

# Inteligência Artificial

## Primeira Lista de Exercícios – Gabarito

Prof. Norton Trevisan Roman

11 de março de 2019

1. (a) *Espaço de estados*: podemos representar um estado como um vetor de dois elementos: o primeiro indica o local em que o robô está, e o segundo a lista de pedras que ele está carregando: [local, pedras], onde:

- local pode ser Módulo (M), local 1 (L1), local 2 (L2) ou local 3 (L3)
- pedras pode ser , p1,,, ,p2,, ,p3, p1,p2,, p1,,p3, ,p2,p3, p1,p2,p3

*Estado objetivo*: [M,p1,p2,p3].

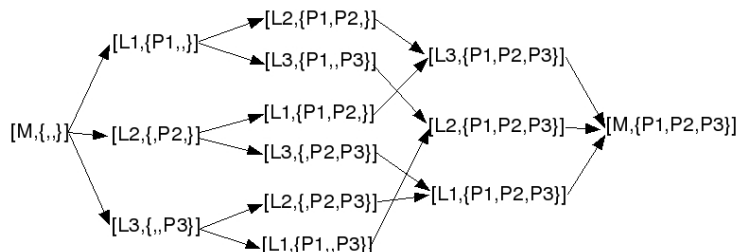
*Estado inicial*: [M,].

*Ações*: as definidas no problema (vá ao módulo de aterrissagem, vá à pedra1, vá à pedra2, vá à pedra3).

*Teste*: se o robô está no módulo com as 3 pedras.

*Avaliação do custo*: tempo (queremos que o robô execute a tarefa no menor tempo possível).

(b)



- (c) Como há pesos diferentes nas arestas do grafo, representando os diferentes tempos de locomoção entre os locais, a busca de custo uniforme é a mais adequada.
2. (a) *Espaço de estados*: todos os pares possíveis de cidades  $(i, j)$ , onde os amigos estão  
*Estado objetivo*:  $(i, i)$ , para algum  $i$   
*Estado inicial*: Algum  $(x, y)$   
*Ações*: a partir de  $(i, j)$ , ir para algum estado  $(x, y)$ , tal que  $Adjacente(x, i)$  e  $Adjacente(y, j)$   
*Teste*: Se estamos em um estado objetivo  
*Avaliação do custo*: O custo de ir de  $(i, j)$  a  $(x, y)$  é  $\max(d(i, x), d(j, y))$  (lembre que quem chega antes espera o outro)
- (b) Na melhor das hipóteses, os amigos andam em linha reta um em direção ao outro, sem um ter que esperar o outro (ou seja, em passos de igual tamanho), reduzindo sua distância em 2 vezes o tamanho de cada passo. Assim, somente (iii) é admissível.
- (c) Sim. Por exemplo, um mapa com apenas dois nós conectados por um link. Os dois amigos irão trocar de posições indefinidamente.
- (d) Sim. Considere o mapa do item anterior. Adicione um laço levando de um dos nós a ele mesmo. Nesse caso, toda solução exigirá que um dos amigos tome esse laço para cair novamente no mesmo nó, enquanto o outro amigo apenas muda de nó. Se o laço nunca for percorrido, não se chegará à solução.
3. (a) Dado o objetivo de coletar 10 tipos de rochas em 3 dias, uma possível escolha de variáveis seria (as demais variáveis, embora interessantes, não são tão relevantes):

- Tipos de rocha já coletados (para saber quantas rochas faltam)
  - Localização atual do robô (para saber quanto teria que andar até algum outro lugar)
  - Tempo decorrido desde a saída do módulo (para saber quanto ainda tem)
  - Nível de carga atual da bateria (para incluir paradas no tempo estimado para retorno)
- (b) Se o robô está de volta ao módulo, com os 10 tipos de rochas, e em menos de 3 dias.
- (c) (do que o robô pode fazer)
- Carregar a bateria: Incrementa o nível de carga da bateria em 1 unidade, incrementa o tempo desde a saída em 1h. Não há pré-requisitos.
  - Mover-se: decresce a carga da bateria com o valor especificado no mapa, incrementa o tempo desde a saída pela quantia especificada no mapa, muda a localização do robô. Pré-requisito: carga na bateria suficiente para atravessar um quadrado.
  - Pegar uma rocha: decresce a carga da bateria em uma carga, muda o conjunto de tipos de rochas já coletados. Pré-requisitos: carga na bateria suficiente para executar a ação.
- (d) Digamos que usemos a distância percorrida como custo, então:
- Carregar a bateria: 0
  - Mover-se: 10m (a resolução do mapa)
  - Pegar uma rocha:  $\alpha \times$  peso das rochas no local atual
- (e)
- H1: Não é admissível. Imagine que todas estão alinhadas; a heurística daria um valor muito maior que o real (uma linha reta da primeira à última rocha)
  - H2: Não é admissível. O circuito não leva em conta a localização do módulo, o que pode determinar que rocha é pega primeiro.
  - H3: Totalmente admissível (ainda que exagerada), não há como percorrer um caminho mais curto.
4. (a) A-B-D-E-F-C-G
- (b) A-C-H-I. A solução não foi encontrada por não termos mais para onde ir.
5. (a) A-B-D-E-O'
- (b) A-C-O
6. (a) Por ter peso nas arestas e exigir o menor caminho, Busca de Custo Uniforme
- (b) A\*. Valores consistentes (admissíveis) garantem o menor caminho.
- (c) Busca Iterativa em Profundidade. Ela dá o menor caminho, mantendo o menor número possível de nós na memória.
7. (a) E (na sequência A-B-D-E)
- (b) Local, pois há um nó com valor objetivo maior: K.