Fast Obstacle k-Nearest Neighbour Query on Navigation Mesh

Final Presentation

Shizhe Zhao (27505928)
Supervisor: David Taniar, Daniel Harabor

Summary

1 Introduction

2 New Framework

Outline

1 Introduction

2 New Framework

Introduction

A Introduction

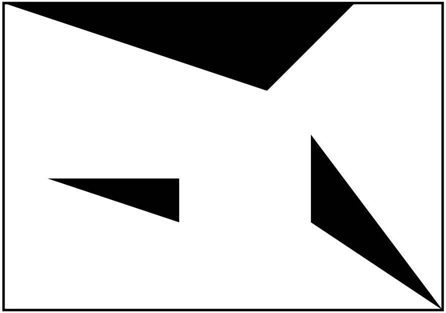
Outline

1 Introduction

2 New Framework

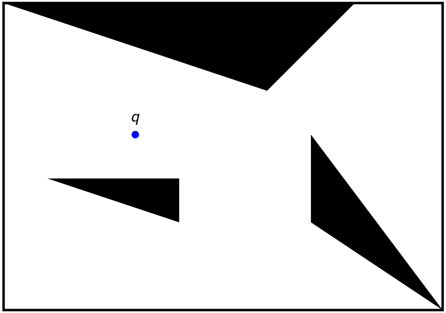
Polyanya: Overview

- black polygons are obstacles
- q: query point
- t: target
- gray border convex polygons are meshes
- mesh: all inside points are visible
- mesh: guides pathfinding



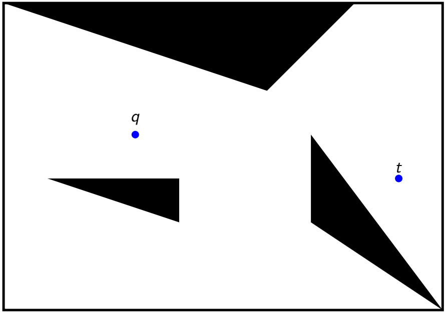
Polyanya: Overview

- black polygons are obstacles
- q: query point
- t: target
- gray border convex polygons are meshes
- mesh: all inside points are visible
- mesh: guides pathfinding



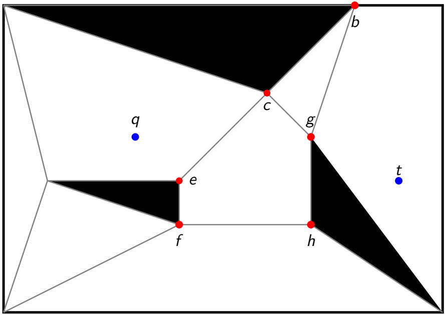
Polyanya: Overview

- black polygons are obstacles
- q: query point
- t: target
- gray border convex polygons are meshes
- mesh: all inside points are visible
- mesh: guides pathfinding



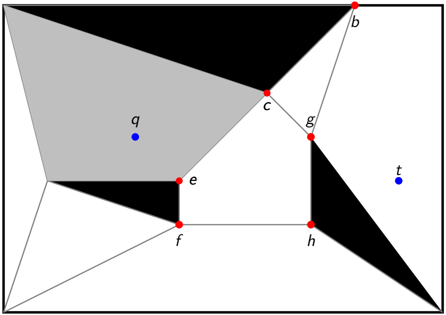
Polyanya: Overview

- black polygons are obstacles
- q: query point
- t: target
- gray border convex polygons are meshes
- mesh: all inside points are visible
- mesh: guides pathfinding



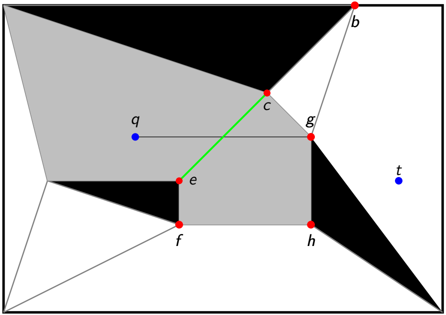
Polyanya: Overview

- black polygons are obstacles
- q: query point
- t: target
- gray border convex polygons are meshes
- mesh: all inside points are visible
- mesh: guides pathfinding



