

2022 7th International Conference on Automation, Control and Robotics Engineering

# Adaptively Dynamic RRT\*-Connect: Path Planning for UAVs Against Dynamic Obstacles

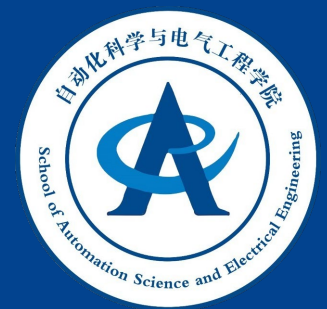
Yicheng Chen and Lingling Wang

Speaker | Yicheng Chen

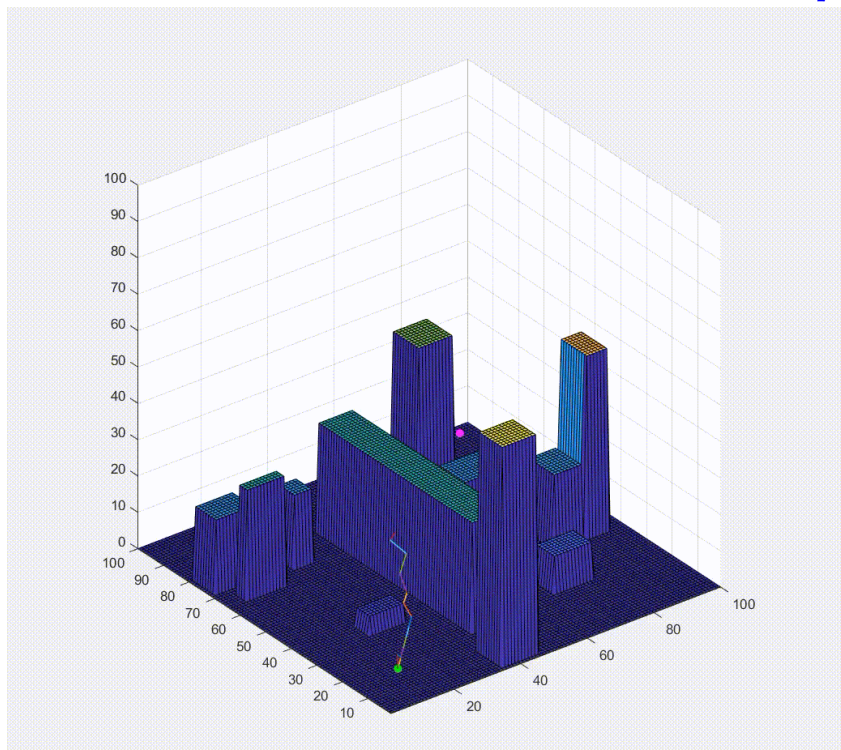
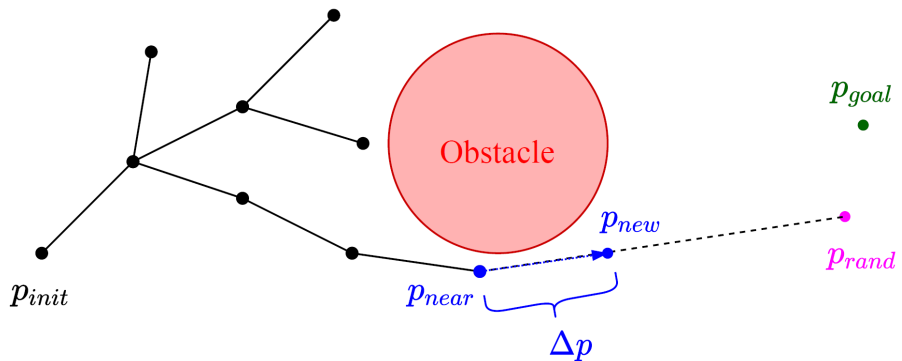
Master student

