Exact Algorithms and Lowerbounds for Multiagent Pathfinding

Foivos Fioravantes, Dušan Knop, Jan Matyáš Křišťan, Nikolaos Melissinos, Michal Opler

1Department of Theoretical Computer Science, FIT, Czech Technical University in Prague, Czechia

JGA 2023
The problem:
- Each robot wants to reach its colour
- Move in parallel
- Centralised decisions
- Two versions: swap or not

Question:
What is the makespan = minimum number of rounds?
The problem:
- Each robot wants to reach its colour
- Move in parallel
- Centralised decisions
- Two versions: swap or not

Question:
What is the makespan = minimum number of rounds?
→ Here, at least 5 rounds for the blue robot
Multiagent Pathfinding - Swaps allowed

The problem:
- Each robot wants to reach its colour
- Move in parallel
- Centralised decisions
- Two versions: swap or not

Question:
What is the makespan = minimum number of rounds?
→ Here, at least 5 rounds for the blue robot
The problem:

- Each robot wants to reach its colour
- Move in parallel
- Centralised decisions
- Two versions: swap or not

Question:
What is the **makespan** = minimum number of rounds?

→ Here, at least 5 rounds for the **blue** robot
The problem:
- Each robot wants to reach its colour
- Move in parallel
- Centralised decisions
- Two versions: swap or not

Question:
What is the makespan = minimum number of rounds?
→ Here, at least 5 rounds for the blue robot
Multiagent Pathfinding - Swaps allowed

The problem:
- Each robot wants to reach its colour
- Move in parallel
- Centralised decisions

Two versions:
- swap or not

Question:
What is the makespan = minimum number of rounds?
→ Here, at least 5 rounds for the blue robot
The problem:
- Each robot wants to reach its colour
- Move in parallel
- Centralised decisions
- Two versions: swap or not

Question:
What is the makespan = minimum number of rounds?
→ Here, at least 5 rounds for the blue robot
The problem:
- Each robot wants to reach its colour
- Move in parallel
- Centralised decisions
- Two versions: swap or not

Question:
What is the makespan = minimum number of rounds?
→ Here, at least 5 rounds for the blue robot

Round 4

Makespan = 5
The problem:
- Each robot wants to reach its colour
- Move in parallel
- Centralised decisions
- **Two versions:** swap or not

**Question:**
What is the makespan = minimum number of rounds?
→ Here, at least 5 rounds for the blue robot

Round 5
Makespan = 5
The problem:
- Each robot wants to reach its colour
- Move in parallel
- Centralised decisions
- Two versions: swap or not

Question:
What is the makespan = minimum number of rounds?
→ Here, at least 5 rounds for the blue robot
Multiagent Pathfinding - Swaps not allowed

The problem:
- Each robot wants to reach its colour
- Move in parallel
- Centralised decisions

Two versions:
- swap or not

Question:
What is the makespan = minimum number of rounds?
→ Here, at least 5 rounds for the blue robot
Multiagent Pathfinding - Swaps not allowed

The problem:
- Each robot wants to reach its colour
- Move in parallel
- Centralised decisions
- Two versions: swap or not

Question:
What is the makespan = minimum number of rounds?
→ Here, at least 5 rounds for the blue robot
Multiagent Pathfinding - Swaps not allowed

The problem:
- Each robot wants to reach its colour
- Move in parallel
- Centralised decisions
- Two versions: swap or not

Question:
What is the makespan = minimum number of rounds?
→ Here, at least 5 rounds for the blue robot
The problem:
- Each robot wants to reach its colour
- Move in parallel
- Centralised decisions
- Two versions: swap or not

Question:
What is the makespan = minimum number of rounds?
→ Here, at least 5 rounds for the blue robot
Multiagent Pathfinding - Swaps not allowed

The problem:
- Each robot wants to reach its colour
- Move in parallel
- Centralised decisions
- **Two versions:** swap or not

Question:
What is the **makespan** = minimum number of rounds?
→ Here, at least 5 rounds for the **blue** robot
The problem:
- Each robot wants to reach its colour
- Move in parallel
- Centralised decisions
- Two versions: swap or not

Question:
What is the makespan = minimum number of rounds?
→ Here, at least 5 rounds for the blue robot
Multiagent Pathfinding - Swaps **not** allowed

The problem:
- Each robot wants to reach its colour
- Move in parallel
- Centralised decisions
- **Two versions:** swap or not

**Question:**
What is the makespan = minimum number of rounds?

→ Here, at least 5 rounds for the **blue** robot

Makespan ≤ 7
### What is already known?

#### The problem is hard
- NP-complete (2010, Surynek),
- even on planar graphs (2019, Yu)

#### Heuristics (2019, Stern)
- SAT-based (2017, Surynek et al.)
- Scheduling (2018, Barták, Švancara, Vlk)

#### Theorem (2023, Eiben, Ganian, Kanj)
When allowing swaps, deciding if makespan $\leq 26$ remains NP-complete even when $G$ is planar and $\Delta(G) = 4$.

