



**Faculdade de Engenharia**  
**Departamento de Electrotecnia**  
**Licenciatura em Engenharia Informática**  
**Inteligência Artificial I**

**Tema: Agente**

<b>Estudante</b>	Lino, Miro Pedro Tipaneque
<b>Docentes</b>	Ruben Manhica e Roxan Cadir

**2.1 Conceitos de Agente**

**i. Agente:**

Um agente em IA é uma entidade que percebe o ambiente através de sensores e age sobre ele usando actuadores. Um agente pode ser um software (como um chatbot) ou um sistema físico (como um robô). O comportamento de um agente é governado por seu programa de agente.

**ii. Função Agente:**

A função de agente é uma descrição matemática do comportamento do agente. Ela mapeia cada sequência de percepções que o agente experimenta para uma ação. Essencialmente, é o que define como o agente decide o que fazer com base no que percebeu.

**Programa de Agente:**

O programa de agente é a implementação concreta da função de agente. Ele determina o comportamento do agente ao processar as percepções e escolher as acções. Em termos práticos, é o código ou algoritmo que executa a função de agente em um determinado ambiente.

### **iii. Racionalidade:**

Racionalidade em IA refere-se à capacidade do agente de tomar acções que maximizam o sucesso esperado, com base nas percepções e no conhecimento que ele tem do ambiente. Um agente racional age de acordo com a sua função de desempenho, tentando obter o melhor resultado possível.

### **iv. Autonomia:**

Autonomia é o grau em que um agente opera de forma independente de qualquer intervenção externa. Um agente com alta autonomia toma decisões e realiza acções baseadas principalmente em suas próprias percepções, enquanto um agente com baixa autonomia pode depender de instruções contínuas de um ser humano ou de outro sistema.

### **v. Agente Reactivo:**

Um agente reactivo é um tipo de agente que toma decisões de forma directa e simples, com base em regras predefinidas, reagindo ao ambiente de forma imediata. Ele não mantém um histórico detalhado do passado nem planeia para o futuro distante; sua acção é uma resposta directa às percepções atuais. Esses agentes são eficazes em ambientes dinâmicos e imprevisíveis.

### **vi. Agente Baseado em Modelo:**

Um agente baseado em modelo mantém um modelo interno do mundo, ou seja, ele armazena informações sobre o estado actual do ambiente, bem como sobre as consequências de suas ações. Esse modelo permite que o agente tome decisões mais informadas, levando em consideração como o mundo funciona. Esse tipo de agente pode prever o efeito de suas ações futuras, e assim, planejar melhor suas ações.

### **vii. Agente Baseado em Objectivos:**

Um agente baseado em objectivos toma decisões com base em metas que ele precisa alcançar. Além de ter um modelo do mundo, ele é capaz de escolher ações que levam a estados que atendem a seus objectivos. O agente seleciona suas acções com o objetivo de

reduzir a distância entre o estado atual e o estado desejado. Esse tipo de agente não apenas reage ao ambiente, mas também busca activamente um estado final específico.

#### **viii. Agente Baseado em Utilidade:**

Um agente baseado em utilidade vai além de apenas buscar objectivos. Ele atribui uma "utilidade" (um valor de satisfação) a diferentes estados possíveis e tenta maximizar esse valor. Isso significa que ele não apenas tenta atingir um objectivo, mas também faz escolhas entre diferentes caminhos que levam a esse objetivo, considerando o quão "útil" cada caminho é. Em outras palavras, ele escolhe o curso de acção que oferece a maior satisfação ou benefício, levando em consideração riscos, custos e recompensas.

#### **ix. Agente com Aprendizagem:**

Um agente com aprendizagem é capaz de melhorar seu desempenho ao longo do tempo, adquirindo novos conhecimentos a partir da experiência. Esse tipo de agente é capaz de adaptar seu comportamento com base em feedback recebido do ambiente ou de outros agentes. Ele aprende a identificar padrões, aprimora sua função de agente e se ajusta às mudanças no ambiente para tomar melhores decisões no futuro. Agentes com aprendizagem são úteis em ambientes dinâmicos onde a informação é incompleta ou em constante mudança.

## **2.5 PEAS**

### **a. Robô Jogador de Futebol**

#### **➤ Performance (Medida de Desempenho):**

- Marcar gols.
- Defender o golo (para um guarda-redes).
- Colaborar com outros jogadores da equipe.
- Cumprir as regras do jogo (sem cometer faltas).
- Minimizar o consumo de energia.

