Dedução Natural na LPC

Considere mais uma vez o problema apresentado no capítulo anterior:

Considere uma ilha onde moram apenas dois tipos de pessoas: as honestas, e que portanto sempre falam a verdade; e as desonestas, que sempre mentem. Um viajante, ao passar por esta ilha encontra três moradores chamados A, B e C. O viajante pergunta para o morador A: "Você é honesto ou desonesto?" A responde algo incompreensível, e o viajante pergunta para B: "O que ele disse?" B então responde "Ele disse que é desonesto". Neste momento C se manifesta: "Não acredito nisto! Isto é uma mentira!". Questão: C é honesto ou desonesto?

Concluímos, tanto por meio de uma análise informal, quanto por meio das tabelas de verdade que C é honesto. Neste capítulo estudaremos um sistema de provas que nos permitirá construir uma prova direta deste fato. O sistema que utilizaremos é conhecido como *Dedução Natural* (DN), e foi criado pelo lógico alemão Gerhard Gentzen (1909-1945). Este sistema consiste em um conjunto de regras de inferência que tenta capturar o raciocínio matemático da forma mais *natural* possível. Neste contexto, o primeiro conceito importante que aparece é o de *sequente*. Formalmente, um sequente é um par cujo primeiro elemento é um conjunto finito de fórmulas (hipóteses), e o segundo elemento é uma fórmula (conclusão). Assim, se $\varphi_1, \varphi_2, \ldots, \varphi_n$ são as hipóteses de um dado problema, e ψ é a conclusão, escrevemos $\varphi_1\varphi_2\ldots\varphi_n \vdash \psi$ para representar o fato que ψ pode ser provado a partir das hipóteses $\varphi_1, \varphi_2, \ldots, \varphi_n$.

Para exemplificar, considere a seguinte variante do problema acima:

Considere uma ilha onde moram apenas dois tipos de pessoas: as honestas, e que portanto sempre falam a verdade; e as desonestas, que sempre mentem. Um viajante, ao passar por esta ilha encontra três moradores chamados A, B e C. O viajante pergunta para o morador A: "Você é honesto ou desonesto?" A responde algo incompreensível, e o viajante pergunta para B: "O que ele disse?" B então responde "Ele disse que é desonesto". Neste momento C se manifesta: "Não acredito nisto! Isto é uma mentira!". **Prove** que C é honesto.

Agora queremos **provar** que C é honesto, e considerando as representações do que disse cada um dos moradores feitas no capítulo anterior, podemos codificar este problema por meio do sequente

$$p_B \leftrightarrow (p_A \leftrightarrow (\neg p_A)), p_C \leftrightarrow (\neg p_B) \vdash p_C \tag{3}$$

Em DN, uma prova consiste em uma árvore que tem as hipóteses do sequente como folhas, e a conclusão como raiz. Os nós internos destas árvores correspondem a aplicações de regras do sistema DN como veremos a seguir. As regras do sistema DN são regras de inferência com a seguinte estrutura:

$$\frac{\varphi_1, \varphi_2, \ldots, \varphi_n}{\psi}\ (\mathrm{R})$$

onde $\varphi_1, \varphi_2, \ldots, \varphi_n$ e ψ são fórmulas da LPC. A ideia desta regra é que a partir de provas de cada uma das fórmulas $\varphi_1, \varphi_2, \ldots, \varphi_n$ geramos uma nova árvore que corresponde a uma prova de ψ.

Antes de construirmos uma prova para o sequente (3), precisamos apresentar as regras do sistema DN. Estas regras expressam o comportamento dos conectivos lógicos, e são divididas em dois conjuntos: **regras de introdução** e **regras de eliminação**. As regras de introdução podem ser vistas como as regras que definem os conectivos, enquanto que as regras de eliminação nos permitem construir provas a partir de fórmulas que possuem o conectivo em questão. A seguir, apresentaremos as regras de introdução e eliminação para cada um dos conectivos da LP. Iniciaremos com a regra de introdução da conjunção, que diz que para construirmos uma prova de $\varphi \wedge \psi$ precisamos de uma prova de φ, e de uma prova de ψ. Este raciocínio é representado pela regra $(\wedge_i)$ dada a seguir:

$$\frac{\varphi \qquad \psi}{\varphi \wedge \psi}\ (\wedge_i)$$

Uma analogia que pode ser útil para entender como é feita a combinação das regras do sistema DN para construir uma prova, é visualizar cada regra como uma peça de um quebra-cabeça. Assim, as provas de φ e de ψ seriam vistas como partes já montadas deste quebra-cabeça. Ou seja, suponha que tenhamos uma prova de φ:

$$\begin{array}{c} \nabla \\ \varphi \end{array}$$

onde o símbolo ∇ acima corresponde a uma árvore que possui φ como raiz. Similarmente, uma prova de ψ teria a forma:

$$\begin{array}{c} \nabla \\ \psi \end{array}$$

Uma prova da conjunção $\varphi \wedge \psi$ consistiria então na junção das provas de φ e de ψ por meio da regra $(\wedge_i)$:

$$\frac{\begin{array}{c} \nabla \\ \varphi \end{array} \qquad \begin{array}{c} \nabla \\ \psi \end{array}}{\varphi \wedge \psi}\ (\wedge_i) \tag{4}$$

Um aspecto importante que deve ser observado em t

Existem duas regras de eliminação para a conjunção porque podemos extrair qualquer das componentes de uma conjunção:

$$\frac{\varphi \wedge \psi}{\varphi}\ (\wedge_{e_1}) \qquad\qquad \frac{\varphi \wedge \psi}{\psi}\ (\wedge_{e_2})$$

Estas duas regras podem ser representadas de forma mais concisa da seguinte forma:

$$\frac{\varphi_1 \wedge \varphi_2}{\varphi_{i \in \{1,2\}}}\ (\wedge_e) \tag{5}$$

Usaremos então o nome $(\wedge_e)$ para designar a utilização de qualquer uma das regras $(\wedge_{e_1})$ ou $(\wedge_{e_2})$. Observe que a ideia destas regras é que elas possam se encaixar em diversos contextos diferentes. Assim, o importante é conseguir identificar quais as regras podem ser aplicadas em uma determinada situação concreta. O sistema DN é usado para construir provas, e uma prova pode ser vista como uma árvore cujas folhas correspondem às informações dadas, e com o fato que queremos provar como raiz. Por exemplo, suponha que queiramos provar que a conjunção é comutativa, isto é, se A e B são fórmulas da LP, e temos $A \wedge B$ então queremos concluir que $B \wedge A$. Assim, a nossa árvore terá a fórmula $A \wedge B$ em suas folhas, e a fórmula $B \wedge A$ como raiz. Vamos construir esta prova passo a passo. Construiremos esta prova (em forma de árvore) de baixo para cima, mas poderia ser de cima para baixo também. A conclusão a que queremos chegar é $B \wedge A$:

$$\frac{?}{B \wedge A}$$

Considerando as únicas regras que temos até o momento, a saber (4) e (5), fica claro que nossa única opção é a regra $(\wedge_i)$ onde B vai corresponder a fórmula φ, e A, vai corresponder à ψ. Isto nos permite subir um nível na construção da árvore:

$$\dfrac{\dfrac{?}{B} \qquad \dfrac{?}{A}}{B \wedge A}\ (\wedge_i)$$

Agora precisamos por um lado construir uma prova de B, e por outro, uma prova de A. Considerando que a fórmula $A \wedge B$ é dada, é fácil ver que uma prova de B pode ser obtida por meio da regra $(\wedge_e)$:

$$\dfrac{\dfrac{A \wedge B}{B}\ (\wedge_e) \quad \dfrac{?}{A}}{B \wedge A}\ (\wedge_i)$$

Por fim, a prova de A que está faltando para completar a prova também pode ser obtida por meio de uma aplicação da regra $(\wedge_e)$:

$$\dfrac{\dfrac{A \wedge B}{B}\ (\wedge_e) \quad \dfrac{A \wedge B}{A}\ (\wedge_e)}{B \wedge A}\ (\wedge_i)$$

