Receding Horizon "Next – Best – View" Planner for 3D Exploration

Sungwook Jung

2017.05.23
CS686
Presentation #2

KAIST

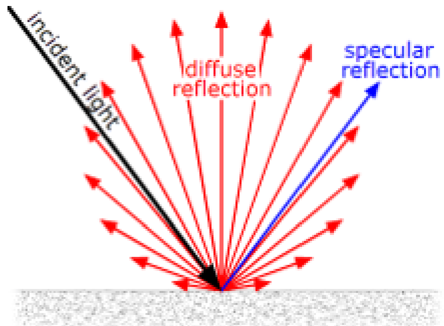
Review

Schissler et al., SIGGRAPH 2014

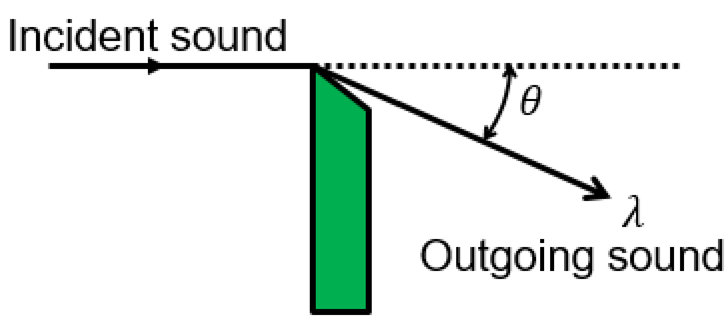
Inkyu An

Motivation | Sound Source Localization

- They suggested two directions.
  1) Radiosity(Diffuse) → To improve the realistic & reduce the time



  2) High order Diffraction
  → To reduce the computation time with simplification



Introduction

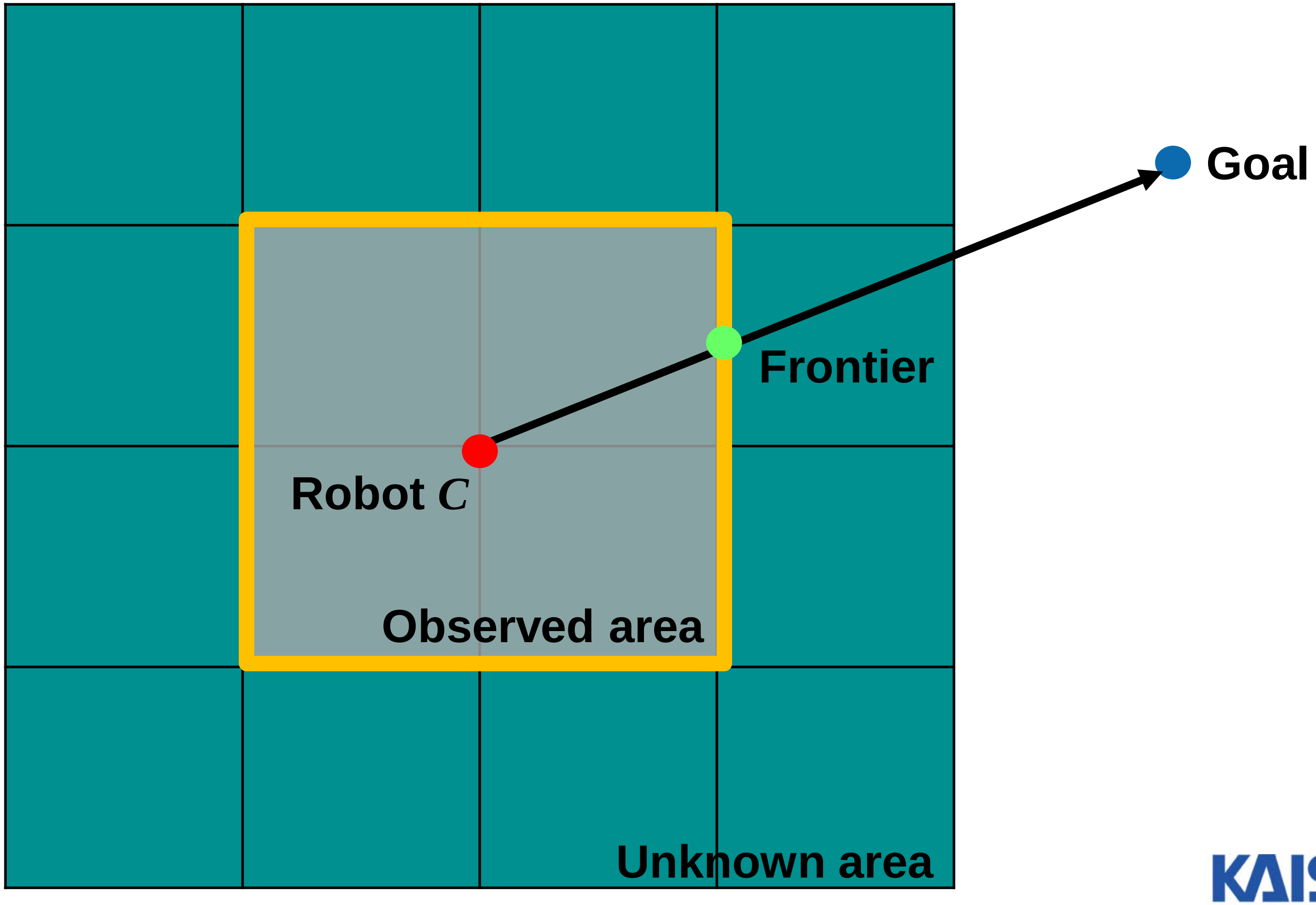
- This paper presents a novel path planning algorithm for the autonomous exploration of unknown space using aerial robotic platforms.
- running online, onboard a robot with limited resources.
- Evaluated in simulations as well as in a real world experiment using a UAV.
- Analysis on the computational complexity of the algorithm is provided
- Its good scaling properties enable the handling of large scale and complex problem setups.

