Revista Eletrônica de Filosofia
Philosophy Eletronic Journal
ISSN 1809-8428

São Paulo: Centro de Estudos de Pragmatismo
Programa de Estudos Pós-Graduados em Filosofia
Pontifícia Universidade Católica de São Paulo
Disponível em http://www.pucsp.br/pragmatismo

Vol. 9, nº. 2, julho-dezembro, 2012, p. 227-242

A LÓGICA PROPOSICIONAL DO 'QUASE SEMPRE'

Angela Pereira Rodrigues
Universidade Estadual Paulista "Júlio de Mesquita filho" – UNESP/Campus de Marília
Marília – SP – Brasil
angela.p.rodrigues@bol.com.br

Hércules de Araújo Feitosa
Universidade Estadual Paulista "Júlio de Mesquita filho" – UNESP/Campus de Bauru
Bauru – SP – Brasil
haf@fc.unesp.br

Resumo: Introduzimos, neste trabalho, uma lógica proposicional para tratar da noção de 'quase sempre'. Fazemos isso através da inclusão de um novo operador na lógica proposicional clássica. Este novo operador captura as mesmas noções do quantificador introduzido na lógica do ultrafiltro de Sette, Carnielli e Veloso (1999).

Palavras-chave: Lógica do ultrafiltro. Lógica proposicional do 'quase sempre'. Álgebra do 'quase sempre'.

THE PROPOSITIONAL LOGIC FOR 'ALMOST ALWAYS'

Abstract*: We introduce, in this paper, a propositional logic to deal with de notion of 'almost always'. We do this by adding a new operator in classical propositional logic. This new operator captures the same notions of the quantifier introduced in the Ultrafilter logic by Sette, Carnielli and Veloso (1999).*

Keywords*: Ultrafilter logic. 'Almost always' propositional logic. 'Almost always' algebra.*

***

Introdução

R. Reiter, em 1980, introduziu a Lógica do Padrão. A preocupação de Reiter (1980) foi com argumentos padrões do tipo "na ausência de alguma informação contrária, assume-se que...", ou seja, argumentos que 'quase sempre' são verdadeiros, com algumas exceções.

Uma crítica relevante sobre a Lógica do Padrão de Reiter é que este sistema não é monotônico. Em todas as deduções feitas num sistema não-monotônico, precisamos analisar todas as regras que formalizam as crenças existentes, as suposições iniciais e os teoremas já deduzidos para que não surja alguma inconsistência no sistema.

Por esta e outras desvantagens dos sistemas não-monotônicos, Sette, Carnielli e Veloso (1999) desenvolveram um sistema monotônico, baseado no conceito de ultrafiltro, a fim de oferecer uma alternativa à Lógica do Padrão.

A Lógica do Ultrafiltro é formalizada numa linguagem $L(\nabla)$ que é uma extensão da linguagem clássica de primeira ordem L, obtida através da inclusão do quantificador generalizado ∇. Denotamos a Lógica do Ultrafiltro por $\mathcal{L}(\nabla)$.

Os axiomas de $\mathcal{L}(\nabla)$ são formados por todos os axiomas da lógica clássica de primeira ordem $\mathcal{L}$ mais os seguintes axiomas:

(∇_1) $\forall x\ (\varphi(x) \to \psi(x)) \to (\nabla x\ \varphi(x) \to \nabla x\ \psi(x))$

(∇_2) $(\nabla x\ \varphi(x) \wedge \nabla x\ \psi(x)) \to \nabla x\ (\varphi(x) \wedge \psi(x))$

(∇_3) $\nabla x\ \varphi(x) \vee \nabla x\ \neg\varphi(x)$

(∇_4) $\nabla x\ \varphi(x) \to \exists x\ \varphi(x)$.

As regras de inferência do sistema são:

Modus Ponens (MP): $\varphi, \varphi \to \psi \vdash \psi$

Generalização (Gen): $\varphi \vdash \forall x\ \varphi(x)$.

Neste trabalho, assim como feito na Lógica do Ultrafiltro, introduziremos uma lógica para tratar do conceito de 'quase sempre'. Porém, ao invés de trabalharmos num ambiente quantificacional, faremos isto num ambiente proposicional.

1. A lógica proposicional do 'quase sempre'

A lógica proposicional do 'quase sempre', denotada por $\mathcal{L}(☼)$, é uma extensão da lógica proposicional clássica. Assim, todos os resultados da lógica proposicional clássica, ou cálculo proposicional clássico (CPC), são também resultados de $\mathcal{L}(☼)$. Além destes resultados, teremos outros dados pelo novo operador lógico, que captura a noção de 'quase sempre'.

Indicaremos o conjunto de variáveis proposicionais de $\mathcal{L}(☼)$ e o conjunto de fórmulas de $\mathcal{L}(☼)$, respectivamente, por **Var**$\mathcal{L}(☼)$ e **For**$\mathcal{L}(☼)$.

A *lógica proposicional do 'quase sempre'* é construída sobre a linguagem $L(\wedge, \vee, \to, \neg, ☼)$, em que $\wedge$, $\vee$, $\to$ e $\neg$ são os conectivos lógicos usuais e ☼ é um novo operador. A lógica $\mathcal{L}(☼)$ fica determinada por meio dos seguintes axiomas e regras de dedução:

Axiomas:

(Ax_0) Axiomas do cálculo proposicional clássico (CPC)

(Ax_1) $(☼\varphi \wedge ☼\psi) \rightarrow ☼(\varphi \wedge \psi)$

(Ax_2) $☼\varphi \vee ☼\neg\varphi$

(Ax_3) $☼\bot \rightarrow \bot$.

Regras de dedução:

(MP) $\varphi \rightarrow \psi, \varphi / \psi$

(R☼) $\vdash \varphi \rightarrow \psi / \vdash ☼\varphi \rightarrow ☼\psi$.

Observações: Denotamos que a fórmula σ é deduzida a partir do conjunto Γ por $\Gamma \vdash \sigma$. Quando o conjunto Γ é vazio, a expressão $\vdash \sigma$ denota que a fórmula σ é um teorema de $\mathcal{L}(☼)$. Além disso, nossos axiomas e regras de dedução são esquemas, ou seja, φ e ψ representam fórmulas quaisquer.

Os novos axiomas e regra de dedução podem ser entendidos da seguinte maneira:

(Ax_1) Se φ ocorre 'quase sempre' e ψ ocorre 'quase sempre', então $\varphi\wedge\psi$ ocorre 'quase sempre';

(Ax_2) φ ocorre 'quase sempre' ou $\neg\varphi$ ocorre 'quase sempre';

(Ax_3) Se uma contradição ocorre 'quase sempre', então a contradição ocorre;

(R☼) Se ψ ocorre quando φ ocorre, então ψ ocorre 'quase sempre' quando φ ocorre 'quase sempre'.

Proposição 1.1: $\vdash \neg☼\bot$.

Demonstração:

1. $\vdash ☼\bot \rightarrow \bot$	(Ax_3)
2. $\vdash (☼\bot \rightarrow \bot) \rightarrow (\neg\bot \rightarrow \neg☼\bot)$	(CPC)
3. $\vdash \neg\bot \rightarrow \neg☼\bot$	(MP) em 1 e 2
4. $\vdash \neg\bot$	(CPC)
5. $\vdash \neg☼\bot$	(MP) em 3 e 4. ■

Na proposição anterior, temos que a contradição não ocorre 'quase sempre'. Este é um resultado intuitivo para a noção de 'quase sempre' e desejável em nosso sistema. Na proposição a seguir, demonstraremos que uma tautologia ocorre 'quase sempre', resultado este também desejável.

