

Sistemas Multi-agentes

Projeto dos agentes

- ★ O problema é resolvido por um conjunto de agentes, fisicamente distribuídos em diversas máquinas conectadas.
- ★ Os agentes são concebidos para solucionar um problema particular.
- ★ Os agentes operam de modo concorrente.

Projeto da interação

- ★ A interação entre os agentes é realizada seja por troca de mensagens, seja por meio do compartilhamento de dados comuns.

Projeto da organização

- ★ Uma organização destes agentes é então imposta para restringir o comportamento.

Sistemas Multi-agentes: *Agente*

- 🐜 Um agente apresenta as seguintes características:
 - ★ Pode ser uma entidade *real* ou *virtual*
 - ★ Dispõe de uma capacidade de *percepção* e de *representação* parcial deste ambiente
 - ★ Pode se *comunicar* com outros agentes
 - ★ Possui um *comportamento autônomo*, baseado em suas observações, seu conhecimento parcial obtido durante suas interações com os outros agentes.
 - ★ Tem uma identidade única
 - ★ É capaz de executar um processamento de cálculo
 - ★ Produz ações, a partir de seu conhecimento e dos mecanismos internos que lhe são próprios.

Sistemas Multi-agentes: *Interação*

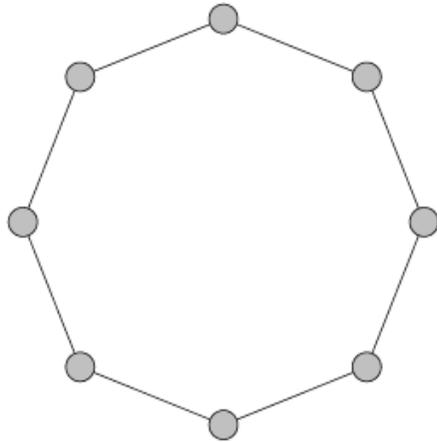
- 🐜 Os agentes cooperam, dividindo entre si as diversas partes do problema original (sub-problemas, tarefas),
- 🐜 Podem aplicar diferentes estratégias de resolução para uma mesma tarefa;
- 🐜 Pode existir a noção de um controle global, na maior parte dos casos implícito aos agentes, que garante um comportamento global coerente do sistema, conforme a organização pré-definida.
- 🐜 Este controle pode ser implementado de modo:
 - ★ Centralizado: eleição de um agente responsável pela gerência do sistema
 - ★ Distribuído: utilizando pontos de sincronismo no sistema

Sistemas Multi-agentes: *Interação*

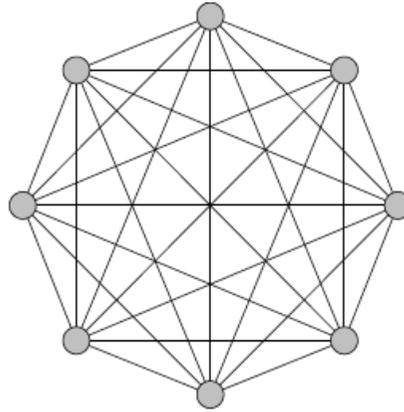
- ❗ A concepção das interações também é realizada independentemente de uma aplicação alvo particular.
- ❗ Busca-se desenvolver protocolos de interação genéricos, que possam ser reutilizados em várias aplicações similares.
- ❗ Um exemplo de um tal protocolo seria, por exemplo, um protocolo de apresentação de um agente quando este ingressa numa sociedade.
- ❗ Obviamente, um protocolo deverá ser instanciado com dados do domínio do problema para poder ser efetivamente ser utilizado numa aplicação;

Sistemas Multi-agentes: *Organização*

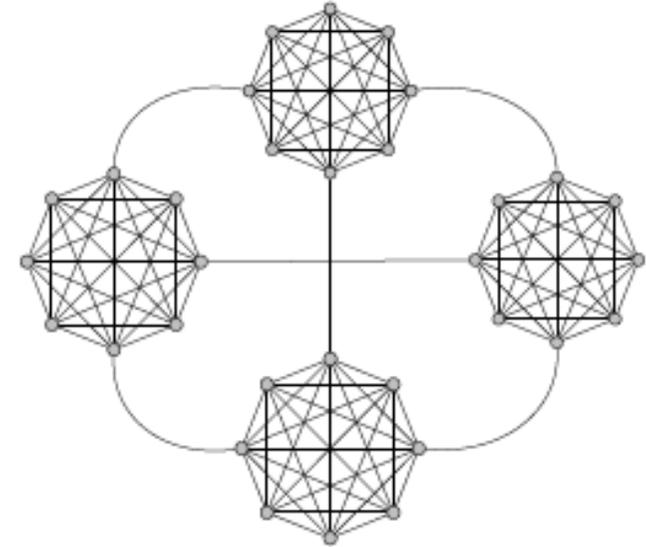
 Topologias mais comuns:



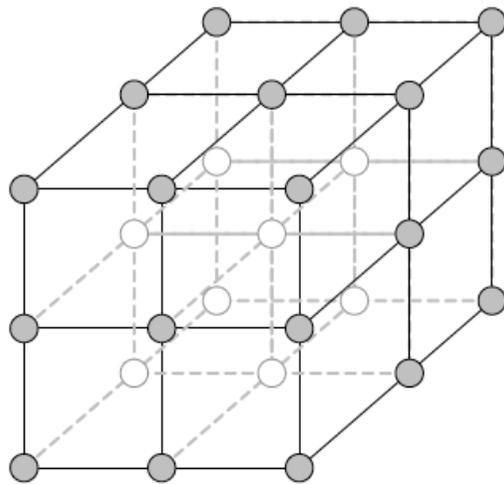
anel



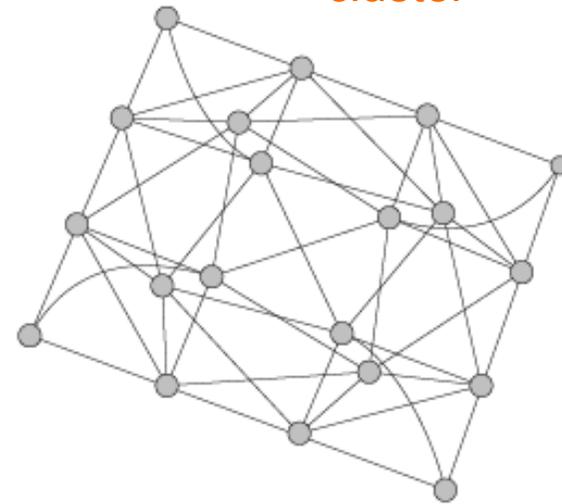
estrela



cluster



Von Neuman



Pirâmide

Sistemas Multi-agentes: *Problemas*

- Descrição, decomposição e alocação de tarefas:
 - ★ Como deve-se descrever e decompor uma tarefa complexa em sub-tarefas, de modo estático ou dinâmico
 - ★ Como estas sub-tarefas devem ser alocadas e em que ordem devem ser executadas;
- Interação, linguagem e comunicação:
 - ★ Quais primitivas básicas de comunicação devem ser utilizadas em um trabalho cooperativo
 - ★ Como estas primitivas podem ser combinadas em protocolos de resolução de problemas;

Sistemas Multi-agentes: *Problemas*



Coordenação, controle e comportamento coerente:

- ★ Como se pode garantir um comportamento global coerente numa coleção de agentes, cada qual com seus próprios objetivos e habilidades
- ★ Como um controle de tal sistema pode ser projetado;



Conflito e incerteza:

- ★ Os agentes não têm uma informação completa sobre o seu ambiente
- ★ Como os conflitos que surgem a partir de visões locais podem ser detectados e resolvidos
- ★ Como dados incertos e incompletos devem ser manipulados de modo a garantir resultados globais coerentes;

Sistemas Multi-agentes: *Ambientes*

- 🐜 Totalmente observável *vs.* Parcialmente observável
 - ★ Valores fornecidos pelos sensores definem completamente o estado do ambiente;
 - ★ Não é preciso manter um estado interno;
 - ★ Sensores precisos/imprecisos; falta de informação;
 - ★ Jogo de xadrez *vs.* direção de carro.
- 🐜 Determinístico *vs.* Estocástico
 - ★ Próximo estado do ambiente é completamente definido pelo se estado atual e as ações do agente;
 - ★ Palavras cruzadas *vs.* direção de carro.

Sistemas Multi-agentes: *Ambientes*

Estático vs. Dinâmico

- ★ O ambiente pode mudar enquanto o agente está processando uma percepção;
- ★ Jogo de xadrez vs. direção de carro;

Discreto vs. Contínuo

- ★ As possíveis percepções e ações do agente são numeráveis;
- ★ Jogo de xadrez vs. direção de carro.

Sistemas Multi-agentes: *Ambientes*

Episódico vs. Sequencial

- ★ A experiência do agente é um conjunto de episódios;
- ★ Um episódio é um par (percepção, ação);
- ★ Um episódio é independente do anterior.
- ★ Jogo de roleta vs. Jogo de xadrez

Agente único vs. Agentes múltiplos

- ★ Ambiente competitivo (agente para jogar xadrez);
- ★ Ambiente cooperativo (agente para dirigir carro);

Sistemas Multi-agentes: *Ambientes*

Exemplos de ambientes

	Palavras Cruzadas	Xadrez temporizado	Diagnóstico médico	Direção de carro
Observável	sim	sim	não	não
Determinístico	sim	não	não	não
Episódico	não	não	não	não
Estático	sim	sim	não	não
Discreto	sim	sim	não	não
Agente único	sim	não	sim	não

Sistemas Multi-agentes: *Agente*

 Combinação mais difícil:

- ★ Parcialmente observável;
- ★ Estocástico;
- ★ Sequencial;
- ★ Dinâmico;
- ★ Contínuo;
- ★ Multi-agentes.

