

Agentes de Software

(com inspiração na indústria do sw de entretenimento)

Agente?

- Entidade que interage com o meio ambiente, percebendo-o através de «sensores» e actuando nele, utilizando «actuadores».

Agente Inteligente?

- Um agente racional.

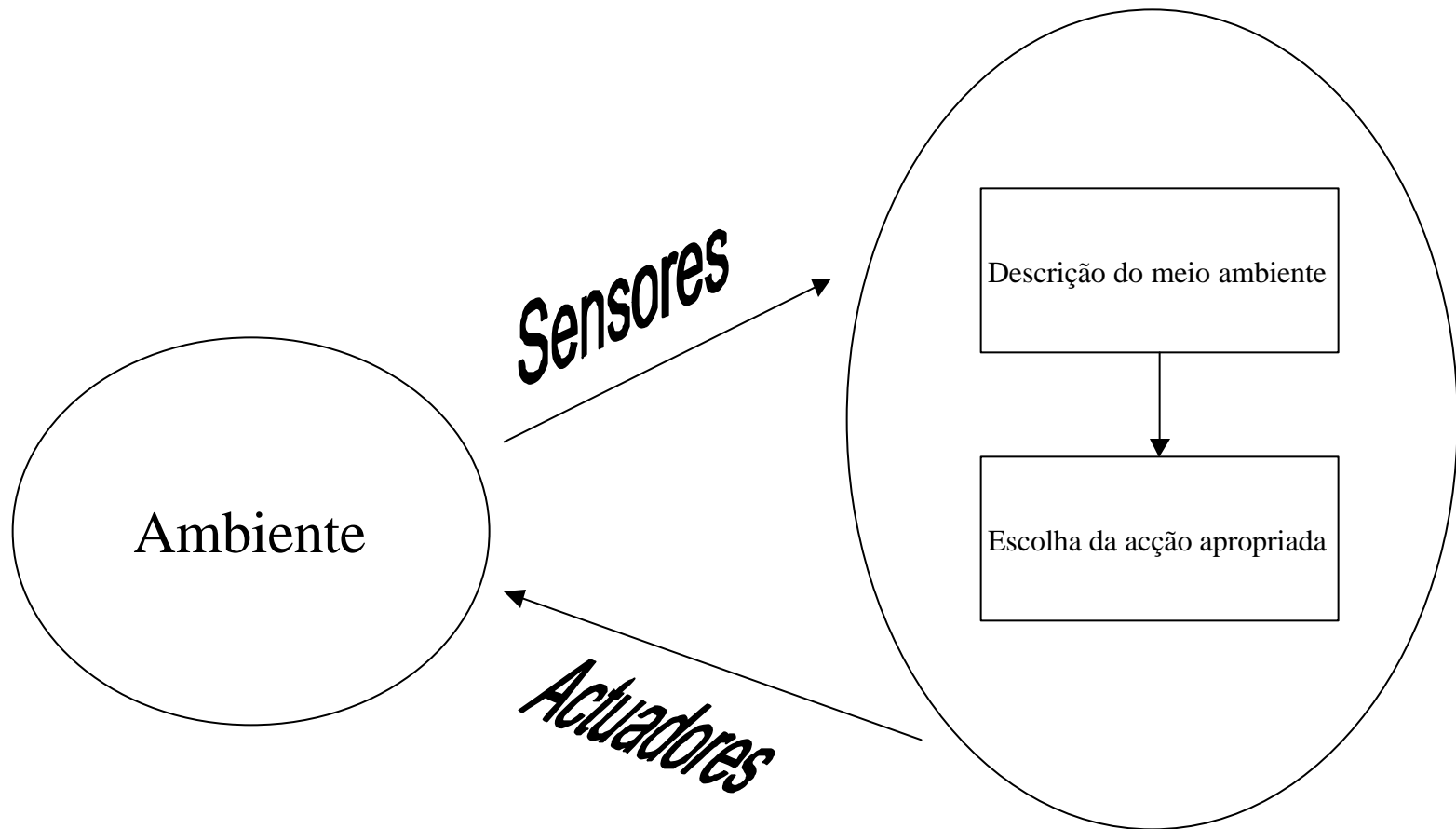
Agente Racional?

