UNIVERSIDADE FEDERAL DE MINAS GERAIS
FACULDADE DE FILOSOFIA E CIÊNCIAS HUMANAS
DEPARTAMENTO DE FILOSOFIA

# LÓGICA II

Prof. Paulo Roberto Margutti Pinto

2006

SISTEMAS FORMAIS CLÁSSICOS E NÃO-CLÁSSICOS

SEMÂNTICA FORMAL

METALÓGICA

**Sumário**

## 1. Introdução

O presente texto constitui uma versão preliminar de uma apresentação futuramente mais detalhada e elaborada dos conteúdos da disciplina Lógica II do Curso de Graduação em Filosofia do Departamento de Filosofia da Fafich da UFMG. Estes conteúdos envolvem uma apresentação dos sistemas formais clássicos e não-clássicos, bem como dos principais conceitos e resultados metalógicos. Tendo em vista a vastidão do assunto e as dificuldades de uma exposição rigorosa, o que se tentou aqui foi uma apresentação resumida dos principais aspectos ligados a alguns dos sistemas formais clássicos, como o sistema lógico dos *Principia Mathematica*, e a alguns dos sistemas formais não-clássicos, como a lógica modal, a lógica trivalente, a lógica imprecisa e a lógica paraconsistente. Estratégia semelhante foi adotada no caso da metalógica e dos problemas a ela relacionados. O texto inclui, na parte final, uma bibliografia para os interessados em aprofundar esses estudos. Nem sempre o texto terá a extensão e o detalhamento desejáveis, mas pelo menos se poderá contar com uma exposição em português dos principais assuntos ligados ao programa de Lógica II.

## 2. Sistemas Formais

### 2.1. Caracterização geral

Charles Morris, em 1938, propôs a constituição de uma teoria geral dos signos, à qual ele denominou *semiótica* e que se divide em três partes fundamentais:

a) *sintaxe*, que estuda as relações entre signos, independentemente dos objetos por eles designados. Exemplos de conceitos considerados pela *sintaxe* são o de *axioma*, *regra de inferência*, *fórmula bem formada*, *teorema*, etc.

b) *semântica*, que estuda as relações entre os signos e os objetos por eles designados. Exemplos de conceitos considerados pela *semântica* são o de *sentido*, *significado*, *verdade*, *designação*, *referência*, *nomear*, *condições-verdade*, etc.

c) *pragmática*, que estuda as relações entre os signos, os objetos por eles designados e os falantes. Exemplos de conceitos considerados pela *pragmática* são o de *uso*, *contexto*, *termo indexical*, etc. Bar Hillel, em 1954, propôs uma definição de *pragmática* como a disciplina que trata de *expressões indexicais* e Montague, em 1972, propôs uma outra definição desta disciplina como tratando de *sentenças indexicais*.

A divisão acima é importante para nós porque o estudo dos sistemas formais envolve a construção de linguagens formalizadas tanto em nível sintático como em nível semântico. Ainda não foi possível construir linguagens formalizadas satisfatórias em nível pragmático.

Do ponto de vista sintático, um sistema formal é uma linguagem formalizada não-interpretada, ou seja, uma linguagem em que importam apenas as relações entre os signos que a constituem. Um sistema formal se constitui dos seguintes elementos:

1) signos primitivos: são os signos utilizados na construção das expressões bem formadas do sistema. Eles correspondem ao vocabulário básico da linguagem formal a que pertencem e não são definidos.

2) regras de formação: estabelecem quais as combinações de signos primitivos são bem formadas e quais não o são, separando as expressões consideradas corretas das incorretas na linguagem formal. Uma combinação correta de signos é uma *fórmula bem formada* e é representada abreviadamente por *fbf*.

3) proposições primitivas: são fbfs consideradas válidas por definição, constituindo assim o ponto de partida para as deduções a serem feitas no sistema formal.

4) regras de dedução: estabelecem a forma pela qual é possível deduzir uma fbf de outra. Aplicando-as às proposições primitivas, podemos demonstrar teoremas do sistema formal; aplicando posteriormente as mesmas regras às combinações possíveis de proposições primitivas e teoremas, podemos demonstrar mais teoremas. Tendo em vista que expressam como podemos deduzir uma sentença do sistema a partir de outras sentenças do sistema, as regras de dedução não são expressas no interior do sistema, mas sim numa meta-linguagem que fala sobre o sistema.

Alguns dos principais aspectos de um sistema formal são os seguintes:[1]

1) apresentação: depende da escolha dos símbolos através dos quais o sistema será exibido. Por exemplo, no cálculo sentencial, a conectiva *se ..., então ...* pode ser

formalizada com o auxílio de $\supset$ ou de =>. Duas sentenças quaisquer podem ser formalizadas com o auxílio de $p$ e $q$ ou de $A$ e $B$. Neste caso, a implicação de duas sentenças poderia ser formalizada como $p \supset q$ ou como $A \supset B$. A relação lógica é a mesma nos dois casos, mas as respectivas apresentações são diferentes.

2) interpretação: processo que liga as sentenças do sistema formal a uma classe de enunciados cuja verdade ou falsidade pode ser determinada independentemente do sistema. Cabe aqui destacar a relação entre *interpretação* e *modelo*. Um modelo é um conjunto M de elementos colocados em correspondência com os componentes de um sistema formal de tal modo que: a) a cada sentença do sistema formal corresponde um e somente um enunciado formado com elementos de M; b) é possível determinar se um enunciado formado com elementos de M é verdadeiro ou falso independentemente do sistema formal; c) a cada proposição derivável do sistema formal corresponde um e somente um enunciado verdadeiro do modelo. Assim, dar uma interpretação de um sistema formal equivale a dar um modelo para o sistema formal.

A construção de sistemas formais evoluiu muito, desde a Grécia Antiga até os dias de hoje. Inicialmente, adotou-se uma forma de axiomática intuitiva, em que os conceitos primitivos eram dados intuitivos, as proposições primitivas eram evidências e as regras de dedução pertenciam à lógica natural. O melhor exemplo disso está na geometria euclidiana, em que os conceitos primitivos são o ponto, a reta e o plano, as proposições primitivas eram axiomas evidentes a respeito das relações entre estes conceitos — ex.: dois pontos determinam uma reta — e as regras de dedução, embora pertencentes à lógica natural, não eram totalmente explicitadas. Com o tempo, a concepção de sistema formal foi-se tornando cada vez mais abstrata, até que se chegou à concepção atual, em que os conceitos e proposições primitivas não possuem qualquer conteúdo determinado, os procedimentos de dedução são totalmente explicitados. Num sistema formal puro, utiliza-se uma linguagem simbólica rigorosamente definida, mas vazia de conteúdo, e a intuição se reduz à mera manipulação de símbolos.

---

[1] As informações desta seção foram extraídas de Ladrière, J. *Limitaciones Internas de los Formalismos*. Madrid: Ed. Tecnos, s/d, p. 51 ss.

### 2.2. O sistema o/+

A título de ilustração, apresentaremos aqui um exemplo bastante simples de sistema formal, que denominaremos *sistema o/+*. São as seguintes as suas características:

1) Símbolos primitivos: o, +.

2) Regras de formação:

*o* é fbf.

\+ é fbf.

*o* precedido ou seguido de qualquer nº de *o* ou + é fbf.

\+ precedido ou seguido de qualquer nº de *o* ou + é fbf.

3) Proposições primitivas:

Ax 1: o

4)Regras de inferência:

Regra de ligação (RL): φ/φ+

5) Deduções no sistema o/+

$T_1$: o+

Dem.:

| | |
|---|---|
| 1. o | Ax 1 |
| 2. o+ | 1, RL |

$T_2$: o++

Dem.:

| | |
|---|---|
| 1. o+ | $T_1$ |
| 2. o++ | 1, RL |

Outros teoremas do sistema o/+, que podem ser facilmente demonstrados: *o*+++, *o*++++, *o*+++++, etc.

## 2.3. Exercícios sobre Sistemas Formais

2.3.1. Seja o Sistema Formal S, conforme definido abaixo:

Símbolos primitivos:

| | |
|---|---|
| variáveis individuais | α, β, γ |
| operador unitário | $ |
| operadores binários | o, # |
| Parênteses | ( e ) |

Regras de formação:

1- α, β, γ são fbf.
2- se φ é fbf apenas com variável individual, φ$ é fbf.
3- se φ e ψ são fbfs contendo apenas 1 variável individual, φ o ψ e φ # ψ são fbf.
4- se χ é fbf contendo operadores binários, então (χ)$ é fbf.
5- se χ e θ são fbfs contendo operadores binários, então (χ) o φ , φ o (χ) , (χ) # ψ , ψ # (χ) , (χ) o (θ) , (χ) # (θ) , (θ) o (χ) , (θ) # (χ) são fbf.

Definição:

(α o β) = (α$ # β)

Proposições primitivas:

| |
|---|
| $Ax_1$ (α # α) o α |
| $Ax_2$ β o (α # β) |
| $Ax_3$ (α # β) o (β # α) |
| $Ax_4$ (β o γ) o ((α # β ) o (α # γ)) |

Regras de inferência:

| | |
|---|
| Regra de substituição: pode-se derivar de 1 teorema a fbf que se obtém substituindo, em todos os lugares que ocorra, uma variável α por uma fbf β qualquer. Notação: sub α/β |
| Regra de separação: $\dfrac{\underline{\alpha \text{ e } \alpha \text{ o } \beta}}{\beta}$ |
| Regra de equivalência: se A $=_{DF}$ B, então A e B são intercambiáveis em qualquer fbf. |

Demonstrar os seguintes teoremas no Sistema S:

2.3.1.1. (α o α$) o α$

2.3.1.2. β o (α o β)

2.3.1.3. (β o γ) o ((α o β) o (α o γ))

2.3.1.4. α o (α # α)

2.3.1.5. α o α

2.3.1.6. α # α$

Observação: O sistema S constitui, na realidade, apenas uma apresentação diferente do Cálculo Proposicional Clássico, como se pode ver a partir das correspondências estabelecidas na tabela abaixo:

| | Sistema S | Cálculo Proposicional |
|---|---|---|
| variáveis individuais | α, β, γ | p, q, r |
| operador unitário | $ | ~ |
| operadores binários | o, # | ⊃, v |
| Parênteses | ( e ) | ( e ) |
| $Ax_1$ | (α # α) o α | (p v p) ⊃ p |
| $Ax_2$ | β o (α # β) | q ⊃ (p v q) |
| $Ax_3$ | (α # β) o (β # α) | (p v q) ⊃ (q v p) |
| $Ax_4$ | (β o γ) o ((α # β ) o (α # γ)) | (q ⊃ r) ⊃ ((p v q) ⊃ (p v r)) |

2.3.2. Seja agora o Cálculo Proposicional dos PM, definido como segue:[2]

Símbolos primitivos:

| | |
|---|---|
| variáveis individuais | p, q, r |
| operador unitário | ~ |
| operadores binários | v, ⊃ |
| Parênteses | '(' e ')' |

Regras de formação:

| |
|---|
| p, q, r... são fbf. |
| Se α é fbf, ~ α também é fbf. |
| Se α e β são fbf, α v β e α ⊃ β são fbf |
| Se α é fbf, (α) também é fbf |
| definição: α ⊃ β = $_{DF}$ ~ α v β |
| definição: (p . q) = $_{DF}$ ~ (~ p v ~ q) |
| definição: (p ≡ q) = $_{DF}$ (p ⊃ q) . (q ⊃ p) |

Proposições primitivas:

| |
|---|
| A1 (p v p) ⊃ p |
| A2 q ⊃ (p v q) |
| A3 (p v q) ⊃ (q v p) |
| A4 (p v (q v r)) ⊃ (q v (p v r)) |
| A5 (q ⊃ r) ⊃ ((p v q ) ⊃ (p v r)) |

[2] As informações sobre os cálculos proposicional e de predicados dos PM foram extraídas de Russell, B. &

Regras de inferência:

| | |
|---|
| Substituição:<br>A expressão que resulta da substituição de todas as ocorrências de uma variável em um axioma ou um teorema por uma fbf é um teorema |
| Separação: $\dfrac{p \supset q \text{ e } p}{q}$ |
| Equivalência: uma expressão pode ser substituída por outra que lhe seja equivalente |

Demonstrar os seguintes teoremas do Cálculo Proposicional dos PM:

2.3.2.1. $T_1$. $(p \supset \sim p) \supset \sim p$

2.3.2.2. $T_2$. $q \supset (p \supset q)$

2.3.2.3. $T_3$. $(p \supset \sim q) \supset (q \supset \sim p)$

2.3.2.4. $T_4$: $(p \supset (q \supset r)) \supset (q \supset (p \supset r))$

2.3.2.5. $T_5$: $(q \supset r) \supset ((p \supset q) \supset (p \supset r))$

2.3.2.6. $T_6$: $(p \supset q) \supset ((q \supset r) \supset (p \supset r))$

2.3.2.7. $T_7$: $p \supset (p \vee p)$

2.3.2.8. $T_8$: $p \supset p$

2.3.3. Seja agora o Cálculo de Predicados dos PM, definido como segue:

Símbolos primitivos:

| | |
|---|---|
| variáveis proposicionais individuais | p, q, r |
| variáveis individuais | x y, z... |
| constantes individuais | a, b, c... |
| funções proposicionais | Φx, Ψx... |
| quantificadores | (x), (∃x) |
| operador unitário | ~ |
| operadores binários | v, ⊃ |
| Parênteses | ( e ) |

---

Whitehead, A. N. *Principia Mathematica to *56*. Cambridge, Un. Press, 1964.

Regras de formação — valem todas as do cálculo proposicional mais as seguintes:

| |
|---|
| $\Phi x$, $\Psi x$ são fbfs |
| Se $\alpha$ é fbf, então $(x)\ \alpha$ e $(\exists x)\ \alpha$ também são fbfs |
| Definição: $\sim\Phi x = \sim[(x)\ \Phi x]$ |
| Definição: $\sim(x)\Phi x = (\exists x) \sim\Phi x$ |
| Definição: $\sim(\exists x)\ \Phi x = (x) \sim\Phi x$ |
| Definição: $(x)(\Phi x \vee p) = (x)\Phi x \vee p$ |
| Definição: $p \vee (x)\Phi x = (x)(p \vee \Phi x)$ |
| Definição: $p \vee (\exists x)\Phi x = (\exists x)(p \vee \Phi x)$ |
| Definição: $(\exists x)\ \Phi x \vee p = (\exists x)(\Phi x \vee p)$ |
| Definição: $(x)\Phi x \vee (\exists y)\Psi y = (x)(\exists y)(\Phi x \vee \Psi y)$ |
| Definição: $(\exists y)\Psi y \vee (x)\Phi x = (x)(\exists y)(\Psi y \vee \Phi x)$ |

Proposições primitivas — valem as do Cálculo Proposicional dos PM mais as seguintes:

| |
|---|
| A6 $\Phi x \supset (\exists y)\Phi y$ |
| A7 $(\Phi x \vee \Phi y) \supset (\exists z)\ \Phi z$ |

Regras de inferência:

| |
|---|
| Substituição |
| Separação |
| Equivalência |
| Generalização Universal (GU): $\dfrac{\Phi y}{(x)\Phi x}$ |
| Generalização Existencial (GE), que pode ocorrer de três modos diferentes: $\dfrac{\Phi x}{(\exists z)\Phi z}$ , $\dfrac{\Phi x \vee \Phi y}{(\exists z)\Phi z}$ , $\dfrac{\Phi a}{(\exists x)\Phi x}$ |

Demonstrar os seguintes teoremas do Cálculo de Predicados dos PM:

2.3.3.1. T11: $(x)(\Phi x \supset \Psi x) \supset [(x)\Phi x \supset (x)\Psi x]$

2.3.3.2. T12: $(x)\Phi x \supset \Phi y$

2.3.3.3. T13: $(x)\Phi x \supset (x)\Phi x$

2.3.3.4. T14, $(x)(\Phi x \supset \Psi x) \supset [(\exists x)\Phi x \supset (\exists x)\Psi x]$

2.3.3.5. T15: $(\exists x)\Phi x \supset (\exists x)\Phi x$

2.3.3.6. T16: $[(x)(\Phi x \vee (x)\Phi x] \supset (x)\ \Phi x$

## 3. Semântica Formal

### 3.1. A linguagem formal L

Como vimos, um sistema formal, considerado em si mesmo, constitui um conjunto de relações entre símbolos não-interpretados. Isto significa que, neste estágio, ainda não fizemos uma conexão entre o sistema formal e a realidade. Ainda não sabemos sobre que entidades tratam as proposições do sistema. Isto deve ter ficado bem claro no caso do sistema formal S. Ora, uma vez construído um sistema formal em nível puramente sintático, podemos estabelecer uma conexão entre ele a a realidade, através de uma interpretação. Do ponto de vista semântico, podemos construir uma interpretação para o sistema formal através de um modelo. Para ver como isso pode ser feito, construiremos uma linguagem formal L, formada por símbolos não-interpretados, e depois faremos uma conexão entre esta linguagem e algum domínio da realidade, através de uma *semântica formal*. A linguagem formal L é definida como segue:

a) a, e, i são nomes próprios;

b) x, y, z são variáveis individuais;

c) P(x), Q(x) e R(x) são predicados de um lugar;

d) S(x, y) é predicado de dois lugares;

e) T(x, y, z) é predicado de três lugares;

f) (x) e (∃x) são os quantificadores universal e existencial, respectivamente, que funcionam como os quantificadores clássicos;

g) ~, &, v, $\supset$ e $\Leftrightarrow$ são respectivamente as conectivas da negação, conjunção, condicional e equivalência, que também funcionam como as conectivas clássicas;

h) as regras de formação de fbf são as mesmas do cálculo clássico de predicados;

i) as condições-verdade para as fórmulas de L também são as mesmas do cálculo clássico de predicados.

### 3.2. Construção de um modelo M para L

Construamos agora o modelo $M = <D, F>$ para L. Este modelo é constituído pelo conjunto D de objetos e pela função F de interpretação tais que:

$D = \{b, c, d\}$

$F(a) = b; F(e) = c; F(i) = d$

$F(x) = F(a)$ ou $F(e)$ ou $F(i)$

$F(y) = F(a)$ ou $F(e)$ ou $F(i)$

$F(z) = F(a)$ ou $F(e)$ ou $F(i)$

$F[P(x)] = \{b, c, d\}; F[Q(x)] = \{c, d\}; F[R(x)] = \{b\};$

$F[S(x, y)] = \{<b, b>, <b, c>, <b, d>, <d, d>\}$

$F[T(x, y, z)] = \{<c, b, d>\}$

Como se pode ver, através da função de interpretação, fizemos uma ligação entre a linguagem L e uma determinada região do mundo, representada pelo conjunto D de objetos. Podemos agora estabelecer um conjunto de condições de acordo com as quais as expressões de L terão seus valores-verdade determinados com relação ao modelo M:

1) Se $\delta$ é predicado de um lugar e se $\alpha$ é um nome, então $\delta(\alpha)$ é verdadeira com relação a M sse $F(\alpha) \in F(\delta)$. Exemplo: P(a) é V com relação a M sse $F(a) \in F[P(x)]$. Ora, $F(a) = b$ e $F[P(x)] = \{b, c, d\}$. Como $b \in \{b, c, d\}$, segue-se que P(a) é V com relação a M.

2) Se $\gamma$ é predicado de dois lugares e se $\alpha$ e $\beta$ são nomes, então $\gamma(\alpha, \beta)$ é verdadeira com relação a M sse $< F(\alpha), F(\beta) > \in F(\gamma)$. Exemplo: S(e, e) é V com relação a M sse $< F(e), F(e) > \in F[S(x, y)]$. Ora, $F(e) = c$ e $F[S(x, y)] = \{<b, b>, <b, c>, <b, d>, <d, d>\}$. Como $< c, c > \notin \{<b, b>, <b, c>, <b, d>, <d, d>\}$, segue-se que S(e, e) é F.

3) Se $\chi$ é predicado de três lugares e $\alpha$, $\beta$ e $\eta$ são nomes, então $\chi(\alpha, \beta, \eta)$ é verdadeira com relação a M sse $< F(\alpha), F(\beta), F(\eta) > \in F[\chi(\alpha, \beta, \eta)]$. Exemplo: T(e, e, i) é

V com relação a M sse < F(e), F(e), F(i) > $\in$ F[T(x, y, z)]. Ora, F(e) = c, F(i) = d) e F[T(x, y, z)] = {<c, b, d>}. Como < c, c, d > $\notin$ {< c, b, d>}, segue-se que T(e, e, i) é F.

4) Se φ é uma sentença de L, ~φ é V com relação a M sse φ não é V com relação a M.

5) Se φ e ψ são sentenças de L, φ & ψ é V com relação a M sse φ é V com relação a M e ψ é V com relação a M.

6) Se φ e ψ são sentenças de L, φ v ψ é V com relação a M sse φ é V com relação a M ou ψ é V com relação a M.

7) Se φ e ψ são sentenças de L, φ ⊃ ψ é V com relação a M sse φ não é V com relação a M ou ψ é V com relação a M.

8) Se φ e ψ são sentenças de L, φ ⇔ ψ é V com relação a M sse φ e ψ são V ou φ e ψ são F com relação a M.

9) Se u é uma variável individual, então (u)φ é V com relação a M sse φ é V com relação a M para todas as atribuições possíveis de valores a u. Exemplo: (x)P(x) é V com relação a M sse P(x) é V com relação a M para todas as atribuições possíveis de valores a x, ou seja, para x = a e x = e e x = i. Se estas condições forem preenchidas, (x)P(x) é V com relação a M. O esquema abaixo ilustra a situação:

| Proposição de L | Condição para verdade com relação a M |
|---|---|
| (x)P(x) | P(a) é V<br>& P(e) é V<br>& P(i) é V |

Não faremos a avaliação aqui, uma vez que estamos no momento apenas interessados em indicar o processo através do qual ela é feita.

10) Se u é uma variável individual, então (∃u)φ é V com relação a M sse φ é V com relação a M para pelo menos uma das atribuições possíveis de valores a u. Exemplo: (∃x)R(x) é V com relação a M sse R(x) é V com relação a M para pelo menos uma das

atribuições possíveis de valores a x, ou seja, ou para x = a ou x = e ou x = i. Isso pode ser representado através do seguinte esquema:

| Proposição em L | Condição para verdade com relação a M |
|---|---|
| $(\exists x)R(x)$ | OU R(a) é V<br>OU R(e) é V<br>OU R(i) é V |

Não faremos essa avaliação aqui, uma vez que no momento estamos apenas interessados em indicar o processo através do qual ela é feita.

Vejamos agora os casos de dupla quantificação. Se u e v são variáveis individuais, então:

11) $(u)(v)\varphi$ é V com relação a M sse $(v)\varphi$ é V com relação a M para todas as atribuições possíveis de valores a u. $(v)\varphi$, por sua vez, é V com relação a M sse $\varphi$ é V com relação a M para todas as atribuições possíveis de valores a v. Isto envolve uma combinação de cálculos.

Exemplo: (x)(y)S(x, y) em L é V com relação a M sse (y)S(x, y) é V com relação a M para para x = a e x = e e x = i.

Para ser V quando x = a, (y)S(a, y) deve ser V para y = a e y = e e y = i.

Para ser V quando x = e, (y)S(e, y) deve ser V para y = a e y = e e y = i.

Para ser V quando x = i, (y)S(i, y) deve ser V para y = a e y = e e y = i.

Se essas condições forem preenchidas, então (x)(y)S(x, y) em L será V com relação a M. Não faremos avaliação aqui. O processo todo pode ser esquematizado da seguinte maneira:

| Proposição em L | Condição para verdade com relação a M |
|---|---|
| (x)(y)S(x, y) | (y)S(a, y) é V. Neste caso:<br>S(a, a) é V<br>& S(a, e) é V<br>& S(a, i) é V<br>& (y)S(e, y) é V. Neste caso:<br>S(e, a) é V<br>& S(e, e) é V<br>& S(e, i) é V<br>& (y)S(i, y) é V. Neste caso:<br>S(i, a) é V<br>& S(i, e) é V<br>& S(i, i) é V |

Como nos casos anteriores, não faremos a avaliação aqui, deixando apenas apontadas as condições para a sua realização.

12) (u)(∃v)φ é verdadeira com relação a M sse (∃v)φ é V com relação a M para todas as atribuições possíveis a u. (∃v)φ, por sua vez, é V com relação a M sse φ é V com relação a M para pelo menos uma das atribuições possíveis a v.

Exemplo: (x)(∃y)S(x, y) em L é V com relação a M sse (∃y)S(x, y) é V com relação a M para x = a e x = e e x = i (todas as atribuições possíveis a x).

Para ser V quando x = a, (∃y)S(a, y) deve ser V ou para y = a ou y = e ou y = i (pelo menos uma das atribuições a y).

Para ser V quando x = e, (∃y)S(e, y) deve ser V ou para y = a ou y = e ou y = i (pelo menos uma das atribuições a y).

Para ser V quando x = i, (∃y)S(i, y) deve ser V ou para y = a ou y = e ou y = i (pelo menos uma das atribuições a y).

Se essas condições forem preenchidas, então (x)(∃y)S(x, y) em L será V com relação a M. Não faremos a avaliação aqui. O processo todo pode ser esquematizado como segue:

| Proposição em L | Condição para verdade com relação a M |
|---|---|
| (x)(∃y)S(x, y) | (∃y)S(a, y) é V. Neste caso:<br>Ou S(a, a) é V<br>Ou S(a, e) é V<br>Ou S(a, i) é V<br>& (∃y)S(e, y) é V. Neste caso:<br>Ou S(e, a) é V<br>Ou S(e, e) é V<br>Ou S(e, i) é V<br>& (∃y)S(i, y) é V. Neste caso:<br>Ou S(i, a) é V<br>Ou S(i, e) é V<br>Ou S(i, i) é V |

Não efetuaremos a avaliação aqui.

13) (∃v)(u)φ é verdadeira com relação a M sse (u)φ é V com relação a M para pelo menos uma das atribuições possíveis a v. (u)φ, por sua vez, é V com relação a M sse φ é V com relação a M para todas as atribuições possíveis a u.

Exemplo: (∃x)(y)S(x,y) em L é V com relação a M sse (y)S(x,y) é V com relação a M ou para x = a ou x = e ou x = i (pelo menos uma das atribuições a x).

Para ser V quando x = a, ela deve ser V para y = a e y = e e y = i (todas as atribuições a y).

Para ser V quando x = e, ela deve ser V para y = a e y = e e y = i (todas as atribuições a y).

Para ser V quando x = i, ela deve ser V para y = a e y = e e y = i (todas as atribuições a y).

Se estas condições forem preenchidas, (∃x)(y)S(x,y) em L será V com relação a M. Não faremos a avaliação aqui. O processo todo pode ser esquematizado da seguinte maneira:

| Proposição em L | Condição para verdade com relação a M |
|---|---|
| (∃x)(y)S(x,y) | Ou (y)S(a,y) é V. Neste caso:<br>S(a, a) é V<br>& S(a, e) é V<br>& S(a, i) é V<br>Ou (y)S(e, y) é V. Neste caso:<br>S(e, a) é V<br>& S(e, e) é V<br>& S(e, i) é V<br>Ou (y)S(i, y) é V". Neste caso:<br>S(i, a) é V<br>& S(i, e) é V<br>& S(i, i) é V |

Observação importante. (x)(∃y)S(x,y) e (∃y)(x)S(x,y) dizem coisas muito diferentes. (x)(∃y)S(x,y) diz que cada x tem pelo menos um y para o qual S(x, y) é V, significando com isso que este y não precisa ser o mesmo para todo x. (∃y)(x)S(x,y) diz que existe pelo menos um y que torna S(x, y) V para cada x, significando com isso que este y é o mesmo para todo x. A diferença pode ser expressa através da seguinte comparação intuitiva:

(x)(∃y)(y é antepassado de x) significa que todo x tem pelo menos um antepassado y. Aqui, este y não precisa ser o mesmo para todo x.

(∃y)(x)S(y é antepassado de x) significa que existe pelo menos um y que é antepassado de todo x. Aqui, este y precisa ser o mesmo para todo x.

