Hierarchical Planning

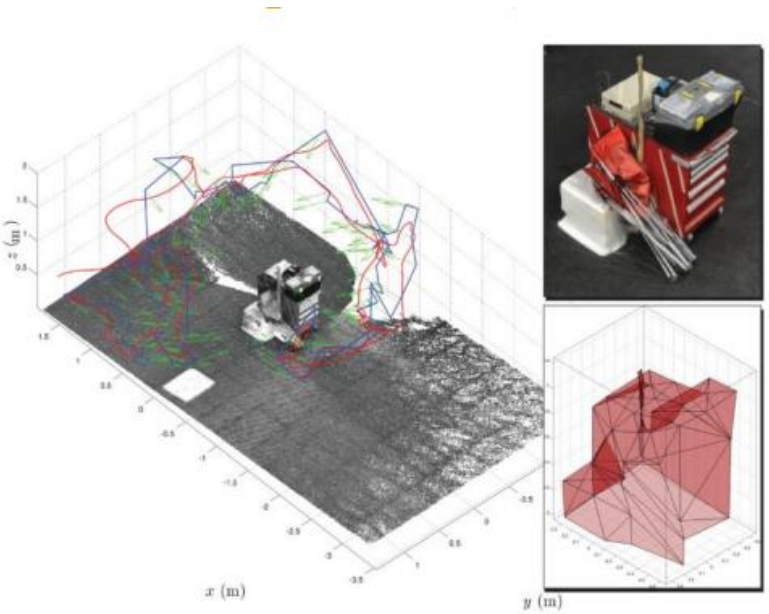
2019/11/26

20195062 Jaeyoon Kim

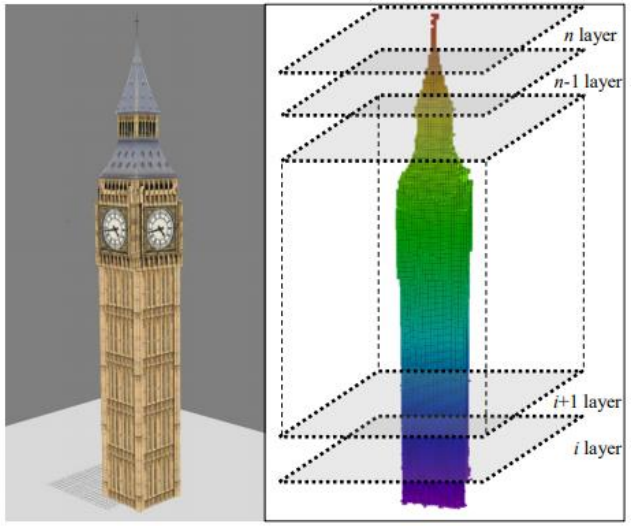
KAIST

Recap.

1. Structural Inspection Path Planning via Iterative Viewpoint Resampling with Application to Aerial Robotics
   - Minimize redundant viewpoints in terms of 3D recon.



2. Multi-layer Coverage Path Planner for Autonomous Structural Inspection of High-rise Structures



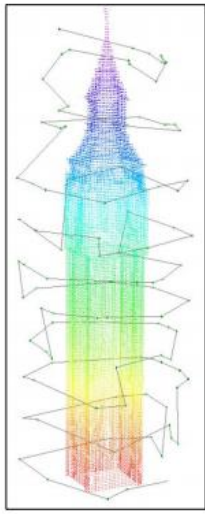


Table of Contents

Background of hierarchical planning

-Issue of local planner
-Hierarchical planner as its solution

Local planner in hierarchical planner

- Local planner (like RRT*):
  - Should consider kinodynamic, dynamics and other constraints while planning.