Proposição 1.2: $\vdash ☼\top$.

Demonstração:

1. $\vdash ☼\bot \vee ☼\neg\bot$	(Ax_2)
2. $\vdash (☼\bot \vee ☼\neg\bot) \rightarrow (\neg☼\bot \rightarrow ☼\neg\bot)$	(CPC)
3. $\vdash \neg☼\bot \rightarrow ☼\neg\bot$	(MP) em 1 e 2
4. $\vdash \neg☼\bot$	Proposição 1.1
5. $\vdash ☼\neg\bot$	(MP) em 3 e 4
6. $\vdash \neg\bot \rightarrow \top$	(CPC)
7. $\vdash ☼\neg\bot \rightarrow ☼\top$	(R☼)
8. $\vdash ☼\top$	(MP) em 5 e 7. ■

Proposição 1.3: $\vdash \varphi \Rightarrow \vdash ☼\varphi$.

Demonstração:

1. $\vdash \varphi$	premissa
2. $\vdash \top \rightarrow \varphi$	(CPC) em 1
3. $\vdash ☼\top \rightarrow ☼\varphi$	(R☼) em 2
4. $\vdash ☼\top$	Proposição 1.2
5. $\vdash ☼\varphi$	(MP) em 3 e 4. ■

Proposição 1.4: $\vdash ☼(\varphi\wedge\neg\varphi) \rightarrow \bot$.

Demonstração:

1. $\vdash (\varphi\wedge\neg\varphi) \rightarrow \bot$	(CPC)
2. $\vdash ☼(\varphi\wedge\neg\varphi) \rightarrow ☼\bot$	(R☼)
3. $\vdash ☼\bot \rightarrow \bot$	(Ax_3)
4. $\vdash (☼(\varphi\wedge\neg\varphi) \rightarrow ☼\bot) \wedge (☼\bot \rightarrow \bot)$	(CPC) em 2 e 3
5. $\vdash ((☼(\varphi\wedge\neg\varphi) \rightarrow ☼\bot) \wedge (☼\bot \rightarrow \bot)) \rightarrow (☼(\varphi\wedge\neg\varphi) \rightarrow \bot)$	(CPC)
6. $\vdash ☼(\varphi\wedge\neg\varphi) \rightarrow \bot$	(MP) em 4 e 5. ■

Proposição 1.5: $\vdash ☼\neg\varphi \rightarrow \neg☼\varphi$.

Demonstração:

1. $\vdash (☼(\varphi\wedge\neg\varphi) \rightarrow \bot) \rightarrow (\neg\bot \rightarrow \neg☼(\varphi\wedge\neg\varphi))$	(CPC)
2. $\vdash ☼(\varphi\wedge\neg\varphi) \rightarrow \bot$	Proposição 1.4
3. $\vdash \neg\bot \rightarrow \neg☼(\varphi\wedge\neg\varphi)$	(MP) em 1 e 2
4. $\vdash \neg\bot$	(CPC)

5. $\vdash \neg ☼(\varphi\wedge\neg\varphi)$ (MP) em 3 e 4

6. $\vdash (☼\varphi \wedge ☼\neg\varphi) \to ☼(\varphi \wedge \neg\varphi)$ (Ax_1)

7. $\vdash ((☼\varphi\wedge☼\neg\varphi) \to ☼(\varphi\wedge\neg\varphi)) \to (\neg☼(\varphi\wedge\neg\varphi) \to \neg(☼\varphi\wedge☼\neg\varphi))$ (CPC)

8. $\vdash \neg☼(\varphi\wedge\neg\varphi) \to \neg(☼\varphi\wedge☼\neg\varphi)$ (MP) em 6 e 7

9. $\vdash \neg(☼\varphi\wedge☼\neg\varphi)$ (MP) em 5 e 8

10. $\vdash \neg(☼\varphi\wedge☼\neg\varphi) \to (☼\neg\varphi \to \neg☼\varphi)$ (CPC)

11. $\vdash ☼\neg\varphi \to \neg☼\varphi$ (MP) em 9 e 10. ■

Proposição 1.6: $\vdash ☼(\varphi\wedge\psi) \to (☼\varphi \wedge ☼\psi)$.

Demonstração:

1. $\vdash (\varphi\wedge\psi) \to \varphi$ (CPC)

2. $\vdash ☼(\varphi\wedge\psi) \to ☼\varphi$ (R☼) em 1

3. $\vdash (\varphi\wedge\psi) \to \psi$ (CPC)

4. $\vdash ☼(\varphi\wedge\psi) \to ☼\psi$ (R☼) em 3

5. $\vdash ((☼(\varphi\wedge\psi)\to☼\varphi)\wedge(☼(\varphi\wedge\psi)\to☼\psi))\to(☼(\varphi\wedge\psi)\to(☼\varphi\wedge☼\psi))$ (CPC)

6. $\vdash (☼(\varphi\wedge\psi) \to ☼\varphi) \wedge (☼(\varphi\wedge\psi) \to ☼\psi)$ (CPC) em 2 e 4

7. $\vdash ☼(\varphi\wedge\psi)\to(☼\varphi\wedge☼\psi)$ (MP) em 5 e 6. ■

Proposição 1.7: $\vdash ☼\varphi \to ☼(\varphi\vee\psi)$.

Demonstração:

1. $\vdash \varphi \to (\varphi\vee\psi)$ (CPC)

2. $\vdash ☼\varphi \to ☼(\varphi\vee\psi)$ (R☼) em 1. ■

Proposição 1.8: $\vdash (☼\varphi \vee ☼\psi) \to ☼(\varphi\vee\psi)$.

Demonstração:

1. $\vdash ☼\varphi \to ☼(\varphi\vee\psi)$ Proposição 1.7

2. $\vdash \psi \to (\varphi\vee\psi)$ (CPC)

3. $\vdash ☼\psi \to ☼(\varphi\vee\psi)$ (R☼) em 2

4. $\vdash ((☼\varphi\to☼(\varphi\vee\psi))\wedge(☼\psi\to☼(\varphi\vee\psi)))\to((☼\varphi\vee☼\psi)\to☼(\varphi\vee\psi))$ (CPC)

5. $\vdash (☼\varphi \to ☼(\varphi\vee\psi)) \wedge (☼\psi \to ☼(\varphi\vee\psi))$ (CPC) em 2 e 3

6. $\vdash (☼\varphi \vee ☼\psi) \to ☼(\varphi\vee\psi)$ (MP) em 4 e 5. ■

Proposição 1.9: $\vdash \varphi \leftrightarrow \psi \Rightarrow \vdash ☼\varphi \leftrightarrow ☼\psi$.

Demonstração:

1. $\vdash \varphi \leftrightarrow \psi$ premissa

2. $\vdash (\varphi \to \psi) \land (\psi \to \varphi)$	(CPC) em 1
3. $\vdash \varphi \to \psi$	(CPC) em 2
4. $\vdash \psi \to \varphi$	(CPC) em 2
5. $\vdash$ ☼$\varphi \to$ ☼ψ	(R☼) em 3
6. $\vdash$ ☼$\psi \to$ ☼φ	(R☼) em 4
7. $\vdash$ (☼$\varphi \to$ ☼ψ) $\land$ (☼$\psi \to$ ☼φ)	(CPC) em 5 e 6
8. $\vdash$ ☼$\varphi \leftrightarrow$ ☼ψ	(CPC) em 7. ■

Estes são alguns resultados formais do sistema proposto. A seguir, mostramos algumas concepções semânticas envolvidas.

