Efficient Configuration Space Representation in Path Planning

Akhil Avula, Aidan Cookson, Calvin Chang, Daniel Truong
aavula@g.ucla.edu, aucookson@g.ucla.edu, calvinchang33@g.ucla.edu, dktruong@g.ucla.edu
UCLA ECE 209AS Fall 2020

Overview

- Introduction to Path Planning
- Previous Research
- Shortcomings of Research
- Problem Statement
- Procedural Map Generation
- Quadtrees Design Process
- Implementation
- Expected Results
- Conclusion and Future Works

Path Planning Problem

- Automated traveling from point A to point B
- Feasible and Optimal Path
- Assume holonomic robot

Configuration space: C

Obstacles: $C_{obs} \in C$

Feasible robot states: $C_{feasible} \in C$

$C_{feasible} \cup C_{obs} = C$

$C_{feasible} \cap C_{obs} = \emptyset$

Starting state: $s_i \epsilon C_{feasible}$

Goal state: $s_f \epsilon C_{feasible}$

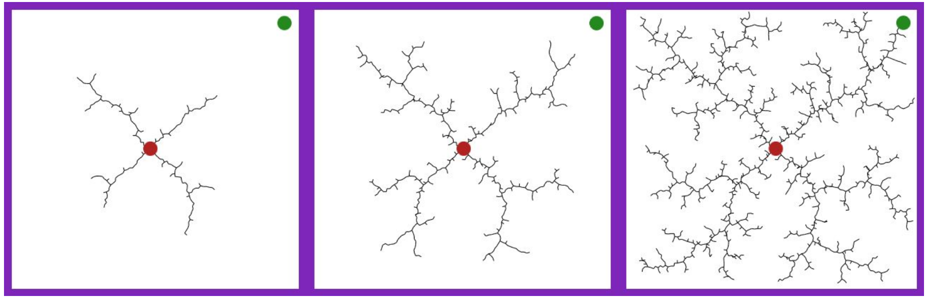
Taken from National Stone, Sand, and Gravel Association [8]

Taken from Wikipedia: user Dtom [9]

How to represent configuration space?

Previous Research

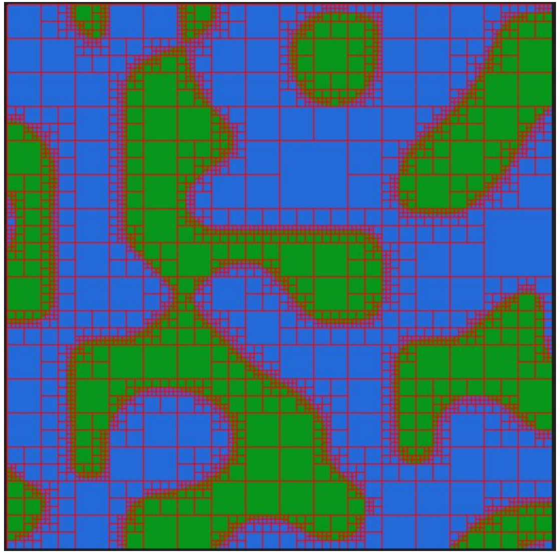
- Mapping configuration space into graph
  - Grid
  - Probabilistic Roadmap[1]
  - Quadtrees[2]
  - Visibility Graph[3]
  - Triangulation[4]
  - Rapidly Exploring Random Test[5]



From RRT paper by LaValle [5]

