



UNIVERSITY OF
SOUTH CAROLINA

CSCE 574 ROBOTICS

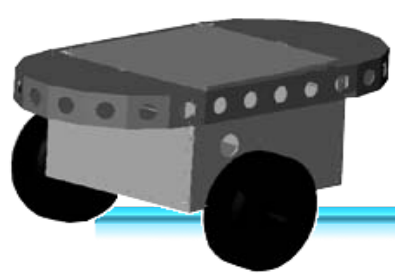
Coverage



Coverage

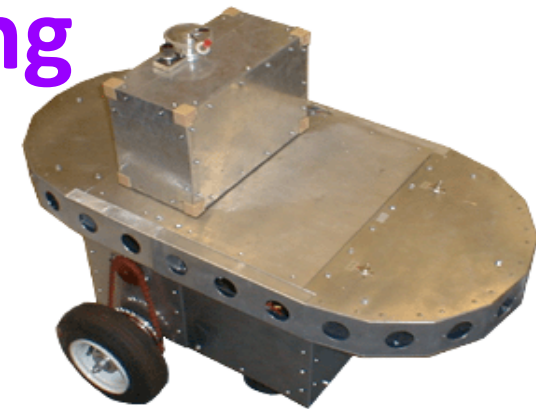
- A task performed quite often in everyday life:
 - Cleaning
 - Painting
 - Plowing/Sowing
 - Tile setting
 - etc.



