

Multi-Agent Path Finding

K-Robust Conflict-Based Search (kR-CBS)

Student: Wang Xiaoyu

Supervisor: Assoc Prof Tang Xueyan

Project Objectives:

1. To implement kR-CBS algorithm using Python.
2. To implement a graphical user interface (GUI) to visualise paths taken by agents.
3. To improve the capabilities and performance of the algorithm.

Problem definition:

Given a set of n agents, each having a **start position** and a **goal position**, find a set of n paths for the agents to move from their respective start positions to goal positions **without collision**. The agents may have **delays up to k time steps** during their movements. This plan should minimize a **cost function**.

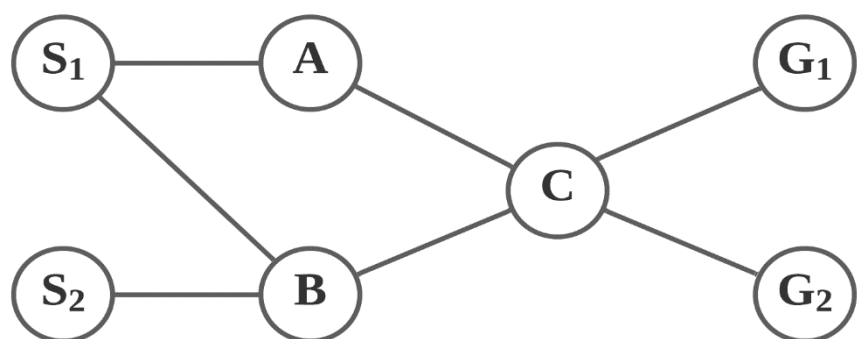


Fig.1 Example map

Implementation:

1. Given the map and agents in Fig.1, the CT root node generates a shortest path for each agent using A* algorithm.
2. In the solution in node 1, both agents are at position C at time 2. This conflict is denoted as $(2, a_1, 2, a_2, C)$.
3. Two child nodes are generated to resolve this conflict. Node 2 forbids a_1 from entering C at time 2 with a constraint $(2, a_1, C)$, while node 3 forbids a_2 from entering C at time 2.
4. In node 2, path for a_1 is updated with respect to the constraint. A k -delay conflict is found and denoted as $(2, a_2, 3, a_1, C)$.
5. New child nodes are generated to resolve the conflict, until a solution with no conflict is found.

Improvements:

1. Generalise the algorithm to include:
 - Node conflicts, k -delay conflict, edge conflicts
 - Cost functions: sum-of-costs, makespan, fuel
2. Improve the speed of the solver by:
 - Bypassing conflict (BP)
 - Prioritising conflict (PC)

