Hierarchical planning

20194524

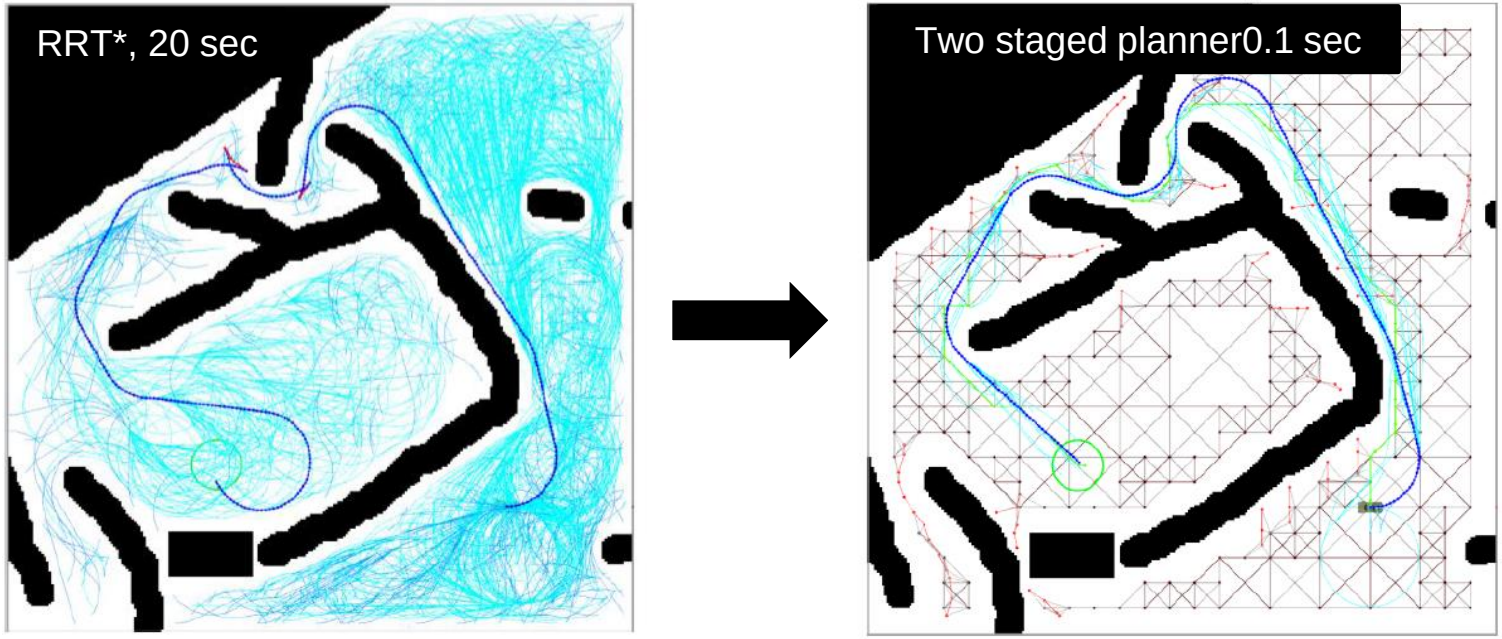
Jinhyeok Jang

A. Online Multilayered Motion Planning with Dynamic Constraints for Autonomous Underwater Vehicles, Eduard Vidal et el. ICRA2019

B. Hierarchical Path Planner using Workspace Decomposition and Parallel Task-Space RRTs, Geoge Mesesan et el. IROS2018

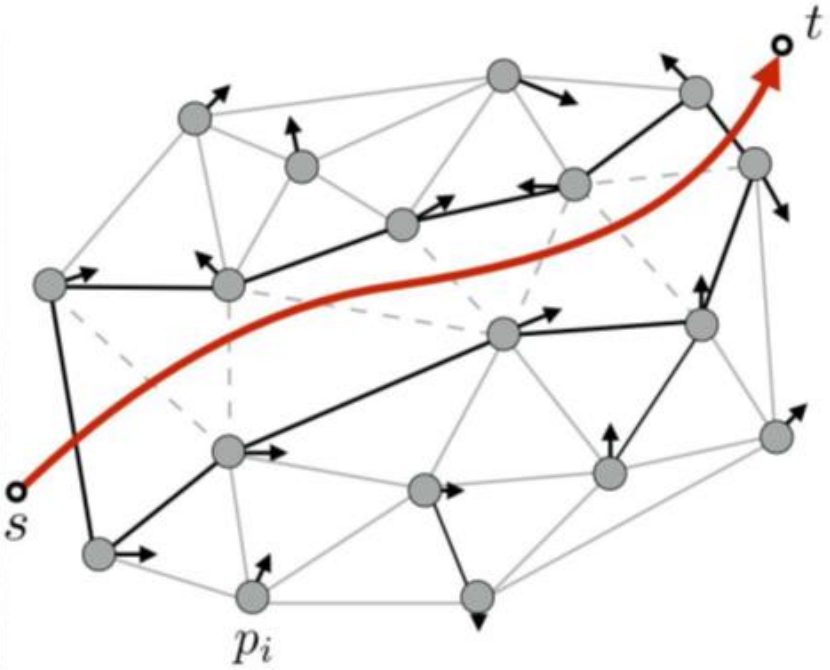
Review

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



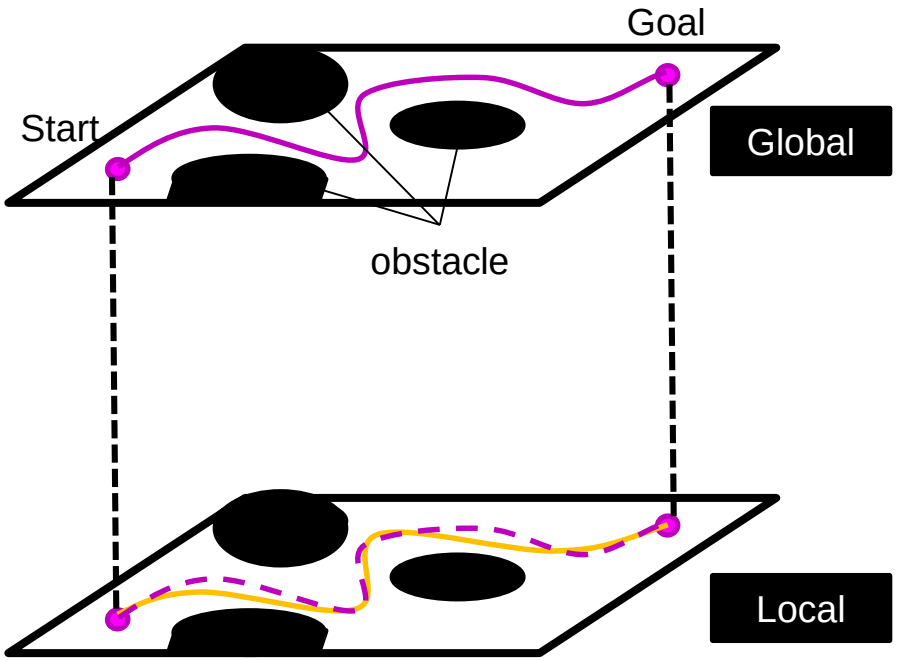
Review

- **Hierarchical planner**
  - Dynamic Channel: A Planning Framework for Crowd Navigation, ICRA 19

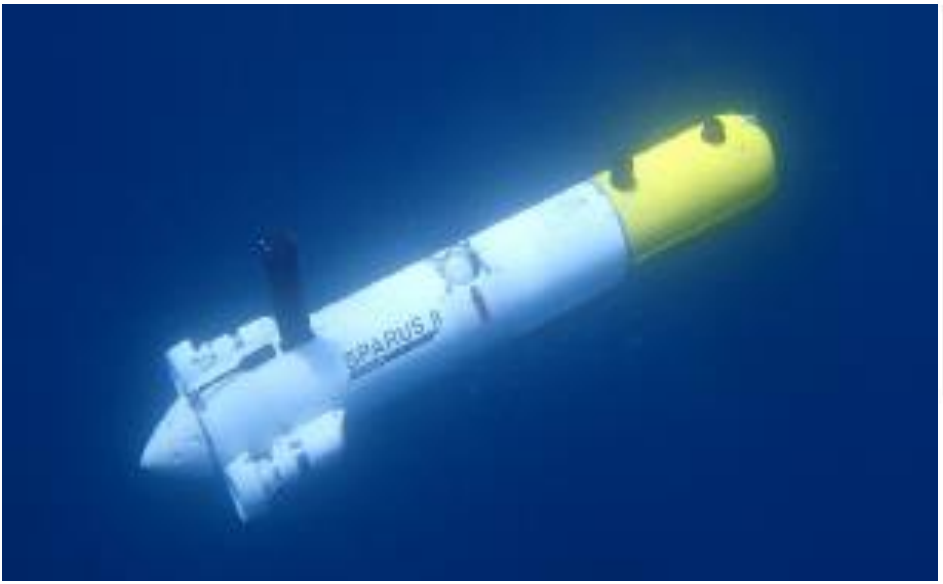


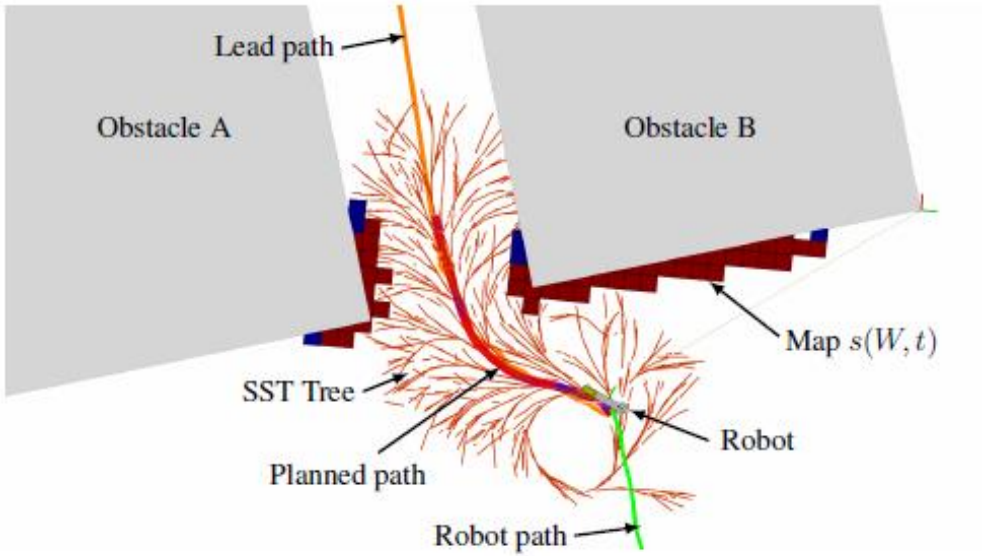
Background

- **Hierarchical planner**
  - Usually use two kind of planner
  - **Global planner**
    - Gives guidance
  - **Local planner**
    - Find the actual path



