Artificial Intelligence
CSE 5/7320

Eduardo Blanco

February 5, 2013

Informed Search

Last Lecture

- Solving problems by search
  - Problem formulation / definition
  - Search tree

- Search algorithms
  - Uninformed
    - Breadth-first
      - Complete, optimal? Time, space?
    - Depth-first
      - Complete, optimal? Time, space?
    - Depth Limited
      - Complete, optimal? Time, space?
    - Iterative deepening
      - Complete, optimal? Time, space?

Today

- Informed Search
  - Greedy best-first search
  - A*
- Heuristic Functions
- Adversarial Search

Informed Search

- Uninformed search algorithms expand nodes blindly
  - They do not use any information about the problem

- Informed (or heuristic) search
  - Use knowledge about the problem to guide the search
    - idea: expand first nodes that are more likely to lead to the optimal solution (less nodes expanded = less time wasted)
  - Always expand the node with the **lowest evaluation function**
    - Lowest = Cheapest (*more optimal*)
    - The key is to decide the evaluation function
      - It depends on the problem

- What can you use to estimate how *good* a stat is?
  - where are you coming from and how far the GOAL is

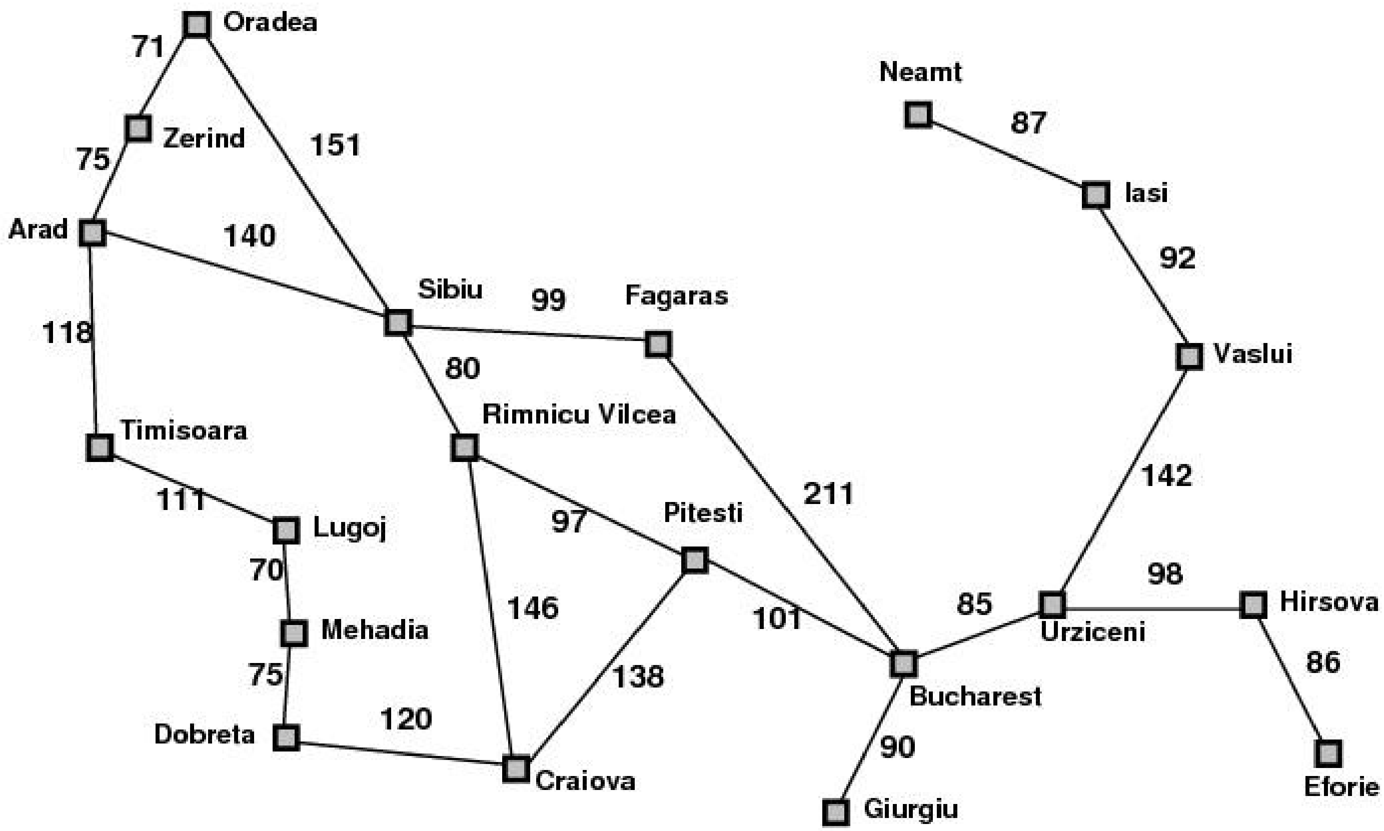
Informed Search

- Evaluation function f(n)
  - Given a node *n*, it tells you how *"good / desirable"* it is
    - You can think of it as a way of ranking nodes ready to be expanded
      - ... what is the frontier?
    - The lower the evaluation function for a node, the better

- Some notation:
  - f(n): evaluation function for node *n*
  - h(n): estimated cost of the cheapest path from state at node *n* to a GOAL state

- For now, heuristics are given (more about how to define them later)