14) (∃v)( ∃u)φ é verdadeira com relação a M sse (u)φ é V com relação a M para pelo menos uma das atribuições possíveis a v. (∃u)φ, por sua vez, é V com relação a M sse φ é V com relação a M para pelo menos uma das atribuições possíveis a u.

Exemplo: (∃x)( ∃y)S(x,y) em L é V com relação a M sse (∃y)S(x,y) é V com relação a M ou para x = a ou x = e ou x = i (pelo menos uma das atribuições a x).

Para ser V quando x = a, ela deve ser V ou para y = a ou y = e ou y = i (pelo menos uma das atribuições a y).

Para ser V quando x = e, ela deve ser V para ou y = a ou y = e ou y = i (pelo menos uma das atribuições a y).

Para ser V quando x = i, ela deve ser V ou para y = a ou y = e ou y = i (pelo menos uma das atribuições a y).

Se estas condições forem preenchidas, (∃x)( ∃y)S(x,y) em L será V com relação a M. Não faremos a avaliação aqui. O processo todo pode ser esquematizado da seguinte maneira:

| Proposição em L | Condição para verdade com relação a M |
|---|---|
| (∃x)( ∃y)S(x,y) | Ou (y)S(a,y) é V. Neste caso:<br>Ou S(a, a) é V<br>Ou S(a, e) é V<br>Ou S(a, i) é V<br>Ou (y)S(e, y) é V. Neste caso:<br>Ou S(e, a) é V<br>Ou S(e, e) é V<br>Ou S(e, i) é V<br>Ou (y)S(i, y) é V". Neste caso:<br>Ou S(i, a) é V<br>Ou S(i, e) é V<br>Ou S(i, i) é V |

O que foi apresentado acima corresponde a uma semântica formal para a linguagem L. A construção de uma semântica formal para um sistema formal como S ou os cálculos lógicos dos *Principia Mathematica* é feita através de processos semelhantes.

Cabe aqui uma observação final. Um sistema pode ser considerado pela perspectiva do tratamento axiomático ou pela perspectiva da teoria dos modelos. Existe um paralelismo entre estes dois tipos de tratamento, que pode ser ilustrado como segue:

| Tratamento pela teoria dos modelos | Tratamento axiomático |
|---|---|
| Uma sentença α de L é válida sse α é V com relação a todo modelo M de L | Uma sentença α de L é um teorema de L sse há uma prova para α em L |
| Uma sentença α de L é uma contradição sse α é F com relação a todo modelo M de L | Uma sentença α de L é uma contradição sse a negação de α é um teorema de L |
| Uma sentença β de L é semanticamente implicada por um conjunto A de sentenças de L sse todo modelo M de L em que todas as sentenças de A são V é também um modelo em que β é V | Uma sentença β de L é demonstrável a partir de um conjunto A de sentenças de L sse há uma seqüência de sentenças de L tal que as mesmas são axiomas ou teoremas ou pertencem a A e a última sentença da sequência é β |
| Duas sentenças α e βde L são logicamente equivalentes sse α e β são V exatamente nos mesmos modelos de L | Duas sentenças α e β de L são logicamente equivalentes sse qualquer uma delas é deduzível a partir da outra |

### 3.3. Exercícios de Semântica Formal

3.3.1. Com base nas condições estipuladas na seção anterior, determine o valor-verdade das seguintes expressões de L com respeito ao modelo M (mostre o processo utilizado para obter a resposta):

3.3.1.1. (x)P(x)

3.3.1.2. (∃x)P(x)

3.3.1.3. ~(x)Q(x)

3.3.1.4. ~(∃x)Q(x)

3.3.1.5. P(i) v ~Q(i)

3.3.1.6. P(i) & ~Q(i)

3.3.1.7. S(a, e)

3.3.1.8. ~S(a, i)

3.3.1.9. (x)S(x, i)

3.3.1.10. (x)(y)S(x, y)

3.3.1.11. (x)(∃y)S(x, y)

3.3.1.12. (∃y)(x)S(x, y)

3.3.1.13. (x)[P(x) & Q(x)]

# 4. Lógica modal[3]

## 4.1. Caracterização geral

A Lógica modal surgiu como tentativa de se encontrar uma abordagem mais adequada para a condicional material. Na lógica simbólica clássica, a condicional material possui algumas propriedades paradoxais, como, por exemplo: a) uma proposição verdadeira é implicada por qualquer proposição; b) uma proposição falsa implica qualquer proposição; c) não há uma ligação efetiva entre o antecedente e o conseqüente de uma implicação material. Na lógica modal, a *condicional material*, representada por *se P, então Q*, é substituída pela *implicação estrita*, que supõe uma relação necessária entre o antecedente e o conseqüente. Assim, a sentença *se P, então Q* é interpretada como significando que P implica estritamente *Q* e equivalendo a *necessariamente: se P, então Q*. A condicional usada por Frege é conservada, mas introduz-se um operador modal que reforça a conexão lógica entre antecedente e conseqüente. Esta relação é representada da seguinte maneira:

$L(p \supset q)$.

---

[3] As informações relativas a esta parte foram extraídas de Hughes, G. E. & Cresswell, M. J. *A New Introduction to Modal Logic.* London and N. York: Routledge, 1996, e de Girle, R. *Modal Logics and Philosophy*. Montreal, London, Ithaca: McGill-Queen's University Press, 2000.

Aqui, o símbolo *L* corresponde a *necessariamente*. Outro operador modal muito utilizado é o da *possibilidade*, representado por *M*. Os diversos sistemas modais são construídos pela adição gradativa de axiomas modais ao cálculo clássico.

As proposições declarativas podem ser de dois tipos: as de *inesse* e as modais. As proposições de *inesse* são aquelas em que o predicado é atribuído ao sujeito de maneira simples, sem qualquer modificação adverbial, como, por exemplo, em *S é P*. As propposições modais são aquelas em que o predicado é atribuído ao sujeito através de uma modificação adverbial, como, por exemplo, em *S é possivelmente P*, *S é necessariamente P*, etc. Cabe ainda observar que, nestas proposições, a modalidade pode ser entendida de duas maneiras: *de re* e *de dicto*. A modalidade é *de re* quando afeta a cópula que liga o sujeito ao predicado. Se, por exemplo, dizemos que *S é necessariamente P*, estamos atribuindo o predicado *P* ao sujeito *S* de maneira necessária. Esta modalidade afeta a própria coisa e por isso é denominada *de re*. Se, por outro lado, dizemos que *necessariamente S é P*, estamos considerando a proposição *S é P* necessária. Esta modalidade afeta aquilo que estamos dizendo e por isso é denominada *de dicto*. Como a modalidade *de re* é intratável do ponto de vista formal, utilizamos apenas a modalidade *de dicto* nos sistemas de lógica modal.

A construção de sistemas de lógica modal gera as seguintes conseqüências. Em primeiro lugar, a introdução de novos operadores traz consigo novas relações lógicas, que se baseiam nas definições seguintes:

Lp = é necessário que p;

Mp = é possível que p;

Lp ≡ ~M~p (necessário p = não possível não p);

Mp ≡ ~L~p (possível p = não necessário não p).

Há também dois outros operadores modais, que podem ser definidos com base nos anteriores:

é impossível que p = ~Mp.

é contingente que p = é possível que p e é possível que não p = Mp & M~p.[4]

[4] Esta definição de *contingência* se baseia em Aristóteles. Apesar disso, alguns autores ligados à escolástica medieval definem *é contingente que p* como equivalente a *é possível que não p*. Esta definição, por ser muito simplista, somente será usada mais à frente, em alguns tipos de exercícios.

Estes operadores modais permitem uma ampliação das relações lógicas de oposição entre as proposições. Para esclarecer isso, recapitulemos alguns aspectos relevantes da lógica aristotélica. Segundo Aristóteles e seus seguidores medievais, a lógica é o estudo das inferências feitas com o auxílio de sentenças declarativas. Essas últimas possuem uma estrutura comum, que pode constituir sua *forma geral.* De acordo com ela, toda sentença pode ser analisada nos seguintes elementos constitutivos:

S é P

No esquema acima, *S* corresponde ao *sujeito* — aquele de quem se diz alguma coisa —, *P*, ao *predicado* — aquilo que se diz do sujeito —, sendo ambos ligados através da *cópula,* representada pela palavra *é*. Esta análise permite que Aristóteles faça a seguinte classificação das sentenças declarativas, com base na extensão do sujeito (universal ou particular) e no tipo de ligação entre sujeito e predicado (afirmação ou negação):

Universal afirmativa: Todo A é B.

Universal negativa: Nenhum A é B.

Particular afirmativa: Algum A é B.

Particular negativa: Algum A não é B.

A partir desta classificação, Aristóteles constrói o famoso *quadrado lógico da oposição*, que mostra as principais relações entre essas sentenças:

| | | | | |
|---|---|---|---|---|
| Todo A é B | | <= contrárias => | | Nenhum A é B |
| ⇑ | ↖ | | ↗ | ⇑ |
| subalternas | | Contraditórias | | subalternas |
| ⇓ | ↙ | | ↘ | ⇓ |
| Algum A é B | | <= subcontrárias => | | Algum A não é B |

Aqui, duas sentenças contraditórias não podem ser nem verdadeiras nem falsas ao mesmo tempo — é o caso dos pares *Todo A é B/Algum A não é B* e *Nenhum A é B/Algum A é B*); duas sentenças contrárias não podem ser verdadeiras ao mesmo tempo, mas podem ser falsas ao mesmo tempo — é o caso de *Todo A é B* e *Nenhum A é B*; duas subcontrárias podem ser verdadeiras ao mesmo tempo, mas não podem ser falsas ao mesmo tempo — é o caso de *Algum A é B* e *Algum A não é B*; duas subalternas são tais que a sentença universal implica a particular — é o caso dos pares *Todo A é B*/*Algum A é B* e *Nenhum A é B/Algum A não é B*.

Coisa semelhante acontece com os operadores modais acima apresentados. Representando *é impossível que p* por *Ip*, teremos um novo quadrado lógico da oposição:

| | | | | |
|---|---|---|---|---|
| é necessário que p<br>não é possível que não p<br>é impossível que não p | | <= contrárias => | | é necessário que não p<br>é impossível que p<br>não é possível que p |
| ⇑ | ↖ | | ↗ | ⇑ |
| subalternas | | contraditórias | | subalternas |
| ⇓ | ↙ | | ↘ | ⇓ |
| não é necessário que não p<br>não é impossível que p<br>é possível que p | | <= subcontrárias => | | não é necessário que p<br>é possível que não p<br>não é impossível que não p |

Em símbolos, teríamos:

| | | | | |
|---|---|---|---|---|
| L p<br>~M~p<br>I~p | | <= contrárias => | | L~p<br>Ip<br>~Mp |
| ⇑ | ↖ | | ↗ | ⇑ |
| subalternas | | contraditórias | | subalternas |
| ⇓ | ↙ | | ↘ | ⇓ |
| ~L~p<br>~Ip<br>Mp | | <= subcontrárias => | | ~Lp<br>M~p<br>~I~p |

Todas as expressões que se encontram num mesmo canto do quadrado são equivalentes. Por exemplo, L p ≡ ~M~p ≡ I~p. Aqui também são válidas as mesmas leis que regem as contrárias, as contraditórias, as subalternas e as subcontrárias do quadrado lógico aristotélico.

Em segundo lugar, cai o tratamento extensional dos operadores lógicos. Do ponto de vista da bivalência do cálculo clássico, os operadores modais são *intensionais*. Isso significa que a tabela bivalente de valores-verdade apresenta lacunas em seu preenchimento, como se pode ver pelo quadro abaixo:

| p | Lp | Mp | L~p | M~p |
|---|---|---|---|---|
| V | ? | V | F | ? |
| F | F | ? | ? | V |

Se a sentença *p* é verdadeira, por exemplo, isso não oferece informações suficientes para sabermos se a sentença *Lp* (*é necessário que p*) é verdadeira ou falsa; se *p* é falsa, por outro lado, não temos condições de saber se *Mp* (*é possível que p*) é verdadeira ou falsa; e assim sucessivamente.

Em terceiro lugar, os novos operadores podem ser reiterados indefinidamente, gerando expressões muito pouco intuitivas e difíceis de avaliar, como em *LMMLMLMMMLp*. Para resolver esse problema, são necessárias *regras de redução*, por intermédio das quais as expressões modais mais complexas, envolvendo um grande número de operadores modais, podem ser reduzidas a expressões mais simples, com apenas um operador modal. Essas regras de redução são quatro, conforme explicitado no esquema abaixo:

LMp ≡ Mp

MLp ≡ Lp

MMp ≡ Mp

LLp ≡ Lp

Com base nestas regras, a expressão *LMMLMLMMMLp* pode ser reduzida da seguinte maneira: *LMMLMLMMMLp* ≡ *MMLMLMMMLp* ≡ *MLMLMMMLp* ≡ *LMLMMMLp* ≡

*MLMMMLp* ≡ *LMMMLp* ≡ *MMMLp* ≡ *MMLp* ≡ *MLp* ≡ *Lp*. O sistema modal S5, elaborado por Lewis, contém todas as quatro regras, como veremos mais adiante.

Em quarto lugar, estes operadores também admitem paradoxos. A expressão *L(q ⊃ Lp)*, por exemplo, que é um teorema, nos diz que uma sentença necessária — no caso, *Lp* — é implicada estritamente por qualquer sentença — no caso, *q*. *L(~Mq ⊃ p)*, que também é um teorema, por sua vez nos diz que uma sentença impossível — no caso, *~Mp* — implica estritamente qualquer sentença — no caso, *p*. Nessa perspectiva, a implicação estrita simplesmente "modalizou" os paradoxos da implicação material.

### 4.2. Os sistemas modais K, T, D e S4

Os diversos sistemas modais são construídos pela adição gradativa de axiomas modais ao cálculo clássico. O sistema modal mais simples é o que resulta quando adicionamos aos axiomas do cálculo proposicional clássico o axioma K: *L(p ⊃ q) ⊃ (Lp ⊃ Lq)*. A partir desse, temos, por exemplo, em ordem de complexidade crescente, os sistemas T, D, S4 e S5. Cada um resulta do acréscimo de pelo menos mais um axioma modal ao conjunto de axiomas do sistema anterior.

Como o sistema modal mais simples é o sistema K, começaremos por ele. Podemos defini-lo assim:

Símbolos primitivos:

Variáveis proposicionais: p, q, r etc.

Operadores monádicos: ~, L.

Operador diádico: v.

Parênteses: ( e ).

Regras de formação (além das regras para uso de parênteses):

Uma variável proposicional é fbf.

Se $\alpha$ é fbf, então $\sim\alpha$ e $L\alpha$ são fbf.

Se $\alpha$ e $\beta$ são fbfs, então $(\alpha \vee \beta)$ é fbf.

Definições:

$\alpha \ \& \ \beta =_{Df} \sim(\sim\alpha \vee \sim\beta)$

$\alpha \supset \beta =_{Df} (\sim\alpha \vee \beta)$

$\alpha \equiv \beta =_{Df} (\alpha \supset \beta) \ \& \ (\beta \supset \alpha)$

$M\alpha =_{Df} \sim L \sim\alpha$

Axiomas:

Se $\alpha$ é uma expressão válida no Cálculo Proposicional, então $\alpha$ é axioma.

Axioma K: $L(p \supset q) \supset (Lp \supset Lq)$

Regras de inferência:

Regra de substituição (como no Cálculo Proposicional).

Regra de separação (como no CP)

Regra de equivalência (como no CP)

Regra da inclusão da necessidade: se $\alpha$ é teorema, então $L\alpha$ também é teorema.

Como se pode ver, o sistema K resulta da adição do axioma K e da regra da inclusão da necessidade ao Cálculo Proposicional clássico.

Um sistema modal um pouco mais forte do que K é o sistema T, formado pela adição do seguinte axioma ao sistema K:

Axioma T: $Lp \supset p$.

Esta expressão é conhecida como *axioma da necessidade*.

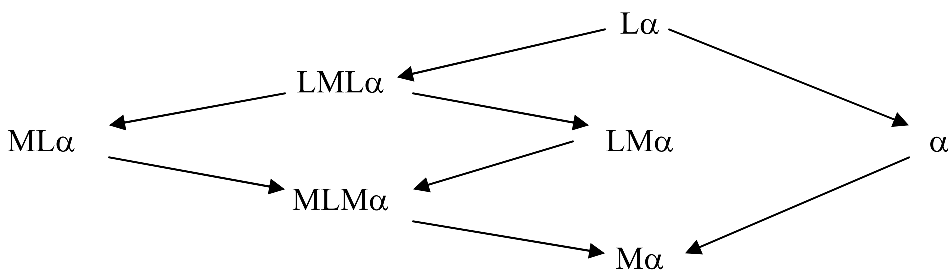
Depois do sistema T, temos o sistema D, que consiste na adição do seguinte axioma ao sistema K:

Axioma D: $Lp \supset Mp$.

A seguir, temos o sistema S4, que é formado pela adição ao sistema T do axioma seguinte:

Axioma 4: $Lp \supset LLp$.

São as seguintes as relações entre as diversas modalidades em S4:




## 4.3. O sistema modal S5

O sistema modal proposicional mais forte que apresentaremos aqui é o sistema S5, o quinto de uma série de sistemas criados por C. I. Lewis (S1-S5). O sistema S5 é importante porque é a mais simples dentre as lógicas baseadas em mundos possíveis. S5 é formado pela adição do seguinte axioma ao sistema T:

Axioma E: $Mp \supset LMp$

No sistema S5, são válidas as seguintes leis de redução para as modalidades reiteradas:

Regra 1: $LMp \equiv Mp$.

Regra 2: $MLp \equiv Lp$.

Regra 3: $MMp \equiv Mp$.

Regra 4: $LLp \equiv Lp$.

Para efeito de comparação com S5, podemos dizer que os sistemas modais apresentados até agora possuem as seguintes características:

Sistema K: formado pelo CP mais o axioma K $[L(p \supset q) \supset (Lp \supset Lq)]$.

Sistema D: formado pelo CP mais o axioma K mais o axioma D $[Lp \supset Mp]$. D não é teorema de K.

Sistema T: formado pelo CP mais os axiomas K e T $[Lp \supset p]$. D é teorema de T, mas T não é teorema de D.

Sistema S4: formado pelo CP mais os axiomas K, T e 4[$Lp \supset LLp$].

Sistema S5: formado pelo CP mais os axiomas K, T e E [$Mp \supset LMp$]. Os axiomas D e 4 são teoremas de S5. S4 e S5 trabalham modalidades iteradas, mas a regra 1 de redução não vale em S4. Apenas em S5 valem todas as regras de redução acima apresentadas.

Exemplos de demonstrações no sistema modal K:

1º Exemplo: demonstrar que: $L(p \& q) \supset (Lp \& Lq)$ (Teorema K1)

Dem:

| Etapa | Justificativa |
|---|---|
| 1. $(p \& q) \supset p$ | CP |
| 2. $L((p \& q) \supset p)$ | 1 nec. |
| 3. $L(p \supset q) \supset (Lp \supset Lq)$ | K |
| 4. $L((p \& q) \supset p) \supset ((L(p \& q) \supset Lp)$ | 3 sub. p/p & q, q/p |
| 5. $L(p\&q) \supset Lp$ | 2,4 sep. |
| 6. $(p \& q) \supset q$ | CP |
| 7. $L((p \& q) \supset q)$ | 6 nec. |
| 8. $L(p \& q) \supset q) \supset (L(p \& q) \supset Lq)$ | 3 sub. p/p & q |
| 9. $L(p \& q) \supset Lq$ | 7,8 sep. |
| 10. $(p \supset q) \supset ((p \supset r) \supset (p \supset (q \& r)))$ | CP |
| 11. $(L(p \& q) \supset Lp) \supset ((L(p \& q) \supset Lq) \supset (L(p \& q) \supset (Lp \& Lq)))$ | 10 sub. p/L(p & q), q/ Lp, r/Lq |
| 12. $(L(p \& q) \supset Lq) \supset (L(p \& q) \supset (Lp \& Lq))$ | 5,11 sep. |
| 13. $L(p \& q) \supset (Lp \& Lq)$ | 9,12 sep. |

2º Exemplo: $(Lp \& Lq) \supset L(p \& q)$ (Teorema K2)

Dem:

| Etapa | Justificativa |
|---|---|
| 1. $p \supset (q \supset (p \& q))$ | CP |
| 2. $L(p \supset (q \supset (p \& q)))$ | 2 nec. |
| 3. $L(p \supset q) \supset (Lp \supset Lq)$ | K |
| 4. $L(p \supset (q \supset (p \& q))) \supset (Lp \supset L(q \supset (p \& q)))$ | 3 sub. q/q ⊃ (p & q) |
| 5. $Lp \supset L(q \supset (p \& q))$ | 2,4 sep. |
| 6. $L(q \supset (p \& q)) \supset (Lq \supset L(p \& q))$ | 3 sub p/q, q/p & q |
| 7. $(p \supset q) \supset ((q \supset (r \supset s)) \supset ((p \& r) \supset s))$ | CP |
| 8. $(Lp \supset L(q \supset (p \& q))) \supset ((L(q \supset (p \& q)) \supset (Lq \supset L(p \& q))) \supset ((Lp \& Lq) \supset L(p \& q)))$ | 7 sub. p/Lp, q/L(q ⊃ (p & q)), r/Lq, s/L(p & q) |
| 9. $(L(q \supset (p \& Q)) \supset (Lq \supset L(p \& q))) \supset ((Lp \& Lq) \supset L(p \& q)))$ | 5,8 sep. |
| 10. $(Lp \& Lq) \supset L(p \& q)$ | 6,9 sep. |

3º Exemplo: $L(p \& q) \equiv (Lp \& Lq)$ (Teorema K3)

Dem:

| Etapa | Justificativa |
|---|---|
| 1. $L(p \& q) \supset (Lp \& Lq)$ | K1 |
| 2. $(Lp \& Lq) \supset L(p \& q)$ | K2 |
| 3. $(p \supset q) \supset ((q \supset p) \supset (p \equiv q))$ | CP |
| 4. $(L(p \& q) \supset (Lp \& Lq)) \supset (((Lp \& Lq) \supset L(p \& q)) \supset (L(p \& q) \equiv (Lp \& Lq)))$ | 3 sub. p/L(p & q), q/Lp & Lq |
| 5. $(Lp \& Lq) \supset L(p \& q)) \supset (L(p \& q) \equiv (Lp \& Lq))$ | 1,4 sep. |
| 6. $L(p \& q) \equiv (Lp \& Lq)$ | 2,5 sep. |

Exemplos de demonstrações no sistema T:

1º Exemplo: $p \supset Mp$ (Teorema T1)

Dem:

| Etapa | Justificativa |
|---|---|
| 1. $Lp \supset p$ | T |
| 2. $L{\sim}p \supset {\sim}p$ | 1 sub. p/~p |
| 3. $(p \supset q) \supset ({\sim}q \supset {\sim}p)$ | CP |
| 4. $(L{\sim}p \supset {\sim}p) \supset ({\sim}{\sim}p \supset {\sim}L{\sim}p)$ | 3 sub. p/L~p, q/~p |
| 5. ${\sim}{\sim}p \supset {\sim}L{\sim}p$ | 2,4 sep. |
| 6. $p \supset Mp$ | 5 eq. |

2º Exemplo: $M(p \supset Lp)$ (Teorema T2)

Dem:

| Etapa | Justificativa |
|---|---|
| 1. $p \supset Mp$ | T1 |
| 2. $Lp \supset MLp$ | 1 sub. p/Lp |
| 3. $M(p \supset q) \equiv (Lp \supset Mq)$ | K7 |
| 4. $M(p \supset Lp) \equiv (Lp \supset MLp)$ | 3 sub. q/Lp |
| 5. $M(p \supset Lp)$ | 2,4 eq. |

Exemplo de demonstração no sistema D: $M(p \supset p)$

Dem:

| Etapa | Justificativa |
|---|---|
| 1. $p \supset p$ | CP |
| 2. $L(p \supset p)$ | 1 nec. |
| 3. $Lp \supset Mp$ | D |
| 4. $L(p \supset p) \supset M(p \supset p)$ | 3 sub. p/p ⊃ p |
| 5. $M(p \supset p)$ | 2,4 sep. |

Exemplos de demonstração no sistema S4:

1º Exemplo: $MMp \supset Mp$

Dem:

| Etapa | Justificativa |
|---|---|
| 1. $Lp \supset LLp$ | Axioma 4 |
| 2. $L\sim p \supset LL\sim p$ | 1 sub p/~p |
| 3. $\sim Mp \supset L\sim Mp$ | 2 eq. |
| 4. $\sim Mp \supset \sim MMp$ | 3 eq. |
| 5. $MMp \supset Mp$ | 4 eq. |

2º Exemplo: $Lp \equiv LLp$

Dem:

| Etapa | Justificativa |
|---|---|
| 1. $Lp \supset p$ | T |
| 2. $LLp \supset Lp$ | 1 sub p/Lp |
| 3. $Lp \supset LLp$ | 4 |
| 4. $(p \supset q) \supset ((q \supset p) \supset (p \equiv q))$ | CP |
| 5. $(Lp \supset LLp) \supset ((LLp \supset Lp) \supset (Lp \equiv LLp))$ | 4 sub. p/Lp, q/LLp |
| 6. $(LLp \supset Lp) \supset (Lp \equiv LLp)$ | 3,5 sep. |
| 7. $Lp \equiv LLp$ | 2,6 sep. |

Exemplos de demonstrações no sistema S5:

1º Exemplo: $MLp \supset Lp$

Dem:

| Etapa | Justificativa |
|---|---|
| 1. $Mp \supset LMp$ | E |
| 2. $M\sim p \supset LM\sim p$ | 1 sub. p/~p |
| 3. $\sim Lp \supset L\sim Lp$ | 2 eq. |
| 4. $\sim Lp \supset \sim MLp$ | 3 eq. |
| 5. $MLp \supset Lp$ | 4 eq. |

2º Exemplo: 4.15.2. $Mp \equiv LMp$

Dem:

| Etapa | Justificativa |
|---|---|
| 1. $Lp \supset p$ | T |
| 2. $LMp \supset Mp$ | 1 sub. p/Mp |
| 3. $(p \supset q) \supset ((q \supset p) \supset (p \equiv q)$ | CP |
| 4. $(Mp \supset LMp) \supset ((LMp \supset Mp) \supset (Mp \equiv LMp)$ | 3 sub. p/Mp, q/LMp |
| 5. $Mp \supset LMp$ | E |
| 6. $(LMp \supset Mp) \supset (Mp \equiv LMp)$ | 4,5 sep. |
| 7. $Mp \equiv LMp$ | 2,6 sep. |

3º Exemplo: $Lp \equiv MLp$

Dem:

| Etapa | Justificativa |
|---|---|
| 1. $p \supset Mp$ | Teorema de S5 |
| 2. $Lp \supset MLp$ | 1 sub p/Lp |
| 3. $MLp \supset Lp$ | Teorema de S5 |
| 4. $(p \supset q) \supset ((q \supset p) \supset (p \equiv q))$ | CP |
| 5. $(Lp \supset MLp) \supset ((MLp \supset Lp) \supset (Lp \equiv MLp))$ | 4 sub. p/Lp, q/MLp |
| 6. $(MLp \supset Lp) \supset (Lp \equiv MLp)$ | 2,5 sep. |
| 7. $Lp \equiv MLp$ | 3,6 sep. |

Como se pode ver, as demonstrações nos sistemas modais acima são perfeitamente possíveis pelo método axiomático. Este processo, contudo, é bastante burocrático e por vezes exige caminhos muito longos e tortuosos para atingir seus objetivos. Ora, além da demonstração de teoremas com base nos axiomas e regras de inferência, os sistemas modais também admitem um processo mais simples para testar a validade de uma expressão bem formada. Este processo é o da dedução natural, que já deve ser conhecido pelo leitor no caso da lógica clássica das proposições. Por se tratar de um procedimento mais acessível e mais fácil com relação ao método axiomático, ele será adotado também aqui. Para aplicar a dedução natural à lógica modal de predicados, teremos de acrescentar, às regras já conhecidas para a lógica das proposições declarativas, algumas regras para operadores modais. As regras de dedução da lógica das proposições declarativas, que o leitor já deve conhecer, são as seguintes:

1) Silogismo copulativo (abreviatura: *sc*):

de $\sim(P \,\&\, Q)$

e $P$,

pode-se concluir $\sim Q$

2) *Modus ponens* (abreviatura: *mp*):

de $P \supset Q$

e $P$,

pode-se concluir $Q$

3) *Modus tollens* (abreviatura: *mt*):

de $P \supset Q$

e $\sim Q$,

pode-se concluir $\sim P$

4) Silogismo hipotético (abreviatura: *sh*):

de $P \supset Q$

e $Q \supset R$,

pode-se concluir $P \supset R$

5) Conjunção (abreviatura: *c*):

de $P$

e $Q$,

pode-se concluir $P \,\&\, Q$

6) Silogismo disjuntivo (abreviatura: *sd*):

de $P \vee Q$

e $\sim P$,

pode-se concluir $Q$

7) Simplificação (abreviatura: *s*):

de $P \,\&\, Q$

pode-se concluir $P$

8) Adição (abreviatura: *a*):

de $P$

pode-se concluir $P \vee Q$

9) Substituição por equivalência (abreviatura: *eq*):

Se $S \equiv T$, então $T$ pode substituir qualquer ocorrência de $S$ nas diversas etapas da prova (por exemplo, poderemos sempre substituir $P \supset Q$ por $\sim P \vee Q$).