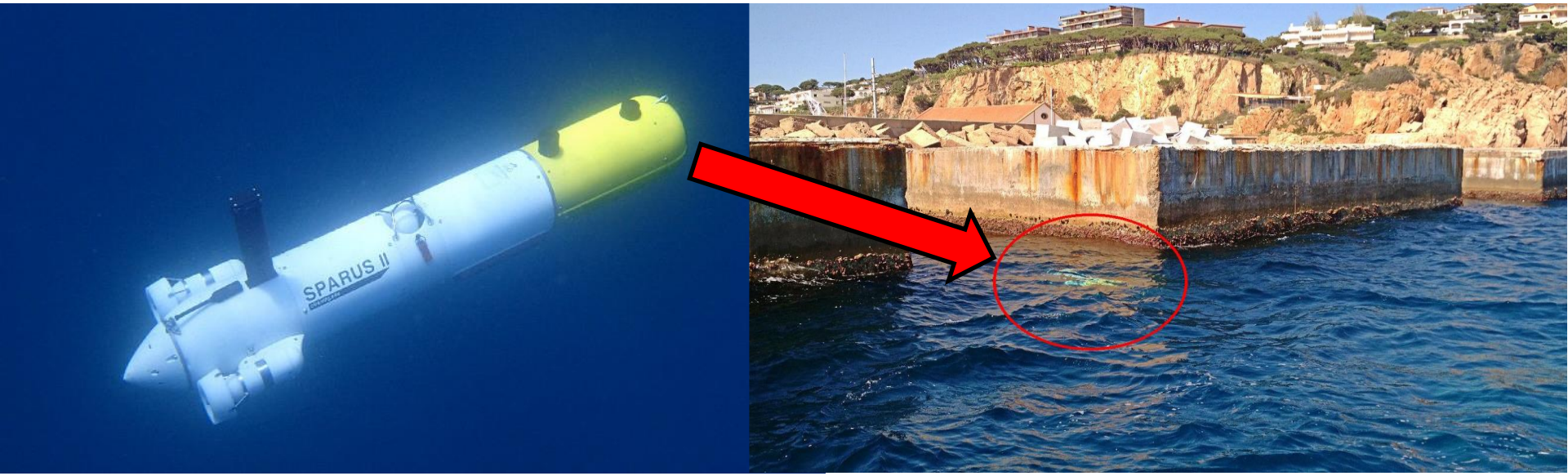
Online Multilayered Motion Planning with Dynamic Constraints for Autonomous Underwater Vehicles





Online Multilayered Motion Planning with Dynamic Constraints for Autonomous Underwater Vehicles

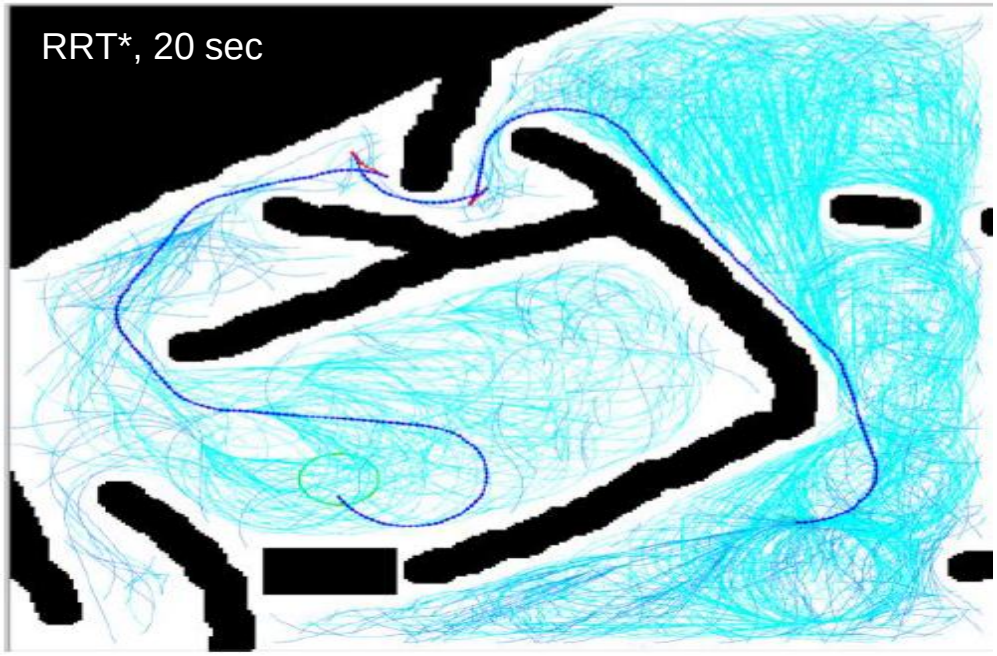
- **Introduction**
  - **Underwater vehicle**
    - Autonomous Underwater Vehicle – AUV
    - Complex dynamics



Online Multilayered Motion Planning with Dynamic Constraints for Autonomous Underwater Vehicles

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

- **Introduction**
- **Kinodynamic planning is required.**
  - Too slow for online planning.

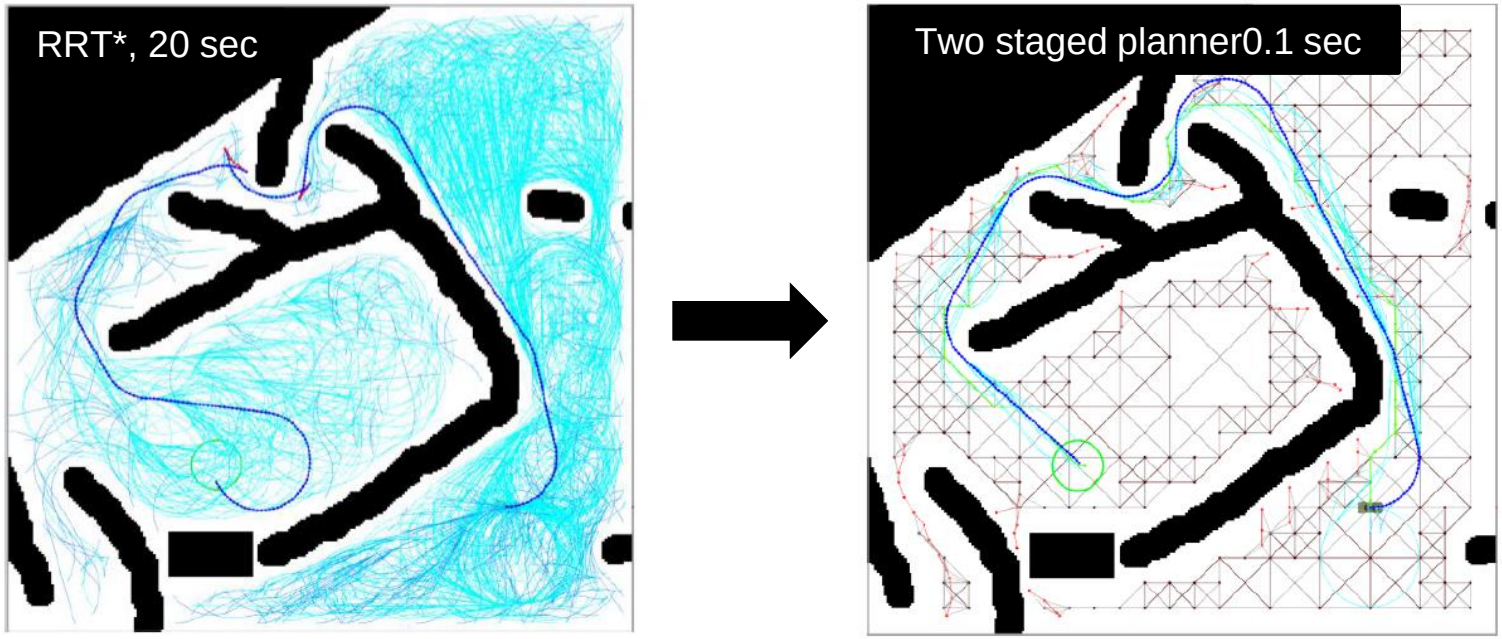


Online Multilayered Motion Planning with Dynamic Constraints for Autonomous Underwater Vehicles

