II. Agentes Reactivos

1. Máximo 10 arpões
Depósito com capacidade para 20 peixes

a) Nº de arpões e Nº de peixes do Nautilus

b)
- Têm um peixe em frente → Peixe_em_Frente
- Atingiu um peixe → Atingiu_Peixe
- Têm N peixes no depósito → Nº de peixes(N)
- Têm K arpões → Nº de arpões(K)

c)

Situação	Actualização	Acção
NºPeixes(N) ∧ NºArpões(K) ∧ $1 \leq N \leq 20$ ∧ $1 \leq K \leq 10$	—	Vaguear
NºPeixes(20) ∧ NºArpões(K) ∧ $1 \leq N \leq 20$ ∧ $1 \leq K \leq 10$	$N=0$	Descarregar
NºPeixes(N) ∧ NºArpões(0) ∧ $1 \leq N \leq 20$ ∧ $1 \leq K \leq 10$	$K=10$	Reabastecer
NºPeixes(20) ∧ NºArpões(0)	$N=0$ $K=10$	Descarregar/Reabastecer
Peixe_em_Frente	$K=K-1$	Disparar
Atingiu_Peixe	$N=N+1$ $K=K+1$	Agarrar

2.

a) Como não existem informações sobre como se deve actualizar a distância percorrida, então não se trata de uma variável de estado.

ARRUMAR: indica se a formiga está à procura ou a arrumar provisões.

b) PROCURAR_OU_ARRUMAR (ARRUMAR)
distância_percorrida (x)
ve_formiga
encontra_local
encontra_provisão

c)

Situação	Actualização	Acção
PROCURA_OU_ARRUMAR(0)	———	PROCURAR_PROVISÃO
PROCURAR_OU_ARRUMAR(0) ∧ encontra_provisão	ARRUMAR=1	Agarrar provisão
PROCURAR_OU_ARRUMAR(1) ∧ distância_percorrida(x) ∧ $x \leq 5$	———	PROCURAR_local
PROCURAR_OU_ARRUMAR(0) ∧ distância_percorrida(x) ∧ $x > 5$ ∧ ve_formiga	———	seguir_formiga
PROCURAR_OU_ARRUMAR(1) ∧ encontra_local	ARRUMAR=0	libertar_provisão

III Representação do Conhecimento

1.

a) Todos em Oxford são espertos

$\forall x \; estudam(x, Oxford) \Longrightarrow Esperto(x)$

b) Alguém em Oxford é esperto

$\exists x \; estuda(x, Oxford) \wedge Esperto(x)$

c) Existe uma pessoa que gosta de toda a gente

G(x,y): x gosta de y

$\exists x \; P(x) \wedge (\forall y \; G(x,y))$

d) só um aluno chumbou a história

$\exists x \; (chumbou(x, história) \wedge \forall y ((y \neq x) \Rightarrow \neg chumbou(y, história)))$

e) Nem todos os estudantes se inscreveram simultaneamente a IIA e a SO. I(x): x está inscrito

$\neg(\forall x \; I(x, IIA) \wedge I(x, SO))$

f) só um aluno chumbou a história e a Biologia

C(x): x chumbou

$\exists x \; C(x, história) \wedge C(x, Biologia) \wedge (\forall y (y \neq x) \neg$
$(C(y, história) \wedge C(y, Biologia)))$

g) A melhor nota a H foi mais elevada do que a melhor a B. M(x): x é a melhor nota

$\exists x \; M(x, H) \wedge (\exists y \; (x \neq y) \wedge M(y, B)) \wedge (x > y)$

h) Todos os portistas gostam do PC

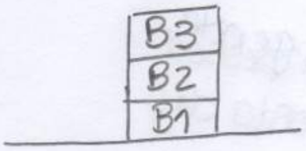
$\forall x \; Portistas(x) \Rightarrow G(x, PC)$

2. Mundo dos n blocos ($B_1, B_2, \ldots, B_n$)

Predicados: . $On(x,y)$
. $Clear(x)$

a) $\neg Clear(B_1) \wedge \neg Clear(B_2) \wedge On(B_3, x) \wedge x \neq Floor$

Se $x = B_2$ então:

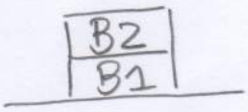


∴ Só precisam dos 3 blocos indicados, logo nº mínimo de blocos é 3.

b) $Clear(B_1) \Rightarrow Clear(B_2)$

$\equiv \neg Clear(B_1) \vee Clear(B_2)$

Nº mínimo de blocos é 2.



