Subgoal Graphs for Fast Optimal Pathfinding

Andreas Thüring

Seminar: Search & Optimization
Universität Basel
October 22, 2015
Introduction

Simple Subgoal Graphs
Two-Level Subgoal Graphs
Conclusion
References

General Idea
Setting
Octile Distance
General Idea

- Don’t search on whole grid
- Use subgoal graph:
  - smaller
  - Nodes = Certain cells which are “subgoals”
  - Edges in the subgoal graph connect nodes which are visible to each other
  - preserve optimality
Introduction

Setting

- 8-neighbor grid
Introduction

Setting

- 8-neighbor grid
- blocked cells
Introduction

Setting

- 8-neighbor grid
- blocked cells
- actor:
  - occupies cell
  - may move in diagonal or cardinal direction if unblocked
Octile Distance

- Used as a heuristic for distance between two cells $h(s, s')$.
- $h(s, s') = \text{length of shortest path between } s \text{ and } s', \text{disregarding obstacles}$
Octile Distance

- Used as a heuristic for distance between two cells $h(s, s')$.
- $h(s, s') = \text{length of shortest path between } s \text{ and } s', \text{ disregarding obstacles}$
- Here: $h(s, s') = 2\sqrt{2} + 1$
Simple Subgoal Graphs

Introduction

Simple Subgoal Graphs
Two-Level Subgoal Graphs
Conclusion
References

Necessary definitions
Construction of a Simple Subgoal Graph
Searching on a simple subgoal graph
Getting All direct-h-reachable Subgoals of a Given Cell
**Subgoals**

Subgoals are defined as the cells at corners of obstacles.

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td><strong>•</strong></td>
</tr>
<tr>
<td>2</td>
<td></td>
<td></td>
<td><strong>•</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td><strong>•</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td></td>
<td><strong>•</strong></td>
<td></td>
<td><strong>•</strong></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td></td>
<td></td>
<td></td>
<td><strong>•</strong></td>
<td></td>
</tr>
</tbody>
</table>
Let \( s \) and \( s' \) be two cells. Cells \( s \) and \( s' \) are:

- **\( h\)-reachable** if there is a shortest path between them.
- **direct-\( h\)-reachable** if none of the shortest paths between them move over a subgoal.
Construction of a Simple Subgoal Graph

Given a grid:

1. Find subgoals
2. Connect direct-h-reachable subgoals

Grid setting taken from Uras and Koenig [2015]
Construction of a Simple Subgoal Graph

Given a grid:

1. Find subgoals
2. Connect direct-h-reachable subgoals

Grid setting taken from Uras and Koenig [2015]
Searching on a simple subgoal graph

Given a start cell \( s \) and a goal cell \( g \):

1. Connect \( s \) and \( g \) to simple subgoal graph via direct-h-reachable subgoals
2. Search high-level path on simple subgoal graph with \( A^* \)
3. Search segments of high-level path for with \( A^* \) to find low-level path on grid

Grid setting taken from Uras and Koenig [2015]
Searching on a simple subgoal graph

Given a start cell $s$ and a goal cell $g$:

1. Connect $s$ and $g$ to simple subgoal graph via direct-h-reachable subgoals
2. Search high-level path on simple subgoal graph with $A^*$
3. Search segments of high-level path for with $A^*$ to find low-level path on grid

Grid setting taken from Uras and Koenig [2015]
Searching on a simple subgoal graph

Given a start cell \(s\) and a goal cell \(g\):

1. Connect \(s\) and \(g\) to simple subgoal graph via direct-h-reachable subgoals
2. Search high-level path on simple subgoal graph with \(A^*\)
3. Search segments of high-level path for with \(A^*\) to find low-level path on grid

Grid setting taken from Uras and Koenig [2015]
To construct simple subgoal graph, and to connect start and goal cells to simple subgoal graph:

- Need to find all direct-h-reachable subgoals in vicinity of given cell

Speed up process?
Clearance value

Use clearance values!