#### **➤ Environment (Ambiente):**

- Campo de futebol (com marcações, linhas, e golos).
- Outros jogadores (aliados e adversários).
- Bola.
- Público e árbitros.
- Condições climáticas (interno ou externo).

➤ **Actuators (Actuadores):**

- Motores para movimentação (pernas mecânicas, rodas ou outros mecanismos de locomoção).
- Atuadores para chutar ou passar a bola.
- Mecanismos de equilíbrio e controle de postura.

➤ **Sensors (Sensores):**

- Câmeras para visão (para detectar a bola, outros jogadores e o campo).
- Sensores de proximidade para evitar colisões.
- Sensores de toque para sentir a bola.
- Giroscópios e acelerômetros para equilíbrio e orientação.

**b. Agente Catálogo de Compras da Internet**

➤ **Performance (Medida de Desempenho):**

- Aumentar o número de vendas.
- Recomendar produtos relevantes ao usuário.
- Minimizar o tempo de busca para o usuário.
- Garantir uma boa experiência de usuário.

- Maximizar a taxa de conversão (número de compras em relação ao número de visitas).

➤ **Environment (Ambiente):**

- Loja online (plataforma de e-commerce).
- Banco de dados de produtos.
- Perfis e preferências de usuários.
- Reviews e avaliações de produtos.
- Concorrentes (outros sites de compras).

➤ **Actuators (Atuadores):**

- Interface de recomendação (exibir produtos sugeridos).
- Filtros de busca (filtrar produtos de acordo com preferências).
- Sistema de notificação (enviar alertas de promoções ou novos produtos).

➤ **Sensors (Sensores):**

- Histórico de compras dos usuários.
- Dados de navegação (páginas visitadas, tempo gasto).
- Feedback dos usuários (avaliações e reviews).
- Dados de pesquisa (produtos procurados).

**c. Andarilho Autônomo de Marte**

➤ **Performance (Medida de Desempenho):**

- Coletar amostras de solo e rochas.

- Navegar por terrenos acidentados de Marte.
- Evitar obstáculos e perigos (como penhascos ou dunas).
- Minimizar o consumo de energia e otimizar o uso de recursos.
- Transmitir dados de volta à Terra com precisão.

➤ **Environment (Ambiente):**

- Superfície de Marte (terreno rochoso, dunas, crateras).
- Atmosfera de Marte (temperaturas extremas, baixa pressão).
- Radiação solar e poeira.
- Localização remota (comunicação a longa distância com a Terra).

➤ **Actuators (Atuadores):**

- Motores para locomoção (rodas, pernas mecânicas).
- Braços robóticos para coleta de amostras.
- Mecanismos de perfuração ou escavação.
- Sistemas de comunicação para transmitir dados.

➤ **Sensors (Sensores):**

- Câmeras e sensores de imagem para navegação.
- Sensores de proximidade e distância para evitar obstáculos.
- Sensores de temperatura, pressão e radiação.
- Instrumentos de análise química e geológica para examinar amostras.

**d. Assistente de Matemático para Demonstração de Teoremas**

➤ **Performance (Medida de Desempenho):**

- Encontrar e demonstrar teoremas de forma correta.
- Otimizar o tempo de resolução e prova.
- Propor abordagens alternativas para a demonstração de teoremas.
- Aprender com exemplos e generalizar para novos problemas.
- Minimizar o número de erros e inconsistências nas provas.

➤ **Environment (Ambiente):**

- Conjunto de axiomas e definições matemáticas.
- Regras e teorias matemáticas.
- Problemas propostos (teoremas a serem demonstrados).
- Interações com o matemático (feedback e revisões).

➤ **Actuators (Atuadores):**

- Interface de escrita (gerar textos com demonstrações).
- Ferramentas de sugestão de estratégias de prova (como indução, contradição, etc.).
- Visualização de fórmulas e diagramas matemáticos.

➤ **Sensors (Sensores):**

- Entrada de problemas e feedback do matemático.
- Análise de demonstrações anteriores (banco de dados de teoremas resolvidos).
- Sensores de validação lógica para verificar a consistência das provas.
- Reconhecimento de padrões matemáticos para facilitar a formulação de hipóteses.

