



Inteligência Artificial

Professor: Elton Sarmanho¹

E-mail: eltonss@ufpa.br



¹Faculdade de Sistemas de Informação - UFPA/CUNTINS

7 de junho de 2025

Roteiro

Planejamento

Fundamentação

Conceitos Fundamentais

Contexto Histórico

Estado da Arte

Agentes Inteligentes

Conceitos Fundamentais

Racionalidade na Prática

Ambientes de Tarefa (PEAS)

Propriedades dos Ambientes de Tarefa

Estrutura de Agente e Programas

Tipos de Agentes Inteligentes



Licença

Este trabalho está licenciado sob a licença Creative Commons:



Planejamento

Nesta aula:

- ▶ Vamos explorar os conceitos fundamentais de Inteligência Artificial.
- ▶ Ter panorama sobre conceito.
- ▶ Códigos do livro estão: <https://github.com/aimacode>
- ▶ Códigos do Professor estão no github



└ Fundamentação

└ Conceitos Fundamentais

Conceitos Fundamentais



└ Fundamentação

└ Conceitos Fundamentais

Inteligência Artificial (IA)

- ▶ IA é o campo de estudo de agentes que percebem o ambiente e realizam ações para maximizar suas chances de sucesso.
 - ▶ Conjunto de técnicas para a construção de máquinas “inteligentes”, capazes de resolver problemas que requerem inteligência humana.
- ▶ IA abrange uma vasta gama de subcampos, incluindo aprendizado de máquina, percepção, raciocínio e ação.



└ Fundamentação

└ Conceitos Fundamentais

Abordagens para Definir IA

► Dimensões Principais:

- **Pensamento vs. Comportamento:** A IA pode ser definida em termos de como os sistemas pensam ou como eles agem.
- **Humanidade vs. Racionalidade:** A IA pode ser medida em relação ao desempenho humano ou em relação ao desempenho ideal.



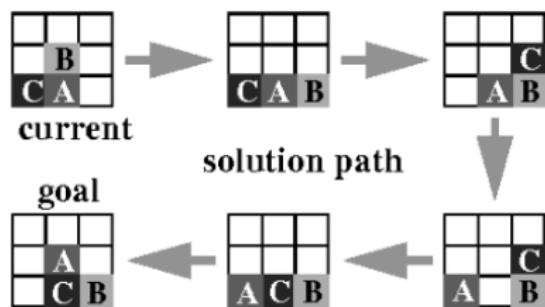
└ Fundamentação

└ Conceitos Fundamentais

Pensar Humanamente

► Abordagem Cognitiva:

- ▶ Envolve a modelagem dos processos cognitivos humanos.
- ▶ Métodos incluem introspecção, experimentos psicológicos e neuroimagem.
- ▶ Exemplos: Modelos baseados em como os humanos resolvem problemas, como o GPS (General Problem Solver).



└ Fundamentação

└ Conceitos Fundamentais

Pensar Racionalmente

► Leis do Pensamento:

- ▶ Baseia-se na lógica formal e em padrões de raciocínio que são corretos.
- ▶ Exemplos: Silogismos de Aristóteles, que fornecem regras para deduzir conclusões lógicas.
- ▶ Limitações: Difícil expressar conhecimento informal e enfrentar problemas práticos devido à complexidade.



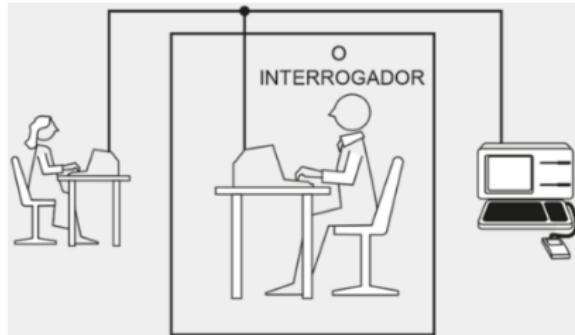
└ Fundamentação

└ Conceitos Fundamentais

Agir Humanamente

► Teste de Turing:

- Proposto por Alan Turing em 1950 como um teste operacional de inteligência.
- Um sistema é considerado inteligente se seu comportamento não puder ser distinguido de um humano.
- Requisitos: Processamento de linguagem natural, representação de conhecimento, raciocínio automatizado, aprendizado de máquina.



└ Fundamentação

└ Conceitos Fundamentais

Agir Racionalmente

► Agente Racional:

- ▶ Um agente racional age de forma a maximizar o sucesso ou a expectativa de sucesso, dado o que sabe.
- ▶ Vantagens: Mais geral e aplicável do que simplesmente seguir leis do pensamento.
- ▶ Foco: Construir agentes que tomem decisões racionais, mesmo quando a solução ideal não é computável em tempo hábil.