2. A álgebra do ‘quase sempre’

As álgebras do ‘quase sempre’ são modelos algébricos da lógica proposicional do ‘quase sempre’.

Definição 2.1: Uma *álgebra do ‘quase sempre’* é uma sétupla **Q** = (Q, $\land$, $\lor$, ~, 0, 1, ▼), em que (Q, $\land$, $\lor$, ~, 0, 1) é uma álgebra de Boole, $0 \neq 1$ e ▼ é o operador que interpreta a noção de ‘quase sempre’ e satisfaz as seguintes condições, para todos a, $b \in$ Q:

(i) $\blacktriangledown a \land \blacktriangledown b \leq \blacktriangledown(a \land b)$

(ii) $1 \leq \blacktriangledown a \lor \blacktriangledown{\sim}a$

(iii) $\blacktriangledown 0 \leq 0$

(iv) $\blacktriangledown a \leq \blacktriangledown(a \lor b)$.

Proposição 2.1: $\blacktriangledown 0 = 0$.

Demonstração: Como $0 \leq \blacktriangledown 0$ e, pela Definição 2.1, $\blacktriangledown 0 \leq 0$, então, $\blacktriangledown 0 = 0$. ■

Proposição 2.2: $a \leq b \Rightarrow \blacktriangledown a \leq \blacktriangledown b$.

Demonstração: Por hipótese, $a \leq b$. Como $a \lor b = b$, então $\blacktriangledown(a \lor b) = \blacktriangledown b$. Pelo item (iv) da Definição 2.1, $\blacktriangledown a \leq \blacktriangledown(a \lor b) = \blacktriangledown b$. Logo, $\blacktriangledown a \leq \blacktriangledown b$. ■

Proposição 2.3: ${\sim}\blacktriangledown 0 = 1$.

Demonstração: Pela Proposição 2.1, $\blacktriangledown 0 = 0$, então, ${\sim}\blacktriangledown 0 = {\sim}0 = 1$. ■

Proposição 2.4: $\blacktriangledown 1 = 1$.

Demonstração: A condição (ii) da Definição 2.1 nos garante que: $1 \leq \blacktriangledown 1 \vee \blacktriangledown{\sim}1 = \blacktriangledown 1 \vee \blacktriangledown 0$. Logo, pela Proposição 2.1, $1 \leq \blacktriangledown 1 \vee 0 = \blacktriangledown 1$. Como $\blacktriangledown 1 \leq 1$, então $\blacktriangledown 1 = 1$. ■

Proposição 2.5: $\blacktriangledown a \vee \blacktriangledown{\sim}a = 1$.

Demonstração: Como $\blacktriangledown a \vee \blacktriangledown{\sim}a \leq 1$ e, pelo item (ii) da Definição 2.1, $1 \leq \blacktriangledown a \vee \blacktriangledown{\sim}a$, então $\blacktriangledown a \vee \blacktriangledown{\sim}a = 1$. ■

Proposição 2.6: ${\sim}\blacktriangledown{\sim}a \vee {\sim}\blacktriangledown a = 1$.

Demonstração: Pela Definição 2.1 (i), $\blacktriangledown{\sim}a \wedge \blacktriangledown a \leq \blacktriangledown({\sim}a \wedge a) \Rightarrow (\blacktriangledown{\sim}a \wedge \blacktriangledown a) \vee \blacktriangledown({\sim}a \wedge a) = \blacktriangledown({\sim}a \wedge a) \Rightarrow (\blacktriangledown{\sim}a \wedge \blacktriangledown a) \vee \blacktriangledown 0 = \blacktriangledown 0 \Rightarrow$ (Pela Proposição 2.1) $(\blacktriangledown{\sim}a \wedge \blacktriangledown a) \vee 0 = 0 \Rightarrow \blacktriangledown{\sim}a \wedge \blacktriangledown a = 0 \Rightarrow {\sim}(\blacktriangledown{\sim}a \wedge \blacktriangledown a) = {\sim}0 \Rightarrow {\sim}\blacktriangledown{\sim}a \vee {\sim}\blacktriangledown a = 1$. ■

Proposição 2.7: $\blacktriangledown{\sim}a = {\sim}\blacktriangledown a$.

Demonstração: Pela Proposição 2.5, $\blacktriangledown a \vee \blacktriangledown{\sim}a = 1 \Rightarrow {\sim}\blacktriangledown a \wedge (\blacktriangledown a \vee \blacktriangledown{\sim}a) = {\sim}\blacktriangledown a \wedge 1 \Rightarrow ({\sim}\blacktriangledown a \wedge \blacktriangledown a) \vee ({\sim}\blacktriangledown a \wedge \blacktriangledown{\sim}a) = {\sim}\blacktriangledown a \Rightarrow 0 \vee ({\sim}\blacktriangledown a \wedge \blacktriangledown{\sim}a) = {\sim}\blacktriangledown a \Rightarrow {\sim}\blacktriangledown a \wedge \blacktriangledown{\sim}a = {\sim}\blacktriangledown a \Rightarrow {\sim}\blacktriangledown a \leq \blacktriangledown{\sim}a$. (I)

Por outro lado, pela Proposição 2.6, ${\sim}\blacktriangledown{\sim}a \vee {\sim}\blacktriangledown a = 1 \Rightarrow \blacktriangledown{\sim}a \wedge ({\sim}\blacktriangledown{\sim}a \vee {\sim}\blacktriangledown a) = \blacktriangledown{\sim}a \wedge 1 \Rightarrow (\blacktriangledown{\sim}a \wedge {\sim}\blacktriangledown{\sim}a) \vee (\blacktriangledown{\sim}a \wedge {\sim}\blacktriangledown a) = \blacktriangledown{\sim}a \Rightarrow 0 \vee (\blacktriangledown{\sim}a \wedge {\sim}\blacktriangledown a) = \blacktriangledown{\sim}a \Rightarrow \blacktriangledown{\sim}a \wedge {\sim}\blacktriangledown a = \blacktriangledown{\sim}a \Rightarrow \blacktriangledown{\sim}a \leq {\sim}\blacktriangledown a$. (II)

Portanto, de (I) e (II), $\blacktriangledown{\sim}a = {\sim}\blacktriangledown a$. ■

Proposição 2.8: $\blacktriangledown(a \wedge b) \leq \blacktriangledown a$.

Demonstração: Como $a \wedge b \leq a$, então, pela Proposição 2.2, $\blacktriangledown(a \wedge b) \leq \blacktriangledown a$. ■

Proposição 2.9: $\blacktriangledown(a \wedge b) = \blacktriangledown a \wedge \blacktriangledown b$.

Demonstração: Pela proposição anterior, $\blacktriangledown(a \wedge b) \leq \blacktriangledown a$ e $\blacktriangledown(a \wedge b) \leq \blacktriangledown b$. Portanto, $\blacktriangledown(a \wedge b) \leq \blacktriangledown a \wedge \blacktriangledown b$. Por outro lado, pela condição (i) da Definição 2.1, $\blacktriangledown a \wedge \blacktriangledown b \leq \blacktriangledown(a \wedge b)$. Assim, $\blacktriangledown(a \wedge b) = \blacktriangledown a \wedge \blacktriangledown b$. ■

Proposição 2.10: $\blacktriangledown(a \vee b) = \blacktriangledown a \vee \blacktriangledown b$.

