# Notas Sobre Lógica Modal: QBF´s

**Flavio S. Yamamoto**

LIDET - *Laboratory of Interactivity and Digital Entertainment Technology*
Instituto de Matemática e Estatística da USP
Rua do Matão, 1010 – 05508-090 - São Paulo, SP

fsy@ime.usp.br

***Resumo.*** *Notas de um rascunho antigo sobre qbf´s e lógica modal. A construção de uma lógica das fórmulas booleanas quantificadas segue o mesmo esquema apresentado nas nota anterior sobre* lógicas modais.

## 1. Introdução

### 1.1. Fórmulas Booleanas Quantificadas

A construção de uma *lógica das fórmulas booleanas quantificadas* (QBF) segue o mesmo esquema apresentado nas nota anterior sobre *lógicas modais*. A base sintática é composta pela estrutura em $\mathsf{CP}(\Phi)$ (vide nota anterior). O conjunto $\mathsf{QBF}(\Phi)$ das *fórmulas booleanas quantificadas* é o menor conjunto contendo todas as fórmulas de $\mathsf{CP}(\Phi)$ e tal que se $\beta \in \mathsf{QBF}(\Phi)$ e $x \in \Phi$, então $\forall x\beta$ e $\exists x\beta$ também são elementos de $\mathsf{QBF}(\Phi)$. Os quantificadores $\forall$ e $\exists$ atuam sobre os valores verdade $\top$ (ou 1) e $\bot$ (ou 0). Dizemos que uma fórmula booleana quantificada (ou $qbf$) é *bem-formada* se todas as variáveis estão quantificadas.

Estamos interessados apenas nas $qbf$'s bem-formadas e que estejam no formato *prenex*, isto é, fórmulas na forma: $Q_1x_1Q_2x_2\cdots Q_mx_m\beta(x_1, x_2, \ldots, x_m)$, com $Q_j \in \{\forall, \exists\}$, $x_j$ variável $(1 \leq j \leq m)$ e $\beta(x_1, x_2, \ldots, x_m)$ fórmula de $\mathsf{CP}(\Phi)$. Chamamos a fórmula $\beta$ de *matriz* da $qbf$. Dada uma $qbf$, dizemos que ela é válida se, e somente se, o valor-verdade de sua matriz for 1 ($\top$). Note que na lógica QBF não há distinção entre verdade e validade.

Intuitivamente, a leitura que fazemos da fórmula $\exists x\beta(x)$ é a de que *existe uma valoração para* $x$ *tal que* $\beta(x)$ *é verdadeira*, analogamente $\forall x\beta(x)$ diz que *para qualquer valoração para* $x$ $\beta(x)$ *é verdadeira*. De modo mais geral, para: $Q_1x_1Q_2x_2\cdots Q_mx_m\beta(x_1, x_2, \ldots, x_m)$, *se* $Q_j = \exists$, *então existe uma valoração para* $x_j$ *tal que* $Q_{j+1}x_{j+1}\cdots Q_mx_m\beta(\cdots, x_{j+1}, \ldots, x_m)$ *é verdadeira*. Por outro lado, *se* $Q_j = \forall$, *então para qualquer valoração para* $x_j$ *tem-se que* $Q_{j+1}x_{j+1}\cdots Q_mx_m\beta(\cdots, x_{j+1}, \ldots, x_m)$ *é verdadeira*.

Podemos associar o processo de determinar se uma $qbf$ é verdadeira ou falsa à construção de uma *árvore de valoração*. Grosso modo, cada fórmula da forma $\forall x\beta(x)$ é substituida por $\beta_0 \wedge \beta_1$ e $\exists x\beta(x)$ por $\beta_0 \vee \beta_1$, tal que $\beta_0$ (resp., $\beta_1$)) é $\beta(x)$ com todas as ocorrências de $x$ substituidas por 0 (resp., 1). Vejamos um exemplo: considere a fórmula

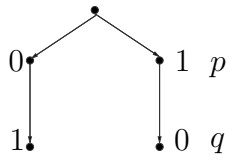


**Figura 1:** Árvore quantificada.

$\forall p \exists q((p \rightarrow \neg q) \wedge (\neg q \rightarrow p))$ e a respectiva árvore das possíveis valorações das letras que ocorrem na matriz.

Efetivamente, a construção da árvore (atribuição de valores-verdade) se dá pela leitura da $qbf$, da esquerda para a direita; os nós da árvore são rotulados com as respectivas valorações das letras do escopo do quantificador. A valoração ocorre do seguinte modo: se o quantificador encontrado for o universal, todas as valorações (distintas) possíveis a variável de seu escopo são consideradas, isto é, tomamos os valores $1$ e $0$. Por outro lado, se o quanficador for o existencial tomamos ou o valor $1$ ou o valor $0$. Os conectivos lógicos funcionam como em CP. Assim, o quantificador universal ramifica-se em dois, enquanto que o quantificador existencial estende-se num único ramo. Veja que a validade de uma $qbf$ ocorre se, e somente se, existe uma árvore para a $qbf$ tal que as valorações asseguram que a matriz assume valor $1$ em algum nó folha.

