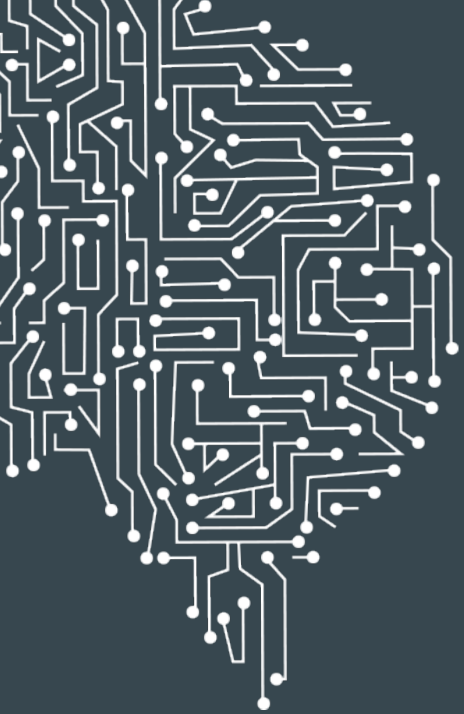


Tutorial 5

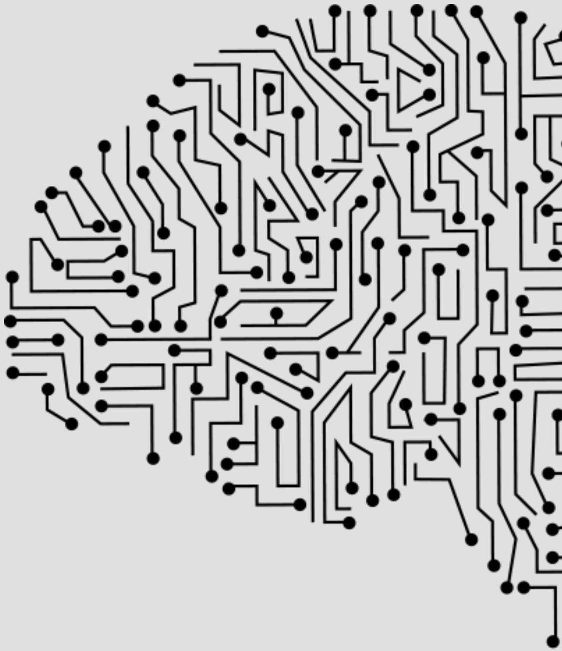
•••

Local Search

Key Points

- Local Search
    - Single Current Node
    - Move only to neighboursof that node
- Recall: Modelling Search Problems
- Midterm Practice

Question 1



Recall: Modeling Search Problems (Tut 2)

States

- State representation
- Initial state
- ~~Goal~~ Best state (use objective function) ~~(use goal test) → may have >~~1

Actions

At each state, what are the permissible actions?
~~⇒ Cost function assigns cost to each action~~

Transition model

At each state, what is the resulting next state when each of the actions is applied?

~~Goal test~~ Best State

Determines if a given state is a ~~goal~~ best (min/max) state using an **objective function**.

Question-1 States

- States: Any valid assignment of the n items to the m boxes
- Let $A = \{a_1, \dots, a_n\}, B = \{b_1, \dots, b_m\}$. Then the assignment is a function $f: A \to B$ with the constraint: $\forall b_j \in B, \sum_{a_i \in \{a_i | f(a_i) = b_j\}} s(a_i) \le c(b_j)$

  For all boxes

  Sum of size of all the stuff in that box

  Is less than the capacity of that box

- Initial state: pick any valid assignment
- Best state: Use objective function to determine

- Is it possible to have no valid states? Yes.
- $A = \{a_1\}, B = \{b_1, b_2, b_3\}, s(a_1) = 2, c(b_1) = c(b_2) = c(b_3) = 1$

Question-1 Actions

- Actions: Moving exactly one of the items into a different box
- Changing to a function f' where
- $\exists!\, a_i \in A, \exists b_j, b_k \in B: \left(f(a_i) = b_j\right) \wedge \left(f'(a_i) = b_k\right) \wedge \left(b_j \neq b_k\right)$

