

Parte II: IA Simbólica ou Good Old-Fashioned Artificial Intelligence (GOFAI).

Prof. Fabiano Araujo Soares, Dr. / FGA 0221 - Inteligência Artificial

Universidade de Brasília

2025



Como um agente artificial consegue descobrir verdades e tomar decisões em ambientes incertos apenas usando lógica e conhecimento disponível?



Como um agente artificial consegue descobrir verdades e tomar decisões em ambientes incertos apenas usando lógica e conhecimento disponível?

A inteligência artificial realmente desperta quando consegue não apenas perceber o mundo, mas também raciocinar sobre ele — transformando conhecimento em decisões.

Definição: Agentes que usam representações lógicas de conhecimento para raciocinar e tomar decisões.

Componentes Fundamentais:

- **Base de Conhecimento (KB):** Conjunto de sentenças que representam afirmações sobre o mundo
- **Operações TELL e ASK:** Adicionar conhecimento e fazer consultas
- **Inferência:** Processo de derivar novas sentenças a partir de conhecimento existente

Diferencial: Ao contrário de agentes reativos, agentes lógicos podem raciocinar sobre situações não experimentadas anteriormente, usando regras lógicas para inferir novos fatos.

Pseudocódigo: Agentes baseado em conhecimento.

```
function KB-AGENT(percept) returns an action  
  persistent: KB, a knowledge base  
             t, a counter, initially 0, indicating time  
  
  TELL(KB, MAKE-PERCEPT-SENTENCE(percept, t))  
  action  $\leftarrow$  ASK(KB, MAKE-ACTION-QUERY(t))  
  TELL(KB, MAKE-ACTION-SENTENCE(action, t))  
  t  $\leftarrow$  t + 1  
  return action
```

Princípios Fundamentais:

- **Representação Explícita:** O conhecimento é armazenado como sentenças lógicas formais
- **Separação:** Conhecimento (dados) separado do mecanismo de inferência (processamento)
- **Modularidade:** Novos conhecimentos podem ser adicionados sem alterar o sistema de inferência

Vantagens:

- Transparência no raciocínio
- Facilidade para adicionar novo conhecimento
- Capacidade de explicar decisões tomadas

Micromundo: **Mundo de Wumpus**

Vamos utilizar um micromundo para exemplificar o uso de agentes baseados em conhecimento:

O mundo de Wumpus

- O ambiente é uma caverna composta por câmaras ligadas por passagens;
- Em algum lugar da caverna está o terrível **Wumpus**, um monstro que devora qualquer um que entrar em sua câmara;
- Um agente pode matar um Wumpus, mas ele tem apenas uma flecha;
- Algumas câmaras contêm poços que vão aprisionar qualquer um que entrar na câmara (Exceto o Wumpus);
- Existe em alguma câmara da caverna um tesouro.

Mundo de Wumpus: Descrição PEAS

Performance:

- +1000 por sair da caverna com o tesouro,
- -1000 por cair em algum poço ou ser devorado pelo Wumpus,
- -1 para cada ação tomada,
- -10 por usar a flecha
- O jogo termina quando o agente morrer ou quando ele sai da caverna.

Mundo de Wumpus: Descrição PEAS

Ambiente:

- Um grid 4×4 de salas, com paredes ao redor;
- O agente começa no quadrado rotulado $[1,1]$, voltado para o leste.
- A localização do ouro e os wumpus são escolhidos aleatoriamente, com distribuição uniforme;
- Além disso, cada quadrado que não seja o início pode ser um poço, com probabilidade $p = 0,2$.

Atuadores:

- Mover para a frente;
- Virar 90° para a esquerda;
- Virar 90° para a direita;
- Atirar a flecha (a flecha segue em linha reta até matar o Wumpus ou atingir uma parede);
- Pegar o ouro (apenas se estiver na mesma câmara que o agente);
- Sair da caverna (Apenas na câmara $[1,1]$).