Introduction

- **Frontier-based Planning?**
- **Receding Horizon Planning?**

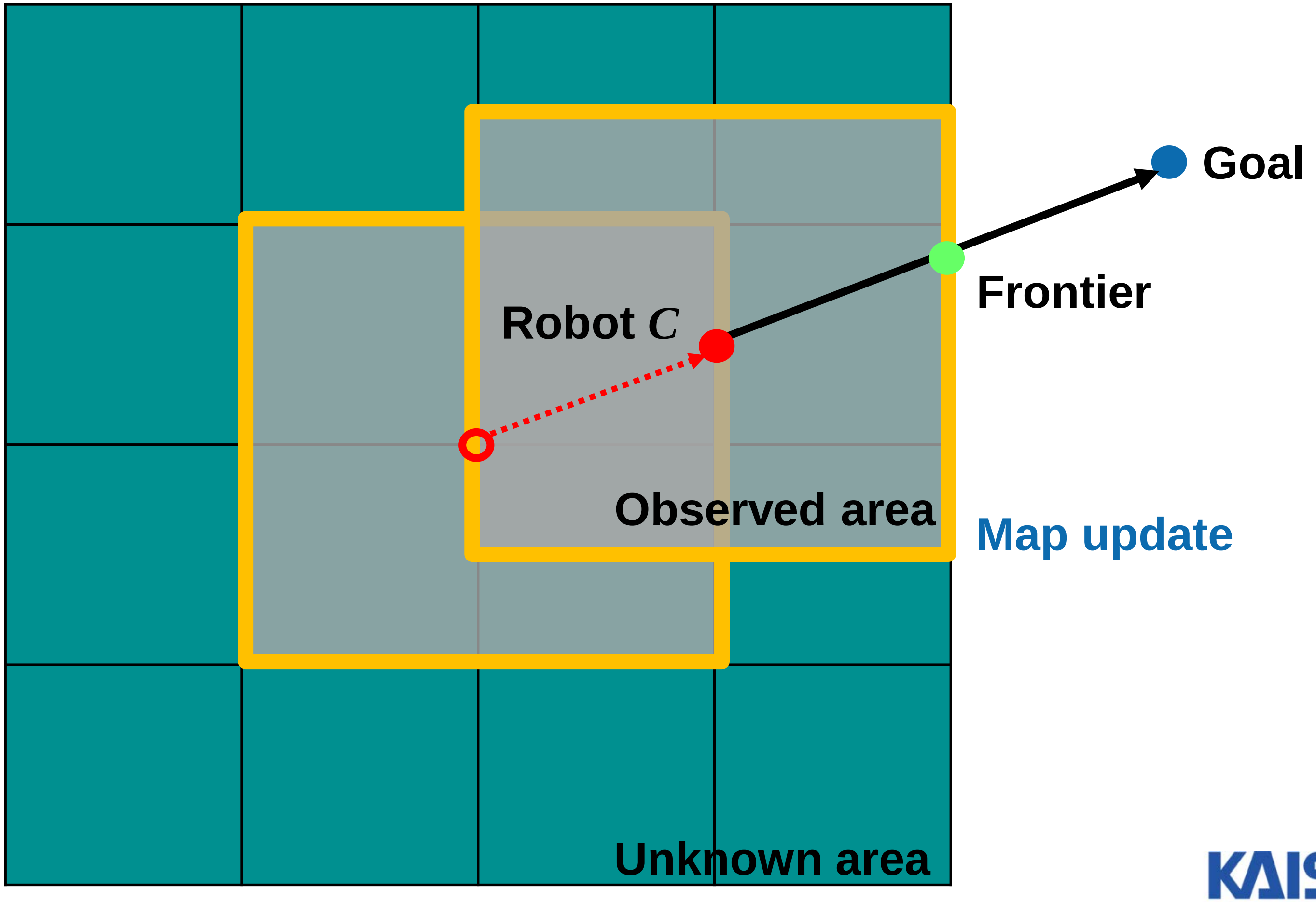
Frontier-based Planning (FBP)

- To gain the most new information about the world, move to the boundary between known space and unknown space

