

Deste modo, um conjunto de cláusulas S é insatisfatível se existe um conjunto finito de instâncias de S que é insatisfatível. Sob esta ótica, o conjunto $S = \{\neg P(x) \vee Q(x), P(f(y)), \neg Q(f(y))\}$, cujas árvores semânticas foram ilustradas na Figura 4.2 é insatisfatível. O teorema, do modo como é colocado, sugere a implementação proposta no Algoritmo 4.2.

Data: Conjunto de cláusulas S

Result: S é satisfatível, ou insatisfatível

$B = \emptyset$;

while B é satisfatível **do**

$b = \text{nova} - \text{instância}(S)$;

$B = B \cup \{b\}$;

end

Algoritmo 4.2: Algoritmo imediato para a implementação do Teorema de Herbrand

Observe que este algoritmo não garante sempre obter uma resposta dado que o loop somente se encerra no momento em que B passa a apresentar um conjunto de cláusulas insatisfatíveis. Neste sentido torna-se importante definir uma estratégia de produção de cláusulas capazes de abreviar a execução do programa. Entre estas estratégias pode-se destacar como mais significativas o método de Gilmore [66], o método de Davis-Putman [67] e finalmente o método da resolução de Robinson [64, 65].

4.7 O Princípio da Resolução

O Teorema de Herbrand permite decidir se um conjunto de cláusulas dadas é insatisfatível. Porém, como a árvore semântica cresce exponencialmente, dependendo das cláusulas em questão, a quantidade de elementos envolvidos pode superar a capacidade de armazenamento dos maiores computadores existentes, o que a torna impraticável.

Para evitar esta excessiva geração de elementos, Robinson estabeleceu em 1965 outro algoritmo que pode ser aplicado diretamente sobre qualquer conjunto S de cláusulas para verificar sua insatisfabilidade. Sua idéia essencial é verificar se S

possui a cláusula vazia, ou se não, se esta pode ser derivada de S . Caso isto ocorra, S é insatisfatível.

Para isto é importante entender a argumentação da dedução do *modus tollendo ponens* (literalmente, o modo no qual, negando, se afirma) que atualmente é conhecido como silogismo disjuntivo. Basicamente a regra é que:

$$((p \vee q) + (\neg p)) \Rightarrow q$$

ou seja, considere as seguintes afirmações: “Ou ele está mentindo, ou eu estou doido”, mas “Eu não estou doido” logo “Ele está mentindo”.

Assim o *modus tollendo ponens* pode ser visto como uma regra de inferência que pode ser usada para gerar novas cláusulas de S . Acrescendo à S estas cláusulas, alguns nós da árvore original tornam-se nós de falha. Então, o número de nós pode ser reduzido e a cláusula vazia pode eventualmente ser derivada, explicitando a contradição existente.

Estendendo a idéia para qualquer par de cláusulas (não necessariamente unitárias), pode-se enunciar o Princípio da Resolução da seguinte forma: para qualquer par de cláusulas $C1$ e $C2$, se existir um literal $L1$ em $C1$ que for complementar de outro literal $L2$ de $C2$, então remova $L1$ de $C1$ e $L2$ de $C2$ e construa a disjunção dos remanescentes nas cláusulas. A nova cláusula construída é o *resolvente* de $C1$ e $C2$, ou seja:

$$\left(\underbrace{(\overbrace{p}^{L1} \vee q)}_{C1} + \underbrace{(\neg p)}_{L2} \right) \Rightarrow \underbrace{q}_{\text{resolvente}}$$

Assim, o procedimento de prova por resolução usa sentenças na forma de cláusulas. Inicialmente os axiomas e a negação do teorema que se deseja demonstrar são convertidos em cláusulas. Então, novas cláusulas, os resolventes, são deduzidas pela regra de inferência do *modus tollendo ponens*. O teorema principal do procedimento estabelece que se um resolvente não for satisfeito, então nenhum de seus antecedentes o será, o que inclui a negação do teorema inicial. Seu objetivo é obter a cláusula vazia, uma contradição explícita.

Seja o conjunto de cláusulas $S = \{P \vee Q, \neg P \vee Q, P \vee \neg Q, \neg P \vee \neg Q\}$. Logo

as cláusulas são dadas por:

$$\begin{aligned} C1 &= P \vee Q \\ C2 &= \neg P \vee Q \\ C3 &= P \vee \neg Q \\ C4 &= \neg P \vee \neg Q \end{aligned}$$

Não há como produzir a cláusula vazia Λ , mas é possível produzir o resolvente $C5 = Q$ a partir de $C1$ e $C2$. Desta forma a nova coleção de cláusulas passa a ser:

$$\begin{aligned} C1 &= P \vee Q \\ C2 &= \neg P \vee Q \\ C3 &= P \vee \neg Q \\ C4 &= \neg P \vee \neg Q \\ C5 &= Q \end{aligned}$$

Mesmo com a adição da cláusula $C5$, ainda não há como produzir a cláusula vazia Λ , mas combinando $C3$ com $C4$ chega-se ao resolvente $C6 = \neg Q$. Desta vez, a nova coleção é:

$$\begin{aligned} C1 &= P \vee Q \\ C2 &= \neg P \vee Q \\ C3 &= P \vee \neg Q \\ C4 &= \neg P \vee \neg Q \\ C5 &= Q \\ C6 &= \neg Q \end{aligned}$$

Desta vez a combinação $C5$ com $C6$ permite derivar a cláusula vazia Λ , logo S é insatisfável.

Para a lógica de primeira ordem, no entanto, nem sempre os literais complementares são evidentes, como por exemplo, nas cláusulas:

$$\begin{aligned} C1 &= P(x) \vee Q(x) \\ C2 &= \neg P(f(x)) \vee R(x) \end{aligned}$$

Eles somente aparecem se x for substituído por $f(a)$ em $C1$ e x por a em $C2$, resultando

$$\begin{aligned} C1 &= P(f(a)) \vee Q(f(a)) \quad \rightsquigarrow x = f(a) \\ C2 &= \neg P(f(a)) \vee R(a) \quad \rightsquigarrow x = a \end{aligned}$$

que permite chegar no seguinte resolvente:

$$C3 = Q(f(a)) \vee R(a)$$

No entanto, se x for substituído por $f(x)$ em $C1$ tem-se um novo resolvente $C3' = Q(f(x)) \vee R(x)$. Note que $C3$ é uma instância de $C3'$, no sentido que todas as demais cláusulas que podem ser produzidas por este processo são instâncias dela. Desta forma, quando aplicado à lógica de primeira ordem, o princípio da resolução requer que se verifique a situação onde dois predicados são unificáveis, isto é, podem ser feitos "idênticos", antes de realizarem o cancelamento para gerar a cláusula resolvente.

Com isso chega-se ao processo de unificação, cuja idéia básica é encontrar um conjunto minimal de substituições que torna duas fórmulas idênticas a fim de que se possa usar resolução. Por exemplo, os literais:

$$\begin{aligned} & amigos(Luis, Pedro) \\ & amigos(Luis, pai(Luis)) \end{aligned}$$

Podem ser unificados. De fato, para unificar os dois literais, deve-se primeiro verificar se o predicado é o mesmo. Já o caso dos literais:

$$\begin{aligned} & amigos(Luis, Pedro) \\ & parentes(Luis, Pedro) \end{aligned}$$

não podem ser unificados, pois os predicados são diferentes (*amigos* e *parentes*). Se os predicados combinarem, devem-se testar os respectivos argumentos. Se o primeiro combina, continua-se com o segundo e assim por diante. Um modo de implementar este teste é chamar, para cada um dos argumentos, o procedimento de unificação recursivamente: funções, predicados ou constantes só podem combinar se forem idênticas. Uma variável pode se combinar com outra, ou com uma constante, ou com uma função ou com uma expressão predicativa, com a exceção de que estas duas últimas não devem ter qualquer instância das variáveis sendo comparadas.