- Aquele que faz aquilo que lhe propicia maior sucesso.
- Mas como avaliar o sucesso do agente?
- É preciso ter uma medida ou critério de desempenho.
- Quem define esse critério?

Ferramentas do Agente Racional

- Medida de desempenho (objectiva / subjectiva)
- Percepções
- Conhecimento *a priori*
- Conjunto de acções disponíveis

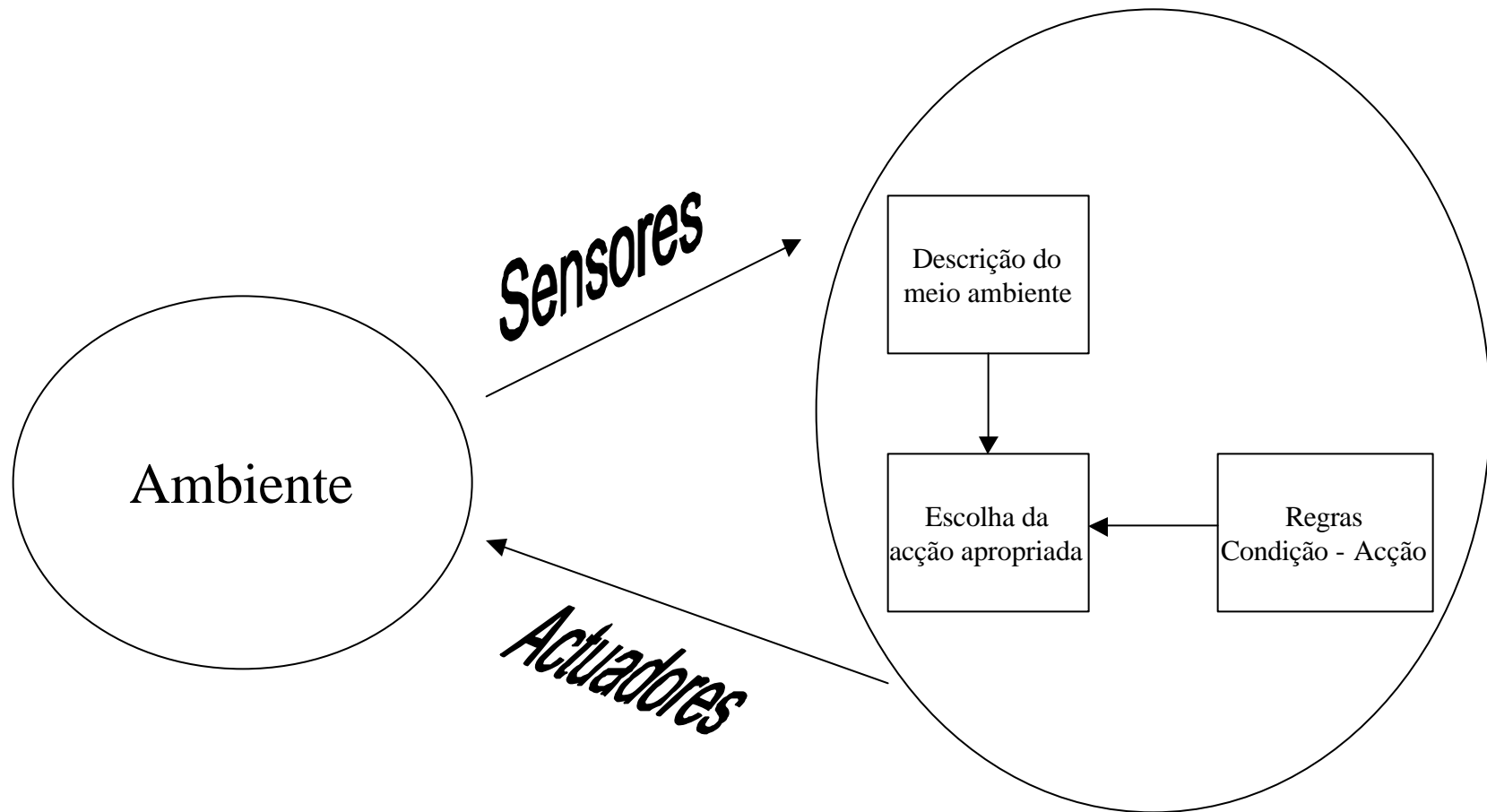
Agente Simples



Agente Simples (considerações)

