



INTELIGÊNCIA COMPUTACIONAL

Prof.: Aydano Pamponet Machado

1. Problema dos missionários e canibais

Três missionários e três canibais estão em um lado do rio, juntamente com um barco que pode conter uma ou duas pessoas. Descubra um meio de fazer todos atravessarem o rio, sem deixar que um grupo de missionários de um lado fique em número menor que o número de canibais nesse lado do rio. Esse problema é famoso em IA, porque foi assunto do primeiro artigo que abordou a formulação de problemas a partir de um ponto de vista analítico (Amarel, 1968).

Implemente e resolva o problema de forma ótima, utilizando um algoritmo de busca apropriado. É boa idéia verificar a existência de estados repetidos?

Amarel, S. (1968). On representations of problems of reasoning about actions, Machine Intelligence, (3), 131—171

2. Problema do metrô de Paris

Suponha que queremos construir um sistema para auxiliar um usuário do metrô de Paris a saber o **trajeto mais rápido** entre a estação onde ele se encontra e a estação de destino. O usuário tem um painel com o mapa, podendo selecionar a sua estação de destino. O sistema então acende as luzes sobre o mapa mostrando o melhor trajeto a seguir (em termos de quais estações ele vai atravessar., e quais as conexões mais rápidas a fazer – se for o caso).

Considere que:

- a distância em linha reta entre duas estações quaisquer é dada em uma tabela. Para facilitar a vida, considere apenas 4 linhas do metrô.
- a velocidade média de um trem é de 30km/h;
- tempo gasto para trocar de linha dentro de mesma estação (fazer baldeação) é de 4 minutos.

Formule e implemente este problema em termos de estado inicial, estado final, operadores e função de avaliação para **Busca heurística com A***.

Mapa do metrô de Paris.

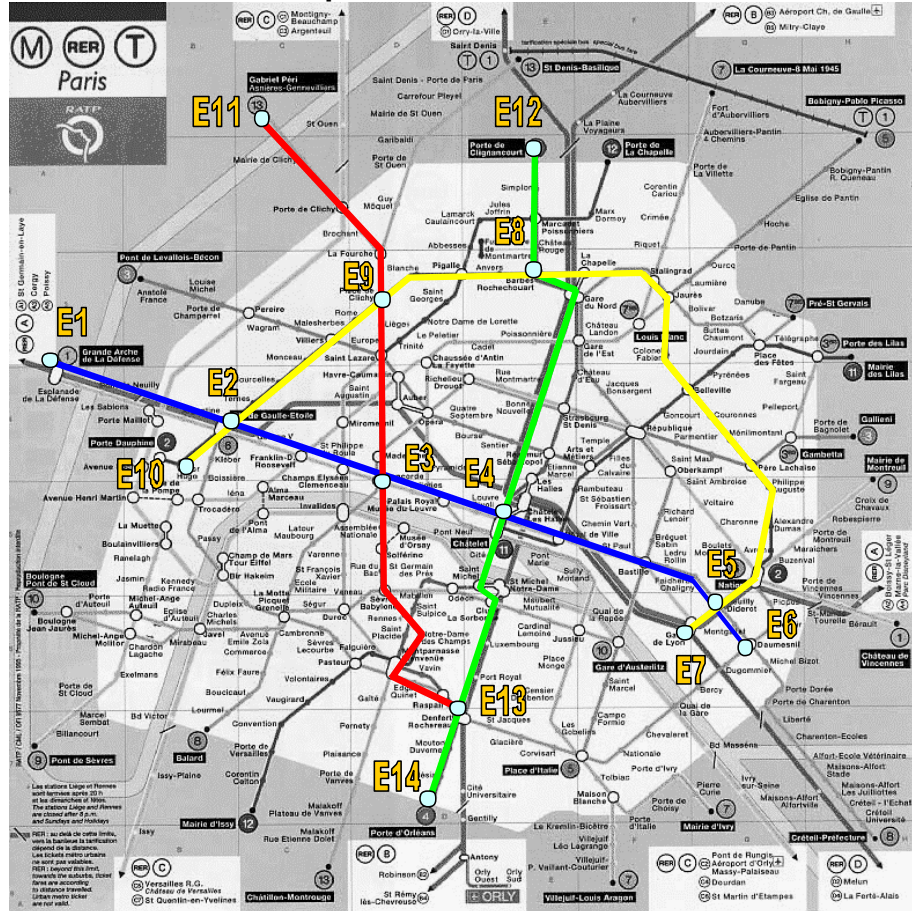


Tabela de distâncias do Metrô de Paris.

	E1	E2	E3	E4	E5	E6	E7	E8	E9	E10	E11	E12	E13	E14
E1	-	11	20	27	40	43	39	28	18	10	18	30	30	32
E2	11	-	9	16	29	32	28	19	11	4	17	23	21	24
E3	20	9	-	7	20	22	19	15	10	11	21	21	13	18
E4	27	16	7	-	13	16	12	13	13	18	26	21	11	17
E5	40	29	20	13	-	3	2	21	25	31	38	27	16	20
E6	43	32	22	16	3	-	4	23	28	33	41	30	17	20
E7	39	28	19	12	2	4	-	22	25	29	38	28	13	17
E8	28	19	15	13	21	23	22	-	9	22	18	7	25	30
E9	18	11	10	13	25	28	25	9	-	13	12	12	23	28
E10	10	4	11	18	31	33	29	22	13	-	20	27	20	23
E11	18	17	21	26	38	41	38	18	12	20	-	15	35	39
E12	30	23	21	21	27	30	28	7	12	27	15	-	31	37
E13	30	21	13	11	16	17	13	25	23	20	35	31	-	5
E14	32	24	18	17	20	20	17	30	28	23	39	37	5	-

3. O problema do caixeiro viajante

Um caixeiro viajante precisa visitar 10 cidades do interior de Pernambuco. Ele pede a um agente de busca que determine uma rota para sua visita tal que cada cidade só seja visitada *uma única vez*, e ele percorra o *menor espaço possível* (em Km). O agente de busca tem um mapa do estado, e portanto sabe as distâncias entre as cidades.

Formule e implemente este problema em termos de estado inicial, estado final, operadores e função de avaliação para **Busca por melhoras iterativas com Hill-Climbing**.

O operador considerado para gerar os filhos do estado corrente é permutar as cidades da rota atual duas a duas, e verificar em seguida se o caminho está conectado (segundo a tabela abaixo, que representa o mapa da questão). A cidade inicial deve ser mantida, uma vez que o caixeiro mora lá ☺ A rota é fechada (ele volta à cidade de origem no final).

	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10
C1	0	30	84	56	-	-	-	75	-	80
C2	30	0	65	-	-	-	70	-	-	40
C3	84	65	0	74	52	55	-	60	143	48
C4	56	-	74	0	135	-	-	20	-	-
C5	-	-	52	135	0	70	-	122	98	80
C6	70	-	55	-	70	0	63	-	82	35
C7	-	70	-	-	-	63	0	-	120	57
C8	75	-	135	20	122	-	-	0	-	-
C9	-	-	143	-	98	82	120	-	0	-
C10	80	40	48	-	80	35	57	-	-	0

4. Jogo para dois jogadores

Escolha um jogo para dois jogadores (ex.: jogo da velha, othelo, damas, xadrez, etc.) e implemente-o utilizando o minimax.