

Mudanças em Problemas de Planejamento sem Solução

Maria Viviane de Menezes ¹ Leliane Nunes de Barros ²

¹Universidade Federal do Ceará

²DCC-IME-USP

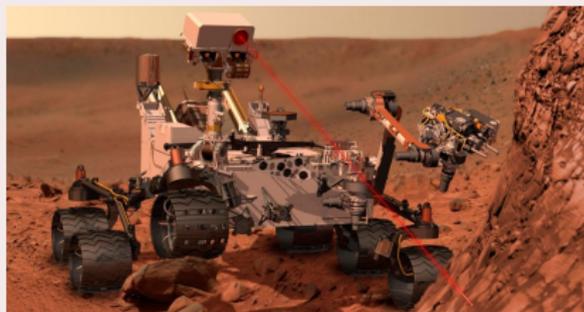
19 de novembro de 2015

Planejamento em Inteligência Artificial

Planejamento Automatizado

É a subárea da IA que se preocupa com a **escolha de ações** para que um agente alcance um dado objetivo.

Robô de Marte

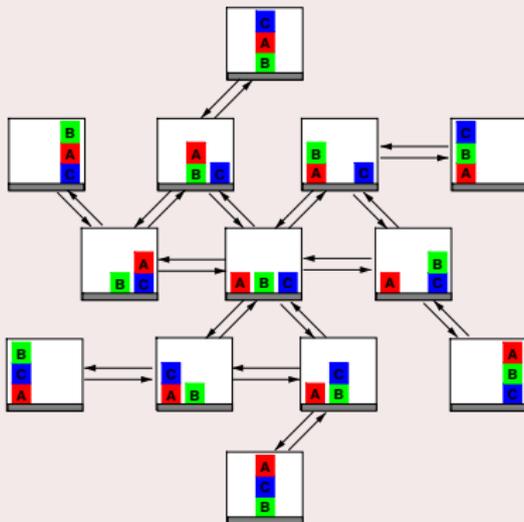


Planejamento Clássico

Suposição

- não há incerteza sobre os efeitos das ações do agente

Domínio, problema e solução

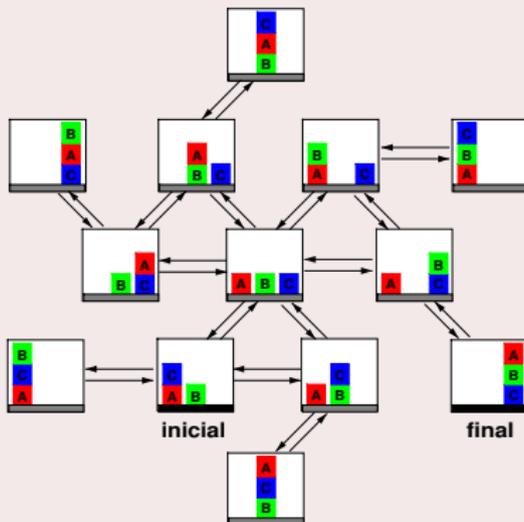


Planejamento Clássico

Suposição

- não há incerteza sobre os efeitos das ações do agente

Domínio, problema e solução

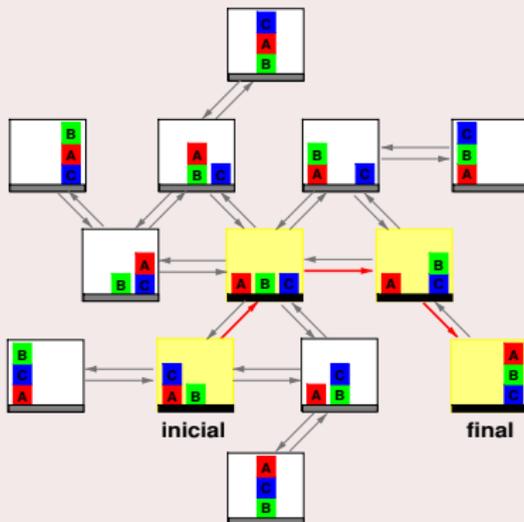


Planejamento Clássico

Suposição

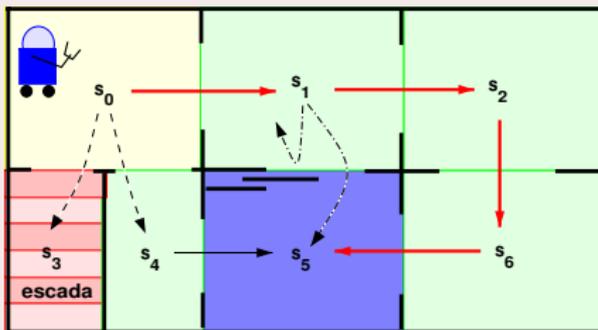
- não há incerteza sobre os efeitos das ações do agente

