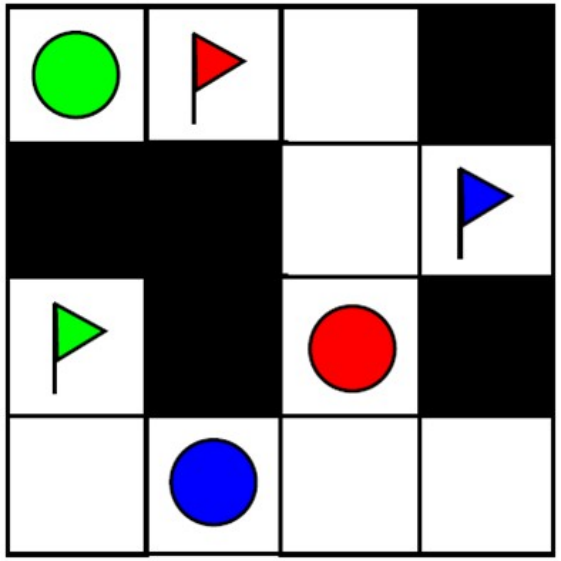
Pathfinding and Routing
NAIL137

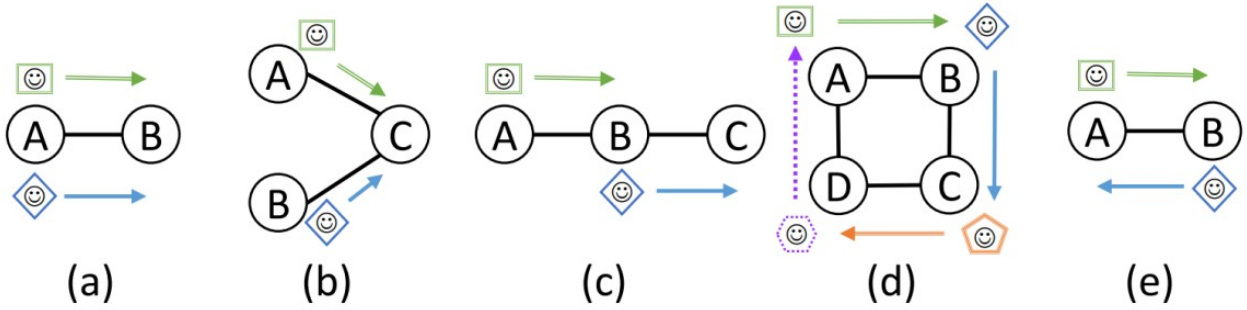
Jiří Švancara

FACULTY
OF MATHEMATICS
AND PHYSICS
Charles University

Multi-agent pathfinding (MAPF)

- Navigate a set of agents
- Agents can not collide
- A plan is a sequence of actions (including no-op)
- Finding optimal plan is NP-Hard





