

Introdução à Inteligência Artificial

Ano 2005/06 – Época recurso

3 horas / com consulta

Grupo 1 (Pesquisa)

Uma empresa de distribuição comercializa n produtos, pretendendo construir n armazéns em n regiões. De acordo com a política da empresa, cada região tem um único armazém, cada armazém só pode armazenar um e um só produto, e cada produto está num e num só armazém. Formalmente, este problema pode ser representado através de 3 conjuntos, todos de cardinalidade n , designados P (produtos), A (armazéns), R (regiões). Adicionalmente, existe um conjunto Q de quádruplos da forma (p_i, a_j, r_k, c_l) indicando que o custo de construir o armazém a_j para armazenar o produto p_i na região r_k é de c_l unidades. O custo é sempre um número inteiro positivo. O objectivo consiste então em encontrar um subconjunto S de Q tal que:

1. Cada elemento de P , A e R ocorre uma e uma só vez num quádruplo de S .
2. Minimiza a soma dos custos de todos os quádruplos em S .

Para exemplificar, considere a seguinte instância com 3 produtos, 3 armazéns e 3 regiões:

$P = \{p_1, p_2, p_3\}$	$Q = \{ (p_1, a_1, r_1, 10), (p_2, a_3, r_2, 5), (p_3, a_2, r_3, 20), (p_1, a_2, r_2, 50), (p_1, a_3, r_1, 60), (p_2, a_2, r_1, 40) \}$
$A = \{a_1, a_2, a_3\}$	
$R = \{r_1, r_2, r_3\}$	

Para este caso a solução óptima é obtida pelo conjunto $\{(p_1, a_1, r_1, 10), (p_2, a_3, r_2, 5), (p_3, a_2, r_3, 20)\}$, com custo total de 35 unidades.

1a) Formule claramente o problema para ser resolvido recorrendo a algoritmos de pesquisa em espaço de estados, indicando o estado inicial, teste de estado objectivo e função que devolve os sucessores de um estado, não esquecendo de indicar o custo dos operadores.

1b) Apresente uma heurística não constante que garanta a obtenção de uma solução óptima pelo algoritmo A^* . Não é necessária a apresentação de qualquer expressão matemática, desde que fique inteiramente clara a forma como a heurística pode ser obtida. Justifique adequadamente.

1c) Indique se este problema poderá ser resolvido pelo algoritmo de “simulated annealing”. Em caso afirmativo, descreva uma modelação do problema apropriada para utilização do referido algoritmo.

Grupo 2 (Redes de Bayes)

Uma selecção participante no mundial de futebol recorreu a redes de Bayes para calcular a probabilidade de ser campeã. A selecção tem 60% de probabilidade de pertencer a um grupo de dificuldade média e 40% de ter dificuldade elevada (não existem grupos fáceis no mundial...). Se o grupo tiver dificuldade média então a probabilidade de ficar em primeiro lugar no grupo é de 80% e de ficar em segundo lugar é de 20%. Contudo, se o grupo tiver dificuldade elevada a probabilidade de ficar em primeiro lugar no grupo é de 30%, em segundo 30% e 40% de ficar qualificada noutro lugar. A probabilidade da selecção estar em forma é de 10%. A probabilidade da selecção chegar à final dado que fica em primeiro lugar no grupo e está em forma é de 40%, sendo o valor de 30% quando fica em segundo lugar e está em forma. Quando a selecção não está em forma então a probabilidade de chegar à final é de 10%, independentemente de ficar em primeiro ou segundo lugar no grupo. Pelas regras do campeonato do mundo, qualquer selecção que não fique em primeiro ou segundo lugar do respectivo grupo não pode participar na fase de eliminatórias, e portanto não pode chegar à final. Obviamente, se a selecção não chegar à final não pode ser campeã de futebol. A probabilidade da selecção ser campeã dado que chega à final e está em forma é de 50%. Contudo, se chega à final e não está em forma, a probabilidade de ser campeã é apenas de 30%.

2a) Modele a situação anterior com uma rede de Bayes, indicando as variáveis aleatórias, seus domínios, topologia da rede e tabelas de probabilidade condicionada.

2b) Calcule a probabilidade da selecção simultaneamente estar em forma, pertencer a um grupo de dificuldade média, chegar à final e não ser campeã.

2c) Calcule a probabilidade da selecção ficar em primeiro lugar no grupo sabendo que não é campeã, que está em forma e que pertence a um grupo de dificuldade média.

Grupo 3 (Planeamento)

O Sr. Viz, cidadão indiano, coleciona visitas a países. O objectivo para este ano é visitar Portugal e Espanha, países da Europa integrados no espaço de Schengen. Como o Sr. Viz é um cidadão honesto, para obter um visto para um país do espaço de Schengen basta não ter um visto para esse país; não é proibido ter vistos para diferentes países do espaço de Schengen. Para se poder entrar num país do espaço de Schengen tem de se ter um visto para esse país e obviamente estar fora do espaço de Schengen. Qualquer pessoa pode-se deslocar entre quaisquer dois países do espaço de Schengen. Para se poder sair do espaço de Schengen basta ter um visto para qualquer país do espaço de Schengen.

