

Introdução à Inteligência Artificial

PVP 3A – Procura Cega

José Coelho,
2023



PVP 3 – Procura Cega de José Coelho é disponibilizado sob a Licença *Creative Commons-Atribuição - NãoComercial-Compartilhaqual 4.0 Internacional*

Índice

1. Problemas
2. Definição de um problema
3. Exemplos de problemas
 1. Aspirador
 2. Puzzle 8
 3. 8 Damas
 4. Partição

Problemas

- Procura de uma solução
 - Objetivo
 - O que pretende?
 - Estado inicial
 - Qual a situação atual?
 - Ações possíveis
 - O que pode fazer?
- Algoritmo Cego/não informado
 - Definição do problema
 - Estado é indivisível

Completamente / ~~parcialmente~~ observável

Determinístico / ~~aleatório~~

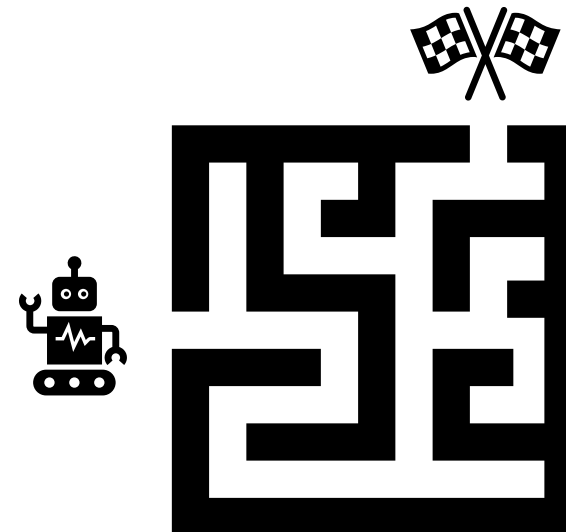
~~Episódico~~ / sequencial

Estático / ~~dinâmico~~

Discreto / ~~contínuo~~

Uni-agente / ~~multiagente~~

**Propriedades
do Ambiente**



Definição de um problema de procura

- **Objetivo**

- O que pretende atingir?

$$teste: S \rightarrow \{0,1\}$$

$$F \subseteq S$$

- **Estado inicial**

- Qual a situação atual?

$$s_0 \in S$$

- **Ações possíveis**

- O que pode fazer?

$$A = \{?\}$$

- **Modelo de transição**

- Executar uma ação num estado

$$exe: S \times A \rightarrow S \cup \{Imp.\}$$

- **Custos**

- Qual o custo de uma ação?

$$custo: S^2 \rightarrow \mathbb{N}$$

- **Sucessores**

- De um estado, para onde pode ir?

$$suc: S \rightarrow 2^S$$

$$suc(s) = \{exe(s, a) | a \in A\} \cap S$$

- **Espaço de estados**

- Quais os estados possíveis?

$$S = \{?\}$$

- **Solução**

- O que se quer saber?

$$sol \in A^d$$

- **Restrições**

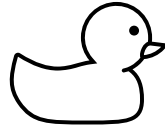
- O que não pode acontecer?

$$suc/exe$$

Exemplos de problemas de procura

- Brinquedo

- Aspirador
- Puzzle 8
- 8 Damas | K damas da mesma linha
- Ciclo do cavalo
- Inverter
- Solitário
- Mouse Trap | Sokoban
- Partição
- Tetris | Buble Breaker
- Puzzles de cartas
- Cubo mágico
- Número de Shur
- Puzzle Hitori
- Ligações entre pares de pontos



- Reais

- Trajeto

- Viajante: caminho mais curto entre duas cidades
- Visitante: passar por todos os locais de interesse
- Caixeiro Viajante: percorrer todas as cidades e voltar à mesma

- Calendarização

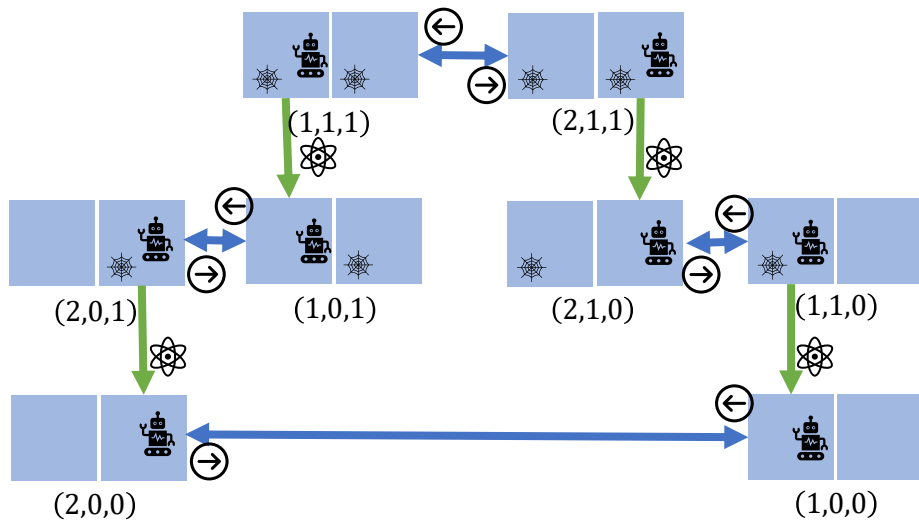
- OpenShop | FlowShop | JobShop | UPMS | RCPSP
- Horários escolares





