Lógica Computacional

Aula Teórica 19: Forma Normal Prenex

António Ravara   Simão Melo de Sousa

Departamento de Informática, Faculdade de Ciências e Tecnologia, Universidade Nova de Lisboa

Departamento de Informática, Faculdade Engenharia, LISP & Release Group Universidade Beira Interior

Formas normais na primeira ordem

Objectivo

Determinar a validade de raciocínios (ou de fórmulas) semânticamente, mas de forma eficaz e automática.

Meio: algoritmo

Define-se uma função que recebe uma fórmula e devolve a sua natureza (possível, contraditória ou válida).

Abordagens computacionais

Há vários algoritmos, de acordo com formas especiais em que as fórmulas podem estar. Uns são mais eficazes que outros. Vamos ver o mais eficaz: a *resolução*.

Forma Normal Prenex

Ideia

Os quantificadores estão todos "na cabeça" da fórmula.

Exemplos

- $Q(x) \vee P(x, y)$;
- $\forall x \, \exists y \, f(x) = y$;
- $\forall x \, \exists y \, P(x, y, z)$;
- $\exists x \, (f(x) = y \wedge P(x, y, z))$.

Contra-Exemplos

- $Q(x) \vee \exists y \, P(x, y)$;
- $\forall x \, \exists y \, f(x) = y \wedge \forall x \, \exists y \, P(x, y, z)$;
- $f(x) = y \wedge \forall x \, \exists y \, P(x, y, z)$;
- $\neg \exists x \, (f(x) = y \wedge P(x, y, z))$.

Forma Normal Prenex

Definição 19.1

Uma fórmula φ da linguagem de primeira ordem está na Forma Normal Prenex ou FNP (e escreve-se FNP(φ)), se

$$\varphi = Q_1 x_1 \dots Q_n x_n \, \psi$$

sendo:

- Cada Q_i um quantificador, com $1 \leq i \leq n$ e $n \geq 0$;
- ψ uma fórmula de primeira ordem sem quantificadores.

Se ψ está na forma normal conjuntiva (aplicando o conceito a fórmulas de primeira ordem), então φ diz-se uma formal normal conjuntiva prenex ou FNCP (e escreve-se FNCP(φ)).

Qualquer fórmula é equivalente a uma FNCP

Lema 19.1: lema das formas normais

Para qualquer fórmula de primeira ordem φ existe uma fórmula de primeira ordem ψ tal que $\varphi \equiv \psi$ e FNP(ψ).

Prova

Mostra-se por indução estrutural.

Casos base: as fórmulas atómicas já estão na FNP.

Passo. Analisa-se por casos a forma de φ. Mostra-se a prova de dois casos tipo; os restantes têm prova semelhante.

Seja $\varphi \stackrel{\text{def}}{=} \exists x\, \phi$. Por hipótese de indução existe $\psi \equiv \phi$ tal que FNP(ψ), logo $\varphi \equiv \exists y\, \psi$ para alguma variável y e FNP(φ).

Resultado central

Lema das Formas Normais

Para qualquer fórmula de primeira ordem φ existe uma fórmula de primeira ordem ψ tal que $\varphi \equiv \psi$ e FNP(ψ).

Prova

Seja agora $\varphi \stackrel{\text{def}}{=} \varphi_1 \wedge \varphi_2$.

Por hipótese de indução existem ψ_1, ψ_2 tal que $\varphi_1 \equiv \psi_1$ e FNP(ψ_1), e $\varphi_2 \equiv \psi_2$ e FNP(ψ_2). Logo, sendo $n, m \geq 0$, cada Q_i e R_j quantificadores, com $1 \leq i \leq n$ e $1 \leq j \leq m$, e ψ_1', ψ_2' fórmulas de primeira ordem sem quantificadores, tem-se que

$$\psi_1 = Q_1 x_1 \, \ldots Q_n x_n \, \psi_1'$$

$$\psi_2 = R_1 x_1 \, \ldots R_m x_m \, \psi_2'$$

Lema das Formas Normais

Continuação da prova: caso $\varphi \stackrel{\text{def}}{=} \varphi_1 \wedge \varphi_2$

Note-se que não se tem que FNP($\psi_1 \wedge \psi_2$). Como transformar para FNP?