Mundo de Wumpus: Descrição PEAS

Sensores:

- Nas câmaras diretamente adjacentes ao wumpus (não na diagonal), o agente percebe um fedor;
- Nas câmaras diretamente adjacentes a um poço (não na diagonal), o agente percebe uma Brisa;
- No quadrado onde está o ouro, o agente percebe um brilho;
- Quando um agente tenta entrar em uma parede, ele percebe um solavanco;
- Quando o wumpus é morto, ele emite um urro que pode ser percebido em qualquer parte da caverna.
- As percepções serão dadas ao agente na forma de uma lista de cinco símbolos, por exemplo, se houver um fedor e uma brisa, mas nenhum brilho, solavanco ou grito, o agente obterá [Fedor,Brisa,Nenhum,Nenhum,Nenhum].

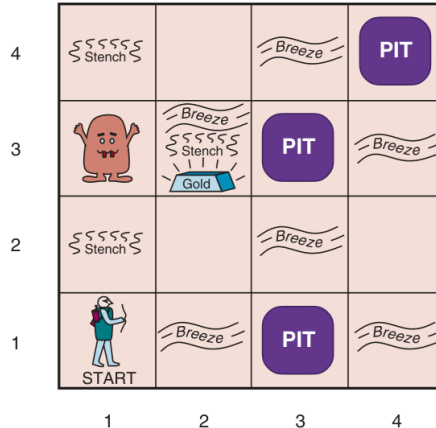
Mundo de Wumpus: Definições finais

Considerações finais sobre o micromundo **Mundo de Wumpus**

- O Wumpus não se move;
- O Wumpus World é um ambiente determinístico, discreto, estático e de agente único;
- É Sequencial pois a recompensa só virá após algumas ações;
- É parcialmente observável.

Exemplo do mundo de Wumpus

Realização do Mundo: Após posicionar o Wumpus, o ouro, e os poços aleatoriamente, incluímos os sinais desses elementos.



Símbolos Proposicionais:

- $P_{x,y}$: Poço na localização (x,y)
- $W_{x,y}$: Wumpus na localização (x,y)
- $B_{x,y}$: Brisa na localização (x,y)
- $S_{x,y}$: Fedor na localização (x,y)

Regras da KB (Knowledge Base):

- $\neg P_{1,1}$ (não há poço no início)
- $B_{x,y} \Leftrightarrow (P_{x-1,y} \vee P_{x+1,y} \vee P_{x,y-1} \vee P_{x,y+1})$
- $S_{x,y} \Leftrightarrow (W_{x-1,y} \vee W_{x+1,y} \vee W_{x,y-1} \vee W_{x,y+1})$

Fundamentos:

- Sentenças atômicas: $P_{1,1}$, $P_{2,2}$ (proposições simples)
- Conectivos lógicos: \neg (não), \wedge (e), \vee (ou), \Rightarrow (implica), \Leftrightarrow (se e somente se)
- Regras de inferência: Equivalência lógica, Validade, Modus Ponens, eliminação por conjunção, eliminação bicondicional, resolução.

Exemplo de Regra:

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

"Há brisa em (1,1) se e somente se há um poço em (1,2) ou em (2,1)"

Limitações: Expressividade limitada para domínios complexos com muitos objetos

Exemplo do mundo de Wumpus

Ações: O agente inicia em [1, 1], não percebe brisa ou mal cheiro, e se desloca para [2, 1] onde ele percebe uma brisa.

1,4	2,4	3,4	4,4
1,3	2,3	3,3	4,3
1,2	2,2	3,2	4,2
OK			
1,1	2,1	3,1	4,1
A			
OK	OK		

(a)

A = Agent
B = Breeze
G = Glitter, Gold
OK = Safe square
P = Pit
S = Stench
V = Visited
W = Wumpus

1,4	2,4	3,4	4,4
1,3	2,3	3,3	4,3
1,2	2,2	3,2	4,2
OK	P?		
1,1	2,1	3,1	4,1
V	A	P?	
OK	B	OK	
OK	OK		

(b)

Exemplo de Inferência no Wumpus World

Vamos rotular cada sentença como R_i de forma que possamos nos referir a ela. Vamos considerar que as regras básicas do **Mundo de Wumpus** foram pré-programadas no agente.