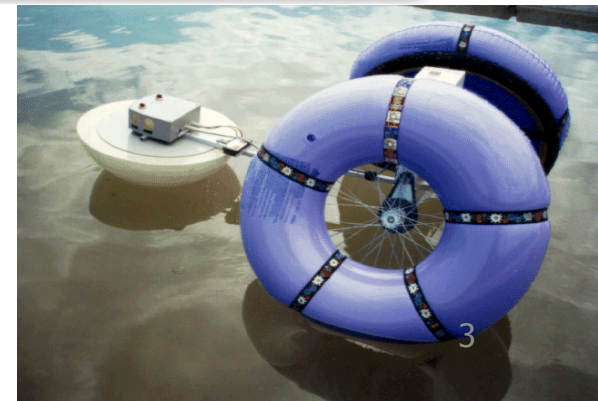


Motivation

Humanitarian Demining

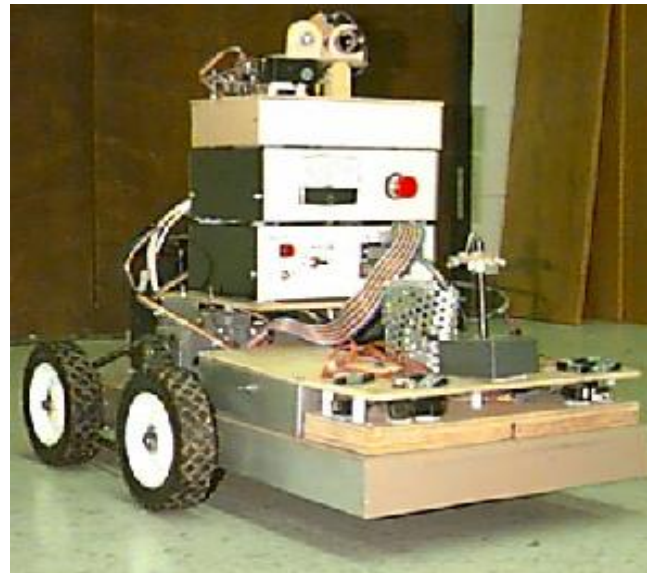


CSCE 574: Robotics



Motivation

Lawn Mowing



Motivation

Vacuum Cleaning



Robotic Coverage

- More than 5 million Roombas sold!
- Automated Car Painting



Roomba Costumes



Coverage

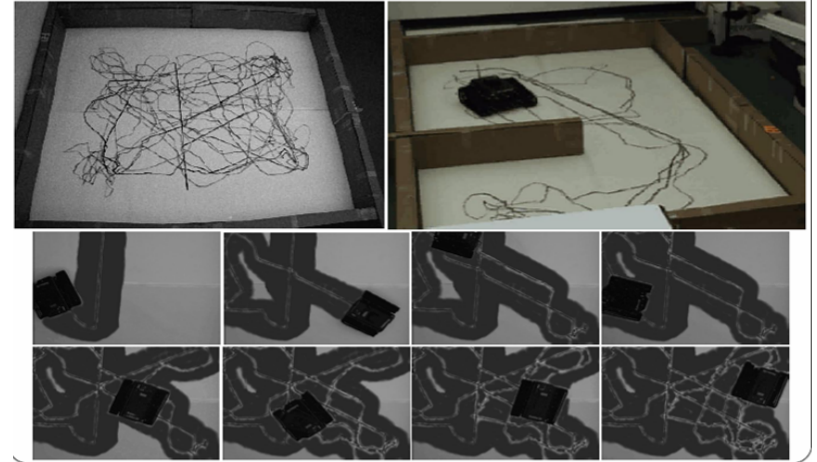
- First Distinction
 - Deterministic **Demining**
 - Random **Vacuum Cleaning**
- Second Distinction
 - Complete
 - No Guarantee
- Third Distinction
 - Known Environment
 - Unknown Environment



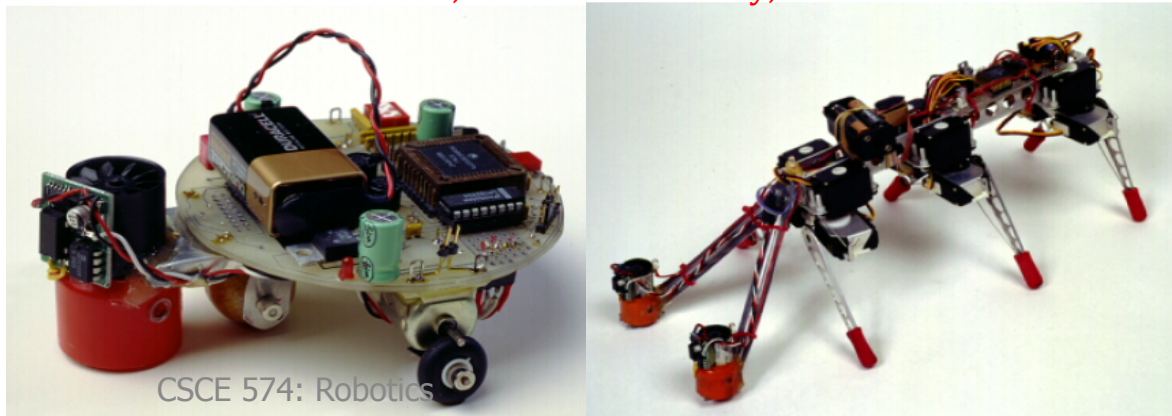
Non-Deterministic Coverage

- Complete Random Walk
- Ant Robotics
 - Leave trail
 - Bias the behavior towards or away from the trails

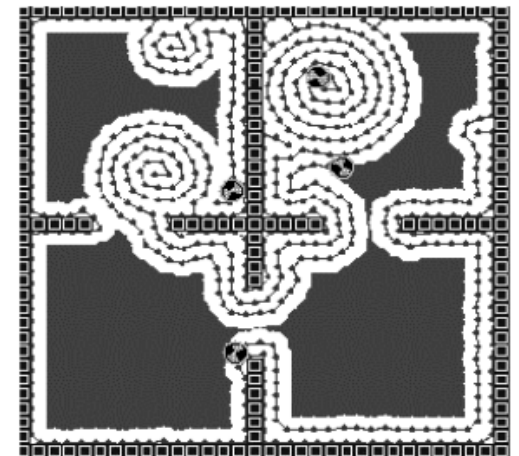
S. Koenig Ant Robotics, terrain coverage



Andrew Russell, Monash University, Australia



CSCE 574: Robotics



Ant Robotics: I. Wagner, IBM & Technion



Deterministic Coverage

- Complete Algorithm
- Guarantees Complete Coverage



Cell-Decomposition Methods

Two families of methods:

- **Exact cell decomposition**

The free space F is represented by a collection of non-overlapping cells whose union is exactly F