School of Automation Science and Electrical Engineering

Beihang University, Beijing, China

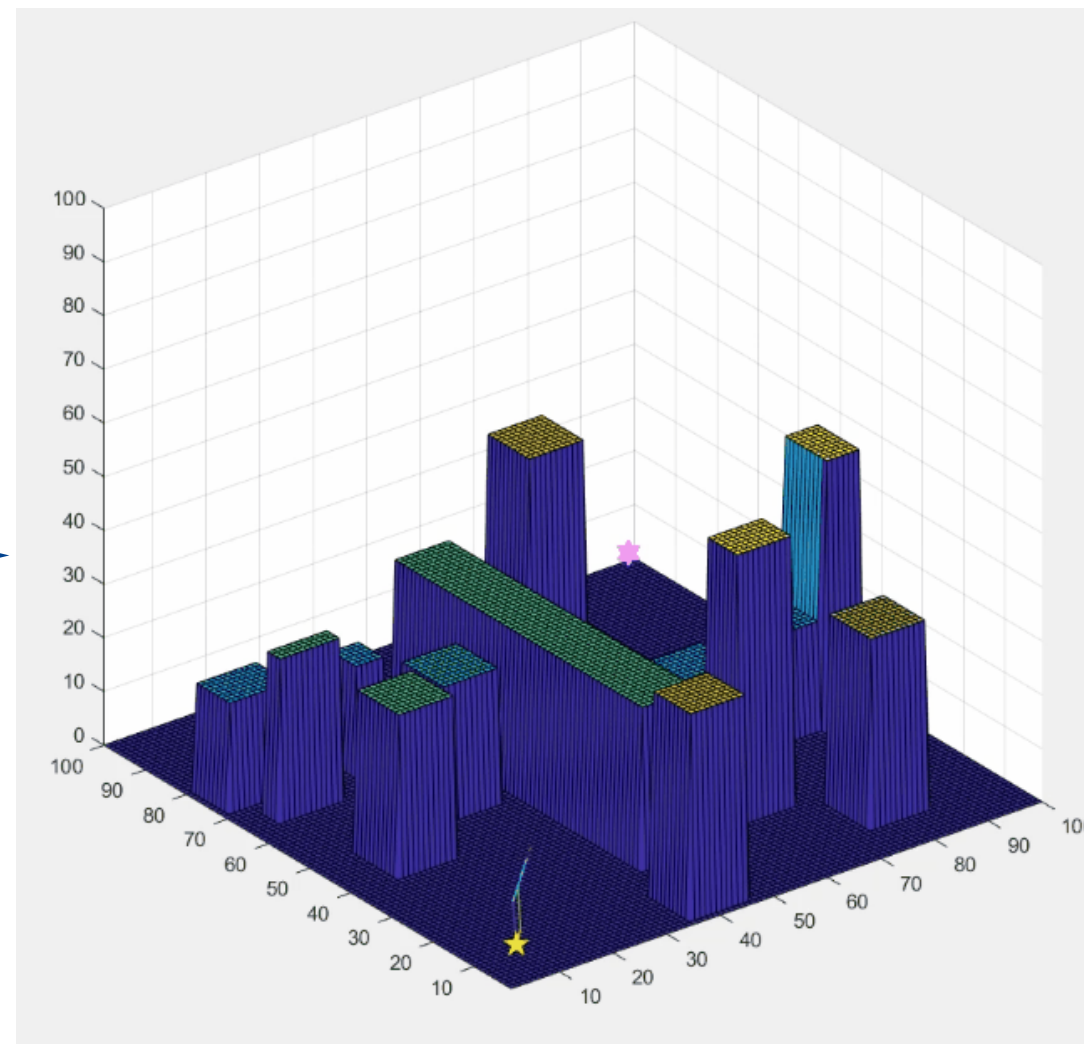


## RRT



## RRT\*

## Goal-bias RRT

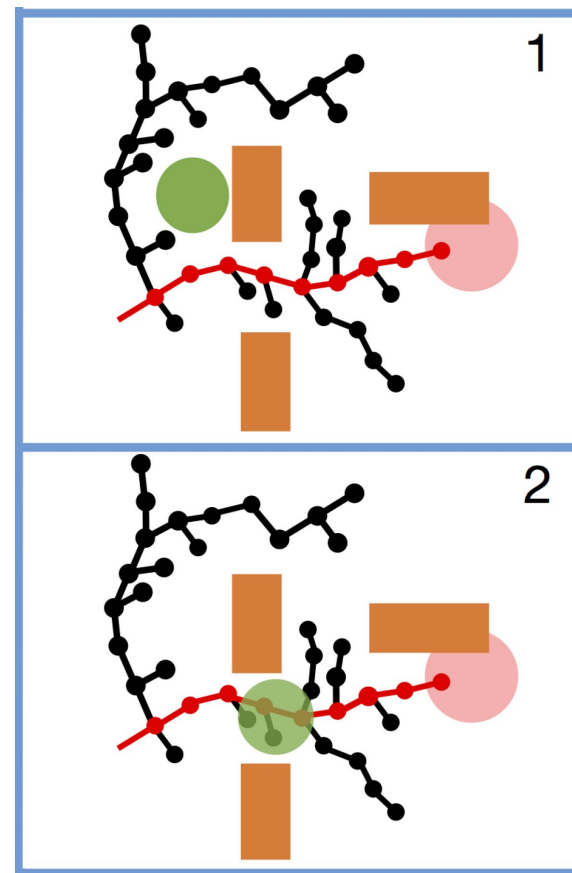


- Change of environment
- Map error or update
- Unreliable perception



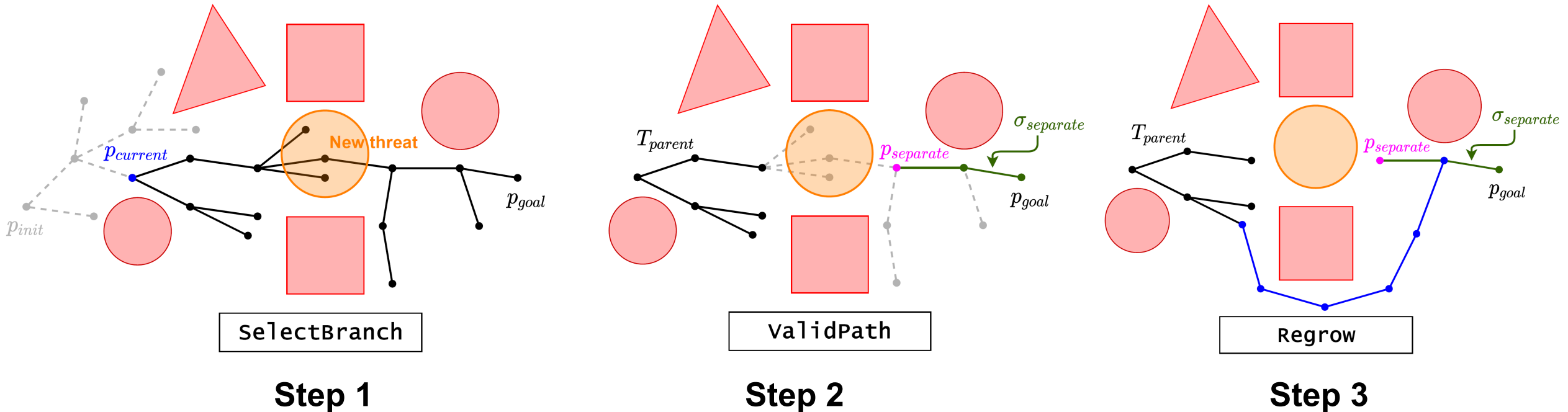
## Challenge:

What if new obstacles break the path?



