Lógica Computacional

DCC/FCUP

2020/21

Sistemas dedutivos de Hilbert, *H*

Usados incialmente nas tentativas de mecanização das demonstrações matemáticas (Sec. XIX e início de XX) (também por G. Frege e B. Russel)

Supondo apenas o conjunto completo de conectivas $\{\neg, \rightarrow\}$, toma a forma:

Axiomas

- $\phi \rightarrow (\psi \rightarrow \phi)$
- $(\phi \rightarrow (\psi \rightarrow \theta)) \rightarrow ((\phi \rightarrow \psi) \rightarrow (\phi \rightarrow \theta))$
- $(\neg\psi \rightarrow \neg\phi) \rightarrow ((\neg\psi \rightarrow \phi) \rightarrow \psi)$

Regras de inferência

- *Modus ponens*: de ϕ e de $\phi \rightarrow \psi$, inferir ψ

Proposição: (da dedução) Se $\Sigma \cup \{\psi\} \vdash_H \theta$ então $\Sigma \vdash_H \psi \to \theta$.
Proposição: $\Sigma \vdash_{DN} \phi$ se e só se $\Sigma \vdash_H \phi$

Demonstração.

($\Leftarrow$): Basta ver que os axiomas de H são teoremas de DN. A regra de inferência corresponde à regra da eliminação de implicação de DN ($\to$ **E**)
($\Rightarrow$) : é possível transformar uma dedução em DN, numa dedução em H. (Não o faremos neste curso...) $\square$

Corolário: O sistema de dedução H é integro e completo para a lógica proposicional.

Sistemas dedutivos automatizáveis

Mesmo no sistema *DN* não é fácil construir um algoritmo para determinar se uma uma fórmula é um teorema, excepto se se usar a completude (e construir a dedução a partir da tabela de verdade). Mas existem outros sistemas de dedução automatizáveis e em que para determinar se uma fórmula ϕ é um teorema não é necessária a semântica.

A propriedade essencial

destes sistemas é que, partindo da fórmula que se pretende deduzir (i.e considerando a dedução da conclusão para as premissas), em cada passo duma dedução, as fórmulas são sub-fórmulas de alguma fórmula de um passo "anterior".

É fácil ver que a regra de *Modus ponens* não verifica esta propriedade!

Alguns desses sistemas

Sequents de Gentzen "Variante" do sistema de dedução natural.

Tableaux deduzir que a fórmula $\neg\phi$ é uma contradição

Resolução também por contradição, mas necessita de $\neg\phi$ em forma normal conjuntiva....cláusulas

Notação uniforme de Smullyan

Permite agrupar as fórmulas da forma $\phi \circ \psi$ e $\neg(\phi \circ \psi)$ em duas categorias:

conjuntivas α, a que se associam duas componentes α_1 e α_2
disjuntivas β, a que se associam duas componentes β_1 ou β_2

α	α_1	α_2	β	β_1	β_2
$\phi \wedge \psi$	ϕ	ψ	$\neg(\phi \wedge \psi)$	$\neg\phi$	$\neg\psi$
$\neg(\phi \vee \psi)$	$\neg\phi$	$\neg\psi$	$\phi \vee \psi$	ϕ	ψ
$\neg(\phi \rightarrow \psi)$	ϕ	$\neg\psi$	$\phi \rightarrow \psi$	$\neg\phi$	ψ

Para toda a atribuição de valores às variáveis v, e todas as fórmulas α e β:

$$\begin{aligned} v(\alpha) &= v(\alpha_1) \wedge v(\alpha_2) \\ v(\beta) &= v(\beta_1) \vee v(\beta_2). \end{aligned}$$

Tableaux semânticos

Para deduzir uma fórmula ϕ, inicia-se com $\neg\phi$ e produz-se uma contradição. Uma dedução é uma árvore em que os nós são etiquetados por fórmulas, sendo a raíz da árvore etiquetada por $\neg\phi$. Em cada passo, expande-se uma folha da árvore de acordo com as regras de expansão dos *Tableaux*:

$$\frac{\neg\neg\phi}{\phi} \qquad \frac{\neg\bot}{\top} \qquad \frac{\neg\top}{\bot} \qquad \frac{\alpha}{\begin{array}{c}\alpha_1\\ \alpha_2\end{array}} \qquad \frac{\beta}{\beta_1 \mid \beta_2}$$

Tableaux semânticos

Seja *r* um ramo da árvore etiquetado com ψ.

- Se ψ é $\neg\neg\phi$, expandimos a folha desse ramo com mais um nó etiquetado por ϕ.
- Analogamente, se ψ é $\neg\bot$, adiciona-se $\top$ ou se é $\neg\top$, adiciona-se $\bot$.
- Se ψ é um α, adicionam-se a *r* dois nós, um etiquetado com α_1 e outro com α_2 (filho de α_1).
- Se ψ é um β, adiciona-se a *r* dois nós filhos, um etiquetado por β_1 outro por β_2.

Tableau

Um *tableau* **T** pode ser definido indutivamente por:
Seja $\{\phi_1, \ldots, \phi_n\}$ um conjunto de fórmulas.