- #1) Percepção
- #2) Procura na tabela
- #3) Acção, conforme indexada
- Problemas:
 - => dimensão da tabela é factor limitador
 - => agentes sem autonomia

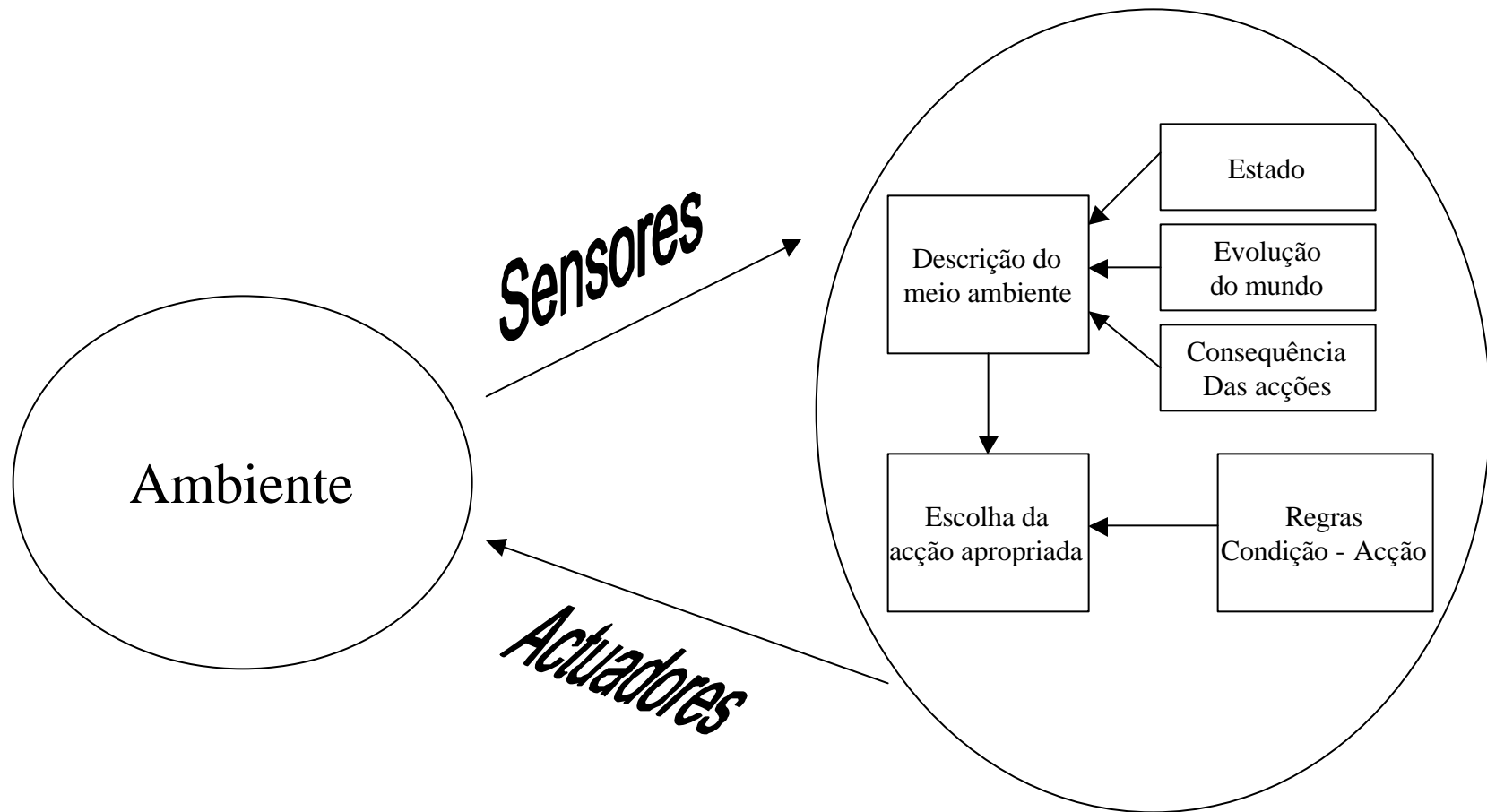
Agente Reactivo



Agente Reactivo (considerações)

- Regras condição – acção contribuem para maior naturalidade das percepções.
- Insuficiência: sem informação *a priori*, não se pode decidir correctamente.
- Exemplo do espelho retrovisor (ângulo morto).