Examples: trapezoidal and cylindrical decompositions





UNIVERSITY OF
SOUTH CAROLINA

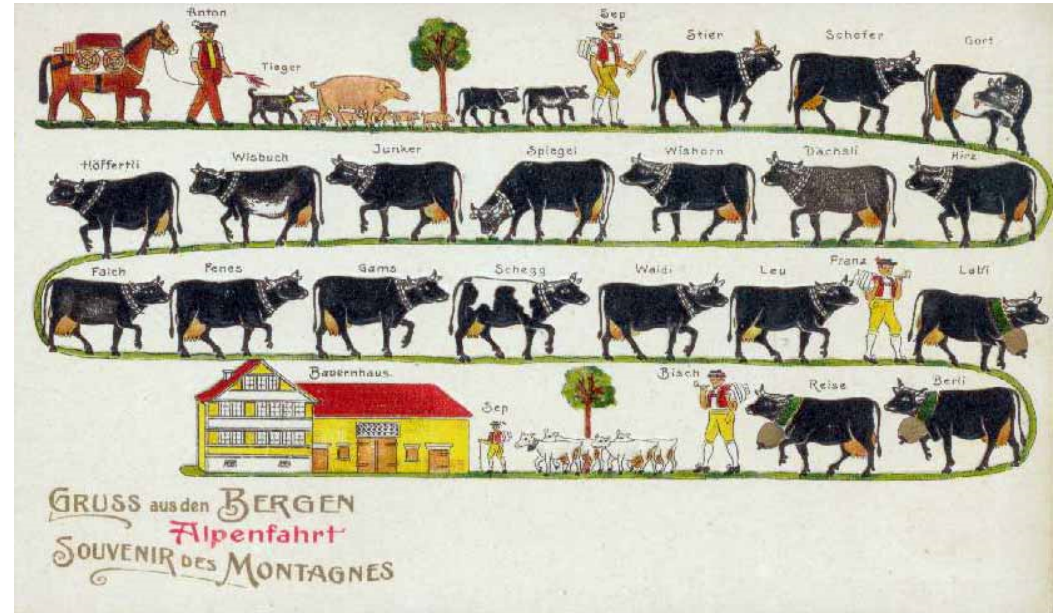
BOUSTROPHEDON CELLULAR DECOMPOSITION

The way of the Ox!



