



Escola Politécnica da USP - Depto. de Enga. Mecatrônica

PMR-3510 Inteligência Artificial

Aula 3 - Resolução de problemas em IA

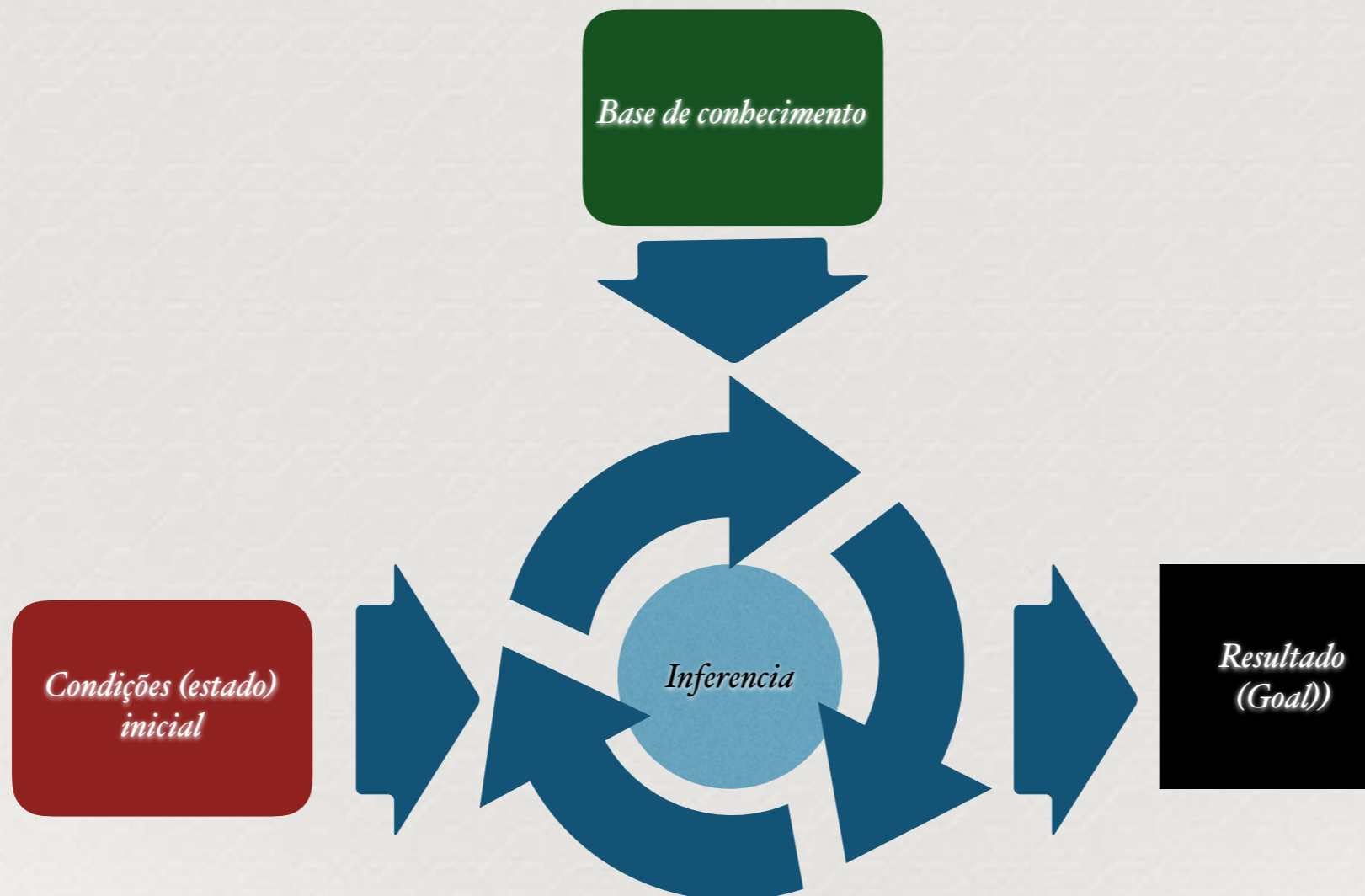
Prof. José Reinaldo Silva

reinaldo@usp.br





Em uma primeira abordagem, gostaríamos de ter “agentes inteligentes” capazes de “resolver problemas”. O que significa isso?





