Agentes Reativos

Capítulo 2:
Costa, E. e Simões, A. (2015). Inteligência Artificial – Fundamentos e Aplicações, 3.ª edição, FCA.

Generalidades

- Aos sistemas computacionais que possuem capacidades como

  Autonomia

  Flexibilidade

  Aprendizagem

  dá-se o nome genérico de AGENTE

Taxonomia de Agentes

- Como definir/classificar um agente ?
- Agentes do nosso dia-a-dia:
  - agentes de interface
  - agentes de procura (web)
  - agentes de filtragem (correio eletrónico)
  - agentes assistentes (agenda)

Taxonomia de Agentes

- Um agente autónomo é um sistema situado num ambiente capaz de percecionar esse ambiente e atuar sobre ele, ao longo do tempo, tendo em vista a satisfação dos propósitos da sua agenda, de modo a afetar o que será percecionado no futuro.

  [Franklin e Graesser, 1977]

Taxonomia de Agentes

- Autonomia

  O agente pode tomar decisões sem a intervenção direta ou indireta de outros agentes (humanos ou não).

Taxonomia de Agentes

- O agente existe mergulhado no ambiente

  Está em permanente interação com ele, podendo modificá-lo.

Taxonomia de Agentes

- O agente tem a sua agenda

  A sua ação é guiada com vista à concretização dessa agenda.

Taxonomia de Agentes

- Assim, são agentes:

  - termóstato para controlar a temperatura numa sala

  - uma bactéria

  - O ser humano (agente bem mais complexo).

Propriedades dos Agentes

- Podemos acrescentar algumas propriedades a esta classificação de agentes

- Reatividade

  Responde em tempo útil a mudanças no ambiente.

Propriedades dos Agentes

- Orientado por objetivos

  Tem iniciativa.

  Não se limita a atuar apenas como resposta ao ambiente.

Propriedades dos Agentes

- Comunicação

  Comunica com outros agentes (eventualmente de outro tipo).

Propriedades dos Agentes

- Aprendizagem

  Muda o seu comportamento de acordo com a sua experiência prévia.

Propriedades dos Agentes

- Mobilidade

  Capaz de se transportar a ele próprio de máquina para máquina.

Propriedades dos Agentes

- Carácter

  Tem um estado emocional e a sua própria personalidade.

Tipos de Agentes Autónomos

- Assim, é possível definir uma hierarquia de agentes
- Agentes:
  - Biológicos (agentes naturais)
  - Robóticos (agentes artificiais)
  - Computacionais (agentes que apenas existem como programas de computador)

Tipos de Agentes Autónomos

- Agentes computacionais:
  - Agentes de Software (em princípio, são eternos)
  - Agentes de Vida Artificial (agentes que morrem)

Normalmente, é a pensar nos agentes de software que usamos a palavra agente

Arquiteturas Abstratas de Agentes

- Como projetar um agente autónomo ?
- Como formalizar a noção de agente e a sua arquitetura ?

Arquiteturas Abstratas de Agentes

- O Agente vive mergulhado num Ambiente
- O Ambiente é caracterizado por um conjunto de estados

$$E = \{e_1, e_2, \ldots\}$$

Conjunto de estados que caracterizam o ambiente

Arquiteturas Abstratas de Agentes

- O agente interage com o ambiente, captando o seu estado e realizando um conjunto de ações sobre o ambiente que podem levar à mudança do seu estado

$$A = \{a_1, a_2, \ldots\}$$

Conjunto de ações que o agente pode realizar

Arquiteturas Abstratas de Agentes

- O agente pode ser visto como uma entidade que faz o mapeamento de estados do ambiente em ações

$$\text{Agente: } E^* \rightarrow A$$

- Dada uma sequência de estados do ambiente (E^*) o agente reage com uma determinada ação (A)

Arquiteturas Abstratas de Agentes

- É normal decompor o agente em duas partes

  Uma transforma os estados do ambiente em perceções;

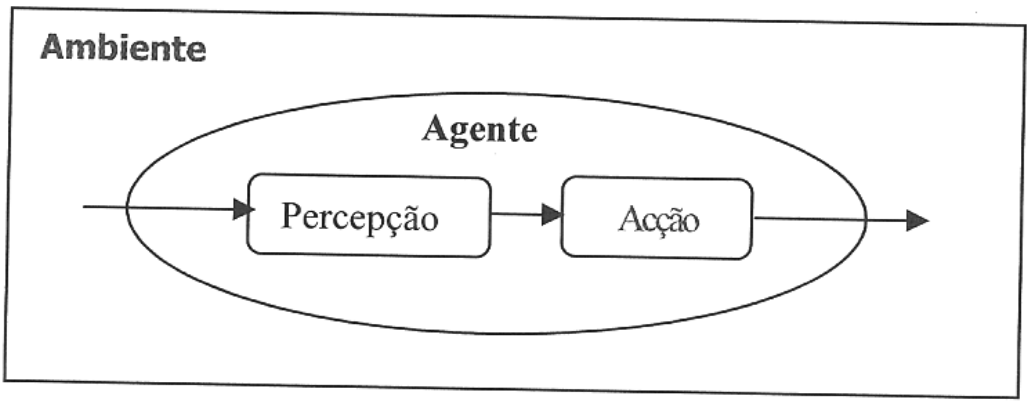
  A outra transforma uma sequência de perceções (P^*) numa ação:

  Perceção: $E \rightarrow P$

  Ação: $P^* \rightarrow A$

Arquiteturas Abstratas de Agentes

- A função Agente não é mais do que a composição das funções Perceção e Ação



Representação de um agente reativo

Arquiteturas Abstratas de Agentes

- Ou seja, se o agente percecionar o estado e, muda para um novo estado e'

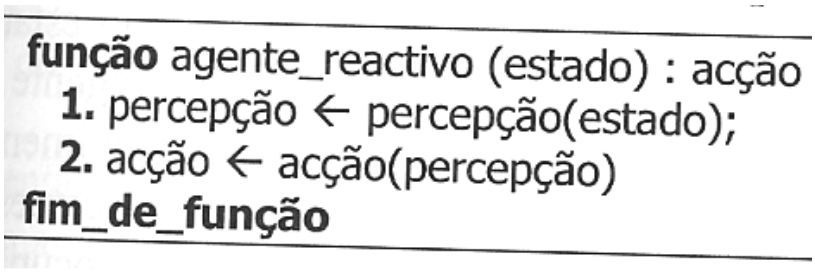
  e'= Agente(e)= Ação (Perceção(e))

- Se o agente não tiver a capacidade de guardar os estados passados do ambiente:

  Agente: E → A

Arquiteturas Abstratas de Agentes

- Descrição simplificada de um agente reativo através de um programa simples:



Ambientes, Modelos e Representações

- Ambiente é um micromundo (visão parcial e limitada do mundo)
- É um espaço a n dimensões, povoado por diferentes tipos de habitantes
- Pode ser:
  - limitado (ou fechado)
  - aberto (ou infinito)
  - toroidal (os extremos opostos do mundo estão ligados)

Ambientes, Modelos e Representações

- Os habitantes têm uma localização nesse ambiente dada pelas suas coordenadas no espaço
- Os habitantes podem ser de 2 tipos:
  - agentes (biológicos ou não)
  - objetos (podem ser fixos, movimentáveis, manipulados pelos agentes, …)

Ambientes, Modelos e Representações

- Quanto à perceção, o conceito de vizinhança do agente é crucial

Vizinhança: zonas do ambiente (conjunto de coordenadas) que o agente consegue percecionar num dado momento

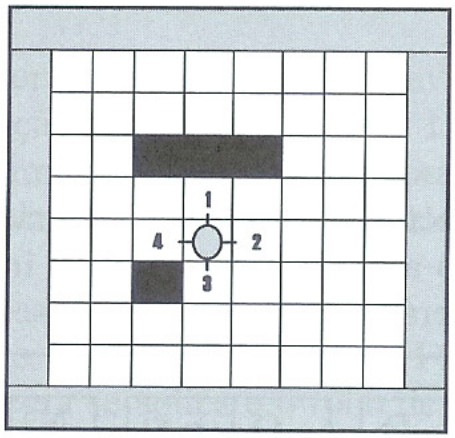
Ambientes, Modelos e Representações

- O agente pode apenas conseguir determinar se as posições da sua vizinhança estão ocupadas ou não e ter como ações primitivas o ato de se mover numa dada direção que esteja livre

- Um agente mais complexo pode ser capaz de determinar o tipo de habitantes presentes na sua vizinhança e realizar ações como comer

Ambientes, Modelos e Representações

- Exemplo: mundo simples a 2 dimensões, fechado, onde habitam 1 agente e alguns obstáculos



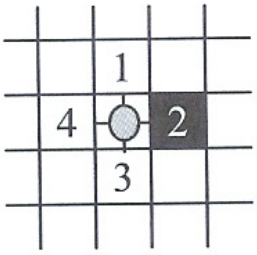
Agente num mundo bidimensional

Ambientes, Modelos e Representações

- O agente tem apenas uma visão local do mundo, conseguindo percecionar apenas as 4 células assinaladas, determinando se estão ocupadas ou não
- No caso de estarem livres, pode movimentar-se para elas

Ambientes, Modelos e Representações

- Exemplo de perceções de um agente



- O estado do ambiente, e, pode ser descrito pelos 4 valores capturados pelos sensores (s_1,s_2,s_3,s_4)

Ambientes, Modelos e Representações

- Ou seja, pelo vetor

$$e = <s_1,s_2,s_3,s_4> = <0,1,0,0>$$

0 – posição livre

1 – posição ocupada

Ambientes, Modelos e Representações

- O agente pode transformar esses dados numa perceção (p) usando uma representação (notação) simples

$\neg O_i$ – livre

O_i – ocupado

i – posição da vizinhança

Ambientes, Modelos e Representações

- Assim, temos

$p = <\text{livre,ocupado,livre,livre}> =$

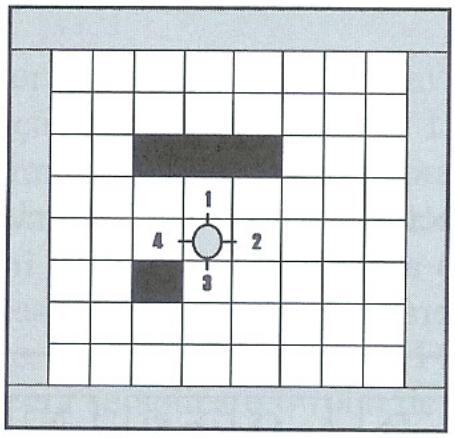
$= < \neg O_1, O_2, \neg O_3, \neg O_4>$

- É necessário representar de algum modo o mundo real

Agentes Puramente Reativos

- Retomemos o ambiente e o agente do exemplo anterior

- Que tarefa ?



