Lógica Computacional

Aula Teórica 3: Sintaxe da Lógica Proposicional

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

Menor conjunto gerado por regras

É uma forma "construtiva" (ou incremental) de definir conjuntos *infinitos*:

Como funciona?

- Diz-se primeiro ("axiomatiza-se") quais são os elementos "básicos" do conjunto (em número *finito*).
- Dão-se depois "regras" (em número *finito*) para obter novos elementos a partir dos que já estão no conjunto.

O conjunto resultante contém *todos* os elementos que se podem gerar com as regras, e *apenas* esses.

Definição indutiva: menor conjunto gerado por regras

Notar bem

- Aplicando infinitas vezes as regras, obtém-se um número infinito de elementos (a partir de um número finito de axiomas e regras).
- Cada elemento do conjunto tem no entanto uma justificação, prova ou *derivação finita*: a sequencia das regras aplicadas para o obter.

Exemplos

A linguagem F_P da Lógica Proposicional foi definida desta forma.
Outro exemplo essencial é o dos números naturais.

Definição indutiva dos naturais

Regras

- Fixa-se primeiro que zero é natural (axioma *ZERO*): $0 \in \mathbb{N}_0$
- Diz-se depois como obter novos naturais a partir dos já gerados (regra *SUCC*): se $n \in \mathbb{N}_0$ então $n + 1 \in \mathbb{N}_0$

Justificações de pertença a conjunto

São cadeias (finitas) de aplicações de regras, terminando num axioma, que mostram como se gerou o elemento.

- $3 \in \mathbb{N}_0$, pois $3 = 2 + 1$ (regra *SUCC*), e $2 \in \mathbb{N}_0$, pois $2 = 1 + 1$ (regra *SUCC*), e também $1 \in \mathbb{N}_0$, pois $1 = 0 + 1$ (regra *SUCC*), uma vez que $0 \in \mathbb{N}_0$ (axioma *ZERO*).
- $3,5 \notin \mathbb{N}_0$, pois $3,5 = 2,5 + 1$ (regra *SUCC*), e $2,5 = 1,5 + 1$ (regra *SUCC*), e $1,5 = 0,5 + 1$ (regra *SUCC*), e $0,5 = -0,5 + 1$ (regra *SUCC*), ... a sequência não termina porque não "encontra" o axioma.

Definição indutiva de pilhas de inteiros

Regras

- Fixa-se primeiro que *vazia* é pilha (axioma *VAZIA*):
  $vazia \in PilhaInt$
- Diz-se depois como obter uma nova pilha a partir de um inteiro e de uma pilha já gerada (regra *PUSH*):
  se $i \in \mathbb{Z}$ e $p \in PilhaInt$ então $push(i, p) \in PilhaInt$

Justificações de pertença a conjunto

$push(4, push(-7, push(-1, vazia))) \in PilhaInt$, pois $4 \in \mathbb{Z}$ e
$push(-7, push(-1, vazia)) \in PilhaInt$ (regra *PUSH*),
uma vez que $-7 \in \mathbb{Z}$ e $push(-1, vazia) \in PilhaInt$ (regra *PUSH*),
já que $-1 \in \mathbb{Z}$ e $vazia \in PilhaInt$ (axioma *VAZIA*).

Sintaxe e semântica

Pilhas de inteiros

A definição indutiva introduziu uma (nova) linguagem (formal) para falar de valores abstractos.

Denotação

- Sintaxe: regras para escrever palavras (*termos*) da linguagem.
  Definição indutiva gera linguagem infinita de palavras finitas.
- Semântica: associa sentido (*i.e.*, *valores*) às palavras.
  Função definida (também) indutivamente nas regras da linguagem: a cada termo corresponde um valor - a sua *denotação*.
  Definição indutiva de funções? Sim, uma função é um conjunto de pares ordenados.

Exemplo: o termo $push(-7, push(-1, vazia))$ denota o valor $\begin{array}{|c|} \text{-7} \\ \text{-1} \\ \hline \end{array}$

Definição 3.1: subfórmulas de uma fórmula

O conjunto SBF(φ) das subfórmulas de $\varphi \in F_P$ é definido indutivamente pelas seguintes regras:

- SBF(φ) = $\{\varphi\}$, se φ é atómica;
- seja $\delta \stackrel{\text{def}}{=} \varphi \vee \psi$, $\delta \stackrel{\text{def}}{=} \varphi \wedge \psi$ ou $\delta \stackrel{\text{def}}{=} \varphi \rightarrow \psi$;

  SBF(δ) = $\{\delta\} \cup$ SBF(φ) $\cup$ SBF(ψ).