3a) Modele na linguagem STRIPS as acções **obterVisto(P1)**, **entrar(P1)**, **deslocar(P1,P2)** e **sair(P1)**, em que P1 e P2 são países do espaço de Schengen. A sua modelação deve estar de acordo com a descrição no enunciado e recorrer à sintaxe STRIPS utilizada nas aulas práticas. Para simplificar, pode assumir que o Sr. Viz visitou um país quando entra ou se desloca para esse país.

3b) Considere a situação inicial em que o Sr. Viz se encontra fora do espaço de Schengen e sem visto para Portugal e Espanha. Construa um plano POP com o menor número de passos que lhe permita visitar Portugal e Espanha e no fim ficar fora do espaço de Schengen. No plano construído inclua todas as restrições de ordem temporal introduzidas por resolução de ameaças, mesmo aquelas que entenda redundantes. Apresente todas as linearizações desse plano POP.

3c) Nas mesmas condições da alínea anterior, indique se é possível com a sua representação construir planos POP em que o Sr. Viz obtenha visto para Portugal e Espanha. Em caso afirmativo, pode obter planos com mais do que uma linearização? Justifique detalhadamente.

Grupo 4 (Representação do conhecimento)

O puzzle “Light Up” consiste em colocar lâmpadas numa grelha quadrada ou rectangular, constituída por casas brancas e pretas. Algumas das casas pretas podem estar numeradas. As regras do puzzle são as seguintes:

1. As lâmpadas (círculos) só podem ser colocadas em casas brancas.
2. Cada lâmpada ilumina todas as casas brancas na sua linha e na sua coluna, até encontrar uma casa preta (ou atingir os limites da grelha).
3. Todas as casas brancas devem ficar iluminadas.
4. Nenhuma lâmpada pode ser iluminada por outra lâmpada.
5. Uma casa numerada indica quantas lâmpadas lhe estão adjacentes (verticalmente e horizontalmente).

Considere-se o seguinte puzzle (à esquerda) e respectiva solução (à direita). Encontra-se já definido um sistema de coordenadas para facilitar a resolução deste problema.

6	2					1
5						
4						
3						
2			2			
1						1
	1	2	3	4	5	6

6	2	O				1
5	O					O
4						
3				O		
2			O	2		
1				O		1
	1	2	3	4	5	6

Recorrendo à programação por conjuntos de resposta, irá produzir neste grupo um programa para o sistema Smodels que lhe permitirá resolver este tipo de puzzles. Considere o seguinte programa, capturando alguma da informação enunciada:

h(1..6).	numerada(6,1,1).	preta(2,3).
v(1..6).	numerada(4,2,2).	preta(5,4).
n(1..4).	numerada(1,6,2).	preta(3,5).
	numerada(6,6,1).	preta(X,Y) :- numerada(X,Y,N).

O predicado **numerada(X,Y,N)** lista as casas pretas numeradas, em que **X** e **Y** são as coordenadas da casa e **N** o número (de 1 a 4) que aí se encontra. O predicado **preta(X,Y)** indica quais as casas pretas, numeradas ou não.

Nas seguintes questões aconselha-se a utilização do predicado **lamp(X,Y)** para representar, num conjunto de resposta, que existe uma lâmpada na casa **(X,Y)**. Para simplificar a resolução pode ignorar as restrições impostas pelas casas numeradas (ponto 5).

4a) Implemente o predicado **branca(X,Y)** que indica quais as casas brancas da grelha. Deve utilizar o predicado **preta/2** na sua definição.

4b) Apresente um programa que gere conjuntos de resposta em que as casas brancas podem ter ou não uma lâmpada.

4c) É necessário um predicado auxiliar **ilum(X,Y,Lx,Ly)** que é verdadeiro num conjunto de resposta quando a casa **(X,Y)** é iluminada por uma lâmpada na casa **(Lx,Ly)**. Este predicado deve ser recursivo e o caso base pode ser encontrado abaixo. Deve completar o predicado de acordo com o descrito no ponto 2 das regras do puzzle “Light Up.”

$$\text{ilum}(Lx, Ly, Lx, Ly) \text{ :- branca}(Lx, Ly), \text{ lamp}(Lx, Ly).$$

4d) Apresente uma ou mais restrições que capturem a informação descrita no ponto 3 do enunciado. Repare que não deve impedir que uma casa seja iluminada por mais do que uma lâmpada.

4e) Adicione ao seu programa uma ou mais restrições garantindo que nenhuma lâmpada é iluminada por outra lâmpada.

FIM