Agentes Puramente Reativos

- Que tarefa pretendemos que cumpra ?

  - deambular pelo ambiente, evitando obstáculos ?

  - procurar algo de valor para o agente ?

- Problema de projeto:

  - Como se transformam os dados dos sensores em perceções para o agente e como dão estas origem às ações ?

  - Como representar e implementar a função ação ?

Agentes Puramente Reativos

- Temos 3 abordagens principais:
  - Sistemas de Produções
  - Unidades Lineares de Limiar
  - Arquitetura de subordinação

Sistemas de Produções (SP)

- É constituído por:

  - Um conjunto de produções ou regras do tipo

Se <condição> então <ação>

Sistemas de Produções (SP)

- É constituído por (cont.):

  - Um interpretador de regras ou mecanismo de controlo.

  Tem de lidar com questões como:

  - que regras podem ser ativadas?

  - que regra é escolhida? (mecanismo de resolução de conflitos)

  - disparar a regra (por execução das ações indicadas)

Sistemas de Produções (SP)

- É constituído por (cont.):

  Um mecanismo simples será

  - As produções são colocadas numa lista ordenada e a primeira produção cujas condições sejam verdadeiras é disparada, sendo executada a respetiva ação

  - Por norma, a última regra tem a sua condição sempre verdadeira (para que exista sempre uma regra disparada)

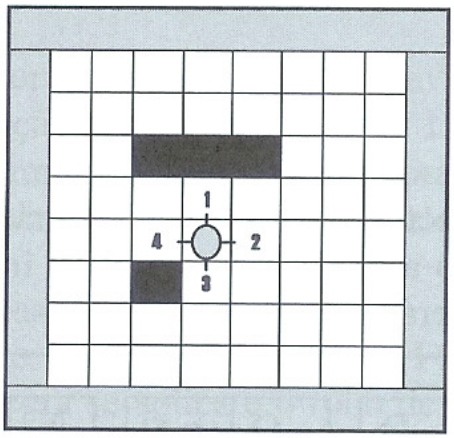
Sistemas de Produções (SP)

- É constituído por (cont.):

  - Memória de trabalho ou ambiente

    - Significa o que é verdadeiro no mundo (do ponto de vista do agente)

SP: Exemplo 1 (4 sensores)



- O comportamento pretendido é o de deambular pelo ambiente evitando os obstáculos

SP: Exemplo 1 (4 sensores)

- Conjunto de ações:

  {norte, este, sul, oeste}

  que significam movimentar-se 1 posição na direção indicada

- Existe 1 conflito: o agente pode deslocar-se para qualquer uma das direções

SP: Exemplo 1 (4 sensores)

- Mecanismo de resolução de conflitos:

  Deve estabelecer que o agente prefere deslocar-se primeiro para Norte, depois para Este, depois para Sul e só no fim para Oeste

SP: Exemplo 1 (4 sensores)

- Descrição da função ação

```
Função Acção (Percepção): acção
    1. Se percepção = ¬O1 Então devolve(norte)
            1.1. Senão_Se percepção = ¬O2 Então devolve(este)
            1.2. Senão_Se percepção = ¬O3 Então devolve(sul)
            1.3. Senão_Se percepção = ¬O4 Então devolve(oeste)
        Fim_de_Se
Fim_de_Função
```

Descrição da função Acção

SP: Exemplo 1 (4 sensores)

- Sistemas de produções simples

Deambular1:

1. $\neg O_1 \rightarrow N$
2. $\neg O_2 \rightarrow E$
3. $\neg O_3 \rightarrow S$
4. $\neg O_4 \rightarrow O$

SP: Exemplo 1 (4 sensores)

- Sistemas de produções simples

Deambular2:

1. $\neg O_1 \rightarrow N$
2. $\neg O_2 \rightarrow E$
3. $\neg O_3 \rightarrow S$
4. $\neg O_4 \rightarrow O$
5. $T \rightarrow NIL$ (não fazer nada)

SP: Exemplo 1 (4 sensores)

- Sistemas de produções simples

Deambular3:

1. $\neg O_1, \neg P_1 \rightarrow N, P_{ag}$
2. $\neg O_2, \neg P_2 \rightarrow E, P_{ag}$
3. $\neg O_3, \neg P_3 \rightarrow S, P_{ag}$
4. $\neg O_4, \neg P_4 \rightarrow O, P_{ag}$
5. $T \rightarrow NIL$ (não fazer nada)

