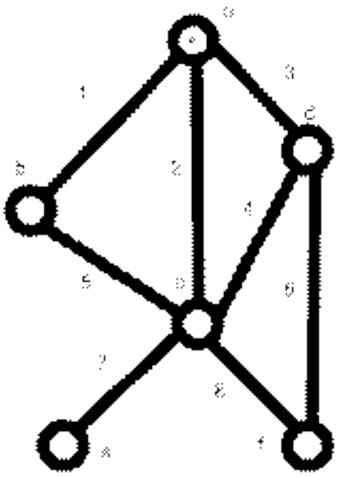
Introdução à Inteligência Artificial

Ano 2004/05 – recurso

3 horas / com consulta

Grupo 1 (Pesquisa)



A última onda de atentados terroristas tem preocupado a opinião pública mundial. Assim, o governo decidiu adoptar a máxima de "um polícia em cada rua" para sossegar os portugueses. Considere o grafo não orientado apresentado na figura, em que os vértices estão identificados por letras e os arcos por números. Cada um dos vértices indica um local onde pode ser colocado um polícia e os arcos uma rua a ser policiada. Dada a contenção orçamental, o objectivo consiste em colocar o menor número de polícias de maneira que toda a rua fique vigiada pelo menos por um polícia numa das suas extremidades.

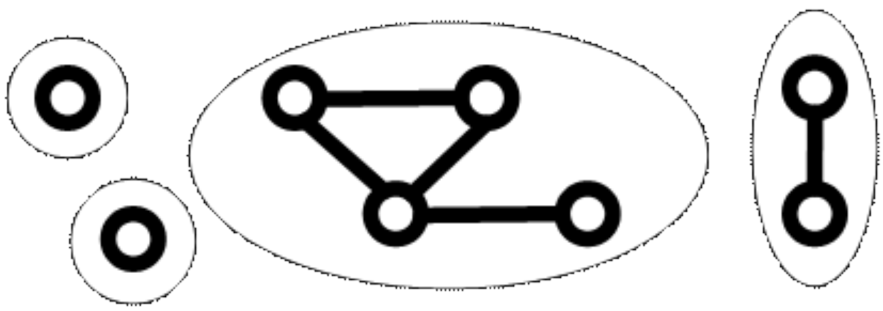
Por exemplo, uma solução óptima recorre apenas a 3 polícias, colocados nos vértices **a**, **c** e **d**. Repare que um polícia em **a** cobre as ruas 1, 2, e 3; um polícia em **c**, vigia as ruas 2, 4, 5, 7 e 8; enquanto que o polícia em **d**, trata das ruas 3, 4 e 6. Portanto, estamos na presença de uma solução pois todas as ruas se encontram vigiadas pelo menos por um polícia, e algumas delas até por dois agentes da autoridade. Ao adicionarmos um ou mais vértices ao conjunto anterior, continuaremos a obter uma solução mas que obviamente já não será óptima.

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) De entre as funções listadas de seguida, indique quais podem ser utilizadas como heurística pelo algoritmo A*, de forma a garantir a obtenção de uma solução óptima. Justifique cuidadosamente a sua resposta.

h1) Número de ruas ainda por vigiar ;
h2) Número de componentes conexas do sub-grafo contendo todos os vértices não ocupados e todas as as ruas ainda por vigiar;
h3) O número de ruas por vigiar dividido pelo número de ruas do vértice com mais ruas;
h4) Número de vértices com ruas ainda não vigiadas.

NOTA: Uma componente conexa de um grafo não orientado é formada pelo maior conjunto de vértices para os quais existe um caminho entre quaisquer dois vértices desse conjunto. Um vértice que não tenha arcos, pertence a uma componente formada só por esse vértice. No grafo seguinte temos 4 componente conexas:



1c) Apresente uma heurística não constante que garanta a obtenção de uma solução óptima pelo algoritmo A* e que não seja idêntica às heurísticas da alínea anterior. Não é necessária a apresentação de qualquer expressão matemática, desde que fique inteiramente clara a forma como a heurística pode ser obtida. Justifique sucintamente.