Definição 4.18 [Substituição] Uma substituição é um conjunto finito da forma $\theta = \{t_1/v_1, t_2/v_2, \dots, t_n/v_n\}$, onde v_i é a variável a ser substituída e t_i um termo substituto diferente de v_i . Desta forma, t_i é o substituto de v_i .

Exemplo 4.15 $\{f(z)/x, y/z\}$ e $\{a/x, g(y)/y, f(g(b))/z\}$ são substituições quais-

quer.

A expressão $E\theta$ é chamada de instância de E , e é obtida a partir de E realizando a troca simultânea de cada ocorrência da variável v_i , $1 \leq i \leq n$, em E pelos termos de t_i . Por exemplo, se $\theta = \{a/x, f(b)/y, c/z\}$ e $E = P(x, y, z)$ então $E\theta = P(a, f(b), c)$.

Definição 4.19 [Composição] Seja $\theta = \{t_1/x_1, \dots, t_n/x_n\}$ e $\lambda = \{u_1/y_1, \dots, u_m/y_m\}$ substituições quaisquer. Uma composição $\theta \circ \lambda$ é a substituição dada pelo conjunto $\{(t_1/\lambda)/x_1, \dots, (t_n/\lambda)/x_n, u_1/y_1, \dots, u_m/y_m\}$ excluídos os elementos $(t_i/\lambda)/x_i$ nos quais $t_i/\lambda = x_i$ e qualquer elemento u_j/y_j tal que y_j esteja entre $\{x_1, x_2, \dots, x_n\}$

Seja:

$$\begin{aligned}\theta &= \{t_1/v_1, t_2/v_2\} &= \{f(y)/x, z/y\} \\ \lambda &= \{u_1/y_1, u_2/y_2, u_3/y_3\} &= \{a/x, b/y, y/z\}\end{aligned}$$

Assim:

$$\begin{aligned}\lambda \circ \theta &= \{(f(y)/\lambda)/x, (z/\lambda)/y, a/x, b/y, y/z\} \\ &= \{f(b)/x, y/y, a/x, b/y, y/z\} \dots \text{substituindo } \lambda \text{ por } t \\ &= \{f(b)/x, a/x, b/y, y/z\} \dots \text{removendo a redundância } y/y \\ &= \{f(b)/x, b/y, y/z\} \dots \text{removendo } a/x \text{ porque } u_i/y_i, y_i = x \in \{x, y\} \\ &= \{f(b)/x, y/z\} \dots \text{removendo } b/y \text{ porque } u_i/y_i, y_i = y \in \{x, y\}\end{aligned}$$

Definição 4.20 [Unificador] A substituição θ é chamada unificador σ para um conjunto de expressões $\{E_1, \dots, E_k\}$, se e somente se, $E_1\theta = E_2\theta = \dots = E_k\theta$.

Definição 4.21 [Unificador Mais Geral - UMG] Um unificador σ para um conjunto de expressões $\{E_1, \dots, E_k\}$ é o unificador mais geral, se e somente se, cada unificador θ para o conjunto, existe uma substituição λ tal que $\theta = \sigma \circ \lambda$.

Definição 4.22 [Conjunto de Discordância] O conjunto de discordância de um conjunto não vazio W de expressões é dado pelos diferentes símbolos que ocupam a mesma posição relativa de argumento da expressão E .

Por exemplo, para o conjunto $W = \{P(x, \underline{f(y, z)}), P(x, \underline{a}), P(x, \underline{g(h(k(x)))})\}$, o conjunto de discordância é dados por $\{f(y, z), a, g(h(k(x)))\}$.

Desta forma, tomando por base as definições apresentadas, pode-se afirmar que o algoritmo de unificação é dados é representado pelo Algoritmo 4.3:

Data: Conjunto W de expressões provenientes das cláusulas da Base de Herbrand

Result: Unificador mais geral UMG

$k = 0$;

$W_k = W$;

$\sigma_k = \epsilon$;

while W_k não é único **do**

if $\exists v_k$ **then**

 Encontre o conjunto de discordância D_k ;

if $\exists v_k, t_k \in D_k$ onde v_k não ocorre em t_k **then**

$\sigma_{k+1} = \sigma \circ \{t_k/v_k\}$;

$W_{k+1} = W_k \{t_k/v_k\}$;

else

W não é unificável ;

$UMG = Nulo$;

 Satura k ;

end

end

$k = k + 1$;

end

$UMG = \sigma_k$ é o unificador mais geral;

Algoritmo 4.3: Unificação de expressões

Para ilustrar o funcionamento do algoritmo 4.3, considere o conjunto $W = \{P(a, x, f(g(y))), P(z, f(z), f(u))\}$. O processo para determinar o unificador mais geral é dado a seguir:

$$\begin{aligned}
\sigma_0 &= \epsilon \\
W_0 &= W \\
W_0 &\quad \text{n\~ao \acute{e} \acute{u}nico (1 s\~o predicado?)} \\
D_0 &= \{a, z\} \\
v_0 &= z \quad \text{n\~ao ocorre em } t_0 = a \\
\sigma_1 &= \sigma_0 \circ \{t_0/v_0\} \\
&= \epsilon \circ \{a/z\} \\
&= \{a/z\} \\
W_1 &= W_0 \{t_0/v_0\} \\
&= \{P(a, x, f(g(y))), P(z, f(z), f(u))\} \{a/z\} \\
&= \{P(a, x, f(g(y))), P(a, f(a), f(u))\} \\
W_1 &\quad \text{n\~ao \acute{e} \acute{u}nico (1 s\~o predicado?)} \\
D_1 &= \{x, f(a)\} \\
v_1 &= x \quad \text{n\~ao ocorre em } t_1 = f(a) \\
\sigma_2 &= \sigma_1 \circ \{t_1/v_1\} \\
&= \{a/z\} \circ \{f(a)/x\} \\
&= \{a/z, f(a)/x\} \\
W_2 &= W_1 \{t_1/v_1\} \\
&= \{P(a, x, f(g(y))), P(a, f(a), f(u))\} \{f(a), x\} \\
&= \{P(a, f(a), f(g(y))), P(a, f(a), f(u))\} \\
W_2 &\quad \text{n\~ao \acute{e} \acute{u}nico (1 s\~o predicado?)} \\
D_2 &= \{g(y), u\} \\
v_2 &= u \quad \text{n\~ao ocorre em } t_2 = g(y) \\
\sigma_3 &= \sigma_2 \circ \{t_2/v_2\} \\
&= \{a/z, f(a)/x\} \circ \{g(y)/u\} \\
&= \{a/z, f(a)/x, g(y)/u\} \\
W_3 &= W_2 \{t_2/v_2\} \\
&= \{P(a, f(a), f(g(y))), P(a, f(a), f(u))\} \{g(y)/u\} \\
&= \{P(a, f(a), f(g(y))), P(a, f(a), f(g(y)))\} \\
&= \{P(a, f(a), f(g(y)))\} \\
W_3 &\quad \text{\acute{e} \acute{u}nico (1 s\~o predicado?)} \\
UMG &= \sigma_3 \\
&= \{a/z, f(a)/x, g(y)/u\}
\end{aligned}$$

Uma vez que o cálculo de unificador mais geral já é conhecido, pode-se retornar ao problema inicial que é o cálculo de resolventes.