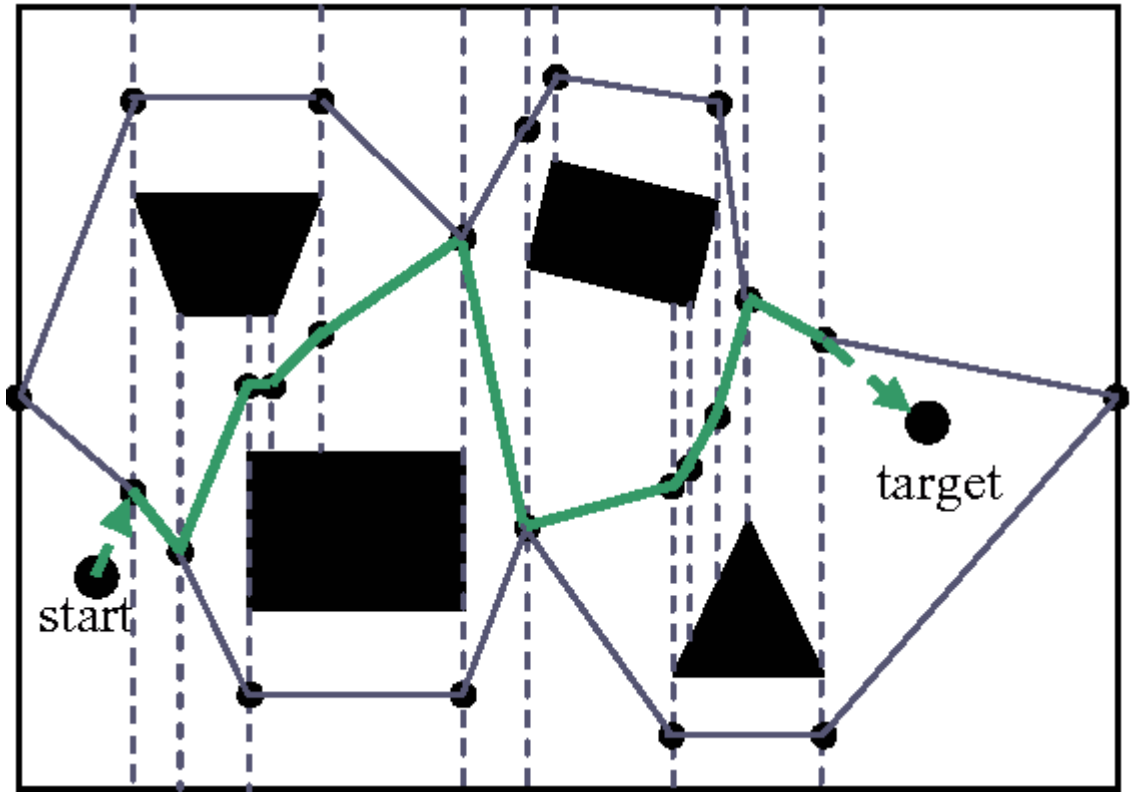
  - Need to handle high dimensional search space that emerges from the number of many constraints.

  - Is suitable for the planning to reflect the real world.

- However, it causes a heavy computational burden to run the local planner over the whole space.

Hierarchical planner as its solution

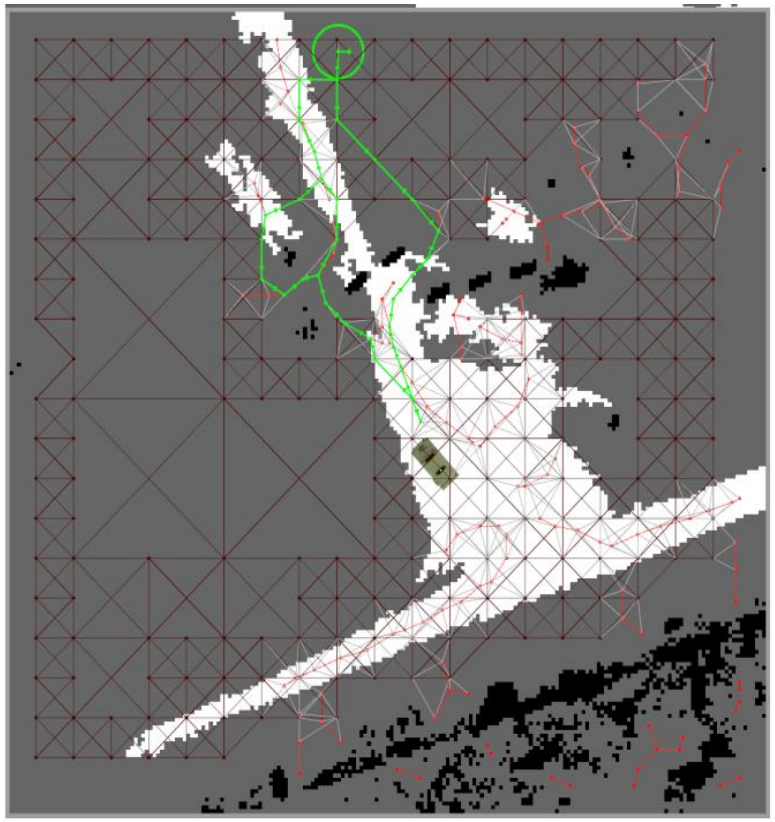
- To reduce the size of searching space for the local planner,
- Global planner (like Voronoi-based planner) should guide the local planner!



The Maverick planner: An efficient hierarchical planner for autonomous vehicles in unstructured environments, IROS 17

The Maverick planner: An efficient hierarchical planner for autonomous vehicles in unstructured environments

- They develop Maverick planner for autonomous vehicles.
- Voronoi diagram and cell decomposition as a global planner.
- RRT* as a local planner.