#### Similar problems
- Same problem but **sequential** moves of the robots (1984, Kornhauser, Miller, Spirakis)
- One robot on each vertex: Token swapping (2022, Aichholzer et al.), (2018, Bonnet, Miltzow, Rzazewski)
What we did

Studied parameterised complexity of the problem:

- vertex cover
- tree + ∆
- makespan
- # agents
- Infeasibility ↑
- Feasibility ↓

Theorem

When (not resp.) allowing swaps, deciding if makespan ≤ 3 (≤ 2 resp.) remains NP-complete even when $G$ is planar and ∆($G$) = 4 (∆($G$) = 5).
What we did

Studied parameterised complexity of the problem:

When (not resp.) allowing swaps, deciding if makespan $\leq 3$ ($\leq 2$ resp.) remains $\text{NP}$-complete even when $G$ is planar and $\Delta(G) = 4$ ($\Delta(G) = 5$).
Main tool for polynomial algorithms
Time-expanded graph

Time expanded graph with 3 layers: $G_T(3)$
Time-expanded graph

Time expanded graph with 3 layers: $G_T(3)$

Ingredients:

1. Given $G$, starting and ending positions $s_i, t_i$, $1 \leq i \leq k$, the makespan is $\ell$ iff there exist $k$ vertex-disjoint paths from the $s_i$'s to the $t_i$'s in $G_T(\ell)$.


3. FPT algorithm for $k$ vertex-disjoint paths parameterised by $k + \ell$ (2011, Golovach and Thilikos).

Careful: $G_T(\ell)$ has treewidth bounded by $tw(G) + \ell$. 

F. Fioravantes
Multiagent Pathfinding
JGA 2023 7 / 11
Time-expanded graph

Time expanded graph with 3 layers: $G_T(3)$
Time-expanded graph

Ingredients:
1. Given $G$, starting and ending positions $s_i, t_i$, $1 \leq i \leq k$, the makespan is $\ell$ iff there exist $k$ vertex-disjoint paths from the $s_i$'s to the $t_i$'s in $G_T(\ell)$.
3. FPT algorithm for $k$ vertex-disjoint paths parameterised by $k + \ell$ (2011, Golovach and Thilikos).

Careful: $G_T(\ell)$ has treewidth bounded by $tw(G) + \ell$. 

Time expanded graph with 3 layers: $G_T(3)$
Time-expanded graph

Given $G$, starting and ending positions $s_i, t_i$, $1 \leq i \leq k$, the makespan is $\ell$ iff there exist $k$ vertex-disjoint paths from the $s_i$'s to the $t_i$'s in $G_T(\ell)$.


FPT algorithm for $k$ vertex-disjoint paths parameterised by $k + \ell$ (2011, Golovach and Thilikos).

Careful: $G_T(\ell)$ has treewidth bounded by $tw(G) + \ell$. 
Ingredients:

1. Given $G$, starting and ending positions $s_i, t_i, 1 \leq i \leq k$, the makespan is $\ell$ iff there exist $k$ VERTEX-DISJOINT PATHS from the $s_i$’s to the $t_i$’s in $G_T(\ell)$. 
Time-expanded graph

Ingredients:

1. Given $G$, starting and ending positions $s_i, t_i, 1 \leq i \leq k$, the makespan is $\ell$ iff there exist $k$ VERTEX-DISJOINT PATHS from the $s_i$’s to the $t_i$’s in $G_T(\ell)$.

2. FPT algorithm for $k$ VERTEX-DISJOINT PATHS parameterised by treewidth (1994, Scheffler).

3. FPT algorithm for $k$ VERTEX-DISJOINT PATHS parameterised by $k + \ell$ (2011, Golovach and Thilikos).
Time-expanded graph

Ingredients:

1. Given $G$, starting and ending positions $s_i, t_i, 1 \leq i \leq k$, the makespan is $\ell$ iff there exist $k$ VERTEX-DISJOINT PATHS from the $s_i$’s to the $t_i$’s in $G_T(\ell)$.

2. FPT algorithm for $k$ VERTEX-DISJOINT PATHS parameterised by treewidth (1994, Scheffler).

3. FPT algorithm for $k$ VERTEX-DISJOINT PATHS parameterised by $k + \ell$ (2011, Golovach and Thilikos).

Careful: $G_T(\ell)$ has treewidth bounded by $tw(G) + \ell$. 
NP-hardness for trees
Multiagent Pathfinding on Trees

**Theorem**

When not allowing swaps, it is NP-hard to compute the makespan of $T$, even when $T$ is a tree with $\Delta(T) = 5$.

Reduction from **Token Swapping**:

- same problem as ours, one robot on each vertex
- swaps allowed
- NP-hard for trees (2022, Aichholzer et al.)
Multiagent Pathfinding is hard on Trees

Main idea on graphs of treewidth 2:
Multiagent Pathfinding is hard on Trees

Main idea on graphs of treewidth 2:
Multiagent Pathfinding is hard on Trees

Main idea on graphs of treewidth 2:
Multiagent Pathfinding is hard on Trees

Main idea on graphs of treewidth 2:
Multiagent Pathfinding is hard on Trees

Main idea on graphs of treewidth 2:
Multiagent Pathfinding is hard on Trees

Main idea on graphs of treewidth 2:
Multiagent Pathfinding is hard on Trees

Main idea on graphs of treewidth 2:
Multiagent Pathfinding is hard on Trees

Main idea on graphs of treewidth 2:
Multiagent Pathfinding is hard on Trees

Main idea on graphs of treewidth 2:
Multiagent Pathfinding is hard on Trees

To go to trees:

- Replace rhombuses by complete binary trees of height $\lceil \log(\Delta) + 1 \rceil$
- Carefully adjust the lengths and agents of the extra paths
Conclusion
Conclusion

Infeasibility ↑
Feasibility ↓

# agents + vertex cover
vertex cover

# agents + makespan

# agents + diameter

# agents + ∆

treewidth

cliquewidth

# agents

treewidth

cliquewidth + makespan

# agents + makespan

Tree + ∆
Conclusion

cliquewidth

# agents + vertex cover

vertex cover

tree + ∆

treewidth

# agents + makespan

# agents + diameter

Infeasibility ↑

Feasibility ↓

# agents + vertex cover

Merci!