10) Redução ao absurdo (abreviatura: *ra*):

Se, a partir de uma determinada premissa, chegamos a uma contradição, então podemos considerar válida a negação daquela premissa.

11) Demonstração condicional (abreviatura: *dc*)

Se, a partir da suposição da premissa $P$, podemos obter $Q$, então a expressão $P \supset Q$ também pode ser inferida.

12) Reiteração (abreviatura: *r*):

de $P$

pode-se concluir $P$.

A estas regras, acrescentamos as seguintes, para trabalharmos com operadores modais:

13) Regra da eliminação da necessidade (abreviatura: *en*):

de $Lp$

pode-se concluir $p$

14) Regra da introdução da necessidade (abreviatura: *in*):

de $p$

pode-se concluir $Lp$

15) Regra da introdução da possibilidade (abreviatura: *ip*):

de $p$

pode-se concluir $Mp$

16) Regra de reiteração modal (abreviatura: *rm*):

de $Lp$ ou de $Mp$

pode-se concluir, respectivamente, $Lp$ ou $Mp$

Através da dedução natural, podemos demonstrar a validade de fórmulas do cálculo modal proposicional S5, como se verá a seguir.

1º Exemplo: demonstrar que $Lp \supset LLp$. A prova está ilustrada no quadro abaixo:

| Etapa | Justificativa |
|---|---|
| 1. Lp | suposição inicial |
| 2.L Lp | 1, in |
| 3. Lp ⊃ LLp | 1-2, dc |

2º Exemplo: demonstrar que *L(p & q) ⊃ (Lp & Lq)*. A prova está ilustrada no quadro abaixo:

| Etapa | Justificativa |
|---|---|
| 1. L(p & q) | suposição inicial |
| 2. p & q | 1, en |
| 3. p | 2, s |
| 4. q | 2, s |
| 5. Lp | 3, in |
| 6. Lq | 4, in |
| 7. Lp & Lq | 5, 6 c |
| 8. L(p & q) ⊃ (Lp & Lq) | 1-7, dc |

### 4.4. Lógica modal de predicados

Por motivos de espaço, apresentaremos aqui apenas o cálculo modal de predicados S5. Este último é obtido através de uma expansão do cálculo proposicional S5, tornando-o um cálculo modal de predicados, através da adição dos símbolos relevantes para predicados e das regras correspondentes de formação e de inferência. Isto corresponde a uma modalização do cálculo clássico de predicados, de tal forma que todas as fórmulas válidas neste cálculo são acrescidas de fórmulas válidas incluindo operadores modais. Deste modo, muitos teoremas da lógica modal de predicados S5 correspondem a instâncias de teoremas do cálculo proposicional modal, como, por exemplo, *L[Px ⊃ Fx] ⊃ [LPx ⊃ LFx]*. Outros teoremas, porém, são instâncias do cálculo de predicados, como, por exemplo, *(x)LPx ⊃ LPx*. O interesse maior da lógica modal de predicados está porém nos princípios "mistos", que exibem interrelações entre operadores modais e quantificadores. Um dos mais conhecidos é a *fórmula de Barcan*, sugerido inicialmente por Ruth Barcan, e cuja expressão mais simples é dada pela expressão:

$(x)LFx \supset L(x)Fx$.

De acordo com a interpretação padrão, esta fórmula significa: se tudo necessariamente possui a propriedade $F$, então necessariamente tudo possui a propriedade $F$. Todas as fórmulas abaixo são derivadas da *fórmula de Barcan* (abreviatura: *fb*):

$L(x)Fx \supset (x)LFx$ (conversa da fb com quantificador universal);

$M(\exists x)Fx \supset (\exists x)MFx$ (fb com quantificador existencial);

$(\exists x)MFx \supset M(\exists x)Fx$ (conversa da fb com quantificador existencial).

Estas expressões permitem concluir que

$(\exists x)MFx \equiv M(\exists x)Fx$

e que

$(x)LFx \equiv L(x)Fx$.

O método de dedução natural também pode ser utilizado aqui, para demonstração de proposições da lógica modal S5 de predicados. Para tanto, basta acrescentar, às regras apresentadas na seção anterior, aquelas relativas aos quantificadores e que já devem ser conhecidas do leitor:

17) Regra de instanciação universal (para eliminar o quantificador universal; abraviatura: *iu*):

de $(x)Fx$

pode-se concluir $Fx$ ou $Fa$

18) Regra de instanciação existencial (para eliminar o quantificador existencial; abreviatura: *ie*):

de $(\exists x) Fx$

pode-se concluir $Fa$

[Restrição: uma constante diferente para cada quantificador eliminado]

19) Regra de generalização universal (para recuperar o quantificador universal; abreviatura: *gu*):

de *Fx*

pode-se concluir *(x) Fx*

20) Regra de generalização existencial (para recuperar o quantificador existencial; abreviatura: *ge*):

de *Fx* ou de *Fa*

pode-se concluir *(∃x) Fx*

Com base nestas regras, podemos realizar demonstrações de proposições válidas no cálculo S5. É o que veremos nos casos abaixo.

1º Exemplo: demonstrar que *(x)LFx ⊃ L(x)Fx*. Esta é a fórmula de Barcan, cuja validade pode ser estabelecida através do quadro abaixo:

| Etapa | Justificativa |
|---|---|
| 1. (x)LFx | suposição inicial |
| 2. LFx | 1, iu |
| 3. Fx | 2, en |
| 4. (x)Fx | 3, gu |
| 5. L(x)Fx | 4, in |
| 6. (x)LFx ⊃ L(x)Fx | 1-5, dc |

2º Exemplo: demonstrar que *L(x)Fx ⊃ (x)LFx*. Esta é a conversa da fb com quantificador universal e sua validade pode ser estabelecida como segue:

| Etapa | Justificativa |
|---|---|
| 1. L(x)Fx | suposição inicial |
| 2.(x)Fx | 1, em |
| 3. Fx | 2, iu |
| 4. LFx | 3, in |
| 5. (x)LFx | 4, gu |
| 6. L(x)Fx ⊃ (x)LFx | 1-5, dc |

3º Exemplo: demonstrar que *M(∃x)Fx ⊃ (∃x)MFx*. Esta é a fb com quantificador existencial e sua validade pode ser assim estabelecida:

| Etapa | Justificativa |
|---|---|
| 1. (x)LFx ⊃ L(x)Fx | fb (já dem.) |
| 2. (x)L~Fx ⊃ L(x)~Fx | 1 sub Fx/~Fx) |
| 3. (x)~MFx ⊃ L~(∃x)Fx | 2 eq |
| 4. ~(∃x)MFx ⊃ ~M(∃x)Fx | 3 eq |
| 5. M(∃x)Fx ⊃ (∃x)MFx | 4 eq |

4º Exemplo: demonstrar que *(∃x)MFx ⊃ M(∃x)Fx*. Esta é a conversa da fb com quantificador existencial. Sua validade pode ser estabelecida através do quadro abaixo:

| Etapa | Justificativa |
|---|---|
| 1. L(x)Fx ⊃ (x)LFx | conversa da fb (já dem.) |
| 2. L(x)~Fx ⊃ (x)L~Fx | 1 sub Fx/~Fx) |
| 3. L~(∃x)Fx ⊃ (x)~MFx | 2 eq |
| 4. ~M(∃x)Fx ⊃ ~(∃x)MFx | 3 eq |
| 5. (∃x)MFx ⊃ M(∃x)Fx | 4 eq |

5º Exemplo: demonstrar que *(x)MPx ⊃ M(x)Px*. Esta fórmula não é válida em S5, como se pode ver pelo quadro abaixo:

| Etapa | Justificativa |
|---|---|
| 1. (x)MPx | suposição inicial |
| 2. MPx | 1, iu |
| 3. MPa | 2 iu |
| 4. ? | |

Não há como levar adiante a demonstração, pois não temos uma regra de eliminação da possibilidade. Isto significa que aqui não conseguiremos chegar a *M(x)Px* a partir da suposição inicial *(x)MPx*. Tentemos um processo alternativo, que costuma funcionar quando a modalidade da possibilidade está envolvida: a demonstração por redução ao absurdo. Neste caso, se *(x)MPx ⊃ M(x)Px* é válida, então a conjunção do antecedente *(x)MPx* com a negação do conseguente *~M(x)Px* deverá levar a uma contradição. O procedimento está ilustrado no quadro abaixo:

| Etapa | Justificativa |
|---|---|
| 1. (x)MPx ⊃ M(x)Px | hipótese |
| 2. ~[(x)MPx ⊃ M(x)Px] | negação da hipótese (nh) |
| 3. (x)MPx & ~M(x)Px | 2 eq |
| 4. (x)MPx | 3 s |
| 5. ~M(x)Px | 3 s |
| 6. L~(x)Px | 5 eq |
| 7. ~(x)Px | 6 en |
| 8. (∃x)~Px | 7 eq |
| 9. ~Pa | 8 ie |
| 10. M~Pa | 9 ip |
| 11. MPx | 4 iu |
| 12. MPa | 11 ie |
| 13. ? | ? |

O máximo de oposição a que podemos chegar neste caso é entre *MPa*, estabelecida na etapa 7, e *M~Pa*, estabelecida na etapa 12. Ora, *Pa* e *~Pa* podem ser possíveis ao mesmo tempo sem gerar contradição. Como se pode ver, nem o método de redução ao absurdo nos leva à validação da expressão *(x)MPx ⊃ M(x)Px*.

### 4.5. Semântica dos mundos possíveis

Embora não seja possível, por motivos de espaço, desenvolver detalhadamente este item, pelo menos algumas informações básicas merecem ser apresentadas aqui. Como sabemos que não há tabelas-verdade binárias para os operadores *L* e *M*, temos de apelar ao conjunto W dos mundos possíveis para avaliar proposições modais. Este conjunto pode ser definido como W = {W1, W2, ... Wn}, em que cada Wi é um mundo possível. Usando a teoria dos modelos, cada sentença é avaliada num mundo possível Wi, pertencente a W, podendo ter valores-verdade diferentes em mundos diferentes. Desse modo:

v(p, Wi) = V sse p ocorre em Wi [leia-se: *o valor-verdade de p no mundo Wi é V sse p ocorre em Wi*].

v(~p, Wi) = V sse não ocorre p em Wi.

v(p ⊃ q, Wi) = V sse v(p, Wi) = F ou v(q, Wi) = V.

Para o operador da necessidade, estabelece-se a seguinte condição:

v(Lp, Wi) = V sse para todo Wk pertencente a W, v(p, Wk) = V. Em outras palavras, *é necessário que p* é verdadeira num mundo Wi sse ela é verdadeira em todos os mundos possíveis que constituem o conjunto W. Convém notar aqui o paralelismo entre o operador modal da necessidade e o quantificador universal.

Para o operador da possibilidade, estabelece-se a seguinte condição:

Como *Mp ≡ ~L~p*, segue-se que: v(Mp, Wi) = V sse existe pelo menos um mundo Wk pertencente a W no qual v(p, Wk) = V, ou seja, *é possível que p* é verdadeira num mundo Wi se ela é verdadeira em pelo menos um dos mundos possíveis que constituem o conjunto W. Aqui também é conveniente notar o paralelismo entre o operador da possibilidade e o quantificador existencial.

Um argumento é dito *S5-válido* para um conjunto W de mundos possíveis quando cada avaliação que atribui o valor *V* às premissas do argumento num mundo Wi pertencente a W também atribui o valor *V* à conclusão do argumento no mundo Wi. O sistema S5 satisfaz à condição de S5-validade.

### 4.6. Outros sistemas de lógica modal

A lógica modal tem sido utilizada atualmente por lógicos, filósofos, físicos e estudiosos da inteligência artificial.

O interesse dos lógicos pelas noções modais está em que, através delas, eles podem tentar esclarecer noções como *possibilidade lógica* e *necessidade lógica*. Aqui, temos um enfoque *alético* da lógica modal, em que o ponto de vista enfatizado é a noção de *verdade*.

O interesse de alguns filósofos pelas noções modais está relacionado com a semântica, que, a partir dos trabalhos seminais de Kripke nesta área, têm caminhado em direção a uma autêntica metafísica dos mundos possíveis. Neste setor, os trabalhos de David Lewis são talvez os mais relevantes atualmente.

O interesse de outros filósofos está relacionado com as noções de *conhecimento* e *crença*. A lógica que trata das aplicações dos princípios modais ao conhecimento é conhecida como *epistêmica*, enquanto aquela que trata da opinião é conhecida como *doxástica*.

Outros filósofos se interessam pelas aplicações das noções modais à ética, desenvolvendo sistemas de *lógica deôntica*.

O interesse dos físicos e dos estudiosos da inteligência artificial está relacionado com o tempo. A partir daí, desenvolveu-se uma aplicação da lógica modal conhecida como *lógica temporal*.

Um dos primeiros autores a desenvolver aplicações da lógica modal foi G. H. von Wright, em seu livro *An Essay in Modal Logic*. Ali, ele construiu sistemas envolvendo modalidades *aléticas*, *epistêmicas*, *deônticas* e *existenciais*. Os estudos atuais, contudo, vão um pouco além dos trabalhos iniciais de von Wright. A seguir, apresentaremos, a título de ilustração, tratamentos de modalidades aléticas, epistêmicas e deônticas.

Modalidades aléticas:

Podemos dizer que os sistemas modais considerados até agora são todos eles aléticos, pois estão ligados a questões sobre a noção de *verdade*. Dentre tais sistemas, S5 é geralmente tido como o mais adequado para lidar com as modalidades do *possível* e do *necessário*. Do ponto de vista alético, o que interessa mais é saber se S5 é adequado para avaliar argumentos da linguagem natural que contêm expressões ligadas à necessidade e à possibilidade. Tendo em vista que o sistema S5 já foi apresentado anteriormente, não nos preocuparemos mais com a modalidade alética.

Modalidades epistêmicas:

Se interpretamos *Lp* como *é sabido que p*, temos uma perspectiva epistêmica para a lógica modal. Se, por outro lado, interpretamos *Lp* como *acredita-se que p*, temos uma perspectiva doxástica para a lógica modal. A perspectiva epistêmica nos dá uma lógica do conhecimento, enquanto a perspectiva doxástica nos dá uma lógica da crença. As duas lógicas usam geralmente um índice para o sujeito que sabe ou acredita. Desse modo, *a sabe que p* é simbolizado por $K_a p$ e *a acredita que p*, por $B_a p$. O texto clássico a respeito das lógicas epistêmica e da crença é o de Hintikka, *Knowledge and Belief* (1962). Apresentaremos aqui sucintamente o sistema epistêmico proposicional S5, proposto por Hintikka nesta obra. Este sistema corresponde ao sistema proposicional S5, já visto, com algumas modificações. As principais delas são relativas à introdução de dois operadores epistêmicos, $K_x$ e $P_x$. O primeiro corresponde a *x tem conhecimento de que* e o segundo, a *x não tem conhecimento de que não*, de acordo com a definição abaixo:

$P_x A =df \sim K_x \sim A$ ($P_x$ = x não tem conhecimento de que A não é o caso).[5]

Outra modificação está na regra de necessidade epistêmica (rne), que substitui a regra de introdução da necessidade:

de $A$

pode-se concluir $K_x A$ (se $A$ é teorema, então x tem conhecimento de que A).

Os axiomas de S5 epistêmico são os seguintes:

K1: $K_x(A \supset B) \supset [K_x A \supset K_x B]$ (se x tem conhecimento de que $A \supset B$, então: se x tem conhecimento de que A, x tem conhecimento de que B).

K2: $K_x A \supset A$ (se x tem conhecimento de que A, então A).

K3: $K_x A \supset K_x K_x A$ (se x tem conhecimento de que A, então x tem conhecimento de que x tem conhecimento de que A).

K4: $K_x K_y A \supset K_x A$ (se x tem conhecimento de que y tem conhecimento de que A, então x tem conhecimento de que A).

K5: $P_x A \supset K_x P_x A$ (se x não tem conhecimento de que A não é o caso, então x tem conhecimento de que x não tem conhecimento de que A não é o caso).

Exemplos de teoremas da lógica epistêmica S5 de Hintikka:

T1: $\sim A \supset K_x \sim K_x A$ (se A não ocorre, então x tem conhecimento de que que x não tem conhecimento de que A ocorre).

T2: $\sim K_x A \supset K_x \sim K_x A$ (se x não tem conhecimento de que A ocorre, então x tem conhecimento de que x não tem conhecimento de que A ocorre).

As proposições do sistema proposicional S5 epistêmico de Hintikka podem ser demonstradas com o auxílio das regras de dedução natural, devidamente adaptadas.[6] Basta substituir, nas regras de 13 a 16, anteriormente apresentadas, os operadores modais *L* e *M* respectivamente pelos operadores epistêmicos $K_x$ e $P_x$.

13') Regra da eliminação da necessidade epistêmica (abreviatura: *ene*):

de $K_xA$

pode-se concluir *A*

14') Regra da introdução da necessidade epistêmica (abreviatura: *ine*):

de *A*

pode-se concluir $K_xA$

15') Regra da introdução da possibilidade epistêmica (abreviatura: *ipe*):

de *A*

pode-se concluir $P_xA$

16') Regra de reiteração epistêmica (abreviatura: *re*):

de $K_xA$ ou de $P_xA$

pode-se concluir, respectivamente, $K_xA$ ou $P_xA$

Convém lembrar aqui que esta adaptação foi feita para o caso do cálculo proposicional epistêmico. Para um cálculo epistêmico de predicados, outras adaptações seriam necessárias, mas este aspecto não será considerado aqui. Vejamos agora como as proposições acima podem ser demonstradas com o uso das regras adaptadas.[7]

1º Exemplo: demonstrar que $K_xA \supset A$ (K2).

| Etapa | Justificativa |
|---|---|
| 1. $K_xA$ | suposição inicial |
| 2. A | 1 ene |
| 3. $K_xA \supset A$ | 1-2 dc |

[5] Em inglês, o símbolo *Kx* é lido como *x knows*. Estamos traduzindo *x knows* por *x tem o conhecimento de que* para mitigar o tom paradoxal dos axiomas e teoremas da lógica epistêmica quando traduzidos para o português.

[6] Esta adaptação é de minha autoria.

[7] Convém lembrar aqui que, pelo método da dedução natural, os axiomas também devem ser demonstrados.

2º Exemplo: demonstrar que $K_xA \supset K_xK_xA$ (K3).

| Etapa | Justificativa |
|---|---|
| 1. $K_xA$ | suposição inicial |
| 2. $K_xK_xA$ | 1 ine |
| 3. $K_xA \supset K_xK_xA$ | 1-2 dc |

3º Exemplo: demonstrar que $K_xK_yA \supset K_xA$ (K4).

| Etapa | Justificativa |
|---|---|
| 1. $K_xK_yA$ | suposição inicial |
| 2. $K_yA$ | 1 ene |
| 3. A | 2 ene |
| 4. $K_xA$ | 3 ine |
| 5. $K_xK_yA \supset K_xA$ | 1-4 dc |

4º Exemplo: demonstrar que $P_xA \supset K_xP_xA$ (K5).

| Etapa | Justificativa |
|---|---|
| 1. $P_xA$ | suposição inicial |
| 2. $K_xP_xA$ | 1 ine |
| 3. $P_xA \supset K_xP_xA$ | 1-2 dc |

5º Exemplo: demonstrar que $\sim A \supset K_x\sim K_xA$ (T1).

| Etapa | Justificativa |
|---|---|
| 1. $\sim A$ | suposição inicial |
| 2. $P_x\sim A$ | 1 ipe |
| 3. $\sim K_xA$ | 2 eq |

| | |
|---|---|
| 4. $K_x{\sim}K_xA$ | 3 ine |
| 5. ${\sim}A \supset K_x{\sim}K_xA$ | 1-4 dc |

6º Exemplo: demonstrar que ${\sim}K_xA \supset K_x{\sim}K_xA$ (T2).

| Etapa | Justificativa |
|---|---|
| 1. ${\sim}K_xA$ | suposição inicial |
| 2. $K_x{\sim}K_xA$ | 1 ine |
| 3. ${\sim}K_xA \supset K_x{\sim}K_xA$ | 1-2 dc |

7º Exemplo: demonstrar que $K_x(A \supset B) \supset [K_xA \supset K_xB]$ (K1).

| Etapa | Justificativa |
|---|---|
| 1. $K_x(A \supset B) \supset [K_xA \supset K_xB]$ | hipótese |
| 2. ${\sim}\{K_x(A \supset B) \supset [K_xA \supset K_xB]\}$ | negação da hipótese |
| 3. ${\sim}{\sim}\{K_x(A \supset B) \,\&\, {\sim}[K_xA \supset K_xB]\}$ | 2 eq |
| 4. $K_x(A \supset B) \,\&\, {\sim}[K_xA \supset K_xB]$ | 3 eq |
| 5. $K_x(A \supset B)$ | 4 s |
| 6. $A \supset B$ | 5 ene |
| 7. ${\sim}[K_xA \supset K_xB]$ | 4 s |
| 8. ${\sim}{\sim}[K_xA \,\&\, {\sim}K_xB]$ | 7 eq |
| 9. $K_xA \,\&\, {\sim}K_xB$ | 8 eq |
| 10. $K_xA$ | 9 s |
| 11. ${\sim}K_xB$ | 9 s |
| 12. $A$ | 10 ene |
| 13. $B$ | 6, 12 mp |
| 14. $K_xB$ | 13 ine |
| 15. $K_xB \,\&\, {\sim}K_xB$ | 11, 14 c |
| 16. $K_x(A \supset B) \supset [K_xA \supset K_xB]$ | 1-15 ra |

A lógica epistêmica envolve uma série de problemas, dos quais talvez o mais interessante seja o relativo às conexões entre *conhecimento* e *inatismo*. As questões que se colocam aqui são duas: a primeira diz respeito à suposição de que todo sujeito cognoscente conhece automaticamente as leis da lógica e a segunda, à suposição de que todo sujeito cognoscente conhece automaticamente todas as conseqüências de seu conhecimento. Estas questões se referem àquilo que tem sido denominado *onisciência epistêmica*. Há três tipos de onisciência: a lógica, a dedutiva e a fatual.

A *onisciência lógica* pode ser de dois tipos: *forte* e *fraca*. A *onisciência lógica em sentido forte* ocorre quando o sujeito cognoscente é caracterizado pelo sistema de lógica epistêmica de um modo tal que ele conhece todas as verdades lógicas envolvidas pelo sistema epistêmico. As lógicas que atribuem este tipo de onisciência lógica ao sujeito cognoscente defendem o que se convencionou chamar de *tese da onisciência lógica em sentido forte*. A *onisciência lógica em sentido fraco* ocorre quando o sujeito cognoscente conhece pelo menos todas as verdades da lógica de primeira ordem. As lógicas que atribuem esse tipo de onisciência lógica ao sujeito cognoscente defendem o que se convencionou chamar de *tese da onisciência lógica em sentido fraco*. A onisciência lógica em sentido fraco está contida na onisciência lógica em sentido forte: se um sujeito cognoscente possui onisciência lógica em sentido forte, então ele também possui onisciência lógica em sentido fraco, mas não conversamente. Descartes é considerado um dos defensores da onisciência lógica em sentido fraco. Com efeito, ele acreditava que há certas verdades eternas que habitam nossas mentes. Em virtude disso, os sujeitos cognoscentes que possuem onisciência lógica em sentido fraco são também denominados *sujeitos cartesianos*.

A *onisciência dedutiva* ocorre quando o sujeito cognoscente é caracterizado pela lógica epistêmica de um modo tal que ele conhece automaticamente todas as conseqüências lógicas daquilo que sabe. Neste caso, o sujeito é *dedutivamente onisciente*. As lógicas que atribuem onisciência dedutiva ao sujeito cognoscente defendem a *tese da onisciência dedutiva*.

A *onisciência fatual* ocorre quando o sujeito cognoscente é caracterizado pela lógica epistêmica de um modo tal que, para qualquer proposição *p*, ele automaticamente sabe se *p* é o caso ou não. Nesta situação, o sujeito cognoscente é *fatualmente onisciente*. A onisciência fatual inclui as anteriores, o que torna o sujeito cognoscente uma verdadeira divindade. Por esse motivo, não há lógicas que atribuam onisciência fatual aos sujeitos cognoscentes.

A título de ilustração, consideremos o sujeito cognoscente do sistema S5 de lógica epistêmica. De acordo com a regra de necessidade epistêmica, se *A* é um teorema de S5, então o sujeito cognoscente sabe que A. O axioma K3 nos diz que, se o sujeito cognoscente sabe que A, então ele sabe que sabe que A. Isto significa que, em S5, o sujeito cognoscente é *logicamente onisciente em sentido forte*. O axioma K1, por sua vez, afirma que, se o sujeito cognoscente sabe que A ⊃ B, então, se ele sabe que A, ele sabe que B. Aqui, o sujeito cognoscente é capaz de extrair todas as conclusões que se seguem daquilo que ele sabe. Em outras palavras, o sujeito cognoscente é também *dedutivamente onisciente* no sistema S5 epistêmico. O teorema T2 afirma que, se o sujeito cognoscente não tem conhecimento de que A ocorre, então ele tem conhecimento de que não sabe que A ocorre. Como se pode ver, o sistema S5 de lógica epistêmica de Hintikka está construído de maneira tal que o sujeito cognoscente conhece conscientemente todas as conseqüências de seu conhecimento, todos os teoremas que constituem a estrutura lógica deste conhecimento e sabe também o que não conhece. Isto equipararia o sujeito cognoscente a um Deus onisciente, coisa que nenhum de nós teria condições de ser.

Modalidades deônticas:

Se interpretamos *Lp* como *é obrigatório que p*, teremos uma perspectiva deôntica para a lógica modal. Neste caso, *Mp* seria interpretado como *é permitido que p*. De acordo com alguns lógicos, os operadores modais deônticos operam sobre *ações* e não sobre *proposições*. Neste caso, *Lp* deve ser entendido como *é obrigatório fazer p*, e *Mp*, como *é permitido fazer p*, sendo *p* o nome de um ato e não a descrição de um estado de coisas. Não nos posicionaremos quanto a esta questão, assumindo que nossa exposição sobre a lógica deôntica é passível de ser interpretada de qualquer um dos dois modos acima.

Num artigo que se tornou clássico, intitulado *Deontic Logic* (1951), von Wright apresentou um sistema de lógica deôntica que pode ser caracterizado como segue.

Ao invés de *L* e *M*, são utilizados os operadores *O* e *P*. O primeiro deles significa *é obrigatório que* e o segundo, *é permitido que*. As relações entre ambos são dadas pela definição:

Op =df ~P~p (*é obrigatório que p* equivale a *não é permitido que não p*).

