

PCS 2059
PCS 2428
Inteligência Artificial

Profa:
Anna H. R. Costa
© 2007

Módulo:
Planejamento

Autores:
Anna H. R. Costa
Jaime S. Sichman

Versão: 2.3
Data: 26/10/07

PCS 2059 –Inteligência Artificial

PLANEJAMENTO

1



PCS 2059
PCS 2428
Inteligência Artificial

Profa:
Anna H. R. Costa
© 2007

Módulo:
Planejamento


Autores:
Anna H. R. Costa
Jaime S. Sichman

Versão: 2.3
Data: 26/10/07

STRIPS: Estados e Objetivos

- STRIPS: Stanford Research Institute Problem Solver (Fikes e Nilsson, 1971)
- Estados: conjunção de literais sem variáveis**
 - Inicial: **At(Home)**
 - Por default, literal não representado é falso (hipótese do mundo fechado); assim, não precisa escrever:
 $\neg \text{Have}(\text{Milk}, S0) \wedge \neg \text{Have}(\text{Bananas}, S0) \wedge \neg \text{Have}(\text{Drill}, S0)$
- Objetivos: conjunção de literais e/ou variáveis (\exists)**
At(Home) \wedge Have(Milk) \wedge Have(Bananas) \wedge Have(Drill)
 - Mas também pode incluir variáveis (ex: o objetivo é estar em um lugar que venda Milk):
 - At(x) \wedge Sells(x, Milk)**

2



PCS 2059
PCS 2428
Inteligência Artificial

Profa:
Anna H. R. Costa
© 2007

Módulo:
Planejamento

Autores:
Anna H. R. Costa
Jaime S. Sichman


Versão: 2.3
Data: 26/10/07

STRIPS: Ações

- Ações:**
 - Descritor da **ação**: predicado lógico
 - Pré-condição**: conjunção de literais positivos
 - Efeito**: conjunção de literais, podendo ser:
 - positivos (adicionados a uma lista)
 - negativos (retirados de uma lista)
- Exemplo: Operador para ir de um lugar a outro**
 - Op(ACTION: Go(there),
 - PRECOND: At(there) \wedge Path(there, there),
 - EFFECT: ADD: At(there), DEL: \neg At(there))
- Notação alternativa**

$$\begin{array}{c} \text{At(there), Path(there, there)} \\ \boxed{\text{Go(there)}} \\ \text{At(there), } \neg \text{At(there)} \end{array}$$

3



PCS 2059
PCS 2428
Inteligência Artificial

Profa:
Anna H. R. Costa
© 2007

Módulo:
Planejamento

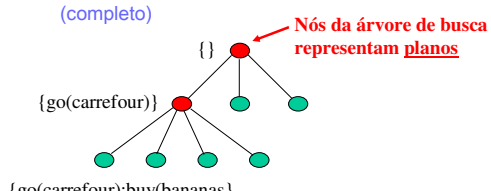
Autores:
Anna H. R. Costa
Jaime S. Sichman

Versão: 2.3
Data: 26/10/07


Busca no Espaço de Planos

- Idéia**
 - Buscar um plano desejado em vez de uma situação desejada (espécie de meta-busca)
 - parte-se de um plano inicial (parcial), e aplica-se os operadores até chegar a um plano final (completo)

Nós da árvore de busca representam planos



4



PCS 2059
PCS 2428
Inteligência Artificial

Profa:
Anna H. R. Costa
© 2007

Módulo:
Planejamento

Autores:
Anna H. R. Costa
Jaime S. Sichman


Versão: 2.3
Data: 26/10/07

Busca no Espaço de Planos: Nós

Cada nó representa um plano parcial e contém:

- Ações** = {A1, A2, A3, ..., An}
- Restrições de Ordem** = { A1 < A2, ..., A3 < An },
- Ligações Causais** = {Ai \xrightarrow{c} Aj, ...}
 - efeito de Ai adiciona c, que é uma pré-condição de Aj
- Pré-Condições Abertas**: conjunto de pré-condições que ainda não têm ligações causais
- Instanciação de Variáveis** = { x = cte1, y = z }