Demonstração: Pela Proposição 2.7 e pela Proposição 2.9, ${\sim}(\blacktriangledown(a \vee b)) = \blacktriangledown{\sim}(a \vee b) = \blacktriangledown({\sim}a \wedge {\sim}b) = \blacktriangledown{\sim}a \wedge \blacktriangledown{\sim}b = {\sim}\blacktriangledown a \wedge {\sim}\blacktriangledown b = {\sim}(\blacktriangledown a \vee \blacktriangledown b)$. Logo, $\blacktriangledown(a \vee b) = \blacktriangledown a \vee \blacktriangledown b$. ■

Exemplo: Seja $A = \{x, y, z\}$ e $(\mathcal{P}(A), \cap, \cup, ^{C}, \varnothing, A)$ uma álgebra de Boole de conjuntos. Definimos um operador do 'quase sempre', $\blacktriangledown$, sobre $(\mathcal{P}(A), \cap, \cup, ^{C}, \varnothing, A)$ da seguinte maneira: $\blacktriangledown\varnothing = \varnothing$; $\blacktriangledown\{x\} = \{z\}$; $\blacktriangledown\{y, z\} = \{x, y\}$; $\blacktriangledown\{y\} = \{x, y\}$, $\blacktriangledown\{x, z\} = \{z\}$; $\blacktriangledown\{z\}$

$= \varnothing$; $\blacktriangledown\{x, y\} = \{x, y, z\}$; $\blacktriangledown\{x, y, z\} = \{x, y, z\}$. Facilmente podemos verificar que ($\mathcal{P}$ (A), $\cap$, $\cup$, ^{C}, $\varnothing$, A, $\blacktriangledown$) é uma álgebra do 'quase sempre'. Tomando $a = \{y, z\}$ e $b = \{y\}$ temos que, $\blacktriangledown\{y, z\} = \{x, y\} \subseteq \{x, y\} = \blacktriangledown\{y\}$, mas $\{y, z\} \not\subseteq \{y\}$. Tomando, agora, $a = \{x\}$ e $b = \{y\}$ temos que, $\blacktriangledown\{x\} = \{z\} \not\subseteq \varnothing = \blacktriangledown\varnothing = \blacktriangledown(\{x\}\cap\{y\})$, ademais, $\blacktriangledown(\{x\}\cup\{y\}) = \blacktriangledown\{x, y\} = \{x, y, z\} \not\subseteq \{z\} = \blacktriangledown\{x\}$.

Nesse exemplo verificamos que as condições (i), (ii) e (iii) abaixo nem sempre valem numa álgebras do 'quase sempre':

(i) $\blacktriangledown a \leq \blacktriangledown b \Rightarrow a \leq b$;

(ii) $\blacktriangledown a \leq \blacktriangledown(a \wedge b)$;

(iii) $\blacktriangledown(a \vee b) \leq \blacktriangledown a$.

Portanto, não podemos trocar as desigualdades por igualdades na Definição 2.1 (iv) e na Proposição 2.8 e a implicação pela equivalência na Proposição 2.2.

Definição 2.2: Sejam **Q** = (Q, $\wedge$, $\vee$, ~, 0, 1, $\blacktriangledown$) e **Q'** = (Q', $\wedge$', $\vee$', ~', 0', 1', $\blacktriangledown$) álgebras do 'quase sempre' e h: **Q** → **Q'** uma função. Dizemos que h é um *homomorfismo de álgebras do 'quase sempre'* se para todo a, $b \in$ Q temos $h(a \wedge b) = h(a) \wedge' h(b)$, $h(a \vee b) = h(a) \vee' h(b)$, $h({\sim}a) = {\sim}'h(a)$ e $h(\blacktriangledown a) = \blacktriangledown h(a)$.

Definição 2.3: Um *isomorfismo de álgebras do 'quase sempre'* é um homomorfismo bijetivo de álgebras do 'quase sempre'.

Definição 2.4: Um *monomorfismo de álgebras do 'quase sempre'* é um homomorfismo injetivo de álgebras do 'quase sempre'.

Teorema 2.1: Para toda álgebra do 'quase sempre' **Q** = (Q, $\wedge$, $\vee$, ~, 0, 1, $\blacktriangledown$) existe um isomorfismo h de **Q** em uma álgebra do 'quase sempre' de conjuntos.

Demonstração: Pelo Isomorfismo de Stone, para toda álgebra de Boole (Q, $\wedge$, $\vee$, ~, 0, 1) existe um monomorfismo de (Q, $\wedge$, $\vee$, ~, 0, 1) em $\mathcal{P}$ ($\mathcal{P}$ (Q)). Seja h um isomorfismo (homomorfismo bijetivo) de (Q, $\wedge$, $\vee$, ~, 0, 1) em (Q', $\cap$, $\cup$, ^{C}, $\varnothing$), de forma que Q' = h(Q) $\subseteq \mathcal{P}$ ($\mathcal{P}$ (Q)).

O isomorfismo h se estende a um isomorfismo entre **Q** = (Q, $\wedge$, $\vee$, ~, 0, 1, $\blacktriangledown$) e **Q'** = (Q', $\cap$, $\cup$, ^{C}, $\varnothing$, $\blacktriangledown$) quando definimos, para cada $a \in$ Q, $\blacktriangledown h(a) = h(\blacktriangledown a)$. A função h é bijetiva e, desse modo, preserva também a nova operação.

A álgebra de conjuntos **Q'** é uma álgebra do 'quase sempre' de conjuntos, pois:

(i) Se a, $b \in$ Q, pela Proposição 2.9, $\blacktriangledown a \wedge \blacktriangledown b = \blacktriangledown(a \wedge b)$, então $h(\blacktriangledown a \wedge \blacktriangledown b) = h(\blacktriangledown(a \wedge b)) \Rightarrow h(\blacktriangledown a) \cap h(\blacktriangledown b) = {\blacktriangledown}h(a \wedge b) \Rightarrow {\blacktriangledown}h(a) \cap {\blacktriangledown}h(b) = {\blacktriangledown}(h(a) \cap h(b))$. Logo, ${\blacktriangledown}h(a) \cap {\blacktriangledown}h(b) \subseteq {\blacktriangledown}(h(a) \cap h(b))$.

(ii) Se $a \in$ Q, pela Proposição 2.5, $\blacktriangledown a \vee \blacktriangledown{\sim}a = 1$, então $h(\blacktriangledown a \vee \blacktriangledown{\sim}a) = h(1) \Rightarrow h(\blacktriangledown a) \cup h(\blacktriangledown{\sim}a) = Q' \Rightarrow {\blacktriangledown}h(a) \cup {\blacktriangledown}h({\sim}a) = Q' \Rightarrow {\blacktriangledown}h(a) \cup {\blacktriangledown}h(a)^C = Q'$. Portanto, $Q' \subseteq {\blacktriangledown}h(a) \cup {\blacktriangledown}h(a)^C$.

(iii) Pela Proposição 2.1, $\blacktriangledown 0 = 0$, logo, $h(\blacktriangledown 0) = h(0) \Rightarrow {\blacktriangledown}h(0) = h(0) \Rightarrow {\blacktriangledown}\varnothing = \varnothing \Rightarrow {\blacktriangledown}\varnothing \subseteq \varnothing$.