Seguem abaixo os princípios em que se baseia von Wright para construir a lógica deôntica:

Princípio de permissão: ou *p* é permitida ou *~p* é permitida (em símbolos: *Pp v P~p*).

Princípio de distribuição deôntica: é permitido ou *p* ou *q* sse ou *p* é permitido ou *q* é permitido (em símbolos: *P(p v q)≡ (Pp v Pq)*).

Princípio da contingência deôntica: as expressões *O(p v ~p)* e *~P(p & ~p)* não são sempre válidas. Em outras palavras, do ponto de vista deôntico, não há a obrigatoriedade de *p ou ~p* nem a proibição de *p & ~p*.

Princípio da extensionalidade: se *p* e *q* são logicamente equivalentes, então *Pp* e *Pq* são também logicamente equivalentes. Este princípio permite a substituição de expressões materialmente equivalentes.

Há outros aspectos que devem ser levados em conta na construção de uma lógica deôntica. Em primeiro lugar, não podemos afirmar que algo é verdadeiro só porque é obrigatório. Isto significa que uma expressão como *Op ⊃ p*, que corresponde ao princípio modal *Lp ⊃ p*, não é válida em lógica deôntica. Em segundo lugar, as iterações de operadores não são adequadas em lógica deôntica. Assim, uma expressão como *OOOp*, por exemplo, não corresponde a uma fbf, e diversos teoremas modais deixam de ser válidos aqui, como, por exemplo, *OOp ⊃ Op*, *OPp ⊃ Pp*, etc. Isto fica mais claro se interpretarmos *p* como nome de uma ação. Em terceiro lugar, as expressões que combinam proposições deônticas com não-deônticas também não são válidas. Desse modo, expressões como *p ⊃ Pq* e *p v Oq* não são válidas aqui. Estas limitações reduzem as proposições do sistema deôntico a um subconjunto das fórmulas válidas do cálculo modal não-deôntico.

Exemplos de fórmulas válidas do cálculo de von Wright:

O(p & q) ≡ (Op & Oq)

(Op v Oq) ⊃ O(p v q) [obs.: a conversa não é válida]

$P(p \& q) \supset (Pp \& Pq)$ [obs.: corresponde a *L(p & q) ⊃ (Lp & Lq)*, mas a conversa não é válida]

$[Op \& O(p \supset q)] \supset Oq$

$[Pp \& O(p \supset q)] \supset Pq$

$[{\sim}Pq \& O(p \supset q)] \supset {\sim}Pp$

$O[p \supset (q \vee r) \& {\sim}Pq \& {\sim}Pr] \supset {\sim}Pp$

${\sim}[O(p \vee q) \& {\sim}Pp \& {\sim}Pq]$

$\{Op \& O[(p \& q) \supset r]\} \supset O(q \supset r)$

$O({\sim}p \supset p) \supset Op$

As sete últimas expressões dessa lista foram chamadas por von Wright de *leis de compromisso*, pois explicitam o que é válido quando o desempenho da ação *p* compromete alguém a desempenhar a ação *q*.

A lógica deôntica está sujeita a paradoxos, como, por exemplo,

$Op \supset O(p \vee q)$.

Essa expressão é válida em lógica deôntica e é conhecida como o *paradoxo de Ross*, pois foi descoberta por este autor em 1941. Uma de suas traduções possíveis seria: *se devo pagar minhas dívidas, então devo ou pagar minhas dívidas ou jogar fora o dinheiro*. Esta afirmação é totalmente contra-intuitiva, mas válida em lógica deôntica, assim como a seguinte: *se posso estacionar aqui, então posso ou estacionar aqui ou roubar um carro*. Esta última proposição tem a forma simbólica

$Pp \supset P(p \vee q)$.

Estes paradoxos parecem ter origem nas propriedades peculiares da implicação material, aqui representada por $\supset$, que não parece expressar adequadamente a relação de implicação utilizada na linguagem natural. Isto fica mais claro se considerarmos o paradoxo abaixo:

$O{\sim}p \supset O(p \supset q)$.

Em outras palavras, se é obrigatório fazer *~p*, ou seja, se é proibido fazer *p*, então é obrigatório que, ao fazer *p*, também se faça *q*. Como *q* é qualquer proposição (ou ação), podemos concluir que, se é proibido fazer *p*, então é obrigatório que, ao fazer *p*, também se faça qualquer outra coisa.

As proposições do sistema deôntico de von Wright podem ser demonstradas com o auxílio das regras de dedução natural, devidamente adaptadas para este fim.[8] As onze primeiras podem permanecer exatamente como estão, mas as regras de 12 a 16 devem ser modificadas e acrescidas para o caso específico dos operadores deônticos:

13') Regra de passagem da obrigação para a permissão (abreviatura: *op*):
de *Op*
pode-se concluir *Pp*.
14') Regra da distribuição da permissão na conjunção (abreviatura: *dpc*):
de *P(p & q)*
pode-se concluir *Pp & Pq*.
15') Regra de distribuição da permissão na disjunção (abreviatura: *dpd*):
de *P(p v q)*
pode-se concluir *Pp v Pq*.
16') Regra de fatoração da permissão na disjunção (abreviatura: *fpd*):
de *Pp v Pq*
pode-se concluir *P(p v q)*.
17') Regra de fatoração da obrigação na disjunção (abreviatura: *fod*):
de *Op v Oq*
pode-se concluir *O(p v q)*.

Vejamos alguns exemplos de demonstrações em lógica deôntica com o uso das regras adaptadas de dedução natural.

1º Exemplo: demonstrar que O(p & q) ⊃ (Op & Oq).

[8] Esta adaptação também é de minha autoria.

| Etapa | Justificativa |
|---|---|
| 1. O(p & q) | suposição inicial |
| 2. O~(~p v ~q) | 1 eq |
| 3. ~P~~(~p v ~q) | 2 eq |
| 4. ~P(~p v ~q) | 3 dpd |
| 5. ~(P~p v P~q) | 4 eq |
| 6. ~~(~P~p & ~P~q) | 5 eq |
| 7. ~P~p & ~P~q | 6 eq |
| 8. Op & Oq | 7 eq |
| 9. O(p & q) ⊃ (Op & Oq) | 1-8 dc |

2º Exemplo: demonstrar que (Op & Oq) ⊃ O(p & q).

| Etapa | Justificativa |
|---|---|
| 1. Op & Oq | suposição inicial |
| 2. ~(~Op v ~Oq) | 1 eq |
| 3. ~(P~p v P~q) | 2 eq |
| 4. ~[P(~p v ~q)] | 3 fpd |
| 5. ~P(~p v ~q) | 4 eq |
| 6. O~(~p v ~q) | 5 eq |
| 7. O(p & q) | 6 eq |
| 8. (Op & Oq) ⊃ O(p & q) | 1-7 dc |

3º Exemplo: demonstrar que (Op v Oq) ⊃ O(p v q).

| Etapa | Justificativa |
|---|---|
| 1. Op v Oq | suposição inicial |
| 2. O(p v q) | 1 fod |
| 3. (Op v Oq) ⊃ O(p v q) | 1-2 dc |

4º Exemplo: demonstrar que P(p & q) ⊃ (Pp & Pq).

| Etapa | Justificativa |
|---|---|
| 1. P(p & q) | suposição inicial |
| 2. Pp & Pq | 1 dpc |
| 3. P(p & q) ⊃ (Pp & Pq) | 1-2 dc |

5º Exemplo: demonstrar que [Op & O(p ⊃ q)] ⊃ Oq.

| Etapa | Justificativa |
|---|---|
| 1. [Op & O(p ⊃ q)] ⊃ Oq | hipótese |
| 2. ~{[Op & O(p ⊃ q)] ⊃ Oq} | negação da hipótese |
| 3. [Op & O(p ⊃ q)] & ~Oq | 2 eq |
| 4. [Op & O(p ⊃ q)] | 3 s |
| 5. ~Oq | 3 s |
| 6. Op | 4 s |
| 7. O(p ⊃ q) | 4 s |
| 8. P~q | 5 eq |
| 9. Pp | 6 op |
| 10. Pp & P~q | 8,9 c |
| 11. O~(p & ~q) | 7 eq |
| 12. ~P(p & ~q) | 11 eq |
| 13. ~(Pp & P~q) | 12 dpc |
| 14. (Pp & P~q) & ~(Pp & P~q) | 10, 13 c |
| 15. [Op & O(p ⊃ q)] ⊃ Oq | 1-14 ra |

### 4.7. Exercícios de Lógica Modal

4.7.1. Usando o operador modal da possibilidade, *M*, simbolizar as seguintes sentenças (convenção: João lê = p; é contingente que p = é possível que não p):
4.7.1.1. É possível que João leia.
4.7.1.2. É contingente que João leia.
4.7.1.3. É necessário que João leia.
4.7.1.4. É impossível que João leia.
4.7.1.5. Se João lê, então João lê [no sentido da implicação estrita].

4.7.2. Usando o operador modal da necessidade, *L*, simbolizar as sentenças acima.

4.7.3. Considere as seguintes proposições modais abaixo. Construa o quadrado lógico da oposição a partir de cada uma, expressando a contrária, a contraditória, a subalterna e a subcontrária de cada quadrado com o auxílio exclusivo do operador modal usado em cada uma delas.

| 4.7.3.1. é necessário que chova | 4.7.3.2. é possível que chova |
|---|---|
| | |

4.7.4. Supondo que cada uma das proposições modais acima é verdadeira, o que se pode afirmar da verdade ou falsidade das demais proposições do quadrado?

4.7.5. Supondo que cada uma das proposições modais acima é falsa, o que se pode afirmar da verdade ou falsidade das demais proposições do quadrado?

4.7.6. Seja a proposição modal ‘é necessário que Deus exista’. Expresse-a com o auxílio: a) da possibilidade; b) da contingência; c) da impossibilidade.

4.7.7. Nas expressões abaixo, troque o operador modal da possibilidade, *M*, pelo da necessidade, *L*:

4.7.7.1. ~M~(p & ~q): ________________________________________
4.7.7.2. M(p v q) : ________________________________________
4.7.7.3. ~M(p & q) : ________________________________________
4.7.7.4. M~(p & q) : ________________________________________
4.7.7.5. ~M~p & Mq: ________________________________________

4.7.8. Mantendo 'p' e 'q', traduza as expressões do exercício anterior para o português.

4.7.8.1. ~M~(p & ~q): ____________________________________________________
____________________________________________________________________
4.7.8.2. M(p v q) : ____________________________________________________
____________________________________________________________________
4.7.8.3. ~M(p & q) : ____________________________________________________
____________________________________________________________________
4.7.8.4. M~(p & q) : ____________________________________________________
____________________________________________________________________
4.7.8.5. ~M~p & Mq: ____________________________________________________
____________________________________________________________________

4.7.9. Nas expressões abaixo, troque o operador modal da necessidade, *L*, pelo possibilidade, *M*:

4.7.9.1. ~L~(p & ~Lq) : ________________________________________
47.9.2. L(p v Lq) : ________________________________________
4.7.9.3. ~L(p & q) : ________________________________________
4.7.9.4. L~(p & q) : ________________________________________
4.7.9.5. ~L~p & Lq: ________________________________________

4.7.10. Traduza as expressões do exercício anterior para o português.

4.7.11. Demonstrar os seguintes teoremas no sistema modal K, usando as regras de dedução natural:

4.7.11.1. K2: $(Lp \,\&\, Lq) \supset L(p \,\&\, q)$

4.7.11.2. K3: $L(p \,\&\, q) \equiv (Lp \,\&\, Lq)$

4.7.12. Demonstrar os seguintes teoremas no sistema T, usando as regras de dedução natural:

4.7.12.1. T1: $p \supset Mp$

4.7.12.2. T2: $M(p \supset Lp)$

4.7.13. Demonstrar o seguinte teorema no sistema D, usando as regras de dedução natural:

4.7.13.1. $M(p \supset p)$

4.7.14. Demonstrar os seguintes teoremas no sistema S4, usando as regras de dedução natural:

4.7.14.1. $LLp \supset Lp$

4.7.14.2. $Lp \equiv LLp$

4.7.14.3. $MMp \supset Mp$

4.7.15. Demonstrar os seguintes teoremas no sistema S5, usando as regras de dedução natural:

4.7.15.1. $MLp \supset Lp$

4.7.15.2. $Mp \equiv LMp$

4.7.15.3. $[L(p \supset q) \,\&\, {\sim}Mq] \supset {\sim}p$

4.7.15.4. $L(p \vee Lq) \supset (Lp \vee Lq)$

4.7.15.5. $L(p \vee Mq) \supset (Lp \vee Mq)$

4.7.15.6. $M(p \vee Mq) \supset (Mp \vee Mq)$

4.7.16. Demonstrar os seguintes teoremas do cálculo modal de predicados S5, utilizando o método de dedução natural:

4.7.16.1. $M(\exists x)Px \supset (\exists x)MPx$

4.7.16.2. $M(x)Px \supset (x)MPx$

4.7.16.3. $(x)(Fx \supset LGx) \supset [(x)Fx \supset L(x)Gx]$

4.7.16.4. $(x)L(Fx \supset Gx) \supset L[(x)Fx \supset (x)Gx]$

## 5. Lógica Trivalente[9]

### 5.1. Caracterização Geral

A **Lógica Trivalente** surge quando a bivalência da lógica fregiana é colocada em questão. Isso acontece a partir do momento em que Lukasievicz tenta encontrar uma forma adequada de estudar os *futuros contingentes*. Uma sentença como *amanhã vai chover* ainda não é nem verdadeira nem falsa no momento de sua enunciação. Caso contrário, tudo já estaria necessariamente determinado antes de acontecer. Isso se chocaria não só com nossa idéia de liberdade, mas também com a noção de que o evento descrito pela sentença *amanhã vai chover* é contingente e não necessário. Para trabalhar com esse tipo de sentença, Lukasiewicz desenvolveu uma **Lógica Trivalente**, em que o valor ½ é acrescentado aos valores *0* e *1*. A tabela abaixo mostra os valores-verdade das principais conectivas da lógica trivalente de Lukasiewicz:

[9] As informações desta parte foram extraídas de Blanché, R. *Introduction à la Logique Contemporaine*. 2 ed. Paris: Lib. Armand Colin, 1957.

| p | q | ~p | ~q | p v q | p & q | p ⊃ q | p ≡ q |
|---|---|---|---|---|---|---|---|
| 1 | 1 | 0 | 0 | 1 | 1 | 1 | 1 |
| 1 | ½ | 0 | ½ | 1 | ½ | ½ | ½ |
| 1 | 0 | 0 | 1 | 1 | 0 | 0 | 0 |
| ½ | 1 | ½ | 0 | 1 | ½ | 1 | ½ |
| ½ | ½ | ½ | ½ | ½ | ½ | 1 | 1 |
| ½ | 0 | ½ | 1 | ½ | 0 | ½ | ½ |
| 0 | 1 | 1 | 0 | 1 | 0 | 1 | 0 |
| 0 | ½ | 1 | ½ | ½ | 0 | 1 | ½ |
| 0 | 0 | 1 | 1 | 0 | 0 | 1 | 1 |

Como se pode ver, para duas sentenças, *p* e *q*, a tabela envolve nove combinações possíveis de valores-verdade, diferentemente das quatro combinações possíveis da lógica binária fregiana.

### 5.2. Regras para atribuição de valores em lógica trivalente

Supondo que *x* e *y* sejam os valores-verdade de *p* e *q*, respectivamente, as regras para calcular os valores de cada coluna, a partir da terceira, são as seguintes.

<u>Regra para a negação</u>: se o grau de verdade duma sentença diminui, podemos supor intuitivamente que o valor correspondente de sua negação aumenta e vice-versa. Neste caso, se *p* vale *x*, a negação de *p* vale $1 - x$. Assim, quando *p* tem o valor *1*, sua negação vale $1 - 1 = 0$; quando *p* tem o valor *0*, sua negação vale $1 - 0 = 1$; quando *p* tem o valor ½, sua negação vale $1 - ½ = ½$. O mesmo se aplica a *q*.

<u>Regra para a disjunção</u>: seguindo o espírito da lógica bivalente, podemos dizer que o valor-verdade da articulação de duas sentenças através da conectiva *ou* é determinado por aquela que apresenta o maior grau de verdade. Desse modo, quando *p* vale *1* e *q* vale *1*, *p v q* vale *1*; quando *p* vale *1* e *q* vale ½, *p v q* vale *1*; quando *p* vale *1* e *q* vale *0*, *p v q* vale *1*; e assim sucessivamente. Isso corresponde à função matemática *Max(x, y)*, que escolhe o maior dentre dois números *x* e *y*.

<u>Regra para a conjunção</u>: seguindo o espírito da lógica bivalente, podemos dizer que o valor-verdade da articulação de duas sentenças através da conectiva *e* é determinado por aquela que apresenta o menor grau de verdade. Desse modo, quando *p* vale *1* e *q* vale *1*, *p & q* vale *1*; quando *p* vale *1* e *q* vale ½, *p & q* vale ½; quando *p* vale *1* e *q* vale *0*, *p & q* vale

*0*; e assim sucessivamente. Isso corresponde à função matemática *Min(x, y,)* que escolhe o menor dentre dois números *x* e *y*.

Regra para a condicional: também podemos seguir aqui o espírito da lógica bivalente. Se $x \leq y$ , o valor de $p \supset q$ é *1*; se $x > y$, então $p \supset q$ vale $1 - (x - y)$ — aqui, *x* e *y* só podem assumir um dentre três valores possíveis: *1*, *½* e *0*. Desse modo, quando *p* vale *1* e *q* vale *1*, $p \supset q$ vale *1*; quando *p* vale *1* e *q* vale *½*, $p \supset q$ vale $1 - (1 - ½) = ½$; quando *p* vale *1* e *q* vale *0*, $p \supset q$ vale $1 - (1- 0) = 0$; e assim sucessivamente. Intuitivamente, esta situação poderia ser descrita da maneira que segue: se o antecedente *p* é menos verdadeiro do que o conseqüente *q*, então o valor de $p \supset q$ é *1*; se o conseqüente *q* é menos verdadeiro do que o antecedente *p*, então o valor de $p \supset q$ não pode ser *1* e seu valor é determinado pelo decréscimo do grau de verdade quando passamos de *p* para *q* — daí termos subtraído de *1* a diferença entre *x* e *y* para determinar o valor de $p \supset q$.

Regra para a equivalência: seguindo ainda o espírito da lógica bivalente, podemos dizer o seguinte: se $x = y$, o valor de $p \Leftrightarrow q$ é 1; se $x \neq y$ e o valor *½* não está envolvido, o valor de $p \equiv q$ é *0*; se $x \neq y$ e o valor *½* está envolvido, o valor de $p \equiv q$ é *½*.

### 5.3. Lógica trivalente, lógica binária e outras lógicas

Uma das conseqüências importantes da lógica trivalente está em que certas sentenças essenciais da abordagem bivalente deixam de ser tautologias. Assim, embora a auto-implicação de uma sentença, representada por $p \supset p$, ainda seja uma lei lógica, o princípio do terceiro excluído, representado por *p v ~p*, e a lei de não-contradição, representada por *~(p & ~p)*, já não possuem validade em todos os casos, como se pode ver pela tabela abaixo:

| p | ~p | p v ~p | p & ~p | ~( p & ~p) | p ⊃ p |
|---|---|---|---|---|---|
| 1 | 0 | 1 | 0 | 1 | 1 |
| ½ | ½ | ½ | ½ | ½ | 1 |
| 0 | 1 | 1 | 0 | 1 | 1 |

Na lógica binária clássica, *p v ~p* e *~( p & ~p)* só apresentariam o valor *1*, enquanto *p & ~p* só teria o valor *0*. Isso significa que o sistema proposto por Lukasiewicz já aponta em direção à lógica paraconsistente, que será considerada mais adiante.

Na lógica trivalente, a maior parte das expressões que servem de base para a demonstração por absurdo deixam de ser tautologias. Mesmo assim, essas expressões jamais apresentam o valor *0* nas tabelas correspondentes, oscilando entre os valores *1* e *½*. Todas as fórmulas que são válidas nessa lógica também se revelam válidas no cálculo bivalente tradicional.

Neste ponto, cabe observar que é possível dar um tratamento extensional à lógica modal através de tabelas trivalentes. Com efeito, a adoção de pelo menos mais um valor pode permitir o preenchimento das lacunas nas tabelas de valores-verdade das expressões modais. O quadro abaixo mostra uma tabela trivalente para os operadores modais da necessidade e da possibilidade:

| p | Lp | Mp | L~p | M~p |
|---|---|---|---|---|
| 1 | ½ | 1 | 0 | ½ |
| ½ | ½ | ½ | ½ | ½ |
| 0 | 0 | ½ | ½ | 1 |

Isso sugere a possibilidade de efetuarmos uma aproximação entre a lógica modal e a trivalente.

A proposta de Lukasiewicz constitui uma primeira instância das chamadas *lógicas plurivalentes*. De fato, uma vez contestada a bivalência, podemos adotar não só três valores-verdade, mas quatro, cinco, ou quantos quisermos. Basta construir uma tabela com n valores-verdade para se obter uma lógica n-valente, sendo n tão grande quanto se queira. Isso permite concluir que há uma grande variedade de lógicas plurivalentes e que as tautologias de muitas delas não constituem um subconjunto das tautologias da lógica bivalente tradicional. E convém acrescentar que os sistemas não-clássicos podem ter tratamento extensional através de tabelas plurivalentes.

### 5.4. Exercícios de Lógica Trivalente

5.4.1. Com base na tabela de valores-verdade para o cálculo trivalente de Lukasievicz, determine os valores-verdade das expressões abaixo:

| | |
|---|---|
| 5.4.1.1. $p \supset (p \vee q)$ | 5.4.1.6. $(p \,\&\, (p \supset q)) \supset q$ |
| 5.4.1..2. $(p \,\&\, q) \supset p$ | 5.4.1.7. $(\sim q \,\&\, (p \supset q)) \supset \sim p$ |
| 5.4.1.3. $\sim\sim p \supset p$ | 5.4.1.8. $\sim p \supset (p \supset q)$ |
| 5.4.1.4. $p \supset \sim\sim p$ | 5.4.1.9. $\sim p \supset (p \supset \sim q)$ |
| 5.4.1.5. $(p \,\&\, \sim p) \supset q$ | 5.4.1.10. $p \supset (\sim p \supset q)$ |

5.4.2. Ainda com base na tabela de valores-verdade para o cálculo trivalente de Lukasievicz e com base na convenção abaixo, determine os valores-verdade das sentenças complexas que seguem [convenção: amanhã choverá = ½; hoje faz sol = 1; hoje está quente = 1]:

5.4.2.1. Hoje faz sol e está quente.
5.4.2.2. Hoje faz sol ou hoje está quente.
5.4.2.3. Se hoje faz sol, então hoje faz sol.
5.4.2.4. Não é o caso que hoje faz sol e hoje não faz sol ao mesmo tempo.
5.4.2.5. Se amanhã choverá, então amanhã choverá.
5.4.2.6. Amanhã choverá ou amanhã não choverá.
5.4.2.7. Não é o caso que amanhã choverá e amanhã não choverá ao mesmo tempo.
5.4.2.8. Se hoje faz sol, então amanhã choverá.
5.4.2.9. Se amanhã choverá, então hoje faz sol.
5.4.2.10. Se hoje não está quente, então amanhã choverá.

## 6. Lógica Imprecisa (*fuzzy*)[10]

### 6.1. Caracterização geral

A **Lógica Imprecisa** (*fuzzy*) foi proposta inicialmente por Lotfi Zadeh em 1965, quando ele descreveu a matemática da teoria dos conjuntos imprecisos e, por extensão, da lógica imprecisa. Esta teoria admite um número infinito de valores-verdade entre 0 e 1, trabalhando com domínios em que predomina a *vaguidade*. Tais domínios são bem caracterizados por predicados como *x está bêbado*, *x está sóbrio*, *x é saudável*, *x é doente*, *x é normal*, *x é anormal*, *x é alto*, *x é baixo*, *x é criança*, *x é adulto* etc.

Para ilustrar o que ocorre nestes casos, suponhamos que João esteja bêbado. Se isso é verdade, então ele também estará bêbado um segundo depois. O mesmo acontecerá um segundo depois e mais um segundo depois. Como não é possível determinar o segundo exato em que João deixará de estar bêbado e ficará sóbrio, esse raciocínio pode ser prolongado indefinidamente. Portanto, se João está bêbado, ele ficará neste estado pelo resto de sua vida. O argumento pode ser expresso através de uma seqüência de inferências do tipo *modus ponens*:

João está bêbado no instante $t$.
Se João está bêbado no instante t, então ele está bêbado no instante $t + 1$.
Logo, João está bêbado no instante $t + 1$.

João está bêbado no instante $t + 1$.
Se João está bêbado no instante $t + 1$, então ele está bêbado no instante $t + 2$.
Logo, João está bêbado no instante $t + 2$.

etc. ...

[10] As informações desta parte foram extraídas de Priest, G. *Non-Classical Logic.* Cambridge: Cambridge Un. Press, 2001 e de Kaehler, S. D. *Fuzzy Logic. An Introduction.* Disponível em <http://www.seattlerobotics.org/encoder/mar98/fuz/flindex.html>. Acesso em 12 Jan. 2003.

João está bêbado no instante $t + n - 1$.

Se João está bêbado no instante $t + n - 1$, então João está bêbado no instante $t + n$.

Logo, João está bêbado no instante $t + n$.

Como se pode ver, $n$ pode ser tão grande quanto se queira, permitindo a conclusão de que João ficará bêbado pelo resto de sua vida. Esse argumento, conhecido como *paradoxo do sorites*, parece ter sido formulado pela primeira vez pelo megárico Eubúlides e também pode ser aplicado aos demais predicados anteriormente listados. Ele decorre do fato de que tais predicados apresentam um tipo especial de vaguidade: o sujeito ao qual cada um deles é atribuído pode passar por mudanças muito pequenas segundo a segundo, sem que isso afete a aplicabilidade do predicado. Conforme mencionado antes, é praticamente impossível estabelecer o instante $t + k$ em que João deixa de estar bêbado para se tornar sóbrio no instante seguinte, $t + k + 1$.