└ Fundamentação

└ Conceitos Fundamentais

Resumo das Abordagens

► Comparação das Abordagens:

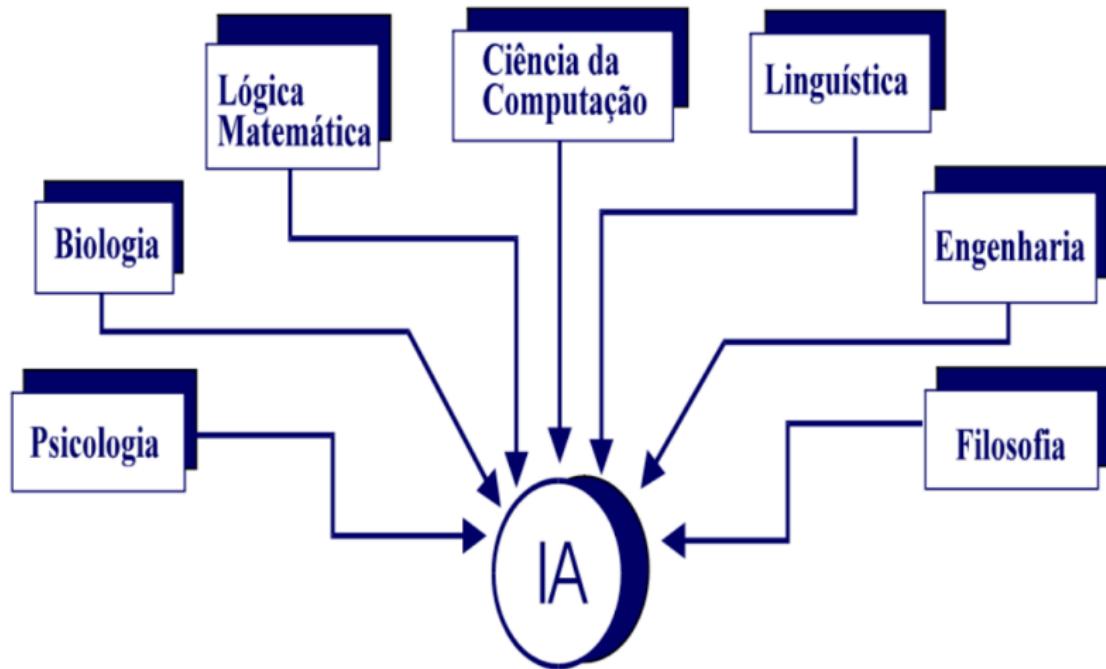
- ▶ **Pensar Humanamente:** Modelagem da mente humana.
- ▶ **Pensar Racionalmente:** Uso de lógica e raciocínio formal.
- ▶ **Agir Humanamente:** Comportamento que imita humanos (Teste de Turing).
- ▶ **Agir Racionalmente:** Ação baseada em raciocínio racional, independente da comparação com humanos.



└ Fundamentação

└ Conceitos Fundamentais

Áreas de Apoio para IA



└ Fundamentação

└ Conceitos Fundamentais

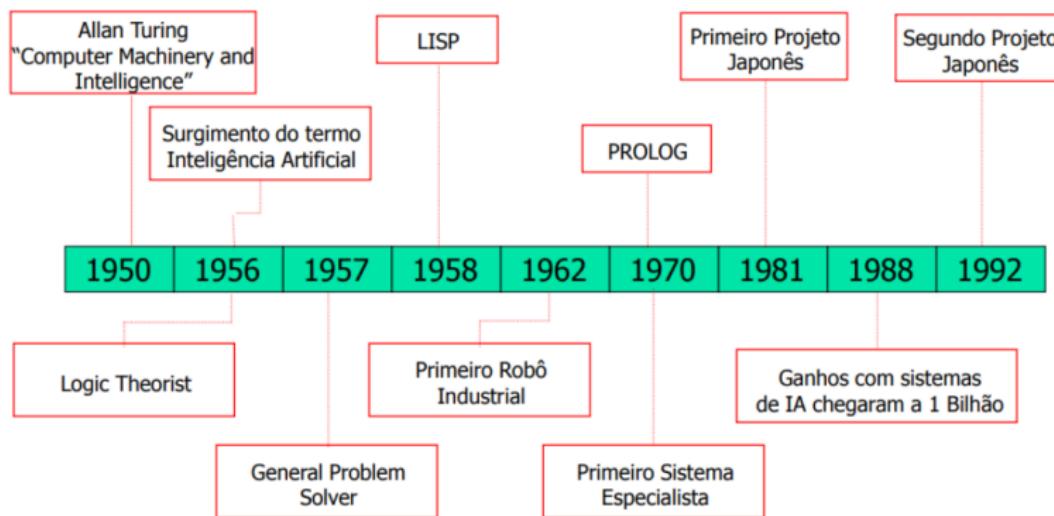
Sub-Áreas da IA



Fundamentação

Contexto Histórico

Linha do Tempo



└ Fundamentação

└ Contexto Histórico

Gestação da IA (1943–1955)