7	2	4
5		6
8	3	1

Start State

	1	2
3	4	5
6	7	8

Goal State



Como “resolver problemas automaticamente”

Podemos utilizar duas grande abordagens para resolver problemas automaticamente:

1. achar uma abordagem geral que leva do estado inicial ao estado final;
2. testar esta abordagem em alguns casos (sem levar em conta o tempo para chegar à solução);
3. checar se a abordagem é completa, isto é, resolve todos os casos ou há casos especiais onde o problema “não converge”;
4. preparar a implementação do revolvedor (estrutura de dados e base de conhecimento, regras de dedução);

Problema



Solução



Achar um método geral de resolução de problemas





Estrutura de um “resolvedor automático de problemas”

Para dotar uma máquina da capacidade de resolver problemas (ou uma classe de problemas) é preciso ter uma estrutura com os seguintes atributos:



1. uma descrição clara do “estado inicial” ou seja das condições iniciais do problema a ser resolvido;
2. uma descrição clara do objetivo ou “estado final”, de modo que seja possível saber quando (e se) o problema foi resolvido;
3. em cada estágio do processo de solução saber quais os próximos estados que podem ser atingidos;
4. poder escolher um (ou o melhor) caminho entre os estados acima;
5. saber que operadores (ou passos) aplicar para fazer a “transição” para um próximo estado;
6. discernir se estamos convergindo para a solução.



Nesta aula vamos discutir a primeira abordagem...

... antes porém vamos deixar claro a hierarquia de "métodos" para resolução de problemas que vamos abordar nesta e nas próximas aulas:

Paradigma de resolução: estado/transição

método geral de resolução (STRIPS)

Solução específica para os problemas


abstração



Uma solução específica para este problema... ?

7	2	4
5		6
8	3	1

Start State

	1	2
3	4	5
6	7	8

Goal State



Uma possível solução...

7	2	4
5		6
8	3	1

Start State

	1	2
3	4	5
6	7	8

Goal State

1. o problema consiste em receber uma configuração genérica de "tiles" (pastilha) - o estado inicial - e colocar os tiles numerados em ordem crescente (o estado final);
2. portanto uma possível solução específica é posicionar o tile maior e assim sucessivamente até ordenar todos;

1. achar uma abordagem geral que leva do estado inicial ao estado final; ✓
2. testar esta abordagem em alguns casos (sem levar em conta o tempo para chegar à solução); ✓
3. checar se a abordagem é completa, isto é, resolve todos os casos ou há casos especiais onde o problema "não converge";
4. preparar a implementação do revolvedor (estrutura de dados e base de conhecimento, regras de dedução);



Uma solução específica para este problema... ?

7	2	4
5		6
8	3	1

Start State

	1	2
3	4	5
6	7	8

Goal State



Uma solução específica para este problema... ?



Paradigma de resolução: estado/transição

método geral de resolução (STRIPS)

Solução específica para os problemas

Como se define “estado” para este problema?



Problem types

Deterministic, fully observable \implies single-state problem

Agent knows exactly which state it will be in; solution is a sequence

Non-observable \implies conformant problem

Agent may have no idea where it is; solution (if any) is a sequence

Nondeterministic and/or partially observable \implies contingency problem

percepts provide **new** information about current state

solution is a **contingent plan** or a **policy**

often **interleave** search, execution

Unknown state space \implies exploration problem ("online")