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

Introduction

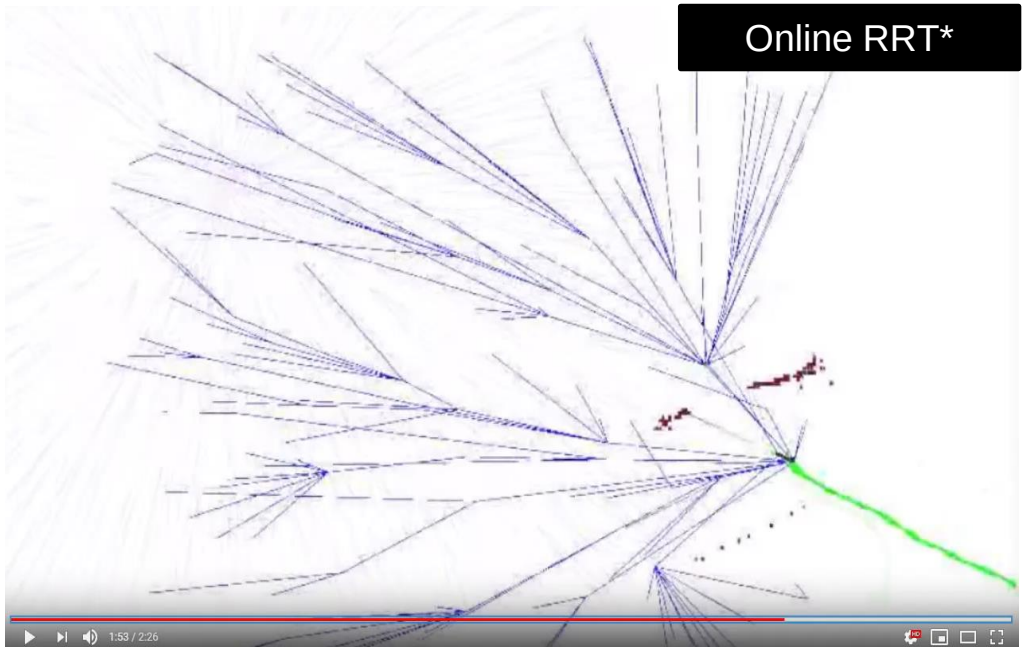
- **Kinodynamic planning is required.**
  - Too slow for online planning.
  - How about multilayered planning?



Online Multilayered Motion Planning with Dynamic Constraints for Autonomous Underwater Vehicles

Planning Feasible and Safe Paths Online for Autonomous Underwater Vehicles in Unknown Environments, IROS2016 Hernandez et el.

- **Two sampling planner**
  - **Online RRT* as global planner**
    - Online geometric path planning[1]
    - Can handle localization error
    - Finite horizon
    - Gives path fast
    - Asymptotically optimal
    - Sometimes gives infeasible path

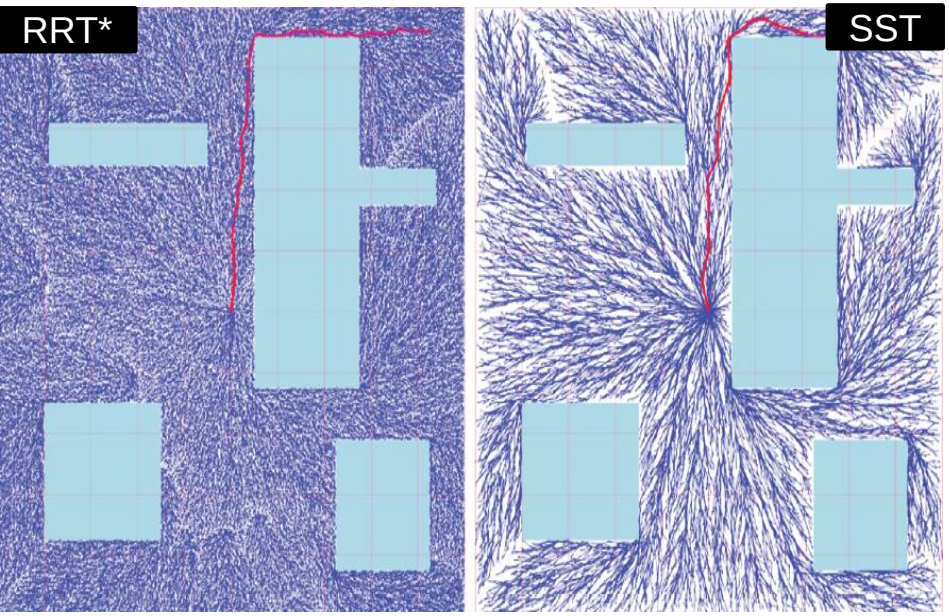


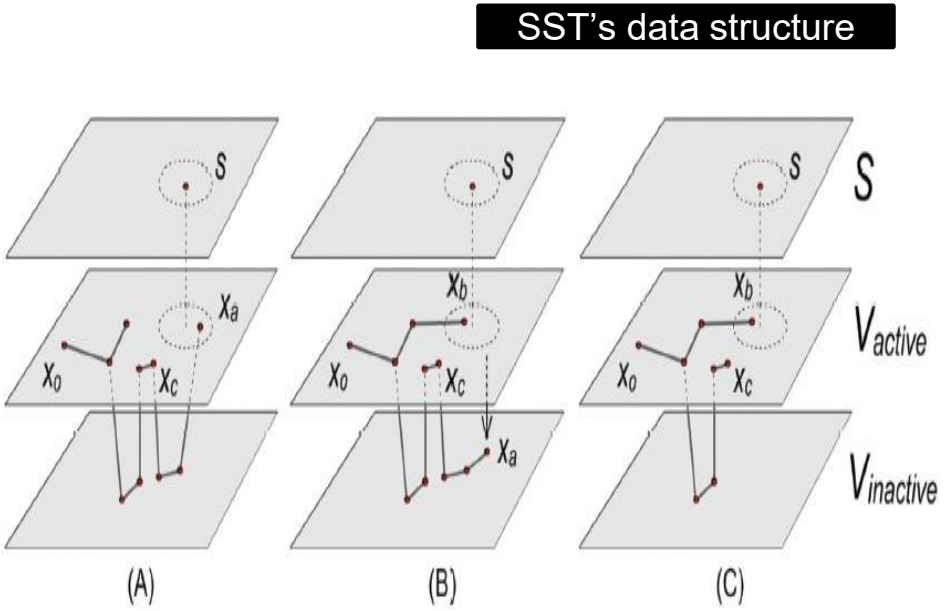
Online RRT*, https://www.youtube.com/watch?v=A5-8LTBxbHQ

Online Multilayered Motion Planning with Dynamic Constraints for Autonomous Underwater Vehicles

Online Multilayered Motion Planning with Dynamic Constraints for Autonomous Underwater Vehicles, ICRA2019 Vidal et el.

- **Two sampling planner**
- **SST as local planner**
  - Stable Sparse RRT(SST) as local planner





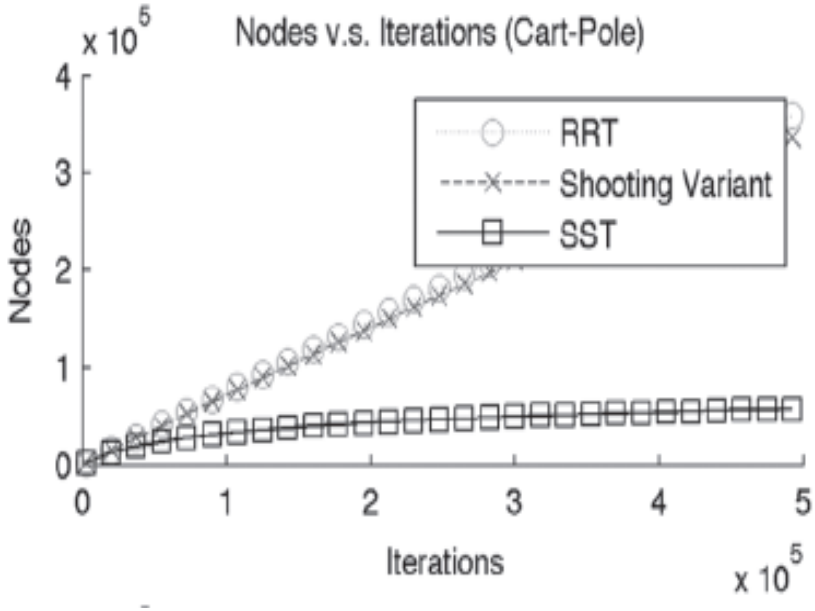
Asymptotically optimal sampling-based kinodynamic planning, Li et el. IJRR 2016

Online Multilayered Motion Planning with Dynamic Constraints for Autonomous Underwater Vehicles

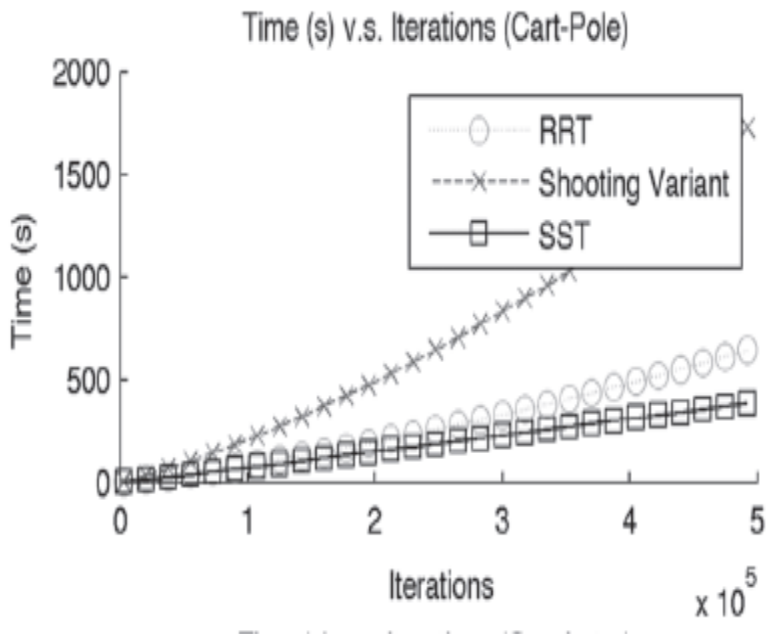
Online Multilayered Motion Planning with Dynamic Constraints for Autonomous Underwater Vehicles, ICRA2019 Vidal et el.

