CS4023D Artificial Intelligence

Assignment 1: Structures and strategies of state space search

Part 2: State Space Representation

Algo Squad

B220424CS - Nandana Anand

B220070CS - Vaishnavi R Pai

B220475CS - Prarthana Phukan

B220587CS - Vinit Keshri

22nd Aug, 2025

1. Group Name

Algo Squad

2. Group Members

B220424CS — Nandana Anand
B220070CS — Vaishnavi R Pai
B220475CS — Prarthana Phukan
B220587CS — Vinit Keshri

3. Problem Statement (8-Puzzle Problem)

Given a 3×3 grid containing eight uniquely numbered tiles (1 to 8) and one blank space (allowing the tiles to be rearranged into different patterns).

The task is to determine an optimal sequence of valid moves that transforms a specified initial configuration (start state) into the desired final configuration (goal state). A valid move consists of sliding the blank up, down, left, or right, equivalently swapping the blank with the adjacent tile in that direction, provided the blank remains within the grid.

If the start state is not solvable, report that no solution exists.

4. State Space Representation

The state space is represented by four-tuple [N,A,S,GD] where:

1. N (Set of Nodes/States):

$N = \{n_1, n_2, n_3, n_4, \ldots\}$

Each n_i is a valid configuration of the 3×3 grid containing 8 unique numbered tiles and one blank (denoted as 0).
Only configurations reachable from the start state are included in N.

For (n-1) puzzle where $n = k^2$ and k is integer
Total number of states = n!
= 9! (here 8-puzzle problem)
= 362880

But only half of these states are reachable from a given state. That is, only half of the states are solvable.

For the 8-puzzle, although there are 9!=362,880 possible arrangements, only half (9!/2=181,440) are reachable, so the effective state space size is 181,440.

Eg- (Refer Fig. 1)

(Each tuple is a linearized representation of the 3×3 grid in row-major order).

n_1= (1, 4, 3, 7, 0, 6, 5, 8, 2), n_2=(1, 0, 3, 7, 4, 6, 5, 8, 2), n_3 = (1, 4, 3, 0, 7, 6, 5, 8, 2), n_4=(1, 4, 3, 7, 8, 6, 5, 0, 2), ...

Reachability: Two configurations are mutually reachable iff they have the same inversion parity. The inversion of a state can be defined as: Given a linear arrangement of the tiles (ignoring the blank 0), an inversion is a pair of tiles (a, b) such that

a appears before b in the sequence,
but a > b numerically.

The inversion count (number of such pairs of tiles which are inversions) can be odd or even, and in short we can call a state even or odd. This is called a state's **parity**.

Eg-

n_1= (1, 4, 3, 7, 0, 6, 5, 8, 2) with parity = 10, even parity
n_{-1}=(4, 1, 3, 7, 0, 6, 5, 8, 2) with parity = 11, odd parity
Since n_1 and n_{-1} are of unequal parity, they are mutually unreachable.

2. A (Set of Arcs/Transitions):

A = {(n_i,n_j) | n_j can be obtained from n_i by a legal blank move}

Each arc corresponds to one legal move of the blank (0) in one of the four directions:

up ↑
right →
down ↓
left ←

Eg- (Refer Fig. 1)

A= {(n_1, n_2), (n_1, n_3), (n_1, n_4), (n_1, n_5), (n_2, n_6),...}

(n_1, n_2) → blank moved up,
(n_1, n_3) → blank moved left,
(n1, n_4) → blank moved down

3. S (Start State):

S= {n_i}

This specifies the initial configuration of the puzzle.

Eg- We consider start state as:

S = { (1, 4, 3, 7, 0, 6, 5, 8, 2) }

4. GD (Goal State):

GD = { n_g }

The goal state is the target configuration we want to reach.

Eg- We consider goal state as:

GD = {(1, 2, 3, 4, 5, 6, 7, 8, 0)}

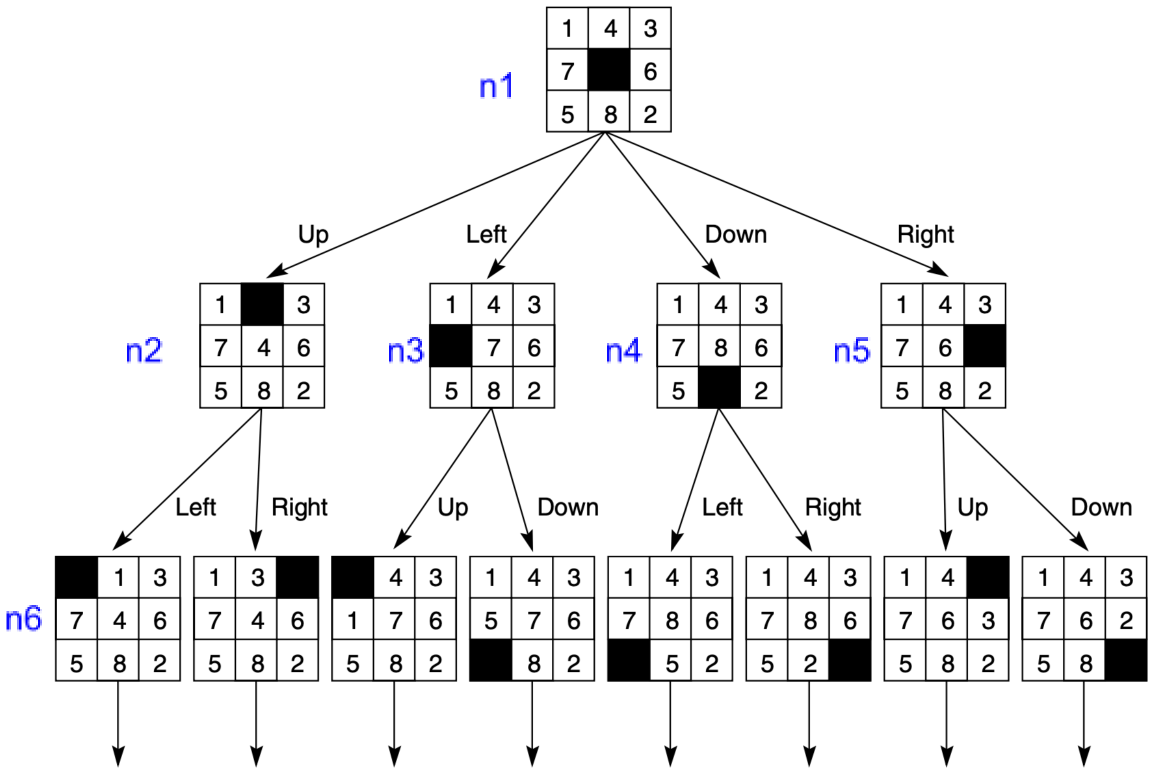

Fig 1 : State Space Graph representing few initial states

THANK YOU