Multi-agent Path Finding

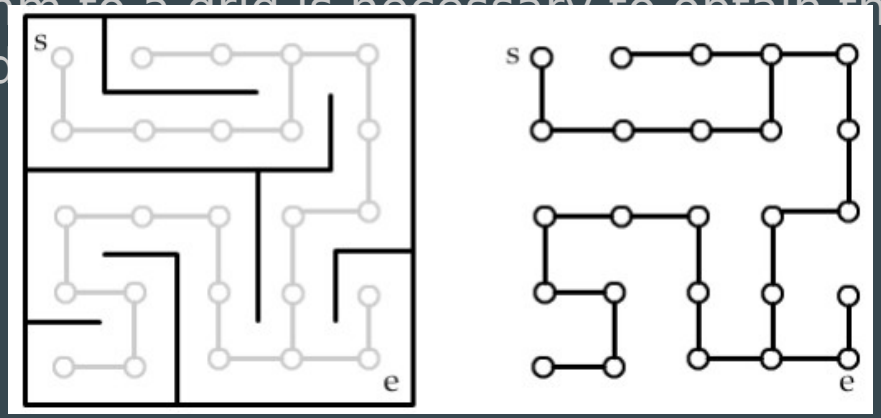
...

Gonzalo Piérola, Josu Campandegui and Óscar Aguinagalde

Path Finding

Is the plotting, by a computer application, of the shortest route between two points.

To apply a path finding algorithm to a grid is necessary to obtain the graph that represents that grid



http://qiao.github.io/PathFinding.js/visual/

MAPF

The pathfinding problem, where there is more than a one agent.

Some of its uses are in traffic control, aviation and video games.

There are 2 approaches for MAPF

- Decoupled approaches:

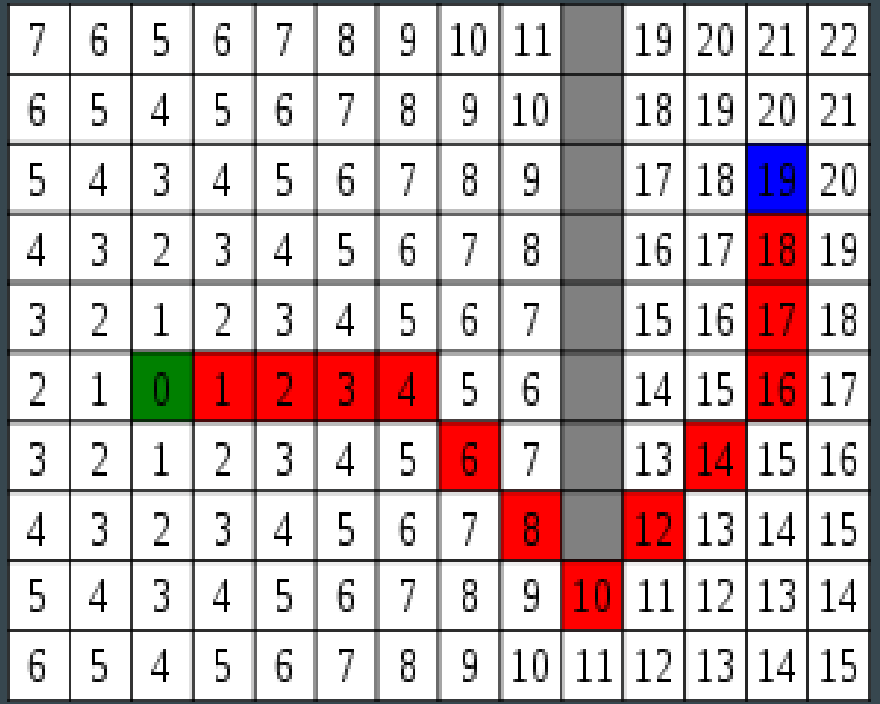
  Those in which paths for each agent is calculated separately.

- Coupled approaches:

  Those where the MAPF problem is formalized as a global, single-agent search problem, where paths are planned for all of the agents simultaneously.