Domínio, problema e solução



Planejamento sob Incerteza

Problema: alcançar a sala s_5

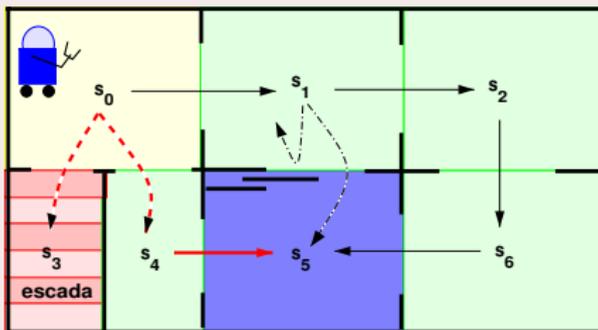


Diferentes qualidades de soluções

- $\pi_1 = \{(s_0, \text{entrar-em-}s_1), (s_1, \text{entrar-em-}s_2), (s_2, \text{entrar-em-}s_6), (s_6, \text{entrar-em-}s_5)\}$

Planejamento sob Incerteza

Problema: alcançar a sala s_5

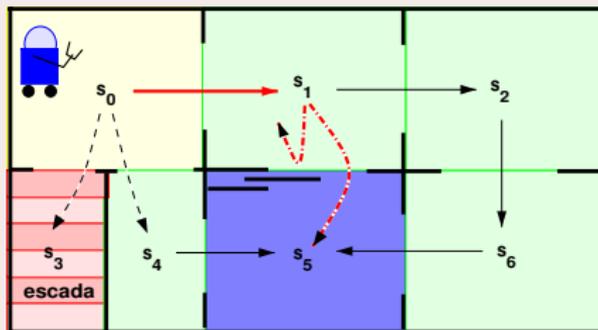


Diferentes qualidades de soluções

- $\pi_1 = \{(s_0, \text{entrar-em-}s_1), (s_1, \text{entrar-em-}s_2), (s_2, \text{entrar-em-}s_6), (s_6, \text{entrar-em-}s_5)\}$
- $\pi_2 = \{(s_0, \text{entrar-em-}s_4), (s_4, \text{entrar-em-}s_5)\}$

Planejamento sob Incerteza

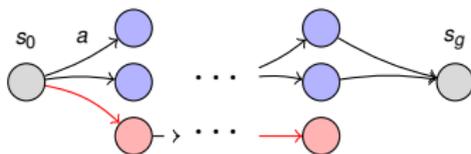
Problema: alcançar a sala s_5



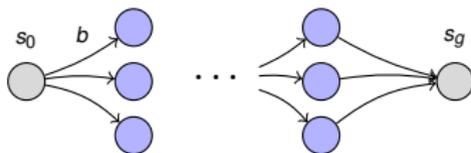
Diferentes qualidades de soluções

- $\pi_1 = \{(s_0, \text{entrar-em-}s_1), (s_1, \text{entrar-em-}s_2), (s_2, \text{entrar-em-}s_6), (s_6, \text{entrar-em-}s_5)\}$
- $\pi_2 = \{(s_0, \text{entrar-em-}s_4), (s_4, \text{entrar-em-}s_5)\}$
- $\pi_3 = \{(s_0, \text{entrar-em-}s_1), (s_1, \text{entrar-em-}s_5)\}$

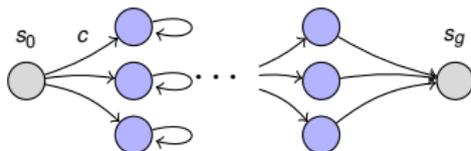
Planejamento sob Incerteza



Solução fraca: existe um caminho que leva ao estado meta.



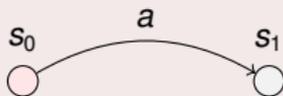
Solução forte: todo caminho leva ao estado meta.