(P_i – indicador de passagem pela posição i)

SP: Exemplo 1 (4 sensores)

- Estes casos mostram algumas dificuldades do projeto quando o agente tem capacidades limitadas

SP: Exemplo 1 (4 sensores)

- Vejamos um caso mais realista, em que o agente tem um objetivo que pretende realizar: localizar e apanhar um objeto importante (ouro, por exemplo)

- Esse objetivo é prioritário, passando a ser a 1.ª produção da lista

SP: Exemplo 1 (4 sensores)

- Sistema de produções simples

Procura_Ouro:

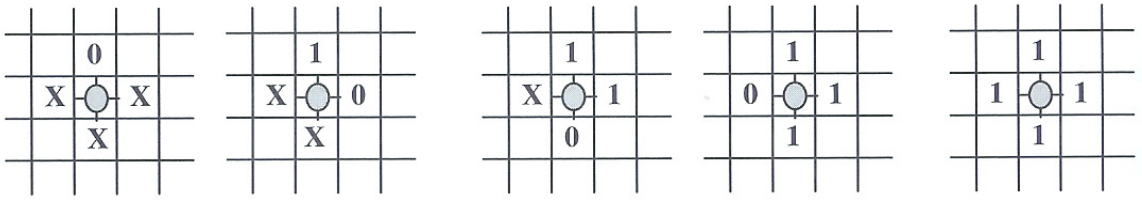
1. Ouro → NIL
2. $\neg O_1$, $\neg P_1$ → N, P_{ag}
3. $\neg O_2$, $\neg P_2$ → E , P_{ag}
4. $\neg O_3$, $\neg P_3$ → S , P_{ag}
5. $\neg O_4$, $\neg P_4$ → O , P_{ag}
6. T → NIL (não fazer nada)

SP: Exemplo 1 (4 sensores)

- Nota final: Com os 4 sensores o agente pode captar 16 estados diferentes, a que correspondem outras tantas perceções

- No entanto, de acordo com o projeto, 8 desses estados originam como resposta a ação Norte, 4 a ação Este, 2 a ação Sul, 1 a ação Oeste e 1 a ação NIL

SP: Exemplo 1 (4 sensores)



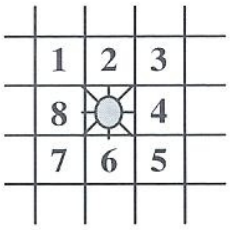
Classes de comportamento do agente

- Isto permite compreender o comportamento do agente que induz rotações pela direita

SP: Exemplo 2 (8 sensores)

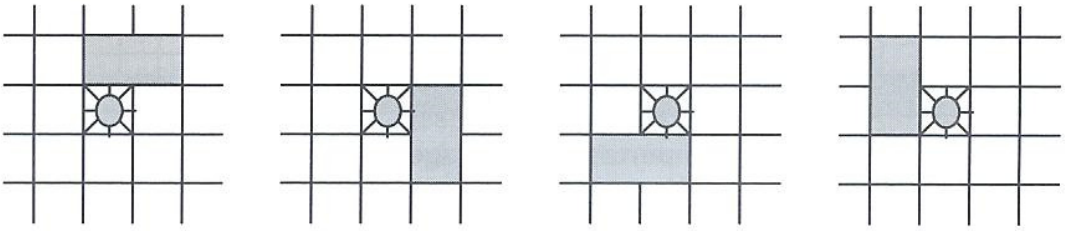
- Suponhamos que neste segundo exemplo o agente tem 8 sensores: s_1 a s_8
- Suponhamos ainda que o agente também só pode movimentar-se para as células vizinhas que se encontram na mesma linha ou na mesma coluna
- Tarefa do agente: uma vez encontrada a parede limitadora do mundo ou a parede de um obstáculo, o agente deve passar a seguir essa parede.

SP: Exemplo 2 (8 sensores)



Agente com oito sensores

- Característica x: indica se pelo menos 1 das 2 posições que cobre está ocupada



- x_1=1 significa que s_2=1 ou s_3=1

SP: Exemplo 2 (8 sensores)

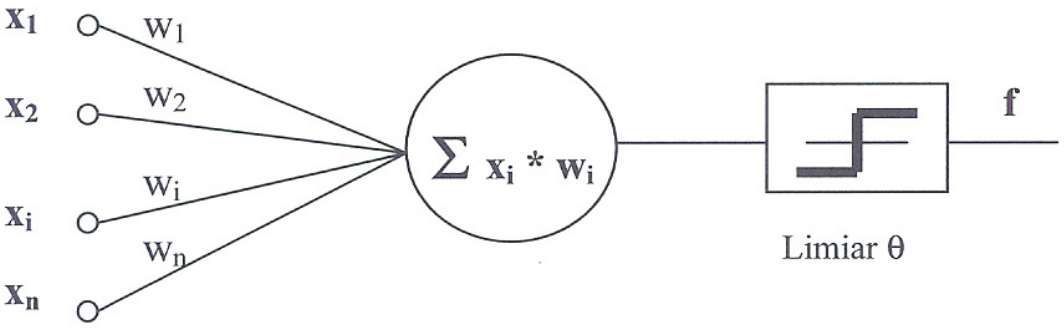
- Sistema de produções simples

Segue_paredes:

1. $x_4, \neg x_1 \rightarrow N$
2. $x_1, \neg x_2 \rightarrow E$
3. $x_2, \neg x_3 \rightarrow S$
4. $x_3, \neg x_4 \rightarrow O$
5. $T \rightarrow N$

Unidades Lineares de Limiar (TLU)

- TLU – *Threshold Linear Unit*
- Uma TLU é uma versão computacional do neurónio biológico



Neurónio artificial segundo McCulloch e Pitts

Unidades Lineares de Limiar (TLU)

- Existe um conjunto de entradas que têm pesos associados
- É calculada a soma pesada das entradas que, se ultrapassar um valor limiar de referência, faz com que o neurónio dispare, produzindo um sinal à sua saída
- De um modo geral, quando uma TLU dispara a sua saída assume o valor 1. Caso contrário, assume o valor 0 (estes valores dependem da função de ativação utilizada)

Unidades Lineares de Limiar (TLU)

- Existem vários tipos de redes de neurónios artificiais (Redes Neuronais Artificiais). Uma classificação possível envolve 3 aspetos:

  - o modo como os neurónios se ligam entre si (topologia)

  - como ficam ativos ou inativos (função de ativação)

  - a forma como modificam alguns dos seus elementos (dinâmica da rede)

Unidades Lineares de Limiar (TLU)

- As TLU podem ser usadas para implementar uma produção
- Consideremos o exemplo do agente segue_paredes e a regra:

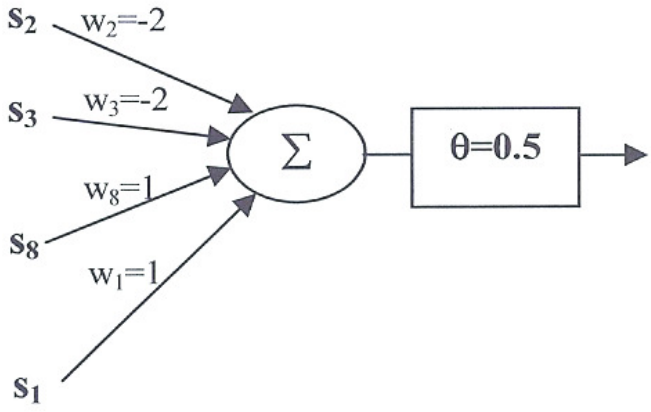
$$x_4, \neg x_1 \rightarrow \text{norte}$$

x_4 será verdadeiro se pelo menos um dos sensores s_8 ou s_1 for 1

$\neg x_1$ será verdadeiro se os dois sensores s_2 e s_3 forem 0

Unidades Lineares de Limiar (TLU)

- Assim, a seguinte TLU apenas terá o valor 1 caso x_4 e $\neg x_1$ sejam simultaneamente verdadeiros



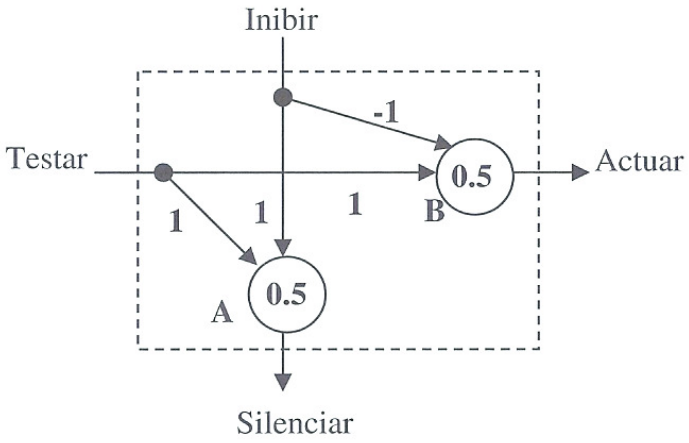
TLU para a regra 1

Unidades Lineares de Limiar (TLU)

- Mas como implementar um mecanismo de controlo e de resolução de eventuais conflitos entre regras ?
- No exemplo anterior, pretende-se que o mecanismo implemente o ordenamento das regras e que faça com que a primeira a ser ativada possa ser disparada.
- Uma possível solução baseia-se em unidades TISA

Unidades Lineares de Limiar (TLU)

- Unidade TISA (Testar, Inibir, Silenciar, Atuar – *Test*, *Inhibit*, *Squelch*, *Act*)



Estrutura de uma unidade TISA

Unidades Lineares de Limiar (TLU)

- Uma unidade TISA é constituída por duas TLU de 2 entradas
- A TLU A implementa um OU Lógico das suas entradas

  A saída Silenciar só será 0 se as suas 2 entradas forem simultaneamente 0

Unidades Lineares de Limiar (TLU)

- No caso da TLU B: se a entrada Testar estiver a 1 então a saída Atuar também será 1, desde que a entrada Inibir não esteja também a 1
- Ou seja, desde que Inibir ou Testar estejam ativos a saída Silenciar fica ativa, enquanto a entrada Inibir controla a passagem do sinal da entrada Testar para a saída Atuar.