There exists exactly 1 item,

And boxes b_j, b_k

f assigned the item to box b_j

While f' assigned the item to box b_k

And boxes b_j and b_k are different

- Transition Model
- $T(f) = f'$

Question 1 – Objective Function

- Objective Function: Number of boxes used

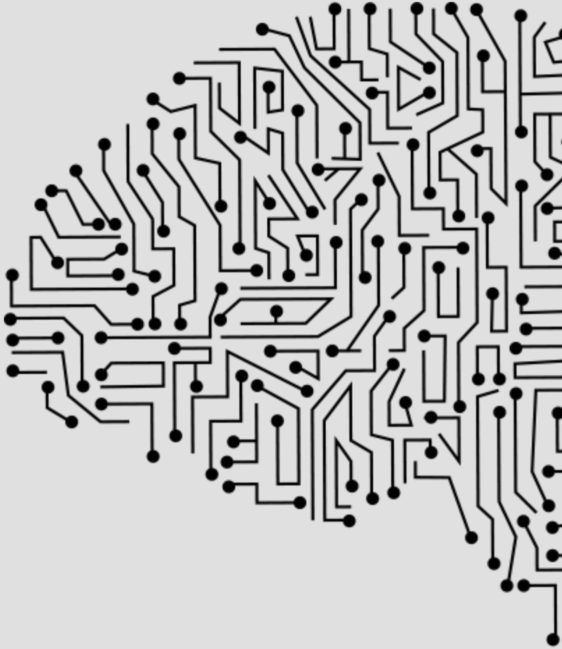
- Objective function is $g: (A \times B)^n \rightarrow \mathbb{Z}^{0^+}$
- $Let\ C = \{b_j | \exists a_i \in A: f(a_i) = b_j\}$ Set of boxes with items inside
- Then $g(\{(a_i, b_j) | f(a_i) = b_j\}) = |C|$ Number of boxes with items inside

  Set of all mappings

- Mapping an assignment – "the function" → to a value – number of boxes used

- Searching for the minimum of g

Question – 2 8-puzzle (variation)



Question 2

- Suppose you are given a 3×3 board with 8 tiles, where each tile has a distinct number between 1 to 8, and one empty space, as shown in Figure 1. You want to change the start state to goal state with the help of empty space.
- There are specific rules to solve this problem: Swapping the tile with the empty space.
    - The empty space can only move in four directions viz., up, down, right, left
    - The empty space cannot move diagonally and can take only one step at a time (i.e. move the empty space one position at a time).

2	1	3
8	6	4
7	5	

Figure 1: Start State

1	2	3
8		4
7	6	5

Figure 2: Goal State

Question 2

- You are also provided with a cost function associated with every move. The cost function is $f(state) =$ number of tiles mismatched from the goal. You will choose to move only if the cost of the next state is less than or equal to the cost of current state. Among different directions to move, you will always choose the one with the minimum cost.
- For example with the start state as shown in Figure 1, there are two possibilities to move the empty space, left and up. Figure 3 and 4 show both the moves and the associated cost with them. As the cost of moving left is less than that of moving up, you will chose to move left.

2	1	3
8	6	4
7	5	

$\longrightarrow$

2	1	3
8	6	4
7		5

Figure 3: Start state cost: $f(state) = 4$. Empty space move left, $f(state) = 3$; As after the move, 3 tiles $\{1, 2, 6\}$ are not at the required position.

2	1	3
8	6	4
7	5	

$\longrightarrow$

2	1	3
8	6	
7	5	4

Figure 4: Start state cost: $f(state) = 4$. Empty space move up, $f(state) = 5$; As after the move, 5 tiles $\{1, 2, 4, 5, 6\}$ are not at the required position.