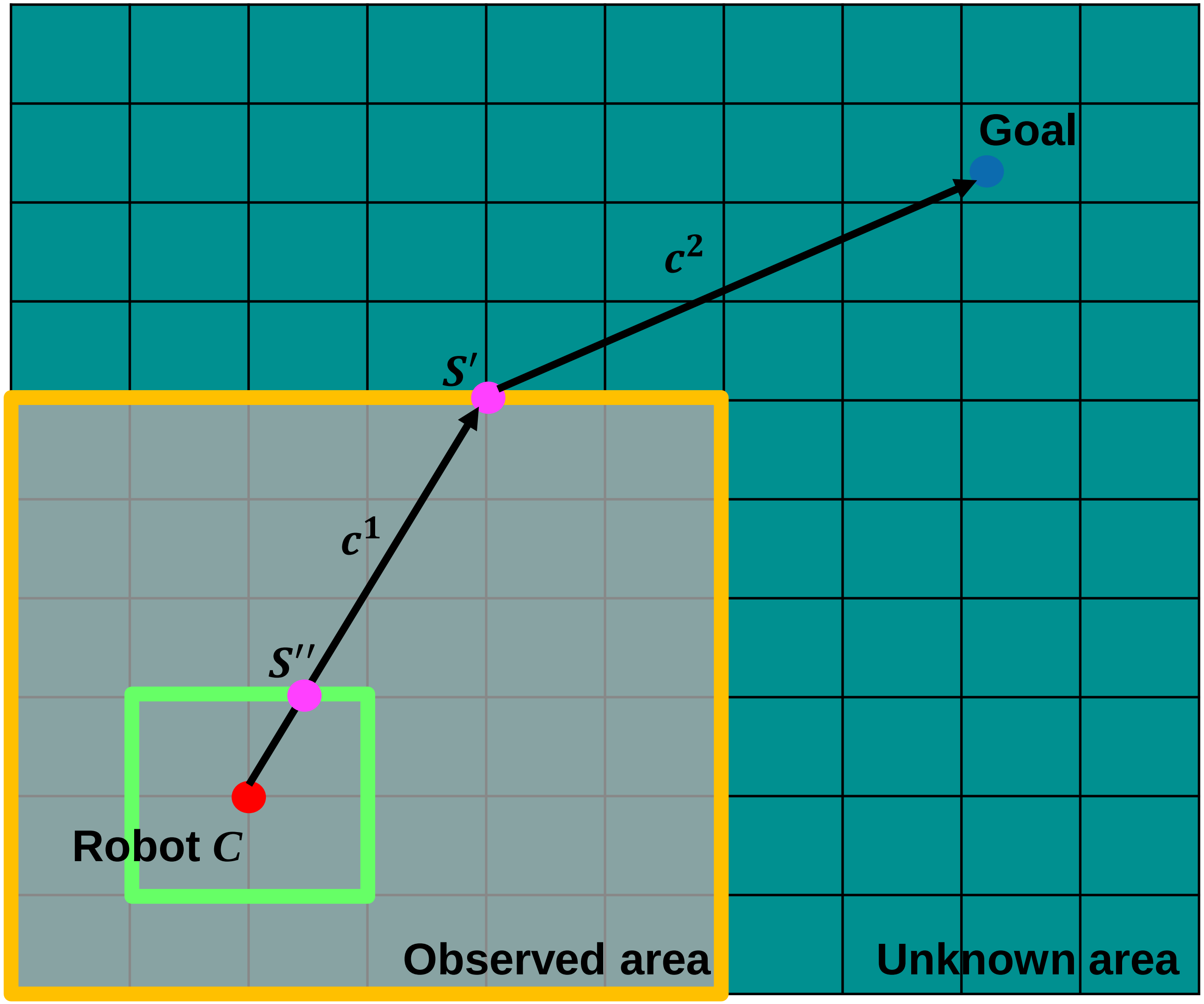


Frontier-based Planning (FBP)

- **To gain the most new information about the world, move to the boundary between known space and unknown space**



Receding Horizon Planner (RHP)