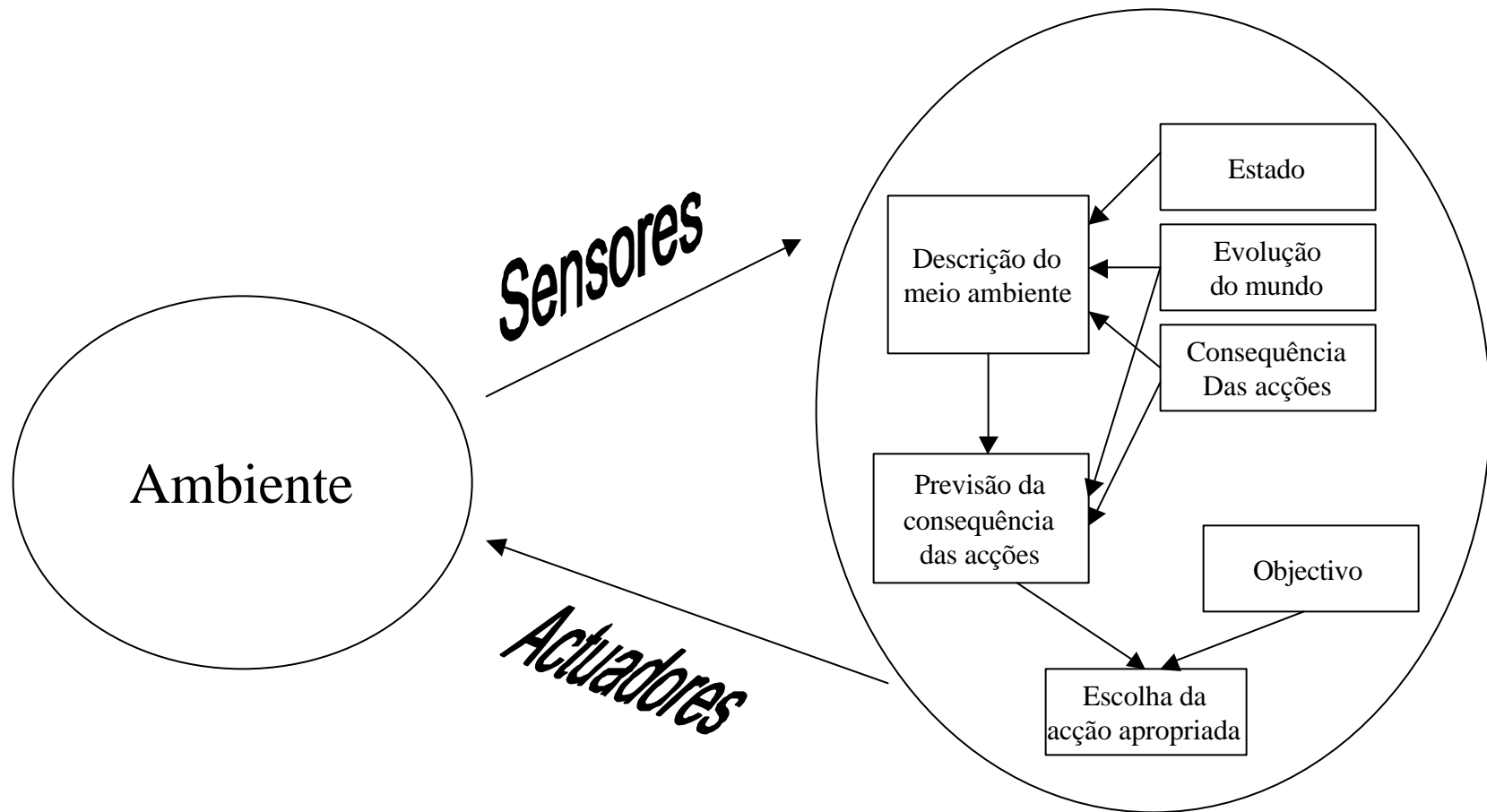
Agente Informado



Agente Informado (considerações)

- Mantém informação sobre a evolução do ambiente.
- Mas pode ser insuficiente para o que se pretende...
- Seria conveniente haver um objectivo final com peso na tomada de decisões...