Unidades Lineares de Limiar (TLU)

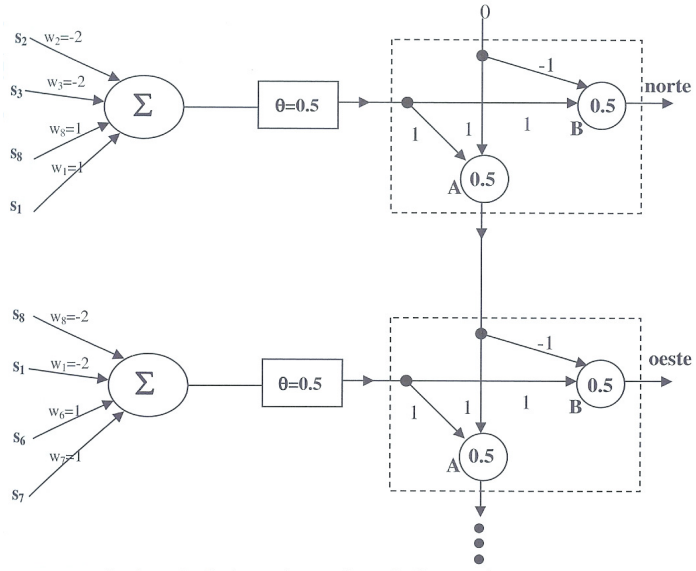
- Para implementar o sistema de produções do exemplo anterior,

  - ligar a saída de uma TLU que implementa uma produção à entrada Testar de uma unidade TISA

  - ligar a saída Silenciar à entrada Inibir da produção que a segue na sequência (e a sua entrada Inibir à saída Silenciar da produção que a antecede na sequência)

Unidades Lineares de Limiar (TLU)

- Para o exemplo do agente segue_paredes (apenas as 2 primeiras regras)



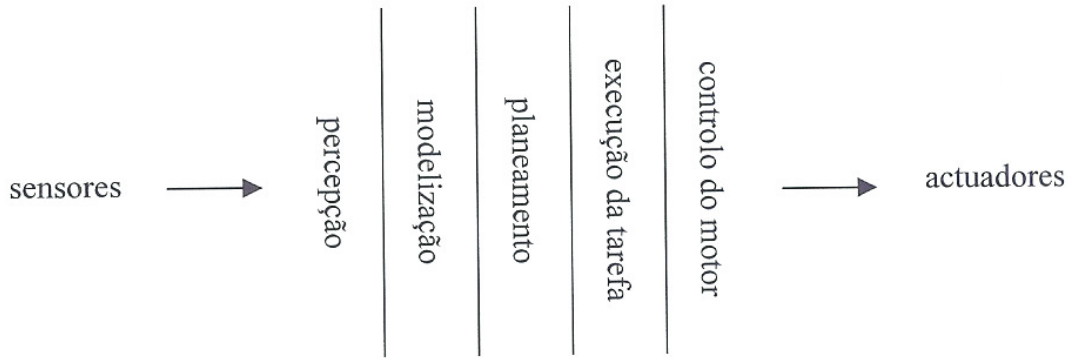
Implementação de um sistema de produções através de unidades TISA

Nota: a entrada Inibir da primeira regra está sempre a 0

Arquitetura de Subordinação

- Esta abordagem (abordagem ao problema de ligar a entrada de um agente à respetiva saída), pensada para a robótica móvel, contrapõe à arquitetura vertical da IA tradicional uma arquitetura horizontal.

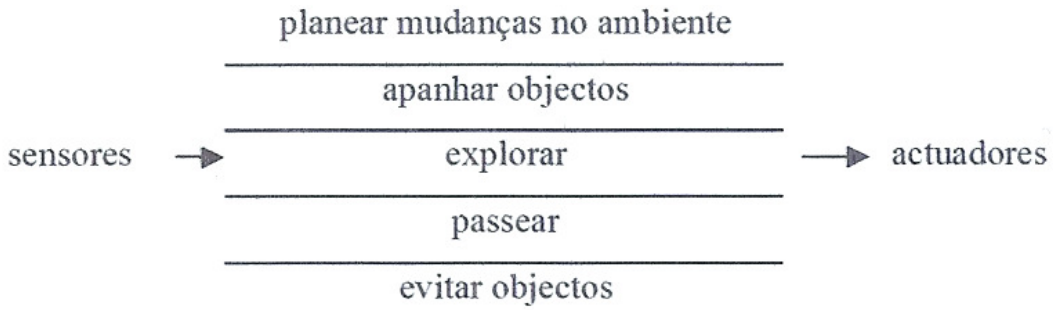
Arquitetura de Subordinação



Exemplo da representação por camadas na IA clássica

- A ligação entre as sensações e as ações é feita por camadas, constituíndo cada uma uma unidade funcional.

Arquitetura de Subordinação



Exemplo da arquitectura de subordinação

- Esta arquitetura está organizada em torno de comportamentos voltados para a concretização de tarefas, ligando diretamente as entradas do agente às suas saídas.

