CSPB3202 Artificial Intelligence

Search

University of Colorado **Boulder**

Note: Some slides were adapted from the materials from Prof. Pieter Abbeel and Prof. Dan Klein at UC Berkeley, and Russell & Norvig’s AI book.

Informed Search- Heuristics

Review- A*

- A* uses both backward costs and (estimates of) forward costs

$$f(n) = g(n) + h(n)$$

- A* is optimal with admissible / consistent heuristics

$$0 \leq h(n) \leq h^*(n)$$

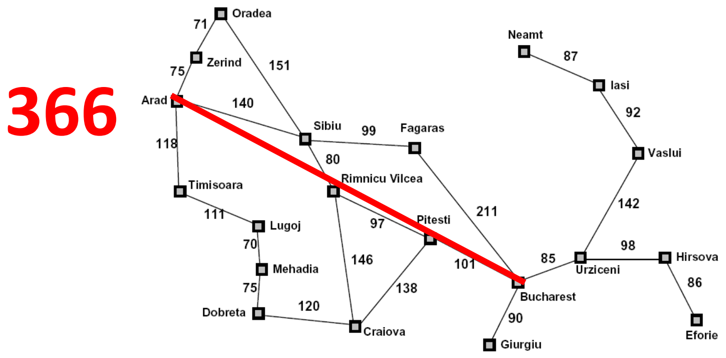
$$h(n_1) - h(n_2) \leq C(n_1, n_2)$$

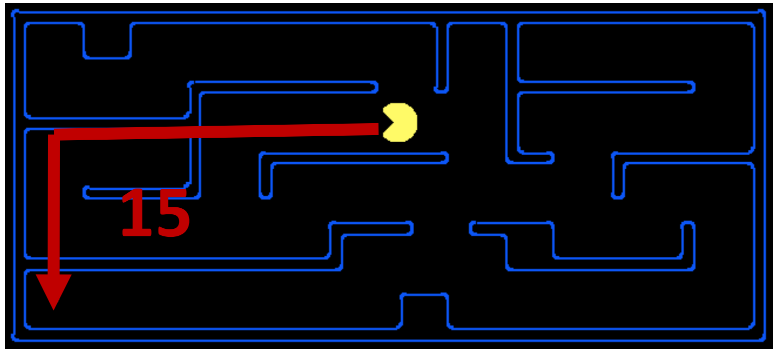
- Heuristic design is key: often use relaxed problems

Creating Admissible Heuristics

- Most of the work in solving hard search problems optimally is in coming up with admissible heuristics

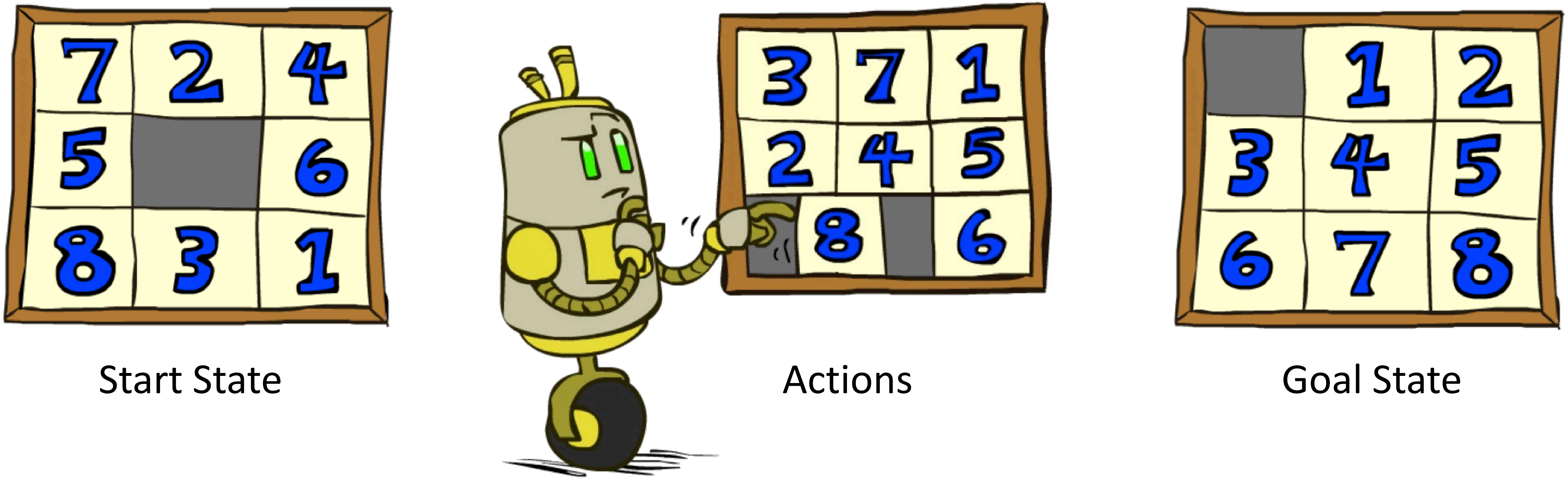
- Often, admissible heuristics are solutions to *relaxed problems,* where new actions are available