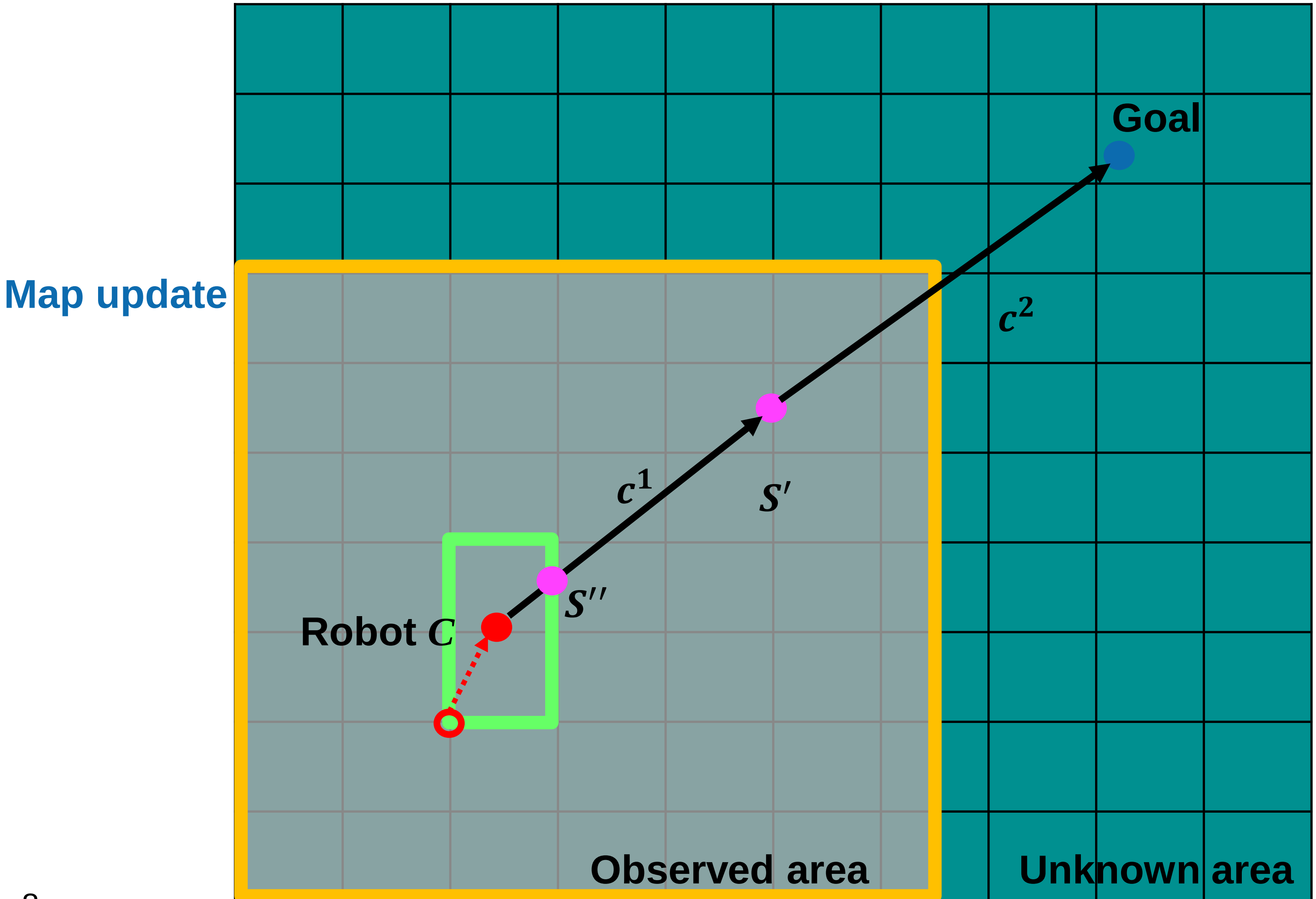


$min(c^1 + c^2)$

$\downarrow$

S'

Receding Horizon Planner (RHP)



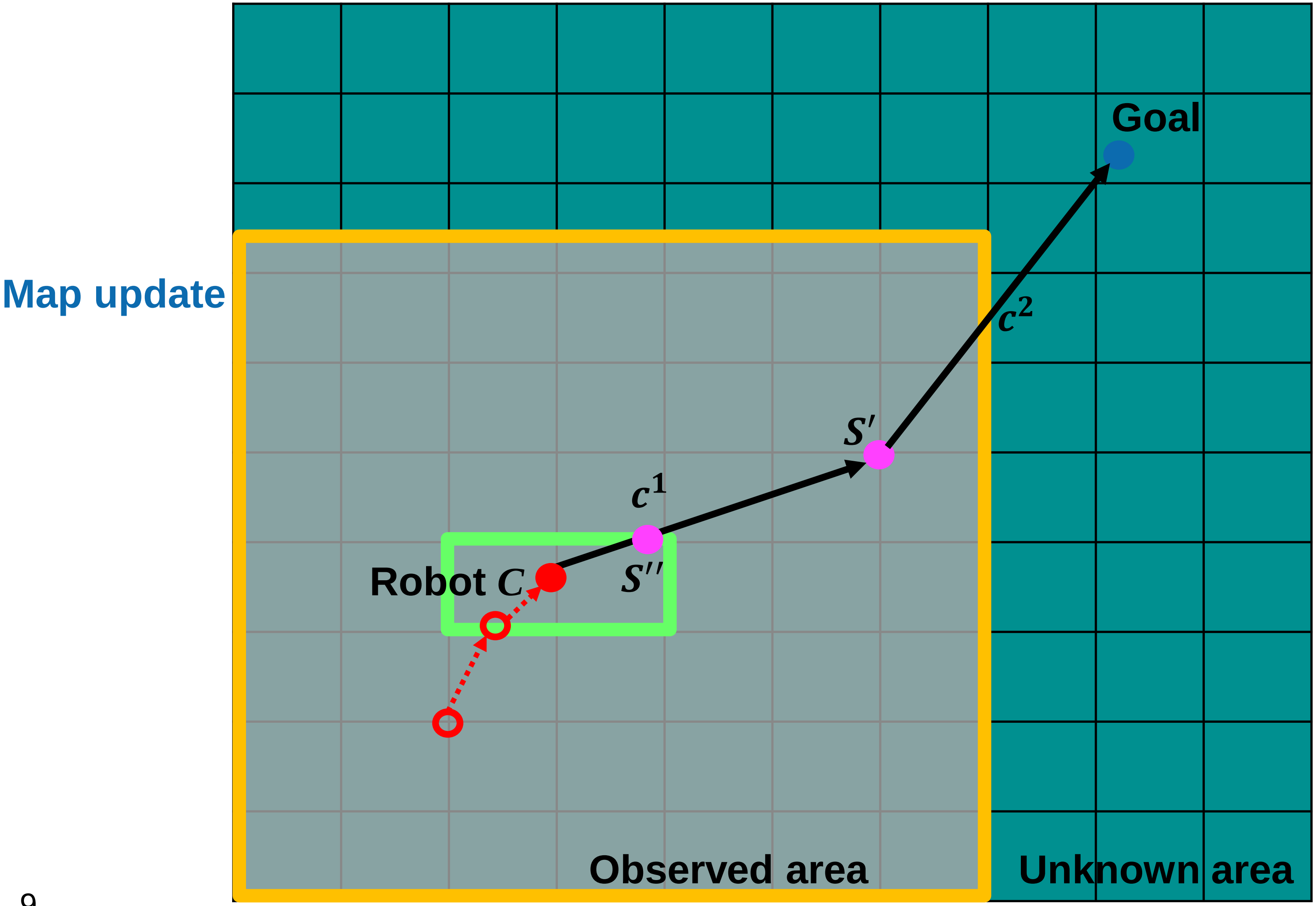
$min(c^1 + c^2)$

↓

S'