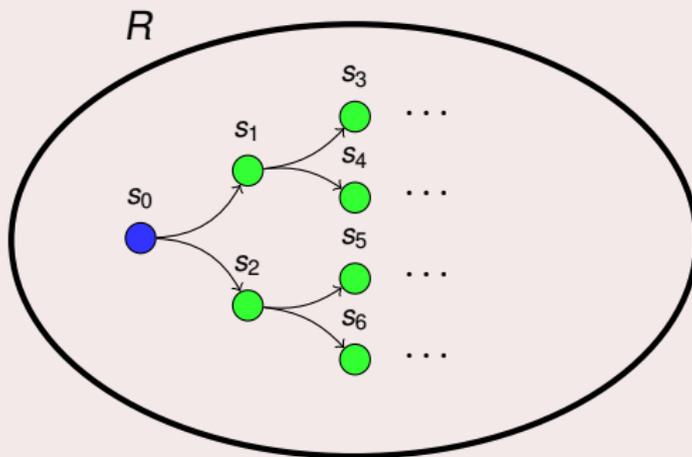
Solução forte-cíclica: todo caminho (mesmo com ciclos) leva ao estado meta.

Encontrando uma solução

Sucedores de um estado: s_1 é sucessor de s_0

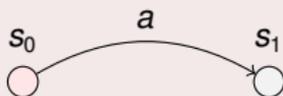


Conjunto R: Estados alcançáveis a partir do estado inicial

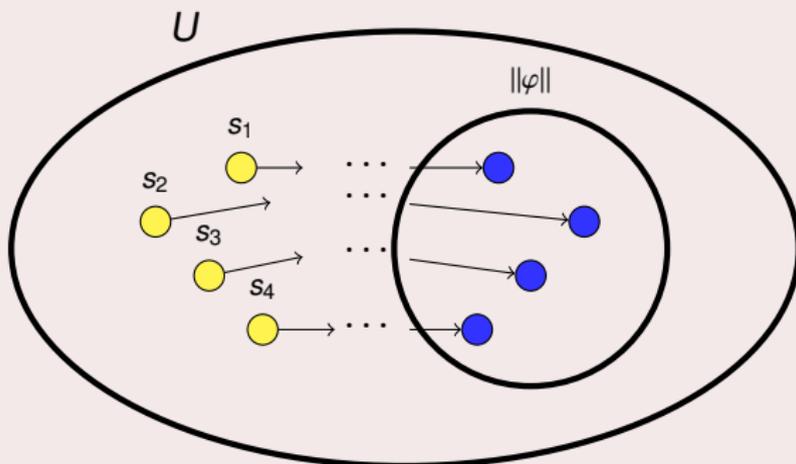


Encontrando uma solução

Predecessores de um estado: s_0 é um predecessor de s_1



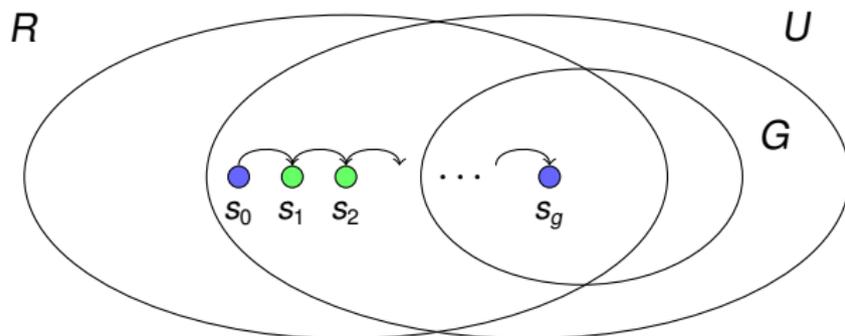
Conjunto U : Estados a partir dos quais se alcança a meta



Encontrando uma solução

Condições de Existência de Plano Solução

- $s_0 \in U$
- $R \cap U \neq \emptyset$
- $R \cap G \neq \emptyset$



O problema de verificação

Verificação de modelos

consiste em decidir se $\mathcal{K} \models \varphi$, onde:

- \mathcal{K} é um modelo formal do sistema e
- φ é uma descrição da propriedade a ser verificada



- o modelo do sistema é uma **estrutura de Kripke**
- a propriedade é especificada em **lógica temporal**

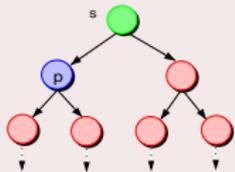
A lógica temporal CTL

Sintaxe

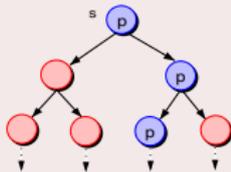
Operadores: $\exists \circ$, $\exists \square$, $\exists \diamond$, $\exists \sqcup$, $\forall \circ$, $\forall \square$, $\forall \diamond$, $\forall \sqcup$