Romanian Map



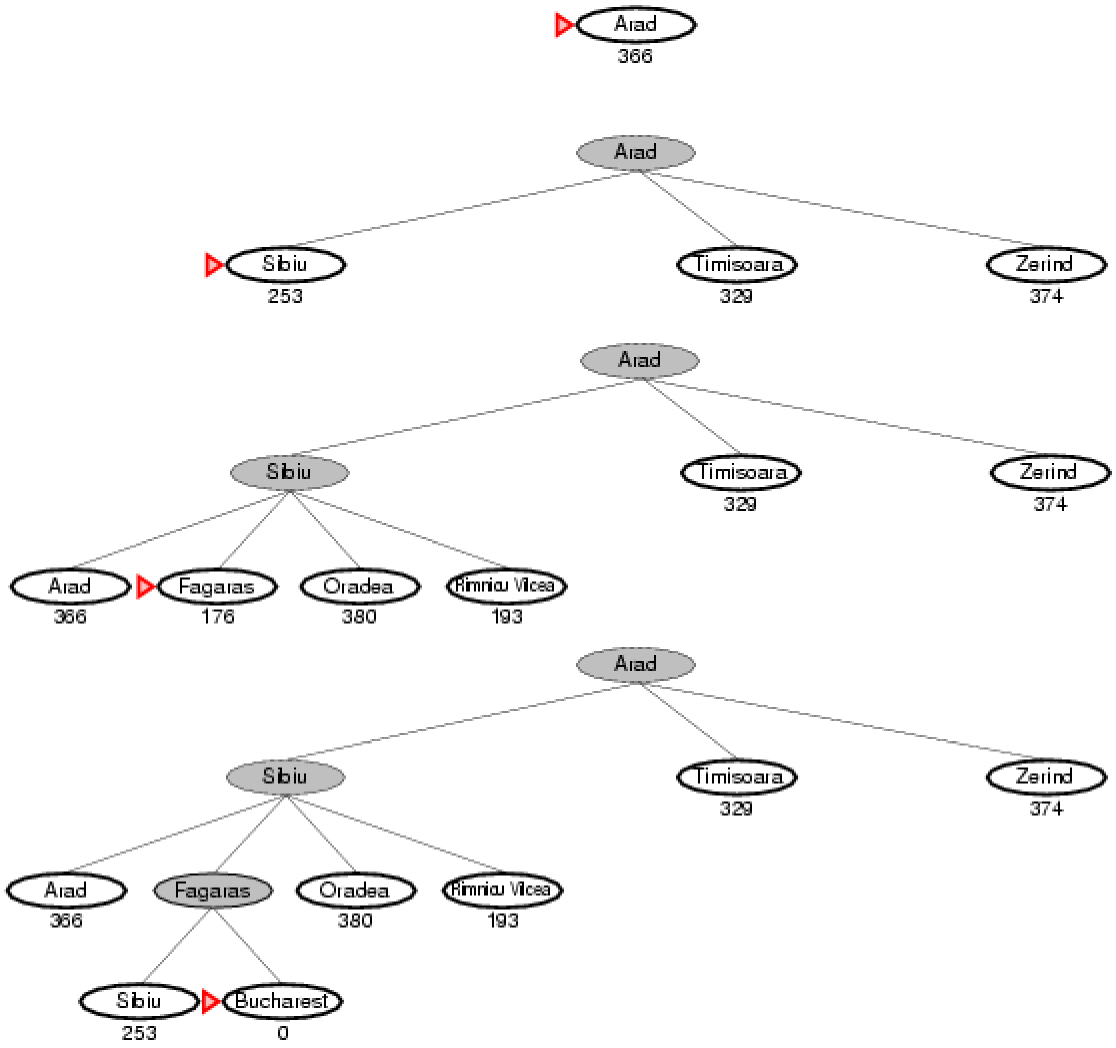
Straight−line distance to Bucharest

Arad	366
Bucharest	0
Craiova	160
Dobreta	242
Eforie	161
Fagaras	178
Giurgiu	77
Hirsova	151
Iasi	226
Lugoj	244
Mehadia	241
Neamt	234
Oradea	380
Pitesti	98
Rimnicu Vilcea	193
Sibiu	253
Timisoara	329
Urziceni	80
Vaslui	199
Zerind	374

Greedy Best-First Search

- Very simple idea: expand the node that is closest to the GOAL [as determined by h(n)]