AIMA - UC Berkeley



The screenshot shows a web browser window with the URL www.roadef.org/challenge/2018/en/. The page features the ROADEF logo and the text "Société Française de Recherche Opérationnelle et Aide à la Décision". A navigation bar includes years from 2018 to 1999. A left sidebar lists features such as Home, Subject, Instances and Checker, Schedule, Rules, Awards, Organizing committee, Registration, Sprint submission, Sprint results, Qualification submission, and Scientific Prize. The main content area is titled "ROADEF/EURO CHALLENGE 2018: CUTTING OPTIMIZATION PROBLEM" and contains the following text:

Thanks to the success of the previous challenges, the French Operational Research (OR) and Decision Support Society (ROADEF) organizes exceptionally jointly with the European Operational Research Society (EURO) the ROADEF/EURO challenge 2018 dedicated to cutting optimization problem in collaboration with [Saint-Gobain](#).

This challenge is open to everyone, and particularly to young researchers, excluding people professionally involved with the industrial partners.

The goal of this challenge has multiple aspects.

First, it allows some of our industrial partners to follow recent developments in the fields of Operations Research and Decision Analysis.

Second, through the junior category young researchers have the opportunity to face up to a complex industrial optimization problem. The challenge will give them an opportunity to explore the requirements and difficulties encountered in industrial applications. We hope that this challenge will help to establish a permanent partnership between manufacturers and young scientists on industrial-sized projects which require both high scientific qualification and the real-life practices in companies making use of decision analysis.

Third, through the senior category, this challenge allows qualified researchers to demonstrate their knowledge and share their know-how and expertise on the practical problems. This also gives them the opportunity to establish partnerships with industrial companies.

Last, a scientific prize dedicated to qualitative submissions is proposed

At the bottom of the page, there are logos for ROADEF, EURO (The Association of European Operational Research Societies), and SAINT-GOBAIN. A footer contains the links "Home | Roadef.org | Contact".

www.roadef.org



Exemplos: i) ROADEF 2009

ROADEF 2009 Challenge: Disruption Management for Commercial Aviation

M. Palpant, M. Boudia, C.-A. Robelin, S. Gabteni, F. Laburthe

Amadeus S.A.S., Operations Research Division
485 route du pin montard, 06902 Sophia Antipolis Cedex, France
mireille.palpant@amadeus.com, mourad.boudia@amadeus.com,
charles-antoine.robelin@amadeus.com, semi.gabteni@amadeus.com,
francois.laburthe@amadeus.com

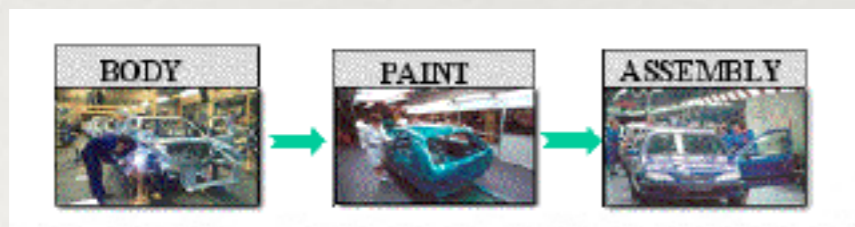
The objective function includes parameters related to additional costs or gains due to the modification of the flight plan (operating costs of added flights, deduction of operating costs for cancelled flights, costs associated with delays and cancellations of flights included in the original schedule), as well as a measure of the disutility to passengers. The objective is to minimize a weighted sum of those factors.