Definição 4.23 [Resolvente de Primeira Ordem] Sejam $C1$ e $C2$ duas cláusulas sem variáveis em comum. Sejam $L1$ e $L2$ dois literais das cláusulas $C1$ e $C2$, respectivamente. Se o cálculo do unificador mais geral entre $L1$ e $\neg L2$ é dado por σ , então o resolvente entre $C1$ e $C2$ é dado pela cláusula $(C1\sigma - L1\sigma) \cup (C2\sigma - L2\sigma)$

Desta forma, seja um conjunto de cláusulas $S = \{P(x) \vee Q(x), \neg P(a) \vee R(x)\}$, onde $C1 = P(x) \vee Q(x)$ e $C2 = \neg P(a) \vee R(x)$. As cláusulas $C1$ e $C2$ são candidatas a serem unificadas devido aos literais $L1 = P(x)$ e $L2 = \neg P(a)$.

Para calcular o unificador mais geral de modo a ser aplicada a definição de resolvente de primeira ordem, deve-se tomar $W = \{L1, \neg L2\} = \{P(x), P(a)\}$. Assim, tem-se que o unificador mais geral é dado por:

$$\begin{aligned}
W &= \{P(x), P(a)\} \\
\sigma_0 &= \epsilon \\
W_0 &= W \\
W_0 &\text{ não é único (1 só predicado?)} \\
D_0 &= \{a, x\} \\
v_0 &= x \text{ não ocorre em } t_0 = a \\
\sigma_1 &= \sigma_0 \circ \{t_0/v_0\} \\
&= \epsilon \circ \{a/x\} \\
&= \{a/x\} \\
W_1 &= W_0 \{t_0/v_0\} \\
&= \{P(x), P(a)\} \{a/x\} \\
&= \{P(a), P(a)\} \\
&= \{P(a)\} \\
W_1 &\text{ é único (1 só predicado?)} \\
UMG &= \sigma_1 \\
&= \{a/x\}
\end{aligned}$$

De posse do unificador mais geral, pode-se então determinar o resolvente de $C1$ e $C2$.

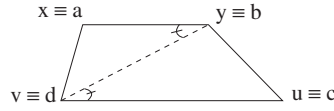


Figura 4.3: Os ângulos interiores alternados formados pela diagonal de um trapézóide são iguais.

Observe que x aparece em ambas as cláusulas $C1$ e $C2$. Contudo, esta variável x de $C1$ é apenas lexicamente igual à variável x de $C2$. Sob a ótica da semântica, estas variáveis são distintas, e por este motivo, uma delas será substituída por y para evitar equívocos, ou seja, $C2 = \neg P(a) \vee R(y)$.

Uma vez resolvida esta questão, o resolvente $C3$ de $C1$ e $C2$ é dado por:

$$\begin{aligned}
 C3 &= (C1\sigma - L1\sigma) \cup (C2\sigma - L2\sigma) \\
 &= (\{P(a), Q(a)\} - \{P(a)\}) \cup (\{\neg P(a), R(y)\} - \{\neg P(a)\}) \\
 &= \{Q(a)\} \cup \{R(y)\} \\
 &= \{Q(a), R(y)\} \\
 &= Q(a) \vee R(y)
 \end{aligned}$$

Neste momento, torna-se importante compreender todo o processo de prova automática de teoremas, a fim de ter uma clara percepção do processo como um todo. Neste sentido, será apresentado a seguir um exemplo completo de dedução. Seja, por exemplo, um trapézóide qualquer $xyuv$, no qual se deseja mostrar que são iguais os ângulos interiores alternados formados pela diagonal do mesmo. A Figura 4.3 apresenta de forma complementar e ilustrativa o que está sendo proposto.

Primeiro deve-se axiomatizar o teorema:

- Seja $T(x, y, u, v)$ um trapézóide com vértice superior esquerdo x , superior direito y , inferior direito u e inferior esquerdo v ;
- Seja $P(x, y, u, v)$ o paralelismo entre o segmento de reta xy e o segmento de reta uv ;
- Seja $E(x, y, z, u, v, w)$ a igualdade entre o ângulo \widehat{xyz} e o ângulo \widehat{uvw} .

Assim, os axiomas do teorema proposto são:

$A_1 = (\forall x)(\forall y)(\forall u)(\forall v) [T(x, y, u, v) \rightarrow P(x, y, u, v)]$, que é a definição de trapezóide.

$A_2 = (\forall x)(\forall y)(\forall u)(\forall v) [P(x, y, u, v) \rightarrow E(x, y, v, u, v, y)]$, isto é, os ângulos interiores de uma reta que cruza duas outras retas paralelas são iguais.

$A_3 = T(a, b, c, d)$, ou seja, uma instância do modelo abstrato de trapezóide que se tem na mente e que foi materializada em um pedaço de papel conforme ilustrado na Figura 4.3.

A partir destes axiomas, deve-se ser capaz de concluir que $E(a, b, d, c, d, b)$ é verdadeiro, isto é, que a fórmula a seguir é válida:

$$A_1 \wedge A_2 \wedge A_3 \rightarrow E(a, b, d, c, d, b)$$

Uma vez que se deseja realizar a prova conforme propôs Herbrand, nega-se a conclusão de modo que $A_1 \wedge A_2 \wedge A_3 \wedge \neg E(a, b, d, c, d, b)$ é uma fórmula insatisfatível.

Para construir o conjunto de cláusulas é preciso escrever a fórmula na forma normal prenex, e em seguida, skolenizá-la. Assim, a fórmula normal prenex para $A_1 \wedge A_2 \wedge A_3 \wedge \neg E(a, b, d, c, d, b)$ são:

$$\begin{aligned} A_1 &= (\forall x)(\forall y)(\forall u)(\forall v) [\neg T(x, y, u, v) \vee P(x, y, u, v)] \\ A_2 &= (\forall x)(\forall y)(\forall u)(\forall v) [\neg P(x, y, u, v) \vee E(x, y, v, u, v, y)] \\ A_3 &= T(a, b, c, d) \end{aligned}$$

A fórmula normal prenex já é uma fórmula conjuntiva, ainda que neste exemplo tenha-se uma degeneração das conjunções, haja vista que cada expressão forma um único literal. A forma de Skolen, por sua vez, exige a retirada dos quantificadores existenciais. Novamente, no caso particular deste exemplo, tais quantificadores não existem, logo:

$$S = \{\neg T(x, y, u, v) \vee P(x, y, u, v), \neg P(x, y, u, v) \vee E(x, y, v, u, v, y), T(a, b, c, d), \neg E(a, b, d, c, d, b)\}$$

Assim,

$$\begin{aligned} C1 &= \neg T(x, y, u, v) \vee P(x, y, u, v) \\ C2 &= \neg P(x, y, u, v) \vee E(x, y, v, u, v, y) \\ C3 &= T(a, b, c, d) \\ C4 &= \neg E(a, b, d, c, d, b) \end{aligned}$$