KAIST

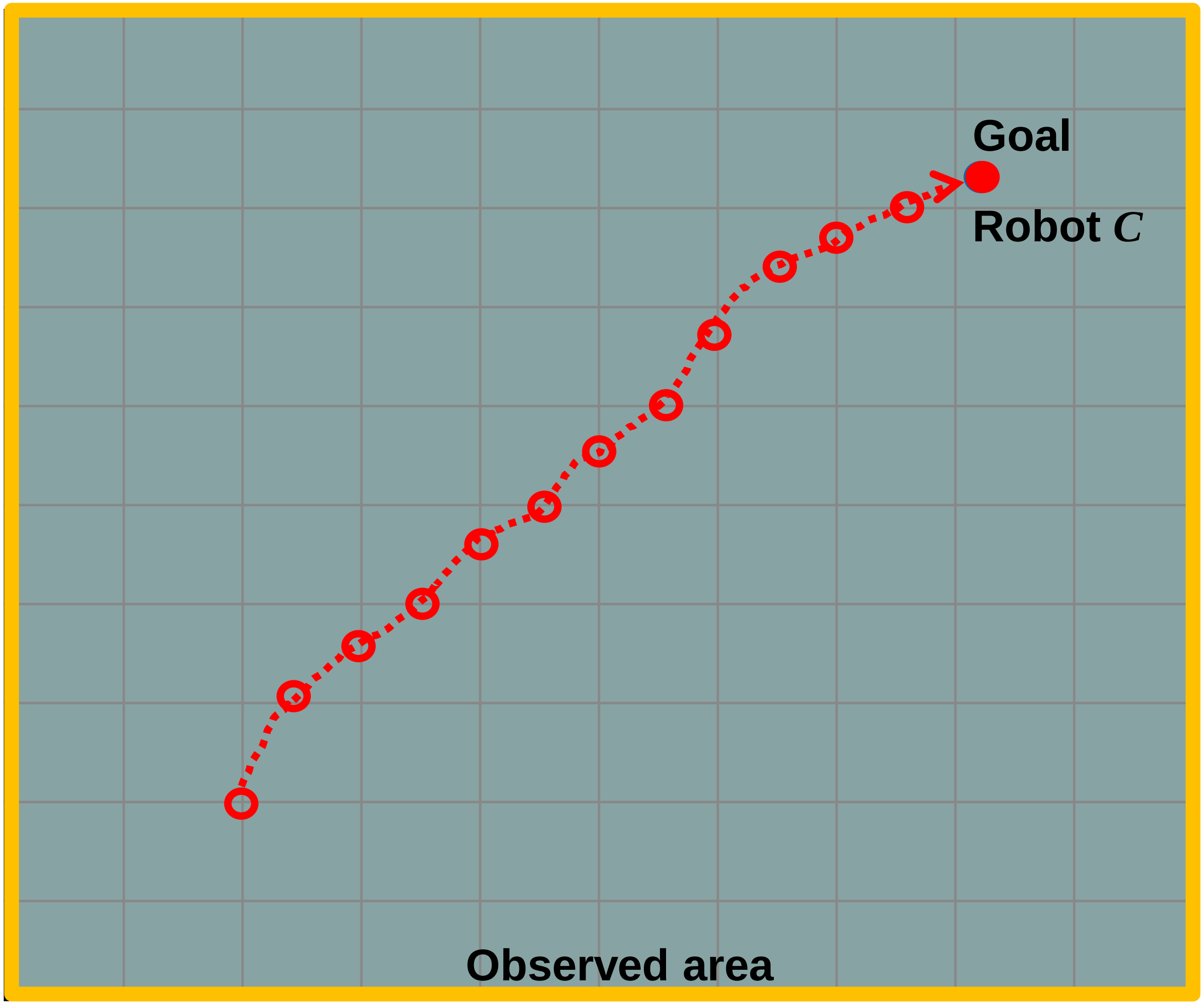
Receding Horizon Planner (RHP)



$$min(c^1 + c^2)$$
$$\downarrow$$
$$S'$$

Receding Horizon Planner (RHP)

Map update



Introduction

- **Receding Horizon Planning (RHP) ?**
  - **Only the first waypoint is executed when the robot moves.**
  - **The map is updated as more grids are explored, and the path is re-planned if necessary.**
  - **The benefit:**
    - **able to plan a smooth path where waypoints can be located on any position on the edge of grids without linear interpolation, which may not work for a cost function that includes nonlinear factors.**