Provou-se já que se $x \notin \mathsf{VL}(\psi)$ então $\forall x\, \varphi \wedge \psi \equiv \forall x\, (\varphi \wedge \psi)$ e $\exists x\, \varphi \wedge \psi \equiv \exists x\, (\varphi \wedge \psi)$.
Facilmente se generaliza este resultado, obtendo-se $Q_1 x_1 \ldots Q_n x_n\, \varphi \wedge \psi \equiv Q_1 x_1 \ldots Q_n x_n\ (\varphi \wedge \psi)$, se $x_1 \ldots, x_n \notin \mathsf{VL}(\psi)$, sendo $Q_1, \ldots, Q_n$ quantificadores.

Também se mostrou já a congruência-α, ou seja, $\forall x\, \varphi \equiv \forall y\, \varphi\{y/x\}$ e que $\exists x\, \varphi \equiv \exists y\, \varphi\{y/x\}$.

Com estas duas leis mostra-se que existe $\psi \equiv \psi_1 \wedge \psi_2$ tal que FNP(ψ).

Continuação da prova: caso $\varphi \stackrel{\text{def}}{=} \varphi_1 \wedge \varphi_2$

Tem-se então que FNP(ψ_1) e FNP(ψ_2) tal que $\varphi_1 \equiv \psi_1$ e $\varphi_2 \equiv \psi_2$, e $\varphi \equiv \psi_1 \wedge \psi_2 = Q_1 x_1 \ldots Q_n x_n \, \psi_1' \wedge R_1 x_1 \ldots R_m x_m \, \psi_2'$.

Tomam-se variáveis novas (*i.e.*, não usadas nas fórmulas ψ_1 e ψ_2) todas diferentes entre si, $y_1, \ldots, y_n, z_1, \ldots, z_m$. Seja $\psi_1'' = \psi_1'\{y_1, \ldots, y_n / x_1, \ldots, x_n\}$ e $\psi_2'' = \psi_2'\{z_1, \ldots, z_m / x_1, \ldots, x_m\}$

Aplicando repetidamente a congruência-α e duas vezes a lei do âmbito dos quantificadores, usando a comutatividade antes da segunda aplicação, obtém-se uma fórmula em FNP.

$$
\begin{aligned}
\psi_1 \wedge \psi_2 &= Q_1 x_1 \ldots Q_n x_n \, \psi_1' \wedge R_1 x_1 \ldots R_m x_m \, \psi_2' \\
&\equiv Q_1 y_1 \ldots Q_n y_n \, \psi_1'' \wedge R_1 z_1 \ldots R_m z_m \, \psi_2'' \\
&\equiv Q_1 y_1 \ldots Q_n y_n \, R_1 z_1 \ldots R_m z_m \, (\psi_1'' \wedge \psi_2'')
\end{aligned}
$$

Conversão de uma fórmula

Seja $\varphi = \forall x\,(P(x) \wedge \exists y\,\neg R(y,z)) \to \exists x\,\forall y\,Q(x,y,z)$

Faz-se a conversão "manual" de φ para FNP.

$$
\begin{array}{rcl}
\varphi & \equiv & \neg\forall x\,(P(x) \wedge \exists y\,\neg R(y,z)) \vee \exists x\,\forall y\,Q(x,y,z) \\
 & \equiv & \exists x\,\neg(P(x) \wedge \exists y\,\neg R(y,z)) \vee \exists x\,\forall y\,Q(x,y,z) \\
 & \equiv & \exists x\,(\neg P(x) \vee \neg\exists y\,\neg R(y,z)) \vee \exists x\,\forall y\,Q(x,y,z) \\
 & \equiv & \exists x\,(\neg P(x) \vee \forall y\,\neg\neg R(y,z)) \vee \exists x\,\forall y\,Q(x,y,z) \\
 & \equiv & \exists x\,(\forall y\,(\neg P(x) \vee R(y,z)) \vee \exists x\,\forall y\,Q(x,y,z) \\
 & \equiv & \exists x\,\forall y\,(\neg P(x) \vee R(y,z)) \vee \exists u\,\forall v\,Q(u,v,z) \\
 & \equiv & \exists x\,\forall y\,((\neg P(x) \vee R(y,z)) \vee \exists u\,\forall v\,Q(u,v,z)) \\
 & \equiv & \exists x\,\forall y\,\exists u\,\forall v\,((\neg P(x) \vee R(y,z)) \vee Q(u,v,z))
\end{array}
$$