5



PCS 2059
PCS 2428
Inteligência Artificial

Profa:
Anna H. R. Costa
© 2007

Módulo:
Planejamento

Autores:
Anna H. R. Costa
Jaime S. Sichman

Versão: 2.3
Data: 26/10/07

Busca no Espaço de Planos: Operadores

Os operadores possíveis no espaço de planos são:

- Adicionar uma **ação** para eliminar uma pré-condição aberta (ACTIONS)
- Adicionar um **link causal** de uma ação já existente para uma pré-condição aberta (LINKS)
- Adicionar uma **restrição de ordem** de uma ação em relação a outra (ORDERINGS)
- Atribuir** a uma variável um valor ou outra variável (BINDINGS)

6

USP

PCS 2059

PCS 2428

Inteligência Artificial

Profa:

Anna H. R. Costa

© 2007

Módulo:

Planejamento

Autores:

Anna H. R. Costa

Jaime S. Sichman

Versão:

2.3

Data: 26/10/07

7

Exemplo de Busca no Espaço de Planos

Objetivo: $\text{RightShoeOn} \wedge \text{LeftShoeOn}$

Operadores

•Op(ACTION:RightShoe

PRECOND: RightSockOn ,

EFFECT: ADD: RightShoeOn)

•Op(ACTION: RightSock

EFFECT: ADD: RightSockOn)

•Op(ACTION:LeftShoe

PRECOND: LeftSockOn ,

EFFECT: ADD: LeftShoeOn)

•Op(ACTION: LeftSock

EFFECT: ADD: LeftSockOn)

USP

PCS 2059

PCS 2428

Inteligência Artificial

Profa:

Anna H. R. Costa

© 2007

Módulo:

Planejamento

Autores:

Anna H. R. Costa

Jaime S. Sichman

Versão:

2.3

Data: 26/10/07

8

Exemplo de Busca no Espaço de Planos

Plano inicial

Plan(ACTIONS = {S1: Op(ACTION: Start),

S2: Op(ACTION: Finish,

PRECOND: RightShoeOn \wedge

LeftShoeOn)),

ORDERINGS: { S1 < S2 },

BINDINGS: {},

LINKS: {})

USP

PCS 2059

PCS 2428

Inteligência Artificial

Profa:

Anna H. R. Costa

© 2007

Módulo:

Planejamento

Autores:

Anna H. R. Costa

Jaime S. Sichman

Versão:

2.3

Data: 26/10/07

9

Plano de Ordem Parcial

Start

LeftSock

RightSock

LeftSockOn

RightSockOn

LeftShoe

RightShoe

LeftShoeOn, RightShoeOn

Finish

geração de plano

USP

PCS 2059

PCS 2428

Inteligência Artificial

Profa:

Anna H. R. Costa

© 2007

Módulo:

Planejamento

Autores:

Anna H. R. Costa

Jaime S. Sichman

Versão:

2.3

Data: 26/10/07

10

POP (Partial Order Planning)

Existindo a linguagem (STRIPS), falta o algoritmo...

Características do POP

– algoritmo não determinista

– a inserção de uma ação só é considerada se atender uma pré-condição aberta

– planejador regressivo (do objetivo para o início)

– é correto e completo, assumindo busca em largura ou em profundidade iterativa

Idéia do algoritmo

– identifica ação com pré-condição aberta

– introduz ação cujo efeito é satisfazer esta pré-condição

– instancia variáveis e atualiza os links causais

– verifica se há ameaças e corrige o plano se for o caso

USP

PCS 2059

PCS 2428

Inteligência Artificial

Profa:

Anna H. R. Costa

© 2007

Módulo:

Planejamento

Autores:

Anna H. R. Costa

Jaime S. Sichman

Versão:

2.3

Data: 26/10/07

11

POP: Exemplo das Compras

Start

Sells(HWS, Drill)

At(Home)

Sells(SM, Bananas)

Sells(SM, Milk)

Conhecimento a priori do mundo

•Ações

Op(ACTION: Go(there),

PRECOND: At(there),

EFFECT: ADD: At(there),

DEL: \neg At(there))