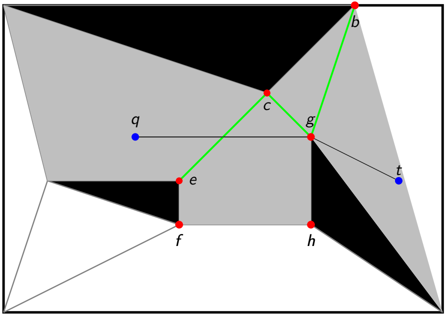
Polyanya: Overview

- black polygons are obstacles
- q: query point
- t: target
- gray border convex polygons are meshes
- mesh: all inside points are visible
- mesh: guides pathfinding



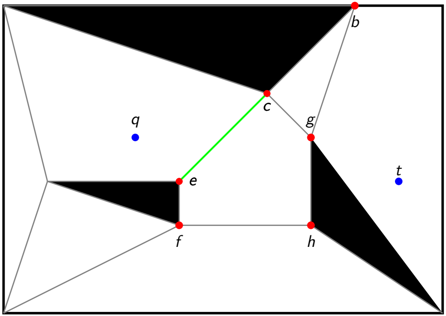
Polyanya: Overview

- black polygons are obstacles
- q: query point
- t: target
- gray border convex polygons are meshes
- mesh: all inside points are visible
- mesh: guides pathfinding



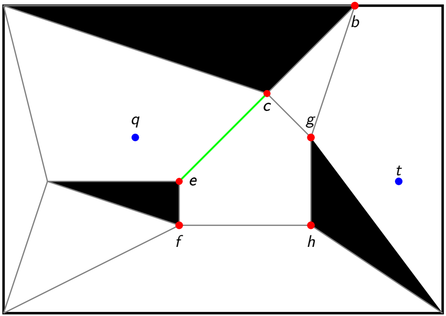
Polyanya: Search Node

- root r: $r \in (V \cup \{q\})$
- interval I: on an edge
- all point $\in I$: visible from r



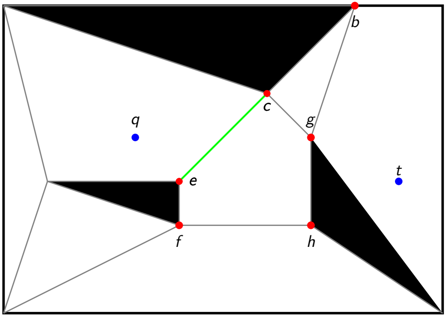
Polyanya: Search Node

- root r: $r \in (V \cup \{q\})$
- interval I: on an edge
- all point $\in I$: visible from r



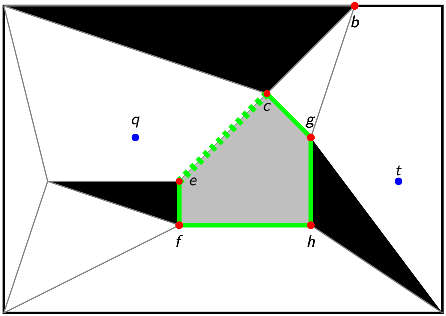
Polyanya: Search Node

- root r: $r \in (V \cup \{q\})$
- interval I: on an edge
- all point $\in I$: visible from r



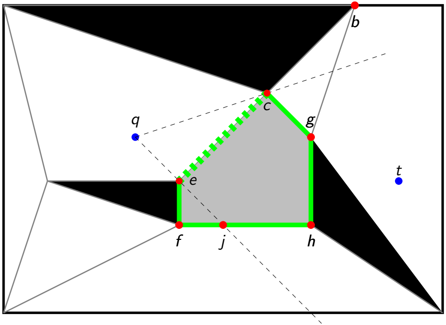
Polyanya: Successors

Successors are generated by pushing the search node away to adjacent mesh.