Agora a prova/árvore está completa porque as folhas da árvore correspondem às informações dadas, neste caso, à fórmula $A \wedge B$; e a raiz da árvore corresponde à fórmula que queríamos provar. A construção acima foi fácil de ser feita pelos seguintes motivos:

1. Temos, até o momento, apenas duas regras para utilizar, e portanto não foi complicado fazer a escolha de qual regra utilizar em cada passo;

2. Cada escolha se deu de forma única, ou seja, cada passo correspondeu a um processo determinístico.

Voltando à analogia com o quebra-cabeça, se temos apenas dois tipos/modelos de peças para usar fica mais fácil de saber qual pode ou não ser usada em cada momento. Por outro lado, se temos diversas peças com formas distintas, o processo de escolha se torna mais complicado. Principalmente se duas (ou mais) peças distintas podem ser encaixadas naquele momento da montagem. Isto é exatamente o que vai ocorrer com o nosso sistema de provas. Teremos várias regras para utilizar, e em um dado momento da prova teremos que escolher entre diversas aplicações possíveis de regras distintas, e é justamente esta escolha que vai nos permitir concluir uma prova, ou então entrar em um beco sem saída... Sim, não é qualquer escolha que vai nos permitir concluir uma prova! Mas se, dentre as escolhas possíveis, não é qualquer uma que vai nos permitir concluir a prova, então qual escolher? A resposta é depende... não existe receita para a construção de provas. De fato, este processo é indecidível! Então aprenderemos técnicas para construir provas, e como veremos, não é incomum existirem mais de uma prova para o mesmo fato, ou seja, é possível que sequentes (que nada mais são do que teoremas ou lemas) possam ser provados de formas distintas.

Vejamos agora as regras da disjunção. A regra de introdução da disjunção consiste em, a partir de uma prova de φ_1, construir uma prova de $\varphi_1 \vee \varphi_2$:

$$\frac{\varphi_1}{\varphi_1 \vee \varphi_2}\ (\vee_{i_1})$$

Analogamente, poderíamos a partir de uma prova de φ_2, construir uma prova de $\varphi_1 \vee \varphi_2$:

$$\frac{\varphi_2}{\varphi_1 \vee \varphi_2}\ (\vee_{i_2})$$

Estas regras nos dizem que para construirmos a prova de uma disjunção da forma $\varphi_1 \vee \varphi_2$, precisamos ter uma prova de φ_1, ou então uma prova de φ_2, e portanto podemos representar estas duas regras por meio de uma regra genérica (assim como fizemos para a eliminação da conjunção):

$$\frac{\varphi_{i \in \{1,2\}}}{\varphi_1 \vee \varphi_2}\ (\vee_i)$$

A regra de eliminação da disjunção apresenta uma característica fundamental do sistema DN: ele é baseado na noção de "suposição (*assumption*)"que consiste em uma fórmula que pode ser livremente introduzida em uma prova, mas que precisa ser descartada para que a prova seja concluída. Note que uma suposição não é uma hipótese do problema original, mas uma fórmula extra que precisa ser descartada segundo condições específicas como veremos. A estrutura da regra de eliminação da disjunção é dada a seguir:

$$\frac{\varphi \vee \psi \qquad \begin{array}{c}[\varphi]^u \\ \vdots \\ \gamma\end{array} \qquad \begin{array}{c}[\psi]^v \\ \vdots \\ \gamma\end{array}}{\gamma}\ (\vee_e)\ u, v$$

Esta regra pressupõe três provas: uma prova da disjunção $\varphi \vee \psi$, uma prova de γ a partir da suposição φ e uma outra prova de γ a partir da suposição ψ.

A regra de introdução da implicação também utiliza a noção de suposição: para provarmos uma implicação da forma $\varphi \to \psi$, inicialmente assumimos φ (suposição), e a partir disto construímos uma prova de ψ conforme o esquema abaixo.

$$
\dfrac{\begin{array}{c}[\varphi]^u\\ \vdots\\ \psi\end{array}}{\varphi \to \psi}\,(\to_i)u
$$