Multi-agent pathfinding (MAPF) - usages

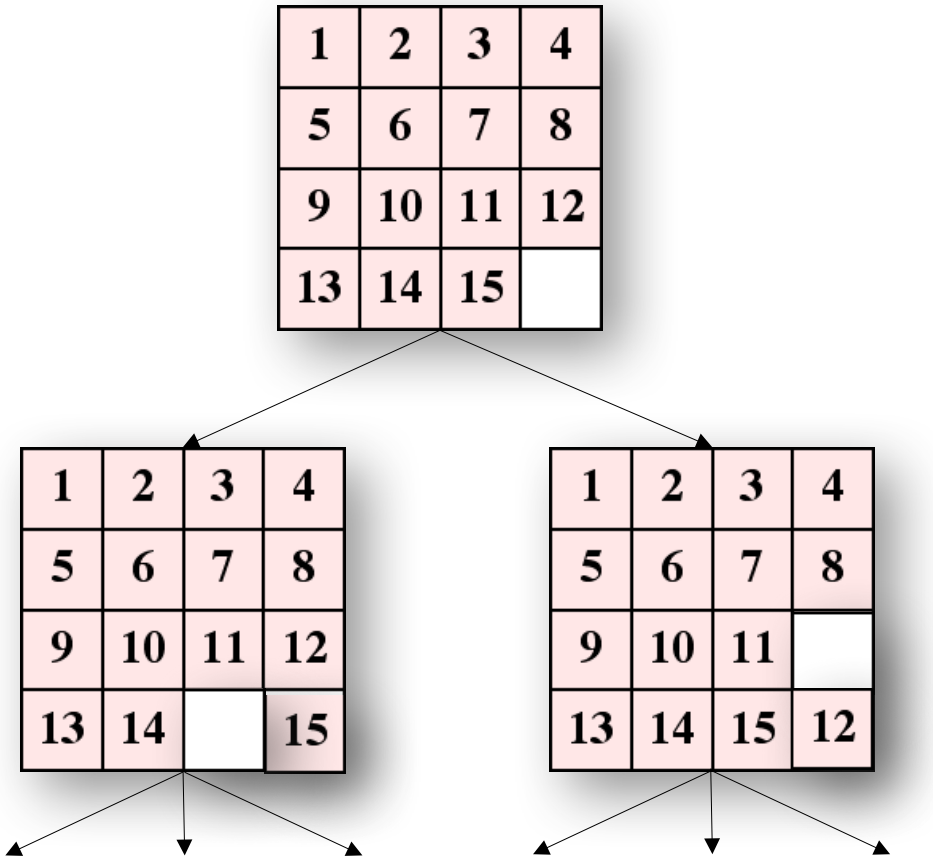
1	2	3	4
5	6	7	8
9	10	11	12
13	14	15	

Brute force solution

- Try all possible actions
- Repeat until solution is found

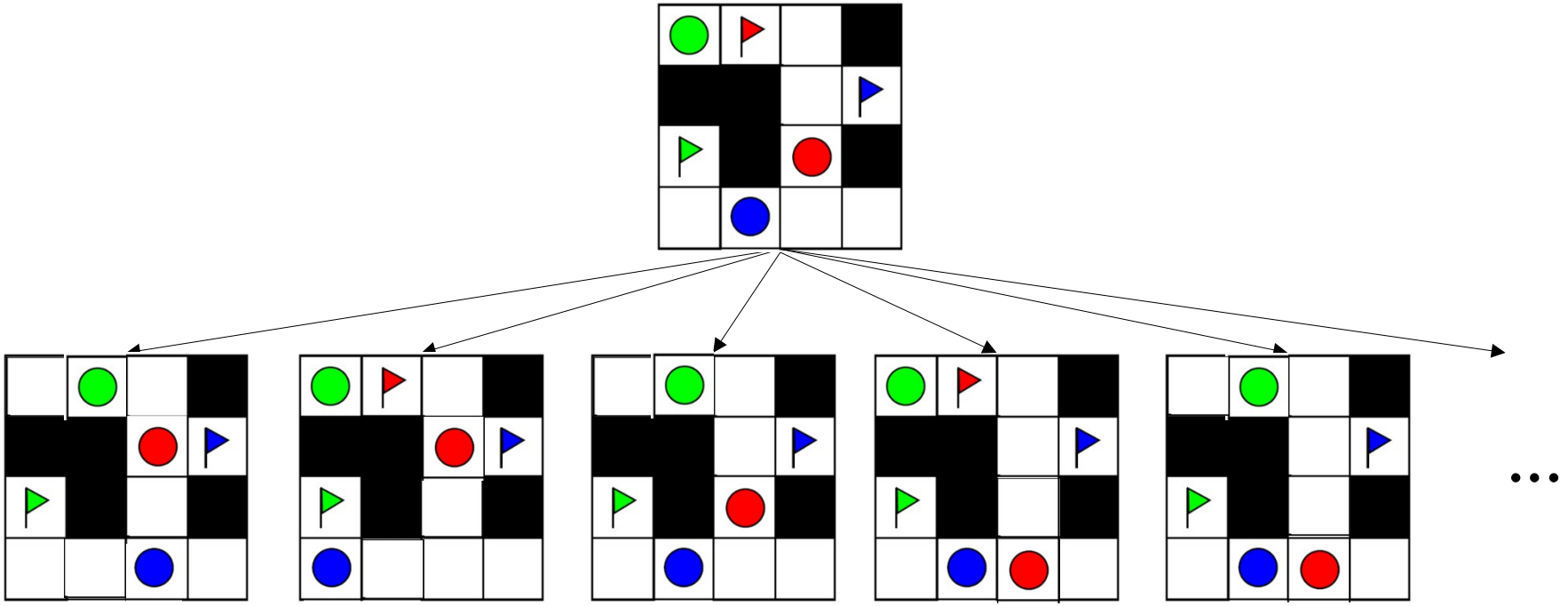
✓ Optimal

× Time complexity



Brute force solution

- Agents can usually move at once



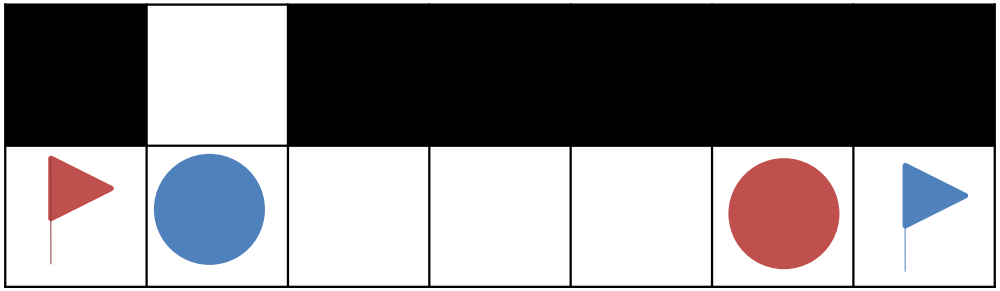
Single agent

- Single agent pathfinding is easy