- ▶ Primeiro trabalho reconhecido como IA por Warren McCulloch e Walter Pitts (1943).
- ▶ Modelo de neurônios artificiais inspirado na fisiologia básica do cérebro.
- ▶ Donald Hebb (1949) propõe a regra de aprendizado hebbiana, ainda influente hoje.
- ▶ Marvin Minsky e Dean Edmonds constroem o primeiro computador de rede neural, o SNARC, em 1950.



└ Fundamentação

└ Contexto Histórico

Nascimento da IA (1956)

- ▶ John McCarthy organiza um workshop no Dartmouth College, considerado o nascimento oficial da IA.
- ▶ Proposta de estudo sobre como construir máquinas que simulam aspectos da inteligência.
- ▶ Participantes incluem Minsky, Shannon, Rochester e outros influentes na área.
- ▶ Criação do termo "Inteligência Artificial" por McCarthy.



└ Fundamentação

└ Contexto Histórico

Entusiasmo Inicial e Grandes Expectativas (1952–1969)

- ▶ Sucessos limitados, mas significativos, como o Logic Theorist de Newell e Simon.
- ▶ Desenvolvimentos como o General Problem Solver (GPS), que imitava protocolos de resolução de problemas humanos.
- ▶ Surgimento da hipótese do sistema de símbolos físicos como base para ação inteligente.
- ▶ Primeiros programas de IA para jogos e provas de teoremas, como o Theorem Prover de Gelernter e os programas de damas de Arthur Samuel.



└ Fundamentação

└ Contexto Histórico

O Retorno das Redes Neurais (1986–presente)

- ▶ Re-invenção do algoritmo de retropropagação no meio dos anos 1980.
- ▶ Confronto entre modelos conexionistas e simbólicos.
- ▶ Convergência de abordagens conexionistas e simbólicas, com bifurcação em duas áreas: arquiteturas eficazes e modelagem empírica.



└ Fundamentação

└ Contexto Histórico

Adoção do Método Científico (1987–presente)

- ▶ Revolução no conteúdo e na metodologia da IA.
- ▶ Abandono do isolacionismo da IA e integração com outros campos como teoria da informação e modelagem estocástica.
- ▶ Importância crescente do método científico, rigor experimental e relevância para aplicações do mundo real.



└ Fundamentação
└ Estado da Arte

Aplicações

- ▶ Apresenta exemplos de aplicações que mostram como a IA está sendo usada no mundo real.



Veículos Robóticos

- ▶ Exemplo: **STANLEY** - Carro robótico que venceu o DARPA Grand Challenge em 2005.
- ▶ Equipado com câmeras, radar, e laser para percepção do ambiente.
- ▶ Aplicações: Condução autônoma em terrenos difíceis.

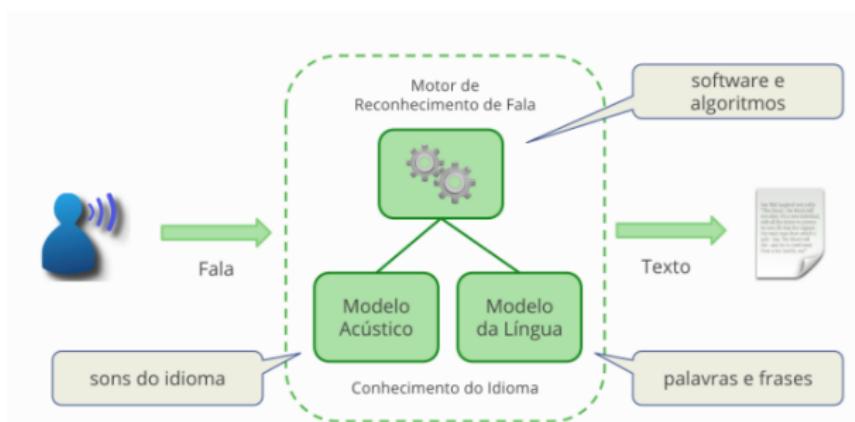


└ Fundamentação

└ Estado da Arte

Reconhecimento de Fala

- ▶ Exemplo: Sistema de reconhecimento de fala automatizado utilizado pela United Airlines.
- ▶ Aplicações: Atendimento ao cliente e interação homem-máquina.
- ▶ Desafios: Compreensão de diferentes sotaques e padrões de fala.