Definição 3.2: símbolos proposicionais de uma fórmula

O conjunto SMB(φ) $\stackrel{\text{def}}{=}$ SBF(φ) $\cap P$ contém os símbolos proposicionais de uma fórmula $\varphi \in F_P$.

Exemplos

Símbolos proposicionais e subfórmulas de uma fórmula

Sejam $p, q, r \in P$ e $\neg p, (p \vee q) \to r \in F_P$.

- $\mathrm{SMB}(\neg p) = \{p\}$

  $\mathrm{SBF}(\neg p) = \mathrm{SBF}(p \to \bot) = \{\neg p\} \cup \mathrm{SBF}(p) \cup \mathrm{SBF}(\bot) = \{\neg p, p, \bot\}$;
- $\mathrm{SMB}((p \vee q) \to r) = \{p, q, r\}$

  $\mathrm{SBF}((p \vee q) \to r) = \{(p \vee q) \to r\} \cup \mathrm{SBF}(p \vee q) \cup \mathrm{SBF}(r) = \{(p \vee q) \to r\} \cup \{p \vee q\} \cup \mathrm{SBF}(p) \cup \mathrm{SBF}(q) \cup \{r\} = \{(p \vee q) \to r, p \vee q, p, q, r\}$.

Como mostrar que dada sequência de símbolos do alfabeto é fórmula da linguagem?

Uma prova é uma sequência de fórmulas, sendo

- cada uma obtida por aplicação de uma regra, eventualmente usando fórmulas anteriores (na sequência) como hipóteses dessa regra;
- a última fórmula da sequência é a conclusão desejada.

Definição 3.3: prova de fórmula

Uma *prova* para $\varphi \in F_P$ é uma sequência $\varphi_1, \ldots, \varphi_n$ de fórmulas de F_P, com $n \geq 1$, tal que:

- $\varphi = \varphi_n$;
- para qualquer $1 \leq i \leq n$ tem-se que φ_i resulta de aplicar uma das regras da definição de F_P a fórmulas da sequência $\varphi_1, \ldots, \varphi_{n-1}$.

Provas como árvores etiquetadas

A ideia

- Uma prova ou inferência é apresentada em árvore, dita de dedução ou derivação.
- Cada árvore é construida a partir de árvores singulares (ou folhas) utilizando-se as regras da definição indutiva.
- Obtém-se um novo nível da árvore por aplicação de uma regra.
- A etiquetas dos nós são fórmulas.
  - As fórmulas nas folhas resultam de axiomas.
  - A fórmula na raíz é a conclusão da prova. Diz-se que a árvore é uma derivação dessa fórmula.

Exemplo de derivação

Sejam $p, q, r \in P$.
O termo $(p \rightarrow (q \wedge (r \vee p))) \in F_P$ (é fómula).
Prova:

$$\dfrac{\dfrac{}{p}\,(\text{PROP}) \qquad \dfrac{\dfrac{}{q}\,(\text{PROP}) \qquad \dfrac{\dfrac{}{r}\,(\text{PROP}) \qquad \dfrac{}{p}\,(\text{PROP})}{(r \vee p)}\,(\text{DIS})}{(q \wedge (r \vee p))}\,(\text{CON})}{(p \rightarrow (q \wedge (r \vee p)))}\,(\text{IMP})$$

Note que as provas não são (necessariamente) únicas.

O princípio de indução estrutural

Motivação

Uma regra de uma definição indutiva de um conjunto S, introduzindo termos construidos com dado operador n-ário op, tem a forma geral seguinte.

Se para qualquer i tal que $1 \leq i \leq n$, com $n \geq 0$, se tem que
$e_i \in S_i$, então $\mathrm{op}(e_1, \ldots, e_n) \in S$,
sendo cada S_i ou o conjunto S ou outro conjunto já previamente definido.

Nota

- Tal como se faz indução sobre os naturais, pode-se fazer indução sobre (os construtores de) qualquer conjunto definido indutivamente.
- Na verdade. a indução natural é um caso particular desta indução *estrutural*.