  - BFS, Dijkstra, A*

Individual planning

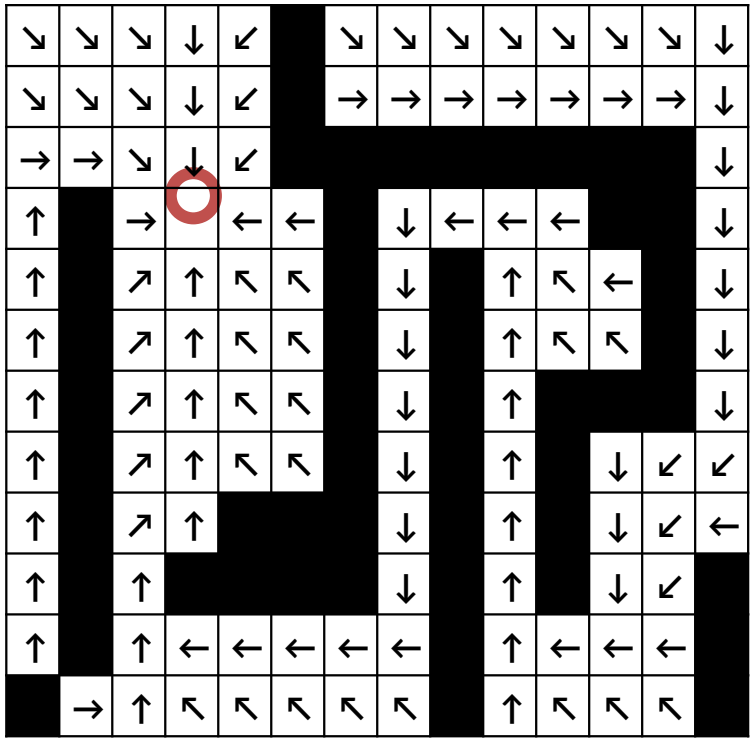
- Each agent is planned individually
  - Pretend other agents do not exist
  - In case of imminent collision, recalculate the path

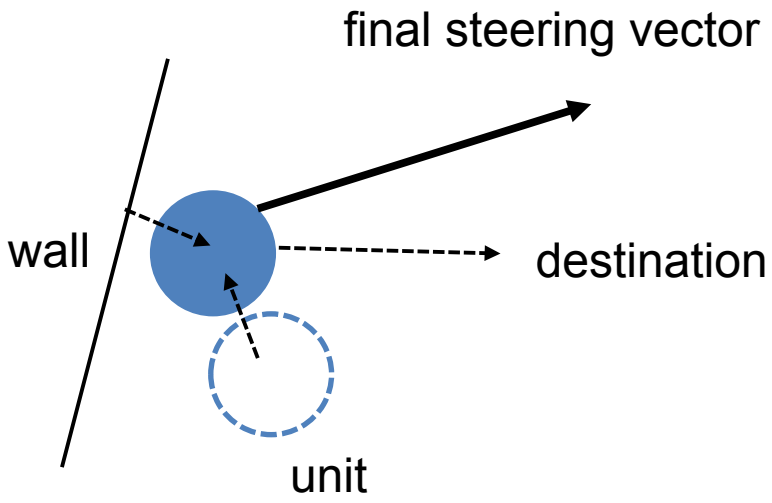
Individual planning



- Each agent is planned individually
  - Pretend other agents do not exist
  - In case of imminent collision, recalculate the path

