Capítulo 9: Inferência em Lógica de Primeira Ordem

Regras de Inferência

Exemplos para parte proposicional:

Modus Ponens (MP):

$$\frac{\alpha \ \ \alpha \Rightarrow \beta}{\beta}$$

$$\frac{\mathrm{em}(\mathrm{João}, \mathrm{DCC}) \quad \mathrm{em}(\mathrm{João}, \mathrm{DCC}) \Rightarrow \mathrm{OK}(\mathrm{João})}{\mathrm{OK}(\mathrm{João})}$$

Introdução do "e" (IE):

$$\frac{\alpha \quad \beta}{\alpha \wedge \beta}$$

$$\frac{\mathrm{OK}(\mathrm{João}) \quad \mathrm{estuda}(\mathrm{João}, \mathrm{CC})}{\mathrm{OK}(\mathrm{João}) \wedge \mathrm{estuda}(\mathrm{João}, \mathrm{CC})}$$

Eliminação do "e" (EE):

$$\frac{\alpha \wedge \beta}{\alpha} \qquad \frac{\alpha \wedge \beta}{\beta}$$

Etc...

Envolvendo quantificadores:

Eliminação do universal (EUn):

$$\frac{\forall x\, \alpha}{\alpha\{x/t\}}$$

onde t é um termo sem variáveis.

$$\frac{\forall x \text{ em}(x, \text{DCC}) \Rightarrow \text{OK}(x)}{\text{em}(\text{João}, \text{DCC}) \Rightarrow \text{OK}(\text{João})}$$

Eliminação do existencial (EEx):

$$\frac{\exists x\, \alpha}{\alpha\{x/k\}}$$

onde t é uma constante **nova**.

$$\frac{\exists x \text{ estuda}(x, \text{CC}) \wedge \text{feliz}(x)}{\text{estuda}(\text{A}, \text{CC}) \wedge \text{feliz}(\text{A})}$$

Introdução do existencial (IEx):

$$\frac{\alpha}{\exists x\, \alpha\{t/x\}}$$

onde t é um termo sem variáveis.

$$\frac{\text{feliz}(\text{João})}{\exists x \text{ feliz}(x)}$$

Um exemplo de Prova

1. *Bob é um boi.*
2. *Pat é um porco.*
3. *Bois são mais velozes que porcos.*

   De 1, 2 e 3: Bob é mais veloz que Pat.

1. boi(Bob)
2. porco(Pat)
3. $\forall x, y[\text{boi}(x) \wedge \text{porco}(y) \Rightarrow \text{m-veloz}(x, y)]$
4. boi(Bob) $\wedge$ porco(Pat) por IE (1, 2)
5. boi(Bob)$\wedge$porco(Pat) $\Rightarrow$ m-veloz(Bob, Pat) por EUn (3)
6. m-veloz(Bob, Pat) por MP (4, 5)

Busca com regras de inferência primitivas:

- **Estados:** conjuntos de sentenças.
- **Operadores:** regras de inferência.
- **Teste de meta:** estado contém sentença relativa à consulta.

⇒ **Problema:** fator de ramificação enorme.

Padrão inferencial do exemplo anterior:

- IE, EUn, MP

⇒ bastante comum!

De

boi(Bob),

porco(Pat), e

$\forall x, y[\text{boi}(x) \land \text{porco}(y) \Rightarrow \text{m-veloz}(x, y)]$

deduzir:

$\text{m-veloz}(\text{Bob}, \text{Pat})$

por MP **generalizada**.

Unificação:

σ **unifica** sentenças atômicas p e q se $p\sigma = q\sigma$.

Exemplos:

- $p = \text{conhece}(\text{João}, x)$ e $q_1 = \text{conhece}(\text{João}, \text{Joana})$
- $p = \text{conhece}(\text{João}, x)$ e $q_2 = \text{conhece}(y, \text{Jorge})$
- $p = \text{conhece}(\text{João}, x)$ e $q_3 = \text{conhece}(y, \text{mãe}(y))$
- $p = \text{conhece}(\text{João}, x)$ e $q_4 = \text{conhece}(x, \text{Joana})$

Idéia:

1. Unificar premissas de uma regra com fatos.
2. Aplicar unificador à conclusão da regra.

Exemplo: sabendo-se q_1, q_2 e q_3 e

$$\text{conhece}(\text{João}, x) \Rightarrow \text{gosta}(\text{João}, x)$$

conclui-se:

- $\text{gosta}(\text{João}, \text{Joana})$
- $\text{gosta}(\text{João}, \text{Jorge})$
- $\text{gosta}(\text{João}, \text{mãe}(\text{João}))$