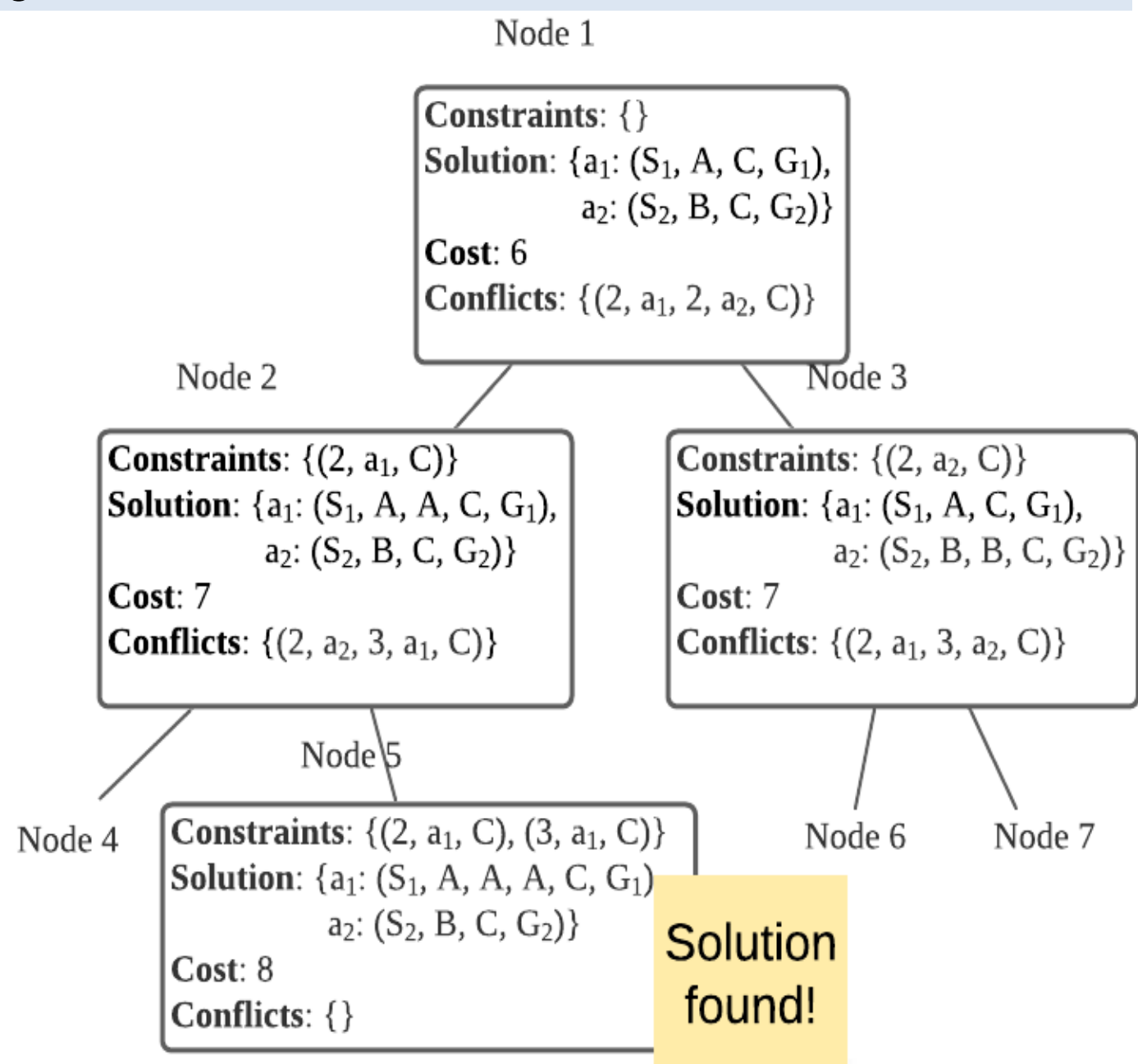


Fig.2 Example Constraint Tree (CT) with $k = 1$

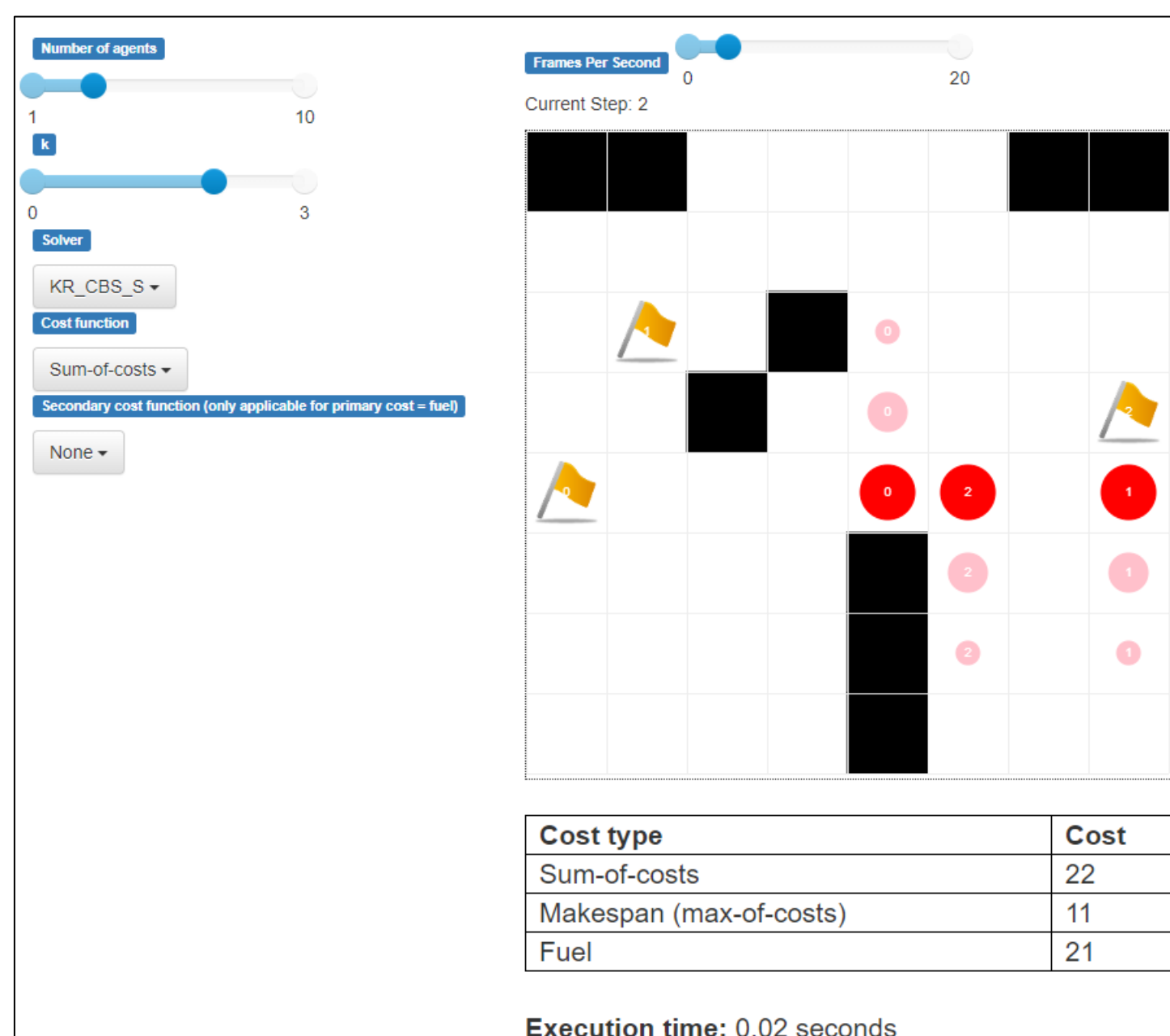


Fig.3 GUI to visualise the movements of agents