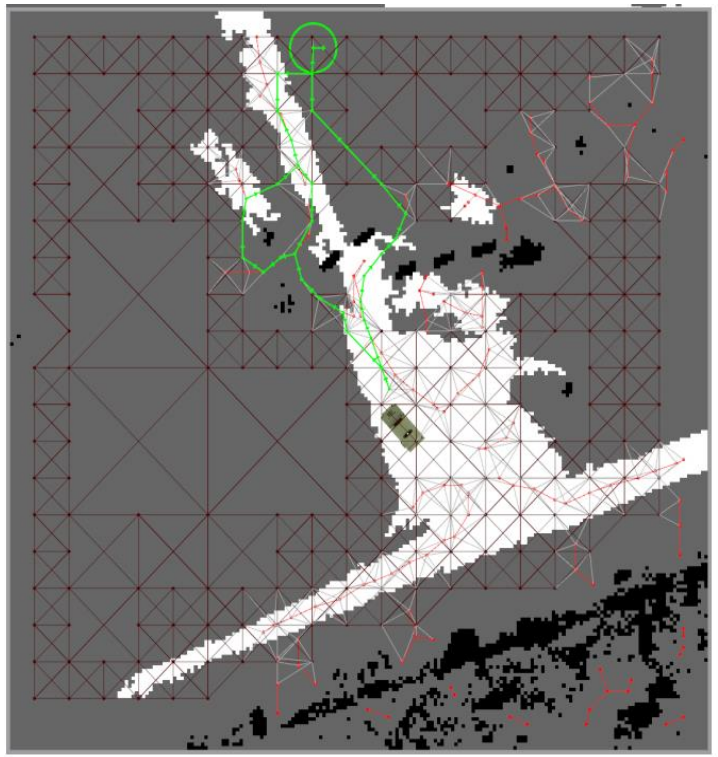
The Maverick planner: An efficient hierarchical planner for autonomous vehicles in unstructured environments

- Key features of Maveric planner:

➢Probabilistic completeness of traditional RRT*.

➢Convergence to the same solution as traditional RRT*

➢Continuous planning -> Anytime property

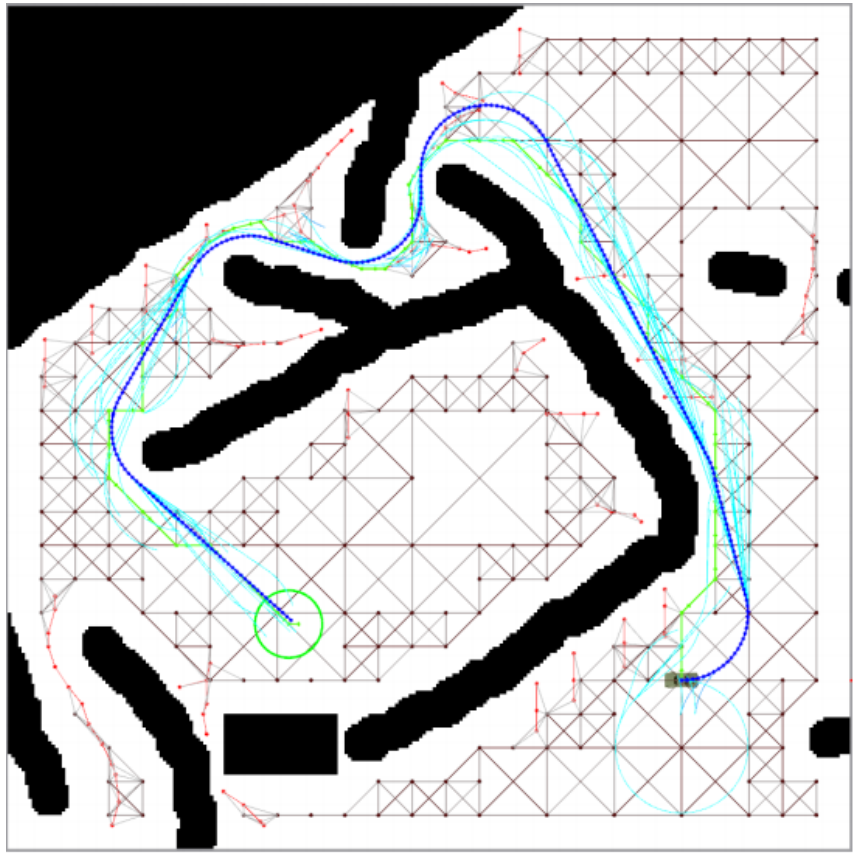


The Maverick planner: An efficient hierarchical planner for autonomous vehicles in unstructured environments

- An experimental result

Traditional RRT*, 20 sec

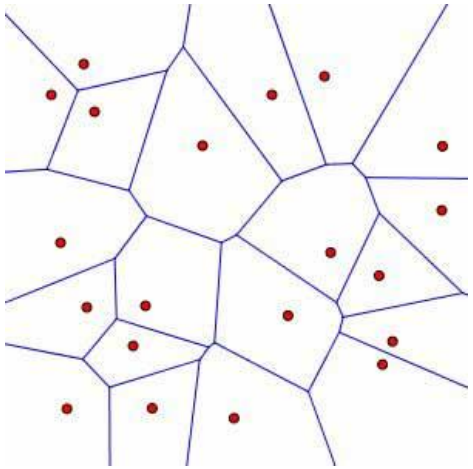
Global planner-guided RRT*, 0.1 sec



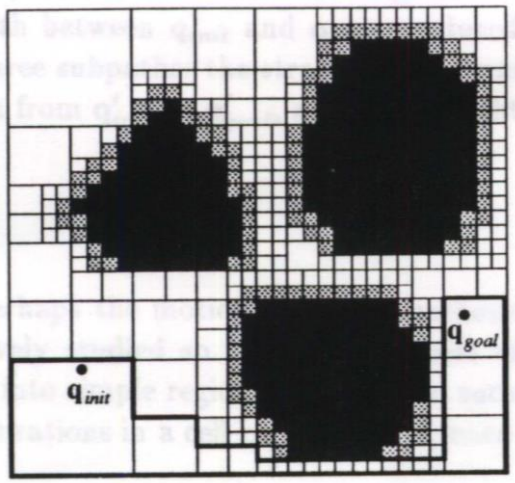
The Maverick planner: An efficient hierarchical planner for autonomous vehicles in unstructured environments