Sistemas Multi-agentes: *Agente*

-  Agente = programa + arquitetura
-  O programa do agente é um função que mapeia as percepções (valores obtidos dos sensores) do agente para suas ações (valores para os atuadores).
-  Arquitetura do agente é o suporte em termos de software básico e hardware que permitem a execução do programa do agente com eficácia e eficiência.

Sistemas Multi-agentes: *Arquiteturas*

 Sistemas multi-agentes podem ser:

★ Reativos

★ Baseados em comportamento fixo

★ Baseados em regras

★ Baseados em modelos

★ Cognitivos

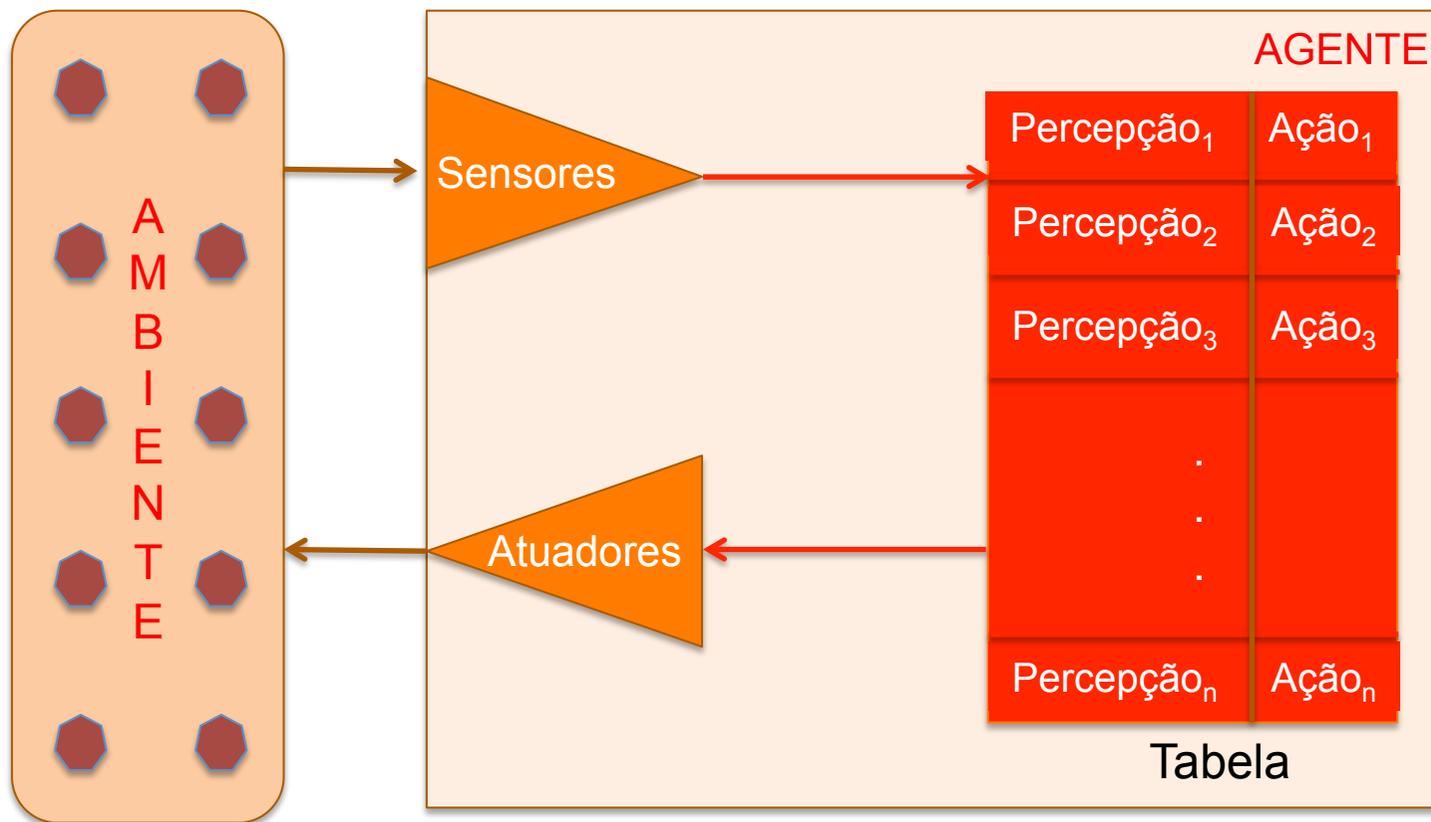
★ Baseados em objetivos

★ Baseados em utilidades

★ Baseados em aprendizagem

Sistemas Multi-agentes: *Arquiteturas*

🐜 Agentes reativos simples:



Sistemas Multi-agentes: *Arquiteturas*



Agentes reativos simples:

- ★ Vantagens

- ★ Fácil de implementar

- ★ Aplicáveis somente em casos onde o ambiente é:

- ★ Observável

- ★ Determinístico

- ★ Episódico

- ★ Estático

- ★ Discreto

- ★ Problemas muito reduzidos

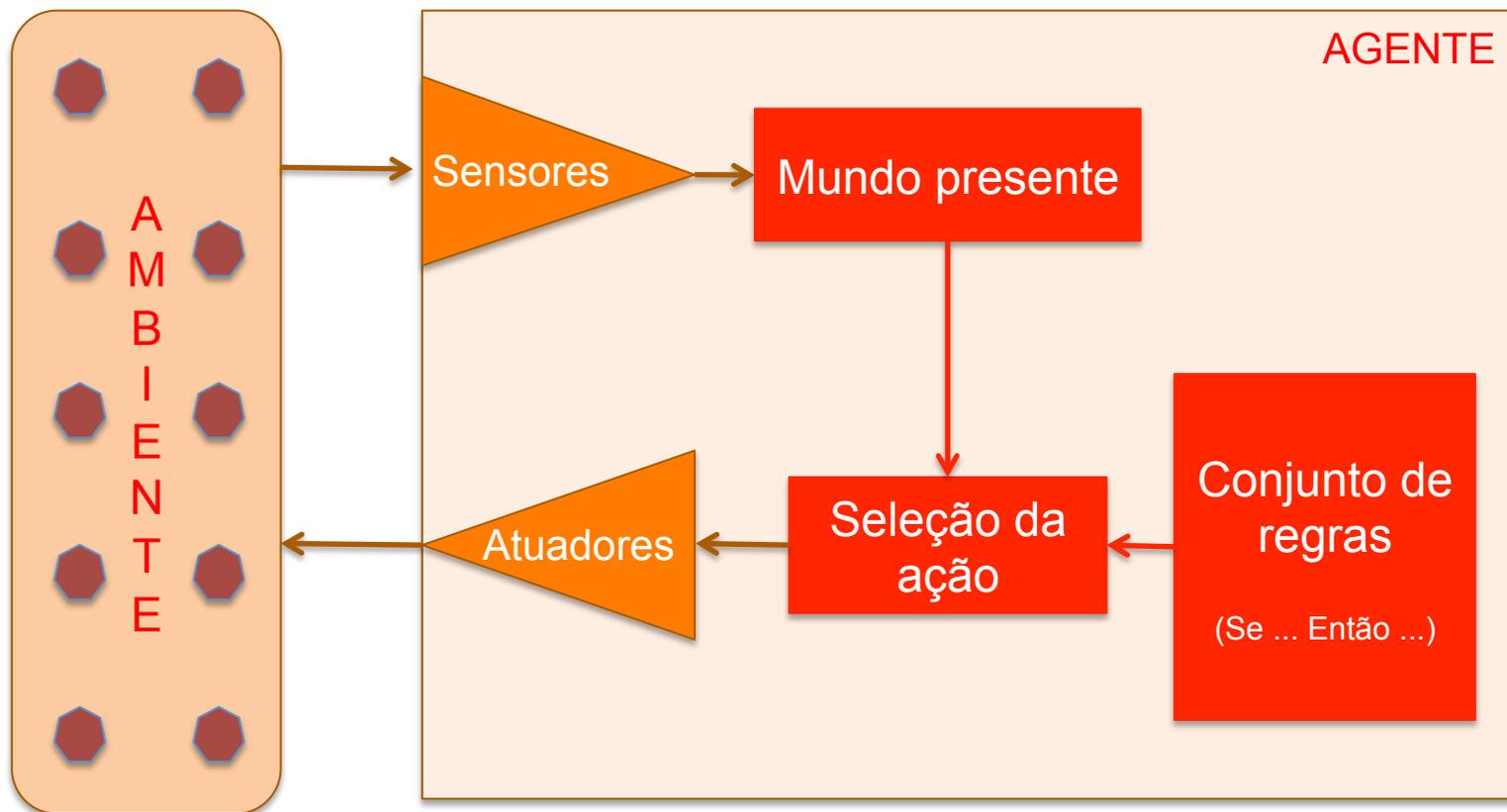
- ★ Desvantagens

- ★ Não têm flexibilidade

- ★ Não têm autonomia

Sistemas Multi-agentes: *Arquiteturas*

🐜 Agentes reativos baseados em regras:



Sistemas Multi-agentes: *Arquiteturas*

 Agentes reativos baseados em regras:

- ★ Vantagens

 - ★ Têm alguma flexibilidade

- ★ aplicáveis somente em casos onde o ambiente é:

 - ★ Observável

 - ★ Determinístico

 - ★ Episódico

 - ★ Estático

 - ★ Discreto

 - ★ Problemas reduzidos

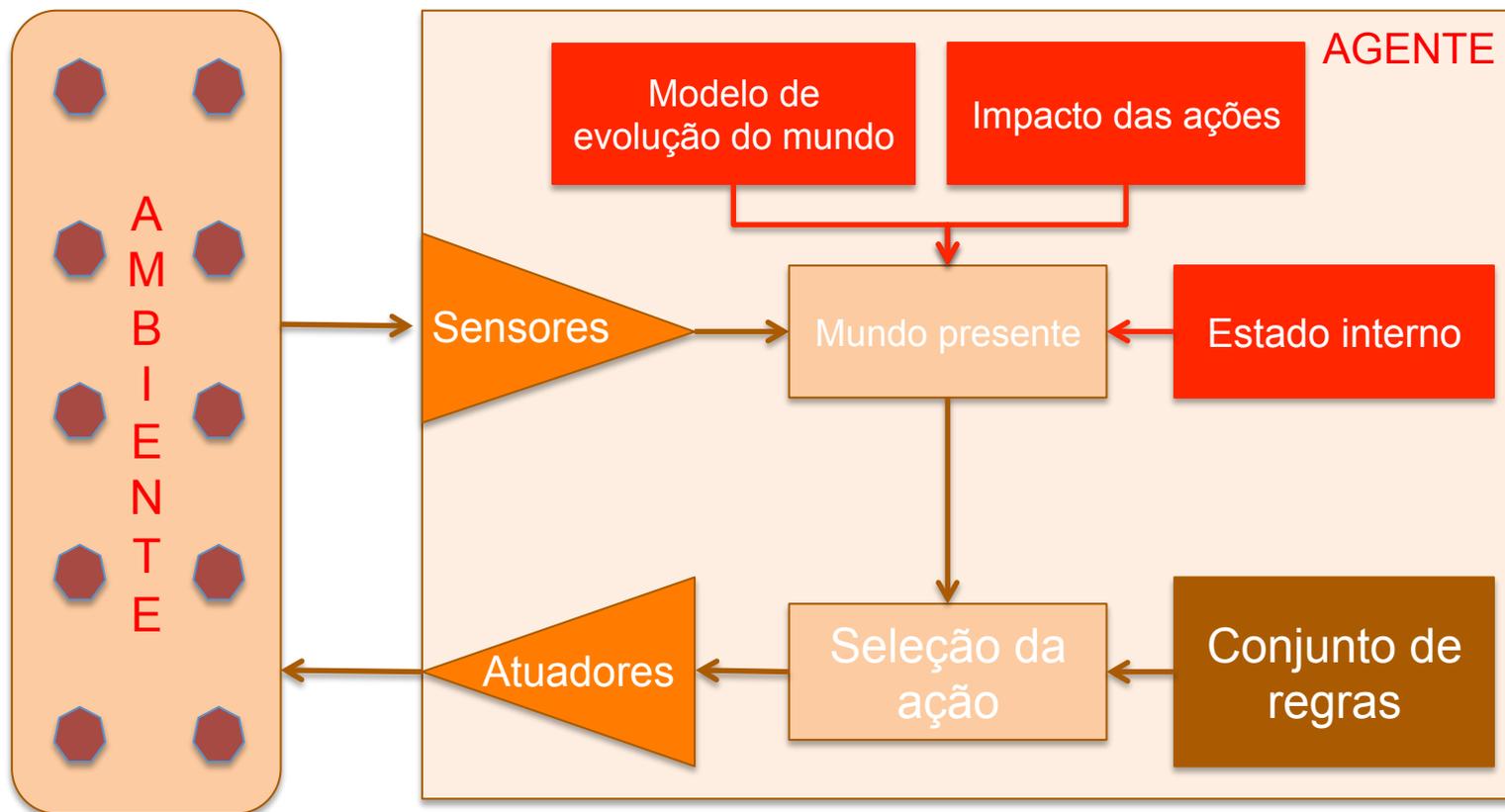
- ★ Desvantagens

 - ★ Têm pouca flexibilidade

 - ★ Não têm autonomia

Sistemas Multi-agentes: *Arquiteturas*

🐜 Agentes reativos baseados em modelos:



Sistemas Multi-agentes: *Arquiteturas*



Agentes reativos baseados em modelos:

- ★ Vantagens

- ★ Conseguem lembrar (impacto das ações)

- ★ Conseguem imaginar (modelo de evolução do mundo)