Semântica

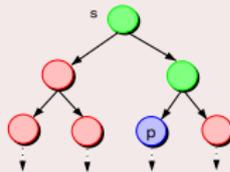
$s \models \exists \circ p$



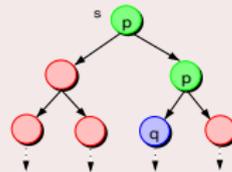
$s \models \exists \square p$



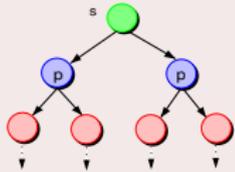
$s \models \exists \diamond p$



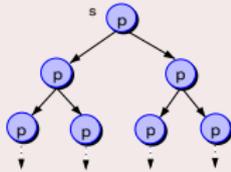
$s \models \exists (p \sqcup q)$



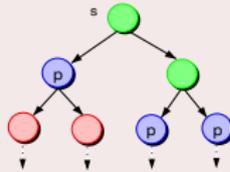
$s \models \forall \circ p$



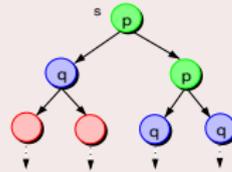
$s \models \forall \square p$



$s \models \forall \diamond p$



$s \models \forall (p \sqcup q)$



Planejamento como Verificação de Modelos

consiste em decidir se $(\mathcal{K}, s_0) \models \varphi$, onde:

- \mathcal{K} é um modelo do ambiente de planejamento
- s_0 é o estado inicial do ambiente
- φ é a meta do agente



O planejador precisa garantir a validade de φ apenas nos estados alcançáveis a partir de s_0 !

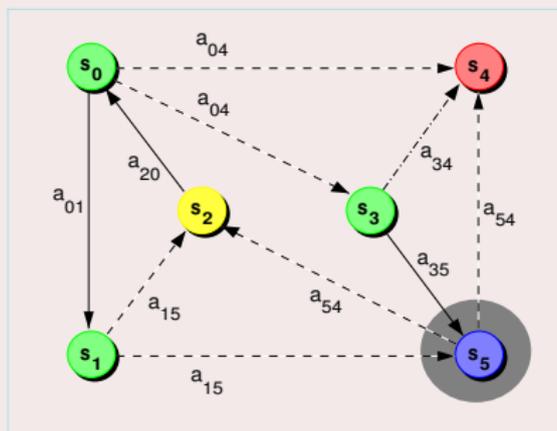
Planejamento como Verificação de Modelos

Função de pré-imagem fraca

$$\text{PRÉIMAGEMFRACA}(S') = \{(s, a) : s \in S, a \in A \text{ e } \mathcal{T}(s, a) \cap S' \neq \emptyset\}$$

Planejamento fraco para alcançar s_5 , a partir de s_0

$$\pi = \{ \}$$



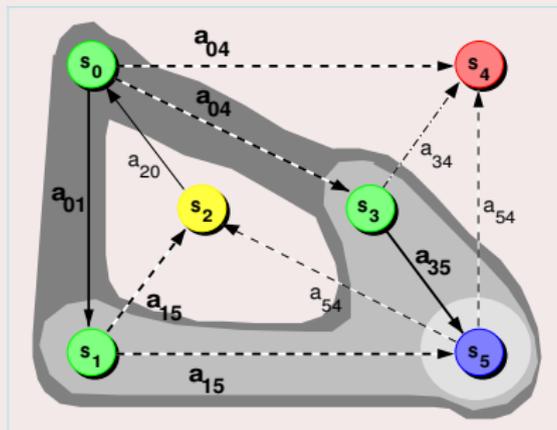
Planejamento como Verificação de Modelos

Função de pré-imagem fraca

$$\text{PRÉIMAGEMFRACA}(S') = \{(s, a) : s \in S, a \in \mathbb{A} \text{ e } \mathcal{T}(s, a) \cap S' \neq \emptyset\}$$

Planejamento fraco para alcançar s_5 , a partir de s_0

$\pi = \{(s_0, a_{01}), (s_0, a_{04}), (s_1, a_{15}), (s_3, a_{35})\}$ ← estado inicial coberto!