  - we hope that the most promising state at any point is in the path to the optimal solution
  - $f(n) = h(n)$
- [Example with Romanian map, from Arad to Bucharest]

Greedy best-first search example



Greedy Best-First Search

- Complete?
- Optimal?
- Time?
- Space?

Greedy Best-First Search

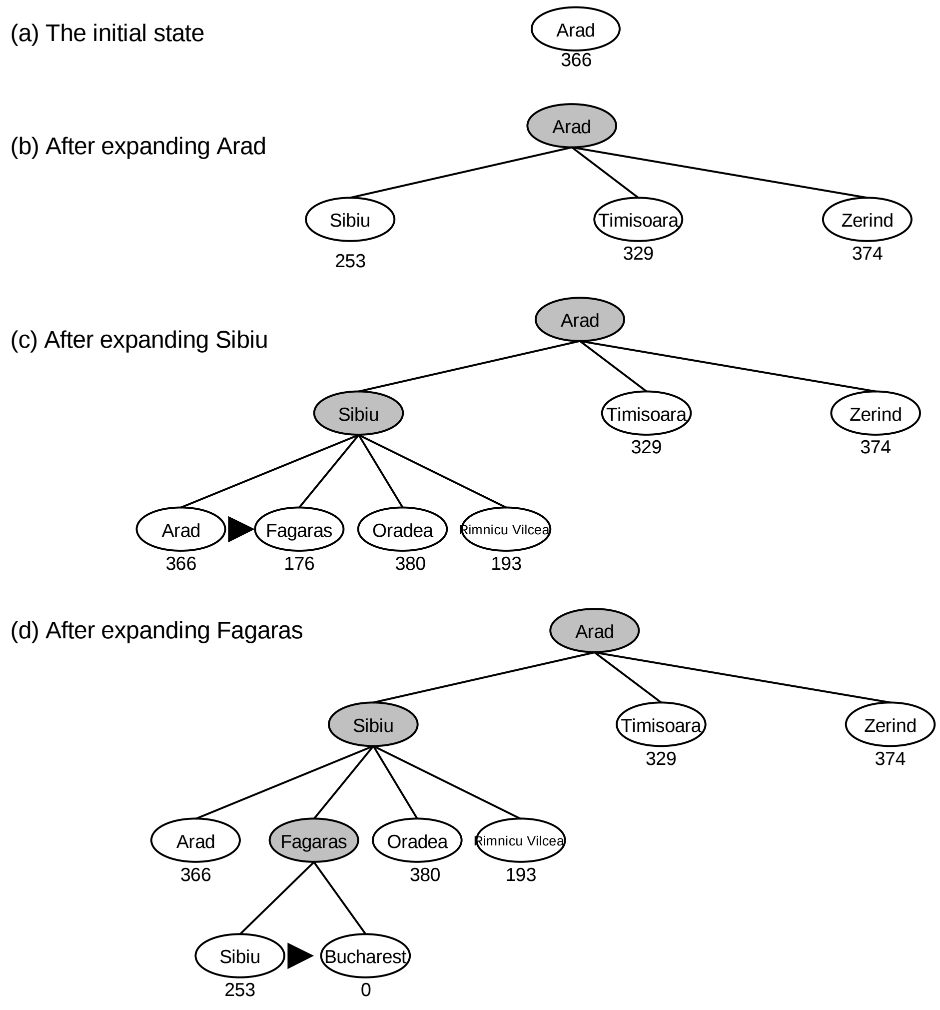
- Complete?
  - No (loops, dead ends)
    - Iasi – Neamt – Iasi – Neamt - Iasi
- Optimal?
  - No
- Time?
  - O(b^m) [worst case]
- Space?
  - O(b^m) [worst case]