## What if new obstacles break the path?

- Existing solution 1: Traditional re-planning method - Re-run a full planning
- Existing solution 2: RRT\*D<sup>[1]</sup>

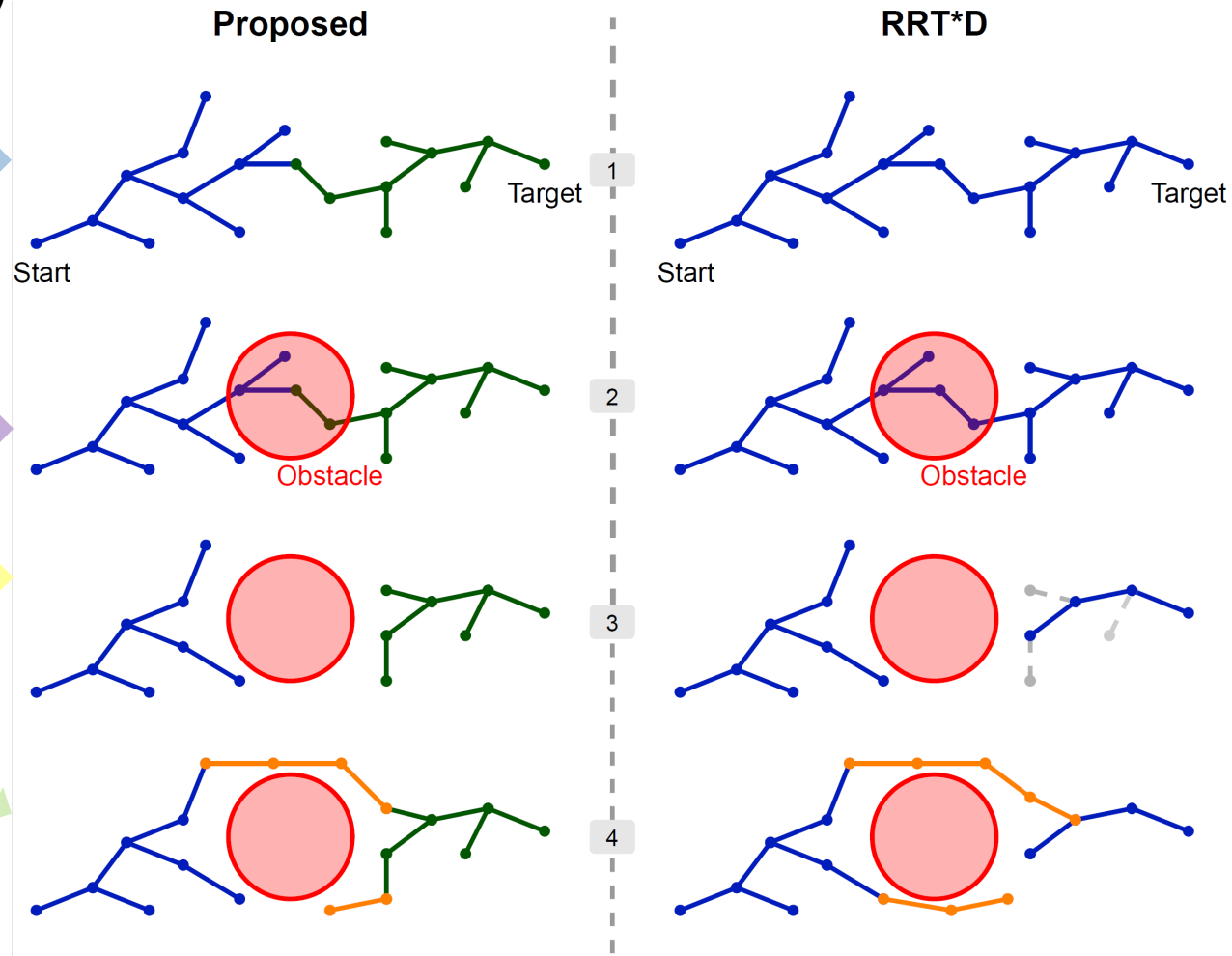


[1] Adiyatov, Olzhas, and Huseyin Atakan Varol. "A novel RRT\*-based algorithm for motion planning in Dynamic environments." *IEEE ICMA 2017*.