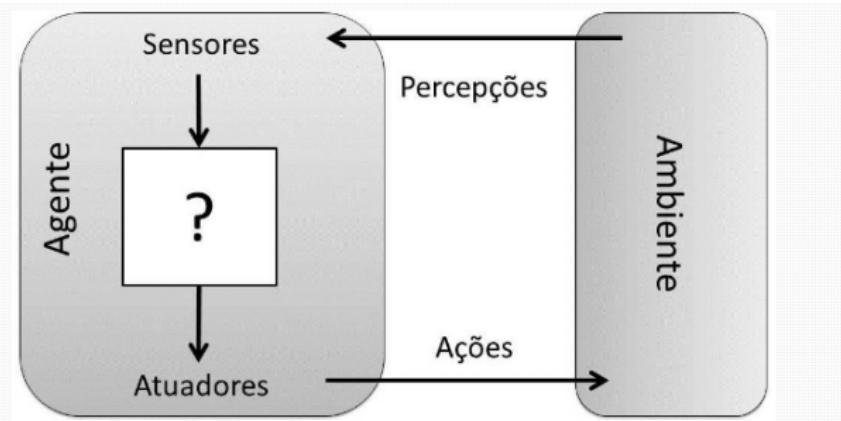


└ Fundamentação

└ Estado da Arte

Planejamento e Agendamento Autônomo

- ▶ Exemplo: **Remote Agent** da NASA - Primeiro programa autônomo a controlar as operações de uma espaçonave.
- ▶ Funções: Geração de planos, monitoramento e correção de falhas.
- ▶ Sucessor: **MAPGEN** - Planejamento diário das operações dos Rovers em Marte.



Jogos

- ▶ Exemplo: **DEEP BLUE** - Computador da IBM que derrotou Garry Kasparov no xadrez em 1997.
- ▶ Impacto: Demonstração do poder da IA em resolver problemas complexos e de alto desempenho.



Luta Contra Spam

- ▶ Uso de algoritmos de aprendizado de máquina para classificar mensagens como spam.
- ▶ Importância: Protege os usuários de e-mails indesejados, que podem representar até 90% do volume total.
- ▶ Desafio contínuo: Adaptação às novas táticas dos spammers.



Planejamento Logístico

- ▶ Exemplo: **DART** - Ferramenta utilizada pelas forças armadas dos EUA durante a crise do Golfo Pérsico de 1991.
- ▶ Funções: Planejamento e agendamento automatizado de logística, envolvendo milhares de veículos e pessoas.
- ▶ Benefícios: Redução significativa no tempo necessário para gerar planos logísticos complexos.



Modelos de Linguagem de Grande Escala (LLM)

- ▶ Exemplo: **Gemini** e **GPT-4** - Modelos de linguagem treinados em grandes volumes de texto.
- ▶ Aplicações: Assistentes virtuais, geração de texto, tradução automática e criação de conteúdo.
- ▶ Capacidade: Entendimento e geração de linguagem natural com alto grau de coerência e relevância.
- ▶ Desafios: Preocupações éticas, vieses nos dados de treinamento e uso responsável.



Conclusão

- ▶ A IA já está integrada em muitas áreas da sociedade, desde veículos autônomos até sistemas de atendimento ao cliente.
- ▶ Os exemplos apresentados mostram o impacto significativo que a IA já tem em nosso dia a dia.



└ Agentes Inteligentes

└ Conceitos Fundamentais

Agentes Inteligentes



└ Agentes Inteligentes

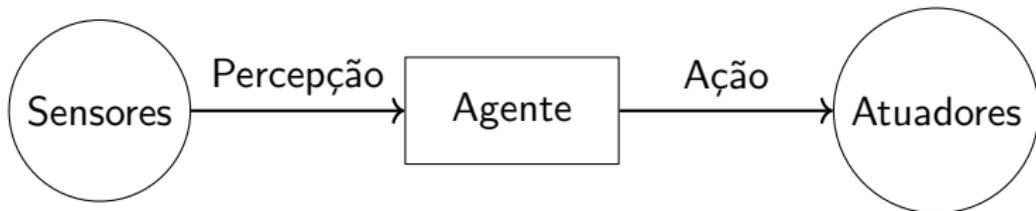
└ Conceitos Fundamentais

Definição de Agente e Comportamento

- ▶ **Agente:** Uma entidade que percebe seu ambiente através de **sensores** e age sobre ele por meio de **atuadores**.
- ▶ O **comportamento** de um agente é definido por uma **função de agente** que mapeia sequências de percepções para ações.