3. Tanques (T_1 e T_2)
Recipiente (R)

a)

i. $Open(T_2) \wedge Over(R, T_2)$

	water(T1)	water(T2)	water(R)
i	F/V	F	V
ii	F	V	V/F
iii	F	F/V	V/F
iv	F/V	F/V	V/F

b)

i. $\forall x (\neg (\neg open(x) \Rightarrow water(x)))$

$\equiv \forall x (\neg (open(x) \vee water(x)))$

$\equiv \forall x (\neg open(x) \wedge \neg water(x))$

$\equiv \forall x (open(x) \Rightarrow \neg water(x))$

Falso

ii. $\forall x (\neg ((open(x) \Rightarrow water(x)))$

$\forall x (open(x) \vee \neg water(x))$

$\forall x (water(x) \Rightarrow open(x))$

Falso

iii $\forall x \; open(x) \Rightarrow \exists y \; water(y)$

Verdade

iv. $\exists x, y \; open(x) \wedge open(y) \wedge x \neq y$

Falso

4.

a)

Predicados:

- comprimento(x,y): comprimento da Rua x é y unidades
- fechada(x,y): a Rua x está fechada no ponto y
- intersecta(x,y,z) a Rua x e a Rua y intersectam-se no ponto z.
- edificio(x,y,z,w): existe um edifício x que se encontra a y unidades da intersecção da Rua z e da Rua w.

b)

comprimento(descanço, 6) ∧ comprimento(Rua norte, 8) ∧
comprimento(Rua sul, 8) ∧ comprimento(Rua principal, 6) ∧
fechada(Rua Norte, 5) ∧ fechada(Rua Norte, 6) ∧ fechada(Rua Norte, 7) ∧
intersecta(Rua descanso, Rua sul, 0) ∧ intersecta(Rua artes, Rua sul, 4) ∧
intersecta(Rua artes, Rua Norte, 4) ∧ intersecta(Rua Norte, Rua principal, 8) ∧
intersecta(Rua descanso, Rua Norte, 6) ∧ intersecta(Rua sul, Rua principal, 8) ∧
edificio(casa, R. descanso, R. sul, 2)

5. A KIF é uma linguagem desenhada para representar o conhecimento trocado entre agentes, podendo também ser usada para representar os modelos internos de cada agente. Pode formalizar-se conhecimento sobre o conhecimento, ou seja, tratar uma expressão como um objecto

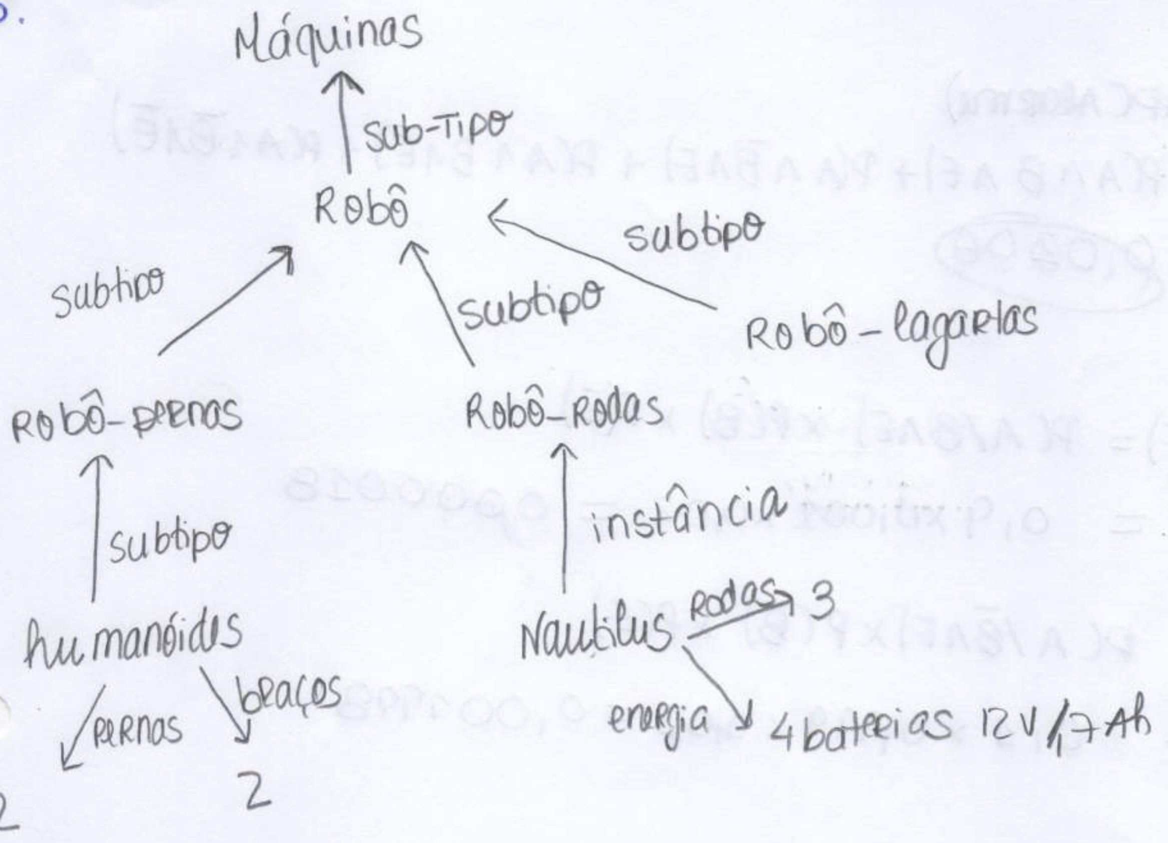
- KIF têm uma semântica puramente declarativa, o prolog também declarativa, mas a semântica depende em parte do modelo