(iv) Se a, $b \in$ Q, pela condição (iv) da Definição 2.1, $\blacktriangledown a \leq \blacktriangledown(a \vee b)$, então $\blacktriangledown a \wedge \blacktriangledown(a \vee b) = \blacktriangledown a \Rightarrow h(\blacktriangledown a) \cap h(\blacktriangledown(a \vee b)) = h(\blacktriangledown a \wedge \blacktriangledown(a \vee b)) = h(\blacktriangledown a) \Rightarrow h(\blacktriangledown a) \subseteq h(\blacktriangledown(a \vee b))$. Portanto, ${\blacktriangledown}h(a) = h(\blacktriangledown a) \subseteq h(\blacktriangledown(a \vee b)) = {\blacktriangledown}h(a \vee b) = {\blacktriangledown}\,(h(a) \cup h(b))$. ■

Apresentamos, na sequência, a correção e completude de **For**$\mathcal{L}$(☼) relativo às álgebras do quase sempre.

3. Adequação da lógica proposicional do ‘quase sempre’

Em geral, os sistemas formais possuem um modelo ou uma semântica adequada a ele. O sistema é adequado quando ele é correto e completo. A correção fraca determina que todo teorema é uma fórmula válida; e a completude fraca que toda fórmula válida é um teorema. Enquanto a correção forte e a completude forte envolvem não só teoremas e fórmulas válidas, mas também consequências semântica e sintática (formal).

Nessa seção, faremos a demonstração da adequação forte entre a lógica proposicional do ‘quase sempre’ e as álgebras do ‘quase sempre’.

Denotaremos uma álgebra do ‘quase sempre’ genérica por A.

Definição 3.1: Uma fórmula $\varphi \in$ **For**$\mathcal{L}$(☼) é *refutável* em Γ quando $\Gamma \vdash \neg\varphi$, caso contrário, φ é *irrefutável*.

Definição 3.2: Uma *valoração restrita* é uma função $\upsilon^\wedge$: **Var**$\mathcal{L}$ (☼) $\rightarrow$ A, que interpreta cada variável de $\mathcal{L}$ (☼) em um elemento de A.

Definição 3.3: Uma *valoração* é uma função υ : **For**$\mathcal{L}$ (☼) $\rightarrow$ A. Considerando p uma fórmula atômica e φ e ψ fórmulas quaisquer, a valoração estende natural e unicamente a valoração restrita do seguinte modo:

$\upsilon(p) = \upsilon^\wedge(p)$

$\upsilon(\neg\varphi) = {\sim}\,\upsilon(\varphi)$

$\upsilon(\varphi \vee \psi) = \upsilon(\varphi) \vee \upsilon(\psi)$

$\upsilon(\varphi \wedge \psi) = \upsilon(\varphi) \wedge \upsilon(\psi)$

$\upsilon(☼\varphi) = \blacktriangledown\upsilon(\varphi)$.

Observações: Os símbolos de operadores do lado esquerdo das igualdades representam os operadores lógicos, enquanto os símbolos de operadores do lado direito das igualdades representam os operadores algébricos. O símbolo $\rightarrow$ é definido da seguinte maneira: $\varphi \rightarrow \psi =_{df} \neg\varphi\vee\psi$.

Definição 3.4: Uma valoração υ : **For**$\mathcal{L}$ (☼) $\rightarrow$ A é um *modelo* para um conjunto $\Gamma \subseteq$ **For**$\mathcal{L}$ (☼) quando $\upsilon(\varphi) = 1$, para toda fórmula $\varphi \in \Gamma$.

Em particular, uma valoração υ : **For**$\mathcal{L}$ (☼) $\rightarrow$ A é um modelo para $\varphi \in$ **For**$\mathcal{L}$(☼) quando $\upsilon(\varphi) = 1$.

Definição 3.5: Uma fórmula φ é *válida* em uma álgebra do 'quase sempre' A quando toda valoração υ : **For**$\mathcal{L}$ (☼) $\rightarrow$ A é um modelo para φ.

Definição 3.6: Uma fórmula φ é *qs-válida* quando ela é válida em toda álgebra do 'quase sempre'.

Denotamos que uma fórmula φ é qs-válida por $\vDash\varphi$.

Definição 3.7: Seja $\Gamma \subseteq$ **For**$\mathcal{L}$(☼), **Ax** o conjunto de axiomas de $\mathcal{L}$(☼) e $C(\Gamma) = \{\psi : \Gamma\cup$**Ax** $\vdash \psi\}$. Dizemos que ψ é *derivável* em $\mathcal{L}$(☼) ou é um *teorema* de $\mathcal{L}$(☼) quando $\psi \in C(\emptyset)$, ou de outro modo, quando $\Gamma = \emptyset$.

Se $\Gamma = \emptyset$, temos os teoremas de $\mathcal{L}$(☼), e assim $\psi \in C(\emptyset) \Leftrightarrow \vdash \psi$.

Definição 3.8: Uma *teoria* de $\mathcal{L}$(☼) é um conjunto $\Delta \subseteq$ **For**$\mathcal{L}$(☼), tal que $C(\Delta) = \Delta$.

Definição 3.9: Seja $\Gamma\cup\{\varphi\} \subseteq$ **For**$\mathcal{L}$(☼). O *conjunto de fórmulas* Γ é *inconsistente* quando conseguimos deduzir $\Gamma \vdash \varphi$ e $\Gamma \vdash \neg\varphi$, para alguma fórmula φ. Caso contrário, Γ é *consistente*.

Definição 3.10: Um sistema, constituído por uma linguagem formal e regras de dedução, é *consistente* quando o seu conjunto de teoremas é consistente. Caso contrário, ele é *inconsistente*.

Definição 3.11: A *álgebra de fórmulas* de $\mathcal{L}(☼)$ é dada por $(\mathbf{For}\mathcal{L}(☼), \wedge, \vee, \rightarrow, \neg, ☼)$, em que $\wedge, \vee, \rightarrow, \neg$ e ☼ são os operadores de $\mathcal{L}(☼)$.

Uma álgebra de Lindenbaum é um conjunto de classes de equivalência obtidas a partir de uma relação de equivalência, ou ainda uma congruência, definida sobre o conjunto de fórmulas de uma determinada lógica. Definiremos a seguir a relação de equivalência que nos dará a álgebra de Lindenbaum de $\mathcal{L}(☼)$.

Definição 3.12: Dado $\Gamma \subseteq \mathbf{For}\mathcal{L}(☼)$, a relação $\equiv_\Gamma$ é definida por:

$$\varphi \equiv_\Gamma \psi \Leftrightarrow \Gamma \vdash \varphi \rightarrow \psi \text{ e } \Gamma \vdash \psi \rightarrow \varphi.$$

A partir daqui omitiremos o índice Γ da relação.

Proposição 3.1: A relação $\equiv$ é uma relação de congruência.

Demonstração: Primeiro, demonstraremos que $\equiv$ é uma relação de equivalência. A relação é reflexiva: para toda fórmula $\varphi \in \Gamma \subseteq \mathbf{For}\mathcal{L}(☼)$, $\Gamma \vdash \varphi \rightarrow \varphi$ e, então, $\varphi \equiv \varphi$. A relação é simétrica: se $\varphi \equiv \psi$, então $\Gamma \vdash \varphi \rightarrow \psi$ e $\Gamma \vdash \psi \rightarrow \varphi$. Logo, $\psi \equiv \varphi$. A relação é transitiva: se $\varphi \equiv \psi$ e $\psi \equiv \sigma$, então $\Gamma \vdash \varphi \rightarrow \psi$, $\Gamma \vdash \psi \rightarrow \varphi$, $\Gamma \vdash \psi \rightarrow \sigma$ e $\Gamma \vdash \sigma \rightarrow \psi$. Logo, $\Gamma \vdash \varphi \rightarrow \sigma$ e $\Gamma \vdash \sigma \rightarrow \varphi \Leftrightarrow \varphi \equiv \sigma$. Assim, a relação $\equiv$ é uma relação de equivalência.