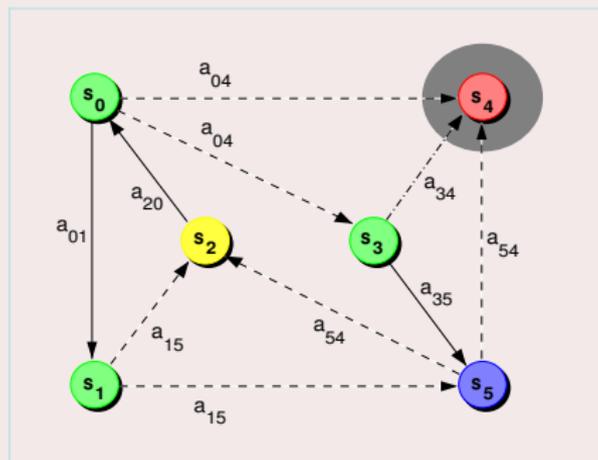
Planejamento como Verificação de Modelos

Função de pré-imagem forte

$$\text{PRÉIMAGEMFORTE}(S') = \{(s, a) : s \in S, a \in A \text{ e } \emptyset \neq \mathcal{T}(s, a) \subseteq S'\}$$

Planejamento forte para alcançar s_4 , a partir de s_0

$$\pi = \{ \}$$



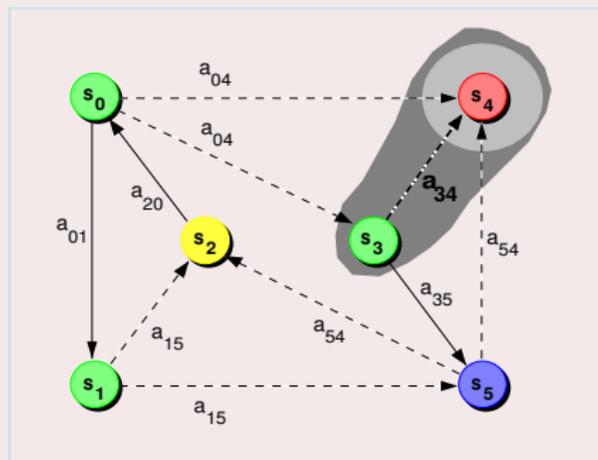
Planejamento como Verificação de Modelos

Função de pré-imagem forte

$$\text{PRÉIMAGEMFORTE}(S') = \{(s, a) : s \in S, a \in A \text{ e } \emptyset \neq \mathcal{T}(s, a) \subseteq S'\}$$

Planejamento forte para alcançar s_4 , a partir de s_0

$$\pi = \{(s_3, a_{34})\}$$



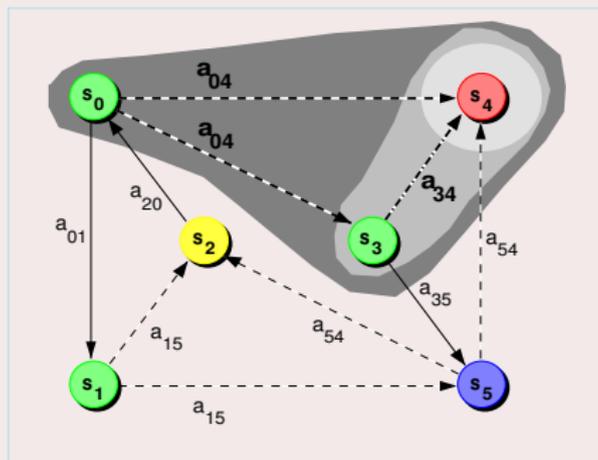
Planejamento como Verificação de Modelos

Função de pré-imagem forte

$$\text{PRÉIMAGEMFORTE}(S') = \{(s, a) : s \in S, a \in A \text{ e } \emptyset \neq \mathcal{T}(s, a) \subseteq S'\}$$

Planejamento forte para alcançar s_4 , a partir de s_0

$\pi = \{(s_0, a_{04}), (s_3, a_{34})\} \leftarrow$ estado inicial coberto!