- Pode ser ser tanto ou mais expressiva quanto a lógica d 1º ordem

- Permite a representação de meta-conhecimento

(Pág. 79)

6.



7. a) objectos: $e_1, e_2, e_3, x_1, a_1, o_1, s_1$

Funções e relações:

OR(x,y)

XOR(x,y)

Not(x)

AND(x,y)

~~b)~~ c)

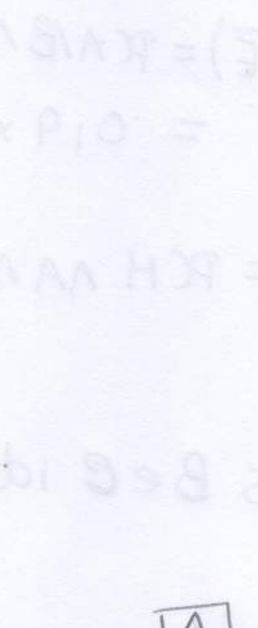
OR(XOR(e_1,e_2), AND(e_2,e_3))

8. $P(a \wedge b \wedge \neg c \wedge \neg d)$

$= P(\neg d/b) \times P(\neg c/b) \times P(b/a) \times P(a)$

$= 1 - P(d/b) \times 1 - P(c/b) \times P(b/a) \times P(a)$

$= 1 - 0{,}9 \times 1 - 0{,}2 \times 0{,}3 \times 0{,}2$



9.

a) $P(A) = P(\text{Alarme})$

$= P(A \wedge B \wedge E) + P(A \wedge \overline{B} \wedge E) + P(A \wedge B \wedge \overline{E}) + P(A \wedge \overline{B} \wedge \overline{E})$

$= 0{,}02 06$

• $P(A \wedge B \wedge E) = P(A / B \wedge E) \times P(B) \times P(E)$

$= 0{,}9 \times 0{,}001 \times 0{,}02 = 0{,}000018$

• $P(A \wedge \overline{B} \wedge E) = P(A / \overline{B} \wedge E) \times P(\overline{B}) \times P(E)$

$= 0{,}1 \times 0{,}999 \times 0{,}02 = 0{,}001998$

• $P(A \wedge \overline{B} \wedge \overline{E}) = P(A / \overline{B} \wedge \overline{E}) \times P(\overline{B}) \times P(\overline{E})$

$= 0{,}001 \times 0{,}999 \times 0{,}98 = 0{,}0009 7902$

• $P(A \wedge B \wedge \overline{E}) = P(A / B \wedge \overline{E})\ P(B) \times P(\overline{E})$

$= 0{,}9 \times 0{,}001 \times 0{,}98 = 0{,}00882$

b) $P(M) = P(M \wedge A \wedge B \wedge E)$

⋮

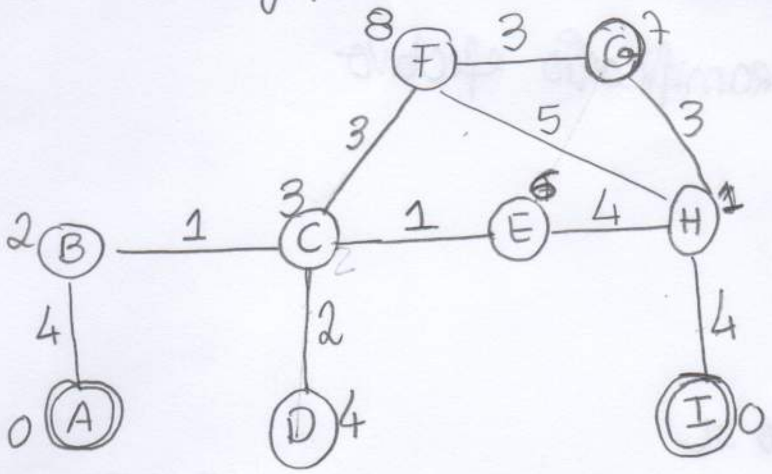
Ambas B e e idênticas à A

IV Técnicas de Resolução automática de problemas

1.