- Inadmissible heuristics are often useful too

Example: 8 Puzzle



Start State

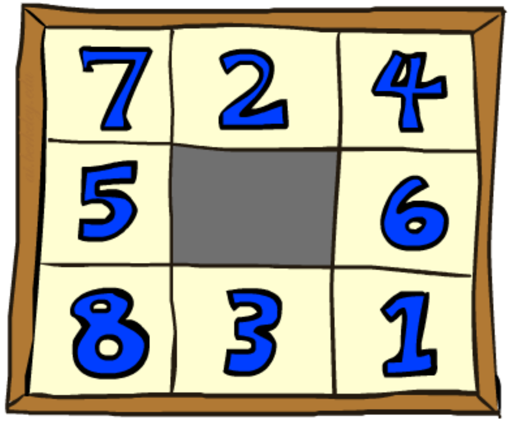
Actions

Goal State

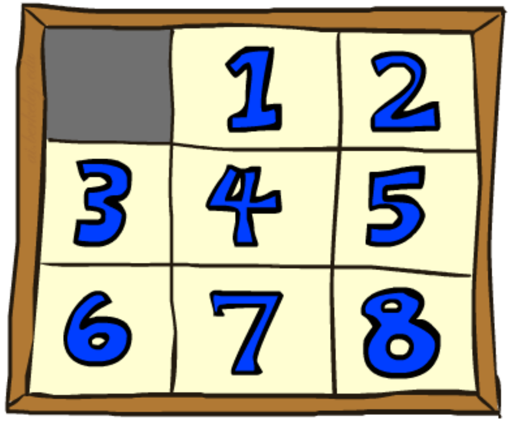
- What are the states?
- How many states?
- What are the actions?
- How many successors from the start state?
- What should the costs be?

8 Puzzle Heuristic I

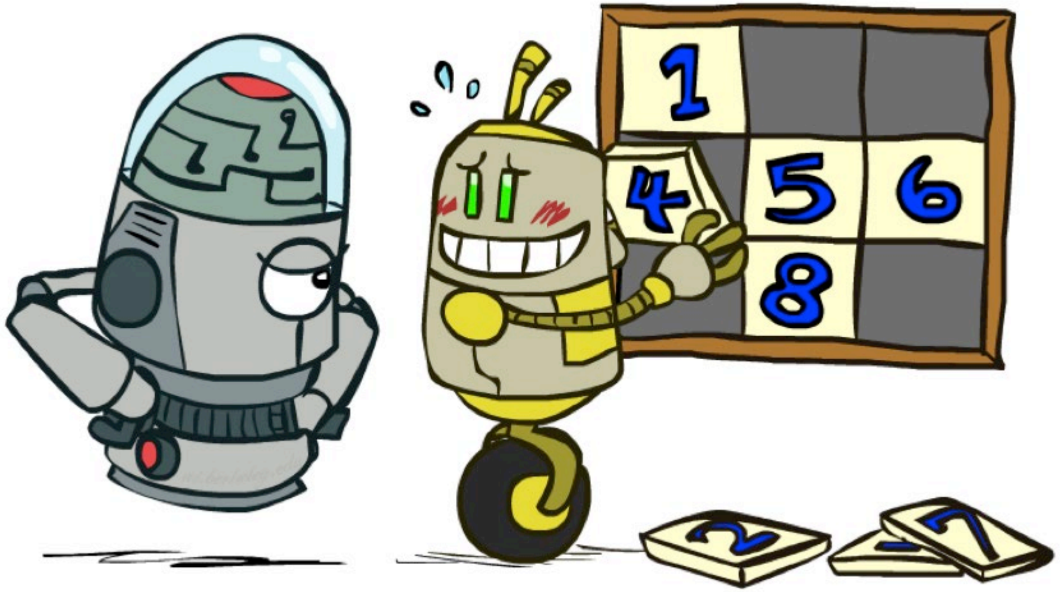
- Heuristic: Number of tiles misplaced
- Why is it admissible?
- h(start) = 8
- This is a *relaxed-problem* heuristic