Planejamento como Verificação de Modelos

Obtenção do conjunto U a partir de φ

Solução fraca:

$$\blacksquare U = \|\exists \diamond \varphi\| = \mu X. (\|\varphi\| \cup pre_{fraca}(X)).$$

Solução forte:

$$\blacksquare U = \|\forall \diamond \varphi\| = \mu X. (\|\varphi\| \cup pre_{forte}(X)).$$

Solução forte-cíclica:

$$\blacksquare U = \|\forall \square \exists \diamond \varphi\| = \nu X. (\mu X. (\|\varphi\| \cup pre_{fraca}(X)) \cap pre_{forte}(X)).$$

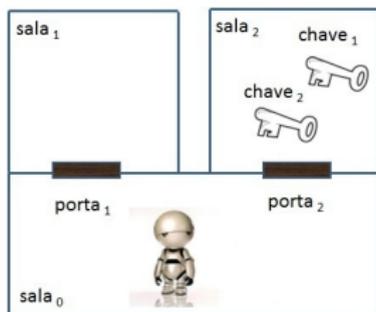
Obtenção do conjunto R a partir de s_0

$$\blacksquare R = \mu X. (\|fml(s_0)\| \cup img(X)).$$

Problemas de Planejamento sem Solução

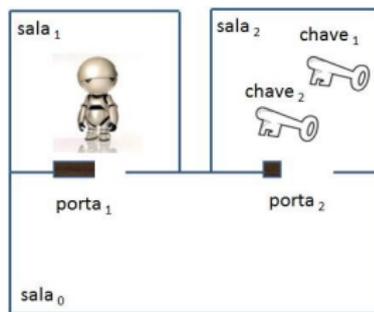
Nem sempre é possível encontrar um plano!

Domínio das Chaves



s_0 : estado inicial

X



φ : robot-pos(room₁)

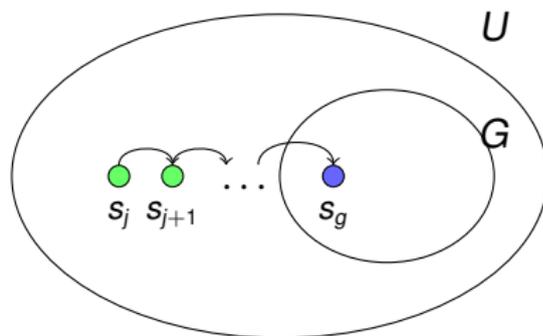
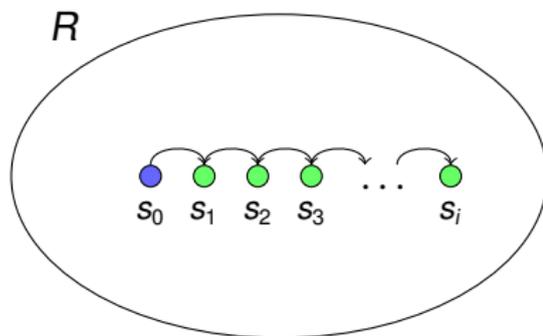
$$\Pi = \langle \mathcal{D}, s_0, \varphi \rangle$$

Π é um problema de planejamento sem solução!

Problemas de Planejamento sem Solução

Condição da Não Existência de Plano Solução

- $s_0 \notin U$
- $R \cap U = \emptyset$
- $R \cap G = \emptyset$



Mudanças em Problemas sem Solução

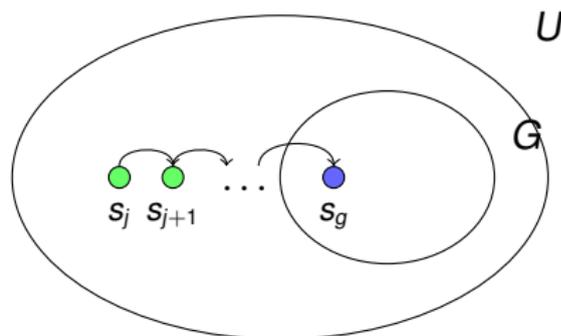
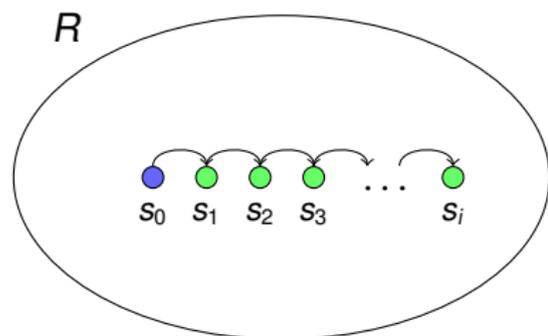
$\Pi = \langle \mathcal{D}, s_0, \varphi \rangle$ é um problema de planejamento sem solução