Inicialmente calcula-se a unificação de $C2$ e $C4$. Fazendo $L1 = E(x, y, v, u, v, y)$ e $L2 = \neg E(a, b, d, c, d, b)$ tem-se que $W = \{L1, \neg L2\}$. Calculando o unificador mais geral tem-se:

$$\begin{aligned}
W &= \{E(x, y, v, u, v, y), E(a, b, d, c, d, b)\} \\
\sigma_0 &= \epsilon \\
W_0 &= W \\
W_0 &\text{ não é único (1 só predicado?)} \\
D_0 &= \{a, x\} \\
v_0 &= x \text{ não ocorre em } t_0 = a \\
\sigma_1 &= \sigma_0 \circ \{t_0/v_0\} \\
&= \epsilon \circ \{a/x\} \\
&= \{a/x\} \\
W_1 &= W_0 \{t_0/v_0\} \\
&= \{E(x, y, v, u, v, y), E(a, b, d, c, d, b)\} \{a/x\} \\
&= \{E(a, y, v, u, v, y), E(a, b, d, c, d, b)\} \\
W_1 &\text{ não é único (1 só predicado?)} \\
D_1 &= \{b, y\} \\
v_1 &= y \text{ não ocorre em } t_1 = b \\
\sigma_2 &= \sigma_1 \circ \{t_1/v_1\} \\
&= \{a/x\} \circ \{b/y\} \\
&= \{a/x, b/y\} \\
W_2 &= W_1 \{t_1/v_1\} \\
&= \{E(a, y, v, u, v, y), E(a, b, d, c, d, b)\} \{a/x, b/y\} \\
&= \{E(a, b, v, u, v, b), E(a, b, d, c, d, b)\} \\
W_2 &\text{ não é único (1 só predicado?)} \\
D_2 &= \{d, v\} \\
v_2 &= v \text{ não ocorre em } t_2 = d \\
\sigma_3 &= \sigma_2 \circ \{t_2/v_2\} \\
&= \{a/x, b/y\} \circ \{d/v\} \\
&= \{a/x, b/y, d/v\}
\end{aligned}$$

$$\begin{aligned}
W_3 &= W_2 \{t_2/v_2\} \\
&= \{E(a, b, v, u, v, b), E(a, b, d, c, d, b)\} \{a/x, b/y, d/v\} \\
&= \{E(a, b, d, u, d, b), E(a, b, d, c, d, b)\} \\
W_3 &\text{ não é único (1 só predicado?)} \\
D_3 &= \{c, u\} \\
v_3 &= u \text{ não ocorre em } t_3 = c \\
\sigma_4 &= \sigma_3 \circ \{t_3/v_3\} \\
&= \{a/x, b/y, d/v\} \circ \{c/u\} \\
&= \{a/x, b/y, d/v, c/u\} \\
W_4 &= W_3 \{t_3/v_3\} \\
&= \{E(a, b, d, u, d, b), E(a, b, d, c, d, b)\} \{a/x, b/y, d/v, c/u\} \\
&= \{E(a, b, d, c, d, b), E(a, b, d, c, d, b)\} \\
&= \{E(a, b, d, c, d, b)\} \\
W_4 &\text{ é único (1 só predicado?)} \\
UMG &= \sigma_4 \\
&= \{a/x, b/y, d/v, c/u\}
\end{aligned}$$

De posse do unificador mais geral, o cálculo do resolvente $C5$, de $C2$ e $C4$ é dado por:

$$\begin{aligned}
C5 &= (C2\sigma - L1\sigma) \cup (C4\sigma - L2\sigma) \\
&= (\{\neg P(a, b, c, d), E(a, b, d, c, d, b)\} - \{E(a, b, d, c, d, b)\}) \cup \\
&\quad (\{\neg E(a, b, d, c, d, b)\} - \{\neg E(a, b, d, c, d, b)\}) \\
&= \{\neg P(a, b, c, d)\} \cup \emptyset \\
&= \neg P(a, b, c, d)
\end{aligned}$$

Deste modo tem-se que a nova coleção de clausulas é dada por:

$$\begin{aligned}
C1 &= \neg T(x, y, u, v) \vee P(x, y, u, v) \\
C2 &= \neg P(x, y, u, v) \vee E(x, y, v, u, v, y) \\
C3 &= T(a, b, c, d) \\
C4 &= \neg E(a, b, d, c, d, b) \\
C5 &= \neg P(a, b, c, d)
\end{aligned}$$

O passo seguinte é prosseguir com a resolução, desta vez unificando $C1$ e $C5$. Nesta etapa, cria-se um $L3 = P(x, y, u, v)$ e $L4 = \neg P(a, b, c, d)$ e tem-se outro conjunto $W = \{L3, \neg L4\}$. Procede-se o cálculo do unificador mais geral, tal como

realizado anteriormente com $W = \{L1, \neg L2\}$. O cálculo do unificador mais geral fornece que $UMG = \{a/x, b/y, c/u, d/v\}$ (deixa-se esta demonstração para o leitor). O resolvente $C6$ é dado por:

$$\begin{aligned}
C6 &= (C1\sigma - L3\sigma) \cup (C5\sigma - L4\sigma) \\
&= (\{\neg T(a, b, c, d), P(a, b, c, d)\} - \{P(a, b, c, d)\}) \cup \\
&\quad (\{\neg P(a, b, c, d)\} - \{\neg P(a, b, c, d)\}) \\
&= \{\neg T(a, b, c, d)\} \cup \emptyset \\
&= \neg T(a, b, c, d)
\end{aligned}$$

Neste sentido, a nova coleção de cláusulas passa ser:

$$\begin{aligned}
C1 &= \neg T(x, y, u, v) \vee P(x, y, u, v) \\
C2 &= \neg P(x, y, u, v) \vee E(x, y, v, u, v, y) \\
C3 &= T(a, b, c, d) \\
C4 &= \neg E(a, b, d, c, d, b) \\
C5 &= \neg P(a, b, c, d) \\
C6 &= \neg T(a, b, c, d)
\end{aligned}$$

A partir desta coleção, seguindo com a resolução, faz-se a unificação de $C3$ e $C6$. Os literais criados para compor o conjunto W são $L5 = T(a, b, c, d)$ e $L6 = \neg T(a, b, c, d)$, isto é, $W = \{L5, \neg L6\} == \{T(a, b, c, d), T(a, b, c, d)\}$. Observe que não há substituição a ser feita e o $UMG = \epsilon$. O resolvente, então, é dado por:

$$\begin{aligned}
C7 &= (C3\sigma - L5\sigma) \cup (C6\sigma - L6\sigma) \\
&= (\{T(a, b, c, d)\} - \{T(a, b, c, d)\}) \cup (\{\neg T(a, b, c, d)\} - \{\neg T(a, b, c, d)\}) \\
&= \emptyset \cup \emptyset \\
&= \Lambda
\end{aligned}$$

Uma vez que se obteve a cláusula vazia em $C7$, realizando derivações de S , conclui-se que S é insatisfatível. Ora, se S é insatisfatível, é porque $A_1 \wedge A_2 \wedge A_3 \rightarrow E(a, b, d, c, d, b)$ é uma fórmula válida.

4.8 Conclusões e leituras recomendadas

Este capítulo foi uma compilação dos tópicos mais relevantes sobre lógica simbólica e prova automática de teoremas de Chang e Lee [68]. Para um estudo

mais aprofundado sobre lógica proposicional e de primeira ordem recomenda-se Kleene [69] e [70]. Um bom livro sobre lógica, com viés voltado para lingüística é McCawley [71]. Uma abordagem bastante leve e introdutória de lógica é encontrada nos capítulos iniciais de Suppes [72], recomendado para leitores sem familiaridade com o assunto. Os conceitos de natureza filosófica para melhor compreensão da lógica podem ser encontrados em Russell [73, 74]. Uma literatura um pouco mais pesada, mas de igual importância a de Russell são os trabalhos de Popper [75], Lakatos e Musgrave [76], e principalmente Wang [77]. Com relação à técnica de solução de problemas, cerne das máquinas de inferência, o capítulo 2 de Green [78] apresenta um resumo rigoroso. Para estudos mais aprofundados existem as excelentes publicações de Chang e Lee [68] e Loveland [79].