“Physics” solution





Historic fun fact

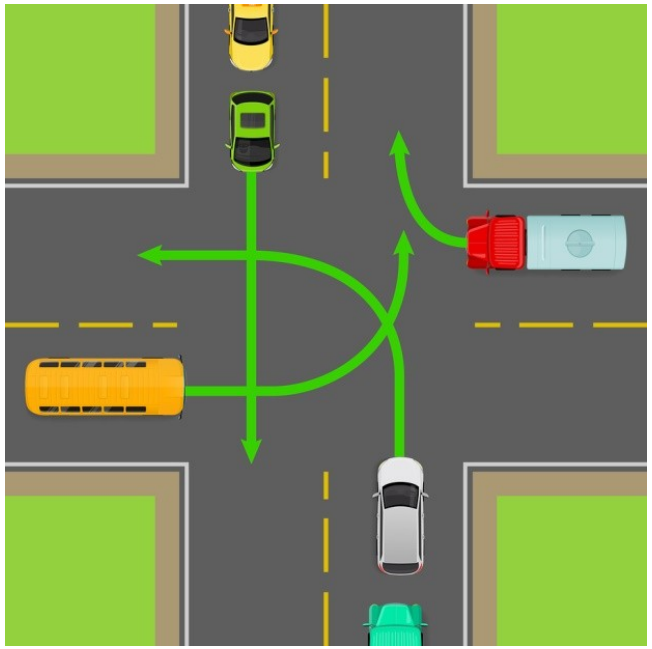
Prioritized planning

- Plan agents in order
  - 1st agent avoids no one
  - 2nd agent avoids 1st agent's plan
  - …
- New tasks are incoming

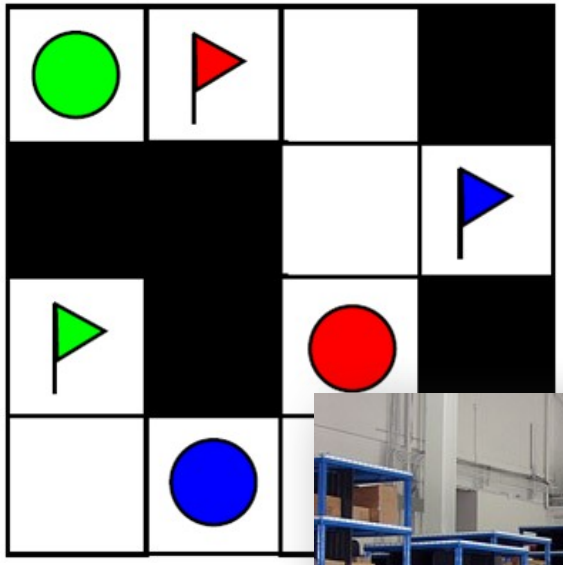
Trajectories

- Trajectories are known in advance
- Speed may be changed
  - Avoid collisions
  - Keep safe distance

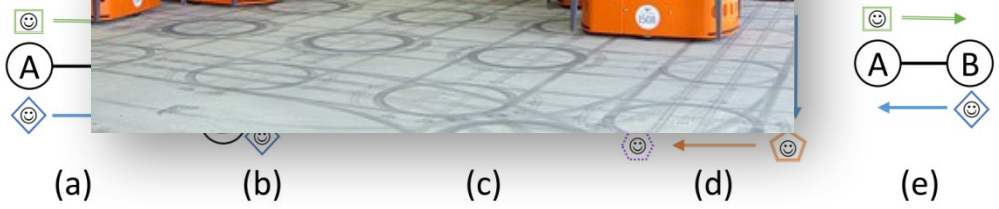
- First come, first served
- Can changing the order improve the solution?



Back to definition



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



(a) (b) (c) (d) (e)

Optimal solvers

Search (CBS)

- Find plan for single agents
- If conflict – add constraints

X Hard when many conflicts

✓ Easy on large open maps

Reduction (SAT, ASP)

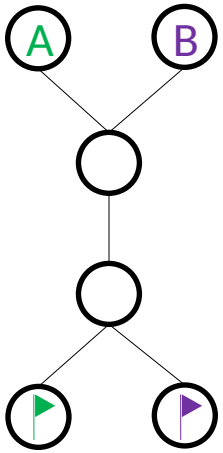
- Create variables
- Create logical formula

✓ Does not care much about conflicts

X Hard on large maps (too many variables)