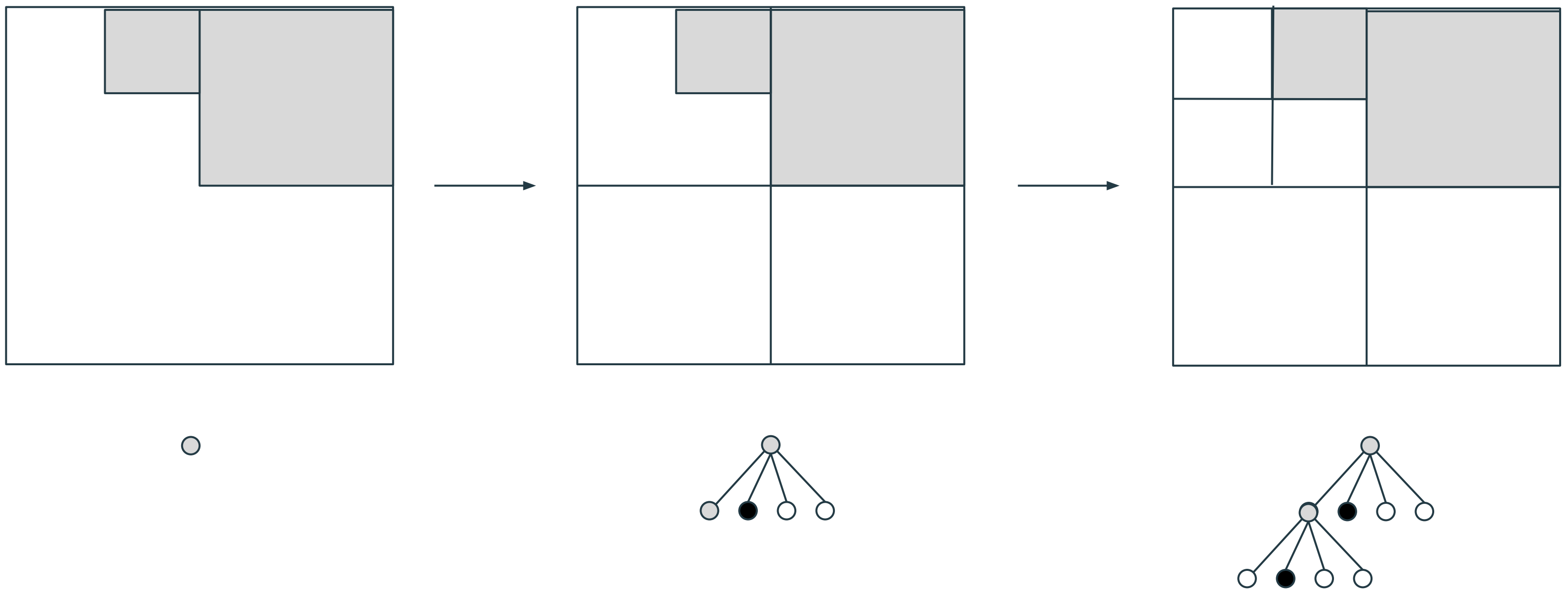
Previous Research: Quadtrees

- Hierarchical data structure
  - Recursively subdivides a space
  - Other Applications:
    - Image processing
    - Cartography