Para concluir a demonstração de que a relação $\equiv$ é uma congruência, basta mostrar que ela preserva o operador ☼, pois, claramente, preserva os operadores booleanos. Pela Proposição 1.9, temos: $\varphi \equiv \psi \Leftrightarrow \Gamma \vdash \varphi \leftrightarrow \psi \Rightarrow \Gamma \vdash ☼\varphi \leftrightarrow ☼\psi \Leftrightarrow ☼\varphi \equiv ☼\psi$. Assim, a relação $\equiv$ é uma congruência. ■

Definição 3.13: A *classe de equivalência de* φ *módulo* $\equiv$ *e* Γ é dada por: $[\varphi]_\Gamma = \{\psi \in \mathbf{For}\mathcal{L}(☼) : \psi \equiv \varphi\}$.

Definição 3.14: A *álgebra de Lindenbaum de* $\mathcal{L}(☼)$, denotada por $A_\Gamma(\mathcal{L}(☼))$, é a álgebra quociente dada por:

$$A_\Gamma(\mathcal{L}(☼)) = (\mathbf{For}\mathcal{L}(☼)\big|_\equiv, \wedge_\equiv, \vee_\equiv, \neg_\equiv, ☼_\equiv, 0_\equiv, 1_\equiv), \text{ tal que:}$$

$[\varphi] \wedge_\equiv [\psi] = [\varphi \wedge \psi]$;

$[\varphi] \vee_\equiv [\psi] = [\varphi \vee \psi]$;

$\neg_\equiv [\varphi] = [\neg\varphi]$;

$☼_{\equiv} [\varphi] = [☼\varphi]$;

$0_{\equiv} = [\varphi\wedge\neg\varphi] = [\bot]$ e

$1_{\equiv} = [\varphi\vee\neg\varphi] = [\top]$.

Observação: Não escreveremos o índice ≡ daqui para frente.

Quando $\Gamma = \varnothing$ denotamos a álgebra de Lindenbaum de $\mathcal{L}(☼)$ relativa a Γ por $A(\mathcal{L}(☼))$, a qual chamaremos simplesmente de álgebra de Lindenbaum de $\mathcal{L}(☼)$.

Proposição 3.2: Em $A_{\Gamma}(\mathcal{L}(☼))$ temos $[\varphi] \leq [\psi] \Leftrightarrow \Gamma \vdash \varphi \to \psi$.

Demonstração:

$[\varphi] \leq [\psi] \Leftrightarrow [\varphi] \wedge [\psi] = [\varphi] \Leftrightarrow [\varphi\wedge\psi] = [\varphi] \Leftrightarrow \Gamma \vdash \varphi\wedge\psi \leftrightarrow \varphi \Leftrightarrow \Gamma \vdash \varphi \to \psi$. ■

Proposição 3.3: A álgebra $A_{\Gamma}(\mathcal{L}(☼))$ é uma álgebra do ‘quase sempre’.

Demonstração: Utilizaremos, nesta demonstração, a Proposição 3.2 e a Definição 3.12.

(i) Pelo (Ax_1), $\vdash (☼\varphi \wedge ☼\psi) \to ☼(\varphi \wedge \psi) \Rightarrow [☼\varphi \wedge ☼\psi] \leq [☼(\varphi \wedge \psi)] \Rightarrow [☼\varphi] \wedge [☼\psi] \leq [☼(\varphi \wedge \psi)] \Rightarrow ☼[\varphi] \wedge ☼[\psi] \leq ☼[\varphi \wedge \psi]$.

(ii) Pelo (Ax_2), $\vdash ☼\varphi \vee ☼\neg\varphi \Rightarrow \vdash (\varphi\vee\neg\varphi) \to (☼\varphi \vee ☼\neg\varphi) \Rightarrow [\varphi\vee\neg\varphi] \leq [☼\varphi \vee ☼\neg\varphi] \Rightarrow 1 \leq [☼\varphi] \vee [☼\neg\varphi] \Rightarrow 1 \leq ☼[\varphi] \vee ☼[\neg\varphi]$.

(iii) Pelo (Ax_3), $\vdash ☼\bot \to \bot \Rightarrow [☼\bot] \leq [\bot] \Rightarrow ☼[\bot] \leq 0 \Rightarrow ☼0 \leq 0$.

(iv) A Proposição 1.7 nos garante que: $\vdash ☼\varphi \to ☼(\varphi\vee\psi) \Rightarrow [☼\varphi] \leq [☼(\varphi\vee\psi)] \Rightarrow ☼[\varphi] \leq ☼[\varphi\vee\psi]$.

Assim, a álgebra $A_{\Gamma}(\mathcal{L}(☼))$ é uma álgebra do ‘quase sempre’. ■

Definição 3.15: A valoração υ : **For**$\mathcal{L}(☼) \to A_{\Gamma}(\mathcal{L}(☼))$ é o *modelo canônico* de $\Gamma \subseteq$ **For** $\mathcal{L}(☼)$.

Proposição 3.4: Seja $\Gamma \cup \{\varphi\} \subseteq$ **For**$\mathcal{L}(☼)$:

(i) $\Gamma \vdash \varphi$ se, e somente se, $[\varphi] = 1$ em $A_{\Gamma}(\mathcal{L}(☼))$;

(ii) $\Gamma \vdash \neg\varphi$ (φ é refutável em Γ) se, e somente se, $[\varphi] = 0$ em $A_{\Gamma}(\mathcal{L}(☼))$.

Demonstração:

(i) ($\Leftarrow$) Se $[\varphi] = 1$, então $[\varphi \to \varphi] \leq [\varphi]$, pela Proposição 3.2, $\Gamma \vdash (\varphi \to \varphi) \to \varphi$. Como $\Gamma \vdash \varphi \to \varphi$, então, pela regra MP, temos $\Gamma \vdash \varphi$.

($\Rightarrow$) Se $\Gamma \vdash \varphi$, então, como $\varphi \to ((\varphi \to \varphi) \to \varphi)$ é um resultado do CPC, pela regra MP, $\Gamma \vdash (\varphi \to \varphi) \to \varphi$. A álgebra $\mathrm{A}_\Gamma(\mathcal{L}(☼))$ sempre tem o elemento 1. Logo, pela Definição 3.12 e pela Proposição 3.2, $1 = [\varphi \vee \neg\varphi] = [\varphi \to \varphi] \leq [\varphi]$ e, portanto, $[\varphi] = 1$.

(ii) Pelo item anterior e pela Definição 3.12, temos: $\Gamma \vdash \neg\varphi \Leftrightarrow [\neg\varphi] = 1 \Leftrightarrow \neg[\varphi] = 1 \Leftrightarrow [\varphi] = 0$. ■

Teorema 3.1: (Correção) As álgebras do 'quase sempre' são modelos corretos para a lógica $\mathcal{L}(☼)$.

Demonstração: Seja $\mathrm{A} = (A, \wedge, \vee, \sim, 0, 1, \blacktriangledown)$ uma álgebra do 'quase sempre'. Resta demonstrar que os axiomas (Ax_1), (Ax_2) e (Ax_3) são válidos e a regra (R☼) preserva a validade. Utilizaremos a Definição 3.3.