Modus Ponens Generalizado (MPG):

$$\frac{p_1' \; p_2' \; \ldots \; p_n' \; (p_1 \wedge p_2 \wedge \ldots \wedge p_n \Rightarrow q)}{q\sigma} \quad \text{onde } p_i'\sigma = p_i\sigma.$$

Exemplo:

De:

$$p_1' = \text{m-veloz}(\text{Bob}, \text{Pat})$$

$$p_2' = \text{m-veloz}(\text{Pat}, \text{Bill})$$

$$p_1 \wedge p_2 \Rightarrow q$$
$$= \text{m-veloz}(x, y) \wedge \text{m-veloz}(y, z) \Rightarrow \text{m-veloz}(x, z)$$

conclui-se ($\sigma = \{x/\text{Bob}, y/\text{Bill}\}$):

$$q\sigma = \text{m-veloz}(\text{Bob}, \text{Bill})$$

MPG é usada com BC que tem as características:

- Cada sentença é uma **cláusula definida**.
- Todas as variáveis são **universalmente quantificadas**.

Dois tipos de **algoritmos de inferência**:

- *Forward chaining*
- *Backward chaining*

Forward chaining

Disparado quando um novo fato p é adicionado à BC:

> Para cada regra tal que p unifica com uma de suas premissas,
>   se as outras premissas são <u>conhecidas</u>
>   então acrescente a conclusão à BC
>   e continue ...

- *Forward chaining* é dirigido por <u>dados</u>: infere propriedades e categorias a partir dos estímulos.
- Cada vez que a BC recebe uma sentença (via TELL), é disparado.

Exemplo de Forward chaining:

a) BC $\Leftarrow$ [1] $\underline{\text{boi}(x) \wedge \text{porco}(y) \Rightarrow \text{m-veloz}(x, y)}$.

b) BC $\Leftarrow$ [2] $\underline{\text{porco}(y) \wedge \text{lesma}(z) \Rightarrow \text{m-veloz}(y, z)}$.

c) BC $\Leftarrow$ [3] $\underline{\text{m-veloz}(x, y) \wedge \text{m-veloz}(y, z) \Rightarrow \text{m-veloz}(x, z)}$.

d) BC $\Leftarrow$ [4] $\underline{\text{boi(Bob)}}$;

un 4 com [1,a]: $\text{porco}(y) \Rightarrow \text{m-veloz}(\text{Bob}, y)$.

e) BC $\Leftarrow$ [5] $\underline{\text{porco(Pat)}}$;

un 5 com [1,b]: $\text{boi}(x) \Rightarrow \text{m-veloz}(x, \text{Pat})$,

un com [4]: $\text{m-veloz}(\text{Bob}, \text{Pat})$;

BC $\Leftarrow$ [6] $\underline{\text{m-veloz}(\text{Bob}, \text{Pat})}$;

un 6 com [3,a], ... un 6 com [3,b], ...

un 5 com [2,a]: $\text{lesma}(z) \Rightarrow \text{m-veloz}(\text{Pat}, z)$.

f) BC $\Leftarrow$ [7] $\underline{\text{lesma(Bill)}}$;

un 7 com [2,b], un com [5]: $\text{m-veloz}(\text{Pat}, \text{Bill})$;

BC $\Leftarrow$ [8] $\underline{\text{m-veloz}(\text{Pat}, \text{Bill})}$;

un 8 com [3,a]: $\text{m-veloz}(\text{Bill}, z) \Rightarrow \text{m-veloz}(\text{Pat}, z)$.

un 8 com [3,b]: $\text{m-veloz}(x, \text{Pat}) \Rightarrow \text{m-veloz}(x, \text{Bill})$,

un com [6]: BC $\Leftarrow$ [9] $\underline{\text{m-veloz}(\text{Bob}, \text{Bill})}$;

un [9] com [3,a], ... un [9] com [3,b], ...

Backward chaining

Disparado para encontrar resposta(s) a uma consulta à BC:

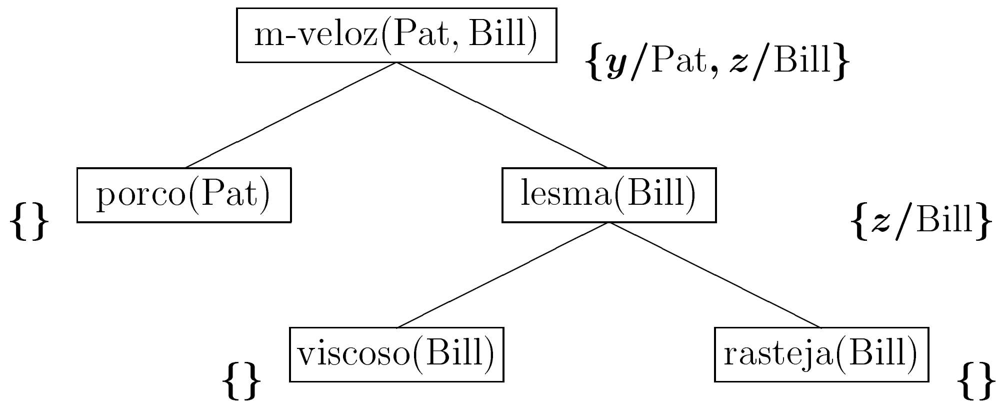
> Para uma consulta q:
> se q unifica com um fato q', retorne o unificador;
> para cada regra cujo consequente q' unifica com q,
> tente provar cada premissa da regra por backward chaining.