A regra de eliminação da implicação é bem conhecida e dispensa comentários:

$$
\dfrac{\varphi \to \psi \qquad \varphi}{\psi}\,(\to_e)
$$

As regras da negação são definidas de forma similar às da implicação:

$$
\dfrac{\begin{array}{c}[\varphi]^u\\ \vdots\\ \bot\end{array}}{\neg\varphi}\,(\neg_i)\ u
\qquad\qquad\qquad
\dfrac{\neg\varphi \qquad \varphi}{\bot}\,(\neg_e)
$$

Por fim, a regra *PBC* (*proof by contradiction*) nos permite construir a prova de uma fórmula φ assumindo a sua negação e chegando a uma contradição:

$$
\dfrac{\begin{array}{c}[\varphi]^u\\ \vdots\\ \varphi\end{array}}{\varphi \to \bot}\,(PBC)\ u
$$

Todas as regras apresentadas estão resumidas na Tabela 1

Considere a árvore de dedução abaixo:

$$
\dfrac{\neg(\neg\phi \wedge \neg\psi) \qquad \dfrac{(\neg_i)\ u\ \dfrac{(\neg_e)\ \dfrac{[\neg(\phi\vee\psi)]^w \qquad (\vee_i)\,\dfrac{[\phi]^u}{\phi\vee\psi}}{\bot}}{\neg\phi} \qquad \dfrac{\dfrac{[\neg(\phi\vee\psi)]^w \qquad \dfrac{[\psi]^v}{\phi\vee\psi}\,(\vee_i)}{\bot}\,(\neg_e)}{\neg\psi}\,(\neg_i)\ v}{\neg\phi \wedge \neg\psi}\,(\wedge_i)}{\dfrac{\bot}{\phi \vee \psi}\,(PBC)\ w}\,(\neg_e)
$$

Analise a aplicação de cada uma das regras, e responda: esta prova está correta? Justifique sua resposta.

E a árvore de dedução abaixo está correta? Justifique sua resposta. (Lembre-se que "$a \leftrightarrow b$" abrevia "$(a \to b) \wedge (b \to a)$".)

$$
\dfrac{\dfrac{\dfrac{[(\neg\phi \to \psi) \to \neg\phi]^x \qquad [\neg\phi \to \psi]^y}{\neg\phi}\,(\to_e)}{((\neg\phi \to \psi) \to \neg\phi) \to \neg\phi}\,(\to_i)x \qquad \dfrac{\dfrac{[\neg\phi]^z}{(\neg\phi \to \psi) \to \neg\phi}\,(\to_i)y}{\neg\phi \to ((\neg\phi \to \psi) \to \neg\phi)}\,(\to_i)z}{((\neg\phi \to \psi) \to \neg\phi) \leftrightarrow \neg\phi}\,(\wedge_i)
$$

Tabela 1: Regras para DN na LPC

regras de introdução	regras de eliminação
$\dfrac{\varphi \quad \psi}{\varphi \wedge \psi}\ (\wedge_i)$	$\dfrac{\varphi_1 \wedge \varphi_2}{\varphi_{i \in \{1,2\}}}\ (\wedge_e)$
$\dfrac{\varphi_{i \in \{1,2\}}}{\varphi_1 \vee \varphi_2}\ (\vee_i)$	$\dfrac{\varphi \vee \psi \qquad \begin{array}{c}[\varphi]^u \\ \vdots \\ \chi\end{array} \qquad \begin{array}{c}[\psi]^v \\ \vdots \\ \chi\end{array}}{\chi}\ (\vee_e)\ u, v$
$\dfrac{\begin{array}{c}[\varphi]^u \\ \vdots \\ \psi\end{array}}{\varphi \to \psi}\ (\to_i) u$	$\dfrac{\varphi \quad \varphi \to \psi}{\psi}\ (\to_e)$
$\dfrac{\begin{array}{c}[\varphi]^u \\ \vdots \\ \bot\end{array}}{\neg\varphi}\ (\neg_i)\ u$	$\dfrac{\varphi \quad \neg\varphi}{\bot}\ (\neg_e)$
	$\dfrac{\begin{array}{c}[\neg\varphi]^u \\ \vdots \\ \bot\end{array}}{\varphi}\ (\text{PBC})\ \textsc{u}$

Caso alguma das provas acima esteja correta, escreva o sequente que corresponde ao que foi provado.