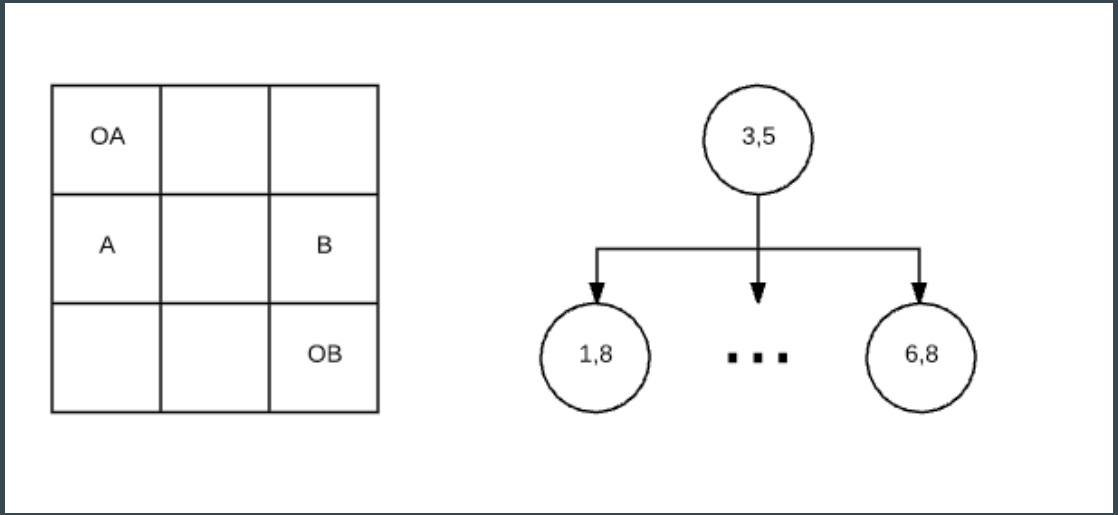
A*

- Finds the shortest path between the initial node and the objective
- It uses an heuristic to select the next node based on this formula: f(x) = g(x) + h(x)
- The heuristic have to be admissible in order to find the optimal solution.



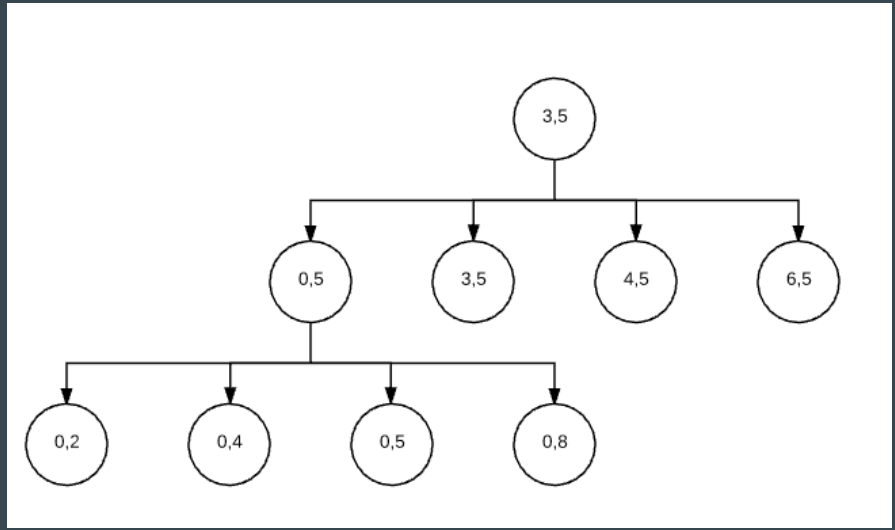
Multi Agent A*

To apply the A* in MAPF it is used the sum of the distance to the objective of all the agents.



A* + OD

- It is a modification of A* for reduce the number of expanded nodes.
- Introduces intermediate steps between each full step.
- Each step represent the movement of one of the agents.