O princípio de indução estrutural

Definição 3.4

Seja S um conjunto definido indutivamente e P um predicado sobre elementos de S.

Se para cada construtor (k-ário) op da definição indutiva de S, se tem $P(\mathrm{op}(e_1, \ldots, e_k))$, se $P(e_i)$, para cada $e_i \in S$ (com $1 \leq i \leq k$), então tem-se $P(e)$ para qualquer $e \in S$.

Este princípio de indução, por sua vez, é um caso particular da chamada *indução generalisada* ou Noetheriana.

Definição 3.5

Seja $(S, \leq)$ um conjunto parcialmente ordenado bem fundado e P um predicado sobre elementos de S.

$$\forall r \in S\ (\forall s \in S\ (s < r \rightarrow P(s)) \rightarrow P(r)) \rightarrow \forall t \in s.P(t)$$

Provas por indução estrutural: um exemplo

Relembre a Definição 3.2 de conjunto de símbolos proposicionais de uma fórmula.

Considere o conjunto F_P das fórmulas da Lógica Proposicional e a propriedade $P(\varphi) : \mathsf{SMB}(\varphi)$ *é finito*.

Prova-se por indução estrutural que P se verifica para todos os elementos de F_P.

- Casos Base.
  - $\varphi = p$, para $p \in P$. Pela definição, $\mathsf{SMB}(p) = \{p\}$, que é finito.
  - $\varphi = \perp$. Pela definição, $\mathsf{SMB}(\perp) = \emptyset$, que é finito.
- Passo (faz-se apenas para $\varphi = \psi_1 \wedge \psi_2$; os restantes casos são semelhantes).
  Pela definição, $\mathsf{SMB}(\varphi) = \mathsf{SMB}(\psi_1) \cup \mathsf{SMB}(\psi_2)$, que é finito pois por hipótese de indução $\mathsf{SMB}(\psi_1)$ e $\mathsf{SMB}(\psi_2)$ são finitos, e a união de conjuntos finitos é finita.

Equivalência entre a definição indutiva e a existência de prova

Proposição 3.1

O termo $\varphi \in F_P$, se e só se, existe uma prova para φ.

Prova do sentido "só se": por indução estrutural

Hipótese: termo $\varphi \in F_P$. Tese: existe uma prova para φ.

Base: $\varphi = \perp$ ou $\varphi = p$ para $p \in P$. Por *BOT* ou por *PROP*, a sequência φ é uma prova.

Passo: seja $\varphi = \psi_1 \vee \psi_2$ (os restantes casos têm prova semelhante). Por hipótese de indução existem sequências $\psi_{11} \cdots \psi_{1n}$ e $\psi_{21} \cdots \psi_{2m}$ (com $n, m \geq 1$), que provam respectivamente ψ_1 e ψ_2. Logo, como $\psi_1 = \psi_{1n}$ e $\psi_2 = \psi_{2m}$, por *DIS* a sequência $\psi_{11} \cdots \psi_{1n}\psi_{21} \cdots \psi_{2m}\varphi$ prova φ.

Equivalência entre a definição indutiva e a existência de prova

Prova do sentido "se": por indução no comprimento da prova

Hipótese: existe uma prova para φ. Tese: termo $\varphi \in F_P$.

<u>Base</u>: a prova é a sequência σ com comprimento 1. Então, a sequência tem apenas uma fórmula ($\sigma = \varphi$), e logo, $\varphi = \bot$ ou $\varphi = p$ para $p \in P$. Por *BOT* ou por *PROP* (respectivamente), a fórmula $\varphi \in F_P$.

<u>Passo</u>: A prova é uma sequência σ de comprimento $1 + n$, sendo $\varphi = \psi_1 \vee \psi_2$ a última fórmula da sequência (os restantes casos têm prova semelhante). Então, a sequência tem duas subsequências σ_1 e σ_2, de comprimentos n_1 e n_2 com $n = n_1 + n_2$, sendo a primeira subsequência a prova de ψ_1 e a segunda a prova de ψ_2.
Por hipótese de indução $\psi_1 \in F_P$ e $\psi_2 \in F_P$. Logo, por *DIS*, $\varphi \in F_P$.