Agentes Lógicos

Profa. Mercedes Gonzales Márquez

Procedimento de Inferência Simples

- Vamos construir uma Base de Conhecimento para o mundo do Wumpus.
- $P_{x,y}$ é verdadeiro se existe um poço em [x,y]
- $W_{x,y}$ é verdadeiro se existe um wumpus em [x,y], vivo o morto.
- $B_{x,y}$ é verdadeiro se o agente percebe uma brisa em [x,y].
- $F_{x,y}$ é verdadeiro se o agente percebe um fedor em [x,y]
- $L_{x,y}$ é verdadeiro se o agente está no local [x,y]

Procedimento de Inferência Simples

- As sentenças que escrevemos serão suficientes para derivar $\neg P_{1,2}$(não há poço em [1,2]) como fizemos anteriormente na verificação de modelos.

- $R_1 : \neg P_{1,1}$ (não há nenhum poço em [1,1])
- Um quadrado tem uma brisa sse existe um poço em um quadrado vizinho. Incluiremos os quadrados relevantes só.
- R_2: $B_{1,1} \Leftrightarrow (P_{1,2} \vee P_{2,1})$
- R_3: $B_{2,1} \Leftrightarrow (P_{1,1} \vee P_{2,2} \vee P_{3,1})$

Procedimento de Inferência Simples

- As sentenças precedentes são verdadeiras em todos os mundos dos wumpus. Agora incluímos as percepções de brisa para os dois primeiros quadrados visitados no mundo específico em que o agente se encontra (visto anteriormente)
- R4: $\neg B_{1,1}$
- R5: $B_{2,1}$

Procedimento de Inferência Simples

- O objetivo agora é saber se α_1 ou α_2 são consequências lógicas de BC.
- Lembre-se que $\alpha_1 = \neg P_{1,2}$ e $\alpha_2 = P_{2,2}$
- Os símbolos proposicionais relevantes são $B_{1,1}$, $B_{2,1}$, $P_{1,1}$, $P_{1,2}$, $P_{2,1}$, $P_{2,2}$, $P_{3,1}$. Com sete símbolos existem $2^7=128$ modelos possíveis.
- Em três desses modelos BC é verdadeira.
- Nesses três modelos α_1 é verdadeira, consequentemente não existe nenhum poço em [1,2].

Procedimento de Inferência Simples

- No entanto $P_{2,2}$ é verdadeira em dois dos três modelos e falsa em um; portanto não podemos dizer ainda se existe um poço em [2,2].

$B_{1,1}$	$B_{2,1}$	$P_{1,1}$	$P_{1,2}$	$P_{2,1}$	$P_{2,2}$	$P_{3,1}$	R_1	R_2	R_3	R_4	R_5	BC
falso	*falso*	*falso*	*falso*	*falso*	*falso*	*falso*	*verd.*	*verd.*	*verd.*	*verd.*	*falso*	*falso*
falso	*falso*	*falso*	*falso*	*falso*	*falso*	*verd.*	*verd.*	*verd.*	*falso*	*verd.*	*falso*	*falso*
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
falso	*verd.*	*falso*	*falso*	*falso*	*falso*	*falso*	*verd.*	*verd.*	*falso*	*verd.*	*verd.*	*falso*
falso	*verd.*	*falso*	*falso*	*falso*	*falso*	*verd.*	*verd.*	*verd.*	*verd.*	*verd.*	*verd.*	<u>*verd.*</u>
falso	*verd.*	*falso*	*falso*	*falso*	*verd.*	*falso*	*verd.*	*verd.*	*verd.*	*verd.*	*verd.*	<u>*verd.*</u>
falso	*verd.*	*falso*	*falso*	*falso*	*verd.*	*verd.*	*verd.*	*verd.*	*verd.*	*verd.*	*verd.*	<u>*verd.*</u>
falso	*verd.*	*falso*	*falso*	*verd.*	*falso*	*falso*	*verd.*	*falso*	*falso*	*verd.*	*verd.*	*falso*
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
verd.	*verd.*	*verd.*	*verd.*	*verd.*	*verd.*	*verd.*	*falso*	*verd.*	*verd.*	*falso*	*verd.*	*falso*

Figura 7.9 Tabela-verdade construída para a base de conhecimento dada no texto. A *BC* é verdadeira se R_1 a R_5 são verdadeiras, o que acontece em apenas três das 128 linhas (as que estão sublinhadas na coluna do lado direito). Em todas as três linhas, $P_{1,2}$ é falsa e, assim, não existe nenhum poço em [1,2]. Por outro lado, pode haver (ou não) um poço em [2,2].

