA),
```

```
    member([HA,R2],NegTT2),
    HR1 = R2,
    !,
    repetitionVerifyNeg(Function,Vs,NegTT1,NegTT2),
    !.
repetitionVerifyNeg(_,[],_,_) :- !.


applyFunction(Function,R,FR) :-
    member([R,FR],Function),!.


printLatexTruthTable_1(Simb,[[X,Y]|T]) :-
    write('\\begin{center}'),nl,
    write('\\begin{tabular}{c|c}'),nl,
    write('     &  '),write(Simb),write('  \\\\ \\hline'),nl,
    write(X),write(' & '),write(Y),write(' \\\\'),nl,
    printLatexAux1(T).

printLatexAux1([[X,Y]|T]) :-
    write(X),write(' & '),write(Y),write(' \\\\'),nl,
    printLatexAux1(T).
printLatexAux1([]) :-
    write('\\end{tabular}'),write('\\hspace{1.0cm}'),
    !.
```

```
printLatexTruthTable_2(Simb,TVals,TTable) :-
    TVals = [A,B,C,D],
    write('\\begin{tabular}{c|cccc}'),nl,
    write(Simb), write(' & '), write(A), write(' & '),
    write(B),
    write(' & '), write(C), write(' & '), write(D),
    write(' \\\\ \\hline'),nl,
    printLatexAux2(TVals,TTable).

printLatexAux2([A|As],[[_,_,Z1],[_,_,Z2],
                [_,_,Z3],[_,_,Z4]|T]) :-
    write(A),write(' & '),write(Z1),write(' & '),write(Z2),
    write(' & '),write(Z3),write(' & '),write(Z4),
    write(' \\\\'),nl,
    printLatexAux2(As,T).
printLatexAux2([],[]) :-
    write('\\end{tabular}'),write('\\hspace{1.0cm}'),!.

printLatexRepetion([[X1,Y1],[X2,Y2],
                    [X3,Y3],[X4,Y4]],TVals,TV) :-
    write('\\noindent É repetição. \\\\'),nl,
    write('$h\\colon \\{'),write_list(TVals),
    write('\\} \\to \\{'),
    write_list(TV),write('\\}$:'),nl,
    write('$'),nl,
    write('\\left\\{ '),nl,
    write('\\begin{array}{l}'),nl,
```

```
    write('h('),write(X1),write(') = '),
    write(Y1),write(', \\ '),nl,
    write('h('),write(X2),write(') = '),
    write(Y2),write(', \\ '),nl,
    write('h('),write(X3),write(') = '),
    write(Y3),write(', \\ '),nl,
    write('h('),write(X4),write(') = '),
    write(Y4),nl,
    write('\\end{array}'),nl,
    write('\\right.'),nl,
    write('$'),nl,nl,
    !.