Example quadtree mapping of the config space

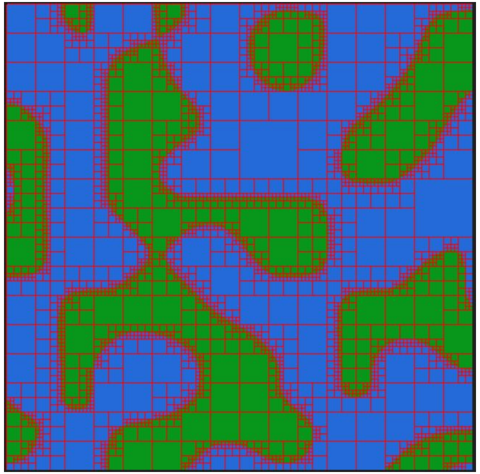
Config space

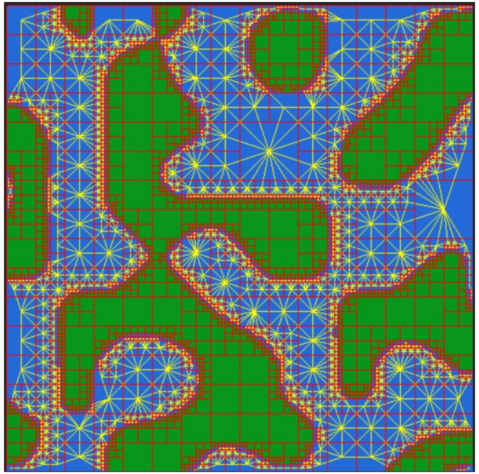


Split when node is gray

Neighbor finding technique for Quadtrees

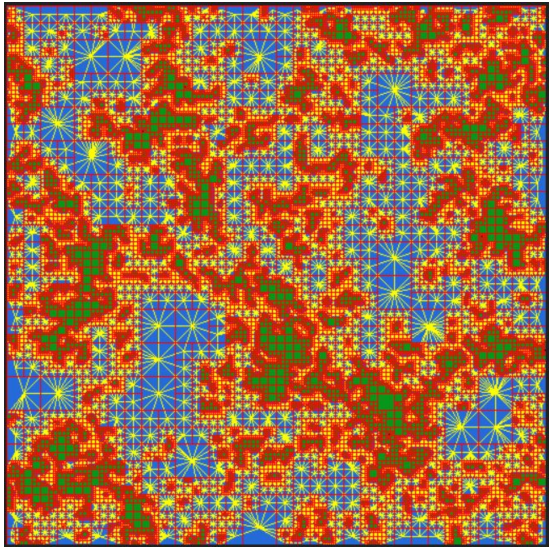
Use techniques from [10] to find neighbors in quadtree





Shortcomings of Previous Work

- Main Issue:
  - Allocation of quadtree nodes in configuration space
- Other shortcomings:
  - Representing vast unstructured environments
  - Configuration spaces of 3D spaces
  - Handling Dynamic obstacles
  - Quadtrees require more memory



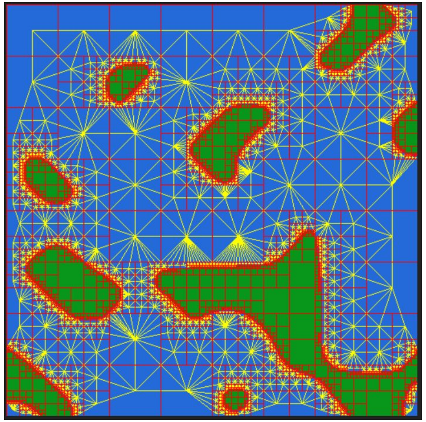
Quadtrees on dense environment
~26000 nodes

Problem Statement

Goal: Create a more **efficient quadtree representation** of the configuration space assuming a holonomic robot

Scenario: Combination of large obstacles and small but dense obstacles

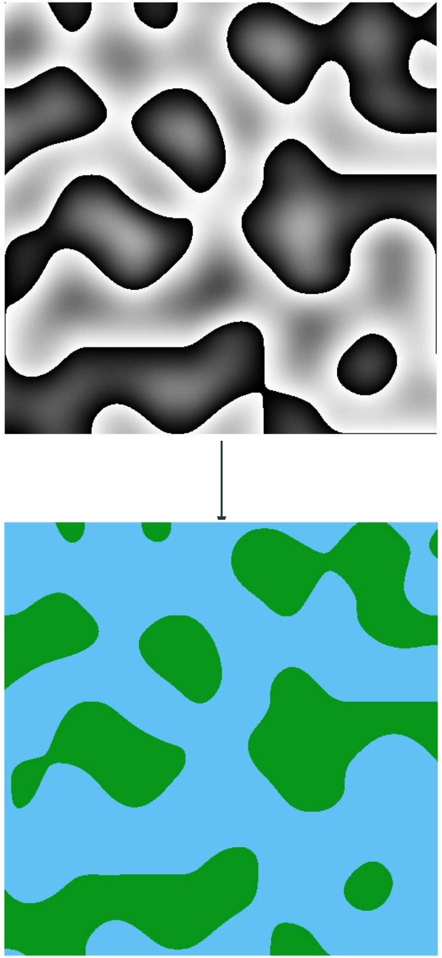
Task: Starting node ⟶ Goal node



Quadtree mapping of the config space

Procedural Map Generation

- Perlin Noise to randomly generate configuration space
    - Gradient noise by Ken Perlin
    - Common in map generation for games
    - Binary threshold is used to generate islands
    - Realistically simulates a naval path planning problem