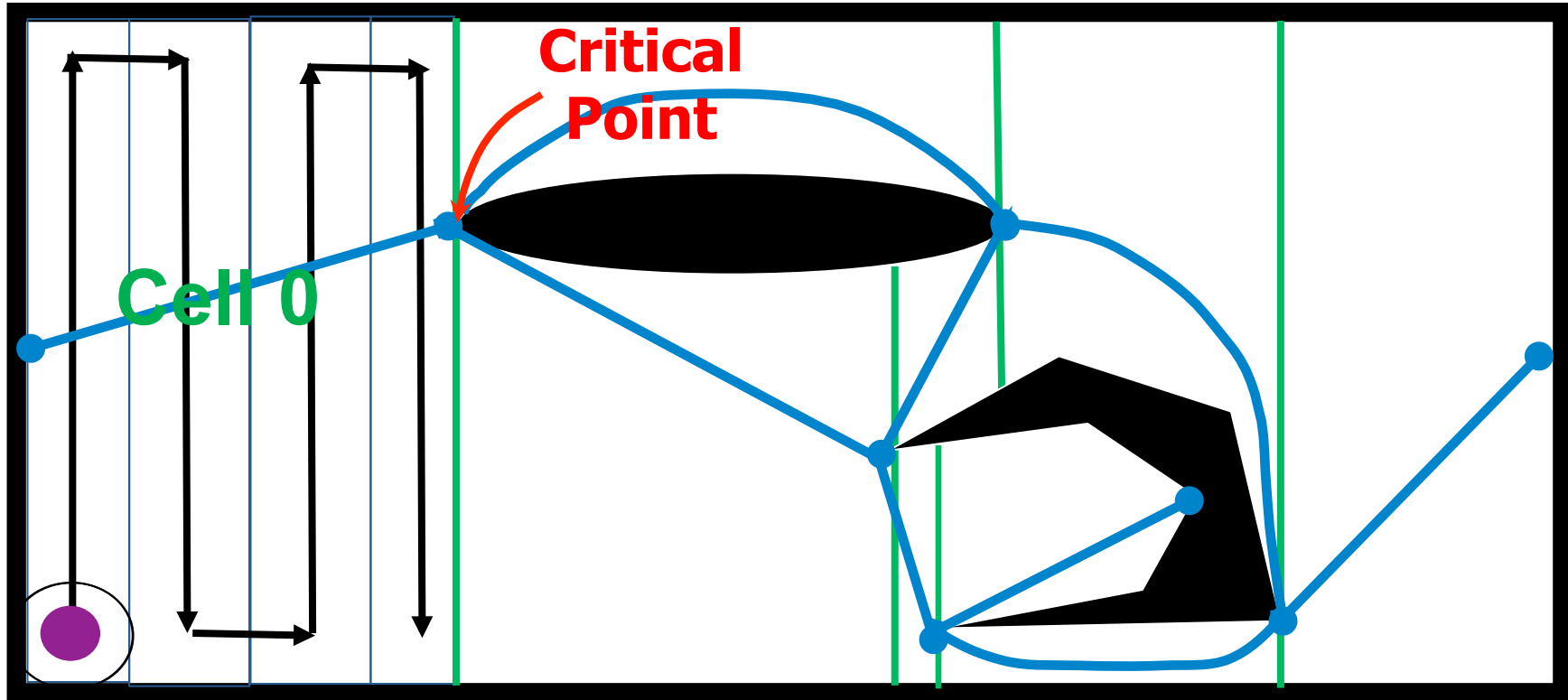
Single Robot Coverage

- Deterministic algorithm
- Guarantee of completeness
- Sensor based
- Unknown Environment



- Seed spreader algorithm: Lumelsky et al, “Dynamic path planning in sensor-based terrain acquisition”, IEEE Transactions on Robotics and Automation, August 1990.
- Boustrophedon algorithm: Choset and Pignon, “Coverage path planning: The boustrophedon cellular decomposition”, International Conference on Field and Service Robotics, 1997.

Single Robot Coverage



→
Direction of Coverage

—
Cellular Decomposition

—•—•—
Reeb graph
Vertices: Critical Points
Edges: Cells

Critical Points

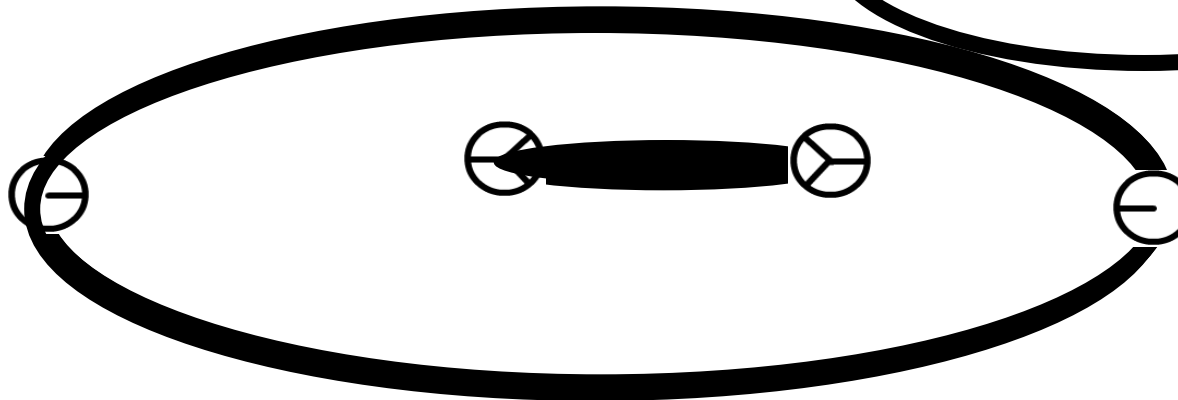
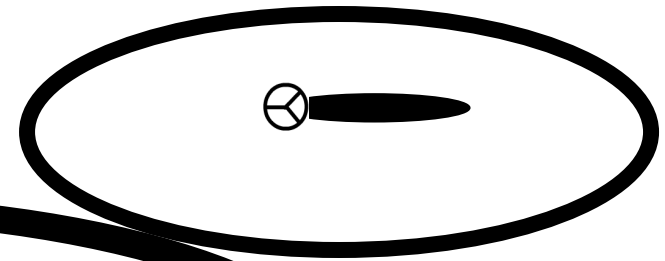
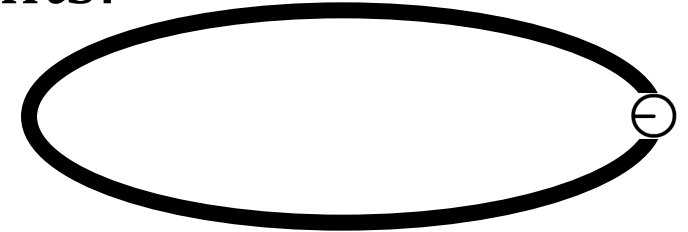
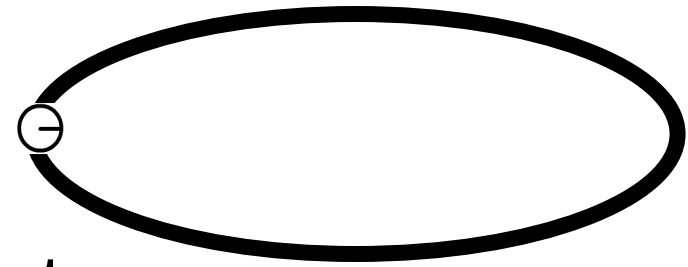
- There are four types of critical points:

⊖ Forward Concave critical point

⊕ Reverse Concave critical point

⊗ Reverse Convex critical point

⊙ Forward Convex critical point



Direction of Coverage



Efficient Coverage

- Find an order for traversing the Reeb graph such that the robot would not go through a cell more times than necessary

Solution

- Use the Chinese Postman Problem



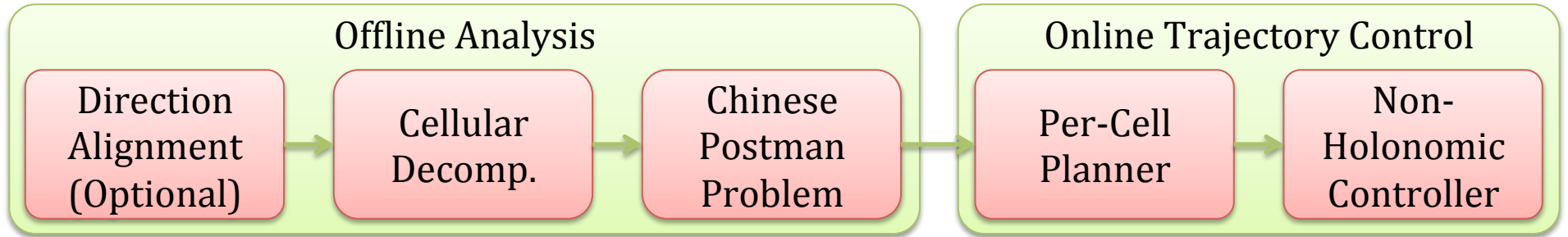
Chinese Postman Problem

- The Chinese postman problem (CPP), is to find a shortest closed path that visits every edge of a (connected) undirected graph. When the graph has an Eulerian circuit (a closed walk that covers every edge once), that circuit is an optimal solution.

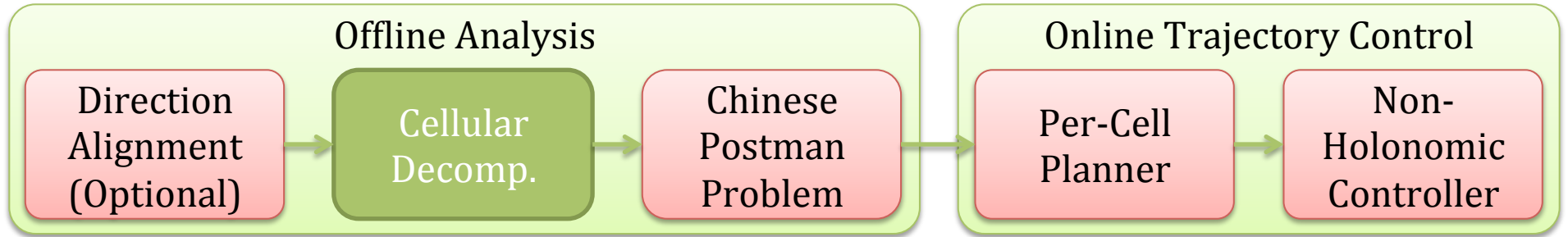
See: J. Edmonds and E.L. Johnson, Matching Euler tours and the Chinese postman problem, Math. Program. (1973).