■ Mudança do Estado Inicial

- *selecionar um estado $s'_0 \in U$ para ser o novo estado inicial.*

■ Mudança da Meta

- *definir uma nova meta φ' tal que $R \cap G \neq \emptyset$ e $G' = \|\varphi'\|$.*



Mudanças em Problemas sem Solução

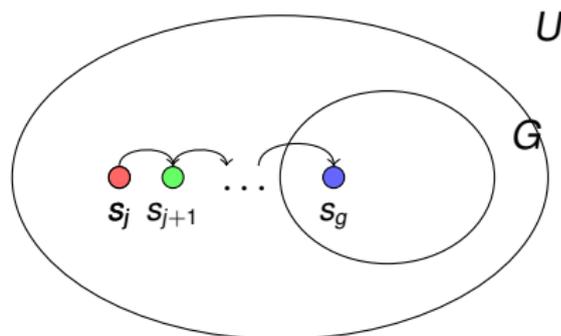
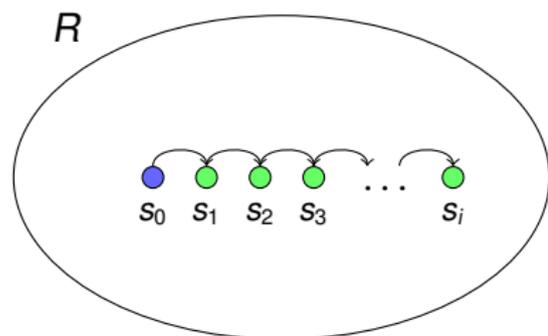
$\Pi = \langle \mathcal{D}, s_0, \varphi \rangle$ é um problema de planejamento sem solução

■ Mudança do Estado Inicial

- *selecionar um estado $s'_0 \in U$ para ser o novo estado inicial.*

■ Mudança da Meta

- *definir uma nova meta φ' tal que $R \cap G \neq \emptyset$ e $G' = \|\varphi'\|$.*



Mudanças em Problemas sem Solução

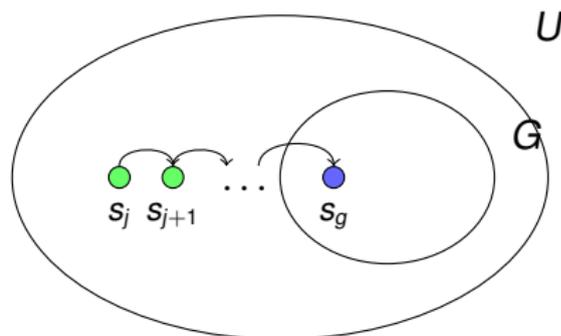
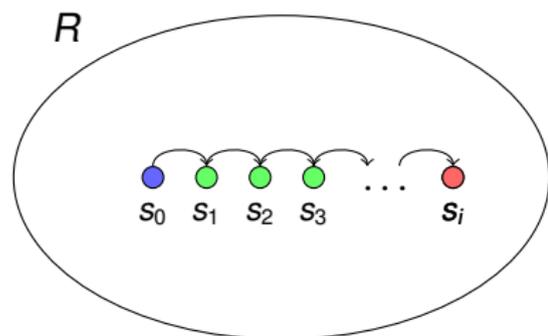
$\Pi = \langle \mathcal{D}, s_0, \varphi \rangle$ é um problema de planejamento sem solução

■ Mudança do Estado Inicial

- *selecionar um estado $s'_0 \in U$ para ser o novo estado inicial.*

■ Mudança da Meta

- *definir uma nova meta φ' tal que $R \cap G \neq \emptyset$ e $G' = \|\varphi'\|$.*



Mudanças em Problemas sem Solução

Além de verificar que não existe solução, é necessário **reparar** o modelo a fim de que haja uma solução.

Revisão e Atualização de Modelos: modificações minimais no modelo original.

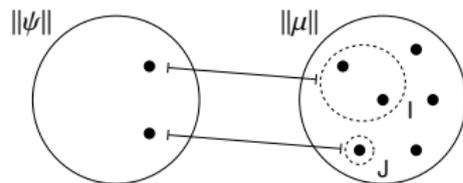
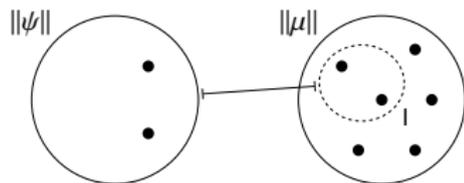
Mudanças em Problemas sem Solução

Revisão de Modelos: ($\psi * \mu$)

seleciona os modelos de μ que **minimizam a distância global** em relação a todos os modelos de ψ (Operador de Dalal).