- **Two sampling planner**
- **SST as local planner**
  - RRT* + SST

Total nodes, RRT vs SST



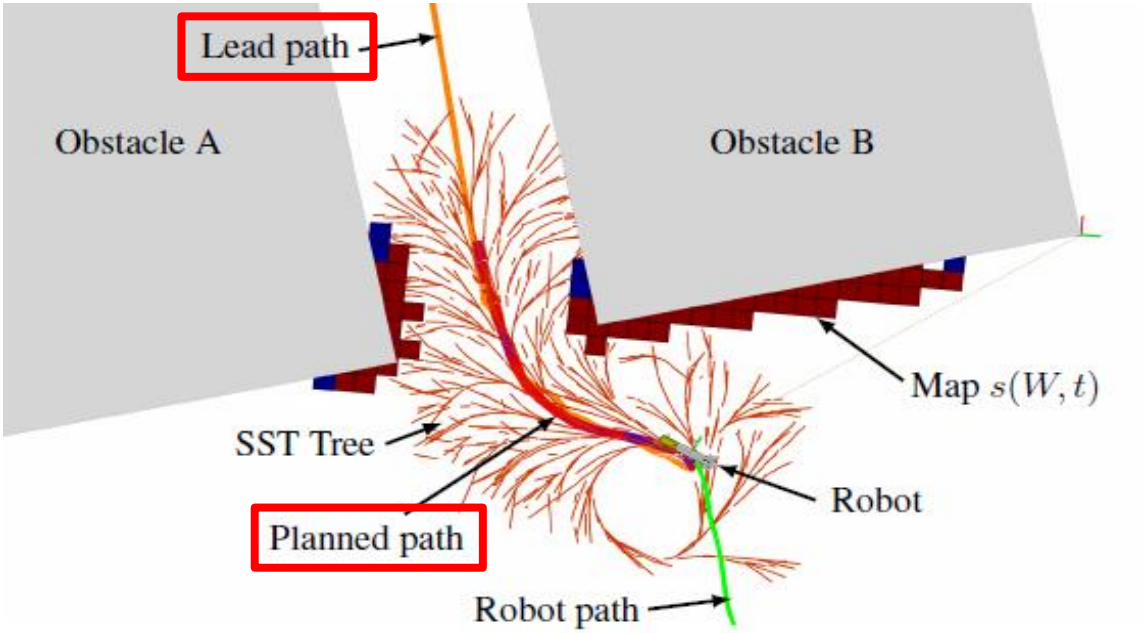
Planning time, RRT vs SST



Asymptotically optimal sampling-based kinodynamic planning, Li et el. IJRR 2016

Online Multilayered Motion Planning with Dynamic Constraints for Autonomous Underwater Vehicles

- **Two sampling planner**
  - **Using two planner**
    - Global RRT* + Local SST

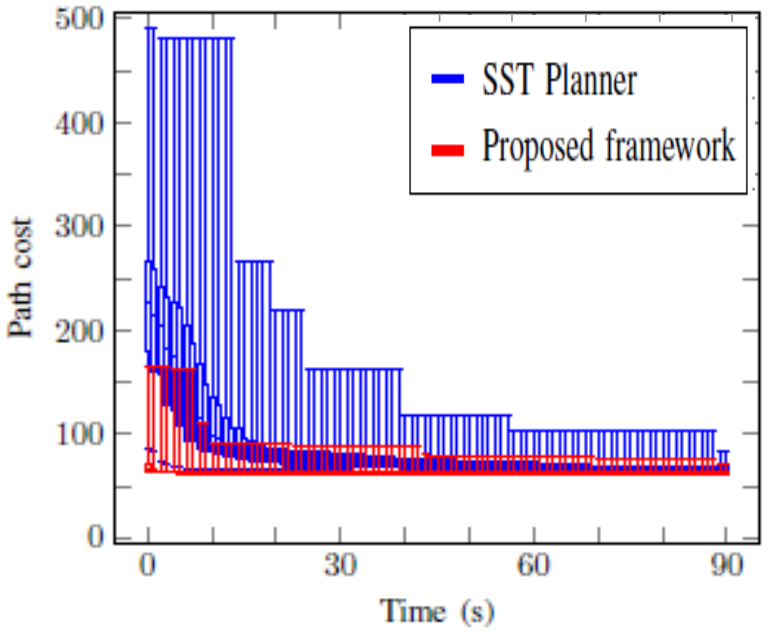


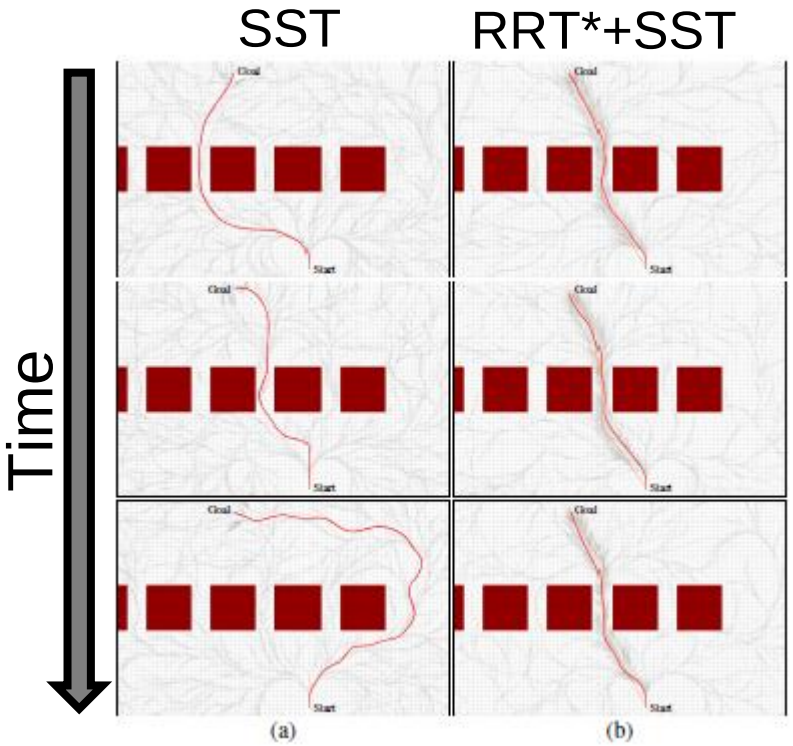
Asymptotically optimal sampling-based kinodynamic planning, Li et el. IJRR 2016

Online Multilayered Motion Planning with Dynamic Constraints for Autonomous Underwater Vehicles

Online Multilayered Motion Planning with Dynamic Constraints for Autonomous Underwater Vehicles, ICRA2019 Vidal et el.

- **Result**
  - **Single SST vs RRT*+SST**

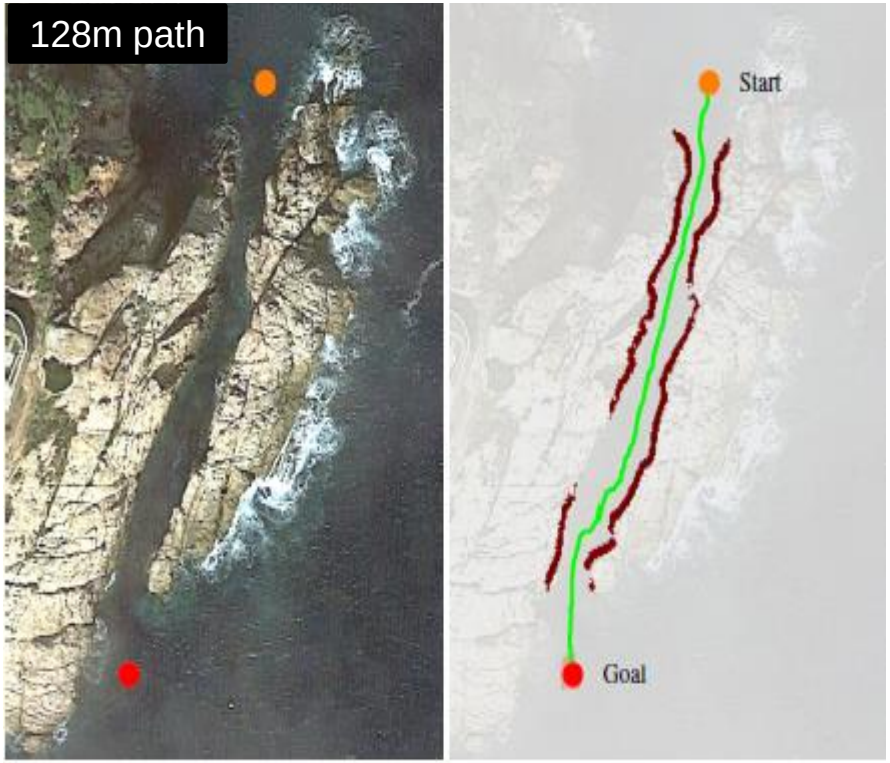


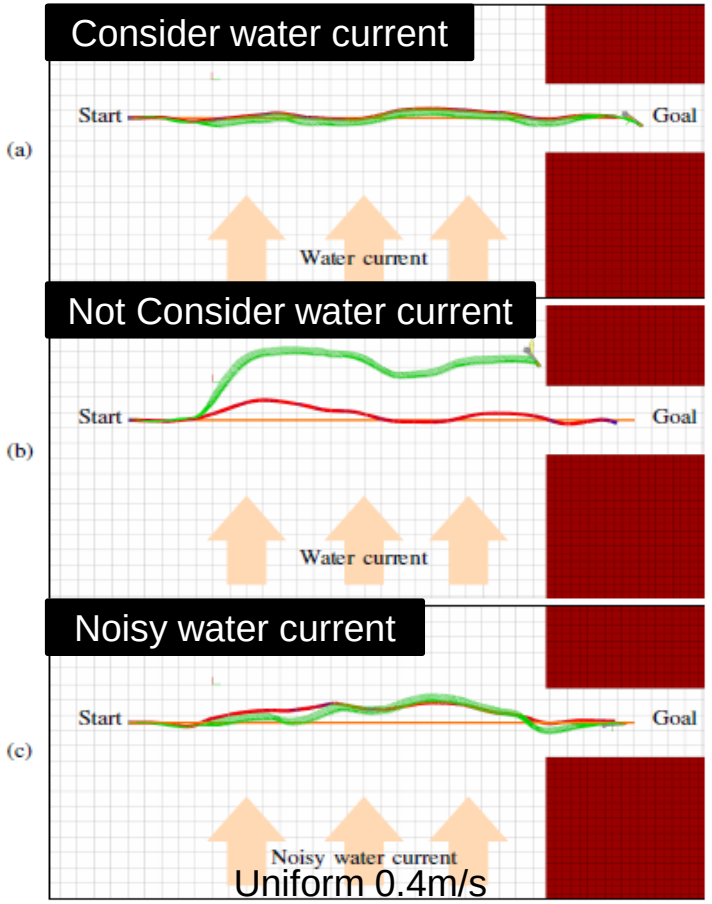


Online Multilayered Motion Planning with Dynamic Constraints for Autonomous Underwater Vehicles