- Details of Maverick planner
  - Global planner

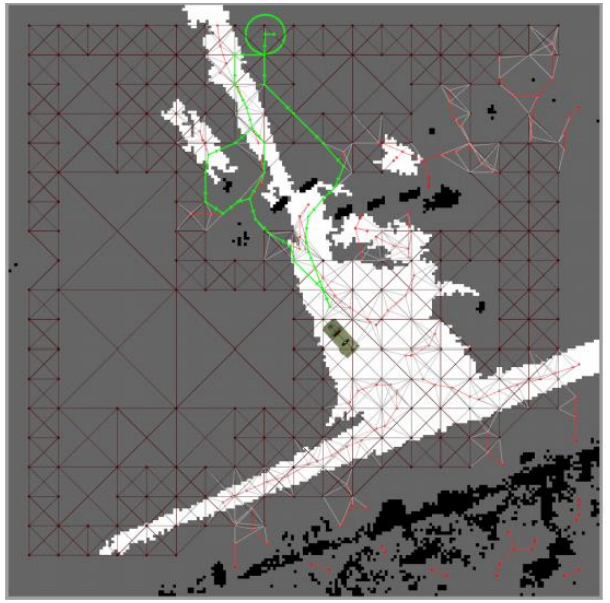
Voronoi diagram



Cell decomposition method



Dark red line means voronoi + cell decomposition result w.r.t. free space(gray, white)



obstacles (black area)

The Maverick planner: An efficient hierarchical planner for autonomous vehicles in unstructured environments

- Searching the graph
  - It can be simply done by running A* algorithm to find a guiding path.

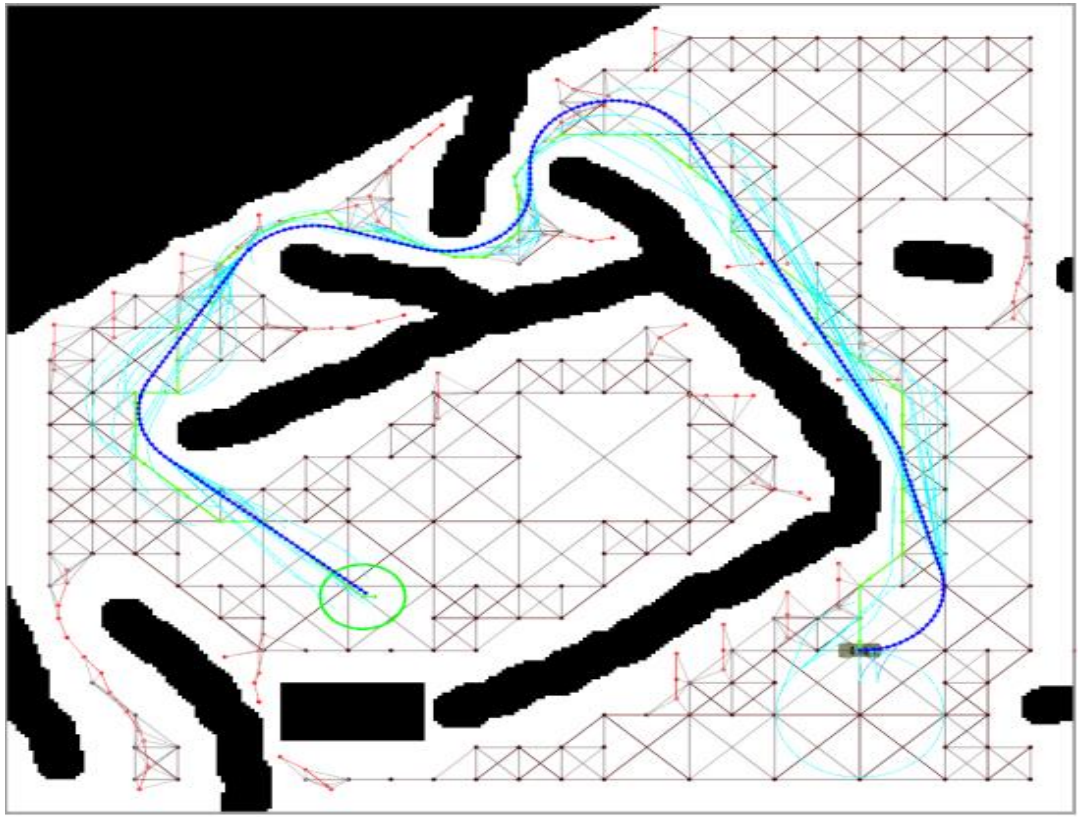
  - However, there can be no kinodynamically feasible path within homotopic paths of A*. (Note global planner doesn't consider kinodynamics)

  - Therefore, they calculated all paths from source to goal in the graph.