Percepção → Ação

$$\{s_1, s_2, \dots, s_n\} \rightarrow f \rightarrow \{a_1, a_2, \dots, a_m\}$$



└ Agentes Inteligentes

└ Conceitos Fundamentais

Agente Inteligente e Racionalidade

Um **agente inteligente** é um agente que toma decisões autônomas para atingir objetivos e utiliza raciocínio e aprendizado para melhorar seu desempenho.

Racionalidade

Um agente é **racional** se ele escolhe a ação que maximiza sua **medida de desempenho esperada** com base nas evidências (percepções) e no conhecimento disponível.

$$\text{Ação ótima} = \arg \max_{a \in A} \sum_{s'} P(s' | s, a) \cdot U(s')$$

onde:

- ▶ s é o estado atual.
- ▶ s' é o estado futuro.
- ▶ $P(s' | s, a)$ é a probabilidade de transição para s' dada a ação a no estado s .
- ▶ $U(s')$ é a utilidade (valor) do estado s' .



└ Agentes Inteligentes

└ Conceitos Fundamentais

Autonomia e Aprendizado

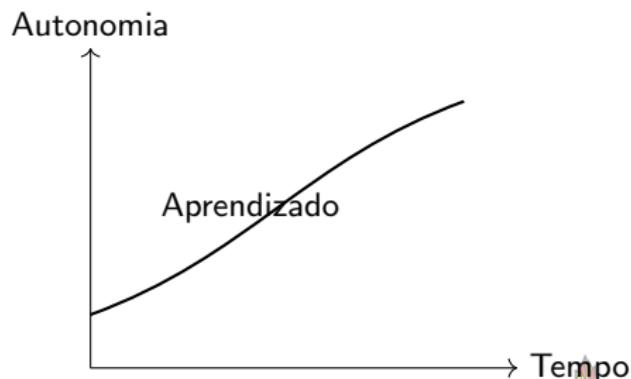
Um agente é **autônomo** se opera sem intervenção humana direta e se adapta a mudanças no ambiente. A autonomia de um agente pode ser melhorada através do aprendizado com suas percepções e experiências.

Autonomia Modelada

Seja E o ambiente e K o conhecimento do agente. A autonomia pode ser modelada como:

$$\text{Ação autônoma} = f(E, K)$$

onde f é a função de decisão do agente.



└ Agentes Inteligentes

└ Racionalidade na Prática

Fatores da Racionalidade e Medida de Desempenho

Um agente racional escolhe ações para maximizar sua medida de desempenho esperada. Essa escolha é influenciada por quatro fatores principais:

- ▶ **Medida de Desempenho:** Define o critério de sucesso. É crucial projetá-la de acordo com o resultado desejado no ambiente, não o comportamento esperado.
- ▶ **Conhecimento Prévio:** Informações que o agente possui sobre o ambiente (e.g., modelos, regras).
- ▶ **Ações Possíveis:** O conjunto de ações que o agente pode realizar.
- ▶ **Sequência de Percepções:** O histórico de percepções que o agente utiliza para atualizar seu conhecimento e refinar decisões.

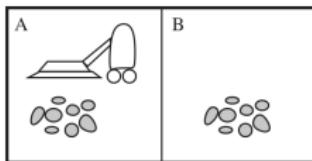


Figura: Aspirador de Pó. Medida de desempenho pode ser nº de quadrados limpos em um período

└ Agentes Inteligentes

 └ Ambientes de Tarefa (PEAS)

Definição e Componentes PEAS

O **ambiente de tarefa** de um agente abrange todos os aspectos externos relevantes para o problema a ser resolvido. A primeira etapa no design de um agente é especificar o ambiente usando o framework **PEAS**:

- ▶ **Performance Measure** (Medida de Desempenho): Critério de sucesso do agente.
- ▶ **Environment** (Ambiente): Onde o agente opera.
- ▶ **Actuators** (Atuadores): Mecanismos para o agente agir no ambiente.
- ▶ **Sensors** (Sensores): Dispositivos para o agente perceber o ambiente.



Agentes Inteligentes

Ambientes de Tarefa (PEAS)