1. Um *tableau* para $\{\phi_1, \ldots, \phi_n\}$ é uma árvore de um só ramo:

$$\begin{array}{c} \phi_1 \\ \phi_2 \\ \vdots \\ \phi_n \end{array}$$

2. Se **T** é um *tableau* para $\{\phi_1, \ldots, \phi_n\}$ e $\mathbf{T}^\star$ resulta de **T** por aplicação duma regra de expansão de *tableaux* então $\mathbf{T}^\star$ é um *tableau* para $\{\phi_1, \ldots, \phi_n\}$.

Exemplo

Um *tableau* para o conjunto $\{p \wedge (\neg q \vee \neg p)\}$ é:

$$p \wedge (\neg q \vee \neg p)$$
$$p$$
$$\neg q \vee \neg p$$
$$\neg q \qquad\qquad\qquad \neg p$$

Satisfabiblidade e dedução em *Tableaux*

Um ramo r de um *tableau* diz-se fechado se existe uma fórmula ϕ tal que ϕ e $\neg\phi$ ocorrem em r.

Um *tableau* diz-se fechado se todos os seus ramos estão fechados.

O *tableau* do exemplo anterior tem, um ramo fechado mas não é fechado.

Satisfabiblidade e dedução em *Tableaux*

Uma dedução por *tableau* de ϕ é um *tableau* fechado para $\{\neg\phi\}$.

A fórmula ϕ é um teorema se ϕ tem uma dedução por *tableau*.

Um ramo *r* de um *tableau* é satisfazível se o conjunto de fórmulas que etiquetam os seus nós é satisfazível.

Um *tableau* **T** é satisfazível se pelo menos um dos seus ramos é satisfazível.

Integridade e Completude

Proposição: Pela aplicação de qualquer regra de expansão de *tableau* a um *tableau* satisfazível, obtêm-se um *tableau* satisfazível.

Proposição: Se existe um *tableau* fechado para um conjunto de fórmulas Γ, então Γ não é satisfazível.

Integridade e Completude

Teorema: Se ϕ tem uma dedução por *tableau*, então ϕ é uma tautologia.

Demonstração.

Uma dedução por *tableau* é um *tableau* fechado para $\{\neg\phi\}$. Então, pela Proposição **??**, $\{\neg\phi\}$ não é satisfazível, logo ϕ é uma tautologia. □

Também se mostra que:

Teorema: Se ϕ é uma tautologia, ϕ tem uma dedução por *tableau*.

Exemplos

Constrói deduções de *tableaux* para as seguintes fórmulas:

a) $\neg((p \vee q) \wedge (\neg p \wedge \neg q))$

b) $((p \rightarrow q) \wedge (q \rightarrow r)) \rightarrow \neg(\neg r \wedge p)$

Sistema dedutivo por Resolução

É um sistema dedutivo por refutação: para deduzir ϕ, deduz-se que $\neg\phi$ é uma contradição.
Seja Σ um conjunto de cláusulas (disjunção de literais) e representamos por $\perp$ a cláusula vazia.
O sistema dedutivo por *resolução* não tem axiomas e apenas uma regra de inferência:

Regra de inferência da Resolução

$$\frac{C \cup \{p\} \qquad C' \cup \{\neg p\}}{C \cup C'}$$

A conclusão $C \cup C'$ diz-se a resolvente das premissas.
Uma dedução de $\perp$ a partir de um conjunto $\mathcal{C}$ de cláusulas diz-se uma refutação $\mathcal{C}$.
Uma dedução por resolução de uma fórmula ϕ é uma refutação de cláusulas que correspondem à FNC de $\neg\phi$.

Dado

$$\{\{\neg p\}, \{q, r\}, \{\neg r, \neg s\}, \{p, s\}, \{\neg q, \neg s\}\}$$

temos

$$\dfrac{\dfrac{\{p,s\}\ \{\neg p\}}{\{s\}} \quad \dfrac{\dfrac{\{q,r\}\ \{\neg r,\neg s\}}{\{q,\neg s\}}\ \{\neg q,\neg s\}}{\{\neg s\}}}{\bot}$$

Integridade e Completude da Resolução

Teorema: (Integridade)
Seja $\mathcal{C} = \{C_1, \ldots, C_n\}$ um conjunto não vazio de cláusulas da lógica proposicional.

i. Sejam C_i e C_j cláusulas de $\mathcal{C}$ e R uma resolvente de C_i e C_j. Então $\mathcal{C} \models R$.

ii. Se $\mathcal{C} \vdash_R \bot$, isto é, existe uma dedução de $\bot$ a partir de $\mathcal{C}$ usando apenas a regra da Resolução, então $\mathcal{C}$ não é satisfazível.

Teorema: (Completude) Se um conjunto de cláusulas $\mathcal{C}$ é não satisfazível então existe uma dedução $\mathcal{C} \vdash_R \bot$.

Exercícios

Constrói deduções de *tableau* e por **resolução** para as seguintes fórmulas:

a) $(p \to q) \to ((\neg p \to q) \to q)$

b) $q \to (p \to (q \wedge p))$

c) $q \to (\neg p \to \neg(q \to p))$