1

Conhecimento e Raciocínio
Lógica Proposicional

2

Agente Baseado em Conhecimento ou Sistema Baseado em Conhecimento

- Representa conhecimento sobre o mundo em uma ***linguagem formal*** (*KB*)
- Raciocina sobre o mundo fazendo ***inferências*** na linguagem de representação adotada na KB

$$KB \vdash_i \alpha$$

- Decidi que ação tomar ***inferindo*** sobre qual ação selecionada é a melhor

Inferência Lógica

- *raciocínio ou inferência*: processos para se chegar a uma conclusão (exemplo no mundo de wumpus: posição (2,2) é segura?)
- *inferência lógica*: processo que implementa a relação de encadeamente entre sentenças
- existem várias maneira de se implementar uma inferência lógica

Modelos

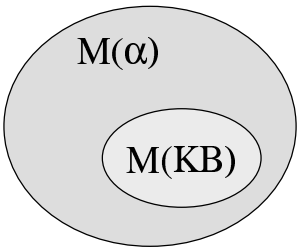
Lógicos pensam em termos de modelos, que são mundos formalmente estruturados com relação à verdade das sentenças

Dizemos que *M* é um modelo de uma sentença α se α é verdade em *M*

Seja M(α) o conjunto de todos os modelos de α

Então *KB* |= α sse M(KB) $\subseteq$ M(α)

interpretações	A	B	α: A=>B
I_1	F	F	V
I_2	F	V	V
I_3	V	F	F
I_4	V	V	V



5

Sentenças Válidas e Satisfatíveis

- Uma sentença é valida (ou necessariamente verdadeira) sse ela for verdade em todas as possíveis interpretações (coluna toda V)

Ex.: $A \vee \neg A$, $A \rightarrow A$, $(A \wedge (A \rightarrow B)) \rightarrow B$

$KB \models S$ sse $(KB \rightarrow S)$ é válida → *independe do significado*

- Uma sentença é *satisfatível* se ela for verdade em algum modelo
- Uma sentença é *não-satisfatível* se, em nenhum modelo, ela é verdade. Ex: $A \wedge \neg A$

$KB \models S$ sse $(KB \wedge \neg S)$ é *não-satisfatível*

6

Satisfatibilidade - exemplo

- *Uma sentença é satisfatível sse for verdadeira para alguma interpretação em algum mundo*
- "Existe um abismo em [2,2]" pode ser verdadeira para algum mundo do Wumpus
- *Uma sentença é não-satisfatível sse não existir nenhuma interpretação em algum mundo em que ela seja verdadeira*
- "Não existe um abismo em [2,2] e existe um abismo em [2,2]"

$$\neg A \wedge A$$

7

Verificação de sentenças válidas

- A pergunta "[2,2] é OK?" deve ser feita da seguinte forma:

  "se KB é verdade então [2,2] é uma posição segura" é uma sentença válida?
- *KB* é uma conjunção de milhares de sentenças e *P* é uma sentença complexa

como construir um procedimento formal para verificação de sentenças válidas?

8

Lógica Proposicional

- sintaxe
- semântica
- validade e inferência
- modelos
- regras de inferência para LP
- provadores de teoremas: método de prova (e resolução de problema) de propósito geral

9

Semântica do Cálculo Proposicional

- Significado de uma sentença através da noção de interpretações de sentenças
  - atribuição de valores verdade para cada um dos símbolos e constantes proposicionais e
  - especificação do significado dos conectivos lógicos (tabelas verdade).
- A verdade da fórmula geral é calculada baseada nas leis da Lógica Booleana: como computar a verdade de sentenças envolvendo conectivos ($\wedge,\vee,\rightarrow,\leftrightarrow$) dados os valores verdade de seus constituintes ?

10

LP - Validade e inferência

- Validade e inferência
  - dada uma sentença podemos fazer uma tabela verdade com uma linha para cada uma das possíveis combinações de valores verdade para os símbolos proposicionais da sentença
  - para cada linha podemos calcular o valor da sentença inteira. Se a sentença for verdade para todas as linhas então a sentença é válida. Ex:

    $((P \vee H) \wedge \neg H) \rightarrow P$ é válida e portanto,

    se é verdade $((P \vee H) \wedge \neg H)$ podemos inferir P

Inferência Proposicional: Método da Enumeração

Seja α: AvB e KB: (AvC)^(Bv~C)

KB |= S ??