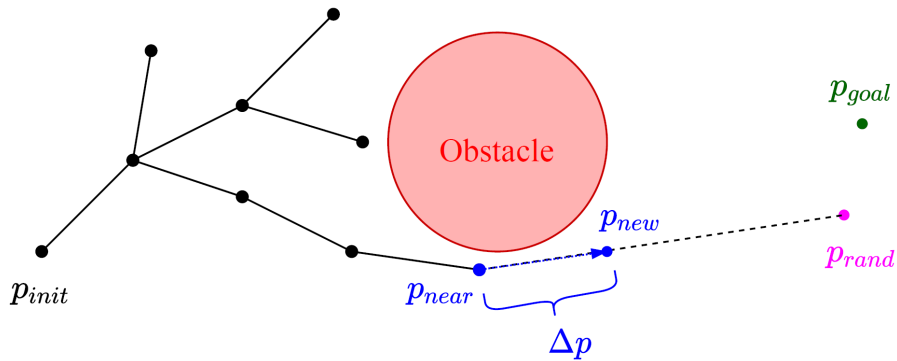
## Adaptively Dynamic RRT\*-Connect (ADRRT\*-Connect)

### Algorithm 6 ADRRT\*-Connect

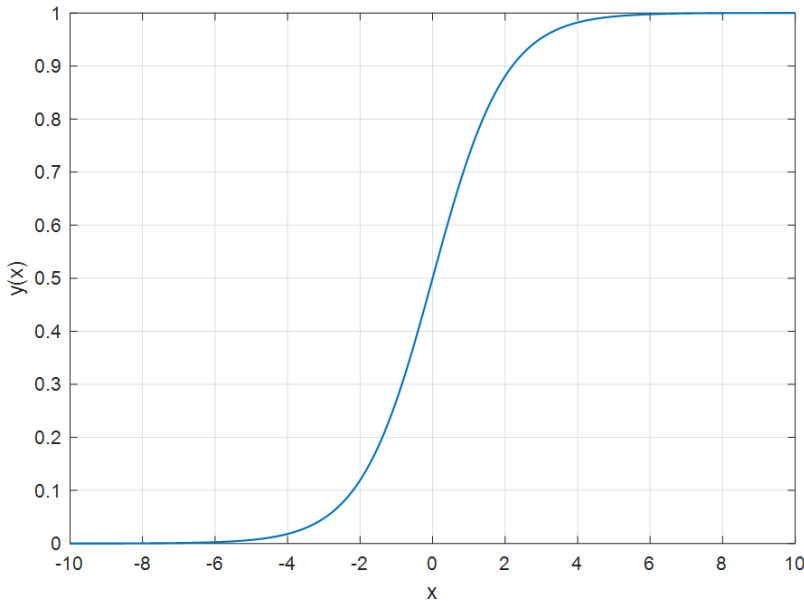
- 1:  $T_a, T_b, \sigma \leftarrow \text{ARRT}^*\text{-Connect}([\ ], [\ ])$
- 2:  $p_{\text{current}} \leftarrow p_{\text{init}}$
- 3:  $\text{InitMovement}()$
- 4: **while**  $p_{\text{current}} \neq p_{\text{goal}}$  **do**
- 5:      $D \leftarrow \text{UpdateObstacles}()$
- 6:     **if**  $\text{DetectCollision}(\sigma, p_{\text{current}})$  **then**
- 7:          $\text{StopMovement}()$
- 8:          $T_a \leftarrow \text{RemoveNodes}(T_a, p_{\text{current}})$
- 9:          $T_b \leftarrow \text{RemoveNodes}(T_b, p_{\text{goal}})$
- 10:          $T_a, T_b, \sigma \leftarrow \text{ARRT}^*\text{-Connect}(T_a, T_b)$
- 11:          $\text{ResumeMovement}()$
- 12:     **end if**
- 13:      $p_{\text{current}} \leftarrow \text{NextNode}(\sigma)$
- 14: **end while**