### 6.2. Semântica para conectivas e predicados imprecisos

Uma das maneiras de enfrentar o tipo de vaguidade descrito acima é construir uma lógica imprecisa plurivalente em que os valores-verdade mudem de maneira contínua, acompanhando a escala de mudanças infinitesimais, admitida pelos predicados acima. Nessa lógica imprecisa, uma sentença se torna verdadeira através de graus contínuos. Tarski e Lukasiewicz chegaram a construir em 1930 uma lógica plurivalente cujos valores-verdade se distribuem através de um contínuo de mudanças infinitesimais, mas sem a preocupação de resolver os problemas relativos à vaguidade. A fim de construir uma lógica capaz de lidar com esta última, podemos utilizar os números reais entre 0 e 1 para caracterizar o *continuum* de valores-verdade que uma dada sentença pode ter. O exemplo abaixo trabalha com definições imprecisas dos predicados *alto* e *velho*:

v(x é alto) = 0, se altura de x ≤ 1,50 m;

= [altura de x – 1,50] ÷ 0,50, se 1,50 m < altura de x < 2,00 m;

= 1, se altura de x ≥ 2,00 m

v(x é velho) = 0, se idade de x ≤ 18 anos

= [idade de x – 18] ÷ 42, se 18 anos < idade de x < 60 anos

= 1, se idade de x ≥ 60 anos

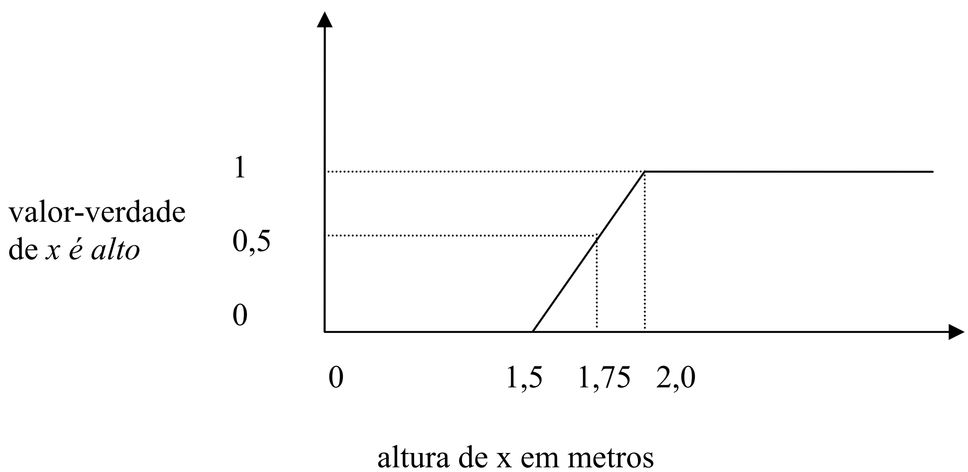
Suponhamos, por exemplo, que Francisco tenha uma altura de 1,90 m. Neste caso, como 1,50 < 1,90 < 2,00, o valor-verdade de *Francisco é alto* deve ser calculado como segue:

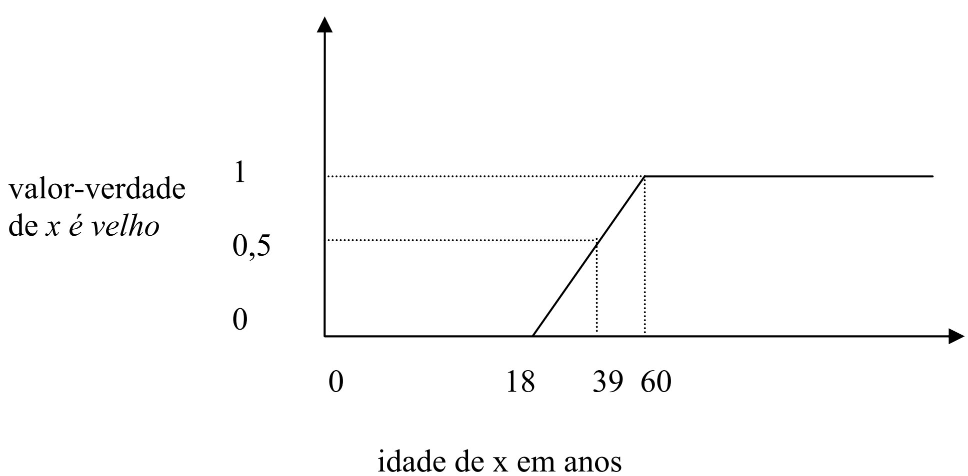
v(Francisco é alto) = [1,90 – 1,50] ÷ 0,50 = 0,40 ÷ 0,50 = 0,80.

Se ele tem 37 anos, então o valor-verdade de *Francisco é velho* deve ser calculado como segue:

v(Francisco é velho) = [37 – 18] ÷ 42 = 19 ÷ 42 = 0,45 (justificativa: 18 < 37 < 60).

Os predicados *x é alto* e *x é velho* podem ser representados pelos gráficos abaixo:



A atribuição de valores-verdade em lógica imprecisa admite também a inclusão de *modificadores* para estes predicados, como, por exemplo, *muito*, *mais ou menos*, etc. Assim, o valor-verdade de *Francisco é muito alto* poderia ser definido como a raiz quadrada do valor-verdade de *Francisco é alto*. No exemplo considerado, teríamos:

v(Francisco é muito alto) = [v(Francisco é alto)]$^{½}$ = (0,80)$^{½}$ = 0,89.

Para *Francisco é muito velho*, teríamos:

v(Francisco é muito velho) = [v(Francisco é velho)]$^{½}$ = (0,45)$^{½}$ = 0,67.

Já o valor-verdade de *Francisco é mais ou menos alto* poderia ser definido como o quadrado do valor-verdade de *Francisco é alto*. No exemplo considerado, isto daria:

v(Francisco é mais ou menos alto) = [v(Francisco é alto)]$^{2}$ = (0,80)$^{2}$ = 0,64.

No caso de *Francisco é mais ou menos velho*, obteríamos:

v(Francisco é mais ou menos velho) = [v(Francisco é velho)]$^{2}$ = (0,45)$^{2}$ = 0,20.

Os valores-verdade das conectivas sentenciais também podem ser calculados em lógica imprecisa. Inspirando-nos nas regras de cálculo da lógica trivalente de Lukasiewicz para estabelecer os valores-verdade duma sentença e generalizando-as, podemos obter as regras abaixo para a lógica fuzzy.

Regra para a negação: se o valor-verdade da sentença *p* é *x*, então o valor de *~p* é *1 – x*. Exemplo: se *p* vale *0,2948672*, então *~p* vale *1 - 0,2948672 = 0,7051328*.

Regra para a conjunção: se os valores-verdade de *p* e *q* são respectivamente *x* e *y*, então o grau de verdade de *p & q* é *Min(x, y)*. Exemplo: se os valores-verdade de *p* e *q* são respectivamente *0,4352984* e *0,6795552*, então o valor de *p & q* é *Min(0,4352984, 0,67955520) = 0,4352984*.

Regra para a disjunção não-exclusiva: o grau de verdade de *p v q* é *Max(x, y)*. Exemplo: se os valores-verdade de *p* e *q* são respectivamente *0,4352984* e *0,6795552*, então o valor de *p v q* é *Max(0,4352984, 0,6795552) = 0, 6795552*.

Regra para a condicional: a) se $x \leq y$, então $p \supset q$ vale *1*; se $x > y$, então $p \supset q$ vale *1 – (x – y) = 1 – x + y*. Exemplo: se os valores de *p* e *q* são respectivamente *0,4352984* e *0,6795552*, então o valor de $p \supset q$ é *1*; se os valores de *p* e *q* são respectivamente *0,8264951* e *0,6795552*, então $p \supset q$ vale *1 - 0, 8264951 + 0,6795552 = 0,8530601*.

Nessa lógica, deixa de ser tautologia o *modus ponens,* segundo o qual da conjunção de $p$ com $p \supset q$ pode-se deduzir $q$. Com efeito, calculemos o valor-verdade de $[p \& (p \supset q)] \supset q$ quando o valor de $p$ é $0,8$ e o de $q$, $0,4$:

$p = 0,8;$

$p \supset q = 1 - 0,8 + 0,4 = 0,6;$

$p \& (p \supset q) = 0,6;$

$[p \& (p \supset q)] \supset q = 1 - 0,6 + 0,4 = 0,8$ (e não $1$, como seria de esperar).

Esse é o preço a ser pago para resolver o paradoxo do sorites. Com efeito, para interromper a série de argumentos baseados no *modus ponens* apresentada acima, é preciso poder estabelecer uma gradação em que a condicional

Se João está bêbado no instante $t$, então ele está bêbado no instante $t + 1$,

representando o início do sorites, seja completamente verdadeira, e a condicional

Se João está bêbado no instante $t + k$, então João está bêbado no instante $t + k + 1$,

representando o final do sorites, seja completamente falsa. Isso significa que os valores-verdade das diversas sentenças condicionais do sorites vão gradativamente decrescendo, partindo do valor $1$ até chegar a $0$. Em virtude da definição da implicação material, representada pela conectiva $\supset$, o valor-verdade do *modus ponens* envolvendo a primeira condicional é $1$. Mas o valor-verdade de cada *modus ponens* após a primeira condicional deixará gradativamente de ser $1$, à medida que cada sentença condicional deixa também de ter o valor $1$.

Segue abaixo uma lista de outras expressões importantes que não são válidas na lógica imprecisa:

$p \vee \sim p$

$\sim(p \& \sim p)$

$(p \& \sim p) \supset q$

$p \supset (q \vee \sim q)$

Além disso, grande parte das propriedades paradoxais do condicional material permanecem na lógica imprecisa. Dentre elas, destacam-se as seguintes:

$p \supset (q \supset p)$

$\sim p \supset (p \supset q)$

$\sim(p \supset q) \supset p$

### 6.3. Exemplo de aplicação da lógica imprecisa

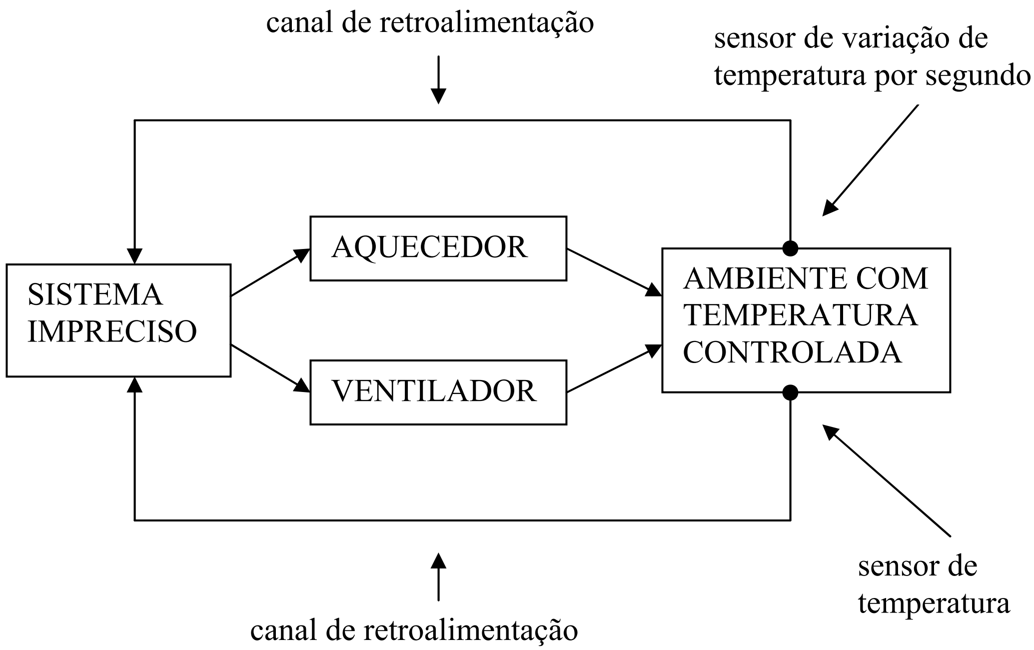
Para elaborar sua teoria, Zadeh levou em conta que, nos seus processos de interação altamente adaptativa com o ambiente, as pessoas não exigem informações precisas de entrada para poderem produzir resultados de saída adequados. Se os equipamentos de controle automático pudessem ser programados para aceitar dados de entrada imprecisos, eles seriam muito mais eficientes e talvez mais fáceis de implementar. A idéia geral da lógica imprecisa é a seguinte: os dados imprecisos de entrada, expressos através de uma terminologia imprecisa na linguagem natural, podem ser transformados em valores numéricos; estes últimos podem ser utilizados em operações matemáticas confiáveis; e os resultados dessas operações também podem ser trazidos de volta à teminologia imprecisa da linguagem natural através de cálculos confiáveis. Como se vê, o processo como um todo admite três etapas básicas:

1ª etapa: tornar impreciso. Isto corresponde a dar uma forma numérica aos dados imprecisos recebidos a título de informação de entrada.

2ª etapa: efetuar inferências imprecisas. Isto corresponde a aplicar as regras de inferência do algoritmo impreciso e efetuar os cálculos matemáticos correspondentes, de modo a processar as informações de entrada.

3ª etapa: recuperar a precisão. Isto corresponde a produzir dados de saída precisos, através, por exemplo, do cálculo das médias dos valores numéricos obtidos na 2ª etapa.

Como exemplo de aplicação da lógica imprecisa, imaginemos um sistema que controle a temperatura de um ambiente e, para tanto, disponha de um ventilador (para resfriar o ambiente), de um aquecedor (para esquentá-lo) e de dois sensores de retroalimentação (um para medir a diferença entre a temperatura desejada e a temperatura medida e outro para medir a taxa de variação da temperatura com o tempo). A figura abaixo ilustra o esquema de funcionamento do sistema impreciso:




Assim, se a temperatura desejada para o ambiente for de 60 ºC e se o ambiente estiver a 50 ºC e esquentando a uma determinada velocidade, o sistema deverá decidir o que fazer: ou ativar o aquecedor, para que a temperatura continue subindo naquela velocidade até atingir o nível desejado, ou ativar o ventilador, para diminuir a velocidade de aquecimento e impedir que o ambiente ultrapasse muito rapidamente o nível desejado, ou deixar as coisas como estão.

Os dados numéricos com os quais o computador impreciso trabalha são os seguintes:

$t_a$: temperatura-alvo, ou seja, temperatura desejada para o ambiente controlado.

$t_m$: temperatura do ambiente controlado, medida pelo sensor num dado instante $\tau_k$.

v: taxa de variação da temperatura medida. Ela é obtida calculando-se a diferença entre a temperatura medida no instante $\tau_2$ ($t_2$) e a temperatura medida no instante $\tau_1$ ($t_1$) e dividindo-se o resultado pelo tempo decorrido entre as medições ($\tau_2 - \tau_1$). Neste caso, $v = (t_2 - t_1) \div (\tau_2 - \tau_1)$. Suponhamos, a título de ilustração, que $t_1$ = 120 ºC, $t_2$ = 130 ºC, $\tau_1$ = 5 s e $\tau_2$ = 10 s. Aqui, v = (130 - 120) ÷ (10 - 5) = 10 ÷ 5 = + 2 ºC/s. O valor positivo obtido significa que a temperatura medida está subindo 2 ºC por segundo e que o ambiente está esquentando. De maneira semelhante, poderemos concluir que, quando v tem valor negativo, o ambiente está esfriando e, quando v tem valor nulo, o ambiente permanece na temperatura desejada, sem mudança.

Os dados numéricos acima podem representar inúmeras situações diferentes, como, por exemplo: o ambiente está quente e sem mudança de temperatura, ou o ambiente está frio e esquentando, ou o ambiente está normal e sem mudança. Estas situações podem se sobrepor umas às outras, gerando imprecisão. Para processar todas essas situações de maneira eficaz, o computador impreciso possui um algoritmo que prevê todas as situações possíveis e o que deve ser feito no caso de cada uma delas. Isto envolve a construção de uma matriz de regras com nove possibilidades, conforme ilustrado abaixo:

| |
|---|
| 1. se o ambiente está quente (Q) e resfriando (R), então o ventilador é acionado (V) |
| 2. se o ambiente está normal (N) e resfriando (R), então o aquecedor é acionado (A) |
| 3. se o ambiente está frio (F) e resfriando (R), então o aquecedor é acionado (A) |
| 4. se o ambiente está quente (Q) e sem mudança (SM), então o ventilador é acionado (V) |
| 5. se o ambiente está normal (N) e sem mudança (SM), então o ambiente é deixado como está (D) |
| 6. se o ambiente está frio (F) e sem mudança (SM), então o aquecedor é acionado (A) |
| 7. se o ambiente está quente (Q) e esquentando (E), então o ventilador é acionado (V) |
| 8. se o ambiente está normal (N) e esquentando (E), então o ventilador é acionado (V) |
| 9. se o ambiente está frio (F) e esquentando (E), então o aquecedor é acionado (A) |

Uma das características centrais da lógica imprecisa é que ela trabalha com todas essas regras ao mesmo tempo, admitindo sobreposições. Supondo, por exemplo, que a temperatura-alvo seja 100 ºC, poderíamos ter as seguintes definições semânticas para os predicados imprecisos utilizados pelo sistema de controle:

v(x está normal) = 1, se $t_m = 100$ ºC

= $1 - (100 - t_m)/2$, se $98 < t_m < 100$

= $1 - (t_m - 100)/2$, se $100 < t_m < 102$

= 0, se $t_m < 98$ ou $t_m > 102$

v(x está quente) = 1, se $t_m \geq 102$

= $1 - (102 - t_m)/2$, se $100 < t_m < 102$

= 0, se $t_m < 100$

v(x está frio) = 1, se $98 \geq t_m$

= 1 - ($t_m$ - 98)/2, se $98 < t_m < 100$

= 0, se $t_m > 100$

v(x está sem mudança) = 1, se v = 0

= (5 - |v|)/5, se 0 < v < +5 ou se –5 < v < 0

= 0, se v ≥ +5 ou se –5 ≥ v

[Obs.: |v| representa o módulo de v]

v(x está esquentando) = 1, se v ≥ +5

= 1 – [(5 - |v|)/5], se 0 < v < + 5

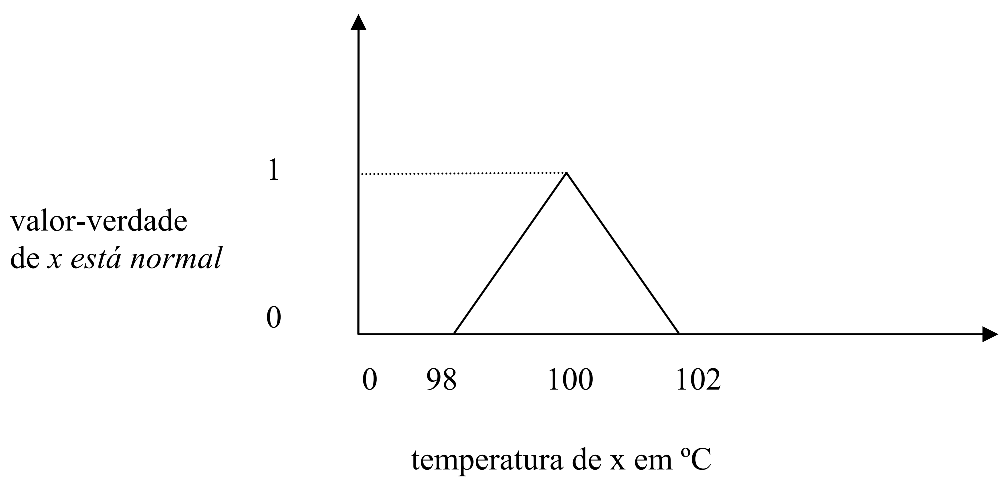
= 0, se 0 ≥ v

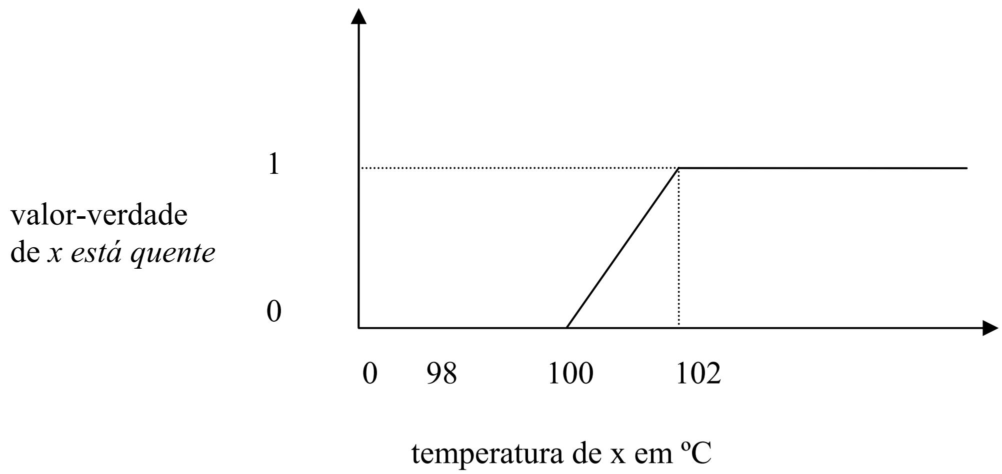
v(x está resfriando) = 1, se –5 ≥ v

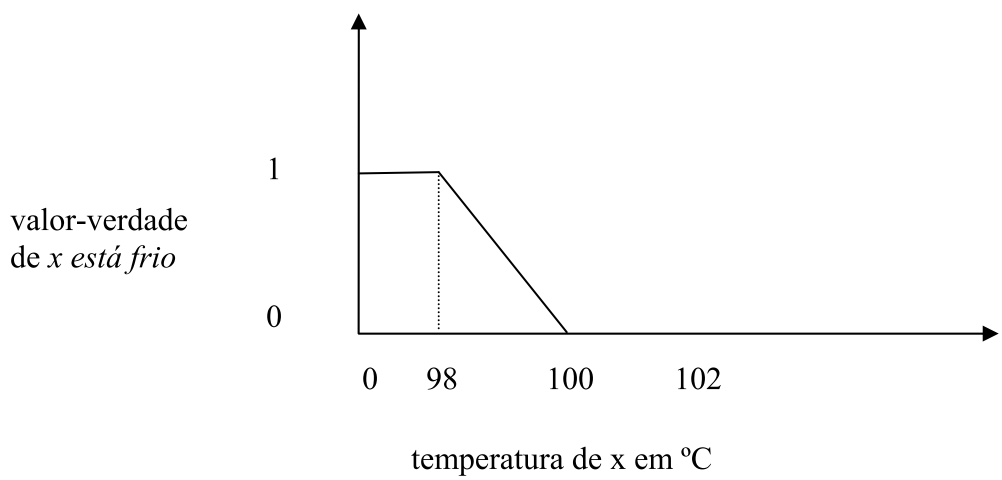
= 1 – [(5 - |v|)/5], se –5 < v < 0

= 0, se v ≥ 0

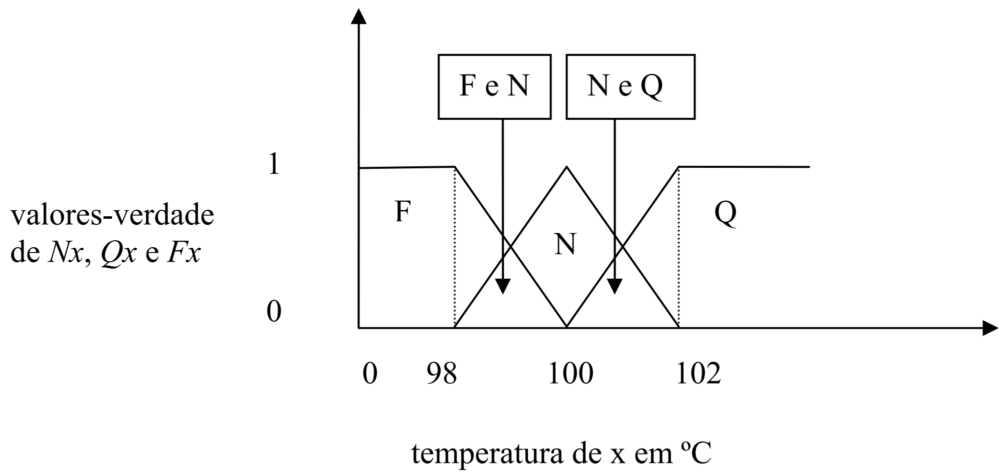
Os gráficos para os três primeiros predicados são os seguintes:





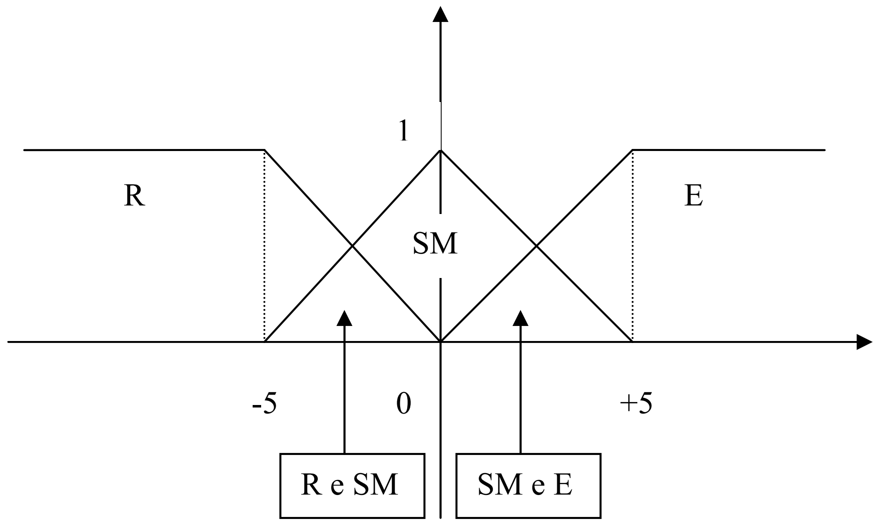




Tendo em vista que o sistema impreciso trabalha com todos esses gráficos ao mesmo tempo, eles podem ser superpostos como segue:




Na figura acima, os predicados *x está normal*, *x está quente* e *x está frio* foram representados, respectivamente, por *Nx*, *Qx* e *Fx*. Como se pode ver, há duas áreas de imprecisão, em que estes predicados se confundem. Na área de temperaturas entre 98 e 100ºC, os predicados *Fx* e *Nx* estão superpostos, de maneira tal que *x está frio* e *x está normal* podem ter valores maiores que zero. Na área de temperaturas entre 100 e 102ºC, os predicados *Nx* e *Qx* estão superpostos. Isto significa que, de acordo com os parâmetros estabelecidos para o sistema impreciso, o ambiente de temperatura controlada pode estar simultaneamente normal e frio, ou normal e quente.

Por motivos de espaço, não mostraremos aqui os gráficos separados para os predicados *x está sem mudança (SMx)*, *x está esquentando (Ex)* e *x está resfriando (Rx)*, relativos à taxa de variação de temperatura. Mas os gráficos superpostos para estes predicados estão ilustrados na figura abaixo:




Aqui também há duas áreas de imprecisão. A primeira delas ocorre quando a taxa de variação assume valores entre -5 e 0ºC/s; a segunda, quando ela assume valores entre 0 e +5ºC/s. Isto significa que, de também de acordo com os parâmetros estabelecidos para o sistema impreciso, o ambiente de temperatura controlada pode estar simultaneamente resfriando e sem mudança, ou sem mudança e esquentando.

Para ilustrar o funcionamento do algoritmo baseado nos predicados imprecisos e nas regras acima estipuladas, suponhamos que o sistema impreciso de controle esteja numa situação tal que $t_a$ = 100 ºC, $t_m$ = 101 ºC e v = + 2,5 ºC/s. Com base nestes dados, o sistema terá de decidir ou se o ventilador deve ser acionado ou se o aquecedor deve ser acionado ou se tudo deve ficar como está. Vejamos como funcionaria o algoritmo impreciso nestas circunstâncias.

Em primeiro lugar, temos a etapa do tornar impreciso. Aqui, o sistema determina os valores-verdade dos predicados imprecisos envolvidos:

v(o ambiente está normal) = 1 – [(101-100)/2] = 1 – ½ = 0,5 [porque 100 < $t_m$ < 102].

v(o ambiente está quente) = 1 – [(102 – 101)/2] = 1 – ½ = 0,5 [porque 100 < $t_m$ < 102].

v(o ambiente está frio) = 0 [porque $t_m$ > 100].

v(o ambiente está sem mudança) = (5 - |2|)/5 = 3/5 = 0,6 [porque 0 < t < +5]

v(o ambiente está esquentando) = 1 – [(5 - |2|)/5] = 1 – 3/5 = 1 – 0,6 = 0,4 [porque 0 < v < + 5]

v(o ambiente está resfriando) = 0 [porque v ≥ 0].

Como se pode ver, a situação é imprecisa: embora não esteja frio nem resfriando, o ambiente está simultaneamente um pouco normal (0,5), um pouco quente (0,5), um pouco sem mudança (0,6) e um pouco esquentando (0,4).