Verificação de todos os modelos possíveis de S: devem ser verdadeiros sempre que KB é verdade

- 2^n mundos ou interpretações

A	B	C	AvC	Bv~C	KB	α
F	F	F				
F	F	V				
F	V	F				
F	V	V				
V	F	F				
V	F	V				
V	V	F				
V	V	V				

Inferência Proposicional: solução

A	B	C	A∨C	B∨¬C	KB	S	KB→S
F	F	F	F	V	F	F	V
F	F	V	V	F	F	F	V
F	V	F	F	V	F	V	V
F	V	V	V	V	V	V	V
V	F	F	V	V	V	V	V
V	F	V	V	F	F	V	V
V	V	F	V	V	V	V	V
V	V	V	V	V	V	V	V

KB |= S

LP - problema

Dada uma fórmula contendo n átomos, existem 2^n interpretações possíveis (atribuição de valores V/F aos átomos individuais da fórmula)

Regras de Inferência

- Tabelas verdade podem ser extendidas para classes de inferências
- Existem certos padrões de inferências que ocorrem com bastante frequência: *regras de inferência*
- podem ser usadas para fazer inferência sem termos que construir tabelas de verdade

15

Notações usadas para regras de inferência

- Sendo α , β sentenças da lógica proposicional (separadas por ",")
- $\alpha \vdash \beta$, significa que β pode ser derivada por α através de inferências.
- Ou:

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

para enfatizar que a regra de inferência não se trata de uma sentença da linguagem (isto é, de uma meta-linguagem)

16

Notações usadas para regras de inferência

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

Sempre que algo na KB casar com o *padrão* acima da linha, a regra de inferência conclue o que está abaixo da linha

7 regras de inferência para a LP

$\frac{\alpha \Rightarrow \beta,\ \alpha}{\beta}$	**Modus Ponens**	**Da implicação e da premissa infere-se a conclusão**
$\frac{\alpha_1 \Lambda \alpha_2 \Lambda \ \ldots \ \Lambda \alpha_n}{\alpha_n}$	**Eliminação**	**Da conjunção infere-se qualquer α_n**
$\frac{\alpha_1, \alpha_2, \ \ldots\ , \ \alpha_n}{\alpha_1 \Lambda \alpha_2 \Lambda \ \ldots \ \Lambda \alpha_n}$	**Introdução da conjunção**	**De uma lista de sentenças infere-se a sua conjunção**
$\frac{\alpha_i}{\alpha_1 \mathbf{V} \alpha_2 \mathbf{V} \ \ldots \ \mathbf{V} \alpha_n}$	**Introdução da disjunção**	**De uma sentença, infere-se sua disjunção com qualquer outra**
$\frac{\neg \neg \alpha}{\alpha}$	**Negação dupla**	**De uma negação dupla infere-se uma senetnça positiva**
$\frac{\alpha \ \mathbf{V}\ \beta, \neg \beta}{\alpha}$	**Resolução simples**	**Se uma das disjunções for falsa, pode-se inferir que a outra é verdade**
$\frac{\alpha \ \mathbf{V}\ \beta, \neg \beta \ \mathbf{V}\ \gamma}{\alpha \ \mathbf{V}\ \gamma}$	**Resolução**	**β , não pode ser Verdade e Falso ao mesmo tempo**

Métodos de prova

Podem ser divididos em dois tipos:

Verificação de modelos:

- enumeração na tabela verdade (correto e completo para a LP, porém = 2^n para n símbolos proposicionais)
- busca heurística no espaço de modelos (correto mas não completo)

Aplicação de regras de inferência:

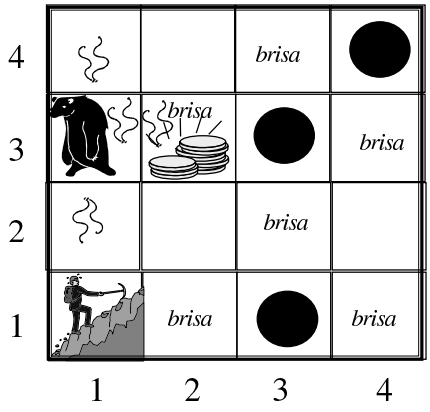
- garante a geração de novas sentenças a partir de velhas
- prova = uma sequência de aplicações de regras de inferência
- pode-se usar regras de inferência como operadores em algoritmos de busca

Complexidade da prova lógica

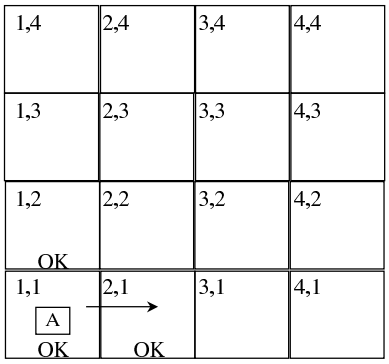
- **Monotonocidade:** uso local de regras de Inferência (se as sentenças derivadas de uma KB, continuam sendo derivadas da KB quando adicionamos novas sentenças a ela)
- **Sentenças de Horn**:
  – *P1 ^ P2 ^ ... ^Pn => Q*

  reduzem a complexidade para polinomial
  – Permitem que se use somente Modus Ponens
  – Importante: nem toda KB pode ser escrita nessa forma