É possível implementar este procedimento.

Transformação de fórmulas para FNP

Objectivo

Definir um algoritmo para transformar qualquer fórmula de Lógica de Primeira Ordem para a Forma Normal Prenex (FNP).

Definição 19.2: Conjuntos de fórmulas de primeira ordem

1. Seja E_Σ^X o conjunto das fórmulas de primeira ordem que se obtém considerando o conectivo de negação ($\neg$) primitivo.
2. Seja $G_\Sigma^X \subseteq E_\Sigma^X$ o conjunto das fórmulas de primeira ordem com negação que não contêm o conectivo de implicação ($\rightarrow$).

Atenção: nas definições das funções recursivas que se apresentam a seguir, assume-se que a análise dos casos se faz sequencialmente (de cima para baixo), sendo cada caso testado apenas se as condições dos anteriores são falsas.

Definição 19.3: Eliminação da implicação

Seja ImplFree: $E_\Sigma^X \to G_\Sigma^X$ a seguinte função.

$$\mathrm{ImplFree}(\varphi) = \begin{cases} \neg\mathrm{ImplFree}(\varphi_1), & \text{se } \varphi = \neg\varphi_1 \\ \mathrm{ImplFree}(\varphi_1) \vee \mathrm{ImplFree}(\varphi_2), & \text{se } \varphi = \varphi_1 \vee \varphi_2 \\ \mathrm{ImplFree}(\varphi_1) \wedge \mathrm{ImplFree}(\varphi_2), & \text{se } \varphi = \varphi_1 \wedge \varphi_2 \\ \neg\mathrm{ImplFree}(\varphi_1) \vee \mathrm{ImplFree}(\varphi_2), & \text{se } \varphi = \varphi_1 \to \varphi_2 \\ \forall x\, \mathrm{ImplFree}(\varphi_1), & \text{se } \varphi = \forall x\, \varphi_1 \\ \exists x\, \mathrm{ImplFree}(\varphi_1), & \text{se } \varphi = \exists x\, \varphi_1 \\ \varphi, & \text{caso contrário} \end{cases}$$

Definição 19.4: Eliminação das duplas negações

Seja NNFC: $G_{\Sigma}^{X} \to G_{\Sigma}^{X}$ a seguinte função.

$$\mathrm{NNFC}(\varphi) = \begin{cases} \mathrm{NNFC}(\varphi_1), & \text{se } \varphi = \neg\neg\varphi_1 \\ \mathrm{NNFC}(\neg\varphi_1) \vee \mathrm{NNFC}(\neg\varphi_2), & \text{se } \varphi = \neg(\varphi_1 \wedge \varphi_2) \\ \mathrm{NNFC}(\neg\varphi_1) \wedge \mathrm{NNFC}(\neg\varphi_2), & \text{se } \varphi = \neg(\varphi_1 \vee \varphi_2) \\ \mathrm{NNFC}(\varphi_1) \vee \mathrm{NNFC}(\varphi_2), & \text{se } \varphi = \varphi_1 \vee \varphi_2 \\ \mathrm{NNFC}(\varphi_1) \wedge \mathrm{NNFC}(\varphi_2), & \text{se } \varphi = \varphi_1 \wedge \varphi_2 \\ \exists x\, \mathrm{NNFC}(\neg\varphi_1), & \text{se } \varphi = \neg\forall x\, \varphi_1 \\ \forall x\, \mathrm{NNFC}(\neg\varphi_1), & \text{se } \varphi = \neg\exists x\, \varphi_1 \\ \forall x\, \mathrm{NNFC}(\varphi_1), & \text{se } \varphi = \forall x\, \varphi_1 \\ \exists x\, \mathrm{NNFC}(\varphi_1), & \text{se } \varphi = \exists x\, \varphi_1 \\ \varphi, & \text{caso contrário} \end{cases}$$