Start State

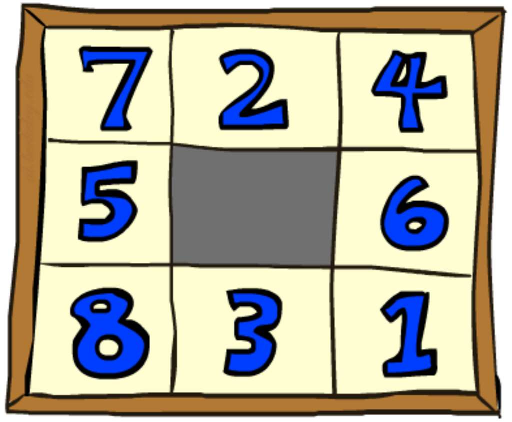


Goal State

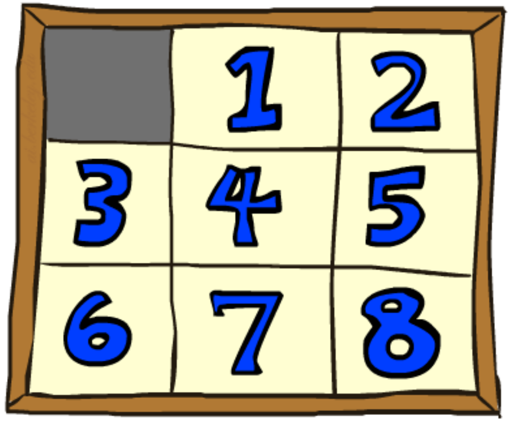


8 Puzzle Heuristic II

- What if we had an easier 8-puzzle where any tile could slide any direction at any time, ignoring other tiles?
- Total *Manhattan* distance
- Why is it admissible?
- h(start) = 3 + 1 + 2 + … = 18



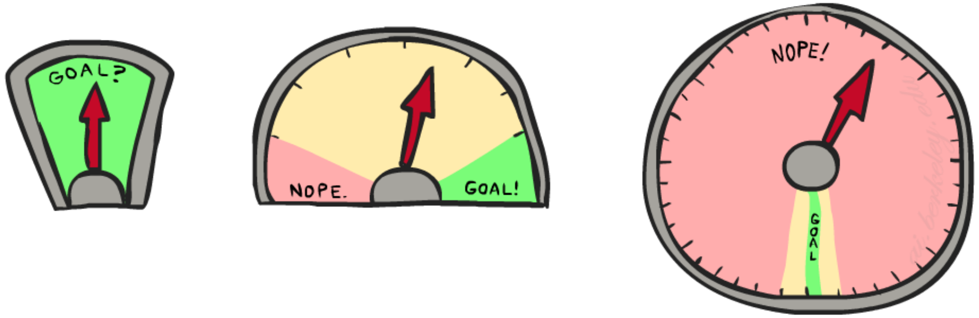
Start State



Goal State

Accuracy of a heuristic

- How about using the *actual cost* as a heuristic?
  - Would it be admissible?
  - Would we save on nodes expanded?
  - What’s wrong with it?



- With A*: a trade-off between quality of estimate and work per node
  - As heuristics get closer to the true cost, you will expand fewer nodes but usually do more work per node to compute the heuristic itself

Quality of Heuristics

Analytic Method

N Total number of nodes expanded

d The solution depth (optimal)

b^* Effective branching factor

Empirical Method

	Average nodes expanded (Search cost)			Effective Branching Factor		
Optimal Soln’	4 steps	8 steps	12 steps	4 steps	8 steps	12 steps
No h (IDS)	112	6,300	3.6×10^6	2.87	2.80	2.78
A* TILES	13	39	227	1.48	1.33	1.42
A* MANHATTAN	12	25	73	1.45	1.24	1.24