Used for local planner

The Maverick planner: An efficient hierarchical planner for autonomous vehicles in unstructured environments

- Local planner
  - Implement with traditional RRT*.
  - The calculated paths from global planner is used for sampling a waypoint of RRT*.



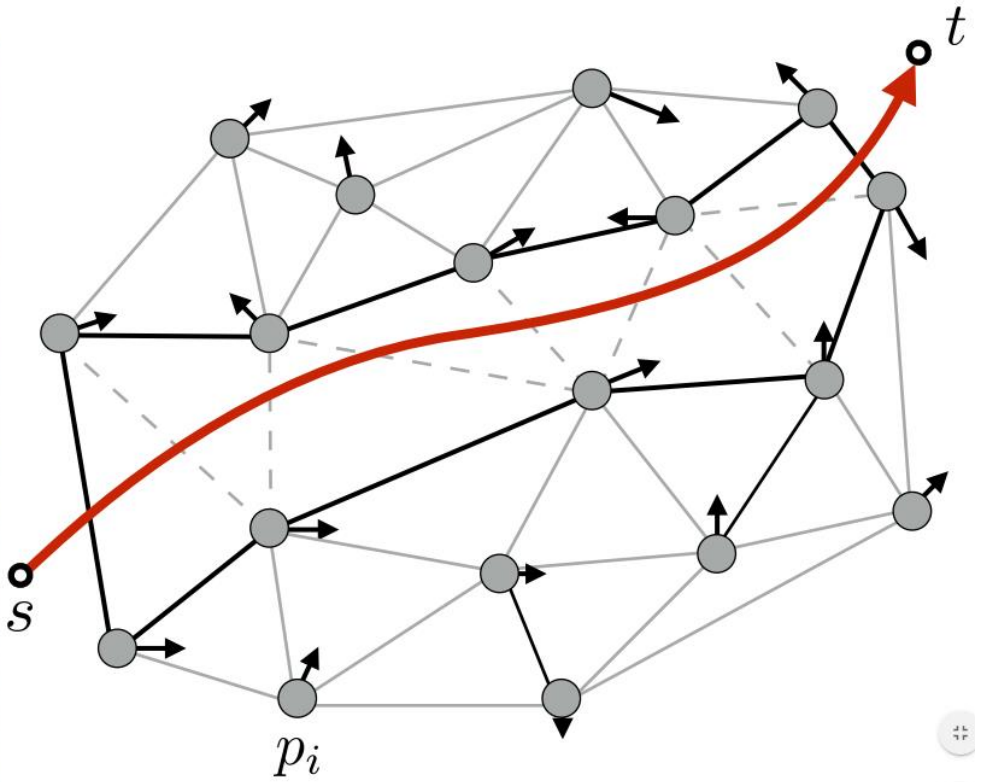
Dark blue: the optimal path
Light blue: visited paths from RRT*
But, not optimal one

Dynamic Channel: A Planning Framework for Crowd Navigation, ICRA 19

Dynamic Channel: A Planning Framework for Crowd Navigation

- Crowd Navigation

(a)



(b)

