Information on paper presentations

- Find and read a recent (2002 or later) paper on a topic that interests you:
  - Google Scholar
- If there are things in the paper you don't understand, look them up, ask me or our TA, or ask your friends!
- Presentation
  - About 15 minutes
  - Summarize paper, show pictures, answer questions
  - Use Powerpoint slides or equivalent
  - Anything you still don't understand, say "I didn't understand this part"

- Let me know if there are any dates when you **cannot** present a paper in class
- Schedule will be put together by next class

Team projects

Second half of today's class will be for project brainstorming and matchmaking.

- **Reading for next two classes:**
  Textbook, Chapter 2 pp. 21-26; Chapter 3 pp. 49-80; Chapter 4, p. 90-112.

- **Optional reading:**
  Textbook, Chapter 4 pp. 112-139

- Next homework will be assigned on Monday.

Problem-solving as search

Some classic AI search problems

"Toy problems"

1. 8-puzzle
2. Missionaries and cannibals

Solving algebraic equations

Proving theorems

Robotic path planning

Game-playing

8-puzzle

initial state

4	5	
6	1	8
7	3	2

goal state

1	2	3
4	5	6
7	8	

Can use search to:
- find any solution
- find solution with minimal number of moves

What is size of state space for 8-puzzle?

Assume that from any possible configuration of the 8-Puzzle we can get to any other possible configuration.

What is size of state space for 8-puzzle?

Assume that from any possible configuration of the 8-Puzzle we can get to any other possible configuration.

Size of state space $\propto 9! = 181{,}440$

Size of 15-puzzle state space?

What is size of state space for 8-puzzle?

Assume that from any possible configuration of the 8-Puzzle we can get to any other possible configuration.

Size of state space $\propto 9! = 181{,}440$

Size of 15-puzzle state space? $\propto 16! = 2 \times 10^{13}$

What is size of state space for 8-puzzle?

Assume that from any possible configuration of the 8-Puzzle we can get to any other possible configuration.

Size of state space $\propto$ 9! = 181,440

Size of 15-puzzle state space? $\propto$ 16! = 2 x 10^{13}

Size of 24-puzzle state space?

What is size of state space for 8-puzzle?

Assume that from any possible configuration of the 8-Puzzle we can get to any other possible configuration.

Size of state space $\propto$ 9! = 181,440

Size of 15-puzzle state space? $\propto$ 16! = 2 x 10^{13}

Size of 24-puzzle state space? $\propto$ 25! = 1.5 x 10^{25}

What is size of state space for 8-puzzle?

Assume that from any possible configuration of the 8-Puzzle we can get to any other possible configuration.

Size of state space $\propto$ 9! = 181,440

Size of 15-puzzle state space? $\propto$ 16! = 2×10^{13}

Size of 24-puzzle state space? $\propto$ 25! = 1.5×10^{25}

Can't do exhaustive search!

Approximate number of states

Tic-Tac-Toe: 3^9

Checkers: 10^{40}

Rubik's cube: 10^{19}

Chess: 10^{120}

How to solve a problem by searching

1. Define search space
   - Initial, goal, and intermediate states

2. Define operators for expanding a given state into its possible successor states

3. Apply search algorithm to find path from initial to goal state, while avoiding repeating a state during the search.

Search methods

Uninformed search:

1. Breadth-first
2. Depth-first
3. Depth-limited
4. Iterative deepening depth-first
5. Bidirectional

Informed* (or *heuristic) search:

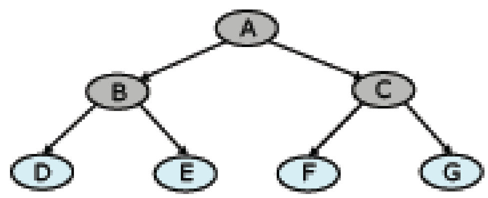
1. Greedy best-first
2. A* (and many variations)

Adversarial search:

1. Minimax with alpha-beta pruning

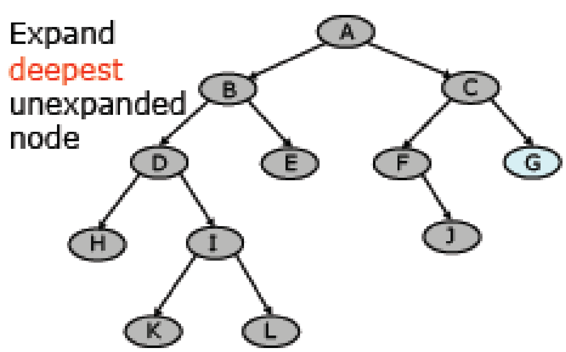
Breadth-first search

Expand all nodes at depth d before proceeding to depth d+1



Depth-first search

Expand deepest unexpanded node



Example: 8-puzzle

Informed (heuristic) Search

- What is a "heuristic"?
- Examples:
  - 8 puzzle
  - Missionaries and Cannibals
  - Tic Tac Toe

Heuristic Search

Best-first greedy search:

1. *current state* = *initial state*
2. Expand *current state*
3. Evaluate offspring states *s* with heuristic *h*(*s*), which estimates cost of path from *s* to goal state
4. *current state* = argmin_s *h*(*s*) for s ∈ *offspring*(*current state*)
5. If *current state* ≠ *goal state*, go to step 2.

Search Terminology

Completeness
- *solution will be found, if it exists*

Optimality
- *least cost solution will be found*

Admissable heuristic *h*

∀s, h never overestimates true cost from state s to goal state

Best first greedy search: Complete? Optimal?

8-puzzle heuristics: admissible?

A* Search

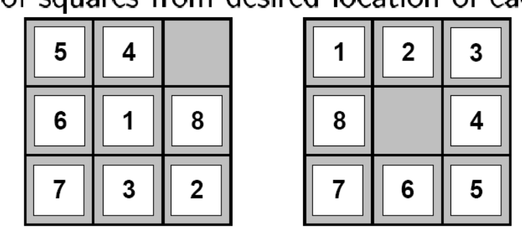
Uses evaluation function **f** = **g** + **h**

1. **g** is a cost function
   - Total cost incurred so far from initial state

2. **h** is an admissible heuristic

Best first search is A* with g = 0.

$h_1(n)$ = number of misplaced tiles
$h_2(n)$ = total Manhattan distance
(i.e., no. of squares from desired location of each tile)



Start State   Goal State

h_1(*start state*) =

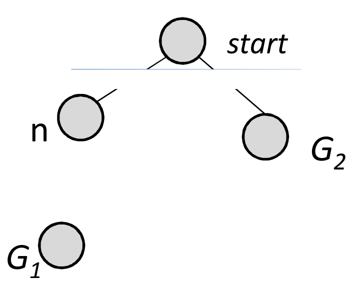
h_2(start state) =

A* search is optimal and complete

Proof of Optimality of A*

Suppose a suboptimal goal G_2 has been generated and is in the OPEN list.

Let *n* be an unexpanded node on a shortest path to an optimal goal G_1.



$f(G_2) = g(G2)$ since $h(G_2) = 0$

$> g(G_1)$ since G_2 is suboptimal

$f(G_2) > f(n)$ since h is admissible

Since $f(G_2) > f(n)$, A* will never select G_2 for expansion

Variations of A*

- IDA* (iterative deepening A*)
- ARA* (anytime repairing A*)
- D* (dynamic A*)

Applications?

Simulated Annealing

Given starting state S:

1. Set Temperature T = max Temperature
2. Calculate the energy E(S).
   Here, E(S) = f(S) = g(S) + h(S)
3. Until (T = 0 or solution found)
   S' = random-peturbation(S)
   ΔE = E(S') – E(S)
   p = exp(- ΔE / T)
   if (p > random[0,1])
   S = S'
   T = T – T_increment

Simulated Annealing

Given starting state S:

1. Set Temperature T = max Temperature

2. Calculate the energy E(S).
   Here, E(S) = f(S) = g(S) + h(S)

3. Until (T = 0 or solution found)
   S′= random-peturbation(S)
   ΔE = E(S′) – E(S)
   p = exp(- ΔE / T)
   if (p > random[0,1])
   S = S′
   T = T – T_increment

What happens if
T = 0 and T_increment = 0?

What happens if
T = max T and T_increment = 0?

What happens when T_increment is large?

What happens when T_increment is small?

Example of Simulated Annealing

Simulated Annealing is complete (if you run it for a long enough time!)

Also covered in textbook:
- Hill climbing
- Tabu search