ICTS

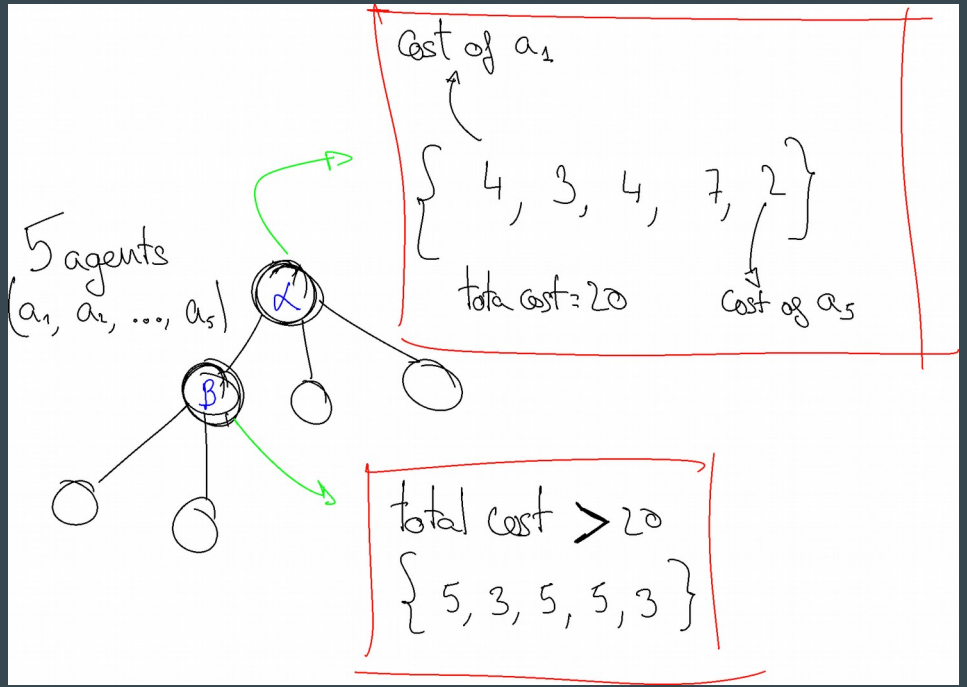
Increasing Cost Tree Search

This algorithm consists of 2 phases:

- **High-level phase:** Each node is a vector {C_1,C_2,C_3, ... , C_k} where each C_i is the cost of movement of a single agent A_i, from initial state to its goal, and k is the number of agents.

- The deeper in the tree, the higher the cost.

- **Low-level phase: Check if there is a valid solution for the selected node.**



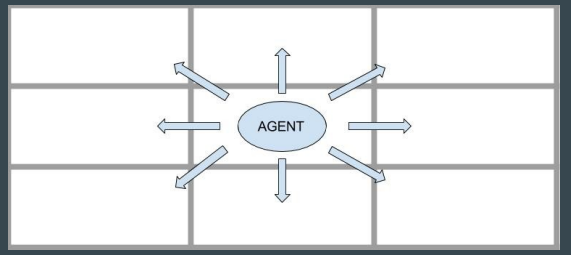
FAR (Flow Annotation Replanning)

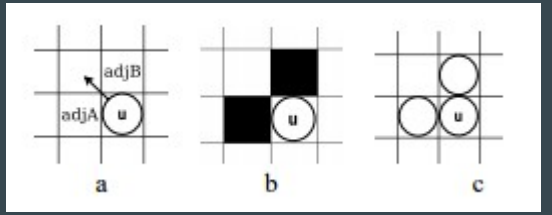
- Multi-agent planning in grid maps
- Uses A* for pathfinding
- Developed with speed and memory efficiency in mind (WHCA*)

- Experiments run for Baldur’s Gate maps
- By Ko-Hsin Cindy Wang and Adi Botea

Problem definition

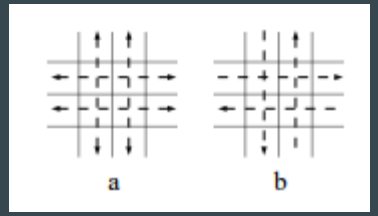
- Map divided in octiles
- Each tile accessible or blocked
- Each tile one agent
- Extra constraint for diagonal moves

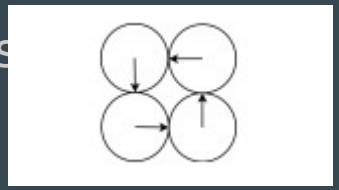




FAR method

1. Abstract grid map into a flow-annotated search graph
2. Avoid replanning if possible (A* modification and straight lines)
3. Provide communication:
    a. Agent make step reservations
    b. Control busy nodes
    c. Manage blocks
4. Identify and manage deadlocks
    a. When agents wait in cycle
    b. Locate critical agent





Conclusion

- Low CPU and memory usage
- Comparison with WHCA*
- Problems with single-width tunnel