Op(ACTION: Buy(x),

PRECOND: At(store) \wedge

Sells(store, x),

EFFECT: ADD: Have(x))

restrição de ordem

Have(Drill)

Have(Milk)

Have(Bananas)

At(Home)

Caracterização do objetivo

Finish

USP

PCS 2059

PCS 2428

Inteligência Artificial

Profa:

Anna H. R. Costa

© 2007

Módulo:

Planejamento

Autores:

Anna H. R. Costa

Jaime S. Sichman

Versão:

2.3

Data: 26/10/07

12

POP: Exemplo das Compras

Start

Sells(HWS, Drill)

At(Home)

Sells(SM, Milk)

Sells(SM, Bananas)

Have(Drill)

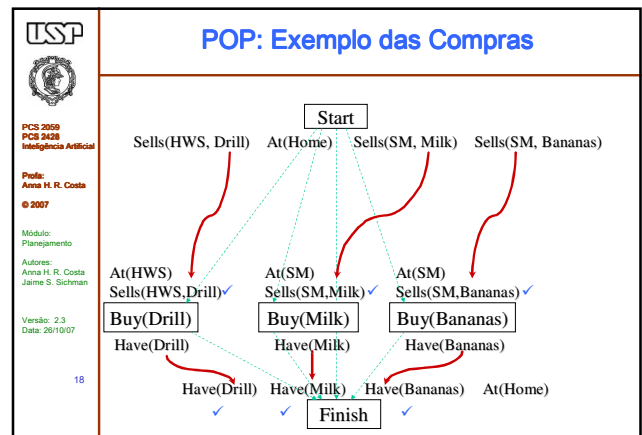
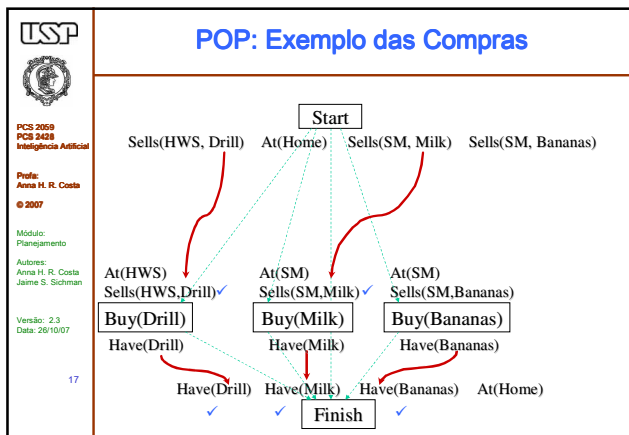
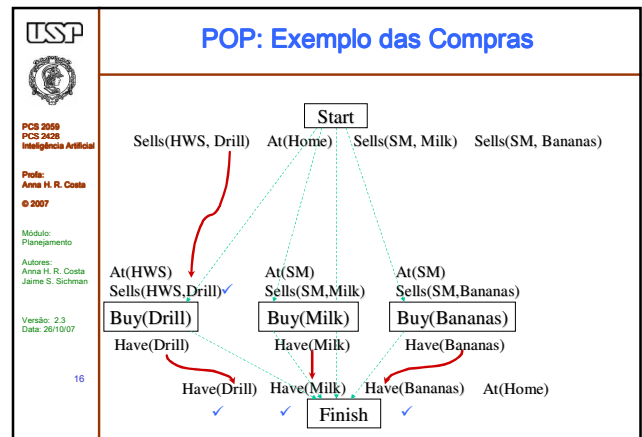
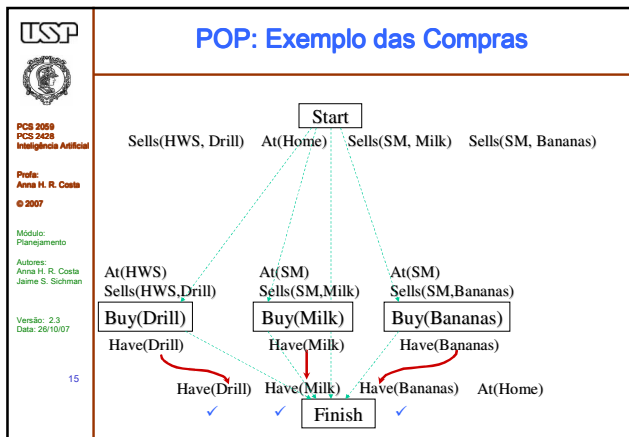
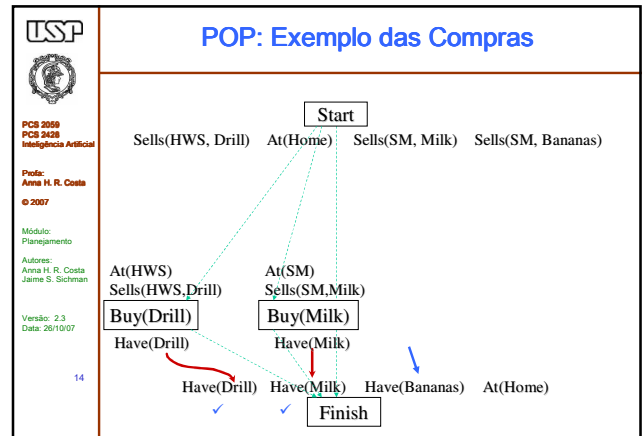
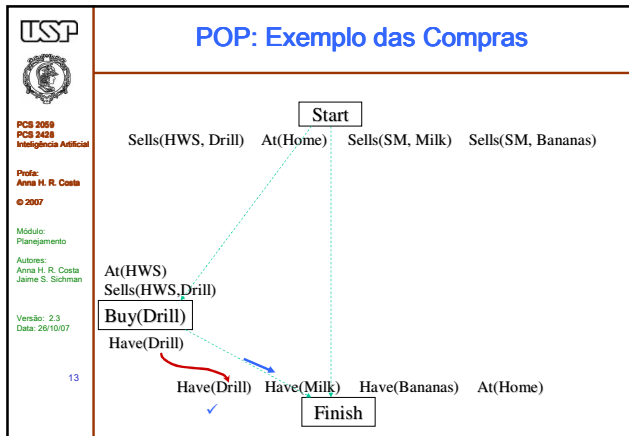
Have(Milk)

