Middle East Technical University | Department of Computer Engineering

CENG 580

Multi Agent Systems

Spring 2025

Homework 2

Due date: May 10 2025, Tuesday, 23:59

1 Problem Definition

In this assignment, you will explore the standard Multi-Agent Path Finding (MAPF) algorithm known as Conflict-Based Search (CBS), with a minor modification. In the standard MAPF problem, each agent must find a path from an initial location (node) to a designated destination (node). In the modified version, however, each agent is also required to return to its initial node after reaching the goal node. In other words, each agent must determine a circuit that starts and ends at its initial node, while visiting a specific goal node along the way. Note that CBS is both complete and optimal, and its use in solving the modified problem must preserve these properties.

2 I/O Format

Input Format: We will use MAPF Benchmarks:

- Benchmark information page: `https://movingai.com/benchmarks/mapf/index.html`
- General information page: `https://movingai.com/benchmarks/mapf.html`
- File formats: `https://movingai.com/benchmarks/formats.html`

Note that the input will be a scenario file, where each line specifies an agent's initial and final locations within a specific bucket. In this assignment, we only consider bucket number 0. We will consider only 2D grid world environments in which each cell is either empty (represented by . in the map file) or occupied by an agent or obstacle (represented by @). Agents can move in the four cardinal directions, as long as the neighboring cell is not occupied by an obstacle. Each movement has a unit cost of 1.

Sample Input: a scenario file such as `maze512-1-0.map.scen`
(Note that scenario file refers to a map file, namely `maze512-1-0.map` for the example scenario)

Output Format: output file name is `output.txt`

Each line must include the agent number, the path cost, and the path information, which is the sequence of locations to be visited, as illustrated below.

Sample Output:

```
Agent01,PathCost: 6,Path:(1,1)(1,2)(1,3)[1,4](1,3)(1,2)(1,1)
Agent02,PathCost: 5,Path:(1,2)(1,3)[2,3](1,3)(1,2)
Agent03,PathCost: 7,Path:(2,1)(1,1)(1,2)(1,2)[1,3](2,3)(2,2)(2,1)
.
.
.
```

Note that locations are listed as coordinates in parentheses (x, y), except for the goal location, which is shown in square brackets [x, y]. The sample output provided above is for formatting reference only.

3 Regulations

1. **Implementation:** Language: Pyhton
2. **Submission:** Submit your sources codes in a zipped file named in the format of `HW2-Name-Surname.zip` through the assignment activity on `OdtuClass`.