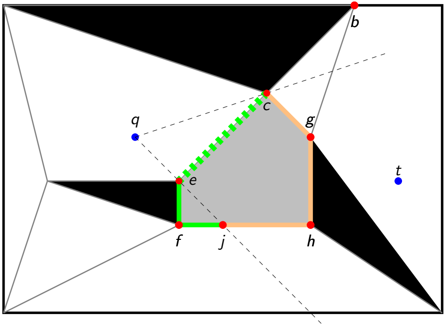
Polyanya: Successors

- Observable successors
  - root: parent's root
- Non-observable successors
  - root: an end point of I



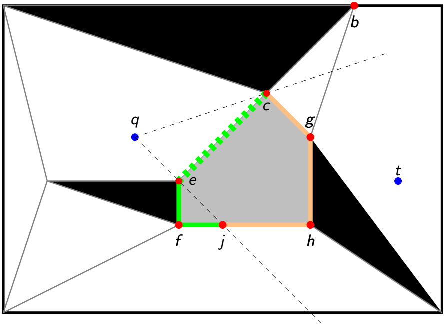
Polyanya: Successors

- Observable successors
  - root: parent's root
- Non-observable successors
  - root: an end point of I



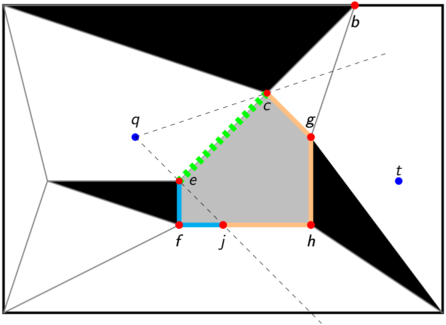
Polyanya: Successors

- Observable successors
  - root: parent's root
- Non-observable successors
  - root: an end point of I



Polyanya: Successors

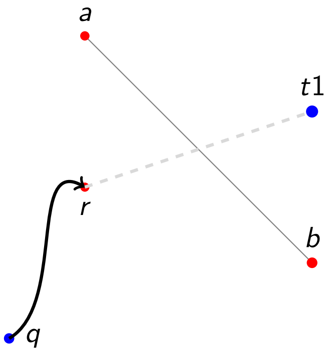
- Observable successors
    - root: parent's root
- Non-observable successors
    - root: an end point of I



Polyanya: Heuristic Search

Heuristic value of Search Node (r, I) has:

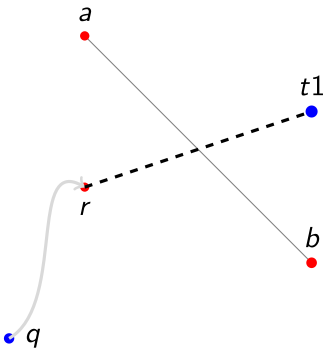
- *g-value*: $|shortestPath(q, r)|$ (certain)
- *h-value*: r to t_1 cross I (underestimation)
- *f-value*: *g-value* + *h-value* (underestimation of $|shortestPath(q, t_1|)$



Polyanya: Heuristic Search

Heuristic value of Search Node (r, I) has:

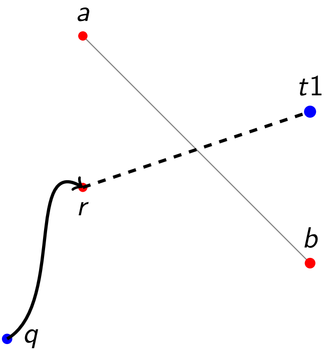
- *g-value*: $|shortestPath(q, r)|$ (certain)
- *h-value*: r to t_1 cross I (underestimation)
- *f-value*: *g-value* + *h-value* (underestimation of $|shortestPath(q, t_1|)$