KAIST

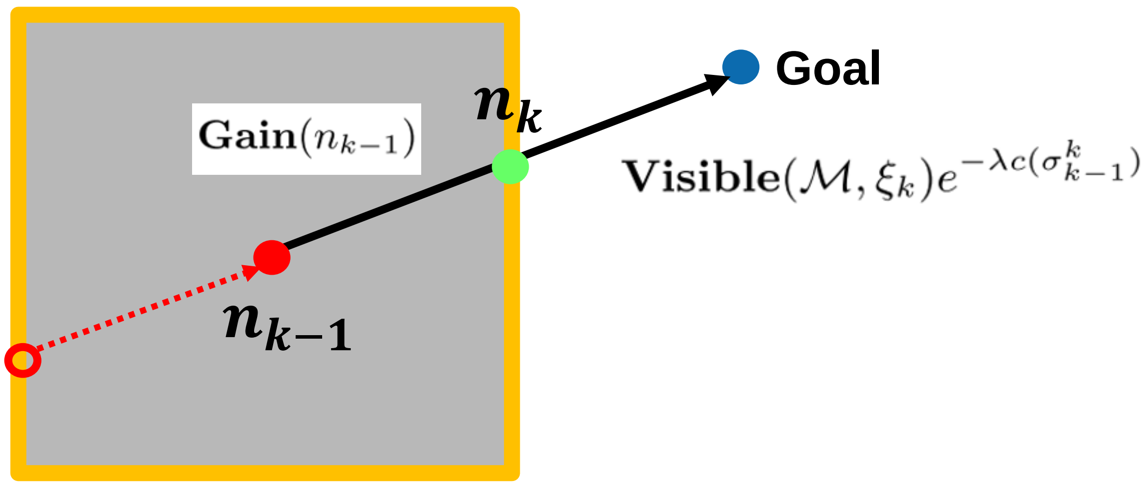
Problem description

- Does not require any prior knowledge of the environment
- From the current pose, expands a geometric tree (RRT) of possible future poses to find a next pose that gives a high exploration gain
- This gain reflects the exploration of space that is not yet known.
- As the vehicle proceeds on the path, the tree is recomputed, taking in account the new information from the sensor.
- In every iteration, the best previous branch is maintained.

Approach

- **The quality(collected information gain) of a new node**

$$\mathbf{Gain}(n_k) = \mathbf{Gain}(n_{k-1}) + \mathbf{Visible}(\mathcal{M}, \xi_k)e^{-\lambda c(\sigma_{k-1}^{k})}$$



- ***λ: tuning factor, penalizing high path cost [1]***
- ***$\mathcal{M}$: world, ξ:collision free vehicle configuration***
- ***$c(\sigma_{k-1}^{k})$: corresponding path cost***

[1] Gonzalez-Banos, Hector H., and Jean-Claude Latombe. "Navigation strategies for exploring indoor environments." The International Journal of Robotics Research 21.10-11 (2002): 829-848.

Approach

Algorithm 1 Exploration Planner - Iterative Step

$\xi_0 \leftarrow$ current vehicle configuration
Initialize $\mathbb{T}$ with ξ_0 and, unless first planner call, also previous best branch
$g_{best} \leftarrow 0$ ▷ Set best gain to zero
$n_{best} \leftarrow n_0(\xi_0)$ ▷ Set best node to root
$N_{\mathbb{T}} \leftarrow$ Number of initial nodes in $\mathbb{T}$
while $N_{\mathbb{T}} < N_{\max}$ **or** $g_{best} = 0$ **do**
Incrementally build $\mathbb{T}$ by adding $n_{new}(\xi_{new})$
$N_{\mathbb{T}} \leftarrow N_{\mathbb{T}} + 1$
if $\mathbf{Gain}(n_{new}) > g_{best}$ **then**
$n_{best} \leftarrow n_{new}$
$g_{best} \leftarrow \mathbf{Gain}(n_{new})$
if $N_{\mathbb{T}} > N_{TOL}$ **then**
Terminate exploration
$\sigma \leftarrow \mathbf{ExtractBestPathSegment}(n_{best})$
Delete $\mathbb{T}$
return σ

Complexity

$$\mathcal{O}(N_{\mathbb{T}} \log(N_{\mathbb{T}}) + N_{\mathbb{T}}/r^3 \log(V/r^3) + N_{\mathbb{T}}(d_{\max}^{\text{planner}}/r)^4 \log(V/r^3))$$

RRT complexity

Occupancy map complexity with 1/r^3 scale

Gain computation complexity

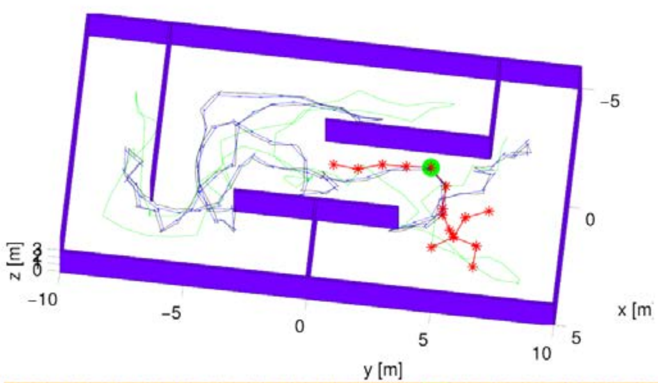
- *V: volume to explore*
- *r: resolution of occupancy map*
- $d_{max}^{planner}$*: sensor range*
- N_T*: number of nodes in the tree*

If map resolution and planning horizon are fixed, only depends on V

KAIST

Simulation (a)

- **RHP vs FBP - indoor**



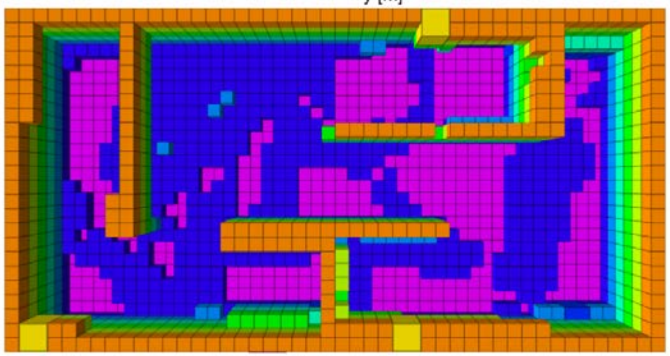


TABLE I: Apartment Exploration Scenario Parameters

Parameter	Value	Parameter	Value
Area	20x10x3m	Map resolution r	0.4m
$v_{\max}$	0.2m/s	$\dot{\psi}_{\max}$	0.75rad/s
FoV	$[60, 90]^{\circ}$	Mounting pitch	15°
$d_{\max}^{\text{planner}}$	2m	$d_{\max}^{\text{sensor}}$	5m
λ	0.5	RRT max edge length	1m
$N_{\max}$	15	Collision box	0.5x0.5x0.3m