(Ax_1) $\upsilon((☼\varphi \wedge ☼\psi) \to ☼(\varphi \wedge \psi)) = \upsilon(\neg(☼\varphi \wedge ☼\psi)) \vee (☼(\varphi\wedge\psi)) = \upsilon(\neg(☼\varphi \wedge ☼\psi)) \vee \upsilon(☼(\varphi\wedge\psi)) = \upsilon(\neg☼\varphi \vee \neg☼\psi) \vee \upsilon(☼(\varphi\wedge\psi)) = (\upsilon(\neg☼\varphi) \vee \upsilon(\neg☼\psi)) \vee \upsilon(☼(\varphi\wedge\psi)) = ({\sim}\blacktriangledown\upsilon(\varphi) \vee {\sim}\blacktriangledown\upsilon(\psi)) \vee \blacktriangledown\upsilon(\varphi\wedge\psi) = ({\sim}\blacktriangledown\upsilon(\varphi) \vee {\sim}\blacktriangledown\upsilon(\psi)) \vee \blacktriangledown(\upsilon(\varphi) \wedge \upsilon(\psi))$ = (Pela Proposição 2.9) $({\sim}\blacktriangledown\upsilon(\varphi) \vee {\sim}\blacktriangledown\upsilon(\psi)) \vee (\blacktriangledown\upsilon(\varphi) \wedge \blacktriangledown\upsilon(\psi)) = (({\sim}\blacktriangledown\upsilon(\varphi) \vee {\sim}\blacktriangledown\upsilon(\psi)) \vee \blacktriangledown\upsilon(\varphi)) \wedge (({\sim}\blacktriangledown\upsilon(\varphi) \vee {\sim}\blacktriangledown\upsilon(\psi)) \vee \blacktriangledown\upsilon(\psi)) = (({\sim}\blacktriangledown\upsilon(\varphi) \vee \blacktriangledown\upsilon(\varphi)) \vee {\sim}\blacktriangledown\upsilon(\psi)) \wedge ({\sim}\blacktriangledown\upsilon(\varphi) \vee ({\sim}\blacktriangledown\upsilon(\psi) \vee \blacktriangledown\upsilon(\psi))) = (1 \vee {\sim}\blacktriangledown\upsilon(\psi)) \wedge ({\sim}\blacktriangledown\upsilon(\varphi) \vee 1) = 1 \wedge 1 = 1$.

(Ax_2) $\upsilon(☼\varphi \vee ☼\neg\varphi) = \upsilon(☼\varphi) \vee \upsilon(☼\neg\varphi) = \blacktriangledown\upsilon(\varphi) \vee \blacktriangledown\upsilon(\neg\varphi) = \blacktriangledown\upsilon(\varphi) \vee \blacktriangledown{\sim}\upsilon(\varphi) = 1$ (Proposição 2.5).

(Ax_3) $\upsilon(☼\bot \to \bot) = \upsilon(\neg(☼\bot) \vee \bot) = \upsilon(\neg(☼\bot)) \vee \upsilon(\bot) = {\sim}\upsilon(☼\bot) \vee 0 = {\sim}\blacktriangledown\upsilon(\bot) \vee 0 = {\sim}\blacktriangledown 0 = {\sim}0 = 1$ (Proposição 2.1).

(R☼) $\upsilon(\varphi \to \psi) = 1 \Rightarrow \upsilon(\varphi) \leq \upsilon(\psi) \Rightarrow$ (Pela Proposição 2.2) $\blacktriangledown\upsilon(\varphi) \leq \blacktriangledown\upsilon(\psi) \Rightarrow \upsilon(☼\varphi) \leq \upsilon(☼\psi) \Rightarrow \upsilon(☼\varphi \to ☼\psi) = 1$. ■

Proposição 3.5: A lógica proposicional $\mathcal{L}(☼)$ é consistente.

Demonstração: Suponhamos que $\mathcal{L}$ (☼) não seja consistente. Então, pela Definição 3.10, existe $\varphi \in$ **For** $\mathcal{L}$ (☼) tal que $\vdash\varphi$ e $\vdash\neg\varphi$. Pelo Teorema da Correção, φ e $\neg\varphi$ são fórmulas válidas. Seja υ uma valoração em uma álgebra do 'quase sempre' com dois elementos **2** = {0, 1}. Como φ e $\neg\varphi$ são válidas, então $\upsilon(\neg\varphi) = 1$ e $\upsilon(\varphi) = 1 \Rightarrow \upsilon(\neg\varphi) = {\sim}\upsilon(\varphi) = 0$, o que é uma contradição. Logo, $\mathcal{L}$ (☼) é consistente. ■

Lema 3.1: As seguintes condições são equivalentes para toda fórmula $\varphi \in$ **For** $\mathcal{L}(☼)$:

(i) $\vdash\varphi$;

(ii) $\vDash\varphi$;

(iii) φ é válida em toda álgebra do 'quase sempre' de conjuntos;

(iv) $\upsilon_0(\varphi) = 1$, em que υ_0 é a valoração do modelo canônico $\mathrm{A}(\mathcal{L}(☼))$.

Demonstração:

(i) $\Rightarrow$ (ii): Segue do Teorema da Correção.

(ii) $\Rightarrow$ (iii): Se a fórmula φ é *qs-válida*, $\vDash\varphi$, então ela é válida em toda álgebra do 'quase sempre', em particular, φ é válida em toda álgebra do 'quase sempre' de conjuntos.

(iii) $\Rightarrow$ (iv): Pela Proposição 3.3, $\mathrm{A}(\mathcal{L}(☼))$ é uma álgebra do 'quase sempre'. Logo, pelo Teorema 2.1, $\mathrm{A}(\mathcal{L}(☼))$ é isomorfa a uma álgebra do 'quase sempre' de conjuntos **Q'** = (Q', $\cap$, $\cup$, ^{C}, $\varnothing$, ▾). Assim, se φ é válida em **Q'**, então φ é válida em $\mathrm{A}(\mathcal{L}(☼))$, ou seja, $\upsilon_0(\varphi) = 1$.

(iv) $\Rightarrow$ (i): Se $\varphi \in$ **For**$\mathcal{L}(☼)$ e $\nvdash\varphi$ em $\mathcal{L}(☼)$, então, pela Proposição 3.4, $[\varphi]$ não coincide com a unidade de $\mathrm{A}(\mathcal{L}(☼))$ e, assim, $\upsilon_0(\varphi) \neq 1$. ■

Teorema 3.2: (Completude) Para toda fórmula $\varphi \in$ **For**$\mathcal{L}(☼)$, se φ é uma fórmula válida, então φ é derivável em $\mathcal{L}(☼)$.

Demonstração: Segue pelo Lema 3.1. ■

Foram demonstrados os Teoremas da Correção e da Completude fraca. A seguir demonstraremos a Adequação (Correção e Completude) forte.

Denotaremos por $\Gamma \vDash \varphi$ todo modelo para Γ que também é modelo para φ.

Lema 3.2: Seja $\Gamma \subseteq$ **For**$\mathcal{L}(☼)$. Se $\Gamma \vdash \varphi$, então $\Gamma \vDash \varphi$.