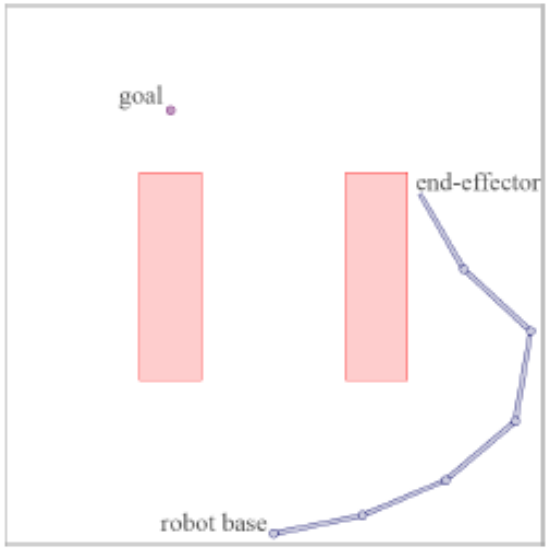
Online Multilayered Motion Planning with Dynamic Constraints for Autonomous Underwater Vehicles, ICRA2019 Vidal et el.

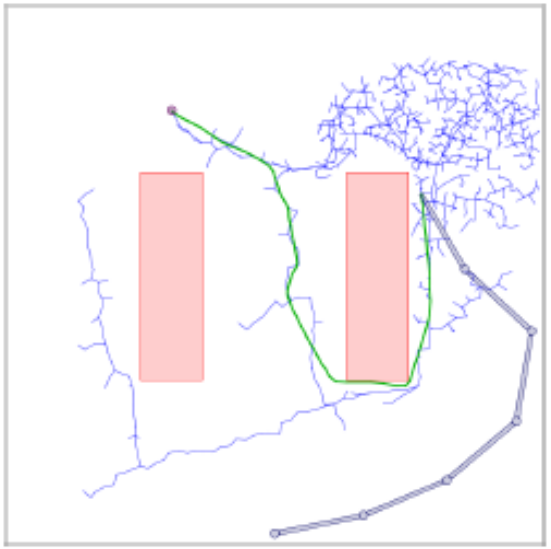
- **Result**
  - **Real world Test**

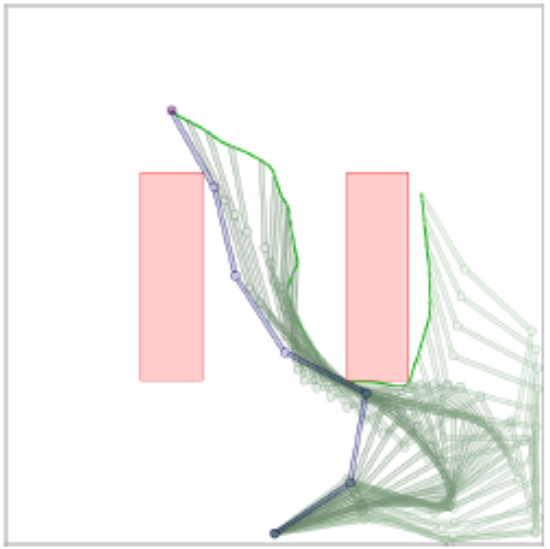




Hierarchical Path Planner using Workspace Decomposition and Parallel Task-Space RRTs





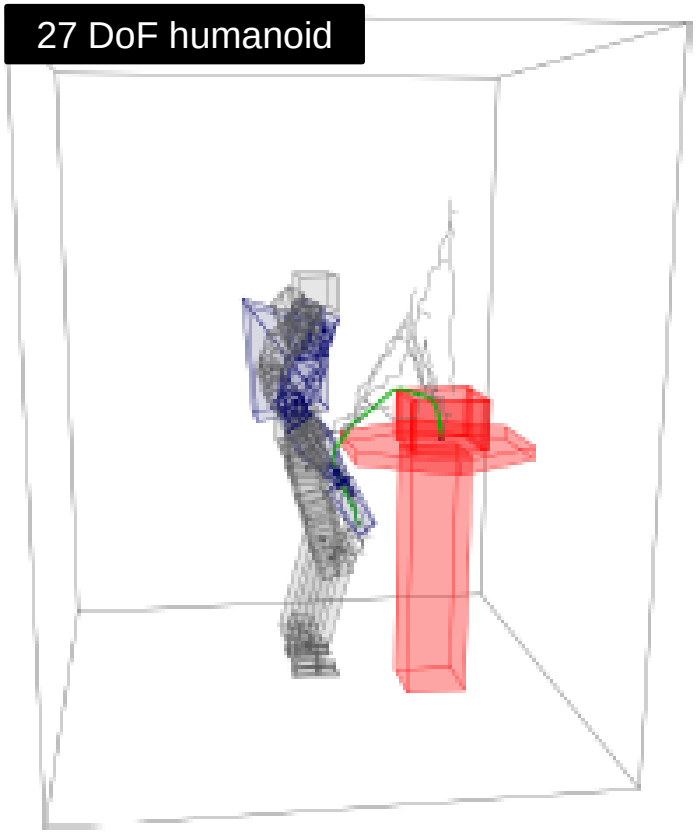


Hierarchical Path Planner using Workspace Decomposition and Parallel Task-Space RRTs

Hierarchical Path Planner using Workspace Decomposition and Parallel Task-Space RRTs
, IROS2018 Geoge Mesesan et el.

Introduction

- **Real world robots**
  - Have very high dimensionality
  - Unavailable for C-space approach
  - Humanoids…

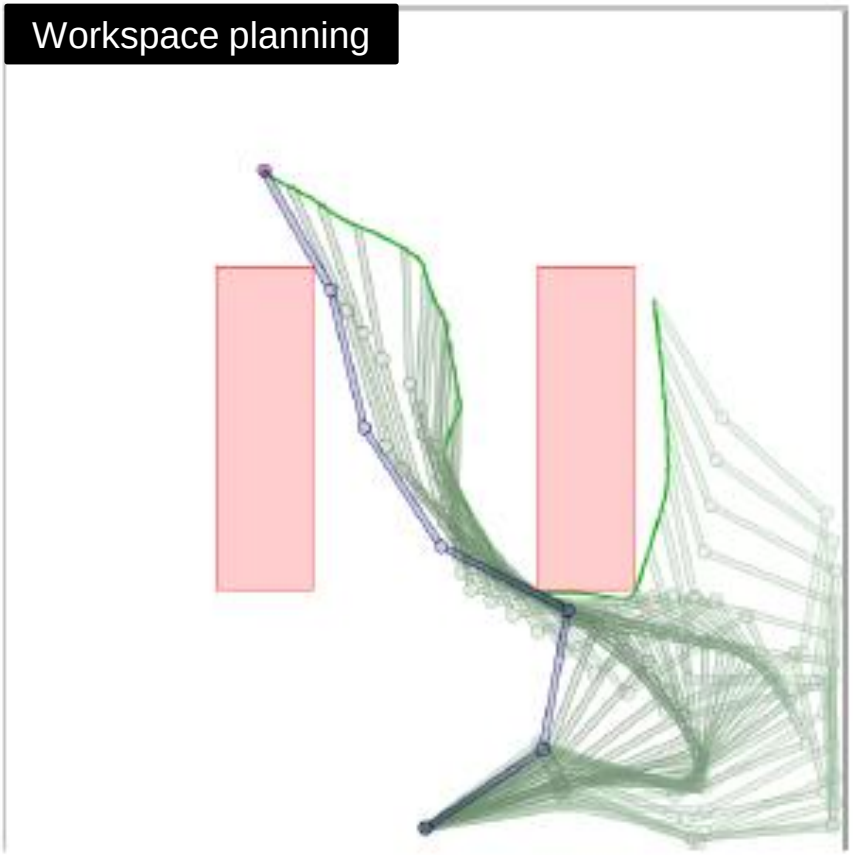


Hierarchical Path Planner using Workspace Decomposition and Parallel Task-Space RRTs

Hierarchical Path Planner using Workspace Decomposition and Parallel Task-Space RRTs
, IROS2018 Geoge Mesesan et el.

Introduction

- **Handling high dimensionality**
  - Probabilistic techniques
  - Task space
  - End- effector's position, orientation, …



Hierarchical Path Planner using Workspace Decomposition and Parallel Task-Space RRTs

- **Introduction**

- **Extract workspace information**
  - Cell decompositionETC.