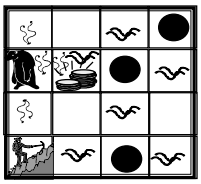
Um agente lógico no mundo de Wumpus



O mundo de Wumpus - LP

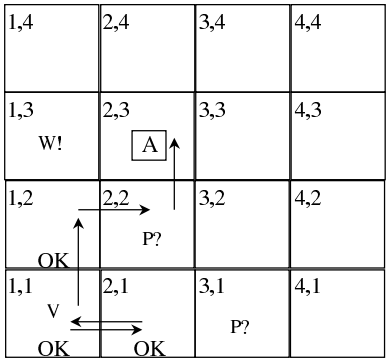


A = Agente
B = Brisa
G = Gold, brilho
OK = quadrado seguro
M = abismo
C = Cheiro
V = Visitado
W = Wumpus

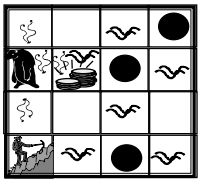


Percepção (i,j): [<*Cheiro*>, <*Brisa*>, <*Brilho*>, <*Choque*>, <*Grito*>]

O mundo de Wumpus - LP



A = Agente
B = Brisa
G = Gold, brilho
OK = quadrado seguro
M = abismo
C = Cheiro
V = Visitado
W = Wumpus



Percepção (1,1): [nada, nada, nada, nada, nada]
Percepção (2,1): [nada, *Brisa*, nada, nada, nada]
Percepção (1,2): [*Cheiro*, nada, nada, nada, nada]
Percepção (2,3): [*Cheiro*, *Brisa*, *Brilho*, nada, nada]

Situações de perigo

- Brisa em (1,2) e (2,1) => não existe ação segura
- Cheiro em (1,1) => não existe ação seguro

A Base de Conhecimento

Escolha de símbolos proposicionais:

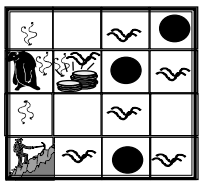
$S_{1,2}$ significa "Existe um cheiro em (1,2)"

$B_{2,1}$ significa "Existe uma brisa em (2,1)"

A sequência de percepçções deve ser convertida em sentenças introduzidas na KB + sentenças válidas derivadas dessas sentenças de percepção

KB depois do terceiro movimento:

$\sim S_{1,1}$	$\sim B_{1,1}$
$\sim S_{2,1}$	$B_{2,1}$
$S_{1,2}$	$\sim B_{1,2}$



A Base de Conhecimento

- Conhecimento sobre o ambiente. Por ex.:
  - Se não tem cheiro em um quadrado, o agente precisa saber que o Wumpus não mora no quadrado e em nenhum dos quadrados vizinhos (para todos os quadrados)

  R_1: $\sim S_{1,1} \Rightarrow \sim W_{1,1} \wedge \sim W_{1,2} \wedge \sim W_{2,1}$

  R_2: $\sim S_{2,1} \Rightarrow \sim W_{1,1} \wedge \sim W_{2,1} \wedge \sim W_{2,2} \wedge \sim W_{3,1}$

  R_3: $\sim S_{1,2} \Rightarrow \sim W_{1,1} \wedge \sim W_{1,2} \wedge \sim W_{2,2} \wedge \sim W_{1,3}$

  Se tem cheiro em (1,2) então o Wumpus deve morar em (1,2) ou em algum dos quadrados vizinhos

  R_1: $S_{1,2} \Rightarrow W_{1,3} \wedge W_{1,2} \wedge W_{2,2} \wedge W_{1,1}$

Procurando o Wumpus

- queremos concluir $W_{1,3}$
- KB $\rightarrow$ $W_{1,3}$ é válida?
  - Considerando 12 símbolos proposicionais:

    $S_{1,1}$, $S_{2,1}$, $S_{1,2}$,$W_{1,1}$,$W_{1,2}$,$W_{2,1}$,$W_{2,2}$,$W_{3,1}$, $W_{1,3}$, $B_{1,1}$, $B_{2,1}$, $B_{1,2}$

    e aplicando o método da Tabela Verdade teriamos: $2^{12} = 4096$ linhas e toda linha que KB é verdade, $W_{1,3}$ também deve ser verdade.
- Ao invés disso usamos as regras de inferência