4.9 Exercícios

4.1 Para cada uma das fórmulas a seguir, verifique se elas são tautologias, contradições, ou nenhuma das duas opções.

- (a) $\neg(\neg P) \rightarrow P$
- (b) $P \rightarrow (P \wedge Q)$
- (c) $\neg(P \vee Q) \vee \neg Q$
- (d) $(P \rightarrow Q) \rightarrow (\neg Q \rightarrow \neg P)$
- (e) $P \vee (P \rightarrow Q)$
- (f) $(P \wedge (Q \rightarrow P)) \rightarrow P$
- (g) $P \vee (Q \rightarrow \neg P)$
- (h) $P \rightarrow \neg P$
- (i) $\neg P \rightarrow P$

4.2 Escreva as fórmulas a seguir na forma normal disjuntiva.

- (a) $(\neg P \wedge Q) \rightarrow R$

- (b) $P \rightarrow ((Q \wedge R) \rightarrow S)$
- (c) $\neg(P \vee \neg Q) \wedge (S \rightarrow T)$
- (d) $(P \rightarrow Q) \rightarrow R$
- (e) $\neg(P \wedge Q) \wedge (P \vee Q)$

4.3 Escreva as fórmulas a seguir na forma normal conjuntiva.

- (a) $P \vee (\neg P \wedge Q \wedge R)$
- (b) $\neg(P \rightarrow Q)$
- (c) $(\neg P \wedge Q) \vee (P \wedge \neg Q)$
- (d) $\neg(P \rightarrow Q) \vee (P \vee Q)$
- (e) $(P \rightarrow Q) \rightarrow R$

4.4 Prove que $(\neg Q \rightarrow \neg P)$ é uma consequência lógica de $(P \rightarrow Q)$.

4.5 Mostre que Q é uma consequência lógica de $(P \rightarrow Q)$ e P . Esta afirmação é conhecida como regra de *modus ponens*.

4.6 Mostre que R é uma consequência lógica de $P \rightarrow Q$, $\neg P \rightarrow R$, $Q \rightarrow S$, $\neg S$.

4.7 Formaliza matematicamente as sentenças a seguir:

- (a) Toda alma será salva.
- (b) Tudo que sobe, desce.
- (c) Alguns times são ruins.
- (d) Nenhum time será desclassificado.
- (e) Há pessoas não confiáveis.
- (f) Existem ladrões que não são políticos.
- (g) Não existe elefante azul.

- (h) Idéias boas são raras.
- (i) Ninguém gosta de sinal fechado.
- (j) Cariocas não gostam de sinal fechado.
- (k) Toda unanimidade é burra.
- (l) Todo papa-léguas que diz "beep-beep" é esperto.

4.8 Seja $C(x)$ a representação de " x é um vendedor de carros" e $H(x)$, " x é honesto". Traduza as seguintes fórmulas para o Português:

- (a) $(\exists x)C(x)$
- (b) $(\exists x)H(x)$
- (c) $(\forall x)(C(x) \rightarrow \neg H(x))$
- (d) $(\exists x)(C(x) \wedge H(x))$
- (e) $(\exists x)(H(x) \rightarrow C(x))$

4.9 Seja $P(x)$, $L(x)$, $R(x, y, z)$ e $I(x, y)$ átomos representando " x é um ponto", " x é uma linha", " z passa por x e y " e " x é igual a y ", respectivamente. Construa a seguinte fórmula: Para todos dois pontos, existe uma e somente uma linha que passa por ambos os pontos.

4.10 Considere a interpretação $D = a, b$, e os valores assumidos por P : $P(a, a) = 1$ | $P(a, b) = 0$ | $P(b, a) = 0$ | $P(b, b) = 1$. Determine os valores verdade para as seguintes fórmulas:

- (a) $(\forall x)(\exists y)P(x, y)$
- (b) $(\forall x)(\forall y)P(x, y)$
- (c) $(\exists x)(\forall y)P(x, y)$
- (d) $(\exists y)\neg P(a, y)$
- (e) $(\forall x)(\forall y)(P(x, y) \rightarrow P(y, x))$

(f) $(\forall x)P(x, x)$

4.11 Considere a fórmula $A : (\exists x)P(x) \rightarrow (\forall x)P(x)$. Prove que esta fórmula é sempre 1 para um domínio qualquer contendo apenas um único elemento.

4.12 Considere a interpretação $D = 1, 2$, os valores de constantes são $a = 1$ e $b = 2$, com uma função f assumindo os valores $f(1) = 2 \mid f(2) = 1$, e os valores assumidos por P : $P(1, 1) = 1 \mid P(1, 2) = 1 \mid P(2, 1) = 0 \mid P(2, 2) = 0$. Determine os valores verdade para as seguintes fórmulas:

(a) $P(a, f(a)) \wedge P(b, f(b))$

(b) $(\forall x)(\exists y)P(y, x)$

(c) $(\forall x)(\forall y)(P(x, y) \rightarrow P(f(x), f(y)))$

4.13 Transforme as fórmulas a seguir para a forma normal prenex.

a) $(\forall x)(P(x) \rightarrow (\exists y)Q(x, y))$

b) $(\exists x)(\neg((\exists y)P(x, y)) \rightarrow ((\exists z)Q(z) \rightarrow R(x)))$

c) $(\forall x)(\forall y)((\exists z)P(x, y, z) \wedge ((\exists u)Q(x, u) \rightarrow (\exists v)Q(y, v)))$

d) $\neg((\forall x)(A(x) \rightarrow ((\forall y)B(x, y))))$

e) $(\forall x)(\forall y)((\exists z)(A(x, z) \wedge B(y, z)) \rightarrow (\exists u)C(x, y, u))$

f) $(\forall x)A(x) \rightarrow \neg(\exists z)(B(w, z) \rightarrow (\forall y)C(w, y))$

g) $A(x, y) \wedge ((\exists x)B(x) \rightarrow (\forall y)(\forall z)C(x, y, z))$

4.14 Efetue a skolenização das fórmulas a seguir.

a) $(\exists x)(\exists y)(\forall z)(P(x, y) \vee \neg Q(z) \vee R(x))$

b) $(\forall x)[P(x) \rightarrow \{(\forall y)(P(y) \rightarrow P(x)) \wedge \neg(\forall y)(Q(x, y) \rightarrow P(y))\}]$

c) $((\forall x)P(x)) \rightarrow ((\exists x)Q(x) \wedge R(x))$

d) $(\exists x)(P(x, y) \wedge ((\exists y)\neg Q(y) \rightarrow (\exists z)R(z)))$

- e) $(\exists x)(\forall y)(\forall z)(\exists u)(\forall v)(\exists w)(P(x, y, z) \wedge Q(u, v) \wedge \neg R(w))$
- f) $(\forall x)(\exists y)(\exists z)((\neg P(x, y) \wedge Q(x, z)) \vee R(x, y, z))$
- g) $(\forall x)((A(x, y) \rightarrow (\exists y)P(x, y, z)) \rightarrow \neg(\forall z)Q(x, z))$
- h) $(\forall x)((\neg P(x, a) \rightarrow (\exists y)(P(y, g(x))) \wedge (\forall z)(P(z, g(x)) \rightarrow P(y, z))))$
- i) $(\forall x)((P(x) \rightarrow \neg(\forall y)(Q(x, y) \rightarrow (\exists z)P(z))) \wedge (\forall t)(Q(x, y) \rightarrow R(t)))$
- j) $(\forall x)P(x) \wedge ((\forall x)Q(x) \rightarrow (\exists y)R(x, y, z))$

4.15 Determine o universo de Herbrant para o conjunto de cláusulas

$$S = \{P(a), \neg P(x) \vee Q(f(x), a)\}.$$

4.16 Determine o universo de Herbrant para o conjunto de cláusulas

$$S = \{P(a), Q(b, c) \vee P(d)\}.$$

4.17 Determine o universo e a base de Herbrant para o conjunto de cláusulas

$$S = \{P(x, y) \vee Q(f(y)), R(x, y, z)\}.$$

4.18 Determine o universo e a base de Herbrant para o conjunto de cláusulas

$$S = \{P(a, b) \vee Q(f(y)) \vee R(x)\}.$$

4.19 Seja $S = \{P, \neg P \vee Q, \neg Q\}$. Determine a a árvore semântica fechada de S .

4.20 Considere $S = \{P(x), \neg P(x) \vee Q(x, a), \neg Q(y, a)\}$. Determine: (a) o conjunto atômico de S ; (b) a árvore semantica completa de S ; (c) a árvore fechada de S .

