

Epoca Normal 0607

1.1 $\begin{array}{c} \boxed{x} \\ \textcircled{A} \end{array} \xrightarrow{c} \begin{array}{c} \boxed{y} \\ \textcircled{B} \end{array}$

$$x \leq c + y$$

$$25 \leq 20 + x$$

x entre 5 e 12

$$15 \leq 15 + x$$

$$x \leq 10 + 2$$

$$x \leq 10 + 27$$

é consistente se a função de avaliação $\tilde{f}$ for monótona

1.2 árvore-lista aberta

grato - lista aberta e fechada

1.3 como maximizar opta pelo nó seleccionado que tenha maior $f(n)$

$$A \rightarrow B$$

$A \rightarrow C \rightarrow F$ como o sucessor é ele próprio e $f(n)$ não aumenta então sai.

1.4 tem de satisfazer a restrição

per. 1

$$A + 2B > 6$$

$$\text{ex) } 1 + 2B > 6$$

ex) $B > 2.5$ inconsistente

1.5 - Depende da ordem (o corte) valor minimax da raiz (5)

O 7 já não é explorado

ramo direito 8 7 já não é explorado porque o segundo nó da direita não tem mais qe 2

Cortar quando:

- limite inferior passa o superior
- quando intervalo fica vazio

~~7a~~ 7a $\wedge$ d

1.6 7a

~~3 7b~~
~~bvd~~ 1

3 ~~a~~ v b v c
7c v 7e

b v e
~~a v d~~ 1

se um conjunto de cláusulas
é satisfazível

algoritmo completo

ao contrário do Horn Sat

tentar a negação de 7a $\wedge$ d

e provar que é insatisfazível

↳ teste de satisfatibilidade

① ver símbolos puros
d = true

KB $\models \alpha$

KB $\cup \{7a\} \models \perp$

outra maneira

$\models (KB \Rightarrow \alpha)$

↳ (por tabelas de
verdade)

eliminar

bvd e avd

② ver símbolos puros outra v
nã existem

③ ver cláusulas unitárias
a = false

↳

7b

b v c

7c $\wedge$ 7e

b v e

b = false $\rightarrow$

c

c = true

7e

e = true

7c v 7e

e

Não é consistente

é consequencialógica 7a $\wedge$ d

a f

b f

é insatisfazível

c t

d t

e f

1.7 $d :- c, \text{not } d$ c não pode ser verdade

$\hookrightarrow d :- \text{not } c$ $D \neq \text{true}$

$d :- \text{not } c$ funciona como restrição de integridade

Conjuntos $\{\emptyset\}, \{a\}, \{b\}, \{a, b\} \rightarrow$ modelos estáveis possíveis

① dividir o programa pelo modelo $\{a, b\}$

$\frac{P}{M} =$ $a :- b$ ~~tirar not c~~ p é true

~~$c :- \text{not } a$~~ antecedente falso
 $d :- c$ regra vai fora

$\text{least} \left(\frac{P}{M} \right) = \{b, a\}$ modelo estável
conclui-se $\emptyset$ é verdade
 b é verdade logo $a + b$ é

$\vdash_P(M)$

$\{c, d\}$ $\frac{P}{M} = c$

$\uparrow$
 $\{b\}$ é modelo mínimo logo
como não é igual a $\{c, d\}$
o modelo não é estável.

O valor não é estável porque P é igual a todas as regras e isso dá o modelo mínimo $\{a, b, c, d\} \neq \{\emptyset\}$