- **Parallel method**
  - Another way of improving path planning algorithm.
  - Running multiple planners at the same time

Hierarchical Path Planner using Workspace Decomposition and Parallel Task-Space RRTs

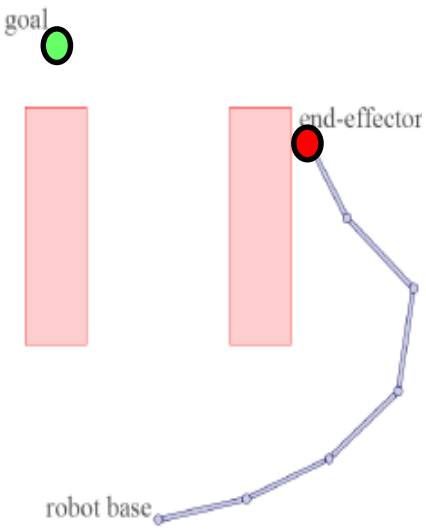
Hierarchical Path Planner using Workspace Decomposition and Parallel Task-Space RRTs
, IROS2018 Geoge Mesesan et al.

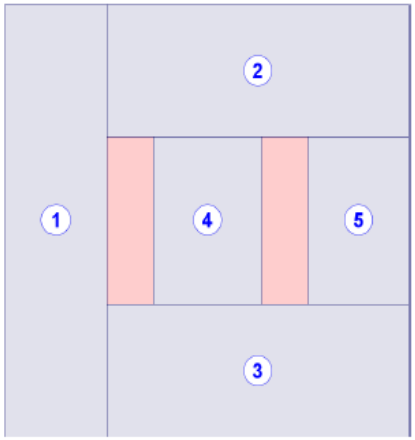
- **Multilayered planner**

- **Global planner**
  - Cell decomposition
  - Find collision free path.

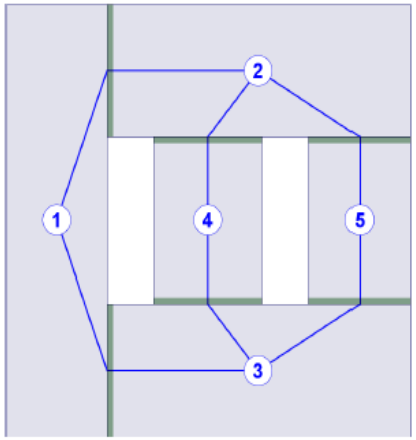
Initial position

Goal

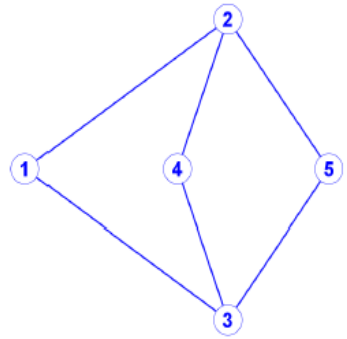




(a) Free-space partitioning (gray: free space, red: obstacle)



(b) Boundaries between adjacent polytopes (green)

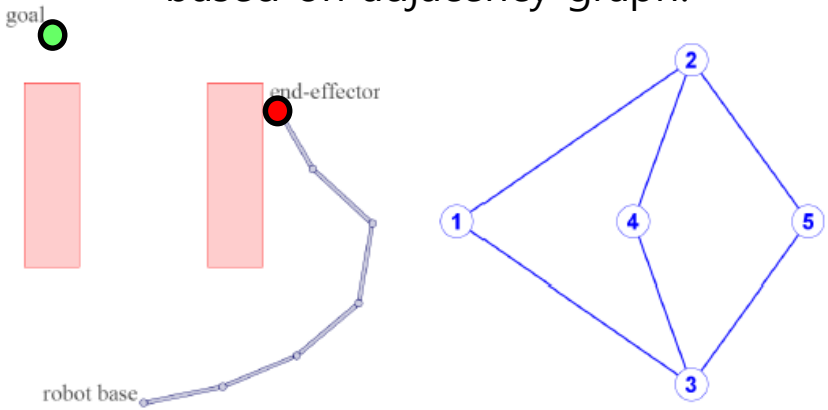


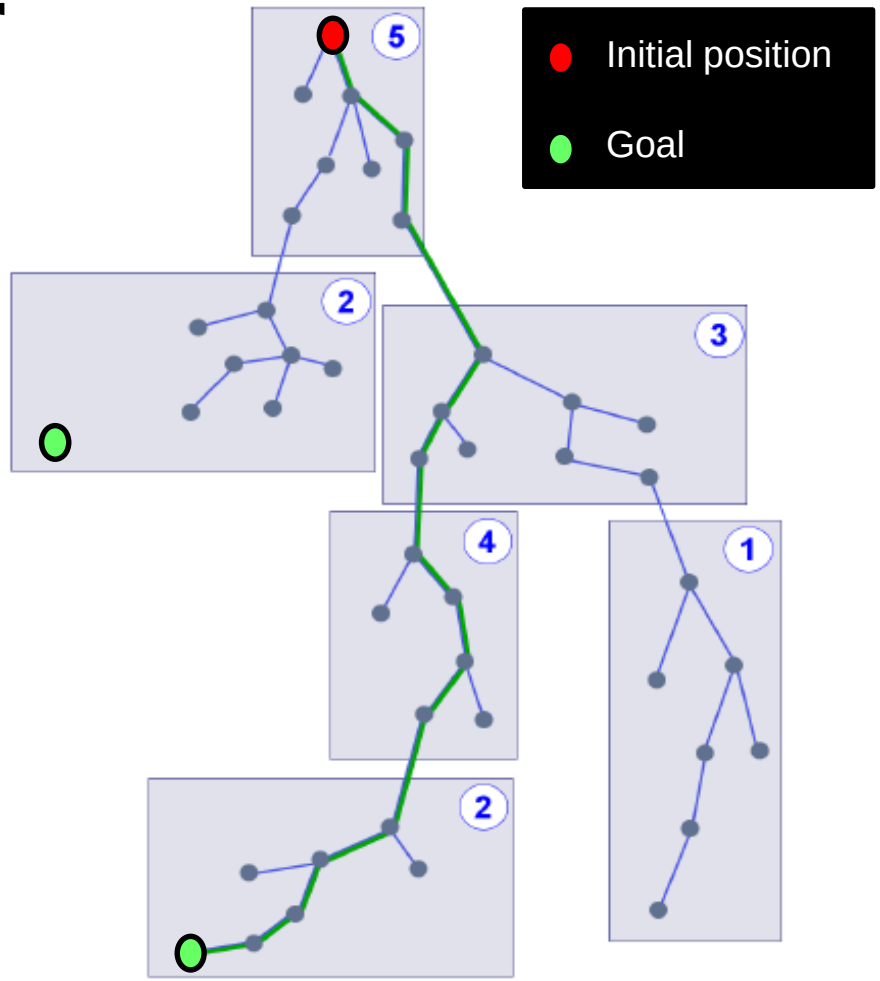
(c) Adjacency graph

Hierarchical Path Planner using Workspace Decomposition and Parallel Task-Space RRTs

Hierarchical Path Planner using Workspace Decomposition and Parallel Task-Space RRTs
, IROS2018 Geoge Mesesan et al.

- **Multilayered planner**
  - **Local planner**
    - Task space RRT
    - Each polytope's planner runs parallel.
    - Expand to adjacent polytope based on adjacency graph.