a) Uma heurística diz-se admissível se a pesquisa em A* encontra sempre uma solução óptica e nunca sobrestima o custo real de chegar a uma solução a partir de n

Neste caso a heurística não é admissível porque o custo real para chegar à solução a partir de F é 8 (3+1+4=8), e a sua heurística é 10, logo, sobrestima o custo real.

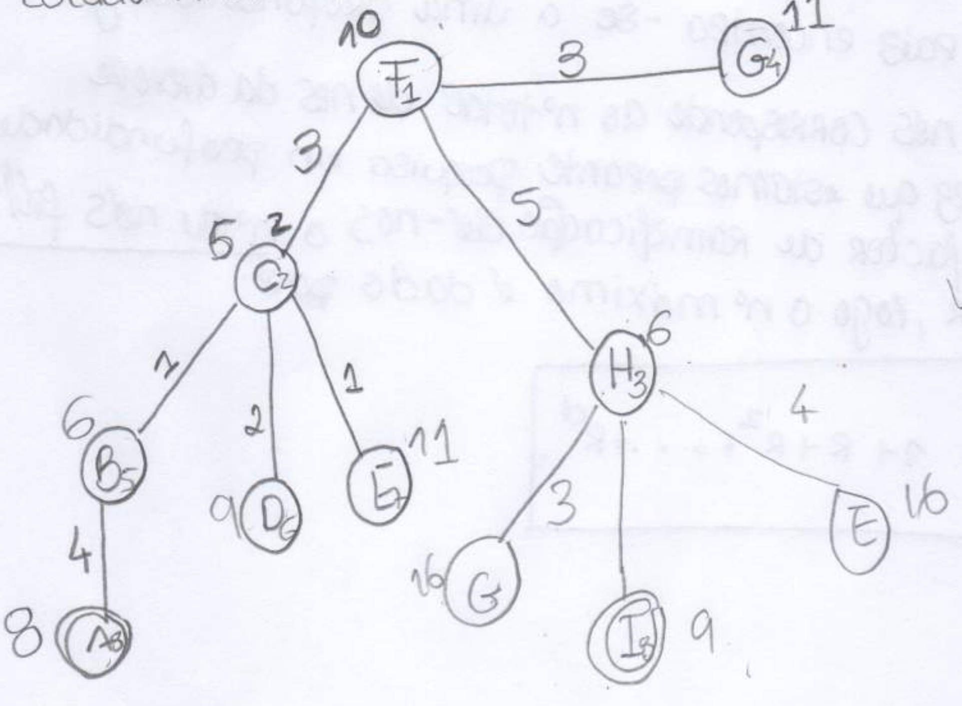


Heurísticas alteradas:

F, G e E

b) Estratégia: A* sem repetição
Estado inicial: F

$f(n) = g(n) + h(n)$



c) Ramificação média

$RM = \frac{N-1}{x}$

x → nº nós não terminais

$x = 4$
$N = 11$

d) $N = \frac{B^{d+1} - 1}{B - 1}$ Ramificação efectiva

d → comprimento da árvore até à solução

B → factor de ramificação efectivo

$d = 4$
$N = 11$

2. factor de ramificação r

solução a uma profundidade g

O número mínimo de nós visitados corresponde ao melhor caso, é $\boxed{g+1}$ nós, pois a solução mais próxima da raiz encontra-se a uma profundidade g.

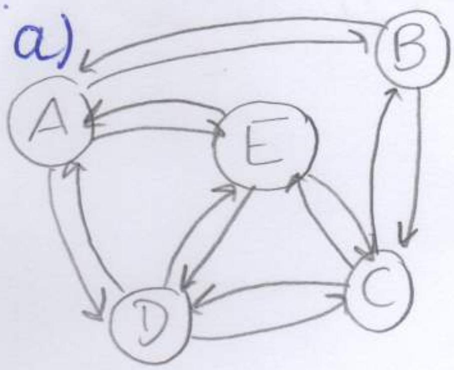
O nº máximo de nós corresponde ao nº total de nós da árvore no pior caso. Uma vez que estamos perante pesquisa em profundidade com limite d e o factor de ramificação dá-nos o nº de nós filhos por nó, dado por R, logo o nº máximo é dado por:

$$\boxed{\sum_{i=0}^{d} R^i = 1 + R + R^2 + \dots + R^d}$$

3. Uma heurística diz-se admissível se nunca sobrestima o custo real de chegar à solução, portanto uma heurística admissível seria calcular a distância em linha recta entre dois locais, através das coordenadas correspondentes ao nó actual e à solução, através das coordenadas cartesianadas.