Polyanya: Heuristic Search

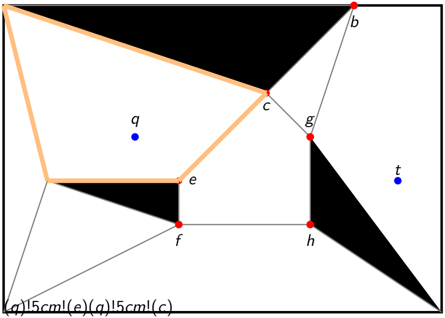
Heuristic value of Search Node (r, I) has:

- *g-value*: $|shortestPath(q, r)|$ (certain)
- *h-value*: r to t_1 cross I (underestimation)
- *f-value*: $g\text{-}value + h\text{-}value$ (underestimation of $|shortestPath(q, t_1|)$



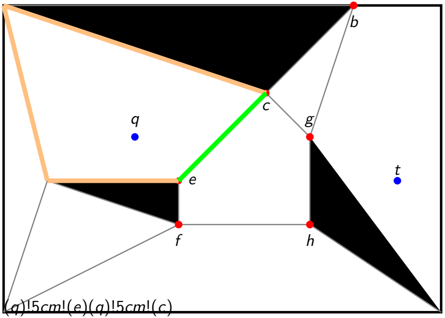
Polyanya: Example

Initial Search Nodes are edges of mesh that contains the q.



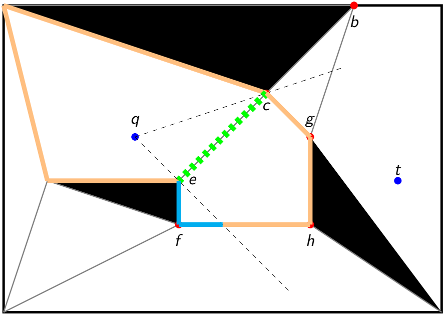
Polyanya: Example

Search Node $(q, [e.c])$ has the best estimation, so popped out



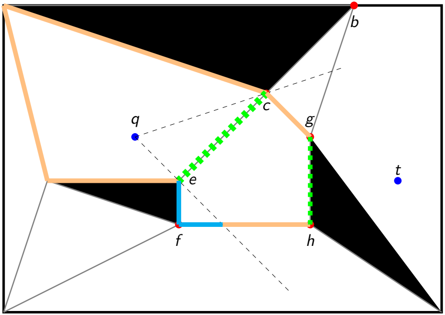
Polyanya: Example

Expand successors in adjacent mesh.



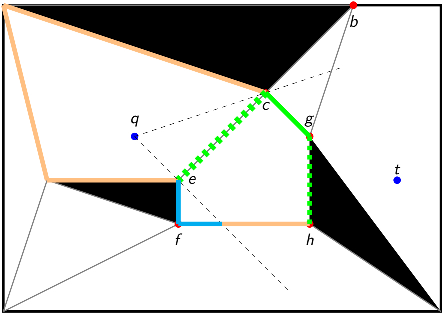
Polyanya: Example

Pop $(q, [g, h])$,
adjacent to obstacle,
so we discard it.



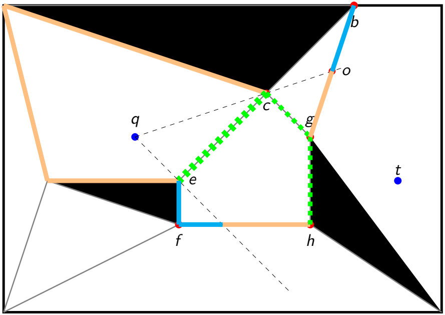
Polyanya: Example

Pop $(q, [c, g])$.



Polyanya: Example

Expand successors.



Polyanya: Example

Pop $(q, [g, o])$,

the adjacent mesh contains t,

we've found the shortest path!