Given cell $s$ and cardinal or diagonal direction $d$:

- $\text{Clearance}(s, d) =$ how many moves can we take in direction $d$
  
  i. before we reach an obstacle, or
  
  ii. until we reach a subgoal
Getting all direct-h-reachable subgoals of a given cell

- Precompute clearance values of every cell on the grid in all diagonal and cardinal directions

Then, given cell $s$:

1. Partition space around $s$ in octants.
2. For each octant, sweep horizontal lines beginning from diagonal lines.

Example taken from Uras et al. [2013]
Two-Level Subgoal Graphs
Idea

Decrease search space with further abstraction:
- Partition subgoals into local and global subgoals, prune local subgoals from graph
- Perform high-level search only on global subgoals
Local and global subgoals

Given subgoal $s$ and subgoals $s'$ and $s''$ to which it is connected via an edge.

$s$ is a local subgoal if:

i. there exists path from $s'$ to $s''$ through only global subgoals and not through $s$.
   
   ■ May not be longer than the original path through $s$!

or

ii. $s'$ and $s''$ are h-reaching
Construction of a Two-Level Subgoal Graph

Subgoal A2:
- Neighbors D2 and A4
  - i √
  - ii ×

A2 is local!

Grid setting from Uras and Koenig [2015]
Construction of a Two-Level Subgoal Graph

Subgoal A4:
- Neighbors A2 and D4
  - i. √
  - ii. ×

A4 is local!

Grid setting from Uras and Koenig [2015]
Construction of a Two-Level Subgoal Graph

Subgoal $D2$:

- Neighbors $A2$ and $D4$
  - $i \times$
  - $ii \times$

$D2$ is global!

Grid setting from Uras and Koenig [2015]
Construction of a Two-Level Subgoal Graph

Subgoal $D_4$:

- Neighbors $D_2$ and $A_4$
  - i $\times$
  - ii $\times$

- Neighbors $A_4$ and $F_6$
  - i $\times$
  - ii $\checkmark$

- Neighbors $D_2$ and $F_6$
  - i $\times$
  - ii $\checkmark$

$D_4$ is global!

Grid setting from Uras and Koenig [2015]
Construction of a Two-Level Subgoal Graph

Subgoal F6:
- Neighbors D4 and H6
  - i ×
  - ii √

Add edge between D4 and H6, F6 is then local

Grid setting from Uras and Koenig [2015]
Construction of a Two-Level Subgoal Graph

Subgoal $H_6$:
- Neighbors $D_4$ and $F_6$
  - $i$ ✓
  - $ii$ ✓

$H_6$ is local!

Grid setting from Uras and Koenig [2015]
Construction of a Two-Level Subgoal Graph

Grid setting from Uras and Koenig [2015]
Searching in a Two-Level Subgoal Graph

1. Connect $s$ and $g$ to global subgoals.
   - direct-h-reachable!
   - may need to make some local subgoals global, temporarily

2. search for high-level path on global subgoals

3. search for actual path on grid between each segment

Grid setting from Uras and Koenig [2015]
Searching in a Two-Level Subgoal Graph

1. Connect $s$ and $g$ to global subgoals.
   - direct-h-reachable!
   - may need to make some local subgoals global, temporarily

2. search for high-level path on global subgoals

3. search for actual path on grid between each segment

Grid setting from Uras and Koenig [2015]
Searching in a Two-Level Subgoal Graph

1. Connect s and g to global subgoals.
   - direct-h-reachable!
   - may need to make some local subgoals global, temporarily

2. search for high-level path on global subgoals

3. search for actual path on grid between each segment

Grid setting from Uras and Koenig [2015]
Two-Level subgoal graphs may drastically reduce search space!

- Especially on bigger maps

But:

- More complex search between subgoals

Figure: Single Subgoal Graph [Uras et al., 2013]
Further remarks

Two-Level subgoal graphs may drastically reduce search space!
- Especially on bigger maps
- More complex search between subgoals

Figure: Two-Level Subgoal Graph [Uras et al., 2013]
Conclusion
Conclusion

Simple Subgoal graphs

- preprocessing strategy for 8-neighbor grid problems
- Reduce search space
  - search nodes = corners of obstacles, “subgoals”
  - edges between direct-h-reachable subgoals

Two-Level subgoal graphs

- Further abstraction to reduce search space
- Prune graph and retain only global nodes: h-reachable!
- Strong performance (Grid-Based Path Planning Competition 2012 and 2013 “nondominated”)

Andreas Thüring
Subgoal Graphs for Fast Optimal Pathfinding
Thank you for your attention!
References