Definição e Componentes PEAS

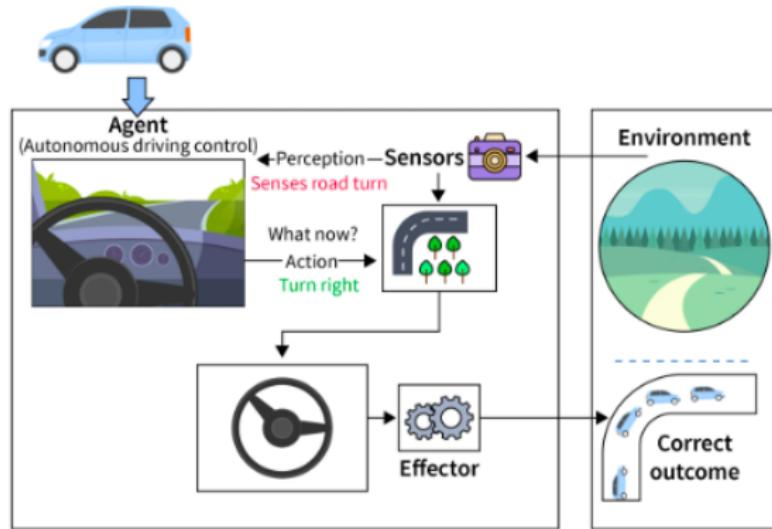


Figura: Diagrama de um agente interagindo com seu ambiente via sensores e atuadores.



└ Agentes Inteligentes

└ Ambientes de Tarefa (PEAS)

Exemplos de PEAS

Agente	P	E	A	S
Aspirador de Pó	Limpeza	Casa	Motores Sucção	Sensores de sujeira
Condução Autônoma	Segurança	Ruas	Volante Freios	Câmeras, radar
Diagnóstico Médico	Precisão	Clínicas	Recomendações	Entrada de dados
Robô de Entrega de Pacotes	Pontualidade	Obstáculos	Rodas Braço robótico	GPS Câmeras Sensores

Tabela: Exemplos de PEAS para diferentes tipos de agentes.



└ Agentes Inteligentes

└ Ambientes de Tarefa (PEAS)

Exercício: Robô de Entrega de Pacotes na UFPA de Cametá

Imagine que você está desenvolvendo um robô autônomo capaz de entregar pacotes em um campus universitário. O robô deve navegar pelos prédios, evitar obstáculos, e garantir que os pacotes sejam entregues no local certo.

- ▶ **Medida de Desempenho:** Como você definiria a medida de desempenho? Quais métricas usaria para avaliar o sucesso?
- ▶ **Ambiente:** Descreva o ambiente. Quais elementos o robô precisará considerar?
- ▶ **Atuadores:** Quais atuadores o robô precisaria?
- ▶ **Sensores:** Quais sensores o robô precisaria para perceber o ambiente e tomar decisões informadas?



└ Agentes Inteligentes

 └ Propriedades dos Ambientes de Tarefa

Classificação de Ambientes

Compreender as propriedades do ambiente de tarefa é fundamental para projetar agentes racionais, pois elas influenciam diretamente a complexidade do agente e os algoritmos necessários.

- ▶ **Totalmente Observável vs. Parcialmente Observável:** Um ambiente é totalmente observável se o agente tem acesso a todas as informações relevantes. Caso contrário, é parcialmente observável. Ex: Xadrez (totalmente) vs. Condução em neblina (parcialmente).
- ▶ **Determinístico vs. Estocástico:** No determinístico, o próximo estado é totalmente previsível. No estocástico, há incerteza e probabilidades. Ex: Xadrez (determinístico) vs. Previsão do tempo (estocástico).
- ▶ **Episódico vs. Sequencial:** Em ambientes episódicos, cada decisão é independente. Em sequenciais, as ações passadas afetam decisões futuras. Ex: Classificação de imagens (episódico) vs. Dirigir um carro (sequencial).



└ Agentes Inteligentes

└ Propriedades dos Ambientes de Tarefa

Mais Propriedades dos Ambientes

- ▶ **Estático vs. Dinâmico:** Ambientes estáticos não mudam durante a deliberação do agente. Dinâmicos podem mudar. Ex: Quebra-cabeça (estático) vs. Jogar futebol (dinâmico).
- ▶ **Discreto vs. Contínuo:** Ambientes discretos possuem um número finito de estados/ações. Contínuos têm infinitas possibilidades. Ex: Jogo de tabuleiro (discreto) vs. Controle de robô móvel (contínuo).
- ▶ **Agente Único vs. Multiagente:** Agente único opera sozinho. Multiagente envolve interação com outros agentes (cooperativos ou competitivos). Ex: Quebra-cabeça (único) vs. Xadrez (multiagente).
- ▶ **Conhecidos vs. Desconhecidos:** Agentes em ambientes conhecidos entendem as regras. Em desconhecidos, precisam explorar para descobrir.



└ Agentes Inteligentes

 └ Propriedades dos Ambientes de Tarefa

Impacto no Design de Agentes

- ▶ A combinação dessas propriedades define a complexidade do ambiente e, consequentemente, a complexidade do agente necessário.
- ▶ O caso mais desafiador é um ambiente **parcialmente observável, multiagente, estocástico, sequencial, dinâmico, contínuo e desconhecido**.
- ▶ **Exemplo:** Dirigir em um país desconhecido, com geografia e leis de trânsito desconhecidas.