Optimal solvers - CBS

Instance

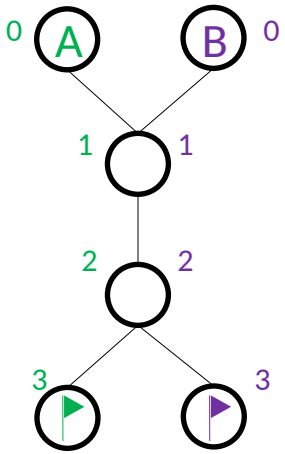


Constraint tree

No constraints

Optimal solvers - CBS

Instance

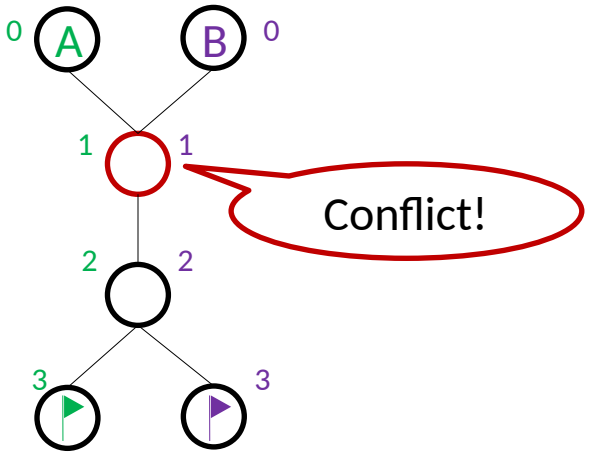


Constraint tree

No constraints

Optimal solvers - CBS

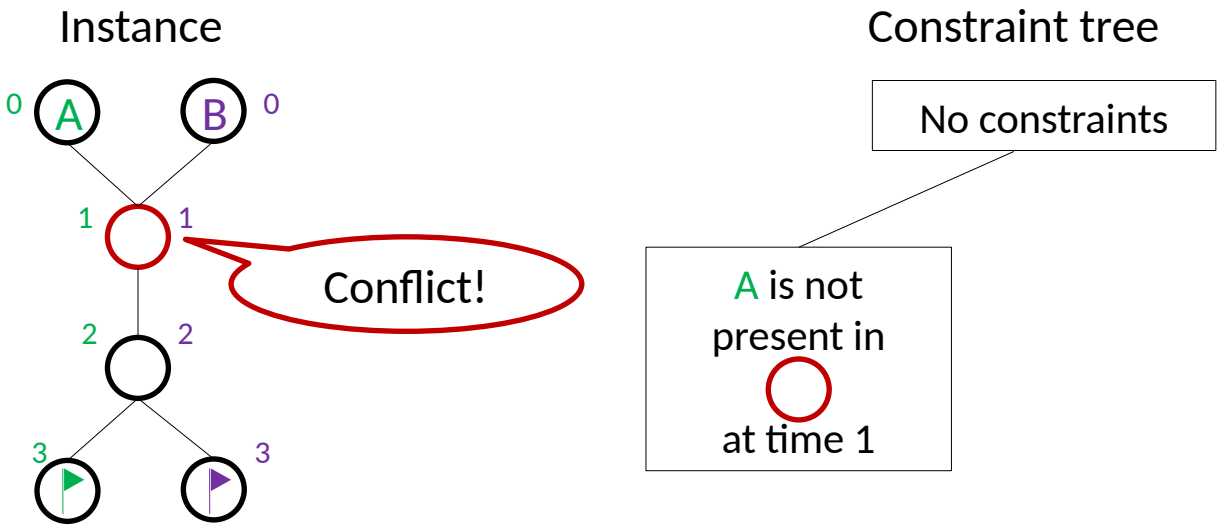
Instance



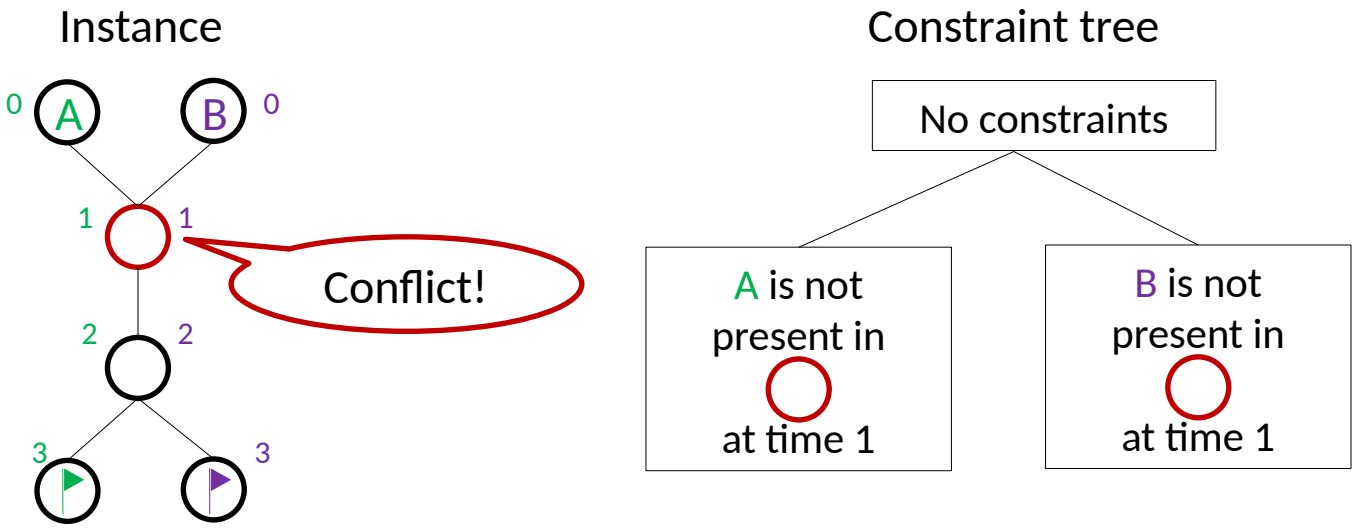