Question 2.1

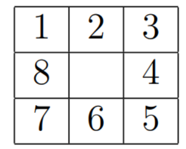
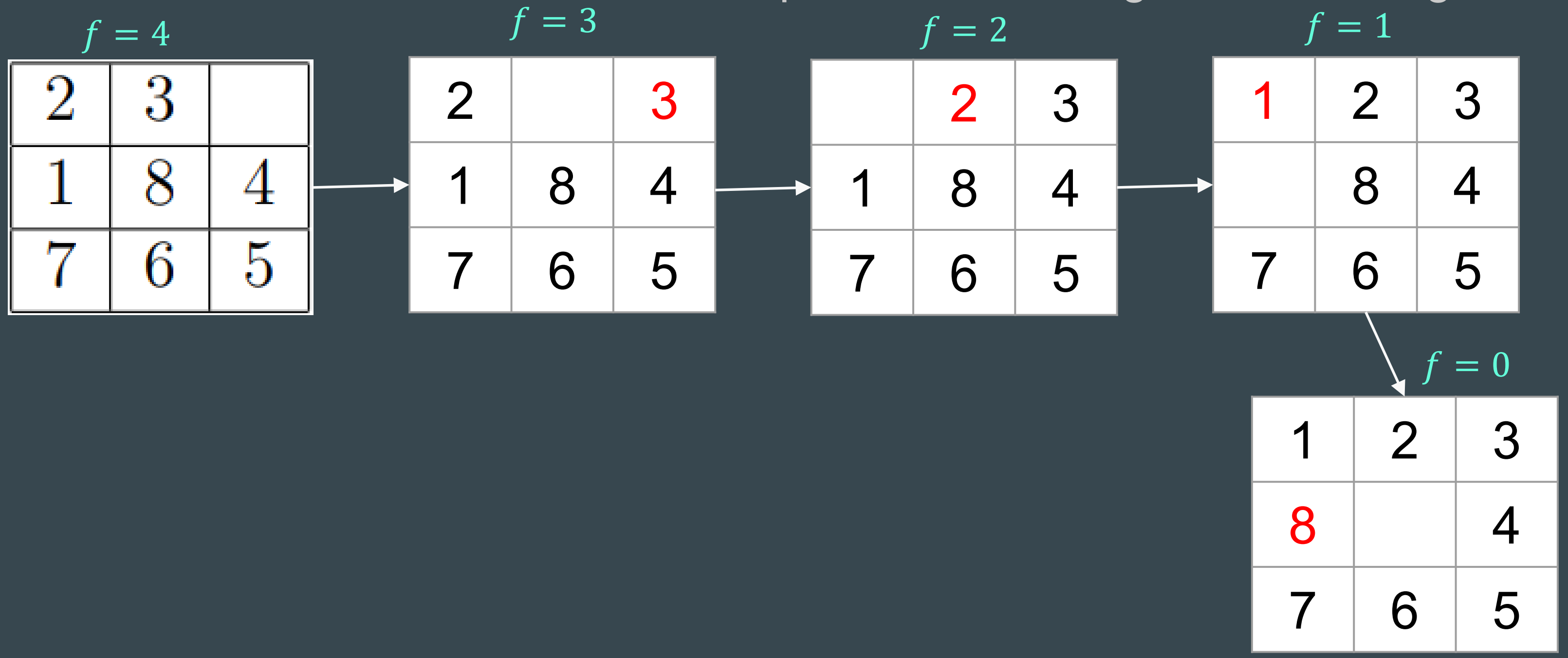


Figure 2: Goal State

- Given the start state show each step to achieve the goal state in Figure 2.



Question 2.2

1	2	3
8		4
7	6	5

Figure 2: Goal State

- Given the start state show each step to achieve the goal state in Figure 2.