A*

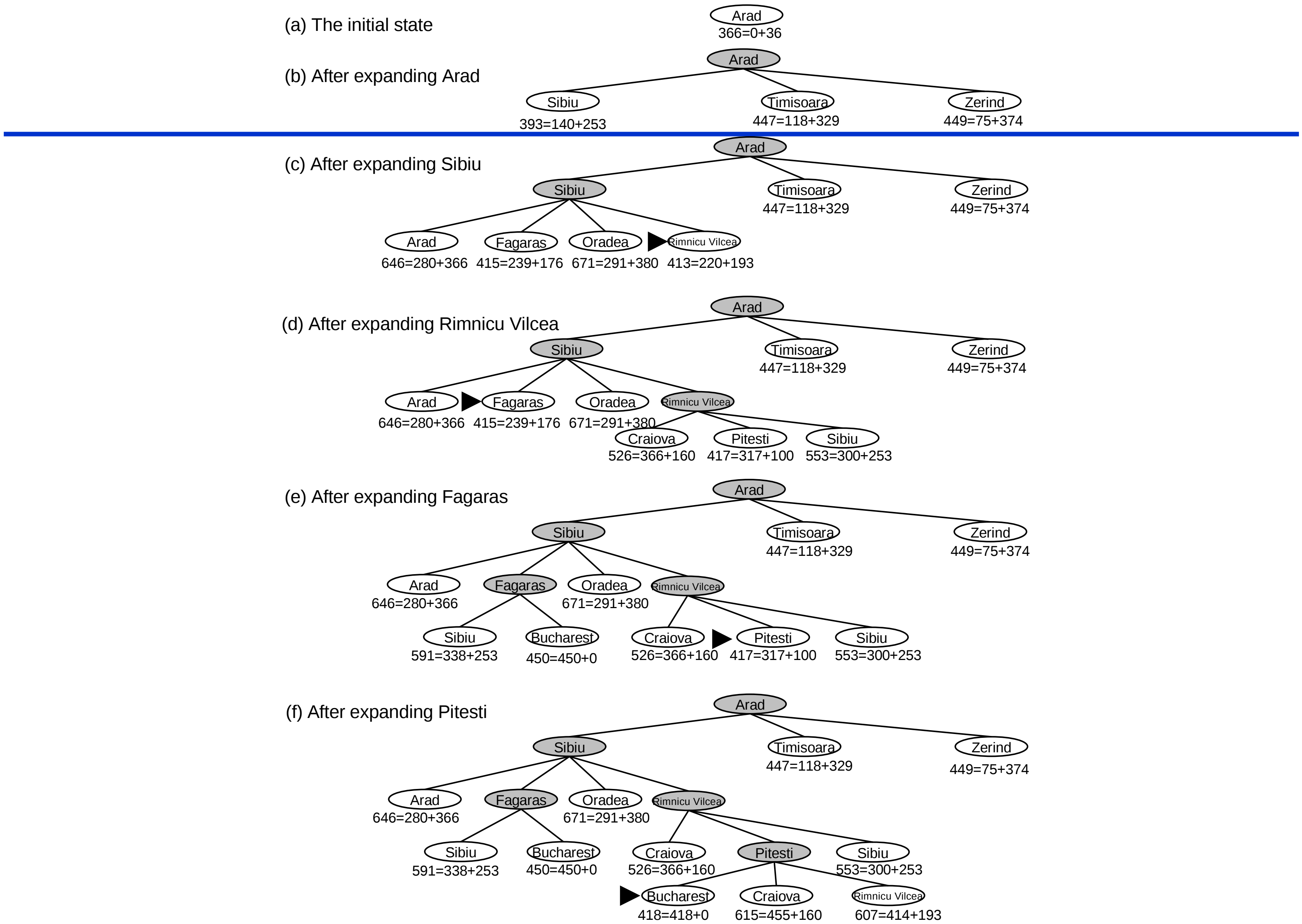
- Evaluation function: f(n) = g(n) + h(n)
  - g(n): cost so far to reach *n*
  - h(n): estimated cost to goal from *n*
  - f(n): estimated total cost of path through *n* to goal

- Is g(n) an actual cost or an estimate?
- Is h(n) an actual cost or an estimate?
- Is f(n) an actual cost or an estimate?