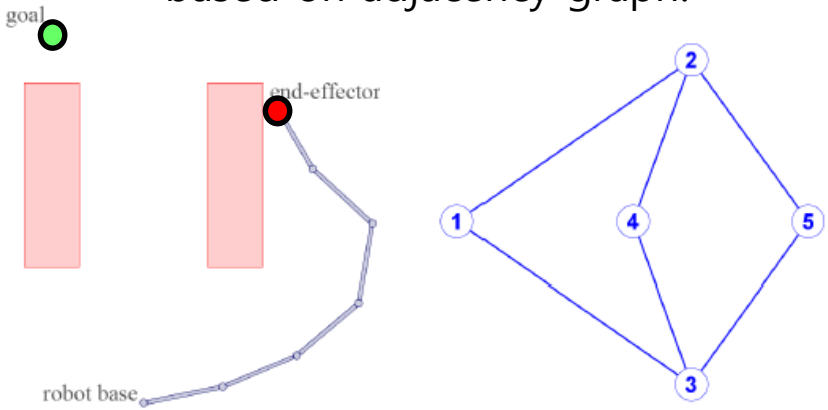


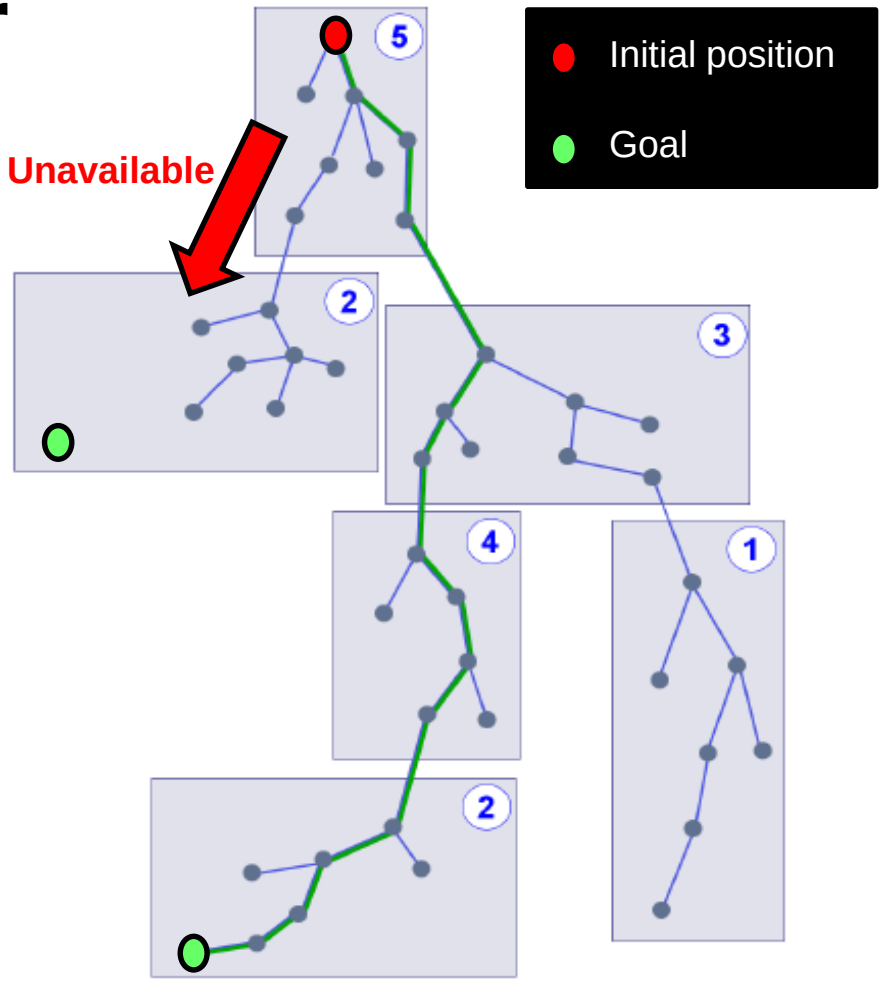


Hierarchical Path Planner using Workspace Decomposition and Parallel Task-Space RRTs

Hierarchical Path Planner using Workspace Decomposition and Parallel Task-Space RRTs
, IROS2018 Geoge Mesesan et el.

- **Multilayered planner**
  - **Local planner**
    - Task space RRT
    - Each polytope's planner runs parallel.
    - Expand to adjacent polytope based on adjacency graph.

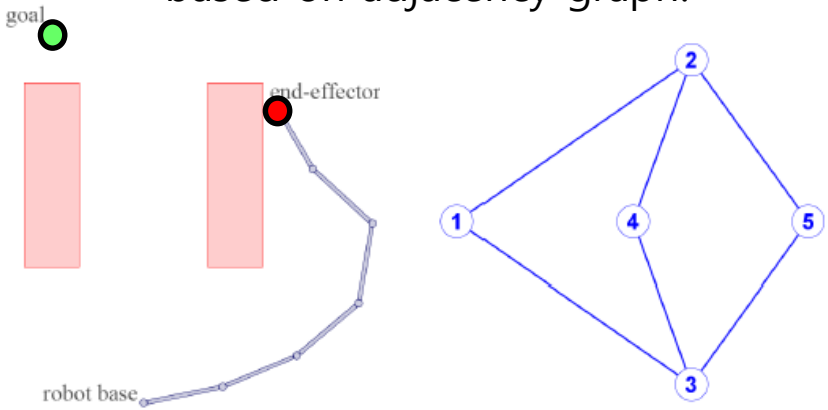


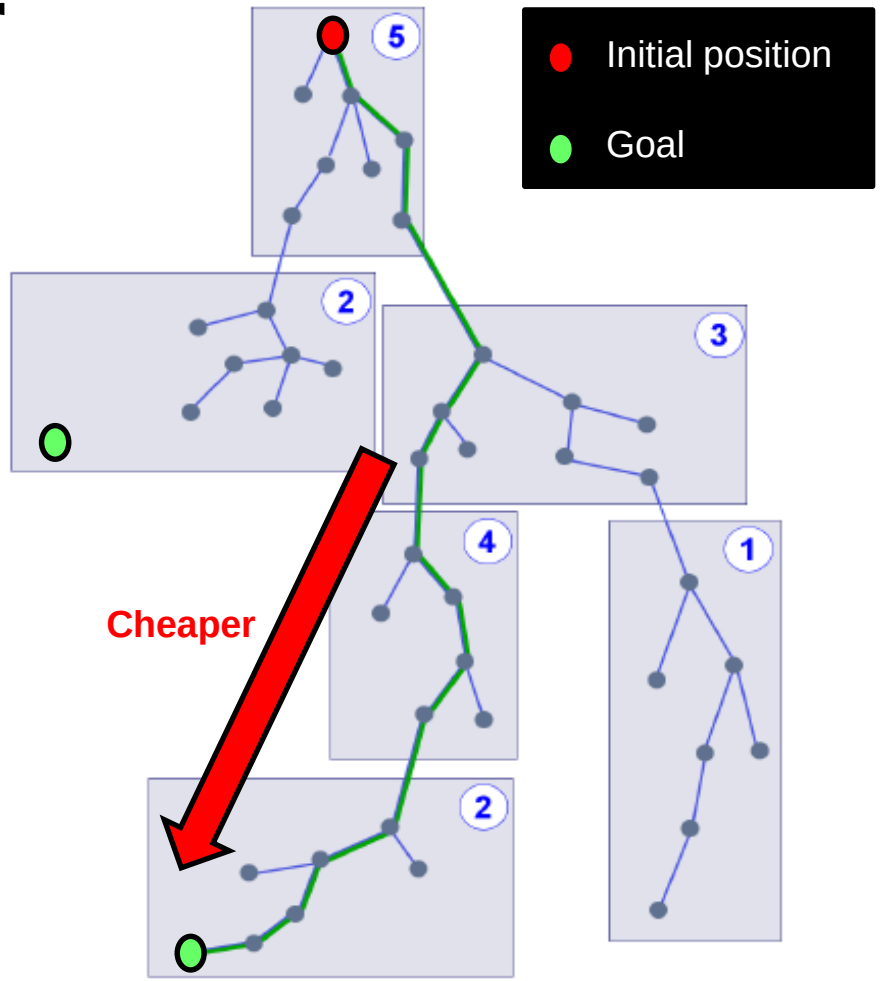


Hierarchical Path Planner using Workspace Decomposition and Parallel Task-Space RRTs

Hierarchical Path Planner using Workspace Decomposition and Parallel Task-Space RRTs
, IROS2018 Geoge Mesesan et el.

- **Multilayered planner**
  - **Local planner**
    - Task space RRT
    - Each polytope's planner runs parallel.
    - Expand to adjacent polytope based on adjacency graph.





Hierarchical Path Planner using Workspace Decomposition and Parallel Task-Space RRTs

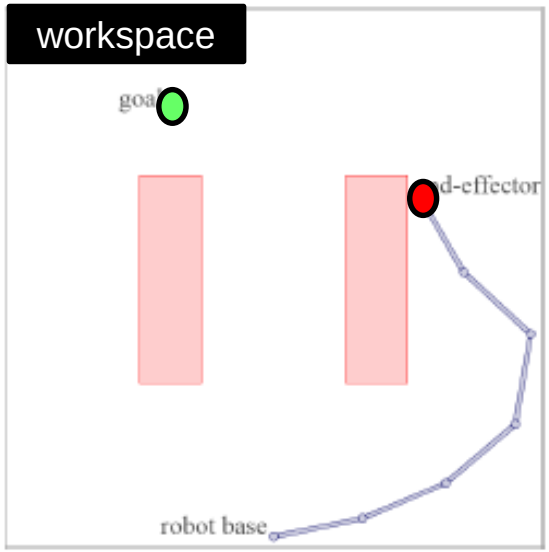
Hierarchical Path Planner using Workspace Decomposition and Parallel Task-Space RRTs
, IROS2018 Geoge Mesesan et al.

- **Multilayered planner**
  - **Final path**
    - Global path connects the local planner's path in workspace.

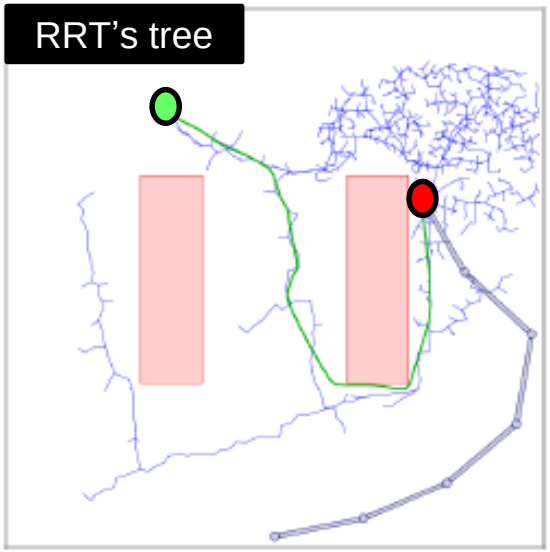
Initial position

Goal

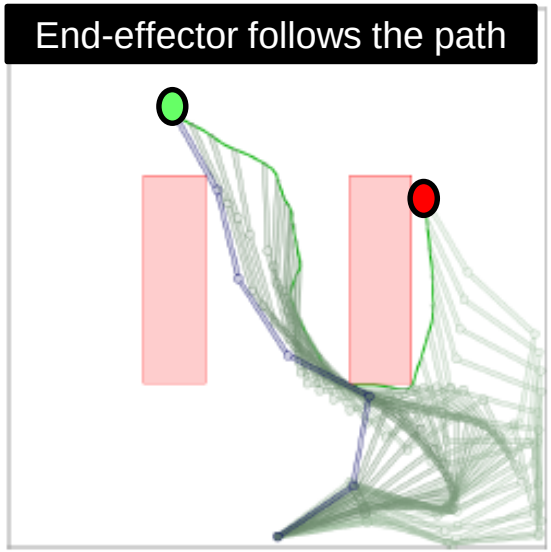
workspace



RRT's tree



End-effector follows the path



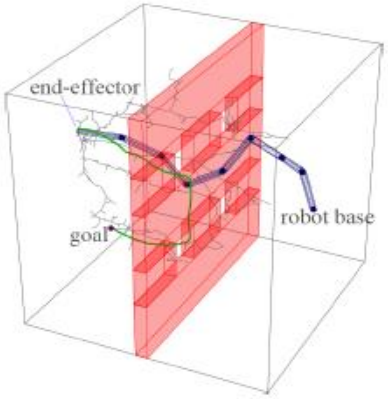
Hierarchical Path Planner using Workspace Decomposition and Parallel Task-Space RRTs

Hierarchical Path Planner using Workspace Decomposition and Parallel Task-Space RRTs
, IROS2018 Geoge Mesesan et al.