write_list([A]):-
    write(A),
    !.
write_list([A|As]) :-
    write(A),write(', '),
    write_list(As).
```

Referências Bibliográficas

[1] Anshakov, O. e Rychkov, S. *On Finite-Valued Propositional Logical Calculi,* Notre Dame Journal of Formal Logic, vol. 36, number 4, 1994.

[2] Borkowski, L. (Ed.) *Selected works of J. L ukasiewicz,* Amsterdam: North-Holland, 1970.

[3] Carnielli, W. A. and Sette, A. M. Maximal weakly-intuitionistic logics, Studia Logica, 55, p. 181-203, 1995.

[4] da Costa, N. C. A. *O ambiente matemático no século XIX e a lógica do século XX.* In : Évora, F. (Org.). *Século XIX: o nascimento da ciência contemporânea.* Coleção CLE, v.11, p. 59-65, 1992.

[5] Dugundji, J. *Note on a property of matrices for lewis and langford's calculi of propositions,* Journal of Symbolic Logic Vol.5, N. 4, 1940.

[6] Euclides. *Os elementos.* Editora UNESP, 2009.

[7] Frege, G. *Begriffsschrift: eine der arithmetischen nach gebildete Formelsprache des reinen Denkens.* Hildesheim: Georg Olms Verlag. (Reimpressão da edição de 1879), 1977. Tradução para o inglês: Bauer-Mengelberg S., *Concept Script, a formal language of pure thought modelled upon that of arithmetic*, Jean Van Heijenoort, ed., 1967.

[8] Gottwald, S. *A Treatise on Many-Valued Logics*, Baldock, Research Studies Press, Hertfordshire, England, 2001.

[9] Haack, S. *Deviant logic*, Cambridge: Cambridge University Press, 1974.

[10] D'OTTAVIANO, Ítala Maria Loffredo e FEITOSA, H. A. *Sobre a história da lógica, a lógica clássica e o surgimento das lógicas não-clássicas*, Página Educacional do CLE, Campinas, p. 1-34, 2003.

[11] Kalicki, J. *Note on truth-tables*, Journal of Symbolic Logic 15, 1950.

[12] Kalicki, J. *A test for the existence of tautologies according to many-valued truth-tables*, Journal of Symbolic Logic 15, 1950.

[13] Kalicki, J. *A test for the equality of truth-tables*, Journal of Symbolic Logic 17, 1952.

[14] Łos, J. *O matrycach logicznych*, Prace Wrocławskiego, Towarzystwar Naukowego, 10, 1949. Tradução em inglês presente na tese de doutorado *Matrices for sentencial calculi* de Dolph Edward Ulrich, Wayne State University, 1967.

[15] Łukasiewicz, J. *O logice trójwartościowez*. Ruch Filosoficzny, 5, 170-171, 1920. Tradução em [2] e [22].

[16] Malinowski, G. *Handbook Of The History Of Logic*, Chapter 1. Editado por Dov M. Gabbay e John Woods. Elsevier Science Ltd, 2009.

[17] McColl, S. (Ed.) *Polish logic 1920-1939*, Oxford: Clarendon Press, 1967.

[18] Mendelson, Elliott. *Introduction to Mathematical Logic*, Wadsworth & Brooks/Cole Advanced Books & Software. Monterey, California, 1964.

[19] Post, E.L. *Introduction to a general theory of elementary propositions*, Bulletin of the American Mathematical Society, 26, 437, 1920.

[20] Rosser, J.B. e Turquette, A. R. *Many-valued logics*, North-Holland, Amsterdam, 1952.

[21] Sette, A. M. *On the propositional calculus* P_1, Matematica Niponicæ, vol. 16, pp. 173–180, 1973.

[22] Tarski, A. *Logic, semantics, metamathematics*, Oxford: Clarendon, 1956.

[23] Ulrich, D. E. *Matrices for sentencial calculi*, PhD. Thesis, Wayne State University, 1967.

[24] Wajsberg, M. *Aksjomatyzacja trówartosciowego rachunku zdan* [Axiomatização do cálculo proposicional trivalente]. Comptes Rendus dês séances de la Societé dês Sciences et dês Lettres de Varsovie, Classe 3, v. 24, pp. 126-148, 1931. Tradução para o inglês por S. McColl [17].

[25] Whitehead, A. N. e Russell, B. *Principia Mathematica*, Cambridge: Cambridge University Press, 1973.

Livros Grátis

(http://www.livrosgratis.com.br)

Milhares de Livros para Download:

Baixar livros de Literatura
Baixar livros de Literatura de Cordel
Baixar livros de Literatura Infantil
Baixar livros de Matemática
Baixar livros de Medicina
Baixar livros de Medicina Veterinária
Baixar livros de Meio Ambiente
Baixar livros de Meteorologia
Baixar Monografias e TCC
Baixar livros Multidisciplinar
Baixar livros de Música
Baixar livros de Psicologia
Baixar livros de Química
Baixar livros de Saúde Coletiva
Baixar livros de Serviço Social
Baixar livros de Sociologia
Baixar livros de Teologia
Baixar livros de Trabalho
Baixar livros de Turismo