Definição 19.5: Alargamento do âmbito dos quantificadores

Seja $* \in \{\vee, \wedge\}$, $Q \in \{\forall, \exists\}$, $x_0 \notin \mathsf{V}(\varphi)$.

Scope: $G_\Sigma^X \to G_\Sigma^X$ é a seguinte função.

$$\mathrm{Scope}(\varphi) = \begin{cases} Qx\,\mathrm{Scope}(\varphi_1), & \text{se } \varphi = Qx\,\varphi_1 \\ Qx_0\,\mathrm{Scope}(\varphi_1\{x_0/x\} * \varphi_2), & \text{se } \varphi = Qx\,\varphi_1 * \varphi_2 \\ & \text{ou } \varphi = \varphi_2 * Qx\,\varphi_1 \\ \mathrm{Scope}(\mathrm{Scope}(\varphi_1) * \mathrm{Scope}(\varphi_2)), & \text{se } \varphi = \varphi_1 * \varphi_2 \\ & \text{e nenhuma começa} \\ & \text{com um quantificador} \\ \varphi, & \text{se não tem quantif.} \end{cases}$$

Algoritmo $\mathcal{P}$ de transformação para FNP

Definição 19.6: Algoritmo $\mathcal{P}$

Seja $\mathcal{P} : E_\Sigma^X \rightarrow \ G_\Sigma^X$ a seguinte função.

$$\mathcal{P}(\varphi) = \mathrm{Scope}(\mathrm{NNFC}(\mathrm{ImplFree}(\varphi)))$$

Definição 19.7: Forma Normal da Negação

Uma fórmula $\varphi \in G_\Sigma^X$ diz-se que está na forma normal da negação (e escreve-se FNN(φ)), se só as suas subfórmulas que são fórmulas atómicas estão negadas.

Lema 19.2: optimização

1. Se $\varphi \in G_\Sigma^X$ então ImplFree(φ)=φ.
2. Se FNN(φ) então NNFC(φ)=φ.
3. Se FNP(φ) então Scope(φ)=φ.

Correcção do algoritmo $\mathcal{P}$

Teorema 19.1: conversão de fórmulas de primeira ordem para para FNP

Dada $\varphi \in E_\Sigma^X$, a fórmula $\psi = \mathcal{P}(\varphi)$ é tal que $\psi \in G_\Sigma^X$, $\varphi \equiv \psi$ e FNP(ψ).

Assuma-se que o algoritmo CNFC definido para fórmulas proposicionais se aplica também a fórmulas de primeira ordem sem quantificadores.

Lema 19.3: conversão para FNCP

Dada uma fórmula $\varphi = Q_1 x_1 \ldots Q_n x_n\, \psi$ da linguagem de primeira ordem tal que FNP(φ), obtém-se uma fórmula equivalente ϕ tal que FNCP(ϕ) fazendo $\phi = Q_1 x_1 \ldots Q_n x_n \,\mathrm{CNFC}(\psi)$.

Lema 19.4: resultados auxiliares

1. Dada $\varphi \in E_{\Sigma}^{X}$, a fórmula $\psi = \mathrm{ImplFree}(\varphi)$ é tal que $\psi \in G_{\Sigma}^{X}$ e $\varphi \equiv \psi$.
2. Dada $\varphi \in G_{\Sigma}^{X}$, a fórmula $\psi = \mathrm{NNFC}(\varphi)$ é tal que $\psi \in G_{\Sigma}^{X}$, $\varphi \equiv \psi$ e $\mathrm{FNN}(\psi)$.
3. Dadas $\varphi \in G_{\Sigma}^{X}$ tal que $\mathrm{FNN}(\varphi)$ a fórmula $\psi = \mathrm{Scope}(\varphi)$ é tal que $\psi \in G_{\Sigma}^{X}$, $\varphi \equiv \psi$ e FNP(ψ).