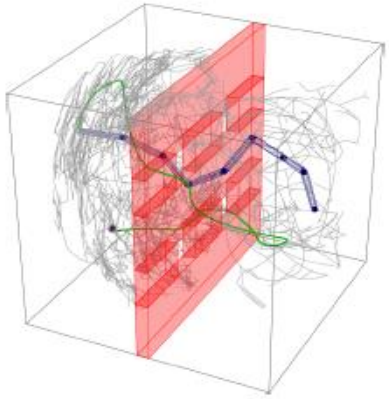
- **Comparison**
  - **8 DoF redundant robot**

l_q C-space path length

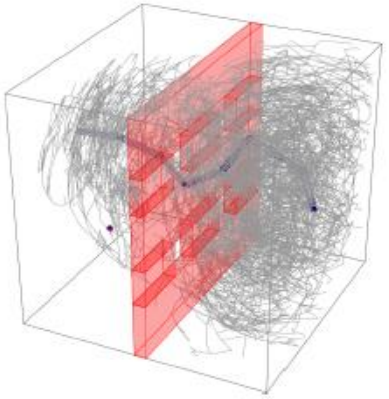
l_p end-effector path length



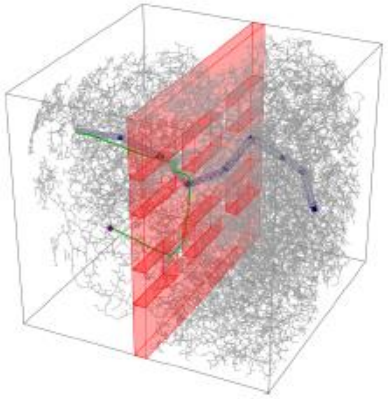
(a) Hierarchical Planner



(b) ADD-RRT



(c) RRT-Connect



(d) TS-RRT

Planner	Time (s)	Collision Checks	l_q	l_p (m)	Success (%)
Hierachical Planner	1.39 ± 0.40	7675 ± 5049	5.147 ± 0.181	1.304 ± 0.036	100
ADD-RRT	2.42 ± 0.79	18809 ± 10117	13.986 ± 3.631	2.699 ± 0.760	100
BiRRT-Connect	2.64 ± 0.94	21637 ± 14349	13.937 ± 3.981	2.664 ± 0.805	100
RRT-Connect	N/A	N/A	N/A	N/A	0
TS-RRT	6.33 ± 4.84	147262 ± 121840	4.706 ± 0.125	1.637 ± 0.079	41

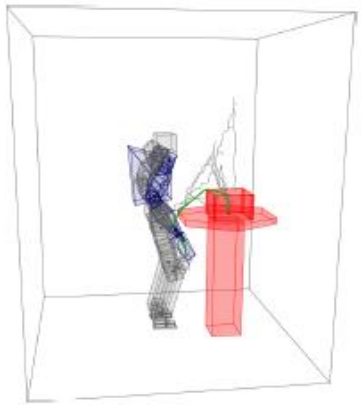
Hierarchical Path Planner using Workspace Decomposition and Parallel Task-Space RRTs

Hierarchical Path Planner using Workspace Decomposition and Parallel Task-Space RRTs
, IROS2018 Geoge Mesesan et al.

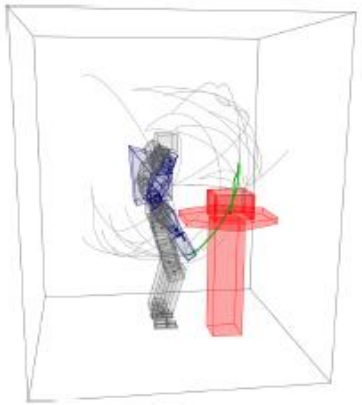
- **Comparison**
  - **9 DoF Humanoid reaching to the box**

l_q C-space path length

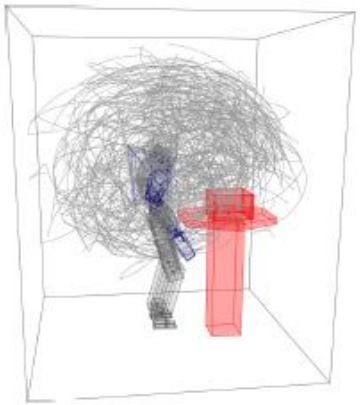
l_p end-effector path length



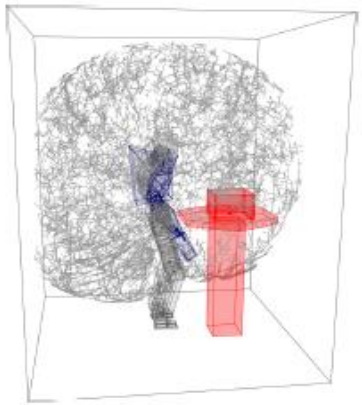
(a) Hierarchical Planner



(b) ADD-RRT



(c) RRT-Connect



(d) TS-RRT

Planner	Time (s)	Collision Checks	l_q	l_p (m)	Success (%)
Hierachical Planner	0.74 ± 0.06	1304 ± 199	2.732 ± 0.185	1.090 ± 0.068	100
ADD-RRT	0.71 ± 0.21	1801 ± 1281	3.411 ± 0.676	1.207 ± 0.173	100
BiRRT-Connect	0.70 ± 0.19	1838 ± 1064	3.359 ± 0.589	1.261 ± 0.190	100
RRT-Connect	2.72 ± 4.41	21517 ± 42611	3.249 ± 0.453	1.334 ± 0.213	9
TS-RRT	N/A	N/A	N/A	N/A	0

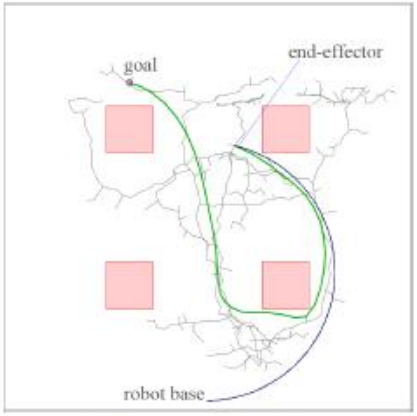
Hierarchical Path Planner using Workspace Decomposition and Parallel Task-Space RRTs

Hierarchical Path Planner using Workspace Decomposition and Parallel Task-Space RRTs
, IROS2018 Geoge Mesesan et al.

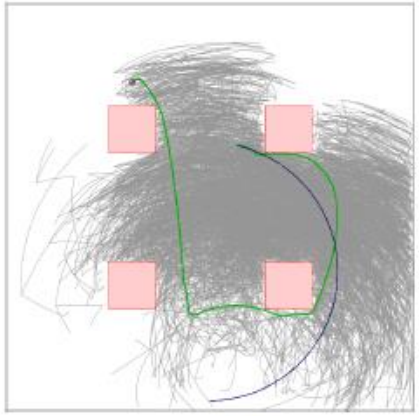
- **Comparison**
  - **100 DoF hyper robot**

l_q C-space path length

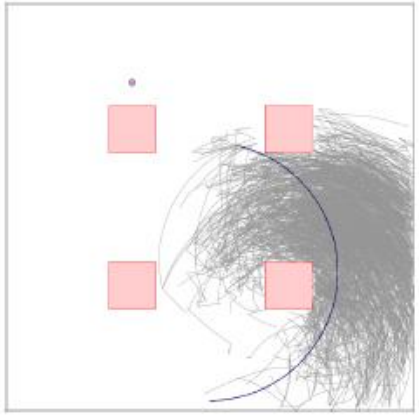
l_p end-effector path length



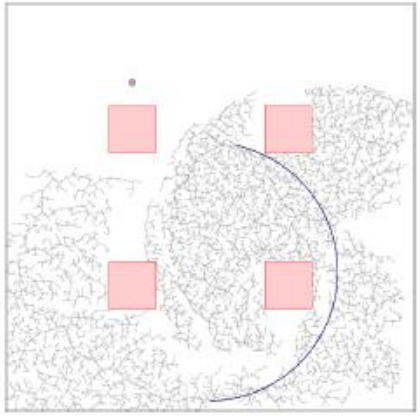
(a) Hierarchical Planner



(b) ADD-RRT



(c) RRT-Connect



(d) TS-RRT

Planner	Time (s)	Collision Checks	l_q	l_p (m)	Success (%)
Hierachical Planner	4.70 ± 1.25	34281 ± 11926	3.498 ± 0.221	1.830 ± 0.064	100
ADD-RRT	18.31 ± 7.37	62625 ± 18305	14.197 ± 1.551	2.197 ± 0.256	49
BiRRT-Connect	18.69 ± 8.04	62874 ± 21273	14.710 ± 1.943	2.296 ± 0.297	37
RRT-Connect	N/A	N/A	N/A	N/A	0
TS-RRT	N/A	N/A	N/A	N/A	0

Hierarchical Path Planner using Workspace Decomposition and Parallel Task-Space RRTs

Comparison

Result

- Suggested model always success 100% even for extreme case.
- Shows shortest path both l_q and l_p .
- Even single tree, matches/outperforms bi-directional planner.

Quiz

- **Q1.**
  - In first paper, authors use **Three stage** of planning **[True/False]**

- **Q2.**
  - In second paper, authors use **Voronoi diagram** for global planning **[True/False]**