## 2.7. Simulador de ambiente de medição de desempenho para o mundo de aspirador de pó

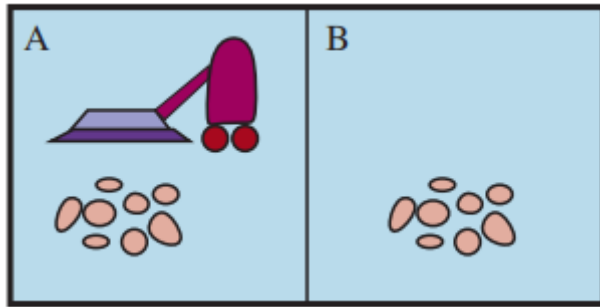


Figure 1. Vacuum-Cleaner World

Função Inicializar():

```
status <- {  
  loc_A: EscolherAleatoriamente(['Limp', 'Sujo']),  
  loc_B: EscolherAleatoriamente(['Limp', 'Sujo'])  
}
```

Função Percepcao(agente):

```
Retorna (agente.localizacao, status[agente.localizacao])
```

Função ExecutarAcao(agente, acao):

Se acao for 'Direita':

```
agente.localizacao <- loc_B
```

```
agente.desempenho <- agente.desempenho - 1
```



Senão, se ação for 'Esquerda':

```
agente.localização <- loc_A
```

```
agente.desempenho <- agente.desempenho - 1
```

Senão, se ação for 'Aspirar':

Se status[agente.localização] for igual a 'Sujo':

```
agente.desempenho <- agente.desempenho + 10
```

```
status[agente.localização] <- 'Limp'
```

Função LocalizaçãoPadrão(coisa):

```
Retorna EscolherAleatoriamente([loc_A, loc_B])
```

**2.10.** Considere uma versão modificada do ambiente de aspirador de pó do exercício 2.7, na qual a geografia do ambiente – extensão, limites e obstáculos – é desconhecida, como também a configuração inicial de sujeira. (O agente também pode se mover Acima e Abaixo, além de Esquerda e Direita.).

a. Um agente reativo simples pode ser perfeitamente racional para esse ambiente? Explique.

**R.:** Um agente reativo simples pode ser perfeitamente racional para esse ambiente se as regras de decisão que ele segue são adequadas para alcançar o objetivo desejado, que é limpar o ambiente. Se o agente tiver uma regra como "aspirar quando estiver sujo e mover-se aleatoriamente quando estiver limpo", ele pode eventualmente limpar o ambiente todo. Contudo, se o ambiente for grande ou tiver uma configuração específica, como um layout complicado, o agente pode demorar muito para alcançar uma solução ótima ou pode nunca conseguir limpar tudo. Assim, a racionalidade depende da eficácia das regras em todas as configurações possíveis do ambiente.

b. Um agente reativo simples com uma função de agente aleatório pode superar um agente reativo simples? Projete tal agente e faça a medição de seu desempenho em vários ambientes.

**R.:** Um agente reativo simples que incorpora alguma aleatoriedade nos seus movimentos pode superar um agente reativo simples dependendo da configuração do ambiente. Por exemplo, se o agente reativo simples segue uma regra fixa, ele pode ficar preso em ciclos em determinadas configurações do ambiente. Ao introduzir aleatoriedade, o agente pode escapar desses ciclos e, assim, cobrir mais áreas do ambiente. Para projetar tal agente, a estratégia pode ser "aspire quando sujo e mova-se aleatoriamente para uma célula adjacente quando o local estiver limpo".

c. Você poderia projetar um ambiente no qual seu agente aleatório terá um desempenho muito ruim? Mostre seus resultados.

**R.:** Sim, podemos projetar um ambiente onde o agente aleatório terá um desempenho ruim, por exemplo, criando um ambiente com várias paredes e compartimentos isolados. Em tal ambiente, o agente aleatório pode passar muito tempo tentando sair de um compartimento, enquanto outras áreas permanecem sujas. O desempenho pode ser medido em termos de tempo necessário para limpar completamente o ambiente e a quantidade de energia gasta.

d. Um agente reativo com estado pode superar um agente reativo simples? Projete tal agente e faça a medição de seu desempenho em vários ambientes. Você pode projetar um agente racional desse tipo?

**R.:** Sim, um agente reativo com estado pode superar um agente reativo simples, porque ele pode "lembrar" onde já esteve e onde ainda precisa limpar. Por exemplo, um agente pode armazenar um mapa do ambiente à medida que se move, marcando áreas limpas e sujas, e então priorizar a limpeza das áreas sujas. Para projetar tal agente, poderíamos ter uma memória que grava os locais já limpos e as posições atuais do agente. Isso evita que o agente repita movimentos desnecessários, melhorando sua eficiência.