- $R_1 : \neg P_{1,1};$
- $R_2 : \neg P_{2,1};$
- $R_3 : \neg P_{1,2};$
- $R_4 : B_{2,1} \iff (P_{2,2} \vee P_{3,1})$
- $R_5 : \neg S_{2,1}$

Exemplo de Inferência no Wumpus World

Ações: Vamos considerar agora que o agente vai até a posição [1, 2] onde ele percebe um mal cheiro.

1,4	2,4	3,4	4,4
1,3 W?	2,3	3,3	4,3
1,2 A S OK	2,2 W? P?	3,2	4,2
1,1 V OK	2,1 B V OK	3,1 P?	4,1

Exemplo de Inferência no Wumpus World

Vamos incluir as informações adquiridas agora na KB.

- $R_6 : \neg B_{1,2}$
- $R_7 : \neg S_{2,1}$
- $R_8 : S_{1,2} \iff (W_{2,2} \vee W_{1,3});$

A partir daqui precisamos usar de inferência para tomar uma decisão em relação ao que fazer.

Premissas e Equivalências

- Premissas:

- $\neg P_{1,1}, \neg P_{2,1}, \neg P_{1,2}$
- $B_{2,1} \iff (P_{2,2} \vee P_{3,1})$
- $\neg B_{1,2}$
- $S_{1,2} \iff (W_{2,2} \vee W_{1,3})$
- $\neg S_{2,1}$

- Notação:

- $P_{x,y}$: poço em (x, y) , $W_{x,y}$: Wumpus em (x, y)

Dedução: Ausência de Poço em (2, 2)

- $\neg B_{1,2}$ e ($P_{2,2}$ adjacente a (1, 2))
- Se $P_{2,2} \Rightarrow B_{1,2}$ (Modus Ponens)
- Como $\neg B_{1,2}$, então $\neg P_{2,2}$ (Modus Tollens)

Dedução: Poço em (3, 1)

- $B_{2,1} \iff (P_{2,2} \vee P_{3,1})$
- Já sabemos $\neg P_{2,2}$, então $B_{2,1} \iff P_{3,1}$ (eliminação bicondicional)
- Suponha $B_{2,1}$ é verdadeiro (ambiente clássico): então $P_{3,1}$ (Modus Ponens)

Dedução Ajustada: Wumpus em (1,3)

- $S_{1,2} \iff (W_{2,2} \vee W_{1,3})$
- Pelas regras do Mundo de Wumpus, o Wumpus em (2,2) geraria cheiro tanto em (1,2) quanto em (2,1).
- Como há cheiro em (1,2) mas **não** há cheiro em (2,1), $\therefore W_{2,2}$ é **falso** (resolução e eliminação por contexto).
- Portanto, $S_{1,2} \iff W_{1,3}$
- Percebendo $S_{1,2}$, concluímos $W_{1,3}$ (Modus Ponens).

Resumo das Estratégias de Inferência

- **Eliminação por conjunção:** extrair cada premissa.
- **Eliminação bicondicional:** converter \iff para duas implicações.
- **Modus Ponens:** $P, P \Rightarrow Q \Rightarrow Q$.
- **Modus Tollens:** $\neg Q, P \Rightarrow Q \Rightarrow \neg P$.
- **Resolução:** eliminação de literais possíveis a partir de contexto e negações.

Conclusão

Há um **Wumpus** em (1, 3) e um **poço** em (3, 1).

Todas as deduções seguem de modo válido usando equivalências lógicas, negações e inferência proposicional.

1,4	2,4	3,4	4,4
1,3 W!	2,3	3,3	4,3
1,2 A S OK	2,2 OK	3,2	4,2
1,1 V OK	2,1 B V OK	3,1 P!	4,1

Aspectos Temporais em Agentes Lógicos

- Em ambientes dinâmicos, é necessário modelar mudanças que acontecem ao longo do tempo.
- O índice de tempo permite distinguir entre fatos que variam em diferentes instantes.
- Exemplo: a posição do agente no tempo t pode ser diferente da posição no tempo $t + 1$.
- Utilizamos variáveis temporais para representar o estado do mundo em cada instante.