Stockmeyer [Stockmeyer and Meyer, 1973] e Ladner [Ladner, 1977] utilizaram certos conjuntos de $qbf$'s para estabelecer alguns resultados sobre a *complexidade computacional* de certos *sistemas lógicos* através da codificação desses sistemas para alguma QBF. Eis alguns conjuntos básicos: Definimos os conjuntos $B_j$, para $j \in \mathbb{N}$ e $j \geq 1$, de modo indutivo (para maiores detalhes consultar [Stockmeyer and Meyer, 1973]): $B_1 = \{\beta(X) \mid \exists X[\beta(X) \leftrightarrow 1]\}$ com $X \equiv x_1 x_2 \cdots x_m$ seqüência de variáveis que ocorrem em $\beta$. Se $j \geq 1$, $B_j = \{\beta(X_1, X_2, \ldots, X_j) \mid \exists X_1 \forall X_2 \exists X_3 \cdots Q_j X_j \ [\beta(X_1, X_2, \ldots, X_j) \leftrightarrow 1]\}$ com $Q_l = \forall$ se $l$ par e $Q_l = \exists$ caso contrário. Por fim, $B_\omega = \bigcup_{j \in \mathbb{N}^*} B_j$. Veja que $B_1$ é o conjunto de todas as fórmulas satisfatíveis de CP e $B_\omega$ o conjunto de todas as $qbf$'s válidas. E segue que:

**Fato 1..1.**

1) *(Stockmeyer e Meyer [Stockmeyer and Meyer, 1973]) $B_\omega$ é log-espaço-completo em PSPACE.*
2) *(Cook [Cook, 1971]) $B_1$ é log-espaço completo em NP.*

O fato em $(1)$ decorre da transitividade da relação $\leq_{log}$ ($cf.$ Stockmeyer [Stockmeyer and Meyer, 1973]), diz que todo problema computável em PSPACE é *log-espaço*-redutível a uma linguagem, por exemplo, $\Gamma$ se pudermos mostrar que $B_\omega$ é *log-espaço*-redutível à $\Gamma$. Esta caracterização de $B_\omega$ e $B_1$ pela classe de complexidade à qual pertencem é utilizado direta ou indiretamente nas seções seguintes. Ladner, utilizando a transitividade de $\leq_{log}$ e o fato em (1), estabeleceu relações de complexidade, via reduções *log-espaço*, de $B_\omega$ com os sistemas modais entre K e S4 ($cf.$ teor. 3.1 [Ladner, 1977]). Ainda, Ladner utilizou[1] o fato em $(2)$ para o estudo da complexidade de S5.

Essa estratégia de introduzir quantificadores gera resultados interessantes e intrigantes, por exemplo, o *problema da validade - o de determinar se uma dada fórmula pertence ao conjunto de fórmulas válidas em relação a uma classe de estruturas*, para

[1] O problema da satisfatibilidade em S5 é *log-espaço-redutível* a NP ($cf.$ Ladner [Ladner, 1977, teor. 6.2]).

a maioria dos sistemas modais é *PSPACE*-completo. O que é um tanto quanto surpreendente, tendo em conta que os sistemas modais, a despeito de sua sintaxe proposicional, é essencialmente uma linguagem da *lógica de primeira ordem* (LPO). Lembrando que, por exemplo, o *problema da validade* para a LPO é computacionalmente difícil (a indecidibilidade da LPO é robusta. Somente fragmentos da LPO são decidíveis, e estes fragmentos são tipicamente definidos em termos de quantificadores limitados e na forma prenex. (Nota: o aninhamento arbitrário de operadores modais implica, a primeira vista, que não correponde a um fragmento da LPO de quantificadores limitados e na forma prenex. Porém, uma análise mais cuidadosa revela que a linguagem da lógica modal pode ser vista como fragmento da LPO restrita a 2-variáveis.)

## Referências


Cook, S. A. (1971). The complexity of theorem proving procedures. In *Proceedings of 3rd ACM Symposium on Theory of Computing*, pages 151–158. ACM.

Ladner, R. E. (1977). The computational complexity of provability in systems of modal propositional logic. *SIAM J. Comput.*, 6(3):467–480.

Stockmeyer, L. J. and Meyer, A. R. (1973). Word problems requiring exponential time: Preliminary report. In *Proc. 5th Ann. ACM Symp. on Theory of Computing*, pages 1–9.