Adaptively adjust heuristic factor according to the sampling result



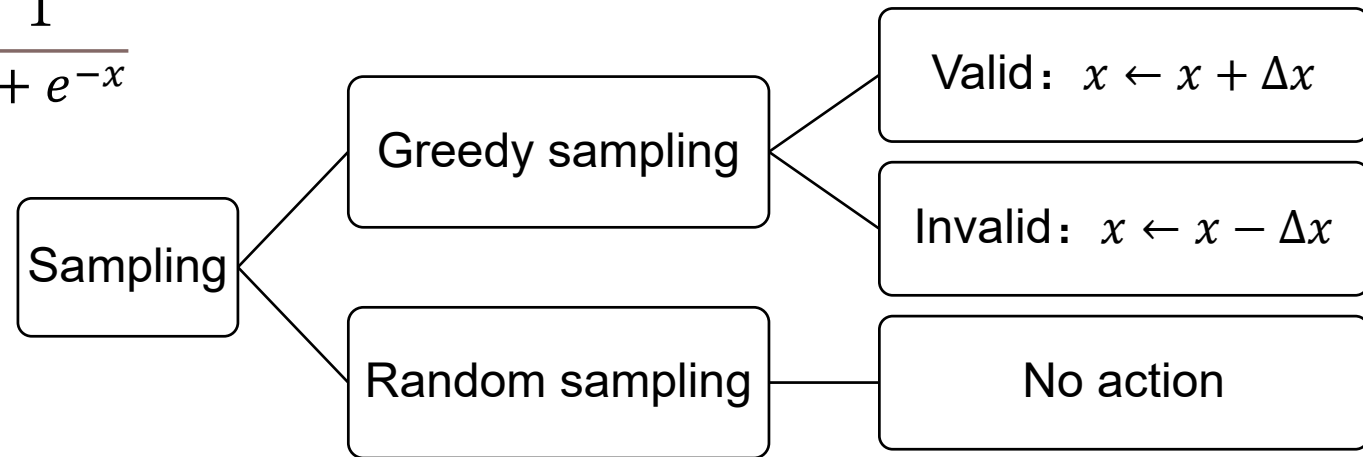
Sample type	Probability	
	Traditional heuristic	<b>Adaptive heuristic</b>
Greedy sampling: Directly sample $p_{goal}$	$\epsilon$	$y(x)$
Randomly sample in the configuration space	$1 - \epsilon$	$1 - y(x)$



Sigmoid function

$$y(x) = \frac{1}{1 + e^{-x}}$$

How to adjust x automatically :



## Two major contributions

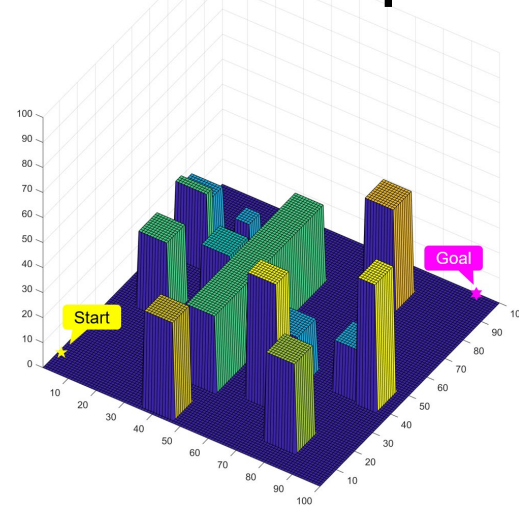
- Pruning-reconnecting mechanism
- Adaptive sampling

## Two groups of simulation

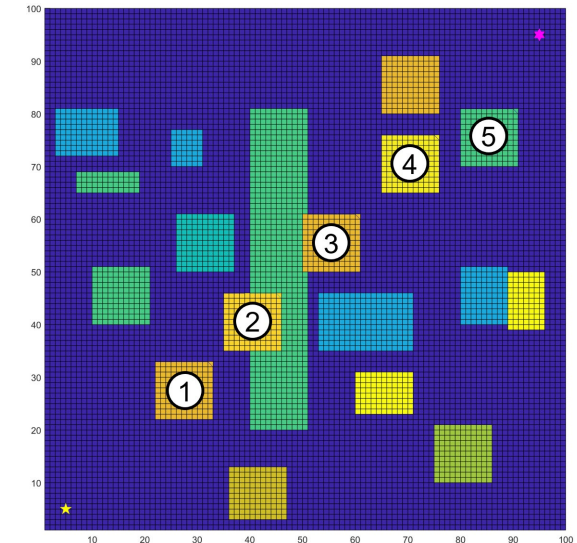
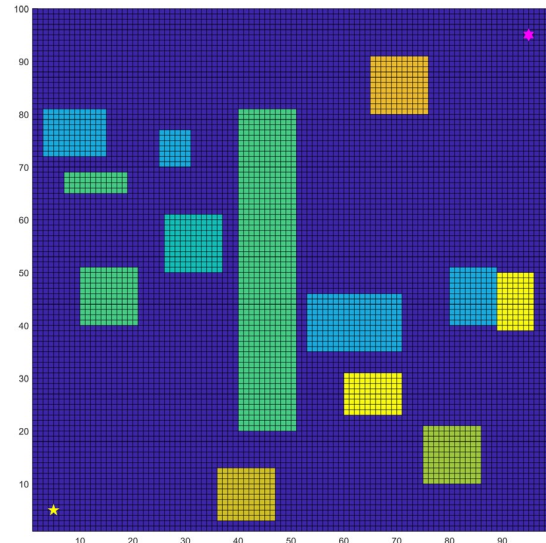
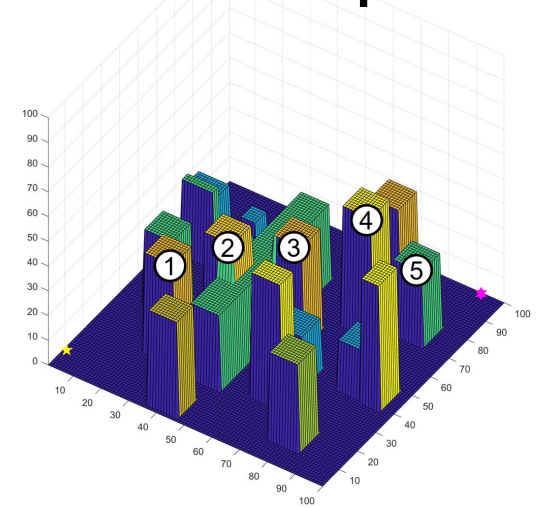
- Proposed vs. RRT\*D
- Adaptive heuristic vs. Traditional heuristic

