Map Abstraction for Multi-Agent Pathfinding problems with Answer Set Programming

Adrian Salewsky

University of Potsdam

00.00.2022

Table of Contents

Introduction

Introduction

- Reducing map size to increase speed

Introduction

- Reducing map size to increase speed
- Predetermined goal coordinates

Introduction

- Reducing map size to increase speed
- Predetermined goal coordinates
- Three methods to achieve goal

Introduction

- Reducing map size to increase speed
- Predetermined goal coordinates
- Three methods to achieve goal
- Asprilo as base

Shortest Path

Shortest Path

- Looking for shortest path of each robot

Shortest Path

- Looking for shortest path of each robot
- Conflicts between robots are ignored

Shortest Path

- Looking for shortest path of each robot
- Conflicts between robots are ignored
- Every node not visited gets deleted

Shortest Path

- Looking for shortest path of each robot
- Conflicts between robots are ignored
- Every node not visited gets deleted
- Remaining nodes are output

Node Combining

Node Combining

- Multiple nodes are combined

Node Combining

- Multiple nodes are combined
- Finding shortest path in smaller map

Node Combining

- Multiple nodes are combined
- Finding shortest path in smaller map
- Nodes that were visited in their combined form are kept

Node Combining

- Multiple nodes are combined
- Finding shortest path in smaller map
- Nodes that were visited in their combined form are kept
- Open Node Combining for open maps

Node Combining

- Multiple nodes are combined
- Finding shortest path in smaller map
- Nodes that were visited in their combined form are kept
- Open Node Combining for open maps
- Complete Node Combining for maps with walls

Reachable Nodes

Reachable Nodes

- Shortest path for each robot

Reachable Nodes

- Shortest path for each robot
- Calculating the amount of steps each node is deviating from the shortest path

Reachable Nodes

- Shortest path for each robot
- Calculating the amount of steps each node is deviating from the shortest path
- Maximum number of deviating steps is the individual makespan

Reachable Nodes

- Shortest path for each robot
- Calculating the amount of steps each node is deviating from the shortest path
- Maximum number of deviating steps is the individual makespan
- Output contains the information about the number of deviating steps

Auxiliary Programs

Auxiliary Programs

- Multiple python programs for easier use

Auxiliary Programs

- Multiple python programs for easier use
- Map generator for creating open maps

Auxiliary Programs

- Multiple python programs for easier use
- Map generator for creating open maps
- Multiple solvers using incrementation for horizon

Auxiliary Programs

- Multiple python programs for easier use
- Map generator for creating open maps
- Multiple solvers using incrementation for horizon
- Result plotters for analyzing benchmark results

Benchmarking

Benchmarking

- Shortest Path and Reachable Nodes are worse than asprilo

Benchmarking

- Shortest Path and Reachable Nodes are worse than asprilo
- Especially Reachable Nodes has performance issues

Benchmarking

- Shortest Path and Reachable Nodes are worse than asprilo
- Especially Reachable Nodes has performance issues
- Node Combining can beat asprilo in certain scenarios

Benchmarking

- Shortest Path and Reachable Nodes are worse than asprilo
- Especially Reachable Nodes has performance issues
- Node Combining can beat asprilo in certain scenarios
- Using the right size for Node Combining is important

Conclusion

Conclusion

- Goal: Achieving time improvement for MAPF-Problems

Conclusion

- Goal: Achieving time improvement for MAPF-Problems
- Method: Decreasing node count to have faster solving

Conclusion

- Goal: Achieving time improvement for MAPF-Problems
- Method: Decreasing node count to have faster solving
- Three different abstraction methods

Conclusion

- Goal: Achieving time improvement for MAPF-Problems
- Method: Decreasing node count to have faster solving
- Three different abstraction methods
- Node Combining shows promise