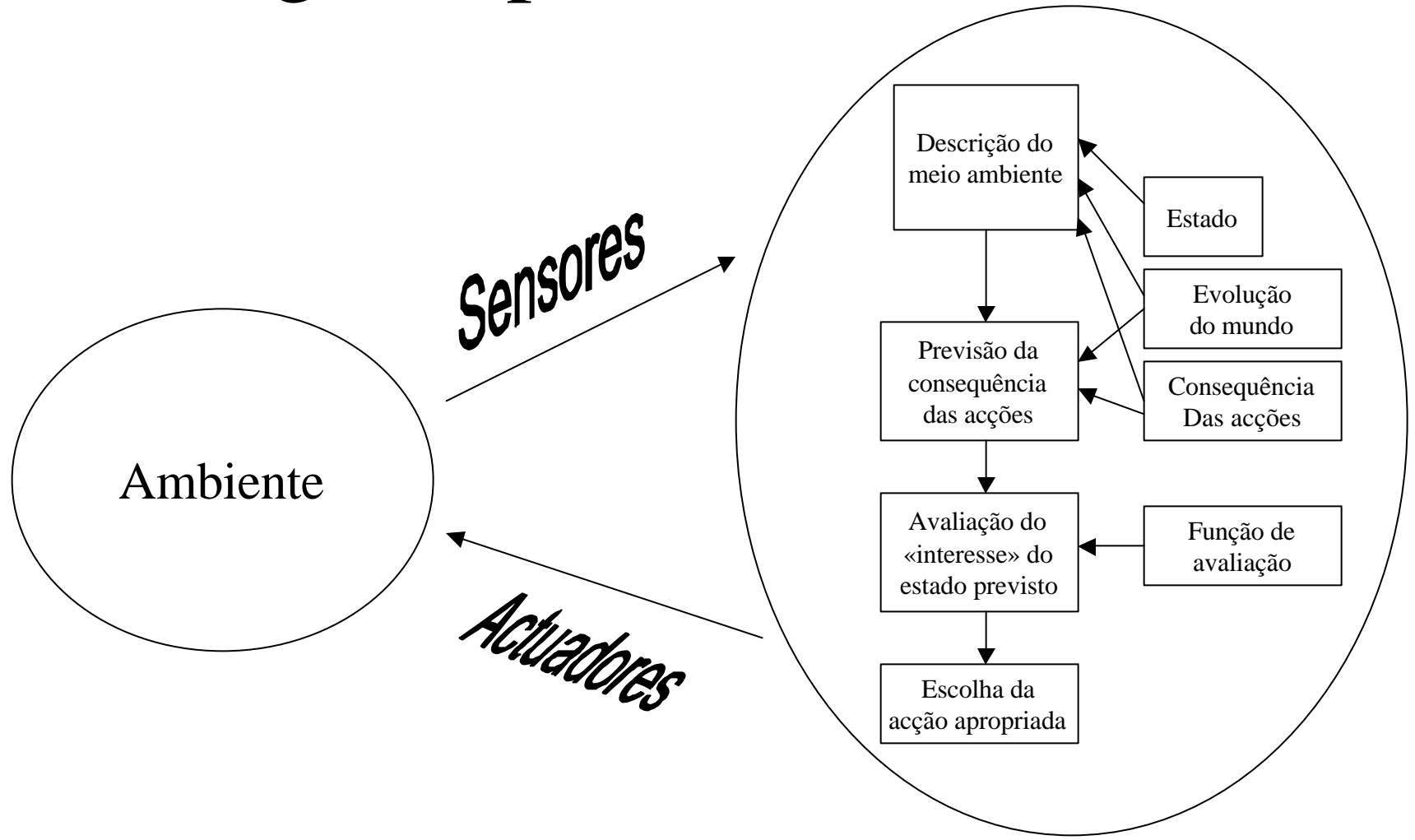
Agente por Objectivos



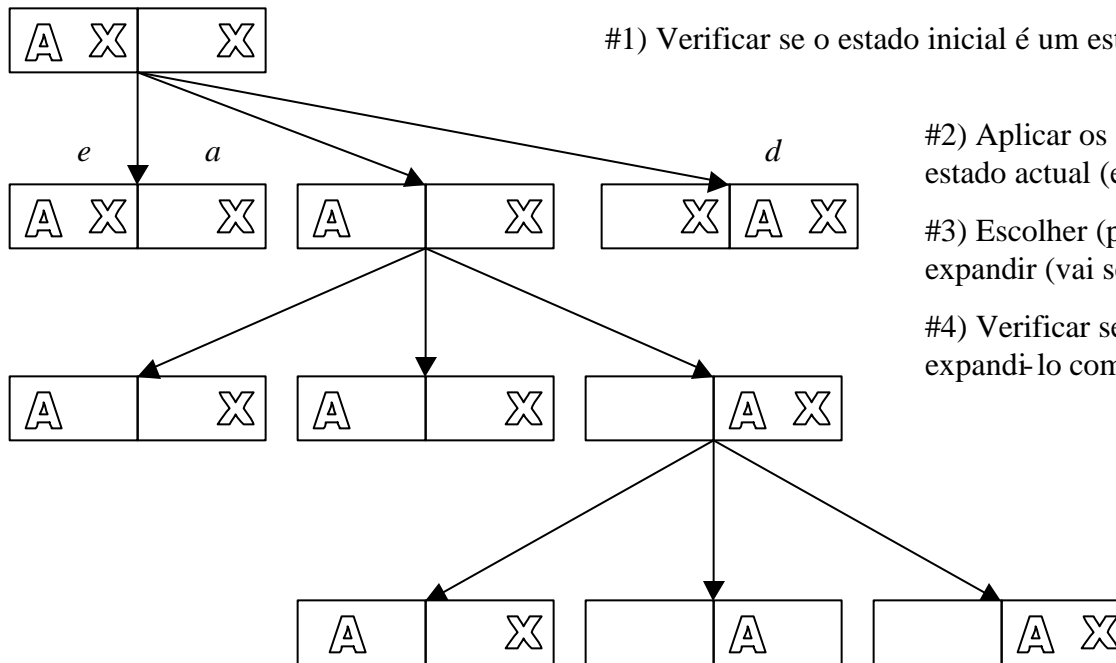
Agente por Objectivos (considerações)

- Necessita de um objectivo
- A tomada de decisões acontece após análise da adequação dos cenários que podem resultar das decisões, relativamente ao objectivo.
- Pode não ser eficaz (todos os caminhos vão dar a Roma...)

Agente por Preferências



Agente Aspirador



#1) Verificar se o estado inicial é um estado objectivo (não é).

#2) Aplicar os operadores, para gerar os sucessores do estado actual (esquerda, aspirar, direita).

#3) Escolher (procurar / seleccionar) o próximo estado a expandir (vai ser o que resulta de aspirar)

#4) Verificar se é um estado objectivo (não é). Se não for, expandi-lo como em #2).

Dependendo da função de avaliação de estados, um destes estados (o do meio) seria seleccionado e - sendo um estado objectivo - a procura estaria completa.

Implementação de Agentes Inteligentes Autônomos

- Modelo para representação do conhecimento (se for um agente com conhecimento *a priori*, ou capaz de aprender)
- Motor de inferência de novo conhecimento
- Algoritmos de procura / selecção no espaço de estados.

Algoritmo Genérico de Procura

- #1) definir o estado inicial como estado actual.
- #2) ver se o estado actual é um estado objectivo. Se sim, terminar com sucesso.
- #3) aplicar operadores ao estado actual, para gerar novos estados.
- #4) dos estados gerados, escolher o próximo estado actual.
- #5) voltar a #2)