Dynamic Channel: A Planning Framework for Crowd Navigation

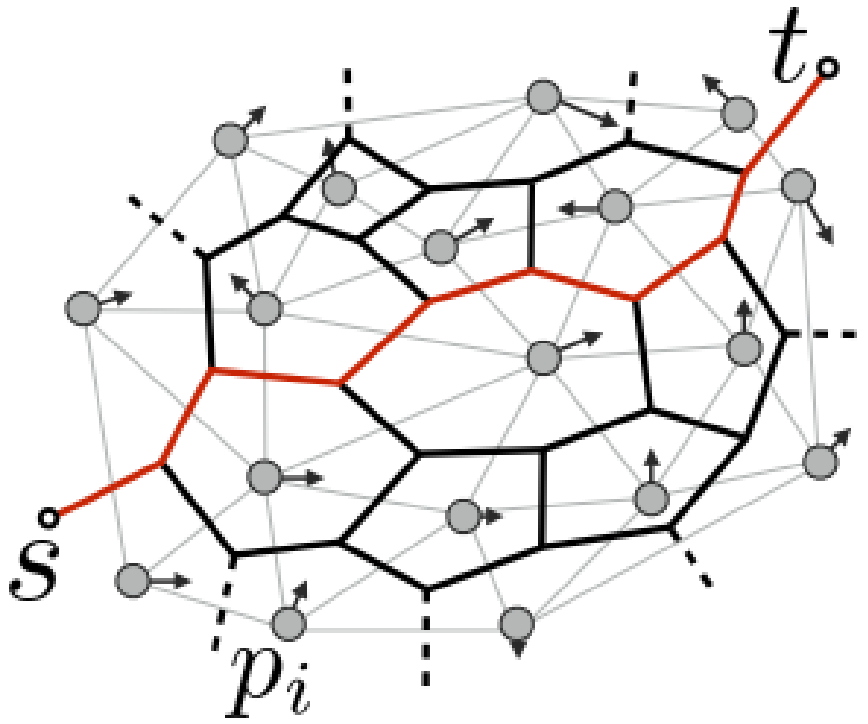
- Detailed method

1. Calculate Voronoi diagram with duality from Delaunay triangulation.
2. Run A* algorithm on the Voronoi graph.
3. Determine a dynamic channel that is a safe area for the robot to move.
4. Perform a path optimization where they consider whether some pedestrians are threatening or not.

Dynamic Channel: A Planning Framework for Crowd Navigation

- Graphical explanation

1. Calculate Voronoi diagram with duality from Delaunay triangulation.
2. Run A* algorithm on the Voronoi graph.

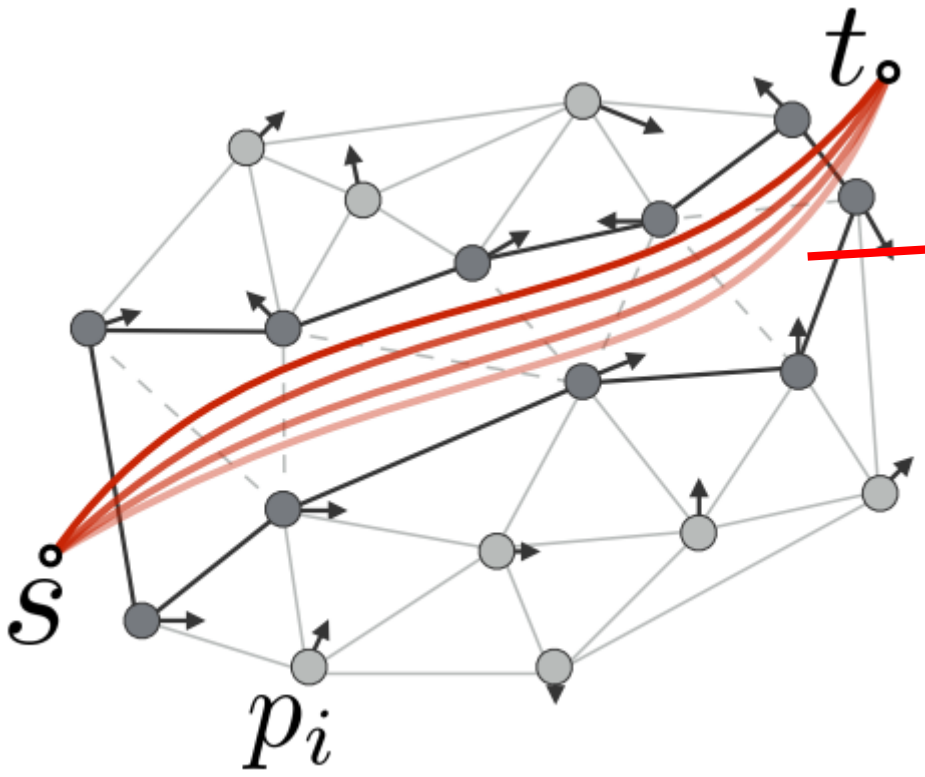


Gray node: pedestrian
Red path: shortest path from A*
Black arrow: velocity of each pedestrian