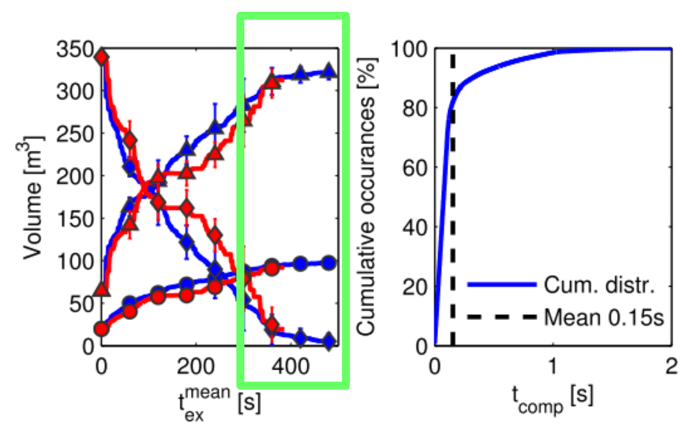


Fig. 4: On the left, statistics compare the receding horizon next–best–view planner (blue) with a frontier–based approach (red) for the apartment scenario. The depicted curves show the mean exploration progress over the execution time (computation time has been subtracted) for **occupied** o, **free** △ and **unmapped** ◊ volumes. The standard deviation is given every minute of execution time.

Simulation (b)

- **RHP vs FBP – large scenario**

Parameter	Value	Parameter	Value
Area	50x26x14m	Map resolution r	0.25m
$v_{\max}$	0.5m/s	$\dot{\psi}_{\max}$	0.75rad/s
FoV	$[60, 90]^\circ$	Mounting pitch	15°
$d_{\max}^{\text{planner}}$	2m	$d_{\max}^{\text{sensor}}$	10m
λ	0.2	RRT max edge length	3m
$N_{\max}$	30	Collision box	0.5x0.5x0.3m

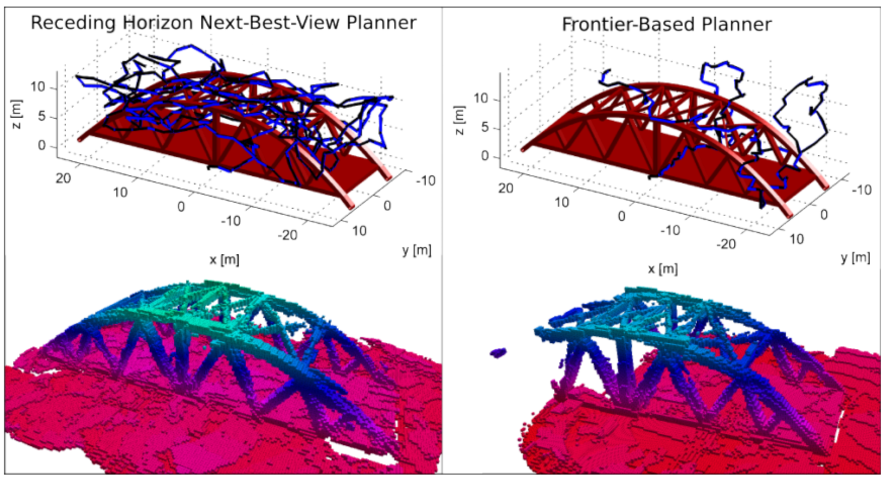
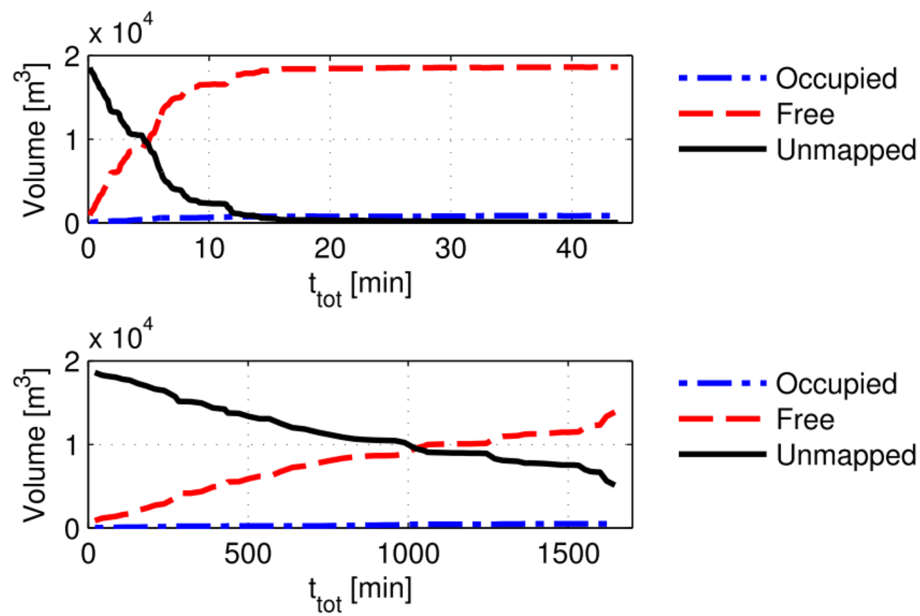


Fig. 5: The bridge model is displayed along with the computed path (blue), the simulated vehicle response (black) and the acquired occupancy map in the lower part of the figure. These results are given for the proposed planner on the left side and the compared frontier–based planner on the right. Note that the latter only made partial exploration due to excessive computation time for this large scale problem.



