

# Inteligência Artificial

Raoni F. S. Teixeira

## Aula 5 - Agentes Lógicos

### 1 Introdução

Esta aula apresenta agentes lógicos e seu uso para resolver problemas com base em fatos e regras.

Diferentemente dos métodos de otimização tradicionais estudados anteriormente, agentes lógicos encontram soluções usando uma abordagem declarativa: o programador especifica **o que** deve ser feito, sem precisar detalhar **como** executar as tarefas.

### 2 Agente Lógico

Um agente lógico toma decisões usando conhecimento formalizado e regras. Ele utiliza um motor de inferência para deduzir novas informações a partir de fatos conhecidos, resolvendo problemas ou executando ações.

Os três componentes principais de um agente lógico são:

- **Base de Conhecimento (KB):** Armazena fatos de forma estruturada.
  - Exemplo: “A água ferve a 100°C”; “Se está chovendo, o chão está molhado.”
- **Regras de Inferência:** Permitem derivar conclusões a partir dos fatos.
  - Exemplo: “Se A implica B, e A é verdadeiro, então B também é verdadeiro.”
- **Motor de Inferência:** Aplica as regras para deduzir novos fatos ou resolver problemas.

A Figura 1 ilustra como o motor de inferência usa fatos e percepções para atualizar a base de conhecimento.

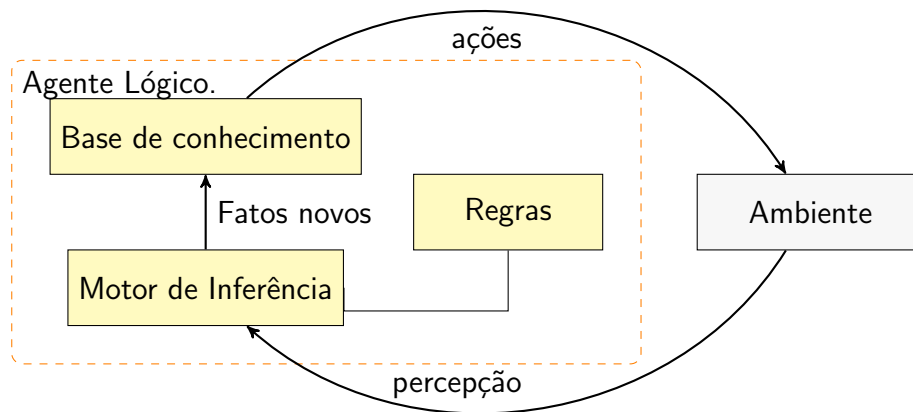


Figura 1: Agente Lógico que usa as percepções e o que já é conhecido para inferir novos fatos.

## 2.1 Tipos de Inferência

- **Modus Ponens:** Se  $A \rightarrow B$  e  $A$  são verdadeiros, então  $B$  é verdadeiro.
  - Exemplo:
    - \* Regra: Se está chovendo, então o chão está molhado.
    - \* Fato: Está chovendo.
    - \* Conclusão (fato novo): O chão está molhado.
- **Modus Tollens:** Se  $A \rightarrow B$  e  $\neg B$ , então  $\neg A$ .
  - Exemplo:
    - \* Regra: Se há fogo, então há fumaça.
    - \* Fato: Não há fumaça.
    - \* Conclusão (fato novo): Não há fogo.
- **Encadeamento para Frente (Forward Chaining):** Parte de fatos conhecidos e aplica regras até alcançar um objetivo.
- **Encadeamento Regressivo (Backward Chaining):** Parte do objetivo e verifica quais fatos o suportam, dividindo-o em subproblemas menores até encontrar uma solução.

### 3 Exemplo: Mundo de Wumpus

No clássico Mundo do Wumpus, um agente explora uma grade  $4 \times 4$  em busca de ouro, enquanto evita perigos como o Wumpus (um monstro) e poços. O agente utiliza percepções para deduzir a localização de perigos e ouro:

- **Fedor:** Indica a proximidade do monstro Wumpus.
- **Brisa:** Indica a proximidade de um poço.
- **Brilho:** Indica a presença de ouro na célula.

