Capítulo 11: Planejamento

Diferenças básicas entre "resolução de problemas" e planejamento:

- Representação de estados, metas e ações.

  Planejamento: estados e metas descritos em linguagem formal (Lógica); ações descritas por precondições $\Rightarrow$ efeitos.

- Geração da sequência de ações do plano.

  Planejamento: flexibilidade na construção da sequência.

- Independência de partes do mundo com relação a outras.

  Planejamento: composição de planos a partir de subplanos (estratégia dividir-para-conquistar).

	Busca	**Planejamento**
Estados	Estrutura de dados	Sentenças lógicas
Ações	Procedimento	Precondições/resultados
Meta	Procedimento	Sentença lógica (conjunção)
Plano	Sequência	Restrições sobre ações

Planejamento no Cálculo das Situações

Estado inicial:

- $\text{at}(\text{Home}, S_0) \wedge \neg\text{have}(\text{Milk}, S_0) \wedge \neg\text{have}(\text{Bananas}, S_0)$
  $\wedge\neg\text{have}(\text{Drill}, S_0)$

Estado meta:

- $s \mid \text{at}(\text{Home}, s) \wedge \text{have}(\text{Milk}, s) \wedge \text{have}(\text{Bananas}, s)$
  $\wedge\text{have}(\text{Drill}, s)$

Operadores:

- $\forall a, s \;\; \text{have}(\text{Milk}, \text{Res}(a, s)) \Leftrightarrow$
  $[(a = \text{Buy}(\text{Milk}) \wedge \text{at}(\text{Supermarket}, s))$
  $\vee(\text{have}(\text{Milk}, s) \wedge a \neq \text{Drop}(\text{Milk}))]$

Solução:

- $s = \text{Res}(\ldots \text{Res}(\text{Buy}(\text{Milk}), \text{Res}(\text{Go}(\text{Supermarket}), S_0))) \ldots)$

Problemas do planejamento usando o Cálculo das situações:

- Ineficiência.
- Planos podem conter passos irrelevantes: $\text{Res}(A^{-1}, \text{Res}(A, s))$, ...

Em direção à prática:

- Linguagem restrita.
- Algoritmo de propósito específico: um **planejador**.

Linguagem tipo STRIPS:

- **Estado:** representado por uma conjunção de literais básicos sem funções.

  Ex: $\text{at(Home)} \land \neg\text{have(Milk)} \land \neg\text{have(Bananas)} \land \neg\text{have(Drill)} \land \cdots$

- **Meta:** representada por uma conjunção de literais.

  Ex1: $\text{at(Home)}\land\text{have(Milk)}\land\text{have(Bananas)}\land\text{have(Drill)}$

  Ex2: $\text{at}(x) \land \text{sells}(x, \text{Milk})$

- **Operadores:** 3 partes:
  - descrição da ação: um átomo (com ou sem varáveis);
  - precondição: conjunção de átomos;
  - efeito: conjunção de literais.

Exemplo de (esquema de) Operador:

$$
\begin{aligned}
&Op(\text{ACTION} : \text{Go}(there),\\
&\quad \text{PRECOND} : \text{at}(here) \land \text{path}(here, there),\\
&\quad \text{EFFECT} : \text{at}(there) \land \neg\text{at}(here))
\end{aligned}
$$

Exemplo de aplicação de Go(Supermarket):

- Estado atual:

  $\text{at}(\text{Home}), \text{path}(\text{Home}, \text{Supermarket}), \ldots$

- Após a aplicação de Go(Supermarket):

  $\neg\text{at}(\text{Home}), \text{path}(\text{Home}, \text{Supermarket}),$
  $\text{at}(\text{Supermarket}), \ldots$

Espaço de estados $\times$ espaço de planos:

- Pesquisa convencional:

  nodo = estado (situação) do mundo

  solução: caminho do estado inicial ao meta.

- Pesquisa de planejamento:

  nodo = **plano parcial**

  (**Precondição aberta:** precondição de um passo ainda não satisfeita.)

  Operadores sobre planos parciais:

  - **acrescentar um passo** ao plano (para satisfazer uma precondição aberta)
  - impor uma **ordem** entre dois passos
  - **acrescentar uma ligação** de uma ação já existente a uma precondição aberta.
  - Etc...

  A partir de planos incompletos/vagos construir paulatinamente planos completos e corretos.

Planos **parcialmente ordenados**:

- Princípio do **menor comprometimento**: só fazer escolhas **necessárias**.
- Planos parcialmente ordenados: não há ordem para alguns passos (só ordenar quando necessário).