Procedimento de Inferência Simples

- O algoritmo é correto, porque implementa diretamente a definição de consequência lógica, e é completo, porque funciona para qualquer Base de Conhecimento e α, e sempre termina – só existe um número finito de modelos a examinar.
- Finito não é sinónimo de poucos. Se tiver n modelos ao todo, então existem 2^n modelos. Desse modo a complexidade de tempo do algoritmo é $O(2^n)$.

Prova de Teoremas Proposicionais

- Até agora mostramos como determinar a consequência lógica por verificação de modelos: enumerar modelos e mostrar que a sentença deve valer em todos os modelos.
- Agora mostraremos como se pode determinar a consequência lógica por meio da **prova de teoremas.** Será feito pela aplicação direta de regras de inferência nas sentenças em nossa BC para construir uma prova da sentença desejada sem consultar os modelos.

Prova de Teoremas Proposicionais

- Se o número de modelos for grande, mas o comprimento da prova for curto, a prova de teoremas pode ser mais eficiente do que a verificação de modelos.

Equivalência lógica

Duas sentenças são logicamente equivalentes sse verdadeiras nos mesmos modelos: $\alpha \equiv \beta$ sse $\alpha \vDash \beta$ e $\beta \vDash \alpha$:

$$
\begin{aligned}
(\alpha \land \beta) &\equiv (\beta \land \alpha) \quad \text{commutativity of } \land \\
(\alpha \lor \beta) &\equiv (\beta \lor \alpha) \quad \text{commutativity of } \lor \\
((\alpha \land \beta) \land \gamma) &\equiv (\alpha \land (\beta \land \gamma)) \quad \text{associativity of } \land \\
((\alpha \lor \beta) \lor \gamma) &\equiv (\alpha \lor (\beta \lor \gamma)) \quad \text{associativity of } \lor \\
\neg(\neg\alpha) &\equiv \alpha \quad \text{double-negation elimination} \\
(\alpha \Rightarrow \beta) &\equiv (\neg\beta \Rightarrow \neg\alpha) \quad \text{contraposition} \\
(\alpha \Rightarrow \beta) &\equiv (\neg\alpha \lor \beta) \quad \text{implication elimination} \\
(\alpha \Leftrightarrow \beta) &\equiv ((\alpha \Rightarrow \beta) \land (\beta \Rightarrow \alpha)) \quad \text{biconditional elimination} \\
\neg(\alpha \land \beta) &\equiv (\neg\alpha \lor \neg\beta) \quad \text{de Morgan} \\
\neg(\alpha \lor \beta) &\equiv (\neg\alpha \land \neg\beta) \quad \text{de Morgan} \\
(\alpha \land (\beta \lor \gamma)) &\equiv ((\alpha \land \beta) \lor (\alpha \land \gamma)) \quad \text{distributivity of } \land \text{ over } \lor \\
(\alpha \lor (\beta \land \gamma)) &\equiv ((\alpha \lor \beta) \land (\alpha \lor \gamma)) \quad \text{distributivity of } \lor \text{ over } \land
\end{aligned}
$$

Validade e Satisfatibilidade

- Uma sentença é válida se verdadeira em todos os modelos. Exemplo : $A \vee \neg A$
- Validade é ligada à inferência via o Teorema da Dedução. Para qq sentenças α e β

  $\beta \models \alpha$ se e somente se $(\beta \Rightarrow \alpha)$ é válida (tautologia).
- Uma sentença é satisfatível se é verdadeira em algum modelo.
- Uma sentença é insatisfatível se não é verdadeira em nenhum modelo.

Validade e Satisfatibilidade

- Satisfatibilidade é ligada à inferência via o seguinte resultado:
  - $\beta \models \alpha$ se e somente se $(\beta \wedge \neg\alpha)$ é insatisfatível

Raciocínio de prova por contradição.

Inferência e Provas

Regras de inferência:

– Modus ponens

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

– Eliminação-de-e

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

– Todas as equivalências anteriores podem ser usadas como regras de inferência.

BC mundo do Wumpus

Seja P_{ij} verdade se existe um poço em [i, j].

Seja B_{ij} verdade se há brisa em [i, j].