Procurando o Wumpus (pag. 176)

1. Modus ponens:

R_1: $\sim S_{1,1} \Rightarrow \sim W_{1,1} \wedge \sim W_{1,2} \wedge \sim W_{2,1}$, $\sim S_{1,1}$

$\sim W_{1,1} \wedge \sim W_{1,2} \wedge \sim W_{2,1}$

2. Eliminação (*And-Elimination*) no resultado acima:

$\sim W_{1,1} \wedge \sim W_{1,2} \wedge \sim W_{2,1}$

$\sim W_{1,1}$, $\sim W_{1,2}$, $\sim W_{2,1}$

… continuação da prova em 7 passos (pag 175, AIMA)

Selecionando ações

- relação entre o estado do mundo e ação
- Exemplo de regra para o caso em que o agente está em (1,1) de frente para o leste:

$A_{1,1} \wedge \text{Leste}_a \wedge W_{2,1} \rightarrow \neg$ ação-parafrente

Lógica Proposicional é fraca

- só para a completar a regra anterior: "não vá para frente se o Wumpus estiver na sua frente" seriam necessárias 64 regras (16 posições x 4 orientações do agente)
- Mudanças no tempo:

$A_{1,1}^{0} \wedge Leste_{a}^{0} \wedge W_{2,1}^{0} \rightarrow \neg$ ação-parafrente ^{0}

- Para 100 tempos: 6400 regras proposicionais (LPO reduz para 1 regra !!! Introdução de objetos e relações)

Lógica de Primeira Ordem

- Objetos: indivíduos do mundo (pessoas, casas, números, cores)
- funções: mapeamento de indivíduos para indivíduos. Ex.: pai-de(Maria)=Joao, cor-de(Carro)=Vermelha
- relações: mapeamento entre indivíduos e valores Verdades. Ex.: maior-que(5,3), verde(grama); cor(Carro, Vermelho)

Lógica de Primeira Ordem

- Constantes: Joao, 2, USP, ...
- Predicados: irmão, >, ...
- Funções: RaizQuadrada, DistânciaXY, ...
- Variáveis: x, y, a, b, ...
- Conectivos: $\wedge$, $\vee$, $\neg$, $\rightarrow$, $\leftrightarrow$
- Igualdade: =
- Quantificadores: $\forall$ (Universal), $\exists$ (Existencial)

Lógica de Primeira Ordem

- Fatos em LPO
- **Sentenças atômicas**: predicado(termo$_1$, ..., termo$_n$)
- **Termo**: constante, variável, função(termo$_1$, ..., termo$_n$)
- Exemplo: casado(Pai-de(Joao), Mae-de(Joao))

Lógica de Primeira Ordem

- Como na Lógica Proposicional, sentenças são formadas a partir dos conectivos:

Se S é uma sentença, $\neg$S é uma sentença

Se S1 e S2 são sentenças, S1 $\wedge$ S2 é uma sentença

Se S1 e S2 são sentenças, S1 $\vee$ S2 é uma sentença

Se S1 e S2 são sentenças, S1 $\rightarrow$ S2 é uma sentença

Se S1 e S2 são sentenças, S1 $\leftrightarrow$ S2 é uma sentença

Lógica de Primeira Ordem: Semântica

- Sentenças são verdadeiras com relação a um modelo.
- Um modelo contém objetos e relações entre eles
- Uma sentença atômica *predicado*($termo_1$, ..., $termo_n$) é verdadeira sse os objetos referidos pelos termos $termo_1$, ..., $termo_n$ estão na relação referida por *predicado*

Quantificador Universal

∀ <variáveis> <sentenças>

Exemplo: **"Todo mundo na USP é inteligente"**

∀ x ∈ ConjPessoas, Em(x,USP) → Inteligente(x)

∀ x *P* equivale à <u>conjunção</u> de instanciações de *P*

Em(Rodrigo, USP) → Inteligente(Rodrigo)
^ Em(Paulo, USP) → Inteligente(Paulo)
^ Em(Renato, USP) → Inteligente(Renato)
^ ...

P

Quantificador Existencial

∃ <variáveis> <sentenças>

Exemplo: **Alguém na USP é inteligente**

∃ x Em(x,USP) ∧ Inteligente(x)

∃ x *P* equivale à <u>disjunção</u> de instanciações de *P*

Em(Fulano, USP) ∧ Inteligente(Fulano)
∨ Em(Cicrano, USP) ∧ Inteligente(Cicrano)
∨ ...

P