- Espaço

- Binpacking | 2D | Problema da Mochila



Aspirador



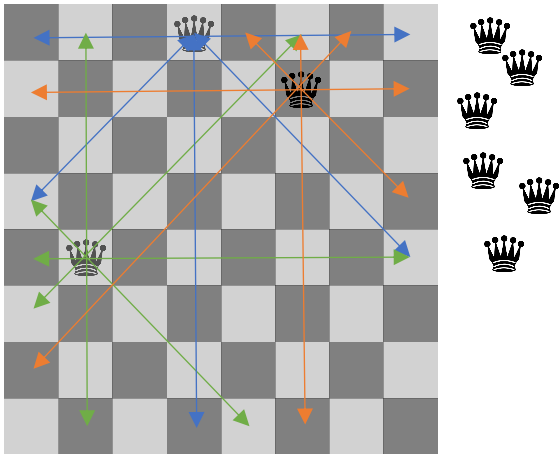
	Aspirador	
Objetivo		$F = \{(r, 0, 0) r \in \{1, 2\}\}$ $teste((r, e, d)) = (e = 0 \wedge d = 0)$
Estado Inicial		$s_0 \in S$
Ações possíveis	  	$A = \{esq, dir, asp\}$
Modelo de Transição		$exe((r, e, d), a) = \begin{cases} (1, e, d), & a = esq \\ (2, e, d), & a = dir \\ (1, 0, d), & a = asp \wedge r = 1 \\ (2, e, 0), & a = asp \wedge r = 2 \end{cases}$
Custos		$Custo((r, e, d), a) = 1$
Sucessores		-
Espaço de Estados		$S = \{1, 2\} \times \{0, 1\}^2$
Solução		$sol \in A^d$
Restrições		-

Puzzle 8



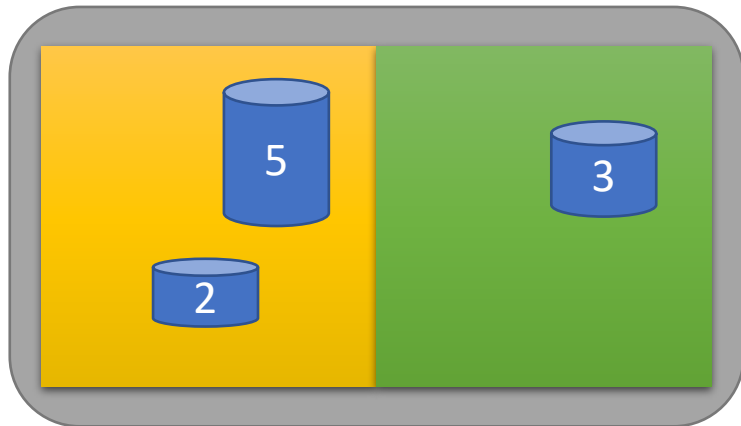
	Puzzle 8
Objetivo	$F = \{((1,2,3), (4,5,6), (7,8,0))\}$
Estado Inicial	$s_0 \in S$
Ações possíveis	$A = \{\text{cima, esq, dir, baixo}\}$
Modelo de Transição	-
Custos	-
Sucessores	-
Espaço de Estados	$S = \left\{ \begin{pmatrix} (x_1, x_2, x_3), \\ (x_4, x_5, x_6), \\ (x_7, x_8, x_9) \end{pmatrix} \in (\{0,8\}^3)^3 \mid x_i = x_j \Rightarrow i = j \right\}$
Solução	$\text{sol} \in A^d$
Restrições	$\text{sol} = (a_1, \dots, a_i, a_{i+1}, \dots, a_d) \in A^d$ $a_i = \text{esq} \Rightarrow a_{i+1} \neq \text{dir} \wedge$ $a_i = \text{dir} \Rightarrow a_{i+1} \neq \text{esq} \wedge$ $a_i = \text{cima} \Rightarrow a_{i+1} \neq \text{baixo} \wedge$ $a_i = \text{baixo} \Rightarrow a_{i+1} \neq \text{cima}$

8 Damas



	8 Damas
Objetivo	$F = \{(x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8) \in S \mid \forall_i x_i > 0\}$
Estado Inicial	$s_0 = (0,0,0,0,0,0,0,0)$
Ações possíveis	-
Modelo de Transição	-
Custos	-
Sucessores	-
Espaço de Estados	$S = \{0,8\}^8$
Solução	$s_f \in F$
Restrições	$(x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8) \in S$ $\forall_{i,j} x_i > 0 \wedge x_j > 0 \Rightarrow x_i \neq x_j \wedge x_i - x_j \neq i - j $

Partição



	Partição
Objetivo	$teste((num, esq, dir)) = num = \{\} \wedge \sum_{n \in esq} n = \sum_{m \in dir} m$
Estado Inicial	$s_0 \subseteq N \times \{\}^2$
Ações possíveis	$A = \{esq, dir\}$
Modelo de Transição	$exe((num, esq, dir), a) = \begin{cases} (num \setminus \{k\}, esq \cup \{k\}, dir), & a = esq \\ (num \setminus \{k\}, esq, dir \cup \{k\}), & a = dir \end{cases}, k = \max num$
Custos	-
Sucessores	-
Espaço de Estados	$S \subseteq N^3$
Solução	$s_f \in S \mid teste(s_f)$
Restrições	$(num, esq, dir) \in S \mid \sum_{n \in esq} n \leq \sum_{m \in (dir \cup num)} m \wedge \sum_{n \in (esq \cup num)} n \geq \sum_{m \in dir} m$

Recursos utilizados

- Microsoft Power Point
- Clipchamp, voz de síntese Duarte
- Vimeo
- Russell, S. J. & Norvig, P. (2010). Artificial intelligence: A modern approach (3rd ed). Prentice Hall.