- [Example with Romanian map, from Arad to Bucharest]



Stages in a greedy best-first search for Bucharest using the straight-line distance heuristic h_{SLD}. Nodes are labeled with the *h*-values.



Stages in a A* search for Bucharest. Nodes are labeled with $f = g + h$. The h values are the straight-line distances to Bucharest

A* - Optimality

- Admissible heuristic:
  - An heuristic that **never overestimates**
  - So the estimated cost from node *n* to GOAL [h(n)] is the actual cost or less (the estimation is optimistic)

- Formally:
  - Let h*(n) be the actual minimal cost to reach a goal from *n*
    - [in reality, you don't know the actual cost]

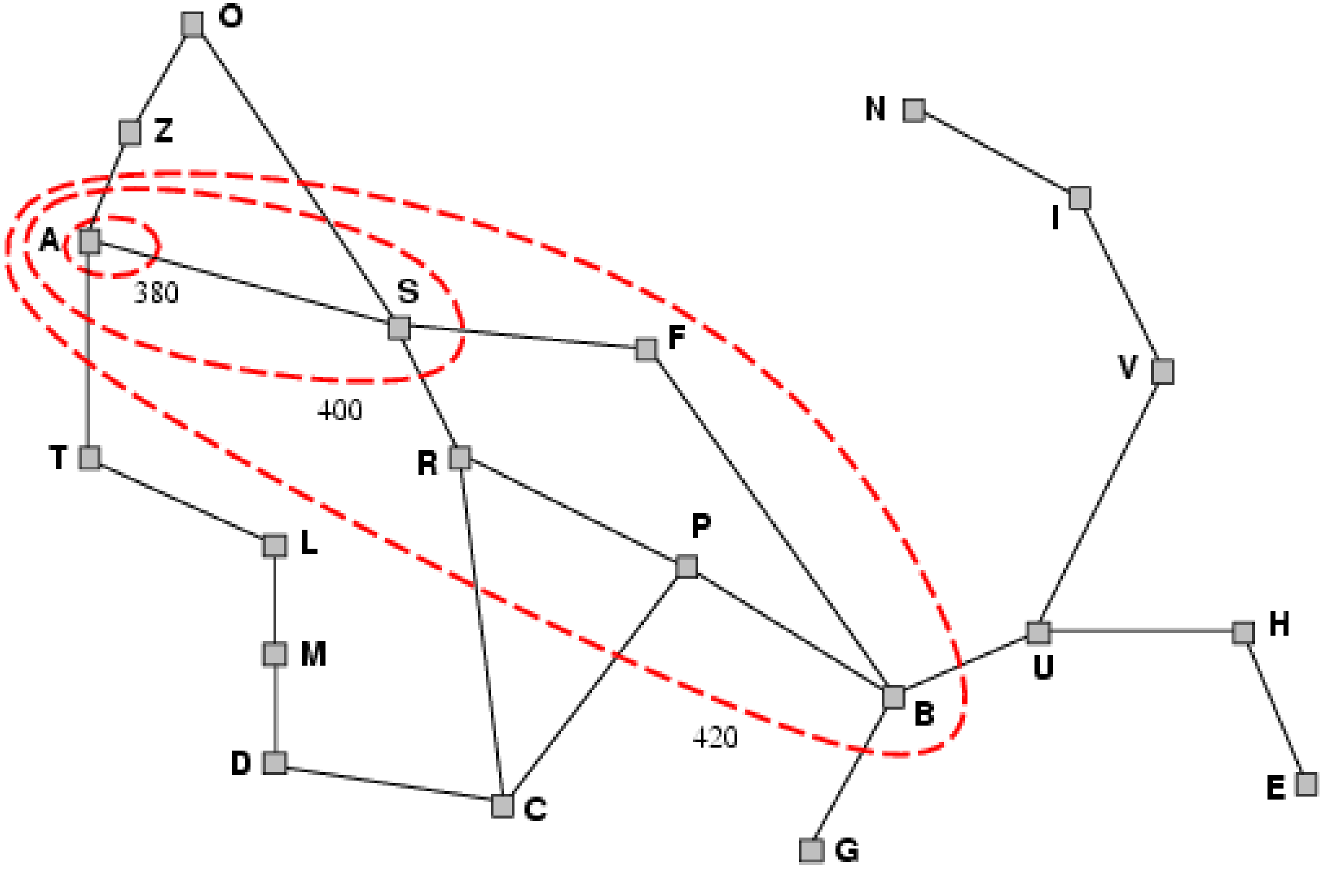
  - h(n) is admissible if
    h(n) <= h*(n) for all states *n*