4.21 Seja $S = \{P(x), Q(x), \neg Q(x), \neg Q(x) \vee P(x)\}$. Determine a a árvore semântica fechada de S .

4.22 Seja $S = \{P(x), Q(x) \vee R(x), \neg P(x) \vee \neg Q(x), \neg P(x) \vee \neg R(x)\}$. Determine a a árvore semântica fechada de S .

4.23 Resolva o seguinte problema utilizando a prova de Herbrand:

Toda mulher é gentil.

Maria é mulher.

Logo, Maria é gentil.

4.24 Resolva o seguinte problema utilizando a prova de Herbrand:

Todo homem é mortal.

O pai de João é homem.

Logo, o pai de João é mortal.

4.25 Sejam as fórmulas $F_1 = P \rightarrow (\neg Q \vee (R \wedge S))$, $F_2 = P$, $F_3 = \neg S$. Utilize o princípio da resolução para verificar que $G = \neg Q$ é consequência lógica de F_1 , F_2 e F_3 .

4.26 Sejam as fórmulas a seguir. Utilizando o princípio da resolução, mostre a relação de consequência lógica desejada.

a) $F_1 = P \rightarrow S$

$$F_2 = S \rightarrow U$$

$$F_3 = P$$

$$G = U$$

$$F_1, F_2, F_3 \models G$$

b) $F_1 = P \rightarrow (\neg Q \vee (R \wedge S))$

$$F_2 = P$$

$$F_3 = \neg S$$

$$G = \neg Q$$

$$F_1, F_2, F_3 \models G$$

c) $F = (\forall x)(\forall y)(P(x, f(y)) \vee P(y, f(x)))$

$$G = (\exists u)(\exists v)(P(u, f(v)) \vee P(v, f(u)))$$

$$F \models G$$

4.27 Zico é carioca. Zé é carioca. Romário também é carioca. Todo carioca é brasileiro. Construa uma base de conhecimento que modele o cenário aqui descrito. Construa um programa capaz de verificar se determinada interpretação é válida para o predicado `é_brasileiro`, sob a ótica da base de conhecimento. Em seguida crie outro programa que apresente todas as interpretações válidas para `é_brasileiro`.

4.28 João VI é pai de Pedro I e de Miguel de Bragança. Pedro I, por sua vez, é pai de Pedro II e Maria da Glória. Pedro II é pai de Princesa Isabel. Sabe-se que o

pai do pai de um indivíduo também é avô deste indivíduo. Construa um programa capaz de verificar se determinada interpretação é válida para o predicado `é_avô-de`, sob a ótica da base de conhecimento. Em seguida crie outro programa que apresente todas as interpretações válidas para `é_avô-de`.

4.29 O resfriado é uma infecção leve das vias aéreas superiores atingindo principalmente nariz e garganta. Seus sintomas são coriza, febre alta ou baixa e eventualmente espirros. Neste caso deve ser ministrado repouso, líquido e boa alimentação. Se necessário podem ser administrados analgésicos e anti-térmicos. A gripe é uma doença muito contagiosa que ataca as vias respiratórias, nariz, garganta e pulmões. Os pacientes com esta enfermidade apresentam febre alta, dores musculares e nas articulações, dor de cabeça e eventualmente inflamação dos olhos. Não existe remédio para a gripe, porém deve-se prescrever ao paciente repouso, boa alimentação, analgésicos, anti-térmicos e descongestionantes. A pneumonia, por sua vez, é uma infecção aguda que pode atingir os pulmões inteiros ou em partes. Certas pneumonias pioram rapidamente e requerem hospitalização do paciente. Nesta situação os pacientes apresentam tosse, dor no peito, febre alta, calafrios e sudorese. Em muitos casos a sudorese e os calafrios podem não estar presentes no quadro patológico do paciente. O tratamento exige anti-bióticos e oxigênio. Um paciente chegou no hospital com febre alta, dor de cabeça, tosse e dor no peito. Após o atendimento foi ministrado antibiótico. Esta combinação de medicamentos irá tratar sua enfermidade (faça um programa que forneça esta decisão)?

4.30 Alonzo, Kurt, Rudolf e Willard são artistas criativos de grande talento, um é dançarino, um é pintor, um é cantor e um é escritor, não necessariamente nesta ordem. Alonzo e Rudolf estavam na audiência na noite em que o cantor fez seu debut no palco. O escritor publicou a biografia de Willard e pretende escrever a biografia de Alonzo. Rudolf não conhece Alonzo. Rudolf e o escritor ambos tiveram seus retratos pintados ao vivo pelo pintor. Quais são as profissões de Kurt, Willard, Rudolf e Alonzo? (Implemente um programa para fornecer estas respostas).

4.31 Circula na Internet um teste de QI supostamente atribuído a Albert Einstein sobre o qual, supostamente, afirmou que apenas 2% das pessoas são capazes de resolvê-lo. Ainda que a veracidade destas informações possa ser questionada, o teste

por si só é bastante interessante, principalmente se este suposto teste de QI possa ser implementado em uma máquina, completamente desprovida de inteligência. Deste modo, implemente um programa de computador capaz de resolver o problema a seguir:

Há cinco casas de diferentes cores. Em cada casa mora uma pessoa de uma diferente nacionalidade. Esses cinco proprietários bebem diferentes bebidas, fumam diferentes tipos de cigarros e têm um certo animal de estimação. Nenhum deles têm o mesmo animal, fumam o mesmo cigarro ou bebem a mesma bebida (isto é, são conjuntos excludentes).

1. O Inglês vive na casa Vermelha.
2. O Sueco tem Cachorros como animais de estimação.
3. O Dinamarquês bebe Chá.
4. A casa Verde fica do lado esquerdo da casa Branca.
5. O homem que vive na casa Verde bebe Café.
6. O homem que fuma Pall Mall cria Pássaros.
7. O homem que vive na casa Amarela fuma Dunhill.
8. O homem que vive na casa do meio bebe Leite.
9. O Norueguês vive na primeira casa.
10. O homem que fuma Blends vive ao lado do que tem Gatos.
11. O homem que cria Cavalos vive ao lado do que fuma Dunhill.
12. O homem que fuma BlueMaster bebe Cerveja.
13. O Alemão fuma Prince.
14. O Norueguês vive ao lado da casa Azul.
15. O homem que fuma Blends é vizinho do que bebe Água.

Pergunta-se, quem tem um peixe como animal de estimação?

4.32 Verificar se $W = \{Q(f(a), g(x)), Q(y, y)\}$ é unificável.

4.33 Verificar se $W = \{P(x), P(f(x))\}$ é unificável.

4.34 Verificar se $W = \{P(a, f(x)), P(y, z)\}$ é unificável.

4.35 Verificar se $W = \{P(a, x, f(g(y))), P(z, h(z, w), f(w))\}$ é unificável.

Referências Bibliográficas

- [1] HOPCROFT, J. E., *Theory os Machines and Compuatations*, chapter An n log n algorithm for minimizing states in a finite automaton, Academic Press, pp. 189–196, 1971.
- [2] MONTEIRO, A. A., PAULO, J. D. S., *Aritmética Racional*. Lisboa, Livraria Avelar Machado, 1945.
- [3] IFRAH, G., *Os Números: a história de uma grande invenção*. São Paulo, Globo S.A., 2005.
- [4] DEDEKIND, R., *Was sind und was sollen die Zahlen*, v. 3, *Gesammelte Mathematische Werke*. New York, Chelsea Publishing Company, 1969. pp. 335-391.
- [5] GÖDEL, K., *Collected Works*, v. 2, *Gesammelte Mathematische Werke*. Oxford, Oxford University Press, 1990.
- [6] ISRAEL, D., “Reflections on Gödel’s and Gandy’s Reflection on Turing’s Thesis”, *Minds and Machines*, v. 12, n. 2, pp. 181–201, 2002.
- [7] SOBRINHO, J. Z., “Aspectos da Tese de Church-Turing”, *Revista Matemática Universitária - USP*, v. 1, n. 6, pp. 1–23, 1987.
- [8] MCDERMOTT, D., “Artificial Intelligence Meets Natural Stupidity”, *SI-GART Newsletter*, v. 57, pp. 4–9, April 1976.
- [9] SETTI, M. D. O. G., *O Processo de Discretiza çã o do Raciocínio Matemático na Tradu çã o para o Raciocínio Computacional*, Report, Universidade Federal do Paraná, 2009.