Pruning Algorithm Heuristics

Quadtrees Characterization and Properties

Splitting rules:

- Minimum Resolution
- Pruning Distance Thresholds t1,t2,and t3

Heuristic Considerations

Optimal path close to:

- Straight line from start to goal

Obstacles considerations:

- Only around starting node, goal node and optimal path

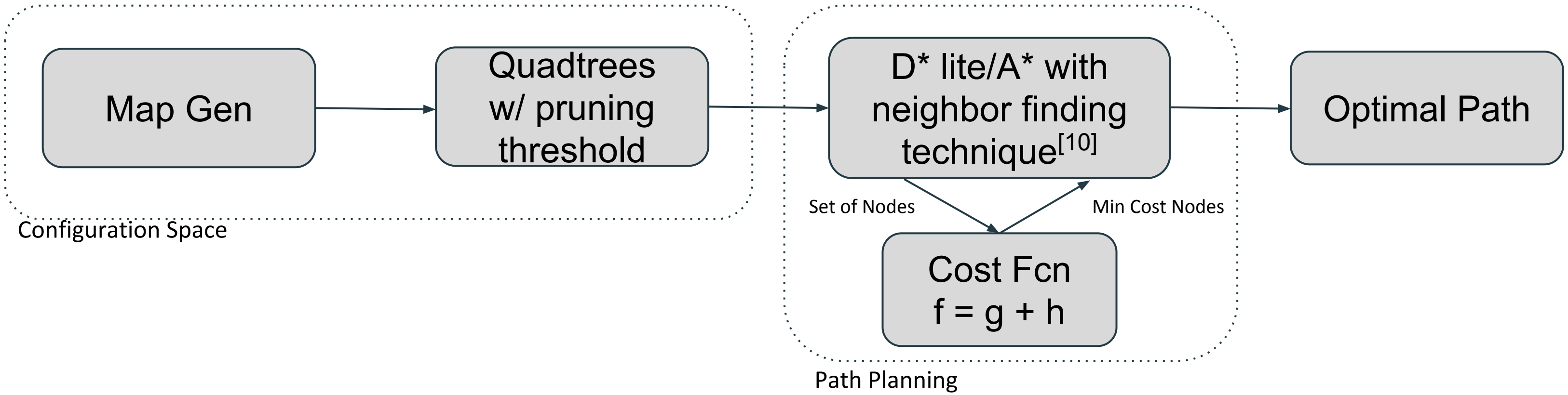
Quadtree Resolution considerations:

- Size of smallest chokepoint

Split if:

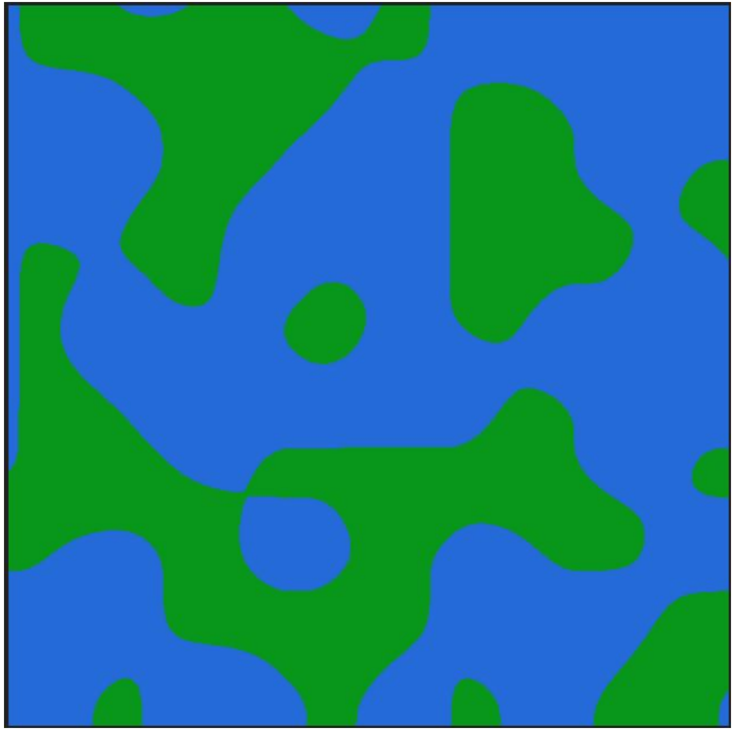
dist(node, start) <= t1 OR dist(node,goal) <= t2 OR dist(node,start) + dist(node,goal) <= t3