Exemplos: ii) ROADEF 2005



RENAULT





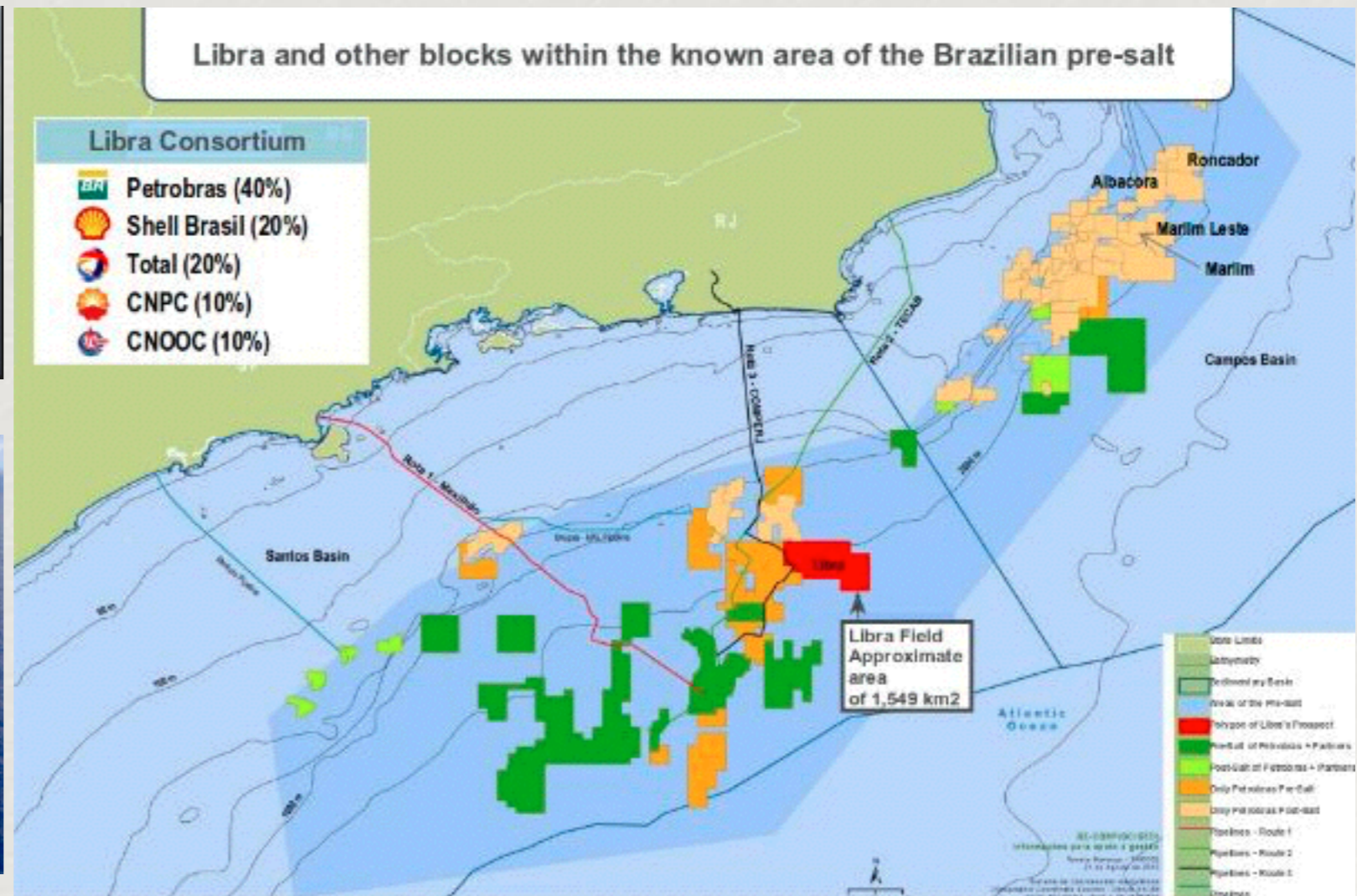
Exemplos: ii) SIPROV

Petroleum exploitation in the pre-salt layer





Exemplos: ii) SIPROV





Portanto a estratégia de solução passa por definir claramente o estado do sistema.