Offline Analysis Algorithm



Offline Analysis Algorithm



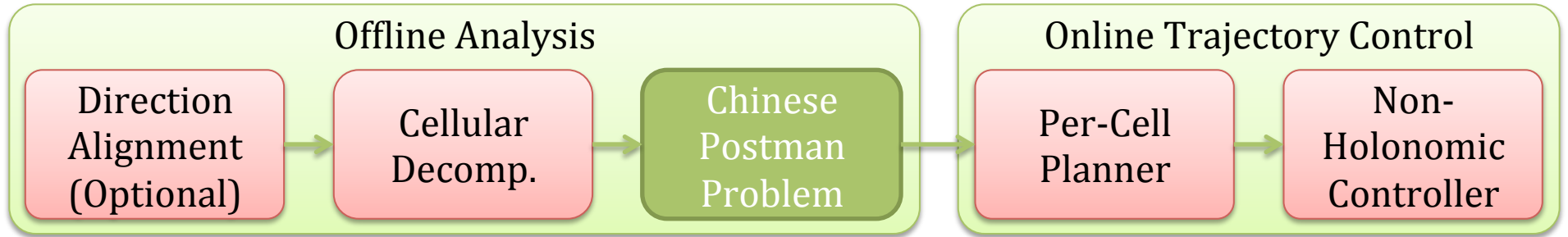
- Input: binary map separating obstacle from free space
- Boustrophedon Cellular Decomposition (BCD)



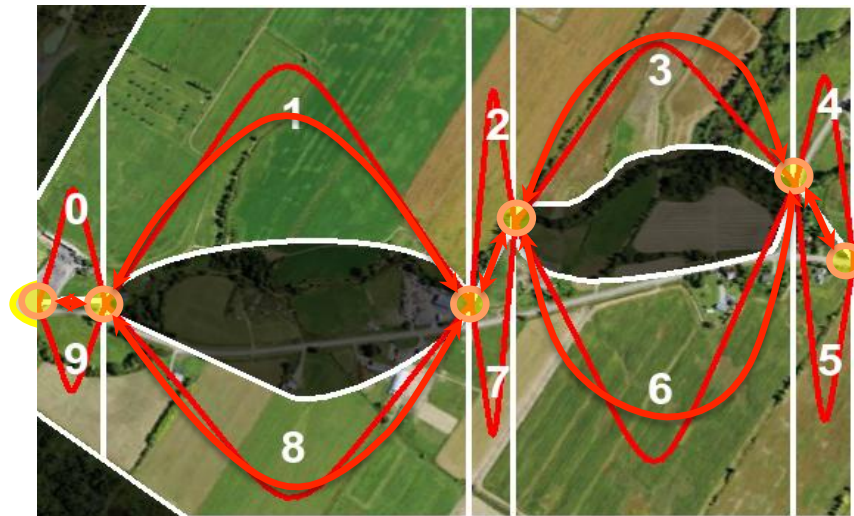
○ : intersections = vertices

↔ : cells = edges

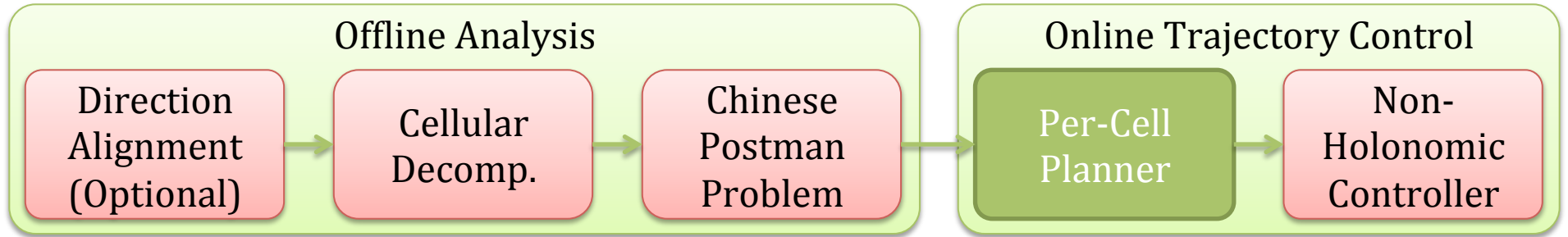
Offline Analysis Algorithm (cont.)