4.

a)



E = COR 1
A = C = COR 2
B = D = COR 3

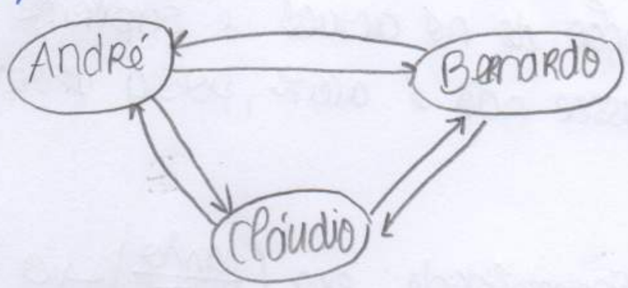
Mínimo de 3 cores

b) Mínimo de 4 cores

c) Mínimo de 4 cores

5. chapéu do Cláudio + bicicleta Bernardo

a)



variáveis: A, B, C

valores possíveis: André ∈ {(B, C), (C, B)}
Bernardo ∈ {(A, C), (C, A)}
Cláudio ∈ {(B, A), (A, B)}

Os tuplos a cima representam: (bicicleta, chapéu)

Restrições: 1º do André ≠ 1º do Cláudio ≠ 1º do Bernardo
2º do André ≠ 2º do Cláudio ≠ 2º do Bernardo

Portanto se o André ∈ (Cláudio, Bernardo) então
o Bernardo ∈ (André, Cláudio) e
o Cláudio ∈ (Bernardo, André)

6. Pesquisa por propagação de restrições

Variáveis: $A_1, \ldots A_9, \ldots B_1, \ldots B_9, \ldots, I_1, \ldots I_9$

Domínio: $A_i \in \{1, \ldots 9\}$
$B_i \in \{1, \ldots 9\}$
$C_i \in \{1, \ldots, 9\}$
...
$I_i \in \{1, \ldots, 9\}$

Restrições:
colunas:
linhas:
Matriz:
3x3

7. Pesquisa por recozimento simulado

com T=0, quando o valor da função do nó actual é superior à função do sucessor, então o sucessor não é aceite, pois a probabilidade para T=0 é 0:

$t \to +\infty$ Probabilidade: $\exp\left(\frac{ganho}{T}\right) \to 0$
$T \to 0$
$\frac{ganho}{T} \to -\infty$

Portanto, com T=0 a pesquisa por recozimento simulado não aceita alterações com ganho negativo, tendo então neste caso algumas semelhanças com a pesquisa por montanhismo.

Semelhanças: - Quando o valor da função no nó actual é superior ao valor da função no sucessor, a pesquisa pára.
- Não há retorno

Diferenças: Na pesquisa por recozimento simulado o nó sucessor é escolhido aleatoriamente, enquanto na pesquisa por montanhismo o nó escolhido é o que tiver maior valor da função de avaliação

8. Pesquisa com propagação de restrições

Variáveis: A, B, C, D

Domínio:

$A \in \{8, 9, 10, \ldots, 18\}$
$B \in \{8, 9, 10, \ldots, 17\}$
$C \in \{8, 9, 10, \ldots, 16\}$
$D \in \{8, 9, 10, \ldots, 15\}$

? Restrições:

- $A+1 \leq B$
- $A+1 \leq C$
- $D \geq B+2$
- $D \geq C+2$
- $(B+2 \leq C) \vee (C+3 \leq 2)$

Solução possível:

$A=8$; $B=9$; $C=11$; $D=14$

9.

a)

em_cima(x, y) : x está em cima da caixa;
na_posição(x, y) : x está na posição y;
no_chão (x) : x está no chão;
na_posse (x, y) : x tem na sua posse y;
Pendurado(x)

b) {na_posição(macaço, A),
na_posição(caixa, C),
na_posição (bananos, B),
no_chão (macaco), No_chão (caixa),
Pendurado(bananas) }

d) Acções :

deslocar (x, P1, P2)
empurar(x, y, P1, P2)
subir(x, y)
agarrar(x, y)

e)