└ Agentes Inteligentes

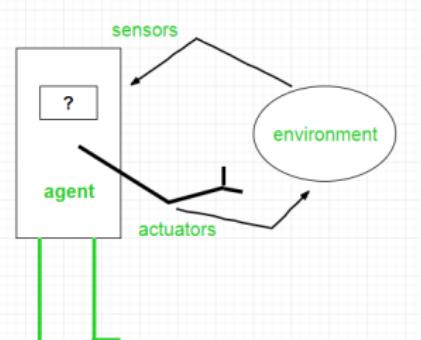
└ Estrutura de Agente e Programas

Estrutura do Agente: Arquitetura e Programa

Um agente é a combinação de sua **arquitetura** (dispositivo físico com sensores e atuadores) e seu **programa de agente** (o algoritmo que controla suas ações).

- ▶ **Arquitetura:** Capta percepções, executa ações. Deve ser compatível com as ações do programa (e.g., pernas para "andar").
- ▶ **Programa de Agente:** O algoritmo que mapeia percepções para ações e, junto com o ambiente, define o comportamento do agente.

$$\text{Agente} = \text{Arquitetura} + \text{Programa}$$



└ Agentes Inteligentes

└ Estrutura de Agente e Programas

Programa de Agente: Exemplo Simples

O programa de agente recebe percepções e retorna uma ação.

```
def program(percepts):
    '''Retorna uma acao baseada na percepcao do
    dog'''
    for p in percepts:
        if isinstance(p, Food):
            return 'come'
        elif isinstance(p, Water):
            return 'bebe'
    return 'fica deitado'
```

Listing: Exemplo de Função de Agente Simples

Este exemplo demonstra uma lógica reativa simples: o agente "reage" diretamente às percepções de "comida" ou "água".



└ Agentes Inteligentes

 └ Tipos de Agentes Inteligentes

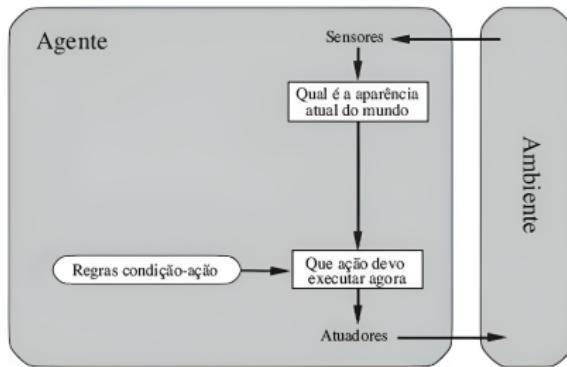
Agentes Simples de Reflexo

- ▶ Escolhem ações com base apenas na **percepção atual**, ignorando o histórico.
- ▶ **Adequados** para ambientes **simples** e **totalmente observáveis**.
- ▶ **Limitação:** Não funcionam bem em ambientes parcialmente observáveis. Se as percepções forem removidas ou limitadas, podem precisar de ações aleatórias.



Agentes Inteligentes

Tipos de Agentes Inteligentes



Exemplo de Código (Aspirador)

```

function REFLEX-AGENT([location, status]) returns an action
  if status = SUJO then return LIMPAR
  else if location = A then return DIREITA
  else if location = B then return ESQUERDA
  
```

Este programa reage diretamente ao status de sujeira ou à localização para decidir a próxima ação.

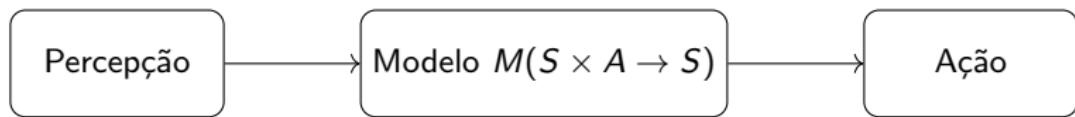


└ Agentes Inteligentes

└ Tipos de Agentes Inteligentes

Agentes Baseados em Modelo

- ▶ Mantêm um **estado interno** que representa informações sobre o ambiente, mesmo quando a percepção atual é insuficiente.
- ▶ Utilizam um **modelo do ambiente** para prever como o mundo evolui e o que suas ações fazem.
- ▶ São mais **robustos** em ambientes parcialmente observáveis.



```

function MODEL-BASED-REFLEX-AGENT(percept) returns action
  PERSISTENT: STATE, MODEL, RULES, ACTION
  STATE ← UPDATE-STATE(STATE, ACTION, percept, MODEL)
  rule ← RULE-MATCH(STATE, RULES)
  action ← rule.ACTION
  return action
  
```

O agente atualiza seu estado interno e usa regras baseadas nesse modelo para decidir a ação.