Arquitetura de Subordinação

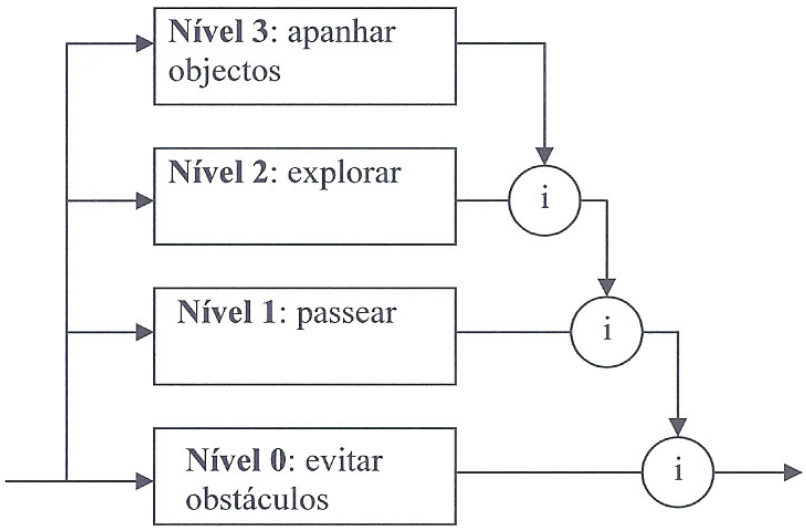
- O projeto de um agente começa pela especificação dos comportamentos pretendidos e a sua diferenciação em níveis de competência.
- Estes níveis de competência podem ser básicos, como *"evitar objetos"*, ou mais complexos, como *"apanhar objetos"*
- Cada nível é implementado como uma camada de controlo.

Arquitetura de Subordinação

- As diferentes camadas podem comunicar através de sinais de supressão (s) dos sinais de entrada ou de inibição (i) dos sinais de saída.

- Exemplo: Se o objetivo pretendido for o agente apanhar um objeto, a camada que implementa este comportamento terá de inibir a camada que implementa o comportamento de evitar obstáculos.

Arquitetura de Subordinação



Arquitectura de subordinação: implementação

- As camadas superiores subordinam as camadas inferiores.

Agentes Reativos com Memória

- Neste ponto veremos como limitações nas capacidades sensoriais dos agentes reativos podem, em parte, ser ultrapassadas pela inclusão de um mecanismo de memória.

Arquitetura Abstrata

- Os agentes podem basear a sua decisão na sua história.
- Uma maneira de tratar esta característica é incluir na arquitetura do agente a capacidade de manter uma descrição do estado do ambiente, presente e passado.

Arquitetura Abstrata

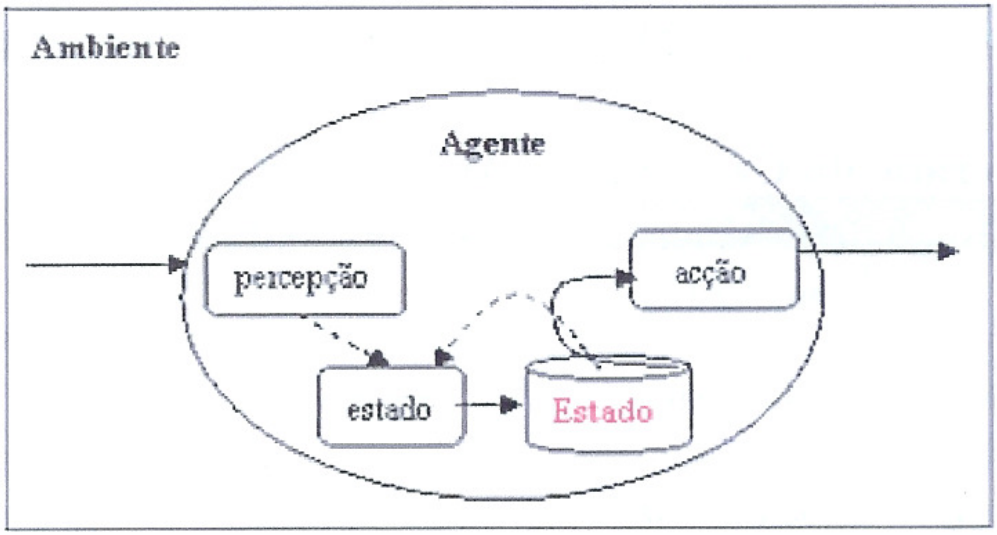
- Seja I o conjunto dos estados (internos) do agente.
- Teremos

$$\text{perceção: } E \rightarrow P$$

$$\text{estado: } I \times P \rightarrow I$$

$$\text{ação: } I \rightarrow A$$

Arquitetura Abstrata



Agente reativo com memória

Arquitetura Abstrata

O comportamento do agente baseado no estado interno pode agora ser descrito do seguinte modo:

- ✓ Admitamos que o agente se encontra num dado estado interno i;
- ✓ O agente observa o estado do ambiente, e, e produz a perceção perceção(e);
- ✓ O estado interno é atualizado pela função estado(i,perceção(e));
- ✓ A ação selecionada será ação(estado(i,perceção(e)));
- ✓ A seguir, um novo ciclo recomeça.