1d) Comente a seguinte afirmação: "*Uma pesquisa em profundidade primeiro permite encontrar uma solução para o problema anterior em tempo linear no número de vértices, apesar de não haver garantia de se obter a solução óptima*". Justifique.

Grupo 2 (Planeamento)

A Guida Gadachim vai efectuar uma viagem de avião e pretende levar alguns objectos com ela: batom, um pente e um corta-unhas. Um objecto que seja despachado no check-in é transportado no avião, na bagagem de porão, e portanto não fica acessível. Um objecto que seja colocado na bagagem de mão fica acessível. Contudo, os objectos colocados na bagagem de mão são sujeitos a um controlo de Raios-X, só sendo transportados no avião os objectos da bagagem de mão que sejam seguros. O pente e o batom são objectos seguros.

2a) Apresente um conjunto de acções na linguagem de planeamento STRIPS que permitam modelar a situação descrita no enunciado. A sua modelação deverá tratar pelo menos os fluentes **noAvião(Objecto)**, **bagagemPorão(Objecto)** e **bagagemMão(Objecto)**. Poderá utilizar fluentes adicionais, caso entenda necessário.

2b) Especifique completamente o estado inicial e os objectivos de maneira a que a Guida Gadachim transporte no avião o batom e o corta-unhas, devendo o batom ficar acessível. Recorrendo ao algoritmo POP construa e apresente graficamente um plano com ordem parcial, indicando também as respectivas linearizações possíveis.

2c) Verifique se a sua modelação permite obter sequências de acções em que a Guida despacha a bagagem de porão depois de ter passado no controlo de Raios-X, com os mesmos objectivos especificados na alínea anterior. Em caso afirmativo, apresente as alterações que efectuaria à sua modelação para evitar a construção de planos em que tal aconteça. Justifique detalhadamente como é que o algoritmo POP impede agora a geração dos referidos planos.

Grupo 3 (Redes de Bayes)

Dos 180 alunos inscritos à disciplina de IIA, 36 realizaram o exame de 1ª chamada enquanto que 90 optaram pelo exame de 2ª chamada, tendo os restantes faltado. Sabe-se ainda que exactamente 25% dos alunos teve nota prática superior a 16 valores. Dos alunos com nota prática superior a 16 valores, 80% que realizou a primeira chamada teve aprovação, subindo a percentagem de aprovação para 90% na segunda chamada. Para os alunos com nota prática inferior ou igual a 16 valores, 40% ficou aprovado na primeira chamada e 60% na segunda chamada. Obviamente, os alunos que faltaram encontram-se automaticamente reprovados. Nenhum dos alunos reprovados e com nota prática superior a 16 realizou oral. Contudo, a percentagem de alunos que realizou oral já foi de 10% para aqueles com nota prática superior a 16 valores e que obtiveram aprovação. Nos restantes casos, 5% realizou oral.

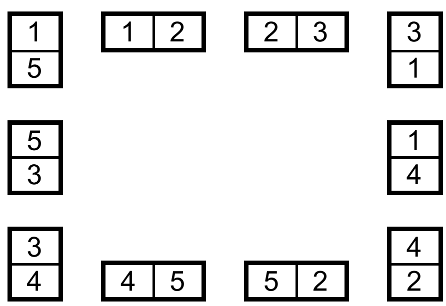
3a) Modele a situação anterior com uma rede de Bayes, indicando as variáveis aleatórias, seus domínios, topologia da rede e tabelas de probabilidade condicionada.

3b) Calcule a probabilidade de um aluno ter aprovação **e** de ter nota prática superior a 16 valores.

3c) Calcule a probabilidade de um aluno ter realizado a 1ª chamada dado que a nota prática foi superior a 16 valores e não realizou oral.

3d) Justifique, sem efectuar quaisquer cálculos, que "*a probabilidade de realizar oral dado que foi aprovado em exame da primeira chamada e que a nota prática é superior a 16 valores é igual à probabilidade de realizar oral dado que foi aprovado em exame e que a nota prática é superior a 16 valores.*"

Grupo 4 (Representação do Conhecimento)



Considere um conjunto de 10 peças de dominó, em que não existem peças duplas (com o mesmo número de pintas em ambos os

lados do dominó):
1-2, 1-3, 1-4, 1-5, 2-3, 2-4, 2-5, 3-4, 3-5 e 4-5. O quebra-cabeças consiste em colocar todas as peças de maneira a formar um quadrado, umas a seguir às outras, de acordo com as regras do dominó (ver figura). Recorda-se que no dominó só se podem colocar duas peças ao lado uma da outra se tiverem o mesmo número de pintas nos lados que ficam contíguos.