Exemplo

Seja $\varphi = \forall x\, \exists y\, P(x, y, z) \to \exists x\, \forall y\, \neg Q(x, y, z)$

Vai-se calcular $\mathcal{P}(\varphi) = \mathrm{Scope}(\mathrm{NNFC}(\mathrm{ImplFree}(\varphi)))$.

$$
\begin{array}{rcl}
\mathrm{ImplFree}(\varphi) & = & \neg(\mathrm{ImplFree}(\forall x\, \exists y\, P(x, y, z)) \\
 & & \vee\;\; \mathrm{ImplFree}(\exists x\, \forall y\, \neg Q(x, y, z))) \\
 & = & \neg \forall x\, \exists y\, P(x, y, z) \vee \exists x\, \forall y\, \neg Q(x, y, z) = \phi
\end{array}
$$

$$
\begin{array}{rcl}
\mathrm{NNFC}(\phi) & = & \mathrm{NNFC}(\neg \forall x\, \exists y\, P(x, y, z)) \vee \mathrm{NNFC}(\exists x\, \forall y\, \neg Q(x, y, z)) \\
 & = & \exists x\, \mathrm{NNFC}(\neg \exists y\, P(x, y, z)) \vee \exists x\, \forall y\, \neg Q(x, y, z) \\
 & = & \exists x\, \forall y\, \mathrm{NNFC}(\neg P(x, y, z)) \vee \exists x\, \forall y\, \neg Q(x, y, z) \\
 & = & \exists x\, \forall y\, \neg P(x, y, z) \vee \exists x\, \forall y\, \neg Q(x, y, z) = \gamma
\end{array}
$$

Exemplo

Seja $\gamma = \exists x \, \forall y \, \neg P(x, y, z) \vee \exists x \, \forall y \, \neg Q(x, y, z)$

$$
\begin{array}{rcl}
\mathrm{Scope}(\gamma) & = & \exists x_0 \, \forall y_0 \, \mathrm{Scope}(\neg P(x_0, y_0, z) \vee \exists x \, \forall y \, \neg Q(x, y, z)) \\
 & = & \exists x_0 \, \forall y_0 \, \exists x_1 \, \forall y_1 \, \mathrm{Scope}(\neg P(x_0, y_0, z) \vee \neg Q(x_1, y_1, z)) \\
 & = & \exists x_0 \, \forall y_0 \, \exists x_1 \, \forall y_1 \, (\neg P(x_0, y_0, z) \vee \neg Q(x_1, y_1, z))
\end{array}
$$

Exemplo

Seja $\varphi = (P \wedge \exists y\, S(y)) \wedge (\forall x\, Q(x) \vee R)$

$$
\begin{aligned}
\mathrm{Scope}(\varphi) &= \mathrm{Scope}(\mathrm{Scope}(P \wedge \exists y\, S(y)) \wedge \mathrm{Scope}(\forall x\, Q(x) \vee R)) \\
&= \mathrm{Scope}(\exists y_0\, \mathrm{Scope}(P \wedge S(y_0)) \wedge \forall x_0\, \mathrm{Scope}(Q(x_0) \vee R)) \\
&= \mathrm{Scope}(\exists y_0\, (P \wedge S(y_0)) \wedge \forall x_0\, (Q(x_0) \vee R)) \\
&= \exists y_1\, \mathrm{Scope}((P \wedge S(y_1)) \wedge \forall x_0\, (Q(x_0) \vee R)) \\
&= \exists y_1 \forall x_1\, \mathrm{Scope}((P \wedge S(y_1)) \wedge (Q(x_1) \vee R)) \\
&= \exists y_1 \forall x_1\, ((P \wedge S(y_1)) \wedge (Q(x_1) \vee R))
\end{aligned}
$$