Have(Bananas)

At(Home)

Finish

2



```

graph TD
    Start[Start]
    Atx[At(x)]
    GoHWS[Go(HWS)]
    AtHWS[At(HWS)]
    AtSM1[At(SM)]
    AtSM2[At(SM)]
    BuyDrill[Buy(Drill)]
    BuyMilk[Buy(Milk)]
    BuyBananas[Buy(Bananas)]
    HaveDrill[Have(Drill)]
    HaveMilk[Have(Milk)]
    HaveBananas[Have(Bananas)]
    AtHome[At(Home)]
    Finish[Finish]

    Start -.-> Atx
    Start -.-> AtSM1
    Start -.-> AtSM2
    Atx -.-> GoHWS
    GoHWS -.-> AtHWS
    AtHWS -.-> BuyDrill
    AtSM1 -.-> BuyMilk
    AtSM2 -.-> BuyBananas
    BuyDrill -.-> HaveDrill
    BuyMilk -.-> HaveMilk
    BuyBananas -.-> HaveBananas
    HaveDrill -.-> HaveMilk
    HaveMilk -.-> HaveBananas
    HaveBananas -.-> AtHome
    AtHome -.-> Finish
  
```

The diagram illustrates the POP (Process of Purchase) for a shopping trip. The process starts at 'Start' and ends at 'Finish'. The path taken is highlighted in red, showing the sequence of actions: Start -> At(x) -> Go(HWS) -> At(HWS) -> Buy(Drill) -> Have(Drill) -> Have(Milk) -> Buy(Milk) -> Have(Bananas) -> Buy(Bananas) -> At(Home) -> Finish. Other possible actions are shown in green dashed lines.