Para determinar a ação correta, o sistema entra na segunda etapa do processo, que envolve a realização de inferências através do algoritmo impreciso. Isto é feito através da matriz de regras apresentada anteriormente, que nos dará os seguintes resultados:

| |
|---|
| 1. se o ambiente está quente (0,5) e resfriando (0), então o ventilador é acionado (V) |
| 2. se o ambiente está normal (0,5) e resfriando (0), então o aquecedor é acionado (A) |
| 3. se o ambiente está frio (0) e resfriando (0), então o aquecedor é acionado (A) |
| 4. se o ambiente está quente (0,5) e sem mudança (0,6), então o ventilador é acionado (V) |
| 5. se o ambiente está normal (0,5) e sem mudança (0,6), então o ambiente é deixado como está (D) |
| 6. se o ambiente está frio (0) e sem mudança (0,6), então o aquecedor é acionado (A) |
| 7. se o ambiente está quente (0,5) e esquentando (0,4), então o ventilador é acionado (V) |
| 8. se o ambiente está normal (0,5) e esquentando (0,4), então o ventilador é acionado (V) |
| 9. se o ambiente está frio (0) e esquentando (E), então o aquecedor é acionado (A) |

Para calcular os valores numéricos de *A*, *V* e *D*, que correspondem respectivamente a *o aquecedor é acionado*, *o ventilador é acionado* e *o ambiente é deixado como está*, utilizaremos a regra da conjunção para os valores correspondentes de *Q*, *N* e *F* da seguinte maneira:

| |
|---|
| 1. se o ambiente está quente (0,5) e resfriando (0), então o ventilador é acionado [Min (0,5, 0) = 0] |
| 2. se o ambiente está normal (0,5) e resfriando (0), então o aquecedor é acionado [Min(0,5, 0) = 0] |
| 3. se o ambiente está frio (0) e resfriando (0), então o aquecedor é acionado [Min(0, 0) = 0] |
| 4. se o ambiente está quente (0,5) e sem mudança (0,6), então o ventilador é acionado [Min(0,5, 0,6) = 0,5] |
| 5. se o ambiente está normal (0,5) e sem mudança (0,6), então o ambiente é deixado como está [Min(0,5, 0,6) = 0,5] |
| 6. se o ambiente está frio (0) e sem mudança (0,6), então o aquecedor é acionado [Min(0, 0,6) = 0] |
| 7. se o ambiente está quente (0,5) e esquentando (0,4), então o ventilador é acionado [Min(0,5, 0,4) = 0,4] |
| 8. se o ambiente está normal (0,5) e esquentando (0,4), então o ventilador é acionado [Min(0,5, 0,4) = 0,4] |
| 9. se o ambiente está frio (0) e esquentando (0,4), então o aquecedor é acionado [Min(0, 0,4) = 0] |

O quadro acima ainda reflete uma situação imprecisa, pois os valores para *o ventilador é acionado*, obtidos a partir das regras 1, 4, 7 e 8, são, respectivamente, 0, 0,5, 0,4 e 0,4; os valores para *o aquecedor é acionado*, obtidos a partir das regras 2, 3, 6 e 9, são respectivamente, 0, 0, 0 e 0; o valor para *o ambiente é deixado como está*, obtido com base na regra 5, é 0,5. Embora tenhamos uma boa indicação de que o aquecedor não deve ser acionado, ainda não temos elementos para saber se o ambiente deve ser ventilado ou deixado como está.

A partir deste momento, o sistema entre na terceira e última etapa do processo, que corresponde à recuperação da precisão. Para obter um valor único para cada uma das proposições acima, o sistema calcula a raiz quadrada da soma dos quadrados dos valores obtidos para cada proposição *A*, *V* e *D*. Os resultados são os seguintes:

$$\text{v(o ventilador é acionado)} = \sqrt{0^2 + 0{,}5^2 + 0{,}4^2 + 0{,}4^2} = \sqrt{0{,}57} = 0{,}754$$

$$\text{v(o aquecedor é acionado)} = \sqrt{0^2 + 0^2 + 0^2 + 0^2} = 0$$

$$\text{v(o ambiente é deixado como está)} = \sqrt{0{,}5^2} = 0{,}5$$

Agora já sabemos que *o ventilador é acionado* tem valor 0,754, que *o aquecedor é acionado* tem valor 0 e que *o ambiente é deixado como está* tem valor 0,5. Para a determinação final do que deve ser feito, o sistema calcula uma média ponderada em que esses valores constituem os pesos. Assim, o valor estabelecido para *o ventilador é acionado* é –100, para *o aquecedor é acionado* é +100 e para *o ambiente é deixado como está* é 0. O número 100 foi escolhido nos dois primeiros casos porque com ele o sistema tem condições de determinar a ação do ventilador ou do aquecedor em termos percentuais. O valor zero foi escolhido para o terceiro caso porque com ele o sistema não acionará nem o ventilador nem o aquecedor, deixando as coisas como estão. Deste modo, se o valor final obtido for negativo, o ventilador deverá ser acionado; se for positivo, o aquecedor deverá ser acionado; se for nulo, nem o ventilador nem o aquecedor serão acionados. Com base nessas definições, obtemos a seguinte média ponderada:

$$[-100 \times 0{,}754 + 0 \times 0{,}5 + 100 \times 0]/[\ 0{,}754 + 0 + 0{,}5] = -\ 75{,}4/1{,}254 = -\ 60{,}128\%$$

Este resultado corresponde à terceira etapa da aplicação do algoritmo impreciso, pois fornece um valor preciso para a ação do sistema: nas condições indicadas, a saber, $t_a$ = 100 ºC, $t_m$ = 101 ºC e v = + 2,5 ºC/s, o ambiente deve ser resfriado em 60,128%. O aquecedor fica como está e o ventilador deve ser acionado. O parâmetro obtido pode indicar para o sistema, por exemplo, a velocidade de rotação das pás do ventilador.

Em resumo, o sistema partiu de uma situação imprecisa, em que o ambiente estava um pouco quente e um pouco normal e, com base nas inferências efetuadas pelo algoritmo impreciso nesta situação imprecisa, recuperou a precisão e decidiu pelo resfriamento do ambiente à taxa de 60,128%.

### 6.4. Exercícios de Lógica Imprecisa (Fuzzy)

6.4.1. Considere as definições dos valores-verdade das conectivas imprecisas seguintes:

se v(p) = x e v(q) = y, então

$v(\sim p) = 1 - x$

$v(p \,\&\, q) = \mathrm{Min}(x, y)$

$v(p \vee q) = \mathrm{Max}(x, y)$

$v(p \supset q) = 1$, se $x \leq y$

$\quad 1 - (x - y)$, se $x > y$

Com base nelas e sabendo que v(p) = 0,2 e v(q) = 0,8, determine os valores-verdade das seguintes expressões:

| | |
|---|---|
| 6.4.1.1. p v ~p | 6.4.1.6. ~(q & ~q) |
| 6.4.1.2. q v ~q | 6.4.1.7. (p & (p ⊃ q)) ⊃ q |
| 6.4.1.3. p & ~p | 6.4.1.8. (~q & (p ⊃ q)) ⊃ ~p |
| 6.4.1.4. ~(p & ~p) | 6.4.1.9. p ⊃ p |
| 6.4.1.5. q & ~q | 6.4.1.10. ~~p ⊃ p |

6.4.2. Considere agora as definições abaixo dos predicados "alto" e "velho" em lógica imprecisa:

v(x é alto) = 0, se altura de x ≤ 1,50 m;

= [altura de x – 1,50] ÷ 0,50, se 1,50 m < altura de x < 2,00 m;

= 1, se altura de x ≥ 2,00 m

v(x é velho) = 0, se idade de x ≤ 18 anos

= [idade de x – 18] ÷ 42, se 18 anos < idade de x < 60 anos

= 1, se idade de x ≥ 60 anos

Sabendo que João tem 1,80 m e 50 anos e que Maria tem 1,40 m e 25 anos, determine os valores-verdade para as sentenças abaixo (Obs.: para as conectivas, use as regras semânticas do exercício anterior):

| | |
|---|---|
| 6.4.2.1. João é velho | 6.4.2.6. Se João é velho, então Maria é alta |
| 6.4.2.2. Maria é alta | 6.4.2.7. Se João é velho e alto, então: Maria não é velha e não é alta |
| 6.4.2.3. João não é alto | |
| 6.4.2.4. Maria é velha e não é alta | 6.4.2.8. Ou Maria é alta e João não é velho, ou Maria é velha e João não é alto |
| 6.4.2.5. João é velho e alto | |

## 7. Lógica Paraconsistente[11]

### 7.1. Caracterização geral

Esta lógica foi criada simultaneamente e independentemente por Jaskowski (1948), discípulo de Lukasiewicz, e Newton C. A. da Costa (década de 1950). O sistema de Jaskowski, porém, limitava-se ao nível proposicional, enquanto os sistemas construídos por Newton da Costa iam além do cálculo proposicional, estendendo-se pelo cálculo de predicados com e sem igualdade e pelos cálculos de descrições e aplicações à teoria dos conjuntos. A expressão *paraconsistente*, utilizada para designar este tipo de lógica, foi introduzida pelo filósofo peruano Francisco Miró Quesada, em 1976.

A lógica paraconsistente tem como finalidade a construção de novas categorias lógicas para examinar trabalhar com sistemas dedutivos inconsistentes. A exposição clássica da lógica paraconsistente criada por Newton da Costa se encontra em sua tese de cátedra para a Universidade do Paraná, intitulada *Sistemas Formais Inconsistentes* e publicada em 1963. Nesta obra, da Costa constrói uma hierarquia de sistemas, $C_0$, $C_1$, $C_2$, ... $C_n$, ... $C_\omega$ , em que $C_0$ representa o cálculo bivalente clássico e os demais, a partir de $C_1$, sistemas paraconsistentes cada vez mais fracos. Embora $C_1$ e seus sucessores na série não admitam como tautologia a fórmula $\sim(A \;\&\; \sim A)$ — lei de não-contradição —, valem neles o maior número possível de expressões e regras de inferência de $C_0$. Mesmo assim, não se pode deduzir qualquer sentença nestes sistemas, o que significa que eles não são triviais. Além disso, eles admitem sentenças “bem comportadas”, para as quais valem a não-contradição e todas as leis clássicas. Uma sentença $A$ é "bem comportada" se vale para ela $\sim(A \;\&\; \sim A)$ e esta última expressão é abreviada através da notação $A^o$.

[11] As informações relativas a esta parte foram extraídas de da Costa, N. C. A. *Lógica Paraconsistente Aplicada*. S. Paulo: Ed. Atlas S.A., 1999, da Costa, N. C. A. *Sistemas Formais Inconsistentes.* Curitiba: Ed. da UFPR, 1993 e da Costa, N. C. A. *Ensaio sobre os fundamentos da lógica*. 2 ed. S. Paulo: Hucitec, 1994.

### 7.2. O sistema $C_1$

Os postulados válidos no sistema $C_1$ de Lógica Paraconsistente são os seguintes:

1. $A \supset (B \supset A)$

2. $(A \supset B) \supset ((A \supset (B \supset C)) \supset (A \supset C))$

3. de $A$ e $A \supset B$ pode-se inferir $B$

4. $A \supset (A \supset (A \& B))$

5. $(A \& B) \supset A$

6. $(A \& B) \supset B$

7. $A \supset (A \vee B)$

8. $B \supset (A \vee B)$

9. $(A \supset C) = ((B \supset C) \supset ((A \vee B) \supset C))$

10. $B^{o} \supset ((A \supset B) \supset ((A \supset \sim B) \supset \sim A))$

11. $A \vee \sim A$

12. $\sim\sim A \supset A$

13. $A^{o} \supset (\sim A)^{o}$

14. $A^{o} \& B^{o} \supset (A \& B)^{o}$

15. $A^{o} \& B^{o} \supset (A \vee B)^{o}$

16. $A^{o} \equiv B^{o} \supset (A \equiv B)^{o}$

O postulado 3 é, na realidade, a regra de inferência do *modus ponens*. Como se pode ver pela lista acima, embora a lei de não-contradição não seja válida em $C_1$, o sistema admite o princípio do terceiro excluído ($A \vee \sim A$) e parte da lei da dupla negação ($\sim\sim A \supset A$).

As demonstrações em $C_1$ são feitas com o auxílio das regras metalógicas de inferência estabelecidas por Kleene em 1952 e que podem ser resumidas como segue:[12]

[12] Kleene, S. C. *Introduction to Metamathematics*. Amsterdam: North Holland Pub. Co.; Groningen: P. Noordhoff N. V., 1952, p. 98-99.

| Conectiva | Regra de introdução | Regra de eliminação |
|---|---|---|
| 1. Implicação | $\dfrac{G, A \vdash B}{G \vdash A \supset B}$ | A e $A \supset B \vdash B$ |
| 2. Conjunção | A e $B \vdash A \,\&\, B$ | $A \,\&\, B \vdash A$<br>$A \,\&\, B \vdash B$ |
| 3. Disjunção | $A \vdash A \vee B$<br>$B \vdash A \vee B$ | $\dfrac{G \text{ e } A \vdash C,\ G \text{ e } B \vdash C}{G \text{ e } A \vee B \vdash C}$ |
| 4. Negação | $\dfrac{G \text{ e } A \vdash B,\ G \text{ e } A \vdash {\sim}B}{G \vdash {\sim}A}$ [red. ao absurdo] | ${\sim\sim}A \vdash A$ |

Estas regras são *metalógicas* porque dizem algo sobre as proposições do sistema e não pertencem a ele. A expressão $A, A \supset B \vdash B$, por exemplo, deve ser interpretada como dizendo que, no sistema analisado, $B$ é dedutível de $A$ e $A \supset B$. A regra 1 de introdução da implicação também é conhecida como *teorema da dedução*, que abreviaremos de agora em diante por *td*. Todas as regras acima são válidas em $C_1$, com exceção da regra 4 de introdução da negação, que é substituída pela seguinte:

$$\frac{G \vdash B^{o} \text{ e } G, A \vdash B \text{ e } G, A \vdash {\sim}B}{G \vdash {\sim}A}$$

Esta é a versão paraconsistente do princípio de redução ao absurdo e pode ser interpretada como segue: se de $G$ se segue uma proposição bem comportada, a saber, $B^{o}$, e se a associação de $G$ com $A$ leva tanto a $B$ como a ${\sim}B$, então de $G$ se segue a negação de $A$.

Esta versão paraconsistente do princípio de redução ao absurdo pode ser demonstrada da seguinte maneira em $C_1$, com o auxílio das regras metalógicas de Kleene devidamente adaptadas:

| Etapa | Justificativa |
|---|---|
| 1. G $\vdash$ Bº | hipótese |
| 2. G, A $\vdash$ B | hipótese |
| 3. G, A $\vdash$ ~B | hipótese |
| 4. G $\vdash$ A $\supset$ B | 2, td |
| 5. G $\vdash$ A $\supset$ ~B | 3, td |
| 6. Bº $\supset$ ((A $\supset$ B) $\supset$ ((A $\supset$ ~B) $\supset$ ~A)) | postulado 10 de $C_1$ (Post 10) |
| 7. Bº e A $\supset$ B e A $\supset$ ~B e Post 10 $\vdash$ ~A | eliminação da implicação (ei) |
| 8. G $\vdash$ ~A | 1, 2, 3, 6, 7 |

Outros exemplos de demonstrações em $C_1$ com o auxílio das regras metalógicas de Kleene adaptadas:

1º Exemplo: demonstrar que Bº, A $\supset$ B $\vdash$ ~B $\supset$ ~A

| Etapa | Justificativa |
|---|---|
| 1. Bº e A $\supset$ B e A e ~B $\vdash$ Bº | hipótese |
| 2. Bº e A $\supset$ B e A e ~B $\vdash$ B | 1, eliminação da implicação |
| 3. Bº e A $\supset$ B e A e ~B $\vdash$ ~B | 1, eliminação da conjunção |
| 4. Bº e A $\supset$ B e ~B $\vdash$ ~A | 1, 2, 3 redução ao absurdo |
| 5. Bº e A $\supset$ B $\vdash$ ~B $\supset$ ~A | 4, td |

2º Exemplo: demonstrar que

$$\frac{G \vdash B^o,\ G \text{ e } A \vdash B,\ G \text{ e } A \vdash \sim B}{G \vdash \sim A}$$

| Etapa | Justificativa |
|---|---|
| 1. G $\vdash$ Bº | hipótese |
| 2. G e A $\vdash$ B | hipótese |
| 3. G e A $\vdash$ ~B | hipótese |
| 4. G $\vdash$ ~A | 1, 2, 3 ra |

A tabela abaixo relaciona algumas expressões válidas e não-válidas em $C_1$, em comparação com o cálculo clássico de proposições:

| Expressão do CP válida em $C_1$. | Expressão do CP não-válida em $C_1$. |
|---|---|
| $\sim\sim A \supset A$ | $A \supset \sim\sim A$ |
| $\sim(\sim A \ \& \ \sim\sim A)$ | $\sim(A \ \& \ \sim A)$ |
| $(A \supset B) \supset (\sim A \vee B)$ | $(\sim A \vee B) \supset (A \supset B)$ |
| $\sim(A \ \& \ \sim B) \supset (A \supset B)$ | $(A \supset B) \supset \sim(A \ \& \ \sim B)$ |
| $\sim(\sim A \ \& \ \sim B) \supset (A \vee B)$ | $(A \vee B) \supset \sim(\sim A \ \& \ \sim B)$ |

Como se pode ver, o tripé proposicional clássico, em que as sentenças $A \supset B$, $\sim A \vee B$ e $\sim(A \ \& \ \sim B)$ são equivalentes entre si, não é válido em $C_1$. Neste sistema, as relações entre essas sentenças ficam reduzidas às seguintes relações de implicação:

$$
\begin{tikzcd}
 & \boxed{\sim(A \ \& \ \sim B)} \arrow[dl] \arrow[dr] & \\
\boxed{A \supset B} \arrow[rr] & & \boxed{\sim A \vee B}
\end{tikzcd}
$$

Acrescentemos agora, aos exemplos de demonstrações já dados, mais alguns, relativos às proposições apresentadas na tabela acima:

3º Exemplo: demonstrar que $\sim\sim A \supset A$.

| Etapa | Justificativa |
|---|---|
| 1. $\sim\sim A \vdash A$ | eliminação da negação |
| 2. $\vdash \sim\sim A \supset A$ | 1 td |

4º Exemplo: demonstrar que $(A \supset B) \supset (\sim A \vee B)$.

| Etapa | Justificativa |
|---|---|
| 1. $A \supset B, A \vdash B$ | eliminação da implicação |
| 2. $B \vdash \sim A \vee B$ | introdução da disjunção |
| 3. $A \supset B, A \vdash \sim A \vee B$ | 1,2 |
| 4. $A \supset B, \sim A \vdash \sim A \vee B$ | introdução da disjunção |
| 5. $A \supset B, A \vee \sim A \vdash \sim A \vee B$ | 3,4 eliminação da disjunção |
| 6. $A \vee \sim A \vdash (A \supset B) \supset (\sim A \vee B)$ | 5 td |
| 7. $\vdash A \vee \sim A$ | postulado 11 |
| 8. $\vdash (A \supset B) \supset (\sim A \vee B)$ | 6,7 |

Em C1, se uma sentença *B* é "bem comportada", então ela implica todas as expressões válidas do cálculo proposicional clássico, inclusive aquelas que, sozinhas, não são válidas no sistema. Isto fica claro na lista abaixo. As expressões nela apresentadas envolvem sentenças pertencentes às segunda coluna da tabela anterior. Embora sozinhas não sejam válidas, elas podem fazer parte de expressões válidas em $C_1$ quando são implicadas por uma sentença "bem comportada":

$B^o \supset (A \supset \sim\sim A)$

$B^o \supset \sim(A \,\&\, \sim A)$

$B^o \supset ((\sim A \vee B) \supset (A \supset B))$

$B^o \supset ((A \supset B) \supset \sim(A \,\&\, \sim B))$

$B^o \supset ((A \vee B) \supset \sim(\sim A \,\&\, \sim B))$

Todas essas expressões poderiam ser interpretadas assim: *se B é bem comportada, então tal lei clássica é válida para B*. Isto significa que, se acrescentarmos o princípio de não-contradição a $C_1$, obteremos o cálculo proposicional clássico. Com efeito, uma das formulações do princípio de não contradição é *~(B & ~B)*, ou seja, $B^o$. Ora, já sabemos que $B^o$ implica todas as leis clássicas em $C_1$. Deste modo, a adição de $B^o$ aos postulados de $C_1$ o tornaria equivalente ao cálculo proposicional clássico.

Vejamos mais alguns exemplos de demonstrações, agora relativos a proposições bem comportadas:

5º Exemplo: demonstrar que $B^o \supset (A \supset {\sim\sim}A)$.

| Etapa | Justificativa |
|---|---|
| 1. $B^o, A, {\sim}A \vdash A$ | eliminação da conjunção |
| 2. $B^o, A, {\sim}A \vdash {\sim}A$ | eliminação da conjunção |
| 3. $B^o, A \vdash {\sim\sim}A$ | 1,2 ra (G é vazio) |
| 4. $B^o \vdash A \supset {\sim\sim}A$ | 3 td |
| 5. $\vdash B^o \supset (A \supset {\sim\sim}A)$ | 4 td |

6º Exemplo: demonstrar que $B^o \supset {\sim}(A \,\&\, {\sim}A)$.

| Etapa | Justificativa |
|---|---|
| 1. $B^o, A \,\&\, {\sim}A \vdash A$ | eliminação da conjunção |
| 2. $B^o, A \,\&\, {\sim}A \vdash {\sim}A$ | eliminação da conjunção |
| 3. $B^o \vdash {\sim}(A \,\&\, {\sim}A)$ | 1,2 ra (G é vazio) |
| 4. $\vdash B^o \supset {\sim}(A \,\&\, {\sim}A)$ | 3 td |

7º Exemplo: demonstrar que Bº ⊃ ~(~A & ~~A).

| Etapa | Justificativa |
|---|---|
| 1. Bº, ~A & ~~A \|— ~A | eliminação da conjunção |
| 2. Bº, ~A & ~~A \|— ~~A | eliminação da conjunção |
| 3. ~~A \|— A | eliminação da negação |
| 4. Bº, ~A & ~~A \|— A | 2,3 |
| 5. Bº \|— ~(~A & ~~A) | 1,4 ra (G é vazio) |
| 6. \|— Bº ⊃ ~(~A & ~~A) | 5 td |

O sistema $C_1$ admite tabelas de valores-verdade trivalentes para as principais conectivas sentenciais:

| A | B | ~A | ~B | A ⊃ B | A & B | A v B |
|---|---|---|---|---|---|---|
| 1 | 1 | 3 | 3 | 1 | 1 | 1 |
| 2 | 1 | 1 | 3 | 1 | 1 | 1 |
| 3 | 1 | 1 | 3 | 1 | 3 | 1 |
| 1 | 2 | 3 | 1 | 1 | 1 | 1 |
| 2 | 2 | 1 | 1 | 1 | 1 | 1 |
| 3 | 2 | 1 | 1 | 1 | 3 | 1 |
| 1 | 3 | 3 | 1 | 3 | 3 | 1 |
| 2 | 3 | 1 | 1 | 3 | 3 | 1 |
| 3 | 3 | 1 | 1 | 1 | 3 | 3 |

Nessa tabela, as conectivas *não foram* definidas da mesma maneira que na lógica trivalente de Lukasiewicz. Cabe observar que o valor *2* só aparece nas colunas relativas a sentenças elementares como *A* ou *B*. Ele nunca aparece nas colunas correspondentes à negação dessas sentenças ou às ligações delas através de conectivas binárias. É justamente esse fato que produz as peculiaridades do sistema $C_1$.

O sistema $C_0$, que corresponde ao cálculo proposicional clássico, contém o sistema $C_1$, que por sua vez contém o sistema $C_2$, que por sua vez contém o sistema $C_3$, etc. Cada um desses sistemas possui uma tabela de valores-verdade com um valor a mais do que o precedente.

### 7.3. O sistema $C^*_1$

Newton da Costa construiu também uma hierarquia de cálculos de predicados para sistemas formais inconsistentes, representando-a pela série $C^*_0$, $C^*_1$, $C^*_2$, *etc.* O cálculo $C^*_0$ representa o cálculo de predicados clássico e $C^*_1$ é construído através da adição, ao cálculo $C_1$, dos seguintes postulados:

17. $\underline{C \supset Px}$

    $C \supset (x)Px$ [introdução do quantificador universal (iqu)]

18. $Pa \supset (\exists x)Px$
19. $(x)Px^o \supset [(x)Px]^o$
20. $(x)Px \supset Pa$
21. $\underline{Px \supset C}$

    $(\exists x)Px \supset C$ [introdução do quantificador existencial (iqe)]

22. $(x)Px^o \supset (\exists x)Px^o$
23. <u>A e B são congruentes</u>

    $A \equiv B$

Observação: $Px^o$ é definido por $\sim[Px \& \sim Px]$.

Todas as regras metalógicas de inferência estabelecidas por Kleene para o cálculo proposicional são válidas também em $C^*_1$, com a óbvia exceção da regra de redução ao absurdo, que só vale na forma adaptada anteriormente apresentada. Para trabalhar com os quantificadores, são acrescentadas as seguintes regras de Kleene:

| Quantificador | Regra de introdução | Regra de eliminação |
|---|---|---|
| 5. Universal | $Px \vdash (x)Px$ | $(x)Px \vdash Pa$ |
| 6. Existencial | $Pa \vdash (\exists x)Px$ | $\underline{Px, Qx \vdash A}$<br>$Px, (\exists x)Qx \vdash A$ |

Seguem abaixo algumas expressões do cálculo clássico de predicados que são válidas em $C^{*}_{-1}$:

$(x)(y)Pxy \equiv (y)(x)Pxy$

$(\exists x)(\exists y)Pxy \equiv (\exists y)(\exists x)Pxy$

$(x)(y)Pxy \supset (x)Pxx$

$(\exists x)Pxx \supset (\exists x)(\exists y)Pxy$

$(x)Px \supset (\exists x)Px$

$(\exists x)(y)Pxy \supset (y)(\exists x)Pxy$

Seguem agora algumas expressões do cálculo clássico de predicados que não são válidas em $C^{*}_{-1}$:

$\sim(\exists x)\sim Px \equiv (x)Px$

$\sim(x)\sim Px \equiv (\exists x)Px$

$\sim(\exists x)Px \equiv (x)\sim Px$

$(\exists x)\sim Px \equiv \sim(x)Px$

Se, porém, uma dada sentença *Px* é bem comportada, ela implica as relações desta última lista:

$Px^{o} \supset [\sim(\exists x)\sim Px \equiv (x)Px]$

$Px^{o} \supset [\sim(x)\sim Px \equiv (\exists x)Px]$

$Px^{o} \supset [\sim(\exists x)Px \equiv (x)\sim Px]$

$Px^{o} \supset [(\exists x)\sim Px \equiv \sim(x)Px]$.

O cálculo $C^{*}_{-1}$ é um subsistema do cálculo clássico de predicados. Aqui também, como no caso de $C_1$, basta acrescentar a $C^{*}_{-1}$ o esquema *~(A & ~A)* como novo postulado para obter o cálculo clássico.

Por motivos de espaço, não daremos exemplos de demonstrações em $C^{*}_{-1}$ aqui.

De acordo com Newton da Costa, a lógica paraconsistente vem encontrando aplicações as mais variadas, como, por exemplo: a) na axiomatização da teoria dos conjuntos sem as restrições fortes estabelecidas para evitar paradoxos; b) em lógica epistêmica e em lógica da crença, pois costumamos manter crenças contraditórias em nossas vidas; c) em física, no tratamento de teorias incompatíveis, como ocorre com a mecânica quântica e a relatividade geral; d) em inteligência artificial, com a criação de algoritmos e robôs paraconsistentes; e) em psicanálise, que, segundo alguns autores, exige uma lógica paraconsistente; f) em questões filosóficas envolvendo a dialética, que, em algumas de suas formulações, exige categorias paraconsistentes.[13]

Os estudos com sistemas paraconsistentes revelam que, apesar de sua artificialidade, eles são tão válidos quanto os sistemas clássicos. Tais estudos também mostram que, para ter aplicações frutíferas, qualquer sistema paraconsistente precisa de sentenças "bem comportadas". Talvez a lição mais importante a ser extraída aqui seja a constatação de que a melhor maneira de compreender certos princípios lógicos, como o de não-contradição, está em construir sistemas em que eles não valem.