Dynamic Path Planning Implementation



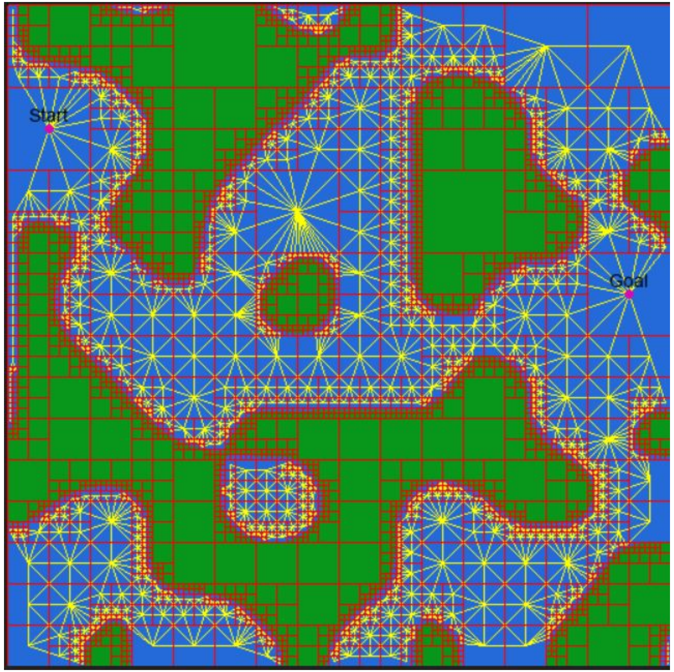
Preliminary Results

Initial Map

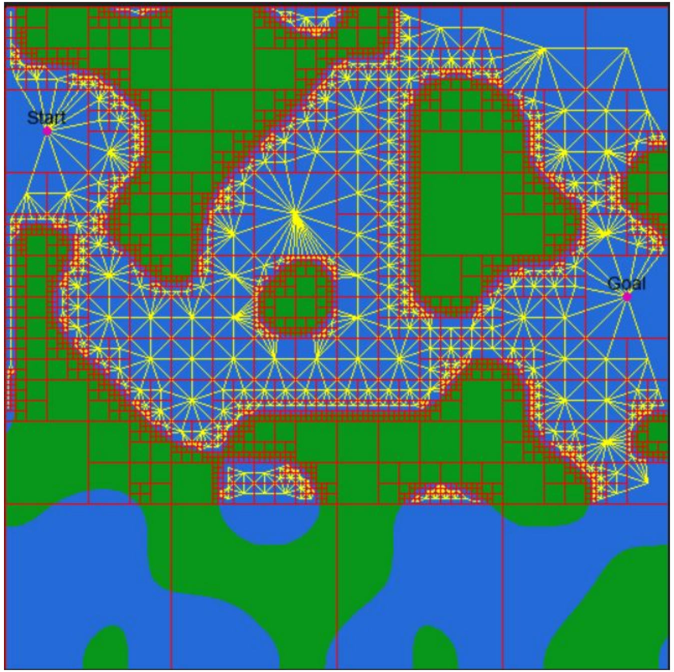


Map Generation Output

Pruned Quadtree Results



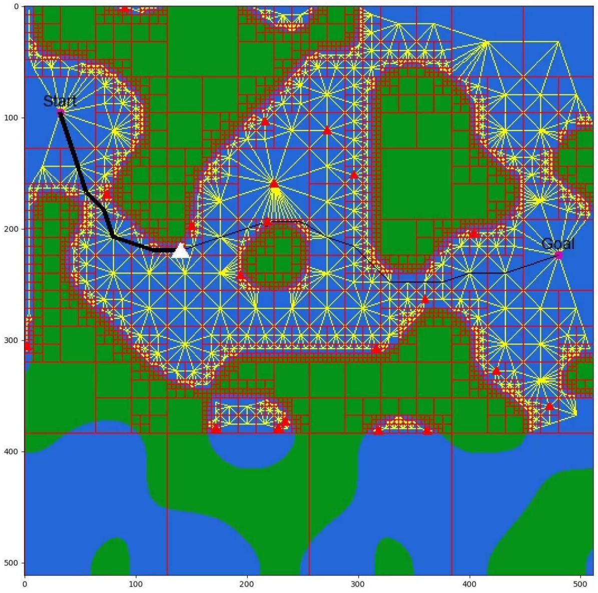
Quad Tree Without Pruning Output (1054 nodes)



Quad Tree With Pruning Output (492 nodes)

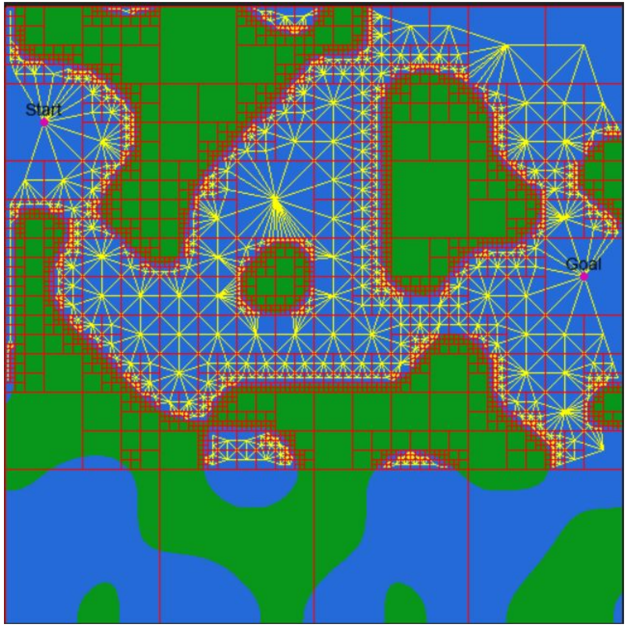
Pruned Path Planning with Dynamic Obstacles (Preliminary Results)

- Moving obstacles (Red triangles) are added to the map
- A* is used every time step to completely recalculate the path



Conclusion

- More efficient representation of configuration space
  - Less memory needed
  - Possibly faster planning
- Pruned Quadtrees Approach
  - Remove unneeded nodes
  - Focus on nodes needed for the path



Future Works

- Applying a better pruning heuristic to be more generalizable to new maps
- Using ALT algorithm[11] to further increase path planning speed
- Use D* Lite to optimize path planning with dynamic obstacles
- K-framed Quadtrees[6] with pruning?
- Testing on larger environments

Contacts

Akhil Avula, Aidan Cookson, Calvin Chang, Daniel Truong
aavula@g.ucla.edu, aucookson@g.ucla.edu, calvinchang33@g.ucla.edu, dktruong@g.ucla.edu

References

[1] Probabilistic Roadmaps for Path Planning in High-Dimensional Configuration Spaces - Robotics and Automation
[2] https://link.springer.com/article/10.1007/BF00288933
[3] https://dl.acm.org/doi/10.1145/359156.359164
[4] https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.83.1711&rep=rep1&type=pdf
[5] http://msl.cs.uiuc.edu/~lavalle/papers/Lav98c.pdf
[6] https://link.springer.com/chapter/10.1007/978-3-319-70833-1_5
[7] https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=8703890
[8] https://www.nssga.org/making-case-infrastructure-investment/cars-in-traffic/
[9] https://en.wikipedia.org/wiki/Cargo_aircraft#/media/File:An-124_ready.jpg
[10] https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.445.7785&rep=rep1&type=pdf
[11] https://www.cs.princeton.edu/courses/archive/spring06/cos423/Handouts/GH05.pdf