$f = 5$

2	3	
1	7	4
8	6	5

$f = 4$

2		3
1	7	4
8	6	5

$f = 3$

	2	3
1	7	4
8	6	5

$f = 2$

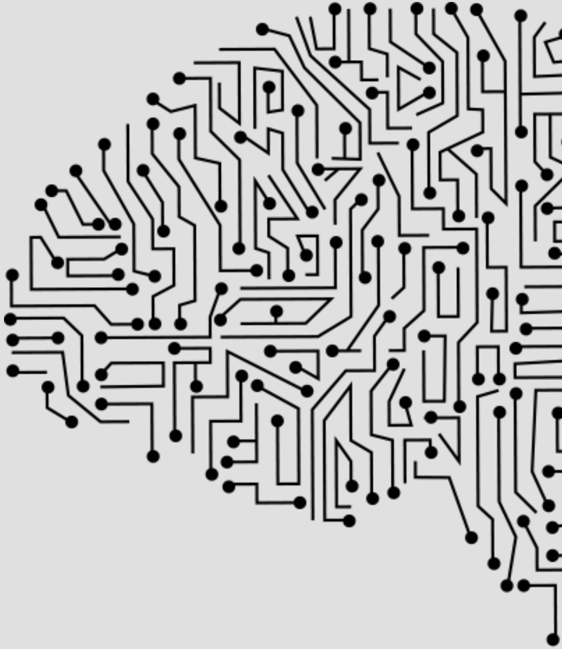
1	2	3
	7	4
8	6	5

$f = 1$

1	2	3
8	7	4
	6	5

Stuck: Any move will cause f to increase

Midterm Practice



Uninformed Search: True/False

State if each of the following statements is True or False. Provide a rationale for your answer.

- Depth-first Search is a special case of Best-first Tree Search.

True. $f(n) = -depth(n)$

- BFS is complete even if 0 step costs are allowed.

True. BFS only cares about depth, not cost.

Informed Search: True/False

State if each of the following statements is **True or False** Provide a rationale for your answer.

- If $h_1(s)$ and $h_2(s)$ are two admissible A* heuristics, let $h_3(s) = w \cdot h_1(s) + (1 - w) \cdot h_2(s)$ for any real number $w \in [0,1]$. Then $h_3(s)$ must also be admissible.

True. $h_3 = wh_1 + (1 - w)h_2 \leq \max(h_1, h_2) \leq h^*$

- h(n) = 0 is an admissible heuristic for 8-puzzle.

True. $h = 0$, the trivial heuristic, is always admissible.

Heuristic: T/F (AY19/20 S2 Midterm Q1)

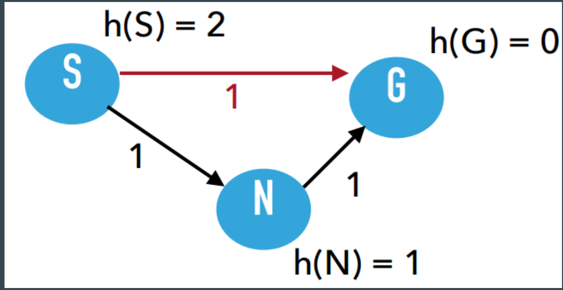
- Recall that a search problem can be defined by a transition graph $G = \langle S, E, c \rangle$ where nodes are possible states S, and there is a directed edge (n, n') if some action a can make state n transition to state n'. The weight of the edge (n, n') is given by the cost $c(n, a, n')$. For the following, we consider a consistent heuristic h under a given transition graph $\langle S, E, c \rangle$.
- (a) Suppose we add new edges to the transition graph, leaving the heuristic h unchanged. Is the heuristic h still consistent? Prove this claim or provide a counterexample.
- (b) Suppose we remove edges from the transition graph, leaving the heuristic h unchanged. Is the heuristic h still consistent? Prove this claim or provide a counterexample.

Heuristic: T/F (AY19/20 S2 Midterm Q1)

(a) Suppose we add new edges to the transition graph, leaving the heuristic unchanged. Is the heuristic h still consistent? Prove this claim or provide a counterexample.

False. Since Consistency → Admissibility, then Inadmissibility → Inconsistency.

Counterexample: Inadmissible after adding edge.



Heuristic: T/F (AY19/20 S2 Midterm Q1)

(b) Suppose we remove edges from the transition graph, leaving the heuristic h unchanged. Is the heuristic h still consistent? Prove this claim or provide a counterexample.

True. If we remove an edge, the cost of getting from a node n to another node n' can only increase. If c' is the new cost function, then $\forall n, n'$,

$$c(n, a, n') \leq c'(n, a, n')$$

$$\boldsymbol{h(n)} \leq c(n, a, n') + h(n') \leq \boldsymbol{c'(n, a, n') + h(n')}$$

Thank you!

•••