### 7.4. Exercícios de Lógica Paraconsistente

7.4.1. Utilizando as regras de Kleene adaptadas para $C_1$, demonstrar que:

7.4.1.1. $\dfrac{G \text{ e } {\sim}A \vdash B^{o},\ G \text{ e } {\sim}A \vdash B,\ G \text{ e } {\sim}A \vdash {\sim}B}{G \vdash A}$

7.4.1.2. $B^{o}, A \supset {\sim}B \vdash B \supset {\sim}A$

7.4.1.3. $B^{o}, {\sim}A \supset B \vdash {\sim}B \supset A$

7.4.1.4. $B^{o}, {\sim}A \supset {\sim}B \vdash B \supset A$

7.4.2. Com base na tabela de valores-verdade para o cálculo $C_1$, construa as tabelas de valores-verdade das expressões abaixo:

[13] Cfr. da Costa, N. et al. *Lógica Paraconsistente Aplicada*. S. Paulo: Ed. Atlas S. A., 1999, p. 15.

| | |
|---|---|
| 7.4.2.1. A v ~A | 7.4.2.7. ~A & ~~A |
| 7.4.2.2. A & ~A | 7.4.2.8. ~(~A & ~~A) |
| 7.4.2.3. ~~A | 7.4.2.9. ~A v ~~A |
| 7.4.2.4. ~~A ⊃ A | 7.4.2.10. ~A ⊃ (A ⊃ B) |
| 7.4.2.5. A ⊃ ~~A | 7.4.2.11. (A & ~A) ⊃ B |
| 7.4.2.6. ~(A & ~A) [ou A°] | 7.4.2.12. (A & (A ⊃ B)) ⊃ B |

## 8. Metalógica

### 8.1. Contexto histórico do aparecimento[14]

No domínio da matemática, o século XIX foi marcado pelo aparecimento dos trabalhos de Cantor sobre o infinito, das geometrias não-euclidianas e de certas antinomias de difícil solução.

Os trabalhos de Cantor levaram a conclusões bastante surpreendentes a respeito de conjuntos infinitos, como, por exemplo: a) não é possível contar as frações decimais infinitas entre zero e um; b) há uma hierarquia de conjuntos infinitos, cada um com mais potência que o anterior, começando pelo conjunto dos números naturais (o primeiro conjunto infinito após $\omega$ é $2^{\omega}$, e $2^{\omega} > \omega$). A possibilidade de chegar às conclusões paradoxais a que Cantor chegou levantaram a suspeita de que poderia haver alguma contradição na matemática.

Paralelamente às descobertas de Cantor, foram construídas novas geometrias, que continham versões diferentes do quinto postulado de Euclides. Este postulado, em uma de suas formulações mais simples, afirma que *por um ponto exterior a uma reta, pode-se traçar uma e somente uma paralela a esta reta*. Na geometria euclidiana, vale o teorema segundo o qual *a soma dos ângulos internos de um triângulo é equivalente a dois retos*. O matemático Lobachewsky propôs uma geometria alternativa, ao alterar este postulado da

[14] As informações desta parte foram extraídas de Körner, S. *Uma introdução à filosofia da matemática*. Rio: Zahar, 1985 e de Ladrière, J. *Limitaciones Internas de los Formalismos*. Madrid: Ed. Tecnos, s/d.

seguinte maneira: *por um ponto exterior a uma reta, pode-se traçar uma infinidade de paralelas a esta reta*. Nesta geometria alternativa, vale um teorema diferente, segundo o qual *a soma dos ângulos internos de um triângulo é inferior a dois retos*. O matemático Riemann, por sua vez, propôs uma outra geometria, ao dar a seguinte formulação ao quinto postulado: *por um ponto exterior a uma reta, não se pode traçar nenhuma paralela a esta reta*. Nesta outra geometria alternativa, vale um teorema diferente dos dois anteriores: *a soma dos ângulos internos de um triângulo é superior a dois retos*. Estas geometrias alternativas foram denominadas *não-euclidianas*, por razões óbvias. O aparecimento destas geometrias criou uma grande perplexidade, que aumentou quando se demonstrou que os teoremas de cada uma delas poderiam ser mapeados nos teoremas da geometria euclidiana de uma forma tal que cada teorema não-euclidiano seria verdadeiro se e somente se o correspondente teorema euclidiano fosse verdadeiro.

As antinomias que surgiram no século XIX dizem respeito principalmente a conceitos que fazem intervir a idéia de *totalização*, como: a) o conjunto de todos os conjuntos; b) o conjunto de todos os conjuntos que não contêm a si próprios como elementos; c) o conjunto de todos os números ordinais (transfinitos), ordenado segundo sua magnitude; d) o conjunto de todos os números cardinais. Tais conceitos levam a contradições e levantam dúvidas a respeito de noções matemáticas importantes, como a de conjunto, e de procedimentos de raciocínio considerados corretos até então. A versão mais simples dessa antinomias está no conjunto de todos os conjuntos que não estão contidos em si próprios, inventado por Russell. O conjunto de todas as coisas verdes, por exemplo, não está contido em si mesmo, pois não é verde. Mas o conjunto de todos os conjuntos com mais de um elemento está contido em si próprio, pois possui mais de um elemento. Podemos, pois, definir o conjunto de Russell da seguinte maneira:

$R = \{x \mid x \not\subset x\}$.

Representando por *V* e por *M* respectivamente o conjunto de todas as coisas verdes e o conjunto de todos os conjuntos com mais de um elemento, podemos dizer que

$V \not\subset R$ e $M \subset R$.

A pergunta que se coloca neste ponto é a seguinte: e o próprio R? Está contido em si próprio ou não? Se $R \subset R$, então R possui a propriedade dos elementos que pertencem a R, a saber, $x \not\subset x$. Portanto, $R \not\subset R$. Como isso é uma contradição, temos de dizer que $R \not\subset R$.

Mas se isso é verdade, então R preenche a condição para ser membro de R, que é $x \not\subset x$. Portanto, $R \subset R$. De qualquer modo, caímos em contradição. E isto permite concluir que

$R \subset R$ se e somente se $R \not\subset R$!

Russell ofereceu também uma formulação brincalhona para este conjunto: o barbeiro de uma certa cidade só faz a barba das pessoas que não se barbeiam. Pergunta-se: o barbeiro faz a barba de si próprio ou não? Se ele faz a barba de si próprio, então ele não faz a barba de si próprio, pois só faz a barba das pessoas que não se barbeiam. Se ele não faz a barba de si próprio, então ele faz a barba de si próprio, pois só faz a barba das pessoas que não se barbeiam. Conclusão: o barbeiro desta cidade faz a barba de si próprio se e somente se ele não faz a barba de si próprio!

Os problemas apresentados acima criaram a necessidade de empreender um trabalho rigoroso de fundamentação da matemática. Assim, para compreender e superar as dificuldades encontradas, surgiram diversos programas de investigação e fundamentação da matemática, como o *logicismo* de Frege, o *intuicionismo* de Brouwer, a *metamatemática* de Hilbert e o *formalismo*.

O logicismo, proposto inicialmente por Frege e logo adotado por Russell, considera que lógica e matemática são a mesma coisa. Os objetos matemáticos são objetos da lógica. Nesta perspectiva, não há como indicar onde termina a lógica e onde começa a matemática num sistema formal que inclua ambas. A formulação mais acabada de um sistema logicista está na obra de Russell e Whitehead, *Principia Mathematica*. A dificuldade do logicismo está em que o sistema fregiano usa de maneira acrítica o conceito de *infinito atual*, permitindo a formulação de paradoxos em seu interior. Outro conceito problemático no sistema fregiano é o de conjunto de conjunto, que permite a introdução de contradição no sistema através da construção do conjunto *R* de Russell, mencionado acima. Para solucionar o problema, Russell criou a teoria dos tipos lógicos, que é controversa. Por motivos de espaço, porém, não a discutiremos aqui.

O intuicionismo, proposto por Brouwer, considera que o objeto matemático não possui realidade autônoma, existindo somente no ato do sujeito cognoscente pelo qual é engendrado. A existência do objeto matemático depende da possibilidade de ser ele construído pelo sujeito cognoscente através de uma intuição. Assim, a base da noção de *número inteiro*, por exemplo, é a intuição da estrutura do tempo. O infinito, nesta perspectiva, só existe como algo potencial, nunca como totalidade dada. Isto implica na recusa em admitir conjuntos infinitos enquanto totalidades dadas, atuais, e, portanto, na recusa em admitir as demonstrações de Cantor sobre conjuntos infinitos. Ao elaborar os princípios da lógica intuicionista, Brouwer recusa a noção de *infinito atual* e o princípio do terceiro excluído. Esta lógica foi formalizada por Heyting. Para o intuicionista, *existência matemática* é sinônimo de *construtibilidade*. A dificuldade da proposta intuicionista está em que, se aderirmos a ela, diversos capítulos importantes da matemática clássica deverão ser excluídos do campo do conhecimento. Além disso, não está claro como é possível ao sujeito cognoscente intuir o processo pelo qual ele chega a uma proposição matemática falsa.

O programa da metamatemática, também conhecido como *teoria da demonstração*, foi proposto por Hilbert, numa tentativa de conciliar os novos resultados matemáticos com as exigências intuicionistas. Ele pretende acomodar a matemática transfinita numa matemática construtiva. Para tanto, ele sugere que trabalhemos com dois níveis: a) o matemático, que admite todo tipo de raciocínio, sem as restrições intuicionistas — aqui, as demonstrações de Cantor, que recorrem a totalidades infinitas atuais, são aceitas; b) o metamatemático, que envolve uma reflexão sobre a matemática em termos também matemáticos e está submetido às restrições intuicionistas — aqui, apenas demonstrações construtivistas são aceitas. O programa de Hilbert estabelece que, no nível matemático, todos os procedimentos usuais são permitidos, desde que, no nível metamatemático, sejamos capazes de demonstrar a consistência do que foi feito no nível matemático. Com esse procedimento, seria possível a fundamentação da matemática de maneira rigorosa. Para Hilbert, *existência matemática* é sinônimo de *não-contradição*. A maior dificuldade do programa da metamatemática está no teorema de Gödel, que mostra a impossibilidade de demonstrar a consistência da matemática nos termos propostos por Hilbert.

A evolução da investigação acerca dos fundamentos da matemática levou os pesquisadores a eliminarem progressivamente as questões filosóficas que afetam as posições anteriores, em benefício de uma concepção rigorosamente formalista do problema. Para o formalista, o resultado concreto de todas as tentativas descritas nas posições anteriores é a construção de um sistema formal, desaparecendo com isso as oposições. O formalista não aceita as limitações intuicionistas para as demonstrações matemáticas, mas também não identifica a essência da matemática com a não-contradição. Nesta perspectiva, *fundamentar* é sinônimo de *formalizar*.

As investigações relativas aos fundamentos da matemática levaram não só ao aparecimento da metamatemática, mas também da *metalógica*, que constitui uma reflexão sobre a lógica em termos também lógicos. Isto aconteceu em virtude da profunda ligação entre os estudos lógicos e matemáticos, desde o estabelecimento do programa logicista de Frege.

### 8.2. Principais conceitos metalógicos

#### 8.2.1. Consistência:

8.2.1.1. Consistência (em sentido sintático):

8.2.1.1.1. Se um sistema formal SF possui o operador de negação, ele é consistente em sentido sintático quando não é possível derivar em SF uma sentença e sua negação. O cálculo proposicional, por exemplo, é consistente neste sentido, porque não é possível derivar nele a expressão $p \,\&\, {\sim}p$.

8.2.1.1.2. Se SF não possui o operador de negação, ele é consistente em sentido sintático quando nem toda fórmula bem formada é derivável em SF. Em outras palavras, um sistema formal é consistente se existe pelo menos uma sentença bem formada que não é derivável nele.

8.2.1.1.3. ω-consistência:

Esta noção sintática foi introduzida por Tarski e é aplicável apenas a sistemas que contenham uma formalização da aritmética, o operador de negação e os quantificadores universal e particular. Um sistema formal SF do tipo indicado é ω-consistente quando *não possui* um predicado Px, aplicável a números, tal que:

(∃x)Px é derivável em SF;

~P(0), ~P(1), ~P(2), etc., são também deriváveis em SF.

Isto equivale a dizer que em SF é possível provar ao mesmo tempo que existe um número que possui a propriedade *P*, mas que o número 0 não tem essa propriedade, o número 1 não tem essa propriedade, o número 2 não tem essa propriedade, etc.

Se um sistema é ω-consistente, ele é também consistente. Todavia, se ele é consistente, isto não significa que seja ω-consistente.

8.2.1.2. Consistência (em sentido semântico):

Um sistema formal SF é consistente em sentido semântico quando SF possui pelo menos um modelo. Em outras palavras, SF é consistente em sentido semântico quando é realizável.

**8.2.2. Completude**:

8.2.2.1. Completude sintática:

8.2.2.1.1. Completude sintática em sentido forte: um sistema formal SF é completo em sentido forte quando toda sentença de SF é derivável ou refutável.

8.2.2.1.2. Completude sintática em sentido fraco: um sistema SF é é completo em sentido fraco quando SF se torna inconsistente com o acréscimo de uma sentença não derivável ao conjunto de seus axiomas.

8.2.2.2. Completude semântica:

8.2.2.2.1. Completude semântica absoluta: um sistema SF é completo semanticamente em sentido absoluto quando toda sentença válida é derivável em SF e toda sentença derivável em SF é válida. Uma sentença válida é um enunciado verdadeiro nas diversas interpretações possíveis de SF.

8.2.2.2.2. Completude semântica relativa a uma interpretação: um sistema SF é completo em relação a uma interpretação quando toda sentença de SF correspondente a um enunciado certo nesta interpretação é derivável em SF. Assim, SF é completo em relação ao modelo M, por exemplo, se a cada enunciado verdadeiro em M corresponde uma e somente uma sentença derivável em SF e conversamente.

**8.2.3. Decidibilidade**:

8.2.3.1. Decidibilidade em sentido sintático: um sistema SF é decidível em sentido sintático quando existe um procedimento efetivo para determinar se cada sentença de SF é derivável em SF ou não.

8.2.3.2. Decidibilidade em sentido semântico: um sistema SF é decidível em sentido semântico quando existe um procedimento efetivo para decidir se cada sentença de SF é válida ou não em um determinado modelo M.

Principais resultados metalógicos positivos:

8.2.3.2.1. Cálculo proposicional: é consistente em sentido sintático, completo em sentido sintático fraco e decidível.

8.2.3.2.2. Cálculo de predicados de primeira ordem: é consistente no sentido sintático, completo semanticamente em sentido absoluto e decidível para certas classes de proposições.

8.2.3.2.3. Cálculo dos Principia Mathematica: é consistente em sentido sintático, incompleto em sentido semântico e indecidível.

**8.3. Exemplos de demonstrações metalógicas:**

**8.3.1. Prova da consistência do Cálculo Proposicional:**

Neste cálculo, tal como definido em seções anteriores do presente texto, a expressão $p \supset (\sim p \supset q)$ é um teorema. Supondo que $p$ e $\sim p$ sejam também dedutíveis em CP, teríamos:

| Etapa | Justificativa |
|---|---|
| 1. $p \supset (\sim p \supset q)$ | Teorema do CP |
| 2. $p$ | Premissa adicional |
| 3 $\sim p \supset q$ | 1, 2 separação |
| 4. $\sim p$ | Premissa adicional |
| 5. $q$ | 3, 4 separação |

Como se pode ver, se $p$ e $\sim p$ fossem também teoremas de CP, qualquer sentença $q$ seria demonstrável em CP. Assim, a existência de uma contradição em CP, representada aqui por $p \& \sim p$, permitiria demonstrar qualquer sentença em CP, que se tornaria trivial.

Para provar que CP é consistente, temos de mostrar, portanto, que existe pelo menos uma sentença bem formada que não é derivável em CP. Para atingir esse objetivo, temos de mostrar que: 1) há uma propriedade comum a todos os axiomas de CP; 2) esta propriedade é hereditária, ou seja, ela se transmite a todos os teoremas através das regras de inferência;

3) nem toda fórmula bem formada de CP possui essa propriedade e, portanto, nem toda sentença é demonstrável em CP. A demonstração envolve, assim três etapas, que são desenvolvidas a seguir:

Etapa 1: a propriedade comum a todos os axiomas é *ser uma tautologia*. Em outras palavras, os axiomas de CP são verdadeiros em todos os casos possíveis. Isto pode ser facilmente estabelecido através das tabelas de valores-verdade.

Etapa 2: a propriedade comum aos axiomas de CP de *ser uma tautologia* é transmitida hereditariamente a todos os teoremas de CP através das regras de inferência. Com efeito, a regra de substituição não afeta o caráter tautológico da expressão à qual foi aplicada. Por exemplo, o Axioma 1, representado por $(p \vee p) \supset p$, não perde o caráter tautológico quando todas as ocorrências de $p$ são substituídas por ocorrências de expressões permitidas pelas regras de formação de fórmulas bem formadas. Assim, substituindo $p$ por $q$, $\sim r$, ou $q \vee r$, obteremos, respectivamente:

$(q \vee q) \supset q$

$(\sim r \vee \sim r) \supset \sim r$

$((q \vee r) \vee (q \vee r)) \supset (q \vee r)$.

Ora, todas essas expressões continuam sendo tautologias. O mesmo acontecerá com quaisquer outras expressões de CP que coloquemos no lugar de $p$ no axioma 1. A outra regra de inferência de CP, a de separação, mantém o caráter tautológico de todas as expressões envolvidas. Com efeito, as deduções em CP são estruturadas de maneira tal que, uma vez estabelecidos os axiomas, pode-se construir, a partir deles e das regras de substituição, expressões da forma $A \supset T$, em que $A$ representa um axioma ou uma variante dele obtida por substituição de expressões e $T$ representa um teorema de CP. Ora, o antecedente $A$ de $A \supset T$ é sempre verdadeiro e a regra de separação, neste caso, só pode ser aplicada se o conseqüente $T$ for também sempre verdadeiro. Deste modo, todas as expressões de CP obtidas através da regra de separação serão também tautológicas. Portanto, a propriedade de *ser uma tautologia* é transmitida hereditariamente aos teoremas de CP através da aplicação das regras de inferência.

Etapa 3: nem toda fórmula bem formada de CP possui a propriedade de *ser uma tautologia*. A expressão $p \vee q$, por exemplo, não possui essa propriedade, como pode ser facilmente estabelecido através da correspondente tabela de valores-verdade. Ora, isto significa que *p v q* não é dedutível a partir dos axiomas e das regras de inferência de CP, uma vez que não possui a propriedade hereditária de *ser uma tautologia*, que é comum a todos os teoremas. Portanto, há pelo menos uma fórmula bem formada que não é teorema de CP. Conclusão: CP é consistente.

### 8.3.2. Prova da existência de proposições indecidíveis no Cálculo dos Principia Mathematica e sistemas semelhantes (1º Teorema da incompletude de Gödel):

Para demonstrar que há proposições indecidíveis em PM, Gödel teve de construir uma proposição muito peculiar, que simbolizaremos por *G*, a qual diz dela mesma que não é demonstrável num sistema P, que é equivalente a PM. Se *G* pudesse ser provada em P, teríamos uma contradição, pois, neste caso, estaríamos demonstrando não somente a verdade de *G*, mas também a sua falsidade. Com efeito, *G* diz de si própria que não é demonstrável. Ora, se ela for demonstrável, teremos, por um lado, *G* como teorema de P. Por outro lado, como *G* diz que não é demonstrável, sua demonstração implica a sua falsidade, o que significa que teremos também ~*G* como teorema de P. Se P admite *G* e ~*G* como teoremas, P é contraditório. Logo, *G* é verdadeira e não pode ser demonstrada em P. Gödel conseguiu mostrar também que, se ~*G* fosse demonstrável em P, então P seria ω-inconsistente. Em virtude disso, *G* é indecidível em P. Isto tem sérias conseqüências para o Programa de Hilbert, como veremos mais adiante.

A proposição *G* está ligada ao famoso *paradoxo do mentiroso*, mas não tem as suas imperfeições. O paradoxo do mentiroso pode ser formulado como segue. Suponhamos que um dado indivíduo A nos diga

*estou mentindo neste exato momento*.

Esta afirmação é verdadeira ou falsa? Se A está de fato mentindo neste exato momento, então A está dizendo a verdade. E se A está de fato dizendo a verdade neste exato momento, então A está mentindo. Conclusão:

*estou mentindo neste exato momento* é verdadeira se e somente se for falsa!

É o fato de a proposição acima referir-se a si própria que leva a uma contradição. Isto significa que pode haver outras formulações do paradoxo do mentiroso, todas elas levando a uma contradição. Seja, por exemplo, a proposição *p diz de si própria que p é falsa*.

Se *p* é verdadeira, então é falsa; se *p* é falsa, então é verdadeira. Portanto, *p* é verdadeira se e somente se *p* é falsa! Suponhamos agora as duas seguintes proposições:

*p1 diz de p2 que p2 é falsa*

e

*p2 diz de p1 que p1 é verdadeira.*

Como *p1* e *p2* fazem referência uma à outra, segue-se que se *p1* é verdadeira, então *p2* é falsa e, em conseqüência, *p1* é falsa; se *p1* é falsa, então *p2* é verdadeira e, em conseqüência, *p1* é verdadeira. Portanto, *p1* é verdadeira se e somente se *p1* é falsa! Sejam agora as seguintes três proposições:

*p1 diz de p2 que p2 é falsa*

e

*p2 diz de p3 que p3 é falsa*

e

*p3 diz de p1 que p1 é falsa*.

Agora criamos uma cadeia de três proposições em que cada uma faz referência a uma outra. E o resultado continua sendo uma contradição: *p1* é verdadeira se e somente se *p1* é falsa!

Este paradoxo pode ser estendido para uma lógica trivalente da seguinte maneira. Suponhamos a seguinte proposição, que pode ser ou verdadeira, ou falsa ou duvidosa:

*p diz de si própria que ou p é falsa ou p é duvidosa* .

Se *p* é falsa, então *p* é verdadeira, pois afirma de si própria que é ou *falsa* ou duvidosa. Se *p* é duvidosa, então *p* é verdadeira, pois afirma de si própria que é ou falsa ou *duvidosa*. Se *p* é verdadeira, então *p* não é verdadeira, pois afirma de si mesma que é ou falsa ou duvidosa. Em todos os casos, temos uma contradição, pois *p* nunca tem um único e inequívoco valor-verdade. Esta antinomia também pode ser expressa através dos valores-verdade *1*, *0* e *½*, embora essa formulação seja menos intuitiva.

O problema com as antinomias acima descritas está em que as proposições envolvidas sâo formuladas de maneiera ilícita. Com efeito, uma proposição que diz de si própria apenas que é verdadeira ou falsa não possui conteúdo definido. Retornemos à proposição *p* considerada acima:

*p diz de si própria que p é falsa*.

Para ser verdadeira ou falsa, uma proposição deve dizer algo além do que *p* está dizendo, como, por exemplo, que está chovendo, ou fazendo sol, etc. Com base neste conteúdo, podemos compará-la com a realidade e, se estiver de fato chovendo, por exemplo, podemos declará-la verdadeira. Uma proposição sem esse tipo de conteúdo, como *p*, não pode ser comparada com a realidade e, portanto, não tem um valor-verdade definido. Se, porém, não atentarmos para esse detalhe, aceitaremos *p* como proposição autêntica e cairemos em contradição.

Para superar essa dificuldade e formular uma proposição autêntica, que diz dela mesma que é indemonstrável, Gödel teve de criar um procedimento especial, através do qual a metateoria de P é mapeada no interior do próprio P. Este procedimento é conhecido como *numeração de Gödel* e consiste em construir uma correspondência biunívoca entre uma proposição de P e um número inteiro pertencente a P, de tal modo que a cada propriedade da proposição corresponderá uma propriedade daquele número inteiro. Como a propriedade da proposição pertence ao nível da metateoria de P e a propriedade do número inteiro correspondente pertence ao nível de P, a numeração de Gödel consegue estabelecer um mapeamento da metateoria de P no interior de P. Isto torna possível não apenas construir uma proposição metateórica que diz dela mesma que é indemonstrável em P, mas que também está mapeada numa proposição *G* de P que atribui uma propriedade a um número inteiro. Desse modo, a proposição *G* possui dois aspectos fundamentais: por um lado, enquanto pertencente a P, diz apenas algo sobre uma propriedade de um número em P; por outro, enquanto está indissoluvelmente ligada a uma proposição da metateoria de P, diz de si própria que não é demonstrável em P. Estes pontos serão esclarecidos no decorrer da exposição do teorema.

A numeração de Gödel se baseia numa tabela semelhante à que segue abaixo, que adaptamos às nossas necessidades:

| Signo primitivo | Número de Gödel | Significado intuitivo | Signo primitivo | Número de Gödel | Significado intuitivo |
|---|---|---|---|---|---|
| ~ | 1 | não | x | 11 | variável individual |
| V | 2 | ou | y | 12 | variável individual |
| ⊃ | 3 | se-então | z | 13 | variável individual |
| ∃ | 4 | existe pelo menos um... | p | 14 | variável sentencial |
| = | 5 | é igual a | q | 15 | variável sentencial |
| 0 | 6 | zero | r | 16 | variável sentencial |
| S | 7 | sucessor imediato de | P | 17 | predicado |
| ( | 8 | parêntese | Q | 18 | predicado |
| ) | 9 | parêntese | R | 19 | predicado |
| , | 10 | vírgula | | | |

O procedimento para estabelecer o número de Gödel de uma dada proposição de P consiste em construir um produto de fatores primos de modo tal que o expoente de cada fator corresponda ao número de Gödel de cada um dos signos primitivos usados na proposição de P. A seqüência dos expoentes deve corresponder à seqüência dos signos primitivos na proposição de P. Seja, por exemplo, a proposição

*(∃x)(x = s0)*.

Ela diz em P que o número zero tem pelo menos um sucessor. Os números de Gödel correspondentes aos signos primitivos utilizados, na ordem de aparecimento, são os seguintes:

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| ( | ∃ | x | ) | ( | x | = | s | 0 | ) |
| 8 | 4 | 11 | 9 | 8 | 11 | 5 | 7 | 6 | 9 |

O número de Gödel de *(∃x)(x = s0)*, neste caso, é:

$2^8 \times 3^4 \times 5^{11} \times 7^9 \times 11^8 \times 13^{11} \times 17^5 \times 19^7 \times 23^6 \times 29^9$.

Suponhamos que esse produto seja igual a *n*. Trata-se, sem dúvida, de um número muito grande, mas isto não vem ao caso. O importante é que a cada proposição de P corresponde um e somente um número inteiro *n* desse tipo e a cada número inteiro *n* desse tipo corresponde uma e somente uma proposição de P. Graças a isso, a numeração de Gödel é um processo reversível: partindo de *(∃x)(x = s0)*, chegamos a *n* através do produto $2^8 \times 3^4 \times 5^{11} \times 7^9 \times 11^8 \times 13^{11} \times 17^5 \times 19^7 \times 23^6 \times 29^9$; partindo de *n*, chegamos a *(∃x)(x = s0)* através da decomposição de *n* em fatores primos, ou seja, através de $2^8 \times 3^4 \times 5^{11} \times 7^9 \times 11^8 \times 13^{11} \times 17^5 \times 19^7 \times 23^6 \times 29^9$.

O mais importante na realização deste processo está em que agora podemos mapear uma proposição metateórica de P numa proposição no interior de P. Seja, por exemplo, a proposição

*(∃x)(x = s0)* começa com o signo *(*.

Essa proposição é metateórica com respeito a P, pois atribui uma propriedade a uma proposição de P. Como *(∃x)(x = s0)* tem o número de Gödel *n*, esta proposição metateórica pode ser colocada em correspondência com uma proposição no interior de P que atribui uma determinada propriedade aritmética ao número *n*. Esta proposição é, por exemplo:

O expoente do fator primo *2* da decomposição de *n* em fatores primos é *8*.