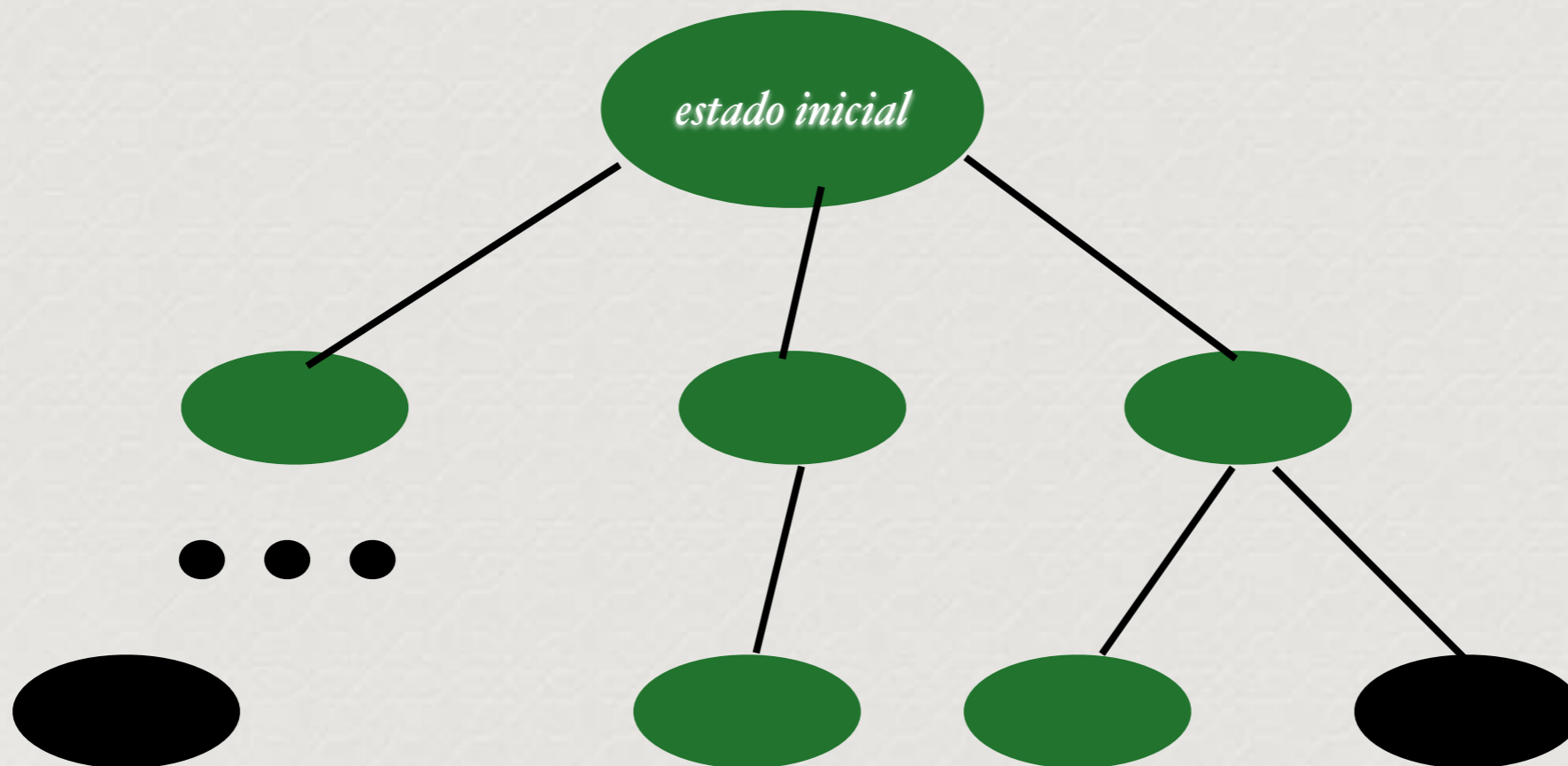
7	2	4
5		6
8	3	1

Start State

- ✦ ao definir o estado define-se também a transição (mudança de estado) já que a idéia inicial é ter um "transition system";
- ✦ voltamos então à questão do "objetivo" (goal) que é o estado final, portanto é preciso definir um critério de convergência para o estado final;
- ✦ define-se então uma "heurística" ou seja, forma de escolher "a melhor opção" entre as possíveis transições;



A estrutura do revolvedor de problemas é uma árvore, onde a raiz é o estado inicial.