The diagram illustrates a search space for a shopping problem. The search space is a directed graph with nodes representing states and edges representing actions. The initial state is 'Start'. The goal states are 'Go(HWS)' and 'Go(SM)'. The search space is divided into three main branches based on the first action: 'Buy(Drill)', 'Buy(Milk)', and 'Buy(Bananas)'. Each branch leads to a sequence of actions and states, eventually leading to the 'Finish' state.

```

graph TD
    Start([Start])
    GoHWS[Go(HWS)]
    GoSM[Go(SM)]
    BuyDrill[Buy(Drill)]
    BuyMilk[Buy(Milk)]
    BuyBananas[Buy(Bananas)]
    HaveDrill[Have(Drill)]
    HaveMilk[Have(Milk)]
    HaveBananas[Have(Bananas)]
    Finish([Finish])

    Start -- "At(x) ✓" --> GoHWS
    Start -- "At(x) ✓" --> GoSM
    Start -- "At(HWS) ✓" --> BuyDrill
    Start -- "At(SM) ✓" --> BuyMilk
    Start -- "At(SM) ✓" --> BuyBananas

    BuyDrill -- "Sells(HWS, Drill) ✓" --> HaveDrill
    BuyMilk -- "Sells(SM, Milk) ✓" --> HaveMilk
    BuyBananas -- "Sells(SM, Bananas) ✓" --> HaveBananas


    HaveDrill -- "At(HWS) ✓" --> GoHWS
    HaveMilk -- "At(SM) ✓" --> GoSM
    HaveBananas -- "At(SM) ✓" --> GoSM

    HaveDrill -- "At(Home) ✓" --> Finish
    HaveMilk -- "At(Home) ✓" --> Finish
    HaveBananas -- "At(Home) ✓" --> Finish
  
```

Diagram illustrating the POP (Partial Order Planning) process for a shopping task. The goal is to 'Finish' (represented by a box). The initial state is 'Start' (represented by a box). The plan consists of actions: Buy(Drill), Buy(Milk), Buy(Bananas), Have(Drill), Have(Milk), Have(Bananas), At(Home), At(SM), At(HWS), Go(HWS), Go(SM), and At(x). The diagram shows the dependencies between these actions, with red arrows indicating the order of execution. The plan is valid because all dependencies are satisfied. The diagram also shows the state of the world after each action, with the final state being 'Finish'.

[illegible]

Diagrama de uma rede semântica para o problema de comprar produtos. O nó inicial 'Start' conecta-se a 'Sells(HWS, Drill)', 'At(Home)', 'Sells(SM, Milk)' e 'Sells(SM, Bananas)'. 'At(Home)' conecta-se a 'Go(HWS)' e 'Go(SM)'. 'Go(HWS)' conecta-se a 'At(HWS)', 'Buy(Drill)' e 'Have(Drill)'. 'Go(SM)' conecta-se a 'At(SM)' e 'Buy(Bananas)'. 'At(HWS)' conecta-se a 'Sells(HWS, Drill)'. 'At(SM)' conecta-se a 'Sells(SM, Milk)' e 'Sells(SM, Bananas)'. 'Buy(Drill)' conecta-se a 'Have(Drill)'. 'Buy(Milk)' conecta-se a 'Have(Milk)'. 'Buy(Bananas)' conecta-se a 'Have(Bananas)'. 'Have(Drill)', 'Have(Milk)' e 'Have(Bananas)' conectam-se a 'Finish'. 'At(Home)' também conecta-se diretamente a 'Finish'. Há uma seta vermelha tracejada de 'At(HWS)' para 'At(Home)' com o texto 'Ameaça: Ao executar Go(SM), a pré-condição At(Home) deixa de valer'.



PCS 2009
PCS 2008
Inteligência Artificial

Profa:
Anna H. R. Costa
© 2007

Módulo:
Planejamento

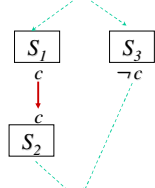
Autores:
Anna H. R. Costa
Jaime S. Sichman

Versão: 2.3
Data: 28/10/07

POP: Problema da Ameaça

24

- Ameaça
 - ocorre quando os efeitos de um passo põem em risco as pré-condições de outro
- Como testar?
 - O efeito do novo passo é inconsistente com alguma condição protegida (link causal)



S_1 ameaça a condição c estabelecida por S_1 e protegida pelo link causal de S_1 para S_2