- [10] GUERREIRO, G., “A Vanguarda Matemática e os Limites da Razão”, *Scientific American Brasil*, v. 5, n. 12, pp. 39–56, 2007. Coleção Gênios da Ciência.
- [11] SMULLYAN, R., *Gödel’s Incompleteness Theorems*. Oxford University Press, 1992.
- [12] GÖDEL, K., *The Undecidable*, chapter On Formally Undecidable Propositions of Principia Mathematica and Related Systems, New York, Raven Press, pp. 5–38, 1965.
- [13] WHITEHEAD, A. N., RUSSELL, B., *Principia Mathematica*. Londres, Cambridge University Press, 1913.
- [14] NAGEL, E., NEWMAN, J. R., *Gödel’s Proof*. USA, Routledge, 1989.
- [15] GÖDEL, K., “Über formal unentscheidbare Sätze der Principia Mathematica und verwandter Systeme - On Formally Undecidable Propositions of Principia Mathematica and Related Systems”, *Kurt Gödel: Collected Works*, v. 1, n. 1, pp. 144–195, 1986. Tradução para o inglês por Martin Hirzel. www.research.ibm.com/people/h/hirzel/papers/canon00-goedel.pdf [capturado em 13 de agosto de 2011].
- [16] MELO, A. C. V. D., SILVA, F. S. C. D., *Modelos Clássicos de Computação*, Coleção Schaum. São Paulo, Thomson, 2006.
- [17] KUBRUSLY, R. D. S., *Uma viagem informal ao Teorema de Gödel ou (O preço da matemática é o eterno matemático)*, Report, Universidade Federal do Rio de Janeiro, 2007. IM/UFRJ.
- [18] SMULLYAN, R., *Recursion Theory for Metamathematics*. Oxford University Press, 1993.
- [19] SMULLYAN, R., *What’s the Name of This Book*. Penguin Books, 1978.
- [20] SMULLYAN, R., *Forever Undecided: A Puzzle Guide to Gödel*. Oxford University Press, 1987.
- [21] GOLDSTEIN, R., *Incompleteness: The Proof and Paradox of Kurt Gödel*. W. W. Norton Company, Inc., 2005.

- [22] HOFSTADTER, D. R., *Gödel, Escher e Bach: an Eternal Golden Braid*. Nova Iorque, Basic Books, 1979.
- [23] Rodríguez-Consuegra, F. A. (ed.), *Kurt Gödel - Unpublished Philosophical Essays*. Berlin, Birkhäuser Verlag, 1995.
- [24] Feferman, S., *et al.* (eds.), *Kurt Gödel - Collected Works*, v. I, II, III. New York, Oxford University Press, 1986.
- [25] WANG, H., *Reflections on Kurt Gödel*. Cambridge, Massachusetts, The MIT Press, 1988.
- [26] SUPPES, P., *Axiomatic Set Theory*. New York, Dover, 1972.
- [27] LIPSCHUTZ, S., *Teoria Elementar dos Conjuntos*, Cole çã o Schaum. Sã o Paulo, McGraw-Hill do Brasil, 1990.
- [28] SUPPES, P., *Axiomatic Theory Set*. USA, Van Nostrand Company Inc., 1960.
- [29] MELLO, F. L. D., CARVALHO, R. L. D., “Knowledge Geometry”, *Journal of Information and Knowledge Management*, v. 14, pp. 1550028, 2015.
- [30] STOLL, R. R., *Set Theory and Logic*. USA, Dover Publications Inc., 1961.
- [31] MIRAGLIA, F., *Teoria dos Conjuntos: um mínimo*. Brasil, Edusp - Editora da Universidade de São Paulo, 1992.
- [32] HALMOS, P., *Teoria Ingênua dos Conjuntos*. Brasil, Edusp - Editora da Universidade de São Paulo, 1970.
- [33] GRATZER, G., *Universal Algebra*. USA, Van Nostrand Company Inc., 1968.
- [34] COHN, P. M., *Universal Algebra*. USA, Harper and Row, 1965.
- [35] GALLIER, J. H., *Logic for Computer Science*. USA, John Wiley and Sons Inc., 1987.
- [36] PREPARATTA, F. P., YEH, R. T., *Introduction to Discreate Structures for Computer Science and Engineering*. USA, Addison-Wesley Publishing Company, 1973.

- [37] SHOENFIELD, J. R., *Degrees of Unsolvability*. North-Holland Publishing Company, 1971.
- [38] BRAINERD, W. S., LANDWEBER, L. H., *Theory of Computation*. USA, John Wiley and Sons, 1974.
- [39] EILENBERG, S., ELGOT, C., *Recursiveness*. New York: Academic Press, 1970.
- [40] CARVALHO, R. L. D., OLIVEIRA, C. M. G. M. D., *Modelos de Computação e Sistemas Formais*, 11^a Escola de Computação. Rio de Janeiro, Universidade Federal do Rio de Janeiro, 1998.
- [41] BRAINERD, W. S., LANDWEBER, L. H., *Theory of Computation*. John Wiley & Sons, 1974.
- [42] KLEENE, S. C., *Introduction to Metamathematics*. D. Van Nostrand Company, Inc., 1952.
- [43] Rogers Jr., H., *Theory of Recursive Functions and Effective Computability*. USA, McGraw-Hill Book Company, 1967.
- [44] DAVIS, M., *Computability and Unsolvability*. New York, Dover, 1983.
- [45] BOOLOS, G. S., JEFFREY, R. C., *Computability and Logic*. Cambridge University Press, 1974.
- [46] MARTIN DAVIS, R. S., WEYUKER, E. J., *Computability Complexity and Languages*. New York, Academic Press, 1994.
- [47] MALLOZZI, J. S., LILLO, N. J. D., *Computability with Pascal*. New Jersey, Prentice-Hall, Inc., 1984.
- [48] TURING, A. M., *The Undecidable*, chapter On Computable Numbers, with an Application to the Entscheidungsproblem, New York, Raven Press, pp. 115–151, 1965.
- [49] ELGOT, C. C., ROBINSON, A., “Random-access stored-program machines, an approach to programming languages”, *Journal of the ACM*, v. 11, pp. 365–399, 1964.