Constraint tree

No constraints

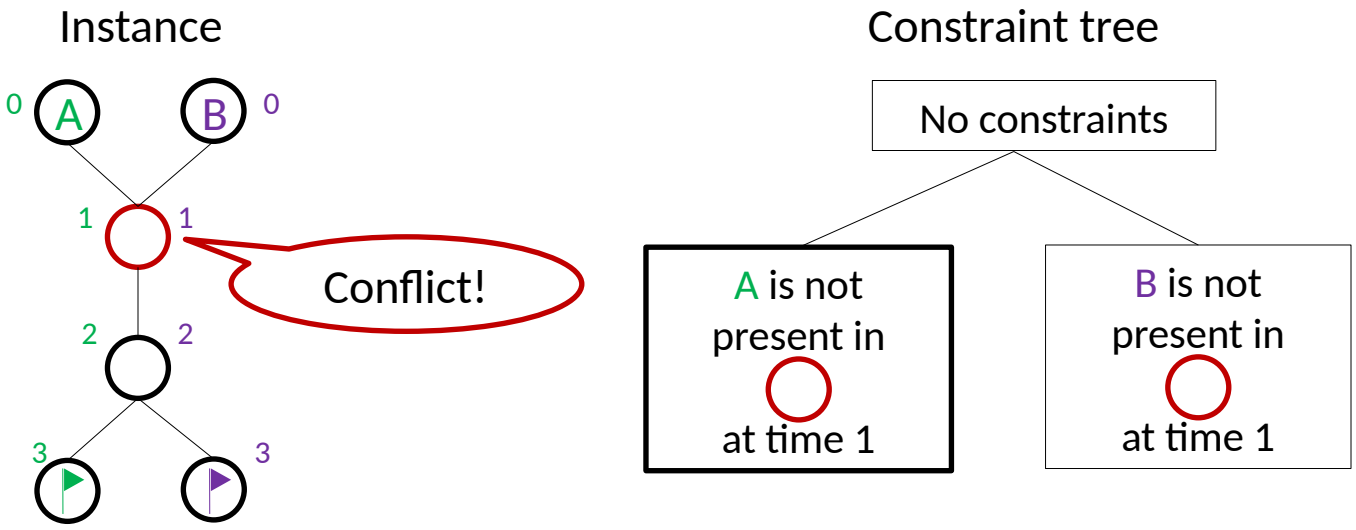
Optimal solvers - CBS



Optimal solvers - CBS

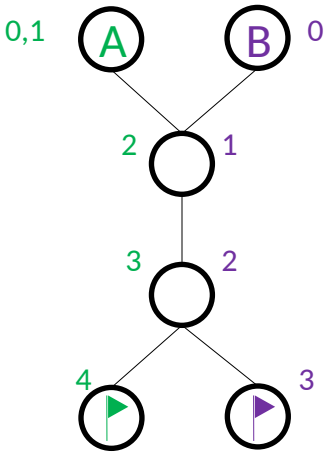


Optimal solvers - CBS

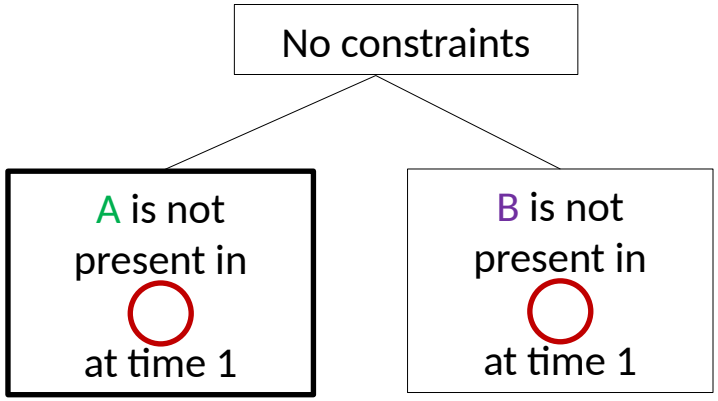


Optimal solvers - CBS

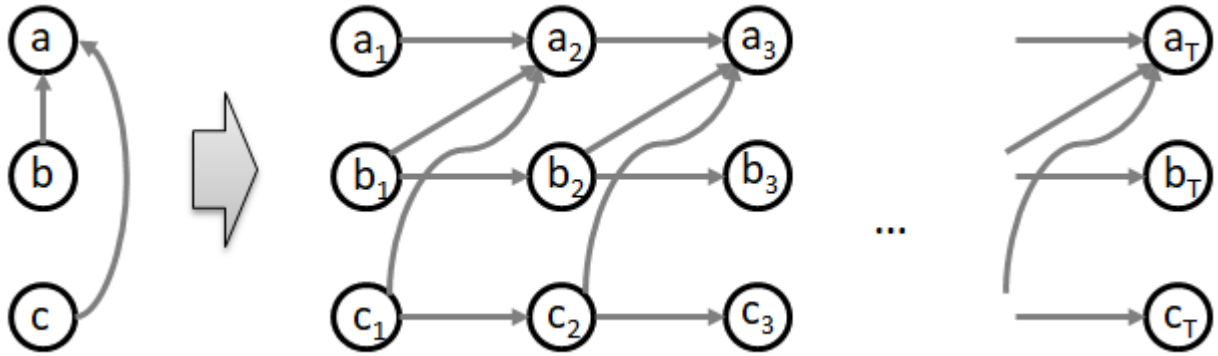