Arquitetura Abstrata

- Esta arquitetura abstrata pode ser traduzida por um programa simples:

```
função agente_reactivo_estado_interno (estado_ambiente) : acção
  1. percepção ← percepção(estado_ambiente);
  2. estado_interno ← estado(estado_interno, percepção);
  3. acção ← acção(estado_interno)
fim_de_função
```

Descrição simplificada de um agente reactivo com memória

Arquitetura Abstrata

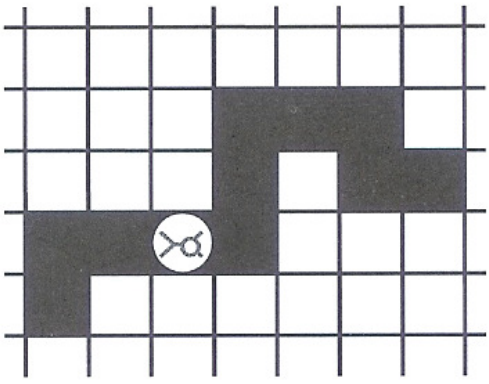
- O estado interno funciona como memória da história das interações entre o agente e o ambiente.
- O agente funciona como uma máquina de estados que vai alterando o seu estado em função do seu estado atual e da informação que recebe do ambiente.

Implementação: FormArt

- Exemplo de implementação de um agente com estado interno: FormArt
- FormArt é uma formiga artificial que vive num mundo 2D aberto
- A tarefa da formiga é seguir um trilho contínuo de feromona
- Cada "pedaço" do trilho ocupa uma célula, tal como a formiga

Implementação: FormArt

- A formiga pode estar direcionada para cima, para baixo, para a esquerda ou para a direita



Exemplo da formiga Formart

Implementação: FormArt

- A formiga pode estar em 2 estados: ativa (On) ou inativa (Off)
- A formiga pode efetuar uma de 5 ações:
  - **A** – avançar uma posição no sentido para que se encontra virada
  - **E** – rodar 90º para a esquerda
  - **D** – rodar 90º para a direita
  - **On** – passar a ativa
  - **Off** – passar a inativa

Implementação: FormArt

- As suas capacidades de perceção limitam-se a detetar a presença de feromona na célula à sua frente:
  - F – presença de feromona
  - ¬F – ausência de feromona
- Também pode determinar se está ativa ou inativa.

Implementação: FormArt

- O mecanismo de controlo da formiga artificial pode ser especificado por intermédio de um sistema de produções:

Segue_trilho:

1. Off, F → A, Off
2. Off, ¬F → E, On
3. On, F → A, Off
4. On, ¬F → D, On

Implementação: FormArt

- O estado interno (On/Off) permite ter informação sobre o passado, isto é, sobre o facto de a formiga andar (On) ou não (Off) a mudar de direção.

Implementação: FormArt

O seu comportamento é o seguinte:

- Enquanto deteta feromona na sua frente a formiga avança (regra 1)
- Quando deixa de sentir, muda de estado e roda à esquerda (regra 2)
- Se mudou de direção e foi bem sucedida, volta ao estado anterior e avança (regra 3)
- Caso tenha sido mal sucedida, volta atrás (ou seja, à direita) e mantém o estado de procura (regra 4)

  Como o trilho é contínuo, é forçoso que na sua 2.ª tentativa encontre feromona, aplicando-se a regra 3.

Implementação: FormArt

- Sendo colocada inicialmente sobre o trilho e com feromona à sua frente, a FormArt nunca sai do trilho
- Também é fácil verificar que nunca volta para trás
- O estado interno funciona como memória e impede que a formiga entre em ciclo

Limitações Sensoriais e Estado Interno

- Em que medida a existência de um estado interno permite ultrapassar limitações sensoriais dos agentes ?

Limitações Sensoriais e Estado Interno

- Considerando o agente segue_paredes e admitindo que o agente
  - Apenas tem acesso direto às posições 2, 4, 6 e 8 (só tem 4 sensores);
  - Consegue memorizar o valor dos sensores s_2, s_4, s_6 e s_8 no instante anterior (t-1), informação esta que pode ser utilizada para determinar o valor dos outros sensores (sensores em falta) s_1, s_3, s_5 e s_7 no instante actual (t).

Limitações Sensoriais e Estado Interno

- Ou seja,

  $x_1(t)=1$ sse $x_2(t-1)=1$ e ação=*este*;

  $x_3(t)=1$ sse $x_4(t-1)=1$ e ação=*sul*;

  $x_5(t)=1$ sse $x_6(t-1)=1$ e ação=*oeste*;

  $x_7(t)=1$ sse $x_8(t-1)=1$ e ação=*norte*;

Limitações Sensoriais e Estado Interno

- Sistema de produções completo

Segue_paredes2:

1. x_8 , $\neg x_2$ → N
2. x_2, $\neg x_4$ → E
3. x_4, $\neg x_6$ → S
4. x_6 , $\neg x_8$ → O
5. x_1 → N
6. x_3 → E
7. x_5 → S
8. x_7 → O
9. T → N