Experiment

Parameter	Value	Parameter	Value
Area	9x7x2m	Map resolution r	0.2m
$v_{\max}$	0.25m/s	$\dot{\psi}_{\max}$	0.3rad/s
FoV	$[60, 90]^{\circ}$	Mounting pitch	15°
$d^{\text{planner}}_{\max}$	1m	$d^{\text{sensor}}_{\max}$	5m
λ	0.5	RRT max edge length	1.5m
$N_{\max}$	20	Collision box	1.2x1.2x0.5m

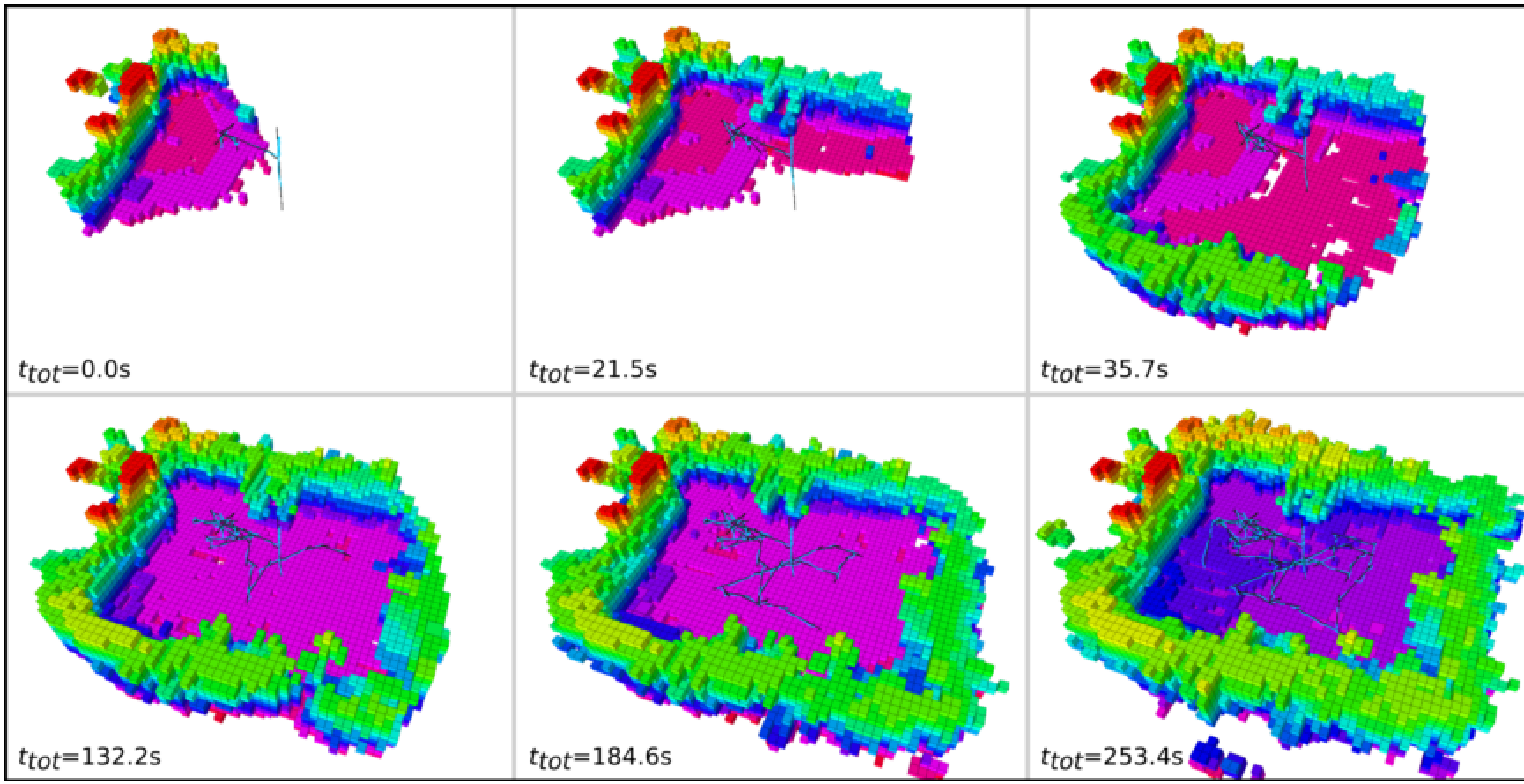


Fig. 8: The exploration experiment in a closed room is depicted. The colored voxels represent occupied parts of the occupancy map (colored according to height) while the computed path is given in black and the vehicle response in light blue. The initial phase of the exploration mission is dominated by yawing motions to maximize exploration without traveling large distances. Subsequently the MAV explores regions further away, to eventually accomplish its mission.

Summary & Conclusions

- **An exploration path planner was proposed that is capable of exploring a unknown area, constructing an occupancy map of the perceived environment.**
- **While collisions are avoided, good exploration paths are computed online, considering the updated model of the environment.**

- **Also, Open-sourced on Github.**

  (https://github.com/ethz-asl/nbvplanner)

- **Thanks**

KAIST