

Inteligência Artificial

Aula 13- Planejamento¹

Sílvia M.W. Moraes

Faculdade de Informática - PUCRS

September 21, 2017

¹Este material não pode ser reproduzido ou utilizado de forma parcial sem a permissão dos autores.

Sinopse

- Nesta aula, continuamos a falar de **planejamento**.
- Este material foi construído com base nos capítulos:
 - 11 do livro Artificial Intelligence – a Modern Approach de Russel & Norvig

Sumário

- 1 O que vimos ...
- 2 Representação de Problemas de Planejamento
- 3 Planejamento com busca no espaço de estados

Aulas anteriores

- Agente Reativos e Cognitivos
- Solução de Problemas
 - Representação, Espaço de Estados, Plano: sequência de ações
 - Busca sem informação
 - Busca com informação: A*, Hill Climbing, Simulated Anneling, Algoritmos Genéticos
 - Algoritmo de busca adversária: Minimax
- Introdução a Planejamento

Linguagem de Representação

- A linguagem de representação básica para planeadores clássicos é chamada de **ST**anford **R**esearch **I**nstitute **P**roblem **S**olver (STRIPS).
 - **Estados:**
 - O mundo é decomposto em condições lógicas
 - Os estados são representados por conjunções de literais positivos e literais básicos de primeira ordem
Ex1: *Pobre* \wedge *Desconhecido* (estado de um agente infeliz)
Ex2: *Em*(*Avião*₁, *Garulhos*) (localização de um avião)
 - Trabalha com a hipótese de mundo fechado: quaisquer condições não mencionadas são falsas.
 - **Objetivo:**
 - Estado parcialmente especificado.
Ex: O estado *Rico* \wedge *Famoso* \wedge *Feio* satisfaz o objetivo *Rico* \wedge *Famoso*
 - **Ações:** ...

Linguagem de Representação

- ...
- **Ações:**
 - Possui uma cabeça, lista de pré-condições e efeitos.
 - *cabeça*: nome da ação e lista de parâmetros;
 - *precondição*: literais positivos, define o que deve ser verdadeiro para a execução da ação
 - *efeito*: o que altera no estado quando a ação é executada. Quando o efeito é $\neg P$, um literal negativo, este é considerado falso.
 - Esquema de ação:

Ação(*Voar*($p, de, para$),

PRECOND: $Em(p, de) \wedge Aviao(p) \wedge Aeroporto(de) \wedge Aeroporto(para)$,

EFFECT: $\neg Em(p, de) \wedge Em(p, para)$)

Linguagem de Representação

- Exemplo 1:
 - Considerando
Ação (*Voar*(*p*, *de*, *para*),
PRECOND: $Em(p, de) \wedge Aviao(p) \wedge Aeroporto(de) \wedge Aeroporto(para)$,
EFFECT: $\neg Em(p, de) \wedge Em(p, para)$)
 - e o **estado atual**:
 $Em(P1, Garulhos) \wedge Em(P2, Galeão) \wedge Avião(P1) \wedge$
 $Avião(P2) \wedge Aeroporto(Garulhos) \wedge Aeroporto(Galeão)$
 - O estado **satisfaz a precondição** da Ação Voar:
PRECOND: $Em(p, de) \wedge Aviao(p) \wedge Aeroporto(de) \wedge Aeroporto(para)$
 - Substituindo $\{p/P_1, de/Garulhos, para/Galeão\}$ a ação *Voar*(*P*₁, *Garulhos*, *Galeão*) é **aplicável**.

Linguagem de Representação

- Exemplo 2: Transporte aéreo de Cargas

- Envolve o carregamento e descarregamento de cargas de aviões e voos desses aviões de um lugar para outro.
- O problema pode ser definido com 3 ações: *Carregar*, *Descarregar* e *Voar*

Ação(*Carregar*(c, p, a),

PRECOND: $Em(c, a) \wedge Em(p, a), Carga(c), Aviao(p) \wedge Aeroporto(a)$,

EFFECT: $\neg Em(c, a) \wedge Em(c, p)$)

Ação(*Descarregar*(c, p, a),

PRECOND: $Em(c, p) \wedge Em(p, a), Carga(c), Aviao(p) \wedge Aeroporto(a)$,

EFFECT: $Em(c, a) \wedge \neg Em(c, p)$)

Ação(*Voar*($p, de, para$),

PRECOND: $Em(p, de) \wedge Aviao(p) \wedge Aeroporto(de) \wedge Aeroporto(para)$,