Instance



Constraint tree



Optimal solvers - reduction



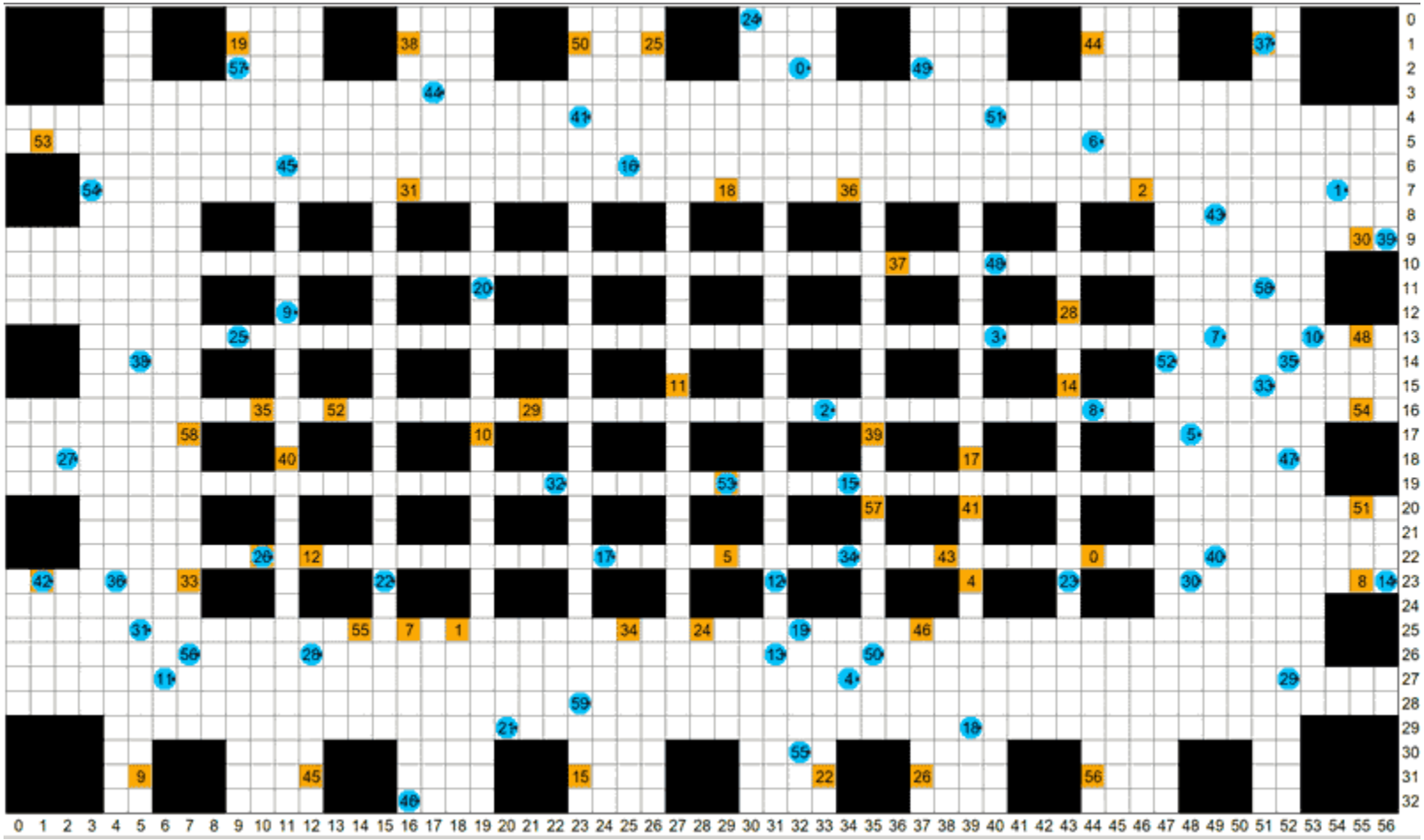
- Variable *At(v,i,t)* – agent a_i is at vertex *v* at timestep *t*
- Encode movement restrictions in SAT, ASP, ILP, …
- Iteratively add layers
- First feasible solution is optimal
- Do not include unreachable states