Axiomas Temporais (Frame Axioms)

- Axiomas que garantem que a maioria dos fatos não mudam a menos que uma ação os modifique.
- Exemplo: se o agente está em (1,1) no tempo 0 e não se movimenta, ele permanece em (1,1) no tempo 1.
- Esses axiomas evitam inconsistências e garantem a coerência da base de conhecimento ao longo do tempo.
- Importantes para raciocínio sobre ações e seus efeitos.

Axiomas de Sucessão

- Descrevem explicitamente as mudanças causadas pelas ações do agente.
- Exemplo: Se o agente estiver em $(1,1)$ no tempo t e executar a ação "andar para leste", então estará em $(2,1)$ no tempo $t + 1$.
- Combinados com os axiomas temporais, permitem rastrear a evolução do mundo.
- Facilita a formulação de planos e previsão dos estados futuros.

Exemplo de Sequência de Decisões com Índice de Tempo

- Estado inicial: agente em (1,1) no tempo 0.
- Ação no tempo 0: andar para leste.
- Estado no tempo 1: agente em (2,1).
- O agente pode perguntar: "É seguro andar para (3,1) no tempo 2?"
- A base de conhecimento inclui regras que consideram o índice de tempo para responder.

Importância do Índice de Tempo

- Permite representar mudanças e ação em ambientes dinâmicos.
- Fundamental para agentes que atuam em sequência de passos (planejamento).
- Ajuda a construir raciocínio temporal e prever consequências de ações.
- Base para modelagem mais realista de ambientes em IA.

Vantagens e Desvantagens das Linguagens de Representação do Conhecimento

- Lógica Proposicional
 - Vantagens
 - Simples, fácil de implementar;
 - Base para provas automáticas;
 - Boas ferramentas de inferência para domínios pequenos.
 - Desvantagens
 - Não expressa relações entre objetos;
 - Cresce exponencialmente para muitos fatos;
 - Baixa escalabilidade em domínios complexos.

Vantagens e Desvantagens das Linguagens de Representação do Conhecimento

- Lógica de 1ª Ordem
 - Vantagens
 - Expressiva (quantificadores, relações e estruturas);
 - Permite generalização e representação abstrata;
 - Fundamento da maioria dos sistemas baseados em regras.
 - Desvantagens
 - Inferência mais complexa e custosa;
 - Não lida naturalmente com incerteza.

Vantagens e Desvantagens das Linguagens de Representação do Conhecimento

- Lógica Fuzzy
 - Vantagens
 - Modela incerteza e vaguidade;
 - Ideal para controle aproximado;
 - Mais próxima da linguagem natural.
 - Desvantagens
 - Inferência pode ser difícil de interpretar;
 - Base teórica menos consolidada para domínios formais;
 - Integração limitada a grandes sistemas lógicos clássicos.

Vantagens e Desvantagens das Linguagens de Representação do Conhecimento

- Planning Domain Definition Language (PDDL)
 - Vantagens
 - Padrão de representação para planejamento automático;
 - Facilita definição de domínios e problemas em IA;
 - Suporte para muitos planejadores e benchmarks.
 - Desvantagens
 - Focado apenas em planejamento (não em conhecimento geral);
 - Sintaxe limitada para certos domínios complexos;
 - Não trata bem incerteza, ausência de probabilidades.

- Russell, S. J., & Norvig, P. (2021). *Artificial Intelligence: A Modern Approach*. 4^a ed.
- Slides oficiais: aima.cs.berkeley.edu
- Genesereth, M. R., & Nilsson, N. J. (1987). *Logical Foundations of Artificial Intelligence*. Morgan Kaufmann. Um livro fundamental que discute a lógica como base para agentes inteligentes.
- Brachman, R. J., & Levesque, H. J. (2004). *Knowledge Representation and Reasoning*. Morgan Kaufmann. Introduz conceitos centrais da representação do conhecimento e inferência usados em agentes lógicos.

Obrigado!

E-mail: fabiano-soares@unb.br

LinkedIn: <https://www.linkedin.com/in/fabiano-soares-06b6a821a/>

Site do curso: [https://www.fabiano-soares.eng.br/fga0221-inteligência-artificial](https://www.fabiano-soares.eng.br/fga0221-inteligencia-artificial)