EFFECT: $\neg Em(p, de) \wedge Em(p, para)$)

Linguagem de Representação

Exemplo 2: Transporte aéreo de Cargas

- **Iniciar**

$(Em(C_1, Garulhos) \wedge Em(C_2, Galeão), Em(P_1, Garulhos) \wedge Em(P_2, Galeão) \\ \wedge Carga(C_1) \wedge Carga(C_2) \wedge Avião(P_1) \wedge Avião(P_2) \\ \wedge Aeroporto(Garulhos) \wedge Aeroporto(Galeão))$

- **Objetivo** ($Em(C_1, Galeão) \wedge Em(C_2, Garulhos)$)

- O **plano** a seguir é a solução do problema:

$[Carregar(C_1, P_1, Garulhos), Voar(P_1, Garulhos, Galeão), Descarregar(C_1, P_1, Galeão), \\ Carregar(C_2, P_2, Galeão), Voar(P_1, Galeão, Garulhos), Descarregar(C_2, P_2, Garulhos)]$

Linguagem de Representação

- Exemplo 3- **Troca de Pneu**: consiste em trocar um pneu furado.
 - O **estado inicial** é o pneu furado e o estepe no porta-malas.
 $\text{Iniciar}(\text{Em}(\text{Furado}, \text{Eixo}) \wedge \text{Em}(\text{Estepe}, \text{PortaMalas}))$
 - O **objetivo** é ter um estepe em bom estado e corretamente colocada no eixo do carro.
 $\text{Objetivo}(\text{Em}(\text{Estepe}, \text{Eixo}))$
 - **Ações**: ...

Linguagem de Representação

- Exemplo 3- **Troca de Pneu:** consiste em trocar um pneu furado.

- ..

- Ações:**

Ação(Remover(Estepe, PortaMalas),

PRECOND: *Em(Estepe, PortaMalas),*

EFFECT: $\neg Em(Estepe, PortaMalas) \wedge Em(Estepe, Chão) \wedge Em(nulo, PortaMalas)$)

Ação(Remover(Furado, Eixo),

PRECOND: *Em(Furado, Eixo),*

EFFECT: $\neg Em(Furado, Eixo) \wedge Em(Furado, Chão) \wedge Em(nulo, Eixo)$)

Ação(Montar(Estepe, Eixo),

PRECOND: $Em(Estepe, Chão) \wedge Em(nulo, Eixo)$

EFFECT: $\neg Em(Estepe, Chão) \wedge \neg Em(nulo, Eixo) \wedge Em(Estepe, Eixo)$)

- Plano: ?

Linguagem de Representação

- Exemplo 3- **Troca de Pneu:** consiste em trocar um pneu furado.
- ...
- O plano a seguir é a solução do problema:
[Remover(Estepe, PortaMalas), Remover(Furado, Eixo), Montar(Estepe, Eixo)]

Linguagem de Representação

- Exemplo 4- **Mundo dos blocos**: consiste em um conjunto de blocos (cubos) dispostos sobre uma mesa.
 - Os blocos podem ser empilhados, apenas um bloco sobre outro.
 - Um braço robô pode levantar um bloco e movê-lo para outra posição, sobre a mesa ou em cima de outro bloco.
 - o braço só pode levantar um bloco de cada vez.

Ação(Mover(b, x, y),
PRECOND : Bloco(b) \wedge Sobre(b, x) \wedge Livre(b) \wedge Livre(y)
EFFECT : Sobre(b, y) \wedge \neg Sobre(b, x) \wedge Livre(b) \wedge \neg Livre(y)

Ação(MoverParaMesa(b, x),
PRECOND : Bloco(b) \wedge Sobre(b, x) \wedge Livre(b)
EFFECT : Sobre(b, Mesa) \wedge \neg Sobre(b, x) \wedge Livre(x)

Linguagem de Representação

- Exemplo 4- **Mundo dos blocos**: consiste em um conjunto de blocos (cubos) dispostos sobre uma mesa.
 - Dado o estado inicial:
$$\text{Iniciar}(\text{Sobre}(A, \text{Mesa}) \wedge \text{Sobre}(B, \text{Mesa}) \wedge \text{Sobre}(C, \text{Mesa}) \\ \wedge \text{Bloco}(A) \wedge \text{Bloco}(B) \wedge \text{Bloco}(C) \\ \wedge \text{Livre}(A) \wedge \text{Livre}(B) \wedge \text{Livre}(C))$$
 - Objetivo:
$$\text{Objetivo}(\text{Sobre}(A, B), \text{Sobre}(B, C))$$
 - Plano: ?