R1: $\neg P_{11}$

R2: $\neg B_{11}$

R3: B_{21}

"Poços causam brisas em quadrados adjacentes "

R4: $B_{11} \Leftrightarrow (P_{12} \vee P_{21})$

R5: $B_{21} \Leftrightarrow (P_{11} \vee P_{22} \vee P_{31})$

BC mundo do Wumpus

Seja a base de conhecimento R1 -- R5, vamos provar $\neg P_{12}$:

- *Eliminação de bicondicional a* **R4***:*

  R6: $(B_{11} \Rightarrow (P_{12} \vee P_{21})) \wedge ((P_{12} \vee P_{21}) \Rightarrow B_{11})$

- *Eliminação-e em* **R6***:*

  R7: $(B_{11} \Rightarrow (P_{12} \vee P_{21}))$ e R7`: $((P_{12} \vee P_{21}) \Rightarrow B_{11})$

- *Contraposição em* **R7`**:

  R8: $(\neg B_{11} \Rightarrow \neg (P_{12} \vee P_{21}))$

- *Modus ponens com* **R2** *e* **R8**:

  R9: $(\neg (P_{12} \vee P_{21}))$

BC mundo do Wumpus

– Regra de de Morgan em R9:

$$\textbf{R10: } \neg P_{12} \wedge \neg P_{21}$$

i.e. nem [1,2], nem [2,1] possui um poço!

Resolucão

Forma Normal Conjuntiva -- *Conjunctive Normal Form* (CNF)

conjunção de disjunções de literais

E.g., $(A \vee \neg B) \wedge (B \vee \neg C \vee \neg D)$

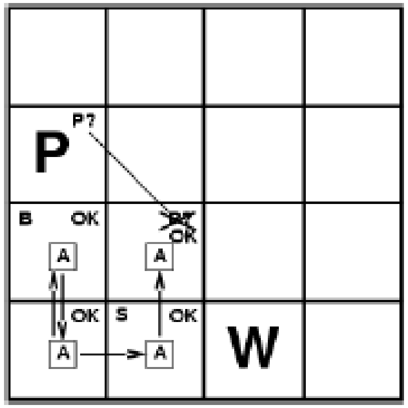
Regra de inferência resolução (para CNF):

$$\frac{\ell_1 \vee \ldots \vee \ell_k, \qquad m_1 \vee \ldots \vee m_n}{\ell_1 \vee \ldots \vee \ell_{i-1} \vee \ell_{i+1} \vee \ldots \vee \ell_k \vee m_1 \vee \ldots \vee m_{j-1} \vee m_{j+1} \vee \ldots \vee m_n}$$

onde ℓ_i e m_j são literais complementares.

$$\frac{\ell_1 \vee \ell_2 \qquad \neg \ell_2 \vee \ell_3}{\ell_1 \vee \ell_3}$$

E.g., $\dfrac{P_{1,3} \vee P_{2,2}, \qquad \neg P_{2,2}}{P_{1,3}}$



correta e completa para lógica proposicional

Conversão para Forma Normal Conjuntiva

$$B_{1,1} \Leftrightarrow (P_{1,2} \vee P_{2,1})$$

1. Eliminar $\Leftrightarrow$, trocando $\alpha \Leftrightarrow \beta$ por $(\alpha \Rightarrow \beta) \wedge (\beta \Rightarrow \alpha)$.

   $(B_{1,1} \Rightarrow (P_{1,2} \vee P_{2,1})) \wedge ((P_{1,2} \vee P_{2,1}) \Rightarrow B_{1,1})$

2. Eliminar $\Rightarrow$, trocando $\alpha \Rightarrow \beta$ por $\neg \alpha \vee \beta$.

   $(\neg B_{1,1} \vee P_{1,2} \vee P_{2,1}) \wedge (\neg(P_{1,2} \vee P_{2,1}) \vee B_{1,1})$

3. Mover $\neg$ para dentro usando as leis de de Morgan e negação dupla:

   $(\neg B_{1,1} \vee P_{1,2} \vee P_{2,1}) \wedge ((\neg P_{1,2} \wedge \neg P_{2,1}) \vee B_{1,1})$

4. Aplicar a lei distributiva ($\wedge$ sobre $\vee$) e eliminar '(' ')':

   $(\neg B_{1,1} \vee P_{1,2} \vee P_{2,1}) \wedge (\neg P_{1,2} \vee B_{1,1}) \wedge (\neg P_{2,1} \vee B_{1,1})$

Algoritmo de Resolução

Primeiramente a fórmula de entrada ($BC \wedge \neg\alpha$) é convertida em CNF.

Em seguida a regra de resolução é aplicada às cláusulas resultantes da conversão. Cada par que contém literais complementares é resolvido para gerar uma nova cláusula (resolvente), que é adicionada a um conjunto, caso ainda não esteja presente.

Algoritmo de Resolução

O processo continua até que:

- não exista nenhuma cláusula nova a ser adicionada; nesse caso, não há consequência lógica.
- a **cláusula vazia** é derivada, ou seja, há duas cláusulas cuja a conjunção é uma contradição ($P \land \neg P$); assim, a consequência lógica é verificada.

Prova por contradição

Exemplo de Resolução

$BC = (B_{1,1} \Leftrightarrow (P_{1,2} \lor P_{2,1})) \land \neg B_{1,1}$

$\alpha = \neg P_{1,2}$