- Complicações maiores;
  - Lidar com unificação.
  - Evitar *loops*.
- Cada vez que a BC recebe uma consulta (via ASK), é disparado.
- Duas versões:
  - Encontrar qualquer solução.
  - Encontrar todas as soluções.
- É o algoritmo base de Prolog.

Exemplo de Backward chaining:

1. porco(y) $\wedge$ lesma(z) $\Rightarrow$ m-veloz(y, z)
2. viscoso(z) $\wedge$ rasteja(z) $\Rightarrow$ lesma(z)
3. porco(Pat)
4. viscoso(Bill)
5. rasteja(Pat)
6. rasteja(Bill)

Árvore de Prova:



Correção e completude:

- *Forward* e *backward chaining* são corretos e **completos** para BC's do tipo Horn (cláusulas definidas).
- *Forward* e *backward chaining* são corretos, mas **incompletos** para lógica de primeira ordem em geral.

BC:

1. doutor(x) $\Rightarrow$ altamente-qualificado(x)
2. $\neg$doutor(x) $\Rightarrow$ trabalha-cedo(x)
3. altamente-qualificado(x) $\Rightarrow$ rico(x)
4. trabalha-cedo(x) $\Rightarrow$ rico(x)

BC $\models$ rico(Eu), mas BC $\not\vdash_i$ rico(Eu) se i é qualquer um dos dois algoritmos.

Outra BC:

1. criança(x) $\Rightarrow$ tomar(x, Krlgsuem)
2. $\neg$criança(x) $\Rightarrow$ tomar(x, Ysoftars)

BC $\models$ tomar(Eu, Krlgsuem) $\vee$ tomar(Eu, Ysoftars).

Resolução

- Consequência lógica é indecidível; é semi-decidível:
  - É possível encontrar prova de α se BC $\models \alpha$.
  - Não é sempre possível determinar que BC $\not\models \alpha$.

  (Como no problema da parada.)

- Resolução é um procedimento de **refutação**:
  - Para provar BC $\models \alpha$, mostrar que BC $\wedge \neg\alpha$ é insatisfatível.

- Resolução usa BC$\wedge\neg\alpha$ na FNC (conjunção de cláusulas).

- A regra de resolução obtém uma cláusula a partir de duas outras:

$$\frac{C_1 \;\; C_2}{C}$$

- Meta: encontrar **cláusula vazia** (contradição).

A regra de inferência Resolução

Versão proposicional básica:

$$\frac{\alpha \vee \beta \quad \neg\beta \vee \gamma}{\alpha \vee \gamma} \quad \text{ou então} \quad \frac{\neg\alpha \Rightarrow \beta \quad \beta \Rightarrow \gamma}{\neg\alpha \Rightarrow \gamma}$$

Versão de primeira ordem completa (cláusulas):

$$\frac{p_1 \vee \ldots p_j \ldots \vee p_m \qquad q_1 \vee \ldots q_k \ldots \vee q_n}{(p_1 \vee \ldots p_{j-1} \vee p_{j+1} \ldots \vee p_m \vee q_1 \vee \ldots q_{k-1} \vee q_{k+1} \ldots \vee q_n)\sigma}$$

onde $p_j\sigma = \neg q_k\sigma$.

Exemplo:

$$\frac{\neg \text{rico}(x) \vee \text{infeliz}(x) \qquad \text{rico}(\text{Eu})}{\text{infeliz}(\text{Eu})}$$

com umg $\{x/\text{Eu}\}$.

Versão de primeira ordem completa (implicações):

$$\frac{p_1 \wedge \ldots p_j \ldots \wedge p_{n_1} \Rightarrow r_1 \vee \ldots r_{n_2} \qquad s_1 \wedge \ldots s_{n_3} \Rightarrow q_1 \vee \ldots q_k \ldots \vee q_{n_4}}{(p_1 \wedge \ldots p_{j-1} \wedge p_{j+1} \ldots \wedge p_{n_1} \wedge s_1 \wedge \ldots s_{n_3} \Rightarrow r_1 \vee \ldots r_{n_2} \vee q_1 \vee \ldots q_{k-1} \vee q_{k+1} \ldots \vee q_{n_4})\sigma}$$

onde $p_j\sigma = q_k\sigma$.