A utilização de uma grelha rectangular só complica o problema. Equivalentemente, podemos imaginar que as peças são colocadas em sequência, mas obrigando a que os valores das pintas nos extremos da sequência coincidam. Por exemplo, o quadrado da figura pode ser entendido como a seguinte sequência de peças:

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

Repare-se que uma peça de dominó pode ser colocada numa determinada posição, no máximo de duas maneiras distintas (número de pintas a crescer ou número de pintas a decrescer). A programação por conjunto de respostas presta-se à resolução (eficiente) deste tipo de problemas, sendo essa a sua tarefa neste grupo. A estratégia para resolver este problema passa pela geração de alternativas em que todas as peças de dominó se encontram colocadas numa posição (de 1 a 10), a crescer ou a decrescer. Obviamente, não se poderão colocar duas peças na mesma posição. Como ponto de partida, considere o seguinte conjunto de regras e de factos:

```
peça(a).     dominó(a,1,2).    n(1).     pintas(1).    cd(cres).
peça(b).     dominó(b,1,3).    n(2).     pintas(2).    cd(decr).
peça(c).     dominó(c,1,4).    n(3).     pintas(3).
peça(d).     dominó(d,1,5).    n(4).     pintas(4).
peça(e).     dominó(e,2,3).    n(5).     pintas(5).
peça(f).     dominó(f,2,4).    n(6).
peça(g).     dominó(g,2,5).    n(7).
peça(h).     dominó(h,3,4).    n(8).
peça(i).     dominó(i,3,5).    n(9).
peça(j).     dominó(j,4,5).    n(10).
```

Nas seguintes questões aconselha-se a utilização do predicado **pos(P,N,CD)** para representar, num conjunto de resposta, que a peça **P** é colocada na posição **N**, a crescer ou a decrescer de acordo com **CD**. Por exemplo, deveremos ter um conjunto de resposta a que pertençam `pos(a,1,cres)`, `pos(e,2,cres)`, `pos(b,3,decr)`, `pos(c,4,cres)`, `pos(f,5,decr)`, `pos(g,6,cres)`, `pos(j,7,decr)`, `pos(h,8,decr)`, `pos(i,9,cres)`, `pos(d,10,decr)`, entre outros, correspondendo à solução apresentada anteriormente.

4a) Apresente um programa cujos conjuntos de resposta tenham uma peça de dominó colocada numa e numa só posição, a crescer ou a decrescer (i.e. para uma dada peça P, um e um só `pos(P,N,CD)` pertence a cada conjunto de resposta).

4b) Claramente, uma posição não pode ser ocupada por duas peças distintas. Indique uma restrição que elimine conjuntos de resposta em que ocorra esta situação indesejável.

4c) Para facilitar a tarefa da próxima alínea, onde irá garantir que as peças são colocadas correctamente ao lado umas das outras, nesta alínea terá que implementar os predicados auxiliares **pintasNaEsquerda(N,V)** e **pintasNaDireita(N,V)**. Num conjunto de resposta, **pintasNaEsquerda(N,V)** (respectivamente, **pintasNaDireita(N,V)**) indica que a posição **N** tem **V** pintas do lado esquerdo (respectivamente, do lado direito). Por exemplo, para a sequência apresentada, deveremos ter `pintasNaEsquerda(9,3)`, `pintasNaDireita(9,5)`, `pintasNaEsquerda(10,5)` e `pintasNaDireita(10,1)` pertencentes ao conjunto de resposta, para além dos relativos às restantes posições.

4d) Para completar o programa, deve agora fazer coincidir o número de pintas em lados adjacentes de peças contíguas, não se esquecendo dos extremos. Explicite as restrições que adicionaria para obrigar a que tal aconteça. Se entender necessário, poderá recorrer a predicados auxiliares.