- ★ Têm flexibilidade

- ★ Têm alguma autonomia

- ★ Aplicáveis somente em casos onde o ambiente é:

- ★ Determinístico

- ★ Problemas reduzidos

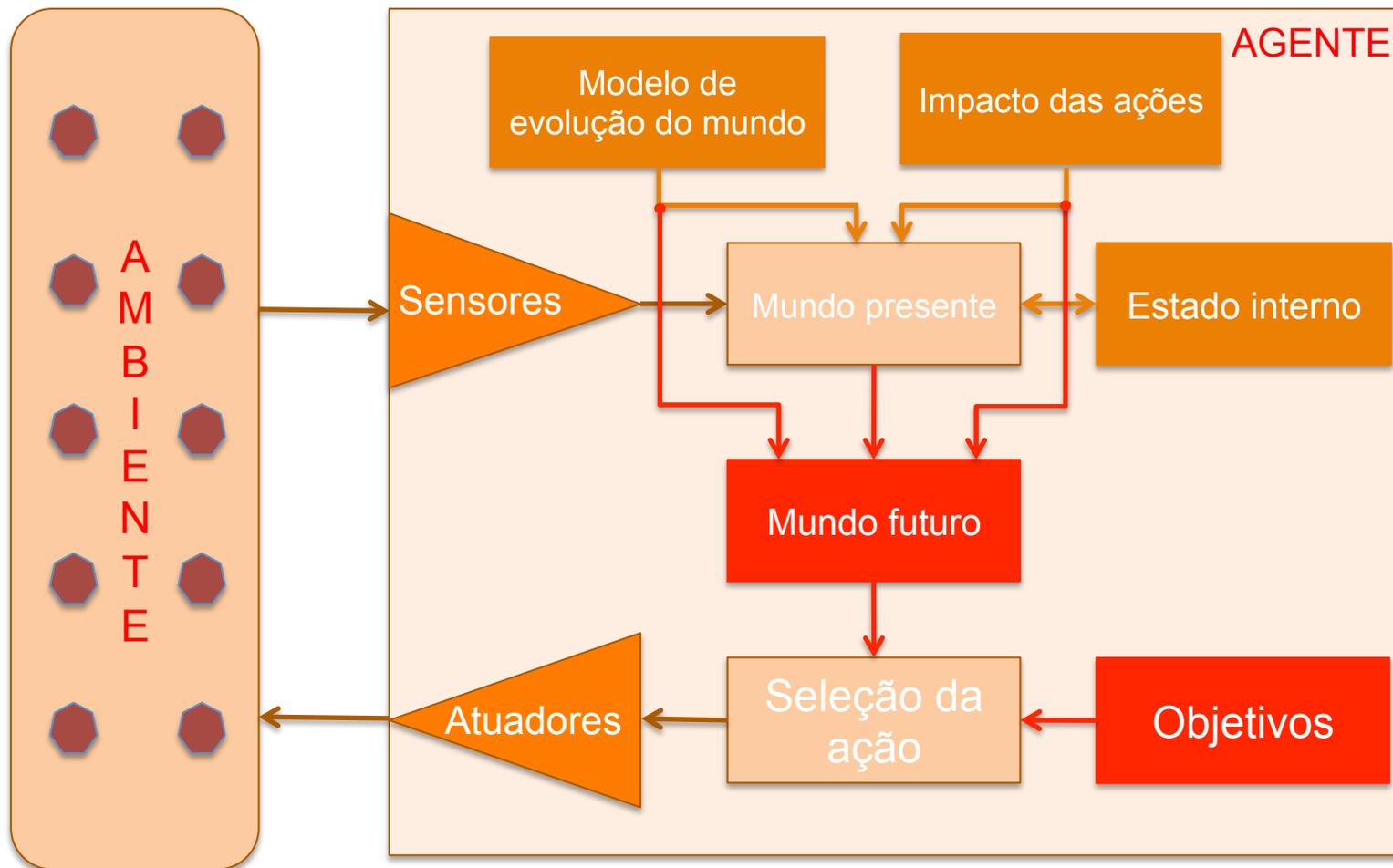
- ★ Desvantagens

- ★ Têm pouca autonomia

- ★ Não têm objetivos

Sistemas Multi-agentes: *Arquiteturas*

🐜 Agentes cognitivos baseados em objetivos:



Sistemas Multi-agentes: *Arquiteturas*

 Agentes cognitivos baseados em objetivos:

★ Vantagens

★ Têm autonomia

★ Têm mais flexibilidade

★ Têm objetivos

★ Aplicáveis somente em casos onde o ambiente é:

★ Determinístico

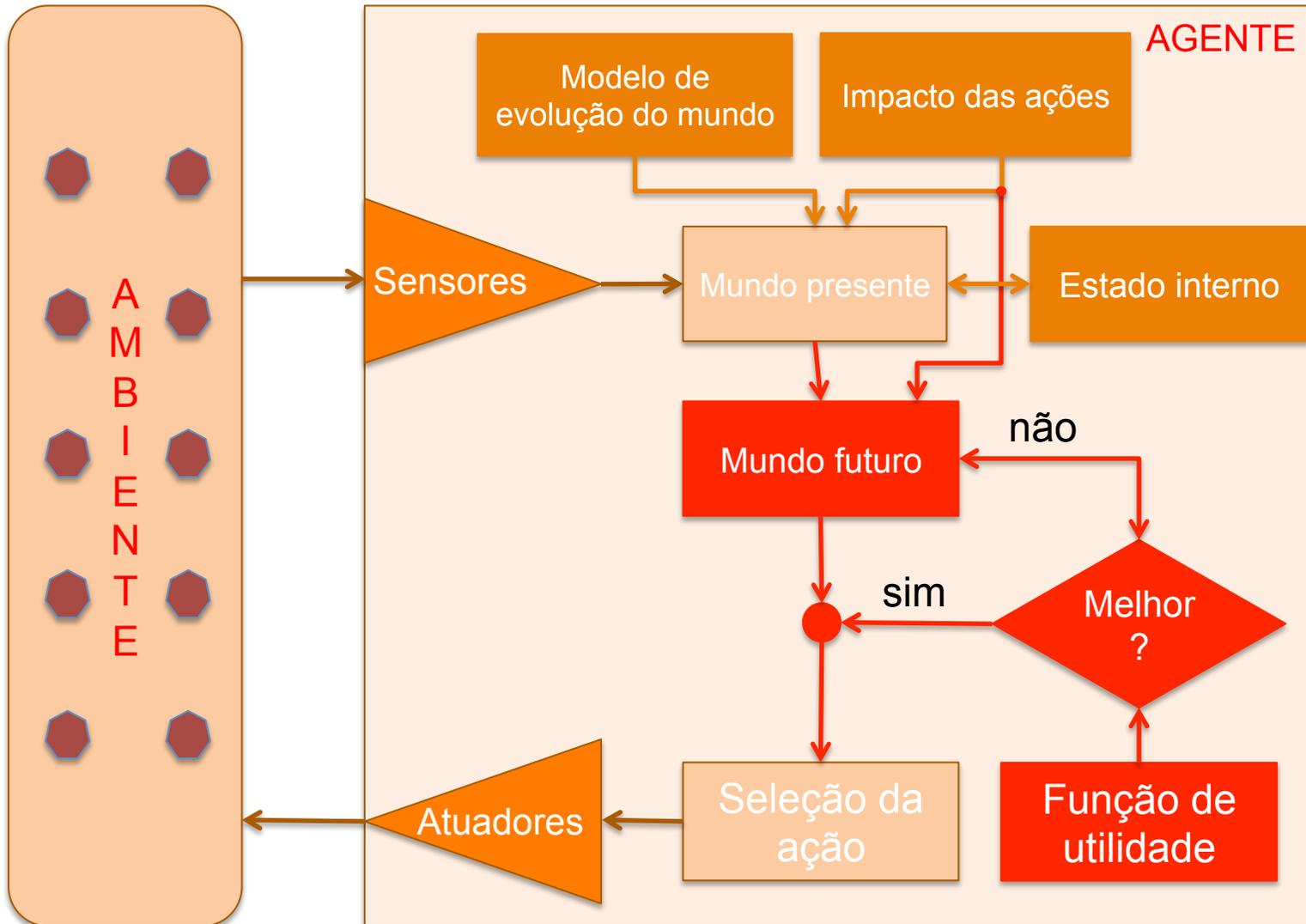
★ Desvantagens

★ Não permite objetivos conflitantes

★ Não consegue avaliar seu desempenho em chegar ao seu objetivos

Sistemas Multi-agentes: *Arquiteturas*

🐜 Agentes cognitivos baseados em utilidades:



Sistemas Multi-agentes: *Arquiteturas*

 Agentes cognitivos baseados em utilidades:

★ Vantagens

★ Lida com objetivos conflitantes

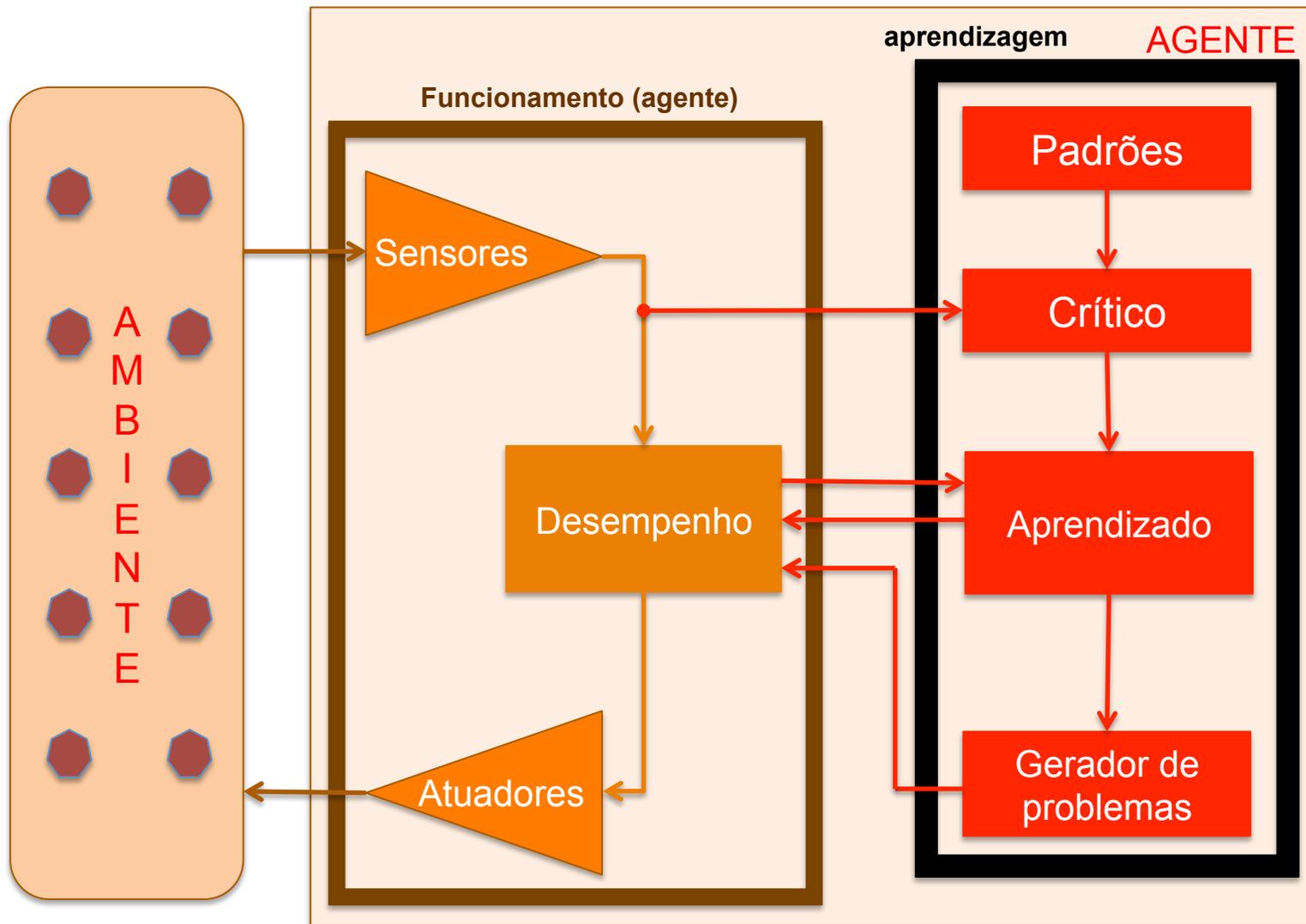
★ Aplicáveis em qualquer tipo de ambiente

★ Desvantagens

★ Não aprende

Sistemas Multi-agentes: *Arquiteturas*

🐜 Agentes cognitivos baseados em aprendizagem:



Sistemas Multi-agentes: *Arquiteturas*

- 🐜 Agentes cognitivos baseados em aprendizagem incluem:
 - ★ Componente de desempenho:
 - ★ Formado por um agente completo para realizar as funções de:
 - ✦ Percepção
 - ✦ Inferência
 - ✦ Ação
 - ★ Pode ser baseado em qualquer arquitetura
 - ★ Componente de aprendizagem:
 - ★ Formado por três componentes:
 - ✦ Crítico
 - ✦ Elemento de aprendizado
 - ✦ Gerador de problemas

Sistemas Multi-agentes: *Arquiteturas*

 Agentes cognitivos baseados em aprendizagem incluem:

★ Crítico:

★ Determina como o agente de desempenho deve ser melhorado, levando em conta padrões pré-definidos.

★ Usa um feedback do elemento de aprendizagem sobre a atuação do agente de desempenho.

★ Elemento de aprendizado:

★ Responsável pelos aperfeiçoamentos do agente na sua ação.

★ Gerador de problemas:

★ Responsável por sugerir ações que levariam a novas situações e poderiam levar a melhoria do agente de desempenho.

Sistemas Multi-agentes: *Arquiteturas*

 Agentes cognitivos baseados em utilidades:

- ★ Vantagens

 - ★ Consegue aprender

- ★ Aplicáveis em qualquer tipo de ambiente

- ★ Desvantagens

 - ★ Difícil de implementar