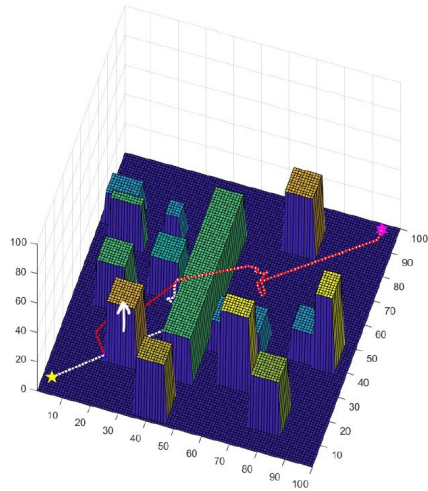
➤ **Dynamic map**

### Initial map

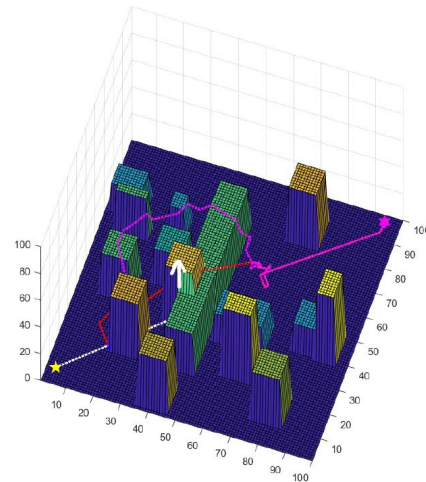


### Final map

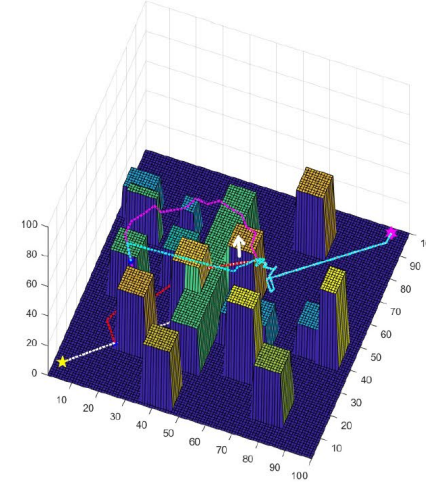




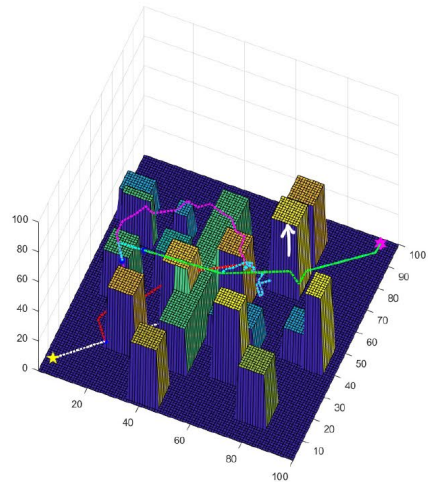
(a)



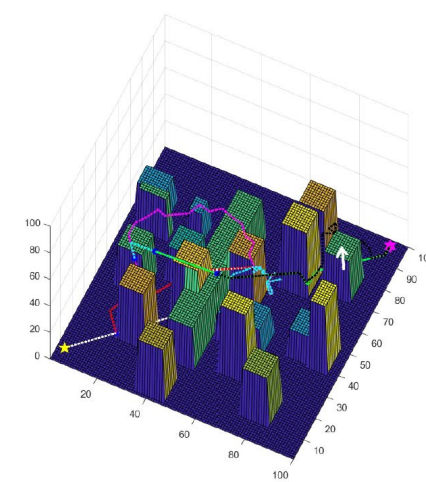
(b)



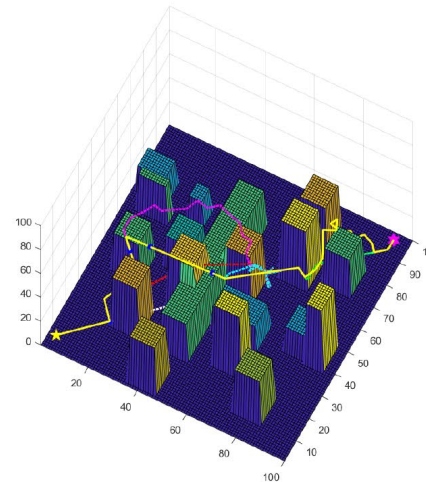
(c)



(d)