Dynamic Channel: A Planning Framework for Crowd Navigation

- Graphical explanation

3. Determine a dynamic channel that is a safe area for the robot to move.



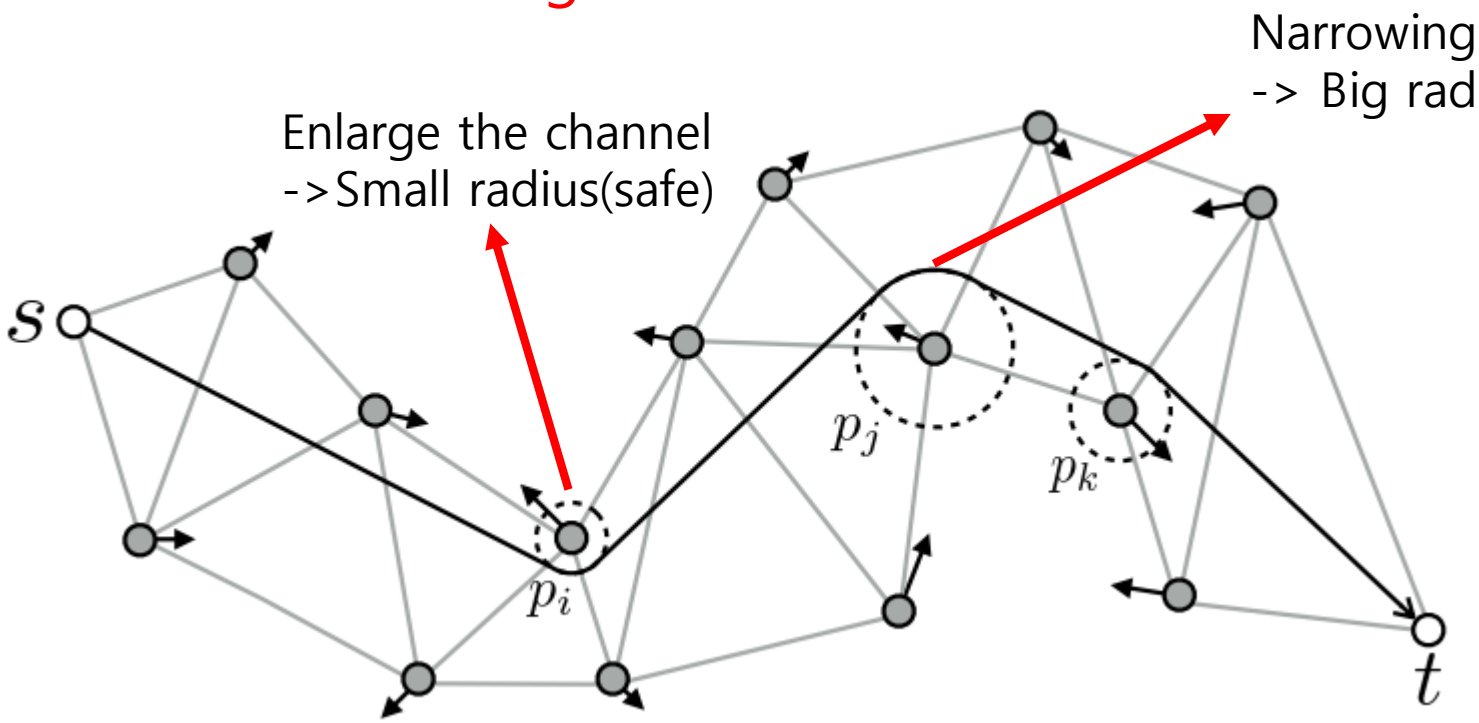
Dynamic channel,
Red lines are homotopic paths

Gray node: pedestrian
Black arrow: velocity of each pedestrian

Dynamic Channel: A Planning Framework for Crowd Navigation

- Graphical explanation

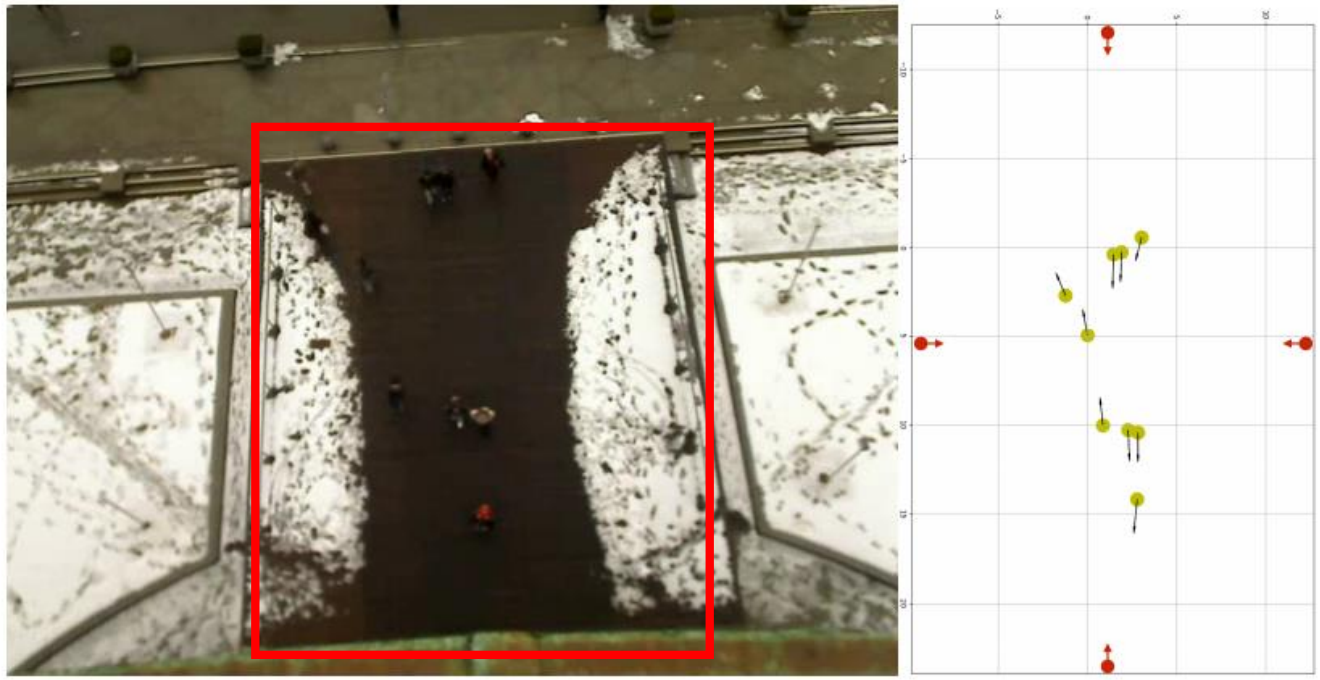
4. Perform a path optimization where they consider whether some pedestrians are threatening or not.