A Figura 2 mostra um tabuleiro. O agente começa no canto inferior esquerdo (1,1) e utiliza percepções para deduzir onde estão os perigos e o ouro, aplicando as seguintes regras:

- Se há *fedor* em uma célula, o Wumpus está em uma célula adjacente.
- Se há *brisa* em uma célula, um poço está em uma célula adjacente.
- Se há *brilho*, o ouro está na mesma célula.

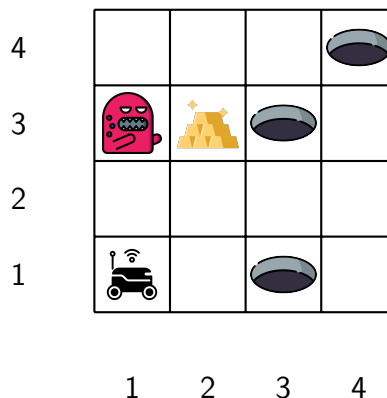


Figura 2: Mundo do Wumpus: o agente deve evitar poços e o monstro enquanto procura ouro.

Enquanto explora, o agente atualiza sua base de conhecimento. Por exemplo:

1. Ao não detectar *fedor* ou *brisa* em (1,1), conclui que (1,2) e (2,1) são seguras.
2. Ao explorar (2,1) e perceber *fedor*, deduz que o Wumpus está em (2,2) ou (3,1).

3. Ao avançar para (1,2) e sentir *brisa*, confirma que (2,2) é segura e que o Wumpus está em (3,1).

Mais tarde, ao encontrar *brilho* em (3,2), o agente pega o ouro e retorna ao ponto inicial, utilizando o mesmo raciocínio lógico.

## Alguns detalhes sobre o processo de inferência

A inferência pode ser realizada de duas formas. A primeira é o **encadeamento para frente**, que parte de fatos conhecidos e aplica regras para deduzir novas informações. No Mundo do Wumpus (Figura 2), usamos a regra Modus Tollens para concluir que a célula (2,2) é segura.

A segunda forma é o **encadeamento regressivo**. Nesse método, começamos definindo uma meta ou proposição a ser provada. No caso do Mundo do Wumpus, a meta pode ser determinar se uma célula é segura para exploração.

Para avaliar a segurança, o motor de inferência segue estas etapas:

1. Identifica uma regra no formato:

Se  $(\text{premissa}_1 \wedge \text{premissa}_2 \wedge \dots \wedge \text{premissa}_n)$ , então (conclusão).

2. Examina cada premissa:

- **Se for um fato conhecido**, como "não há fedor em (1,1)", considera-a verdadeira.
- **Se não for conhecida**, a premissa torna-se uma nova meta. O motor busca regras que possam prová-la, verificando premissas até encontrar fatos conhecidos.

## 4 Problemas de Satisfação de Restrições

Motores de inferência também resolvem problemas como Sudoku (Figura 3), representando restrições declarativamente:

- Cada linha, coluna e subgrade deve conter números de 1 a 9 sem repetições.

O processo de solução envolve:

|   |   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|---|
| 5 | 3 |   |   | 7 |   |   |   |   |
| 6 |   |   | 1 | 9 | 5 |   |   |   |
|   | 9 | 8 |   |   |   |   | 6 |   |
| 8 |   |   |   | 6 |   |   |   | 3 |
| 4 |   |   | 8 |   | 3 |   |   | 1 |
| 7 |   |   |   | 2 |   |   |   | 6 |
|   | 6 |   |   |   |   | 2 | 8 |   |
|   |   |   | 4 | 1 | 9 |   |   | 5 |
|   |   |   |   | 8 |   |   | 7 | 9 |

Figura 3: Sudoku: o jogador deve preencher os espaços vazios respeitando as restrições.

1. **Atribuição:** Selecionar uma célula vazia e atribuir um valor.
2. **Checação de Consistência:** Verificar se a atribuição respeita as restrições.
3. **Retrocesso (Backtracking):** Retornar ao estado anterior em caso de inconsistência.

## 5 Exercícios

1. Encontre uma configuração do Mundo de Wumpus que não pode ser resolvida pelo agente lógico. Por que isso acontece?
2. Escreva predicados lógicos para gerar uma escala semanal para 10 enfermeiras, considerando:
  - Nenhuma enfermeira pode trabalhar em mais de um turno por dia.
  - Algumas enfermeiras podem estar de férias ou indisponíveis.
  - Enfermeiras não podem trabalhar em turnos noturnos seguidos por turnos diurnos.