Demonstração: Seja υ : **For**$\mathcal{L}(☼) \rightarrow \mathrm{B}$ um modelo para Γ. Como $\Gamma \vdash \varphi$, φ pode ser um axioma de $\mathcal{L}(☼)$, ou uma fórmula obtida por meio de regras de dedução de $\mathcal{L}(☼)$, ou uma fórmula de Γ. Pelo Teorema da Correção, os axiomas de $\mathcal{L}(☼)$ são válidos e as regras de $\mathcal{L}(☼)$ preservam a validade. Além disso, como $\upsilon_B(\psi) = 1$, para toda fórmula $\psi \in \Gamma$, então $\upsilon_B(\varphi) = 1$. Logo, υ_B é um modelo para φ. ■

Proposição 3.6: Seja $\Gamma \subseteq$ **For**$\mathcal{L}(☼)$ e B uma álgebra do 'quase sempre'. Se existe um modelo υ : **For**$\mathcal{L}(☼) \rightarrow \mathrm{B}$ para Γ, então Γ é consistente.

Demonstração: Suponhamos que Γ não é consistente. Então, existe φ tal que: $\Gamma \vdash \varphi$ e $\Gamma \vdash \neg\varphi$, além disso, $\upsilon_B(\varphi) = 1$ e $\upsilon_B(\neg\varphi) = 1 \Rightarrow {\sim}\upsilon_B(\varphi) = 1 \Rightarrow \upsilon_B(\varphi) = 0$, temos uma contradição. Portanto, Γ é consistente. ■

Definição 3.16: Um modelo υ : **For**$\mathcal{L}(☼) \rightarrow \mathrm{B}$ é *fortemente adequado* para Γ quando: $\Gamma \vdash \varphi$ se, e somente se, $\Gamma \vDash_B \varphi$.

Lema 3.3: Se $\Gamma \subseteq$ **For**$\mathcal{L}(☼)$ é consistente, então a valoração canônica é um modelo fortemente adequado para Γ.

Demonstração: Considerando a valoração canônica υ_0 : **For**$\mathcal{L}(☼) \to \mathrm{A}_\Gamma(\mathcal{L}(☼))$, $\upsilon_0(\varphi) = [\varphi]$. Pela Proposição 3.4 (i), $\upsilon_0(\varphi) = 1$ se, e somente se, $\Gamma \vdash \varphi$. Consequentemente, a valoração canônica υ_0 é um modelo adequado para Γ. ■

Lema 3.4: As seguintes condições são equivalentes para todo conjunto de fórmulas $\Gamma \subseteq$ **For**$\mathcal{L}(☼)$:

(i) Γ é consistente;

(ii) existe um modelo fortemente adequado para Γ;

(iii) existe um modelo fortemente adequado para Γ em uma álgebra do 'quase sempre' B que é uma álgebra de conjuntos **Q'** = (Q', $\cap$, $\cup$, ^{C}, $\varnothing$, $\blacktriangledown$);

(iv) existe um modelo para Γ.

Demonstração:

(i) $\Rightarrow$ (ii): Segue pela Lema 3.3.

(ii) $\Rightarrow$ (iii): Como, pela Proposição 3.3, $\mathrm{A}_\Gamma(\mathcal{L}(☼))$ é uma álgebra do 'quase sempre' e, pelo Teorema 2.1, toda álgebra do 'quase sempre' é isomorfa a uma álgebra de conjuntos **Q'** = (Q', $\cap$, $\cup$, ^{C}, $\varnothing$, $\blacktriangledown$), então o resultado é imediato.

(iii) $\Rightarrow$ (iv) O resultado é imediato.

(iv) $\Rightarrow$ (i) Segue pela Proposição 3.6. ■

Teorema 3.3: (Adequação forte) Seja $\Gamma \subseteq$ **For**$\mathcal{L}(☼)$. Se Γ é consistente, as afirmações seguintes são equivalentes:

(i) $\Gamma \vdash \varphi$;

(ii) $\Gamma \vDash \varphi$;

(iii) todo modelo de Γ na álgebra do 'quase sempre' de conjuntos **Q'** = (Q', $\cap$, $\cup$, ^{C}, $\varnothing$, $\blacktriangledown$) é um modelo para φ.

(iv) $\upsilon_0(\varphi) = 1$ para toda valoração canônica υ_0 no modelo canônico $\mathrm{A}_\Gamma(\mathcal{L}(☼))$.

Demonstração:

(i) $\Rightarrow$ (ii): Segue do Lema 3.2.

(ii) $\Rightarrow$ (iii): Se $\Gamma \vDash \varphi$, então todo modelo para Γ também é modelo para φ, em particular, todo modelo de Γ na álgebra do 'quase sempre' de conjuntos **Q'** = (Q', $\cap$, $\cup$, ^{C}, $\varnothing$, $\blacktriangledown$) é um modelo para φ.

(iii) $\Rightarrow$ (iv): Por hipótese, Γ é consistente. Logo, pelo Lema 3.4, existe um modelo fortemente adequado para Γ em uma álgebra do 'quase sempre' B que é uma álgebra de conjuntos **Q'** = (Q', $\cap$, $\cup$, ^{C}, $\varnothing$, $\blacktriangledown$). Como $\mathrm{A}_\Gamma(\mathcal{L}(☼))$ é uma álgebra do 'quase sempre' (Proposição 3.3) e toda álgebra do 'quase sempre' é isomorfa a uma álgebra de conjuntos **Q'** = (Q', $\cap$, $\cup$, ^{C}, $\varnothing$, $\blacktriangledown$) (Teorema 2.1), então, para uma valoração canônica υ_0 no modelo canônico $\mathrm{A}_\Gamma(\mathcal{L}(☼))$, $\upsilon_0(\varphi) = 1$.

(iv) $\Rightarrow$ (i): Como, por hipótese, Γ é consistente, segue pelo Lema 3.3 que a valoração canônica υ_0 : **For**$\mathcal{L}(☼) \rightarrow \mathrm{A}_{\Gamma}(\mathcal{L}(☼))$ é um modelo fortemente adequado para Γ, ou seja, $\Gamma \vdash \varphi$ se, e somente se, $\Gamma \vDash_{\mathrm{A}\,\Gamma(\mathcal{L}(☼))} \varphi$. Pelo item (iv) deste teorema, $\Gamma \vDash_{\mathrm{A}\,\Gamma(\mathcal{L}(☼))} \varphi$. Logo, $\Gamma \vdash \varphi$. ■

Considerações Finais

Baseados no quantificador introduzido na Lógica do Ultrafiltro, construímos uma lógica proposicional, de caráter modal, associada ao conceito de 'quase sempre'. Acreditamos que, da mesma forma que fizemos neste trabalho, podemos tratar outras lógicas que introduzem novos quantificadores na linguagem quantificacional clássica num ambiente proposicional.

Agradecimentos

Este trabalho teve o apoio da CAPES.

* * *

Referências

FEITOSA, H. A.; NASCIMENTO, M. C.; GRÁCIO, M. C. C. *Algebraic elements for the notions of 'many'*. CLE e-Prints (Online), v. 9, n. 1, 2009a. Disponível em: <*http://www.cle.unicamp.br*>. Acesso em: 27 fev. 2009.

FEITOSA, H. A.; NASCIMENTO, M. C.; GRÁCIO, M. C. C. A propositional version of the logic of the plausible. In: *Anais do V Simpósio Internacional Principia*. Dutra, L. H. de A. e Mortari, C. A. (orgs.). Florianópolis: NEL/UFSC, p. 184–195, 2009b.

REITER, R. A logic for default reasoning. In: *Artificial Intelligence*. v. 13, p. 81-132, 1980.

SETTE, A. M., CARNIELLI, W. A., VELOSO, P. An alternative view of default reasoning and its logic. In: *Pratica*: Proofs, types and categories. HAUESLER, E. H., PEREIRA, L. C. (Eds.). Rio de Janeiro: PUC, 1999. p. 127-158.