Optimality is hard!

- Divide and conquer
  - By agents (ID)
  - By area (DMAPF)
- Rule-based solvers
  - Grid maps (push and swap)
  - Bi-connected graphs (bibox)
- Windowed approach (lifelong MAPF)
- Prioritized planning (PP)
- LNS (MAPF-LNS)
- DFS style extension (PIBT, LaCAM)

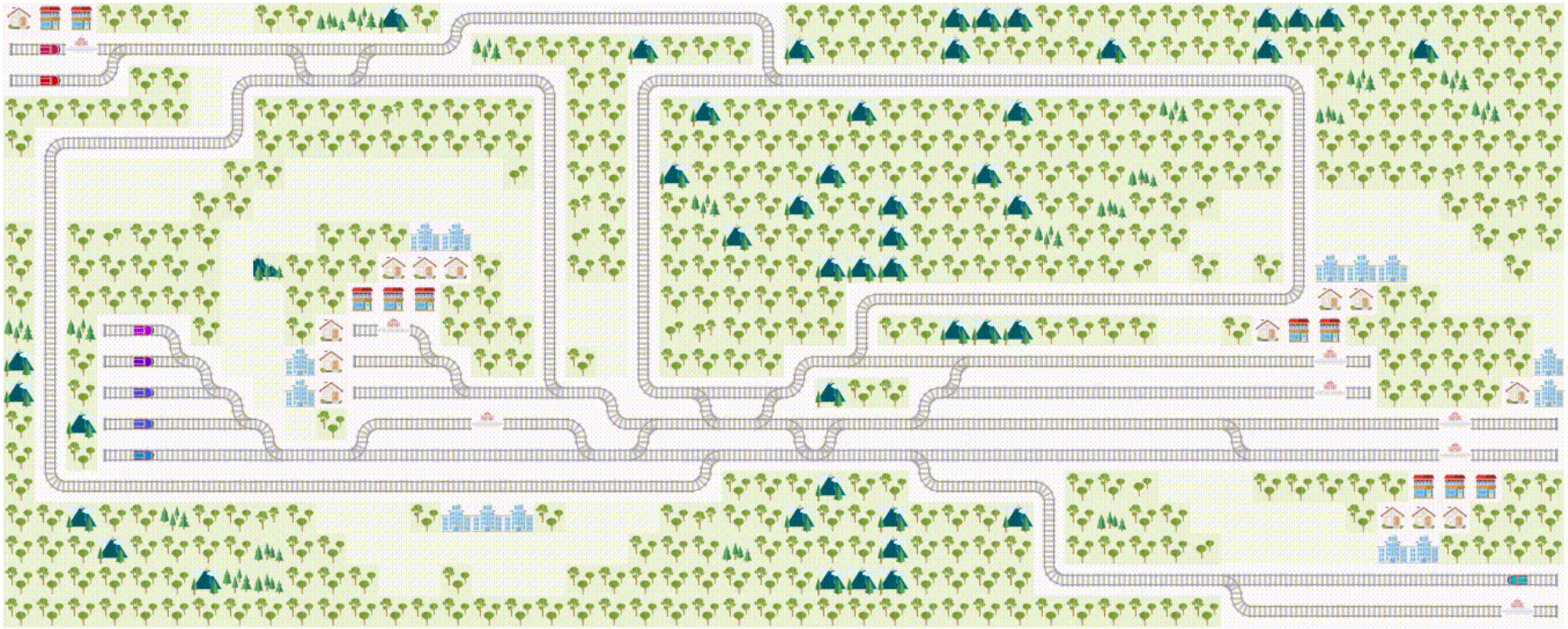
MAPF competition

https://www.leagueofrobotrunners.org/



MAPF competition

https://www.aicrowd.com/challenges/flatland



Outline

- Definitions, cost functions, single agent solvers, optimal solvers
- Sub-optimal solvers, rule-based solvers
- ML-based solvers
- Variants of MAPF
- Intersection management
- MAPF on robots, execution policies
- Demos and open problems