Linguagem de Representação

- Exemplo 4- **Mundo dos blocos**: consiste em um conjunto de blocos (cubos) dispostos sobre uma mesa.
 - ...
 - Plano: $[Mover(B, Mesa, C), Mover(A, Mesa, B)]$

Linguagem de Representação

- A linguagem STRIPS é não expressiva o suficiente para alguns domínios.
- Por essa razão, surgiram variantes dessa linguagem:
 - Action Description Language (ADL): mais expressiva e sucinta
Ação ($Voar(p : Avião, de : Aeroporto, para : Aeroporto)$,
PRECOND: $Em(p, de) \wedge (de \neq para)$,
EFFECT: $\neg Em(p, de) \wedge Em(p, para)$)

Linguagem de Representação

- Mundo dos blocos

- Em STRIPS, a ação $Mover(B, C, C)$ é permitida.
- em ADL

Ação($Mover(b, x, y)$,
 $PRECOND : Bloco(b) \wedge Sobre(b, x) \wedge Livre(b) \wedge Livre(y)$
 $\wedge (b \neq x) \wedge b \neq y \wedge (x \neq y)$,
 $EFFECT : Sobre(b, y) \wedge \neg Sobre(b, x) \wedge Livre(b) \wedge \neg Livre(y)$

Ação($MoverParaMesa(b, x)$,
 $PRECOND : Bloco(b) \wedge Sobre(b, x) \wedge Livre(b) \wedge (b \neq x)$
 $EFFECT : Sobre(b, Mesa) \wedge \neg Sobre(b, x) \wedge Livre(x)$

Linguagem de Representação

- Vários formalismos de planejamento em IA foram sistematizados em uma sintaxe-padrão denominada PDDL (Planning Domain Definition Language), o qual contém sublinguagens correspondentes a STRIPS, ADL e redes hierárquicas de tarefas.

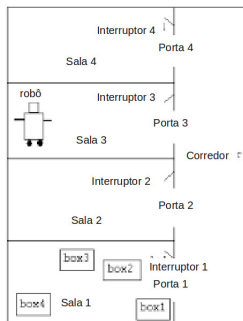
Linguagem de Representação: exercicio

- **Atividade I** - Robô Shakey: nesse problema o mundo do robô é formado por 4 salas, dispostas ao longo de um corredor, onde cada sala tem uma porta e um interruptor de luz. O robô pode movimentar-se de um lugar para outro, empurrar objetos (como caixas), subir e descer de objetos rígidos (como caixas), ligar e desligar interruptores. Ele possui as seguintes ações:
 - $Ir(x,y)$: exige que o robô esteja em x e que x e y sejam posições.
 - $Empurrar(c,x,y)$: Empurrar uma caixa da posição x para y
 - $Subir(c)/Descer(c)$: Subir em (ou Descer de) uma caixa, o robô precisa estar na posição da caixa.
 - $Ligar(i)/Desligar(i)$: Para ligar ou desligar um interruptor o robô precisa estar em cima de uma caixa e na posição do interruptor.

Linguagem de Representação: exercicio

- **Atividade I - Robô Shakey. ...**

- ...
- Descreva as ações e o estado inicial correspondente a imagem a seguir.



- Construa um plano para colocar a Caixa2 na Sala 2.

Algoritmos de Planejamento

- Para construir planos, realizamos busca no espaço de estados.
- Como as ações são especificadas em **Pré-Condições** e **Efeitos**, é possível construir um plano usando qualquer um desses elementos.
 - **Busca para frente** (Forward): a busca é feita para frente a partir do estado inicial.
 - **Busca para trás** (Backward): a busca é feita para trás a partir do objetivo.

Algoritmos de Planejamento

- **Busca para Trás** (do objetivo ao estado inicial)
 - Esse tipo de planejamento é chamado de **regressão**
 - Se o objetivo tiver restrições, esse tipo de planejamento pode ser difícil de usar
 - A linguagem **STRIPS favorece o seu uso**, pois os objetivos são representados como literais positivos.
 - Exige a implementação de uma **função predecessor**.
 - A principal vantagem é que **permite considerar apenas estados relevantes**.

