

PROYECTO 1 SISTEMAS INTELIGENTES

PROBLEMA: 8-PUZZLE

Benjamín Ávila Rosas - A01361843

Task Environment Analysis

Performance	Environment	Actuators	Sensors
<ul style="list-style-type: none">• Movements number• Manhattan Distance• Misplaced Tiles• Execution Time	<ul style="list-style-type: none">• Tiles Number• Blank Position• Puzzle Size	<ul style="list-style-type: none">• Blank Tile• Numbered Tiles	<ul style="list-style-type: none">• Available movements for blank and tiles.• Misplaced Tiles

Environment Analysis

- Observable
- Deterministic
- Sequential
- Static
- Discrete
- Single Agent

Initial State

- 8-puzzle

0	1	3
5	2	6
4	7	8

- 15-puzzle

2	6	3	0
1	11	7	4
5	14	10	8
9	13	15	12

Successor Function

The N-Puzzle game defines certain rules for tiles movement, we will describe how tiles movement are allowed based on the blank space position on the puzzle. As the puzzle is composed of tiles on a matrix, we will represent the position of tiles with a $[x, y]$ pair, this means that the puzzle has N rows and N columns of $[x, y]$ pairs, each of these pairs can only be assigned to a simple tile or blank space, and we will assume that N must be equal or greater than 3.

Puzzle Guide (N = 3)

X axis advances to right and Y axis downwards

0,0	1,0	2,0
0,1	1,1	2,1
0,2	1,2	2,2

Allowed Movements

- **Corners**

If $[x == 0 \ \&\& \ y == 0]$ --> Blank space can only move to right and downwards

If $[x == 0 \ \&\& \ y == N - 1]$ --> Blank space can only move to right and upwards

If $[x == N - 1 \ \&\& \ y == 0]$ --> Blank space can only move to left and downwards

If $[x == N - 1 \ \&\& \ y == N - 1]$ --> Blank space can only move to left and upwards

- **Border (Not Cornered spaced)**

If $[x == 0]$ --> Blank space can move to right, upwards and downwards

If $[x == N - 1]$ --> Blank space can move to left, upwards and downwards

If $[y == 0]$ --> Blank space can move downwards, right, and left

If $[y == N - 1]$ --> Blank space can move to right, left and upwards

- **Anywhere else**

If $[x != 0 \ \&\& \ x != N - 1 \ \&\& \ y != 0 \ \&\& \ y != N - 1]$ --> Blank space can move to right, upwards and downwards

Every time the successor function is executed, it is required to validate if current state is equal to targeted one, in case it is still not correct, PSA might continue searching with other movements.

Target State

- **8-puzzle**

1	2	3
4	5	6
7	8	0

- 15-puzzle

1	2	3	4
5	6	7	8
9	10	11	12
13	14	15	0

Heuristic Function

The N-Puzzle can use different heuristic functions to use informed algorithms solutions. In this case we are using the Manhattan Distance Heuristic, that sums the distances between tiles and their targeted positions. Distance between a single tile and its target is formed by the sum of horizontal and vertical distances as tiles cannot move diagonally.

Cost Function

The implemented solution for the N-Puzzle will calculate its cost by the number of movements required to achieve the targeted state, as movements can only be made vertically and horizontally, every step required costs only 1 point.

Solution Repository URL

<https://github.com/benjaavro/n-puzzle>