Estratégias de Procura

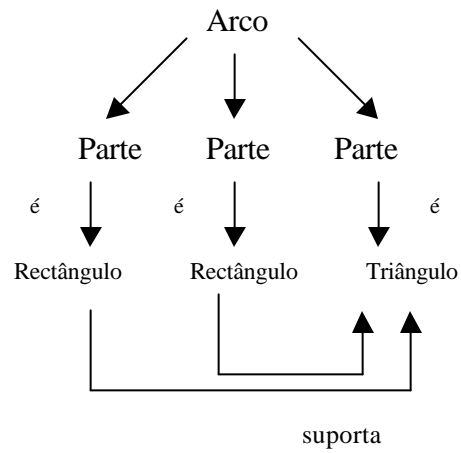
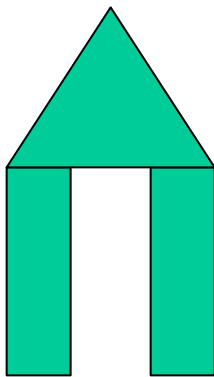
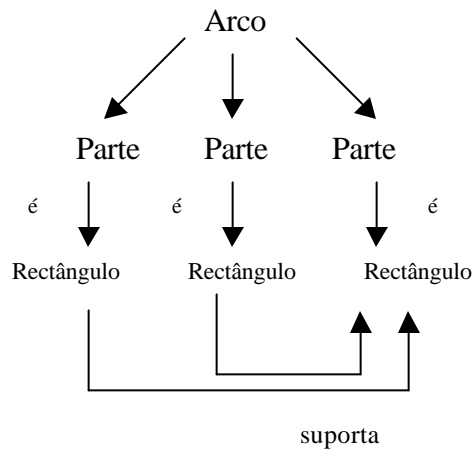
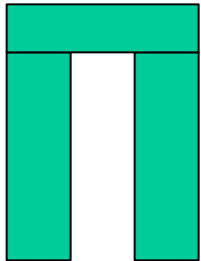
- Procuras cegas / não informadas: não utilizam qualquer informação sobre o número de passos ou o custo do caminho, desde o estado actual, até ao estado objectivo.
- Heurísticas / procuras informadas. Exemplo: procura com custo uniforme (expande sempre o estado de menor custo).

Definição Behaviorista de Aprendizagem

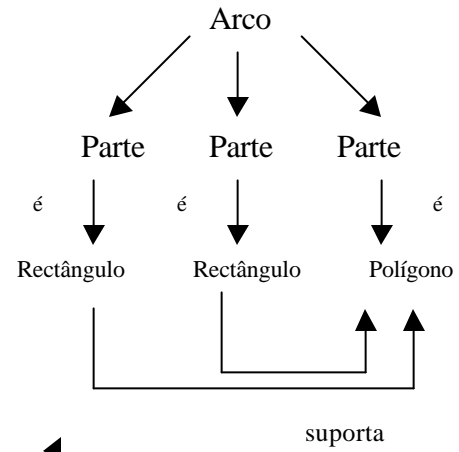
- Um sistema diz-se que aprende, quando desempenha melhor, na repetição de uma tarefa.
- Fica a noção de que a experiência é importante.

Tipos de Aprendizagem:

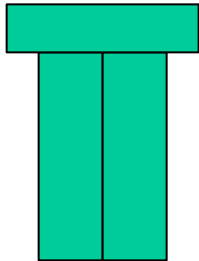
- «Decorar»: memorização de todas as situações. Problema: espaço de procura pode crescer demasiado e ter consequências negativas no desempenho.
- «Por exemplo», a partir de exemplos positivos e negativos. Exemplo dos blocos.
- «Por conselho». Tem o problema da tradução a partir de alto nível.
- «À descoberta».



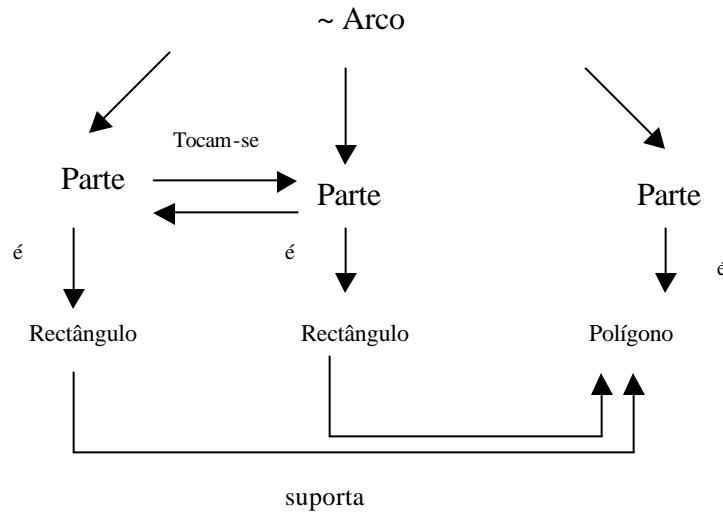
Aprendizagem
por Exemplo
(positiva /
generalização)



Generalização
(a partir de 2 exemplos positivos)



Aprendizagem por Exemplo (negativa / especialização)



Especialização de uma descrição, a partir de um exemplo pela negativa, após uma generalização pela positiva.