7	2	4
5		6
8	3	1

Start State



	1	2
3	4	5
6	7	8



1	2	3
4	5	6
7	8	



Tree search algorithms

Basic idea:

offline, simulated exploration of state space
by generating successors of already-explored states
(a.k.a. expanding states)

```
function TREE-SEARCH(problem, strategy) returns a solution, or failure
  initialize the search tree using the initial state of problem
  loop do
    if there are no candidates for expansion then return failure
    choose a leaf node for expansion according to strategy
    if the node contains a goal state then return the corresponding solution
    else expand the node and add the resulting nodes to the search tree
  end
```

AIMA - UC Berkeley



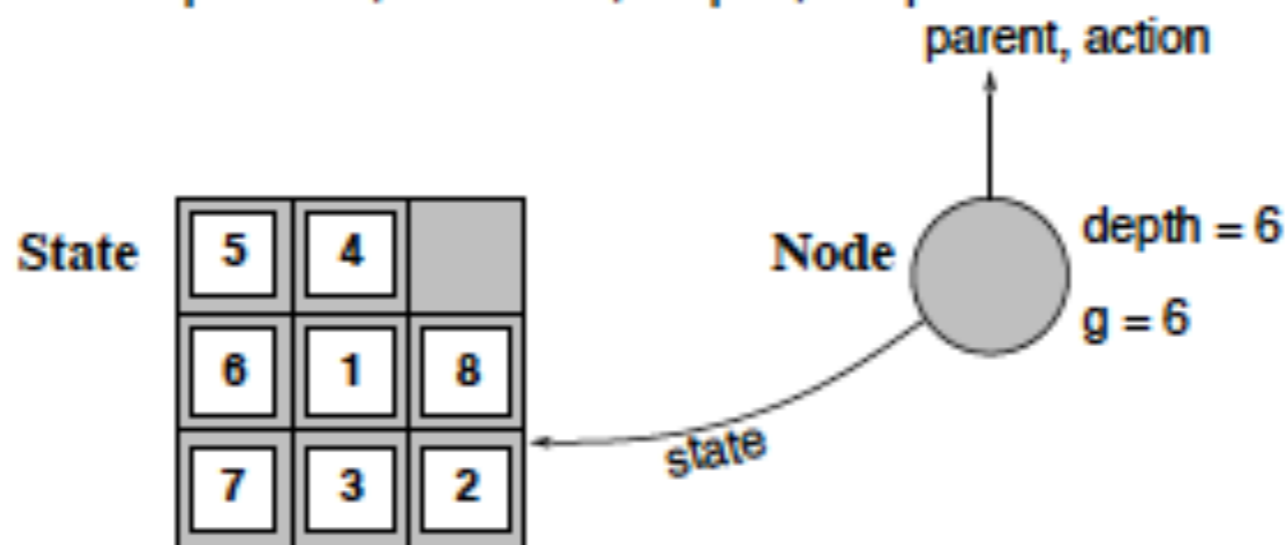
Implementation: states vs. nodes

A **state** is a (representation of) a physical configuration

A **node** is a data structure constituting part of a search tree

includes **parent**, **children**, **depth**, **path cost** $g(x)$

States do not have parents, children, depth, or path cost!



The **EXPAND** function creates new nodes, filling in the various fields and using the **SUCCESSORFN** of the problem to create the corresponding states.



Uninformed search strategies

Uninformed strategies use only the information available in the problem definition

Breadth-first search

Uniform-cost search

Depth-first search

Depth-limited search

Iterative deepening search

AIMA - UC Berkeley



Até a próxima aula a turma deverá se dividir em grupos de quatro alunos para o primeiro trabalho. Os grupos devem escolher uma aplicação e propor um programa inteligente (usando somente regras e problem solving) para ela. A solução deve ser programada em Prolog usando o sistema adotado no curso ou qualquer outro.



Até a próxima aula!