Com efeito, se *(∃x)(x = s0)* começa com o signo *(*, então o expoente do primeiro fator primo da decomposição de *n*, ou seja, *2*, é *8*. Mais ainda: se o expoente do primeiro fator primo da decomposição de *n* é *8*, então *(∃x)(x = s0)* começa com o signo *(*. Isto significa que a proposição metateórica acima sobre *(∃x)(x = s0)* é verdadeira se e somente se a proposição de P sobre *n* é verdadeira. Como se pode ver, a metateoria de P fica devidamente mapeada no interior de P. A tabela abaixo mostra algumas correspondências obtidas através do processo de mapeamento com a numeração de Gödel:

| Proposição da metateoria de P | Proposição no interior de P |
|---|---|
| *(∃x)(x = s0)* termina com o signo *)* | o expoente do fator primo 29 da decomposição de n é 9 |
| a expressão *(∃x)* ocorre no início de *(∃x)(x = s0)* | os expoentes dos fatores primos 2, 3, 5 e 7 da decomposição de n são respectivamente 8, 4, 11 e 9 |
| a variável *x* ocorre duas vezes em *(∃x)(x = s0)* | o expoente 11 ocorre duas vezes na decomposição de n em fatores primos |
| a o termo *s0* está presente em *(∃x)(x = s0)* | os expoentes 7 e 6 estão presentes nesta seqüência na decomposição de n em fatores primos |
| o termo *s0* está na oitava posição em *(∃x)(x = s0)* | os expoentes dos fatores primos 19 e 23 da decomposição de n são respectivamente 6 e 7 |
| a variável *x* ocorre na 3ª e na 6ª posição em *(∃x)(x = s0)* | os expoentes dos fatores primos 5 e 13 da decomposição de n possuem o mesmo valor, que é 11 |

A numeração de Gödel também pode ser aplicada a uma prova. Seja, por exemplo, a demonstração abaixo, pertencente a P:

| Etapa | Justificativa |
|---|---|
| 1. (p v p) ⊃ p | Ax1 |
| 2. (~p v ~p) ⊃ ~p | 1 sub p/~p |
| 3. (p ⊃ ~p) ⊃ ~p | 2 def de *v* |

Para atribuir-lhe um número de Gödel, consideramos apenas as expressões da coluna da esquerda, na ordem em que se apresentam, de tal modo que a última proposição é a tese demonstrada. Ao lado de cada uma colocamos o correspondente número de Gödel:

$(p \vee p) \supset p \Leftrightarrow 2^{8} \times 3^{14} \times 5^{2} \times 7^{14} \times 11^{9} \times 13^{2} \times 17^{14} = m$

$(\sim p \vee \sim p) \supset \sim p \Leftrightarrow 2^{8} \times 3^{1} \times 5^{14} \times 7^{2} \times 11^{1} \times 13^{14} \times 17^{9} \times 19^{3} \times 23^{1} \times 29^{14} = n$

$(p \supset \sim p) \supset \sim p \Leftrightarrow 2^{8} \times 3^{14} \times 5^{3} \times 7^{1} \times 11^{14} \times 13^{9} \times 17^{3} \times 19^{1} \times 23^{14} = j$

Isto feito, podemos agora construir o número da seqüência, que é um novo produto de fatores primos, cujos expoentes são os números das proposições de cada etapa da demonstração, obedecendo à ordem em que se apresentam:

$2^{m} \times 3^{n} \times 5^{j} = k.$

Aqui também, dada a seqüência representada pelas proposições acima, temos um e somente um número de Gödel *k* que corresponde a ela e vice-versa. As propriedades metateóricas das proposições da seqüência correspondem a propriedades do número *k* que podem ser expressas em P. Chamando de *S* à seqüência de três proposições na tabela acima, que representa a demonstração de *(p ⊃~p) ⊃~p* em P, temos, por exemplo:

| Proposição metateórica sobre P | Proposição relativa a números em P |
|---|---|
| a seqüência *S* se compõe de três proposições | o número k se decompõe em três fatores primos |
| *(p v p) ⊃ p* é a primeira sentença da seqüência *S* | o expoente do primeiro fator primo da decomposição de k é m |
| *(~p v ~p) ⊃ ~p* é a segunda sentença da seqüência *S* | o expoente do segundo fator primo da decomposição de k é n |
| a sequência *S* constitui uma demonstração da sentença *(p ⊃~p) ⊃~p* | o expoente do último fator primo da decomposição de k é j |

Consideremos a última linha da tabela acima. Como se pode ver ali, temos, do lado esquerdo, uma sentença metateórica sobre P, dizendo que a seqüência *S* constitui uma demonstração da sentença *(p ⊃~p) ⊃~p*. Do lado direito, temos uma sentença matemática correspondente, exprimível em P. Com efeito, se *j* é o número de Gödel de $(p \supset \sim p) \supset \sim p$ e $k = 2^m \times 3^n \times 5^j$ é o número de Gödel da seqüência *S*, então há uma relação matemática entre *k* e *j*, que pode ser expressa através da seguinte sentença de P: *o expoente do último fator primo da decomposição de k em fatores primos é j*. Esta sentença estabelece uma relação entre dois números, envolvendo, portanto, um predicado de dois lugares, que poderia ser representado assim em P:

Dem(k, j).

Ela diz em P que o número *k* está na relação matemática *Dem* com o número *j*. Esta sentença está ligada à sentença metateórica a respeito de P:

A seqüência *S*, de número *k*, demonstra em P a sentença $(p \supset \sim p) \supset \sim p$, de número *j*.

Generalizando para dois números quaisquer, teríamos:

Dem(x, y).

Esta sentença diz em P que o número *x* está na relação matemática *Dem* com o número *y* e corresponde à seguinte sentença metateórica a respeito de P:

A seqüência de número *x* demonstra em P a sentença de número *y*.

A negação da sentença acima nos daria:

~Dem(x, y),

que diz em P:

O número *x* não está na relação matemática *Dem* com o número *y*.

Isto corresponde à seguinte sentença metateórica a respeito de P:

A seqüência de número *x* não demonstra em P a sentença de número *y*.

Se aplicássemos o quantificador universal à variável *x* dessa última sentença, obteríamos

(x)~Dem(x, y),

que significa, em P,

Para todo *x,* o número *x* não está na relação matemática *Dem* com o número *y*

e corresponde à sentença metateórica sobre P:

Para todo *x*, a seqüência de número *x* não demonstra em P a sentença de número *y*,

ou seja,

A sentença de número *y* não é demonstrável em P.

Como se pode ver, a numeração de Gödel permite que expressemos, no interior de P, uma sentença sobre uma relação matemática entre dois números que corresponde indissoluvelmente a uma sentença metateórica sobre indemonstrabilidade de uma sentença de P. Mais ainda: a numeração de Gödel permite que expressemos, no interior de P, uma sentença sobre uma relação entre dois números que corresponde indissoluvelmente a uma sentença metateórica que afirma a própria indemonstrabilidade em P. Para construir essa sentença, temos inicialmente de definir a função matemática *sub*. Seja, por exemplo, a sentença de P $x = sy$, cujo número de Gödel é $m$. Esta sentença nos diz em P que o número $x$ é sucessor do número $y$. A aplicação da função *sub* a esta sentença consiste em substituir a variável $y$ pelo próprio número $m$, de tal modo que, ao invés de $x = sy$, tenhamos $x = sm$. Esta nova sentença nos diz em P que o número $x$ é o sucessor do número $m$. A operação da função *sub* pode ser representada por *sub(m, 12/m)* e pode ser lida assim: o número de Gödel da sentença que resulta quando se substitui, na sentença de número $m$ — isto é, na sentença $x = sy$ — , a variável de número *12* — isto é, $y$ — pelo próprio número $m$. O processo aqui também é reversível: partindo da sentença de número $m$, chegamos à sentença de número *sub(m, 12/m)* e, partindo desta última, chegamos à primeira. É importante observar que, em P, *sub(m, 12/m)* é um número como qualquer outro. Em termos mais intuitivos, a função *sub* corresponde à operação de, na função $f_n(x)$, substituir a variável $x$ pelo próprio número $n$ que se encontra no índice de $f_n$, gerando o valor $f_n(n)$ para a função em questão.

Podemos agora dar o primeiro passo em direção à proposição de P que estabelece uma relação entre números e ao mesmo tempo está indissoluvelmente ligada a uma sentença da metateoria de P que diz dela mesma que não é demonstável. Seja a proposição

(x)~Dem[x, sub(y, 12/y)].

Ela diz, em P:

Para todo $x$, $x$ não está na relação matemática dem com o número *sub(y, 12/y)*.

Esta sentença corresponde à seguinte sentença metateórica sobre P:

A sentença de número de Gödel *sub(y, 12/y)* não é demonstrável em P.

Suponhamos que o número de Gödel de *(x)~Dem[x, sub(y, 12/y)]* seja $n$. Estamos agora em condições de construir uma proposição de P que está ligada a uma proposição da metateoria de P que diz dela mesma ser indemonstrável em P. Esta proposição é a seguinte:

(x)~Dem[x, sub(n, 12/n)].

Se utilizássemos o quantificador existencial, teríamos a seguinte proposição equivalente:

~(∃x)Dem[x, sub(n, 12/n)].

Abreviemos qualquer uma destas formulações, seja usando o quantificador universal, seja usando o quantificador existential, por *G*. Em P, *G* diz que

Para todo *x*, o número *x* não está na relação matemática *Dem* com o número *sub(n, 12/n)*.

Na metateoria de P, *G* está ligada à seguinte proposição:

A proposição que resulta da substituição, na proposição de número *n*, da variável de número 12 — isto é, *y* — pelo próprio número *n* é indemonstrável em P.

*(x)~Dem[x, sub(n, 12/n)]* diz que, quando se toma a sentença de número *n* e se substitui a variável de número 12 pelo próprio número *n* da sentença, obtém-se uma sentença indemonstrável em P. Vejamos se isto é verdade. Tomemos a sentença de número *n* e vejamos o que acontece com ela quando fazemos a substituição indicada. A proposição de número *n* é, como vimos acima,

(x)~Dem[x, sub(y, 12/y)].

Substituindo nesta proposição a variável *y* pelo próprio número *n* da proposição em questão, obtemos:

(x)~Dem[x, sub(n, 12/n)].

Ora, esta é exatamente a proposição *G*. Além disso, se o número da proposição *(x)~Dem[x, sub(y, 12/y)]*, da qual partimos, é *n*, então o número de *G* é *sub(n, 12/n)*! Com efeito, o número da proposição *G* é o número que resulta quando, na proposição de número *n*, substituímos a variável *y* pelo próprio número *n*. Seguimos as instruções da proposição *G*, procurando a proposição de número *n* e fazendo a substituição da variável *y* por *n*. O resultado obtido foi a própria proposição *G*. Isto significa que, no interior de P, a proposição *G*, cujo número de Gödel é *sub(n, 12/n)*, afirma que nenhum número *x* está na relação matemática *Dem* com o número *sub(n, 12/n)*. Por causa disso, *G* está ligada a uma proposição metateórica sobre P que afirma ser indemonstrável em P a proposição de número *sub(n, 12/n)*, ou seja, a própria *G*. Diferentemente das proposições ligadas ao paradoxo do mentiroso, essa proposição possui conteúdo definido e pode ser declarada verdadeira por comparação com a realidade.

Isto posto, podemos mostrar agora que nem a proposição *G* nem sua negação ~*G* são demonstráveis em P. Em outras palavras, *G* é indecidível em P. A prova disso pode ser estabelecida em duas etapas. Em primeiro lugar, suponhamos que *G* seja demonstrável em P. Neste caso, existe uma seqüência de fórmulas tal que a última delas é *G*. Tal seqüência possui, assim, um número de Gödel *x* que está na relação matemática *Dem* com o número *sub(n, 12/n)*. Em símbolos:

(∃x)Dem[x, sub(n, 12/n)].

Ora, esta proposição equivale a ~*G*, uma vez que, como vimos acima, *G* com o quantificador existencial equivale a

~(∃x)Dem[x, sub(n, 12/n)].

Deste modo, se *G* fosse demonstrável, sua negação ~*G* também o seria e P seria inconsistente. Como P é consistente, *G* não é demonstrável. E isto significa que *G* é verdadeira.

Em segundo lugar, suponhamos que ~*G* seja demonstrável. Neste caso, a proposição

(∃x)Dem[x, sub(n, 12/n)]

seria um teorema de P, afirmando que existe pelo menos um número *x* que se acha na relação matemática *Dem* com o número *sub(n, 12/n)*. Como sabemos que *G* é verdadeira, sabemos também que nenhum número será capaz de satisfazer a relação matemática *Dem* como número *sub(n, 12/n)*. Em virtude disso, seria também possível provar em P:

~Dem[0, sub(0, 12/0)]

~Dem[1, sub(1, 12/1)]

~Dem[2, sub(2, 12/2)]

~Dem[n, sub(n, 12/n)]

etc.

Ou seja, embora ~*G* afirme que existe pelo menos um número que se encontra na relação matemática *Dem* com o número *sub(n, 12/n)*, poderíamos mostrar que este número não é *0*, não é *1*, não é *2*, não é *n*, etc. Neste caso, P seria ω-inconsistente. Como este também não é o caso, segue-se que ~*G* também não pode ser demonstrada em P.

Assim, nem *G* nem sua negação ~*G* podem ser demonstradas em P e *G* é de fato indecidível. Poderíamos, como último recurso, acrescentar *G* ao conjunto dos axiomas de P, com o objetivo de salvaguardar a conexão que acreditamos existir entre o *verdadeiro* e o *demonstrável*. Enquanto axioma, *G* continuaria indemonstrável, mas pelo menos faria parte de P, cujas proposições válidas passariam a abarcar novamente todas as proposições verdadeiras. Infelizmente, esta solução também não seria adequada, porque na verdade estaríamos assim construindo um novo sistema P', resultante da adição de *G* a P. Nesse novo sistema, poderíamos construir uma nova proposição verdadeira e indemonstrável, que simbolizaremos por *G'*. Para tanto, teríamos de construir em P' um novo predicado, *Dem'(x, y)*, que corresponderia ao predicado metateórico

a seqüência de número *x demonstra em P'* a sentença de número *y*.

Isto é possível, porque agora não estamos mais lidando com o sistema P, e sim com o sistema P', que envolve relações matemáticas diferentes daquelas de P, em virtude da adição de *G* ao conjunto dos axiomas de P. Do mesmo modo que o predicado matemático *Dem(x, y)*, que foi construído em P tomando como referência os números associados aos axiomas de P, o predicado *Dem'(x, y)* seria construído tomando como referência os números associados aos axiomas de P', que são diferentes dos de P. Com base nesse novo predicado, seria possível construir *G'*, uma proposição indecidível em P'. Se P' for consistente, então esta proposição seria verdadeira, mas não demonstrável em P'. Se sua negação, *~G'*, for demonstrável em P', então este sistema seria ω-inconsistente. A consistência de P' não poderia ser demonstrada no interior do sistema. Do ponto de vista da incompletude, o novo sistema P' estaria em situação semelhante a P. Se tentássemos resolver o problema através da adição de *G'* aos axiomas de P', cairíamos numa situação idêntica. Com efeito, estaríamos criando assim um novo sistema, P'', que admitiria uma nova proposição indecidível *G''* através do novo predicado *Dem''(x, y)* e seria também incompleto. Esse processo de juntar a última fórmula indecidível aos axiomas do sistema poderia repetir-se indefinidamente, sempre com o mesmo resultado. O teorema de Gödel nos mostra, assim, que o sistema P, além de incompleto, é incompletável.

**8.3.4. Não é possível provar, no interior do Cálculo dos Principia Mathematica e sistemas semelhantes, a consistência do próprio sistema (2º Teorema de Incompletude de Gödel):**

O teorema de Gödel tem uma conseqüência muito importante do ponto de vista da teoria da demonstração. Com efeito, ele permite concluir que, se um sistema é suficientemente amplo para conter a aritmética, não será possível provar, no interior deste sistema, a sua própria consistência. O corolário pode ser assim formulado: se o sistema P é consistente, então a proposição que representa em P a sua própria coerência não é demonstrável em P. Vejamos como é possível chegar a essa conclusão.

Sabemos que P é consistente se e somente se existe pelo menos uma proposição bem formada de P que não é demonstrável em P. Isto pode ser simbolizado como segue no interior de P:

$(\exists y)(x){\sim}Dem(x, y).$

Representemo-la por *P*. Esta proposição afirma em P que existe pelo menos um número *y* tal que, para todo *x*, *x* não está na relação *Dem* com *y* e corresponde à proposição metateórica sobre P que afirma a existência de pelo menos uma proposição bem formada de P que não é demonstrável em P.

Podemos agora mostrar que $P \supset G$. O que justifica isso é a relação abaixo, válida em P:

$(\exists y)(x){\sim}Dem(x, y) \supset (x){\sim}Dem[x, sub(n, 12/n)].$

Nela, com efeito, simplesmente passamos de uma expressão da forma $(\exists y)(x){\sim}F(x, y)$ para outra da forma $(x){\sim}F(x, a)$, em que o quantificador existencial é eliminado através da substituição da variável *y* por uma constante *a*. Trata-se da operação de *instanciação existencial*, inteiramente válida no interior de P. Deste modo, podemos não só representar no interior de P a proposição $P \supset G$, mas também demonstrá-la.

As correspondências que acabamos de estabelecer podem ser esquematizadas através da seguinte tabela:

| Proposição sobre números no interior de P | Proposição metateórica sobre P |
|---|---|
| *P* ou (∃y)(x)~Dem(x, y)<br>[existe pelo menos um número *y* tal que, para todo *x*, *x* não está na relação matemática *Dem* com *y*] | o sistema P é consistente |
| *G* ou (x)~Dem[x, sub(n, 12/n)]<br>[para todo *x*, *x* não está na relação matemática *Dem* com o número *sub(n, 12/n)*] | *G* não é demonstrável em P |
| *P* ⊃ *G* ou (∃y)(x)~Dem(x, y) ⊃ (x)~Dem[x, sub(n, 12/n)]<br>[se existe pelo menos um número *y* tal que, para todo *x*, *x* não está na relação matemática *Dem* com *y* então para todo *x*, *x* não está na relação matemática *Dem* com o número *sub(n, 12/n)*] | se P é consistente, então *G* não é demonstrável em P |

Sabemos, portanto, que *P* e *G* são representáveis no interior de P e que *P* ⊃ *G* é derivável em P. Ora, isto significa que, se *P* fosse demonstrável em P, poderíamos, pela aplicação da regra de separação, também demonstrar *G* em P, como se pode ver pelo esquema abaixo:

| Etapa | Justificativa |
|---|---|
| 1. P ⊃ G | teorema de P |
| 2. P | premissa adicional |
| 3. G | 1, 2 separação |

Como *G* não pode ser demonstrada em P, concluímos que *P*, também, não pode ser demonstrada em P. Deste modo, é válido em P o corolário do teorema de Gödel: se um sistema suficientemente amplo para conter a aritmética como o sistema P é consistente, então a proposição que representa em P a sua própria coerência não é demonstrável em P.

O corolário do teorema de Gödel explica o fracasso das tentativas feitas, entre 1924 e 1932, para demonstrar a consistência da aritmética nas bases estabelecidas por Hilbert. Todas essas tentativas utilizaram um sistema formal F que satisfaz as condições necessárias para a construção da proposição $G$ em seu interior — incluir a lógica de predicados de primeira ordem e os axiomas de Peano relativos à aritmética. Neste caso, não só $P$ e $G$ são representáveis no interior de F, mas também $P \supset G$ é derivável em F. Isto significa que é impossível demonstrar a consistência do próprio sistema F com recursos exclusivamente pertencentes F, como exige o programa de Hilbert. Um sistema como P ou F é capaz de provar as propriedades principais dos números naturais e é por isso denominado *fundamental*. Assim, para demonstrar a consistência de um sistema fundamental, teremos de utilizar procedimentos de raciocínio que não podem ser formalizados nestes sistemas. Por exemplo, a proposição $P$, que afirma a consistência de P, não pode ser demonstrada em P, mas pode ser formulada e demonstrada na teoria dos conjuntos de Zermelo-Fraenkel. Isto significa que há proposições envolvendo apenas a noção de *número natural*, ou seja, proposições que podem ser formuladas na linguagem da aritmética de primeira ordem, as quais só podem ser demonstradas através de conceitos mais poderosos, como, por exemplo, a teoria dos conjuntos. Uma teoria fundamental não pode justificar a si própria: ela é ou inconsistente ou incapaz de resolver alguns de seus problemas.[15]

O teorema de Gödel nos permite também dizer onde termina a lógica e onde começa a matemática: na introdução da teoria dos números, logo após a teoria da quantificação. Com efeito, o cálculo proposicional e o cálculo de predicados são sistemas completos, enquanto o cálculo dos *Principia Mathematica*, que inclui não só os cálculos proposicional e de predicados, mas também a teoria dos números, é incompleto.

Do ponto de vista filosófico, o teorema de Gödel põe em cheque qualquer tentativa de formalizar o Saber Absoluto, uma vez que tal saber envolveria a inclusão de uma teoria fundamental que, como sabemos, não passa de um sistema incompleto e indecidível.

## 8.4. Exercícios de Metalógica

8.4.1. Com base na definição de ω-consistência, mostre que, se um sistema formal é inconsistente, ele também é ω-inconsistente, mas que, se ele é ω-inconsistente, ele não é necessariamente inconsistente.
8.4.2. Sabemos que a proposição *p afirma de si própria que p é falsa* leva a uma contradição. Mostre agora o que acontece com a proposição *q afirma de si própria que q não é verdadeira*.
8.4.3. Construa uma antinomia com base numa cadeia de duas proposições em que a primeira delas é *q1 afirma de q2 que q2 não é verdadeira*.
8.4.4. Construa uma antinomia com base numa cadeia de três proposições em que a primeira delas é *q1 afirma de q2 que q2 é verdadeira*.
8.4.5. Em lógica trivalente, a proposição *p diz de si própria que ou p é falsa ou p é duvidosa* leva a uma contradição. Construa uma antinomia análoga em lógica tetravalente, com os valores-verdade 1, 2, 3 e 4, sendo 1 e 4 os valores atribuídos.
8.4.6. Nos moldes do exercício anterior, construa uma antinomia em lógica pentavalente, com os valores-verdade 1, 2, 3, 4 e 5, sendo 1 e 5 os valores atribuídos.
8.4.7. Com base na numeração de Gödel dada pela tabela anteriormente apresentada, construa os números de Gödel das seguintes expressões:
[Dica: lista dos números primos até 41: 2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41]

| | |
|---|---|
| 8.4.7.1. (p⊃ ~ p) ⊃ ~ p | 8.4.7.6. (∃x)P(x) |
| 8.4.7.2. q ⊃ (p ⊃ q) | 8.4.7.7. ~(x)Q(x) |
| 8.4.7.3. (p ⊃ ~ q) ⊃ (q ⊃ ~ p) | 8.4.7.8. (x)(y)R(x, y) |
| 8.4.7.4. ~(q ⊃ (p ⊃ r) | 8.4.7.9 (∃y)(x)Q(x, y) |
| 8.4.7.5. ~ q v (~ p v r) | 8.4.7.10. (x)(P(x) & Q(x)) |

[15] Ver Podnieks, Karlis. *What is Mathematics: Gödel's Theorem and Around*. Disponível em http://www.ltn.lv/~podnieks/gt.html. Acesso em 01/01/2006.

8.4.8. Com base na mesma tabela, construa as expressões correspondentes aos seguintes números de Gödel:

| | |
|---|---|
| 8.4.8.1. $2^{14}$ | 8.4.8.6. $2^{1} \times 3^{16}$ |
| 8.4.8.2. $2^{15}$ | 8.4.8.7. $2^{14} \times 3^{2} \times 5^{15}$ |
| 8.4.8.3. $2^{16}$ | 8.4.8.8. $2^{14} \times 3^{2} \times 5^{1} \times 7^{14}$ |
| 8.4.8.4. $2^{1} \times 3^{14}$ | 8.4.8.9. $2^{14} \times 3^{3} \times 5^{15}$ |
| 8.4.8.5. $2^{1} \times 3^{15}$ | 8.4.8.10. $2^{14} \times 3^{3} \times 5^{8} \times 7^{14} \times 11^{2} \times 13^{15} \times 17^{9}$ |

8.4.9. Cada uma das meta-sentenças abaixo está descrevendo uma sentença correspondente de PM. Construa para cada uma destas meta-sentenças uma sentença matemática que seja capaz de espelhar através duma propriedade numérica a propriedade metateórica apresentada.

8.4.9.1. apenas a variável proposicional *p* está presente na sentença *p*

8.4.9.2. a sentença *q* se compõe de um único símbolo

8.4.9.3. a variável proposicional *p* não está presente na sentença *r*

8.4.9.4. a sentença *~p* possui apenas dois signos primitivos

8.4.9.5. a sentença *~q* principia pela negação

8.4.9.6. na sentença *~r*, a variável proposicional *r* está sendo negada

8.4.9.7. a sentença *p v q* se compõe de apenas três signos primitivos

8.4.9.8. a variável proposicional *p* ocorre duas vezes na sentença *p v ~p*

8.4.9.9 a sentença $p \supset q$ possui o símbolo da implicação na segunda posição

8.4.9.10. a expressão *(p v q)* faz parte da sentença $p \supset (p \vee q)$

**Bibliografia**


Blanché, R. *Introduction à la Logique Contemporaine*. 2 ed. Paris: Lib. Armand Colin, 1957. 208 p.

Bochenski, I.M. *Historia de la logica formal*. Ed. Espanola de M B Lozano. Madrid, Editorial Gredos, 1966.

da Costa, N. C. A. *Lógica Paraconsistente Aplicada*. S. Paulo: Ed. Atlas S.A., 1999. 214 p.

da Costa, N. C. A. *Sistemas Formais Inconsistentes.* Curitiba: Ed. da UFPR, 1993. 66 p.

da Costa, N. C. A. *Ensaio sobre os fundamentos da lógica*. 2 ed. S. Paulo: Hucitec, 1994.

Girle, R. *Modal Logics and Philosophy*. Montreal, London, Ithaca: McGill-Queen's University Press, 2000.

Haack, S. *Deviant Logic, Fuzzy Logic. Beyond the Formalism.* Chicago: Un. of Chicago Press, 1996. 292 p.

Haack, S. *Filosofia das Lógicas*. Trad. C. A. Mortari e L. H. A. Dutra. S. Paulo: Ed. UNESP, 2002.

Heijenoort, J. van (ed.). *From Frege to Gödel. A source Book in Mathematical Logic, 1879-1931*. Cambridge, Mass. London, England: Harvard Un. Press, 1967.

Hilbert, D & Ackermann, W. *Elementos de logica teorica*. Trad. de V. de Zavala, 2 ed, Madrid, Ed Tecnos, 1968.

Hughes, G. E. & Cresswell, M. J. *A New Introduction to Modal Logic.* London and N. York: Routledge, 1996. 422 p.

Kaehler, S. D. *Fuzzy Logic. An Introduction.* Kaehler, S. *Fuzzy Logic. An Introduction.* Disponível em <http://www.seattlerobotics.org/encoder/mar98/fuz/flindex.html>. Acesso em 12 Jan. 2003.

Kleene, S. C. *Introduction to Metamathematics*. Amsterdam: North Holland Pub. Co. Groningen: P. Noordhoff N. V. 1952.

Kneale, W. & Kneale, M. *O Desenvolvimento da Lógica*. 3 ed. Lisboa: Fundação Calouste Gulbenkian, 1991.

Körner, s. *Uma introdução à filosofia da matemática*. Rio: Zahar, 1985.

Ladrière, J. *Limitaciones Internas de los Formalismos*. Madrid: Ed. Tecnos, s/d.

Lewis, C I & Langford, C H. *Symbolic logic*. 2 ed, N. York, Dover Publications Inc., 1959.

Mortari, C. A. *Introdução à Lógica.* São Paulo: Editora UNESP; Imprensa Oficial do Estado, 2001.

Nagel, E. & Newman, J. R. *Prova de Gödel*. S. Paulo: Ed. Perspectiva, 1973.

Pinto, P. R. M. *Introdução à Lógica Simbólica*. Belo Horizonte: Editora UFMG, 2001.

Priest, G. *Non-Classical Logic.* Cambridge: Cambridge Un. Press, 2001. 242 p.

Russell, B. & Whitehead, A. N. *Principia Mathematica to *56*. Cambridge, Un. Press, 1964.

Smullyan, R. *O Enigma de Sherazade*. Rio: Jorge Zahar Ed., 1998.