Atualização de Modelos: ($\psi \diamond \mu$)

seleciona os modelos de μ que são minimais **em relação a cada um dos modelos** de ψ (Operador de Forbus).



Revisão e Atualização de Crenças

Distância de Hamming

$$h(s_1, s_2) = |L(s_1) - L(s_2)| = (L(s_1) \setminus L(s_2)) \cup (L(s_2) \setminus L(s_1))$$

Revisão de Dalal

$$Y \ast^{dalal} X = \{s_x \in X : \exists s_y \in Y \text{ tal que } h(s_y, s_x) \leq h(s'_y, s'_x) \forall s'_x \in X\}$$

Atualização de Forbus

$$Y \diamond^{forbus} X = \bigcup_{s_y \in Y} \{s_x \in X : \nexists s'_x \in X \text{ tal que } h(s_y, s'_x) \leq h(s_y, s_x)\}$$

Atualização do Estado Inicial

Atualização de s_0 com relação a U

- Encontrar um $s'_0 \in U$ tal que s_0 pertença ao resultado da atualização de s_0 por U

$$s_0 \diamond^{forbus} U = \{s_u \in U : \nexists s'_u \in U \text{ tal que } h(s_0, s'_u) < h(s_0, s_u)\}$$

“atualizar s_0 para que a meta seja alcançada.”

Revisão da Meta

Revisão de φ com relação a R

- Encontrar um φ' tal que $G' = \|\varphi'\|$ é conjunto devolvido pela atualização de $G = \|\varphi\|$ por R .

$$G *^{dala} R = \{s_r \in R : \nexists s'_g \in G \text{ tal que}$$

$$h(s'_g, s_r) < h(s_g, s'_r) \forall s'_r \in R\}$$

“revisar φ para que algum estado satisfazendo a meta revisada seja alcançável a partir de s_0 .”

Resumo

- Planejamento em IA
- Planejamento como Verificação de Modelos
- Problemas de Planejamento sem Solução
- Mudanças em Problemas de Planejamento sem Solução
 - Revisão do Estado Inicial
 - Atualização da Meta

Equipe deste Projeto

Pesquisadores Principais

- Maria Viviane de Menezes (UFC)
- Leliane Nunes de Barros (IME-USP)

Colaboradores

- Silvio do Lago Pereira (FATEC-SP)
- Renata Wassermann (IME-USP)
- Andreas Herzig (*Université Paul Sabatier*)

Trabalhos Realizados

- Lógica α -CTL [Pereira and Barros, 2008]

Trabalhos Realizados

- Lógica α -CTL [Pereira and Barros, 2008]
- Verificação Simbólica de Modelos com Ações.
[Menezes and Barros, 2011, Menezes and Barros, 2013]

Trabalhos Realizados

- Lógica α -CTL [Pereira and Barros, 2008]
- Verificação Simbólica de Modelos com Ações.
[Menezes and Barros, 2011, Menezes and Barros, 2013]
- Raciocínio Simbólico de Ações: Progressão e Regressão de Ações [Menezes et al., 2013, Menezes et al., 2014].

Trabalhos Realizados

- Lógica α -CTL [Pereira and Barros, 2008]
- Verificação Simbólica de Modelos com Ações.
[Menezes and Barros, 2011, Menezes and Barros, 2013]
- Raciocínio Simbólico de Ações: Progressão e Regressão de Ações [Menezes et al., 2013, Menezes et al., 2014].
- Revisão de Problemas de Planejamento com a *Lógica Dinâmica de Atribuições Proposicionais* (DL-PA) [Herzig et al., 2014]

Referências



Herzig, A., Menezes, M. V., Barros, L. N., and Wassermann, R. (2014).

On the revision of planning tasks.

In 21st European Conference on Artificial Intelligence (ECAI).



Menezes, M. V. and Barros, L. N. (2011).

Model update for automated planning.

In AAAI

SIGART Doctoral Consortium.



Menezes, M. V. and Barros, L. N. (2013).

Model checking for unsolvable planning problems.

In Proceedings of 16th Brazilian Symposium on Formal Methods (SBMF), pages 30–35.



Menezes, M. V., Barros, L. N., and Pereira, S. L. (2013).

Regressao de acoes nao determinísticas: uma solucao simbólica.