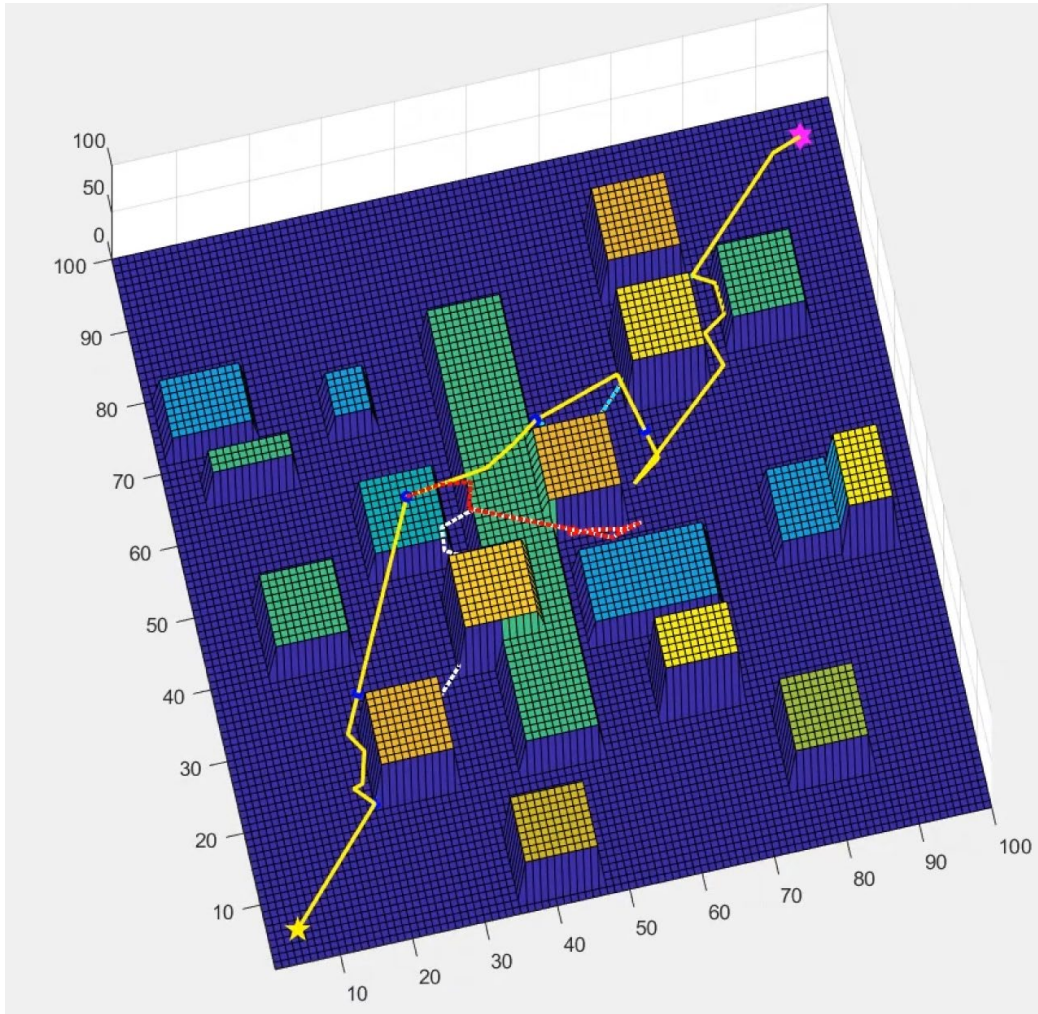


(e)



(f)





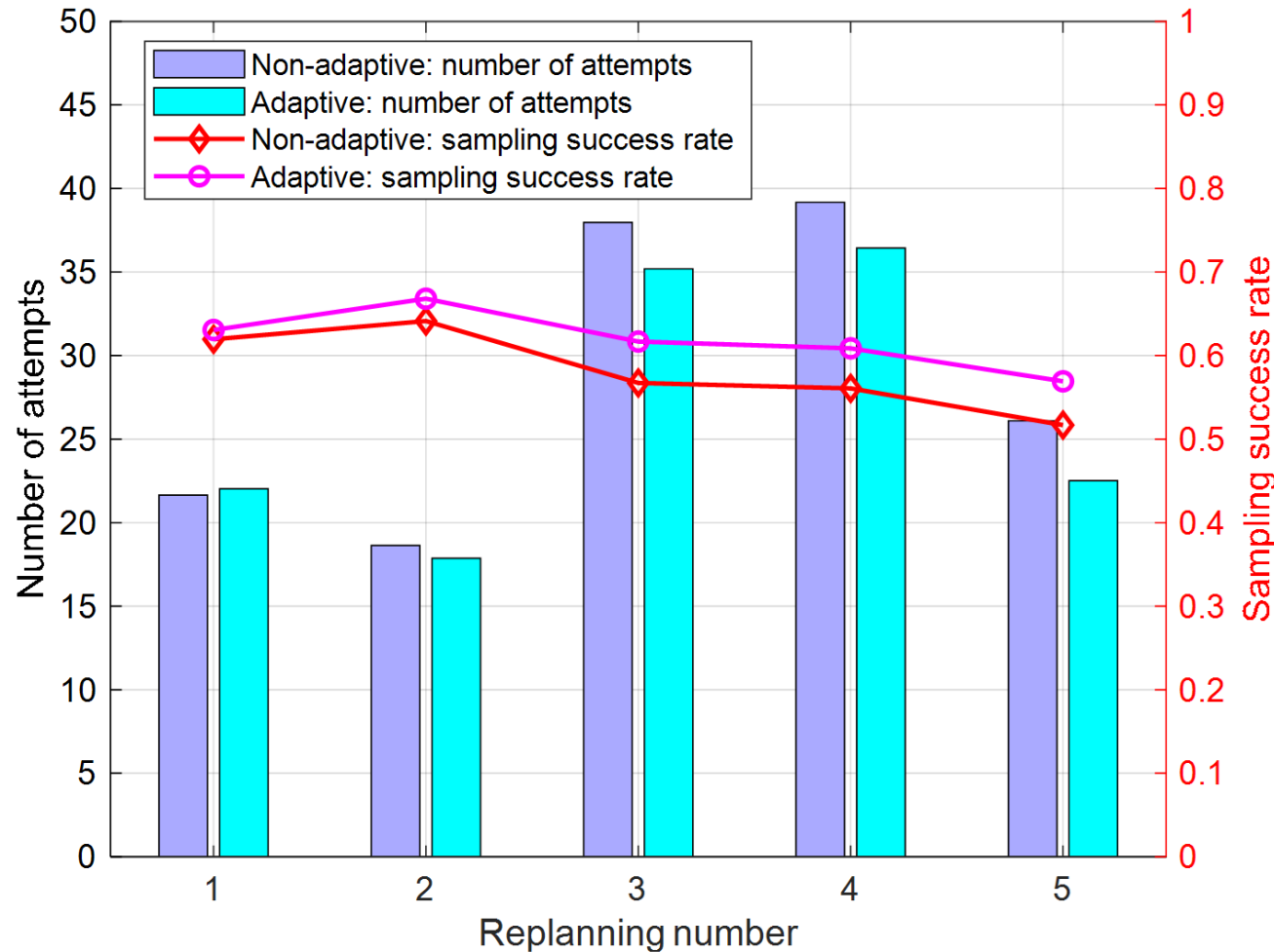
Number of nodes required in five times of re-planning