1.8 todos os pontos com água não são interessantes

$:- \text{ponto}(x), \text{agua}(x), \text{interessante}(x)$

$\forall x [p(x) \wedge a(x) \Rightarrow \neg i(x)]$

$\neg \exists x [p(x) \wedge a(x) \wedge i(x)]$

~~negado~~
 $a \Rightarrow b \Rightarrow \neg a \vee b$ negado $a \wedge \neg b$

1.9 ameaças

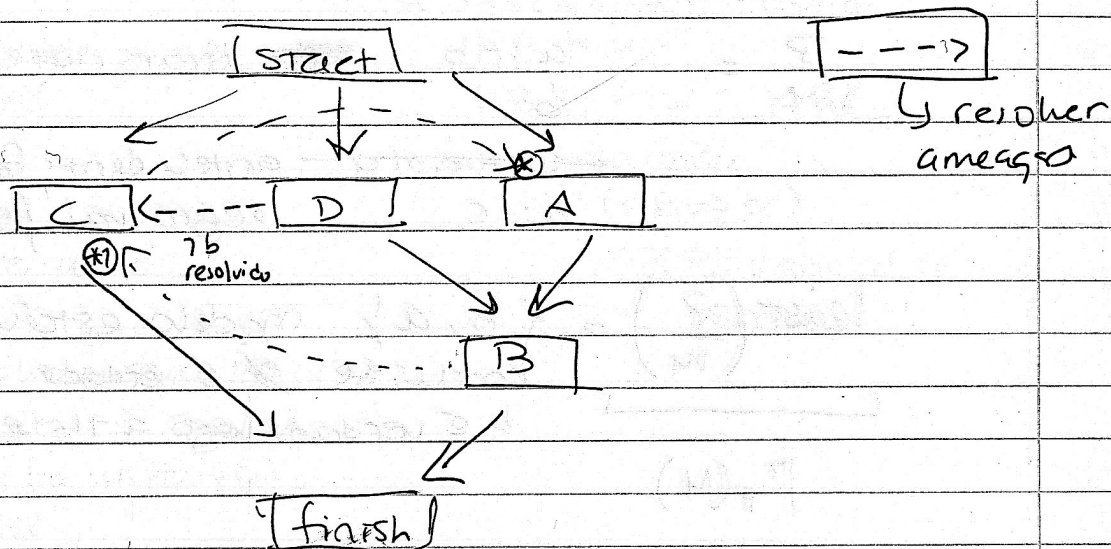
há um passo no plano com efeito negativo

ligação causal (seta)

γ_C e γ_B
ameaças

ligação $C \rightarrow C +$

ligação $B \rightarrow B$



ver se plano é inconsistente depois de *

ver se tem ciclos

D, C, A, B é consistente e linear

so há esta maneira de correr

*1 D, A, B, C
ou A D B C

now há pre condições negativas em plano STRIPS

$$1.10 \quad x_1 w_1 + \dots + x_n w_n = 0,5$$

$$\text{erro } 0,8 - 0,378 = 0,422$$

$$\eta = 0,1$$

$$\Delta w_1 = 0,1 \times 0,422 \times (-2) \times \frac{x_1}{x_1(1 - 0,378)}$$

$$w_1 = 0,5 - 0,02 = 0,48$$

$$= -0,02$$

Redes de busca

na prob condicionada e preciso normalizar

III

- ① mover o T para O_f
mover R $s \rightarrow a \rightarrow b \rightarrow c \rightarrow \dots \rightarrow T$

- ② qual a informaçao de cada estado
posiçao do robot e posiçao dos objectos

estado inicial $\langle S, O \rangle$

teste de estado objectivo posiçao robot = T

funçao sucessor sucessores imediatos do estado actual

operadores - mover robot (MR)

mover objectos (MOB)

custo dos operadores = 1

$(R_1, O_1) \xrightarrow{MR} (R_2, O_1)$

adjacente $(R_1, R_2) \wedge R_2 \in V - O_1$

$(R_2 \text{ esta livre})$

$mob(O, O')$

$(R, O_1) \xrightarrow{mob(O, O')} (R, O_2)$

$O' = \text{casa livre sem robot}$

obstaculo casa

$O \in O_1 \wedge O' \in V - (O_1 \cup \{R\}) \wedge$

$\wedge O_2 = (O_1 - \{O\}) \cup \{O'\} \wedge \text{adjacente}(O, O')$

- ③ a) e admissivel
b) não e admissivel porque a heuristica olha para a frente. estimativo do que falta e não do que já está feito
c) se o menor caminho e admissivel este h_b e $h(c)$ e dominada por $h(a)$.
d) e admissivel porque se houver outro caminho esse tem pelo menos mais uma unidade

e) não é admissível

④ interessa o caminho pa solver logo é falso.

Recurso 0607

II) Quantos escalonamentos diferentes existem.
cada unidade do domínio elevado ao número de tarefas
 m^n

estado: array [n tarefas] int $\{0 \dots m\}$
↳ tarefa n atribuída

estado inicial ~~array~~ array = 0

estado final array sem zeros

operadores escalonar tarefa

① ordenar ~~tarefas~~ tarefas

função sucessor procurar 1ª posição a zero
atribuir um valor de 1 a m
(cd tarefa tem m hipóteses)

custo do operador tempo de concluir (antes) - tempo de concluir (depois) ↗ ↘

- II 3 a) não é admissível +6 custo 6
tempo de conclusão foi +7
- b) não é admissível
- c) é admissível
- d) é admissível (processadores todos ocupados
todos com o mesmo tempo, solução ótima)
- e) admissível

———— u ————— u ————— u —————

R 0607

I1

$$x \leq 0+1$$

$$0 \leq 3+x$$

$$y \leq 7+0$$

$$z \leq 5+y$$

$$z \leq 3+0$$

$$x \in \{0, 1\}$$

$$y \in \{0, 7\}$$

$$z \in \{0, 3\}$$

I5