Gray node: pedestrian
Black arrow: velocity of each pedestrian

Dynamic Channel: A Planning Framework for Crowd Navigation

- Experimental setup



(a) (b)

Fig. 5: Experiment Settings. (a) shows a frame from the ETH dataset. (b) shows the pedestrian positions extracted from the frame. Red circles with arrows show the starting positions of the robot in the experiments.

Dynamic Channel: A Planning Framework for Crowd Navigation

- One prior work and one simple baseline for comparison

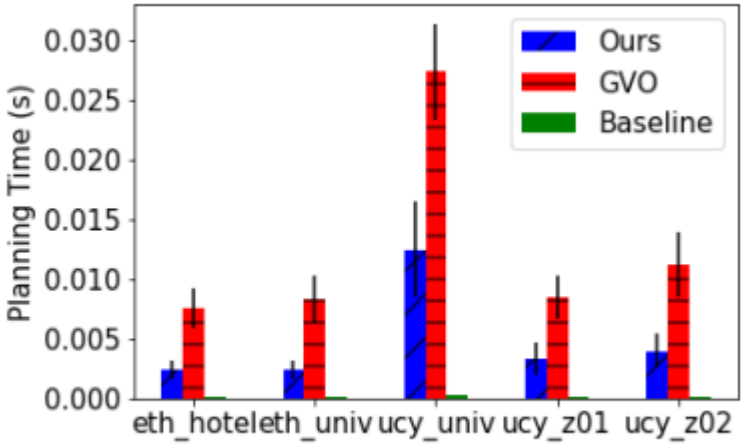
1. Generalized Velocity Obstacle Planner (GVO) [1]
   - Prior work for navigation on dynamic obstacle.

2. Simple Wait-and-Go planner (Baseline)
   - Path is a simple straight-line towards the goal.
   - When the robot met an obstacle, it stops first and then resumes going (when possible).

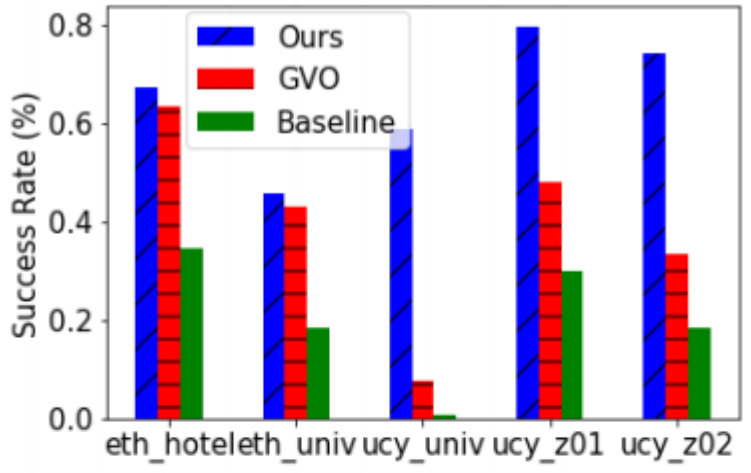
[1]D. Wilkie, J. Van Den Berg, and D. Manocha, "Generalized velocity obstacles," 2009 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS 2009, no. June 2014, pp. 5573–5578, 2009.

Dynamic Channel: A Planning Framework for Crowd Navigation

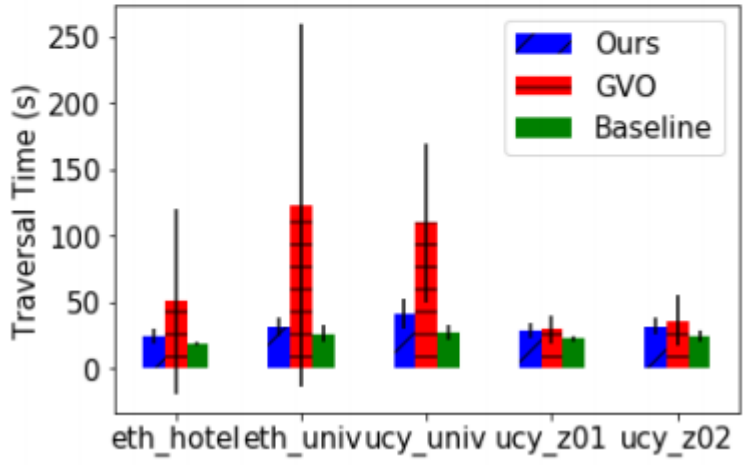
- Performance comparison



(a) Time Complexity



(a) Success Rate



(b) Efficiency

Thank you!!

Small quizzes

1. Local planner usually has a relatively much heavier than global planner. (T/F)

2. In hierarchical planning, global planner guides local planner for reducing computational complexity. (T/F)