\*Average results from 1000 trials

Obstacle	RRT*D	ADRRT*-Connect
1	135	14
2	203	12
3	560	22
4	426	22
5	1086	13
Avg	482	17

ADRRT\*-Connect only requires **3.5%** new nodes to repair the path in re-planning.

## Adaptive heuristic vs. Traditional heuristic



One 'attempt' = once sampling

$$\text{Sampling success rate} = \frac{\text{Number of tree nodes}}{\text{Number of attempts}}$$

## Adaptively Dynamic RRT\*-Connect

Adaptive heuristic

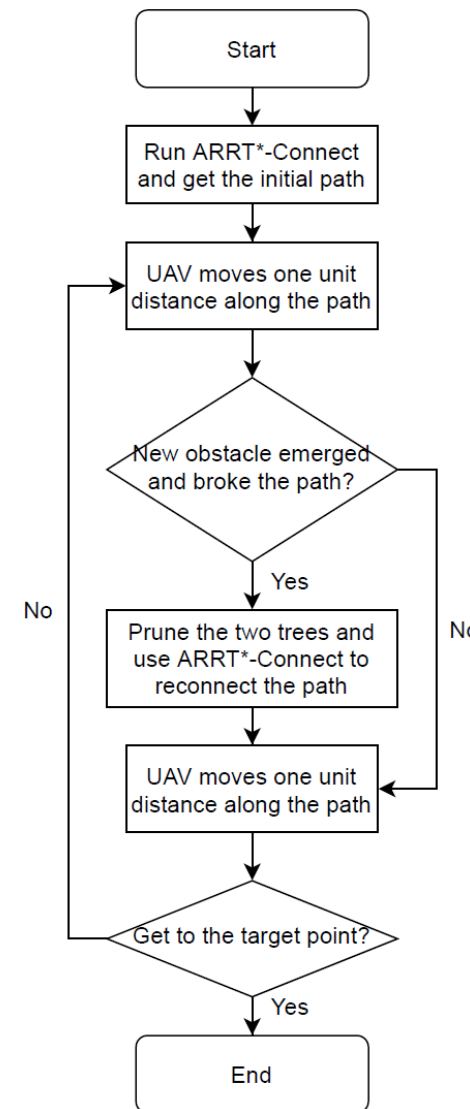
Re-planning

Asymptotically optimality

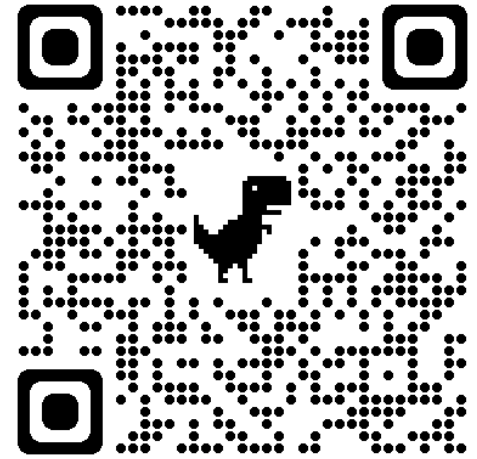
Bidirectional growth

### Possible future work

- More adaptive mechanisms in different stages of the planning process



# Thanks for your attention!



Yicheng Chen

<https://amos-chen98.github.io/>