  Exemplo: *RightShoe* e *LeftShoe*
- Não instanciar variáveis (a menos que necessário).

  Exemplo: seja a meta have(Milk). Pode-se escolher a ação $Buy(item, store)$, com:
  - $item$/Milk
  - $store$: não instanciada...

Um **plano** contém 4 componentes:

- Conjunto de passos (operadores do problema).
- Conjunto de ordenações da forma $S_i \prec S_j$.
- Conjunto de substituições da forma $x = v$ (v pode ser uma constante ou variável).
- Conjunto de ligações causais da forma $S_i \xrightarrow{c} S_j$, significando S_i *atinge a precondição* c *para* S_j.

Exemplo - operadores:

$$Op(\text{ACTION}: RightShoe, \text{PRECOND}: RightSockOn,$$
$$\text{EFFECT}: RightShoeOn)$$

$$Op(\text{ACTION}: RightSock, \text{EFFECT}: RightSockOn)$$

$$Op(\text{ACTION}: LeftShoe, \text{PRECOND}: LeftSockOn,$$
$$\text{EFFECT}: LeftShoeOn)$$

$$Op(\text{ACTION}: LeftSock, \text{EFFECT}: LeftSockOn)$$

Exemplo - plano (parcial) inicial:

$$\begin{aligned}
Plan(\text{STEPS}: \{&S_1: Op(\text{ACTION}: Start),\\
&S_2: Op(\text{ACTION}: Finish,\\
&\qquad \text{PRECOND}: RightShoeOn\\
&\qquad\qquad \wedge LeftShoeOn)\},\\
\text{ORDERINGS}&: \{S_1 \prec S_2\},\\
\text{BINDINGS}&: \{\},\\
\text{LINKS}&: \{\})
\end{aligned}$$

Plano inicial, graficamente:

Figura 11.4

O plano final:

Figura 11.5

Soluções:

- **Solução:** um plano (parcial) *completo* e *consistente*.
- **Plano completo:** toda precondição de qualquer passo é garantida por algum outro passo e não é desfeita por outro.
- **Plano consistente:** não há contradição na ordem dos passos ($S_i \prec S_j$ e $S_j \prec S_i$), e nem nas substituições para as variáveis ($v = A$ e $v = B$).

Um exemplo:

- Operador para definir estado inicial:

$$Op(\textsc{Action}: Start,$$
$$\textsc{Effect}: \text{at(Home)} \wedge \text{sells(HWS, Drill)} \wedge \text{sells(SM, Milk)} \wedge \text{sells(SM, Banana))}$$

- Operador para definir estado meta:

$$Op(\textsc{Action}: Finish,$$
$$\textsc{Precond}: \text{have(Drill)} \wedge \text{have(Milk)} \wedge \text{have(Banana)} \wedge \text{at(Home))}$$

- Ações:

$$Op(\textsc{Action}: Go(there), \textsc{Precond}: \text{at}(here),$$
$$\textsc{Effect}: \text{at}(there) \wedge \neg\text{at}(here))$$

$$Op(\textsc{Action}: Buy(x),$$
$$\textsc{Precond}: \text{at}(store) \wedge \text{sells}(store, x),$$
$$\textsc{Effect}: \text{have}(x))$$

Plano inicial:

Figura 11.6

Satisfazendo 3 precondições de *Finish* e as 3 precondições *sells*:

Figura 11.7

Seta grossa: ligação causal (protegida).

Seta fina: ordenamento.

Satisfazendo as precondições at das ações Buy:

Figura 11.7

Plano incorreto levando o agente ao HWS e ao SM:

Figura 11.8

Duas formas de resolver ameaças a uma ligação causal:

Figura 11.10

(b) **Rebaixamento** de S_3: $S_3 \prec S_1$.

(c) **Promoção** de S_3: $S_2 \prec S_3$.

Não há como resolver a ameaça do exemplo por ordenação:

- O passo *Go* que vir antes elimina a precondição at(Home) do outro.
- Rebaixamento e promoção falhando: *backtracking*.

Após remover $Start \stackrel{\text{at(Home)}}{\longrightarrow} Go(\text{SM})$:

- Outra maneira de assegurar a precondição at(x) de $Go(\text{SM})$: adicionar $Go(\text{HWS}) \stackrel{\text{at(HWS)}}{\longrightarrow} Go(\text{SM})$, etc...

Figura 11.11

Uma solução:

Figura 11.12

O algoritmo POP:

Figura 11.13