- [50] PREPARATTA, F. P., YEH, R. T., *Introduction to Discrete Structures for Computer Science and Engineering*. Addison-Wesley Publishing Company, 1973.
- [51] HENNIE, F., *Introduction to Computability*. Addison-Wesley Publishing Company, 1977.
- [52] NELSON, R. J., *Introduction to Automata*. USA, Jonh Wiley & Sons, Inc., 1968.
- [53] CARVALHO, R. L. D., *Máquinas, Programas e Algoritmos*, 2^a Escola de Computação. Campinas, Universidade Estadual de Campinas, 1981.
- [54] MENDELSON, E., *Introduction to Mathematical Logic*, Cole Mathematics Series. 3 ed. The Wadsworth and Brooks, 1987.
- [55] MANIN, Y. I., *A Course in Mathematical Logic*, Graduate Texts in Mathematics 53. 1 ed. Springer-Verlag, 1977.
- [56] HOMER, S., SELMAN, A. L., *Computability and Complexity Theory*, Texts in Computer Science. 2 ed. Springer, 2011.
- [57] CUTLAND, N. J., *Computability: An introduction to recursive function theory*. Cambridge University Press, 1980.
- [58] SIPSER, M., *Introduction to The Theory of Computation*, Course Technology Series. 2 ed. Thomson, 2006.
- [59] WALTER CARNIELLI, R. L. E., *Computability: computable functions, logic and the foundations of mathematics*. Belmont, Wadsworth and Brooks, 1989.
- [60] TARSKI, A., *Logic, semantics, metamathematics*, chapter Fundamental concepts of the methodology of the deductive sciences, London, Oxford at the Clarendon Press, pp. 60–109, 1969.
- [61] ADAM YOUNG, M. Y., *Malicious Cryptography: Exposing Cryptovirology*. John Wiley and Sons Inc., 2004.

- [62] BONFANTE, G., KACZMAREK, M., MARION, J.-Y., *A Classification of Viruses through Recursion Theorems*, volume 4497 of Lecture Notes in Computer Science. 2 ed. CiE 2007, 2007.
- [63] MACHTEY, M., YOUNG, P., *An Introduction to the General Theory of Algorithms*. New York, North Holland, 1978.
- [64] ROBINSON, J. A., “A machine oriented logic based on the resolution principle”, *J. Assoc. Comput.*, v. 12, pp. 23–41, 1965.
- [65] ROBINSON, J. A., “Automatic deduction with hyper-resolution”, *Internat. J. Comput. Math*, v. 1, pp. 227–234, 1965.
- [66] GILMORE, P. C., “A proof method for quantification theory: Its justification and realization”, *IBM Journal of Research and Development*, v. 4, n. 1, pp. 28–35, 1960.
- [67] DAVIS, M., PUTNAM, H., “A computing procedure for quantification theory”, *Journal of the ACM*, v. 7, n. 3, pp. 201–215, 1960.
- [68] CHANG, C.-L., LEE, R. C.-T., *Symbolic Logic and Mechanical Theorem Proving*. Academic Press Inc, 1973.
- [69] KLEENE, S. C., *Mathematical Logic*. Wiley, 1967.
- [70] HILBERT, D., ACKERMANN, W., *Prinmciples of Mathematical Logic*. Chelsea, 1950.
- [71] MCCAWLEY, J. D., *Everything That Linguists Have Always Wanted To Know About Logic*. 2 ed. The University of Chicago Press, 1993.
- [72] SUPPES, P., *Introduction to Logic*. D. van Nostrand, 1966.
- [73] RUSSELL, B., *A Filosofia do Atomismo Lógico*, *Lógica e Conhecimento*. 1 ed. Abril Cultural, 1974. (Os Pensadores, 42).
- [74] RUSSELL, B., *Significado e Verdade*. 1 ed. Zahar, 1978.
- [75] POPPER, K. R., *A Lógica da Pesquisa Científica*. 2 ed. Cultrix, 1974.

- [76] LAKATOS, I., , MUSGRAVE, A., *A Crítica e o Desenvolvimento do Conhecimento*. 1 ed. EDUSP, Cultrix, 1979. Tradução: M. O. Caiado.
- [77] WANG, H., *From Mathematics to Philosophy*. 1 ed. Cultrix, 1974.
- [78] GREEN, C. C., *The Application of Theorem Proving to Question-Answering Systems*. Ph.D. dissertation, Stanford, June 1969. AI Project MEMO AI-96.
- [79] LOVELAND, D. W., *Automated Theorem Proving: a Logical Basis*. 1 ed. North Holland, 1978.
- [80] HUGHES, G. E., LONDEY, D. G., *The Elements of Formal Logic*. USA, Methuen and Co Ltd, 1965.
- [81] BOOK, R. V., OTTO, F., *String-Rewriting Systems*. USA, Springer-Verlag, 1993.
- [82] HOPCROFT, J. E., ULLMAN, J. D., *Introduction to Automata Theory, Language and Computation*. USA, Addison-Wesley Publishing Company, 1979.
- [83] HOPCROFT, J. E., ULLMAN, J. D., *Formal Languages and their Relation to Automata*. USA, Addison-Wesley Publishing Company, 1969.
- [84] CURRY, H. B., *Foundations of Mathematical Logic*. New York, Academic Press, 1977.
- [85] SMULLYAN, R., *Theory of Formal Systems*. USA, Princeton, 1961.
- [86] BERSTEL, J., BOASSON, L., CARTON, O., *et al.*, *Handbook of Automata: from Mathematics to Applications*, chapter Minimization of automata, European Mathematical Society, pp. 189–196, 2010.
- [87] BEAL, M. P., CROCHEMORE, M., “Minimizing incomplete automata”, *Workshop on Finite State Methods and Natural Language Processing*, , september 2008. Ispra.
- [88] VALMARI, A., LEHTINEN, P., “Efficient minimization of DFAs with partial transition”, *Proc. 25th Symp. Theoretical Aspects of Comp. Sci.*, v. 08001, pp. 645–656, 2008. S. Albers and P. Weil, editors.

- [89] PAPADONIKOLAKIS, M., BOUGANIS, C.-S., CONSTANTINIDES, G.,
“Performance comparison of GPU and FPGA architectures for the SVM training problem”, *IEEE International Conference on FieldProgrammable Technology*, pp. 388–391, 2009.
- [90] MU, S., WANG, C., LIU, M., *et al.*, “Evaluating the potential of graphics processors for high performance embedded computing”, *Proc. IEEE Design, Automation and Test in Europe Conference and Exhibition*, pp. 709–714, 2011.
- [91] KAI HWANG, F. A. B., *Computer Architecture and Parallel Processing*. McGraw-Hill, 1984.
- [92] AGARWAL, P., KRISHNAN, S., MUSTAFA, N., *et al.*, *Algorithms - ESA 2003, Lecture Notes in Computer Science*, chapter Streaming Geometric Optimization Using Graphics Hardware, Springer Berlin-Heidelberg, pp. 115–151, 2003.
- [93] TANENBAUM, A. S., *Organização Estruturada de Computadores*. 3 ed. Prentice Hall do Brasil, 1997.
- [94] BACKUS, J., “Can Programming be Liberated from von Neumann Style? A functional style and its algebra of program”, *ACM Turing Award Lecture, Communications of the ACM*, v. 21, n. 8, pp. 613–641, 1978.
- [95] OWENS, J. D., LUEBKE, D., GOVINDARAJU, N., *et al.*, “A Survey of General-Purpose Computing on Graphics Hardware”, *Eurographics 2005, State of the Art Reports*, pp. 21–51, 2005.
- [96] GUSTAFSON, J. L., “Reevaluating Amdahl’s law”, *Communications of the ACM*, v. 5, n. 31, pp. 532, 1988.
- [97] HANDLER, W., *Parallel Processing Systems, an advanced course*, chapter Innovative computer architecture - how to increase parallelism but not complexity, Cambridge University Press, pp. 1–41, 1982.
- [98] LOBUR, J., NULL, L., *The Essentials of Computer Organization And Architecture*. Jones and Bartlett Pub, 2006.

- [99] LEWIS, H. R., PAPADIMITRIOU, C. H., *Elements of the Theory of Computation*. 2 ed. New York, Prentice-Hall, 1998.
- [100] DUNNE, P., *Computability Theory: Concepts and Applications*. Ellis Horwood, 1991.
- [101] AARONSON, S., “NP-complete Problems and Physical Reality”, *ACM SIGACT News*, , march 2005. Complexity Theory Column 46.