- We also need h(n) >= 0, h(G) = 0 for any goal G

A* - Optimality

- Consistency (or monotonicity)
  - For every node *n* and every successor *n*' of *n* generated by action *a*,
    - h(n) <= c(n,a,n') + h(n')
    - The estimated cost of reaching the goal from *n*, h(n) is no greater than the step cost of getting to *n*', c(n,a,n') plus the estimated cost of reaching the goal from *n*', h(n')
  - Graphically: triangle inequality
    - Each side of a triangle cannot be longer than the sum of the other two sides
  - If h(n) is consistent, then the values f(n) along any path are nondecreasing.
    - Proof:

A* - Optimality



A* - Generalization

$f(n) = (1-w)g + wh$

a) where w is a constant $0 \leq w \leq 1$

for $w = 1$ greedy best-first search

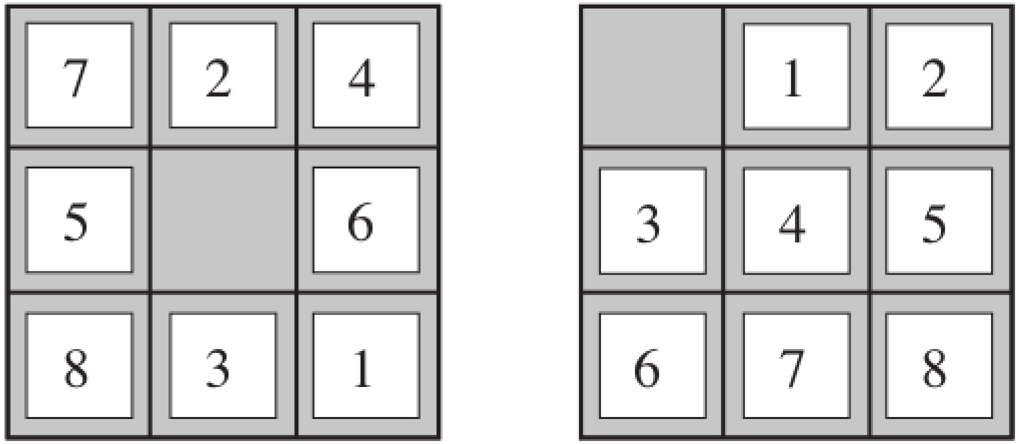
for $w = 0.5$ A*

b) w may not be constant. This is called dynamic weighting w(n) The weight is adjusted at every node.

A* - Properties

- Complete?
  - Yes
- Optimal?
  - yes
- Time?
  - Still exponential
    - how many nodes within the optimal contour?
- Space?
  - Must keep all nodes in memory

Heuristic Functions



Start State Goal State

- h1: number of tiles in the wrong position
- h2: sum of Manhattan distances of all tiles from their goal positions

- Which one is better?

Heuristic Functions

- Given two admissible heuristics h1 and h2,
  h2 **dominates** h1 if for all nodes *n*, h2(n) >= h1(n)

- Intuitively, which one is better?
  - The one that is closer to the true cost (without overestimanting)

Heuristics – how do you define them?

- Heuristics are often solution to simplified problems
  - Relaxed problem
    - Add a new action not available, examples:
      - h1: Move tile anywhere (not only to adjacent empty square)
      - h2: Move tile one square in any direction, even if square is occupied

Heuristics – how do you define them?

- If problem definition is written down in a formal language (we will see logic later in this course), relaxation can be done automatically

  - A tile cam move from square A to square B if A is horizontally or vertically adjacent to B and B is blank

  - Relaxed problem:
    - A tile can mode from square A to square B if A is adjacent to B
    - A tile can move from square A to square B if B is blank
    - A tile can move from square A to square B

Some questions and exercises

- True or False? Why?
  - An agent that senses only partial information about the state cannot be perfectly rational.
  - There exist task environments in which no pure reflex agent can behave rationally.
  - There exists a task environment in which every agent is rational.
  - A perfectly rational poker-playing agent never loses.