Formas normais para Resolução:

- Forma normal conjuntiva (FNC): conjunção de <u>cláusulas</u> da forma:

  $p_1 \vee p_2 \vee \ldots \vee p_n$

  onde cada p_i é um <u>literal</u>.

- Forma normal implicativa (FNI): conjunção de <u>implicações</u> da forma:

  $p_1 \wedge p_2 \wedge \ldots \wedge p_m \Rightarrow q_1 \vee q_2 \vee \ldots \vee q_n$

  onde cada p_i e cada q_j é uma <u>sentença atômica</u>.

Exemplos:

Cláusulas	*Implicações*
$\neg p(x) \vee q(x)$	$p(x) \Rightarrow q(x)$
$\neg p(x) \vee q(x) \vee \neg r(x) \vee s(x)$	$p(x) \wedge r(x) \Rightarrow q(x) \vee s(x)$
$p(x) \vee q(x)$	$\mathbf{V} \Rightarrow p(x) \vee q(x)$
$\neg p(x) \vee \neg q(x)$	$p(x) \wedge q(x) \Rightarrow \mathbf{F}$

Conversão de uma sentença para FNC:

1. Eliminar "$\Rightarrow$":

   $\alpha \Rightarrow \beta$ por $\neg\alpha \vee \beta$

2. Mover "$\neg$" para dentro:

   $\neg\neg\alpha$ por α

   $\neg(\alpha \wedge \beta)$ por $\neg\alpha \vee \neg\beta$

   $\neg(\alpha \vee \beta)$ por $\neg\alpha \wedge \neg\beta$

   $\neg\forall x\alpha$ por $\exists x\neg\alpha$

   $\neg\exists x\alpha$ por $\forall x\neg\alpha$

3. Colocar cada quantificador com uma variável diferente.

4. Eliminar os quantificadores existenciais (Skolemização):

   Tirar $\exists x$ e substituir x por $f(y_1, y_2, \ldots, y_n)$ se x está no escopo de quantificadores universais $\forall y_1 \forall y_2 \ldots \forall y_n$ (ver próxima transparência).

5. Eliminar os quantificadores universais.

6. Distribuir $\wedge$ sobre $\vee$:

   $(\alpha \wedge \beta) \vee \gamma$ por $(\alpha \vee \gamma) \wedge (\beta \vee \gamma)$

Skolemização/Exemplos:

- $\exists x$ feliz(x) torna-se feliz(C)

  C: *constante de Skolem.*

- $\forall x \exists y$ ama(x, y) torna-se $\forall x$ ama$(x, f(x))$

  f: *função de Skolem.*

- $\forall x$ pessoa$(x) \Rightarrow \exists y$ coração$(y) \land$ tem(x, y) torna-se $\forall x$ pessoa$(x) \Rightarrow$ coração$(c(x)) \land$ tem$(x, c(x))$

  c: *função de Skolem.*

- $\forall x \exists y \forall z \exists w p(x, y, z, w)$ por $\forall x \forall z p(x, f(x), z, g(x, z))$

  f e g: *funções de Skolem.*

- $\exists x \forall y [\forall z (p(x, y) \lor q(y, z)) \Rightarrow \exists w r(x, y, w)]$ por:

  $\exists x \forall y [\neg(\forall z (p(x, y) \lor q(y, z)) \lor \exists w r(x, y, w)]$ por:

  $\exists x \forall y [(\exists z \neg(p(x, y) \lor q(y, z)) \lor \exists w r(x, y, w)]$ por:

  $\exists x \forall y [\exists z (\neg p(x, y) \land \neg q(y, z)) \lor \exists w r(x, y, w)]$ por:

  $[\neg p(A, y) \land \neg q(y, f(y))] \lor r(A, y, g(y))$ por:

  $[\neg p(A, y) \lor r(A, y, g(y))] \land [\neg q(y, f(y)) \lor r(A, y, g(y))]$

Prova por Resolução:

Para provar α:

1. Negar α.
2. Converter para FNC.
3. Adicionar à BC (esta na FNC).
4. Inferir contradição (cláusula vazia).

Exemplos: transparências 11 e 12.

Tratamento da Igualdade:

- Regra de **demodulação**:

$$\frac{t_1 = t_2 \quad [\cdots t_1 \cdots]}{[\cdots t_2\sigma \cdots]}$$

  se σ é umg de t_1 e t_2.

- Regra de **paramodulação**.

Estratégias de Resolução:

- Preferência pelas unitárias.
- Conjunto de Suporte.
- Resolução de entrada.
- Resolução linear.
- Estratégias de eliminação:
  - Cláusulas puras.
  - Tautologias.
  - Subjugação.