└ Agentes Inteligentes

└ Tipos de Agentes Inteligentes

Agentes Baseados em Objetivos

- ▶ Escolhem ações que levam à realização de **objetivos específicos**, considerando o resultado desejado.
- ▶ Têm capacidade de **planejar** e prever o resultado de ações futuras.
- ▶ **Exemplo:** Um robô de entrega deve planejar a melhor rota para um local específico, em vez de apenas reagir ao ambiente imediato.



└ Agentes Inteligentes

└ Tipos de Agentes Inteligentes

Agentes Baseados em Objetivos

Algoritmo de Busca (BFS)

Agentes baseados em objetivos frequentemente usam algoritmos de busca, como BFS, para encontrar um caminho que satisfaça o objetivo.

```
function BFS(start, goal) returns a path
    frontier = [start]    // fila de NODOS a serem explorados
    explored = {}         // conjunto de NODOS explorados
    while frontier is not empty:
        node = frontier.pop()
        if node == goal then return solution
        for each neighbor in node.neighbors:
            if neighbor not in explored:
                frontier.push(neighbor)
                explored.add(neighbor)
```



└ Agentes Inteligentes

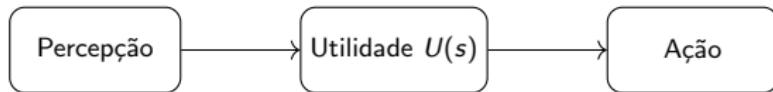
└ Tipos de Agentes Inteligentes

Agentes Baseados em Utilidade e Aprendizado

- **Agentes Baseados em Utilidade:** Maximizam uma função de utilidade $U(s)$ que mede a satisfação com os resultados.

$$U(s) = \sum_i w_i \cdot f_i(s)$$

onde w_i são pesos e $f_i(s)$ características do estado.



- **Agentes de Aprendizado:** Aprendem com a experiência para melhorar o desempenho ao longo do tempo.

$$Q(s, a) \leftarrow Q(s, a) + \alpha \left[r + \gamma \max_{a'} Q(s', a') - Q(s, a) \right]$$

Esta é a equação de atualização Q-Learning, uma função de aprendizado onde $Q(s, a)$ é a função de valor, α a taxa de aprendizado, γ o fator de desconto e r a recompensa.



└ Implementação

Codificar Agentes



└ Implementação

 └ Implementação de um agente

Bibliotecas Populares para Agentes Inteligentes em Python

► AIMA-Python

- ▶ Implementações de algoritmos e agentes do livro *Artificial Intelligence: A Modern Approach*.
- ▶ Link: <https://github.com/aimacode/aima-python>

► MESA

- ▶ Framework para simulação de agentes em ambientes 2D, útil para modelos baseados em agentes.
- ▶ Link: <https://mesa.readthedocs.io/en/stable/>

► PyAgents

- ▶ Biblioteca focada na criação e simulação de agentes inteligentes.
- ▶ Link: <https://github.com/emanuelen5/pyagents>



└ Implementação

 └ Implementação de um agente

Bibliotecas Populares para Agentes Inteligentes em Python

- ▶ **SPADE (Smart Python Agent Development Environment)**
 - ▶ Plataforma para desenvolvimento de agentes inteligentes baseados no padrão FIPA.
 - ▶ Link: <https://github.com/javipalanca/spade>
- ▶ **FLARE (Framework for Language-Agnostic Reinforcement Learning)**
 - ▶ Framework focado em aprendizado por reforço, útil para desenvolvimento de agentes que aprendem.
 - ▶ Link: <https://github.com/facebookresearch/flare>
- ▶ **Gym (OpenAI)**
 - ▶ Biblioteca de simulações e ambientes para desenvolvimento e treinamento de agentes com aprendizado por reforço.
 - ▶ Link: <https://gym.openai.com/>



└ Implementação

 └ Implementação de um agente

Codificação de um Agente

- ▶ Python
- ▶ Códigos estão disponíveis no github.



Referências I

- 📄 Chollet, F. (2021). Deep Learning with Python, Second Edition. Shelter Island, NY: Manning Publications.
- 📄 Géron, A. (2019). Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow. Sebastopol, CA: O'Reilly Media..
- 📄 Russell, S., & Norvig, P. (2021). Artificial Intelligence: A Modern Approach (4^a ed.). Hoboken, NJ: Pearson.
- 📄 Moroney, L. (2020). AI and Machine Learning for Coders. Sebastopol, CA: O'Reilly Media.





Inteligência Artificial

Professor: Elton Sarmanho¹
E-mail: eltonss@ufpa.br



¹Faculdade de Sistemas de Informação - UFPA/CUNTINS

7 de junho de 2025