- Chinese Postman Problem
 - Eulerian circuit, i.e. *single* traversal through all cells (edges)



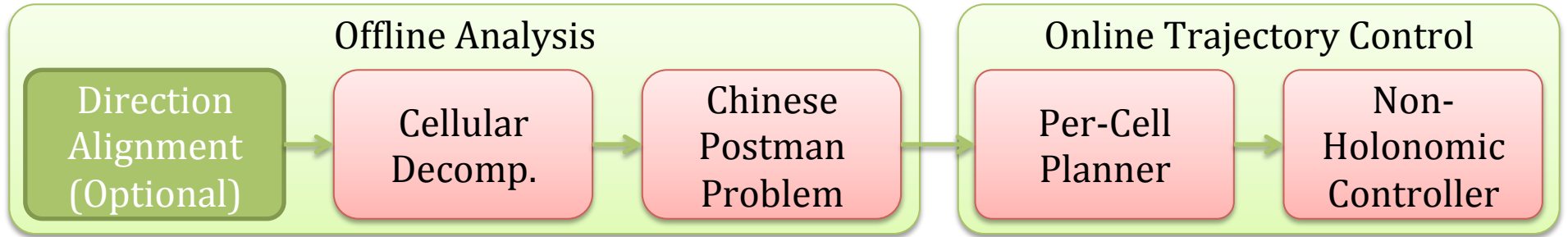
Per-Cell Coverage Planner



- Seed Spreader: piecewise linear sweep lines
- Footprint width



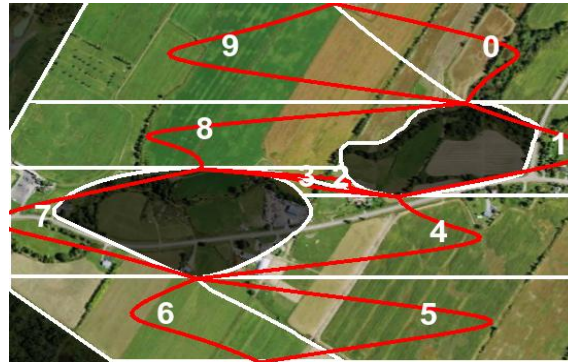
Coverage Direction Alignment



- Static alignment methods



Default



Obstacle Boundaries

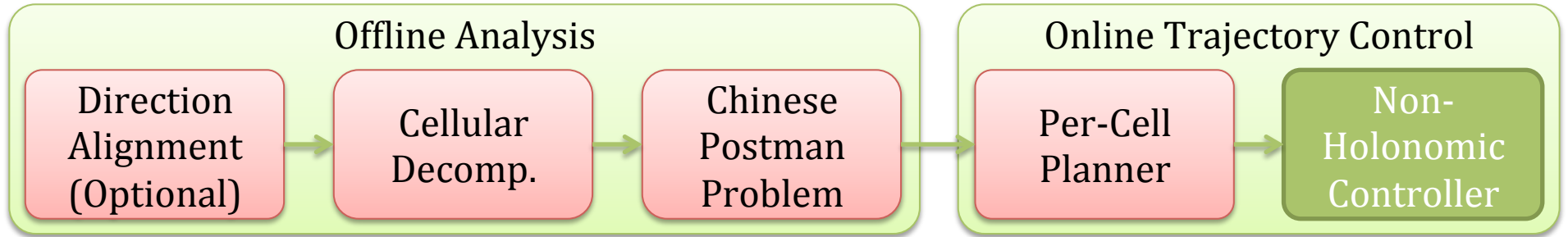


Free Space Distribution

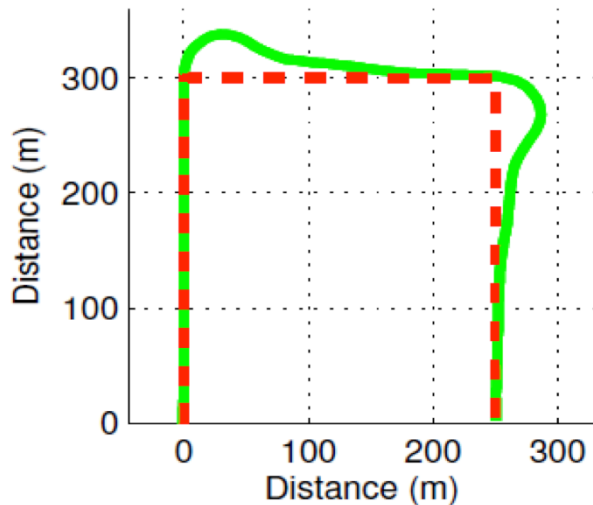
- Alignment with average wind heading (pre-flight)



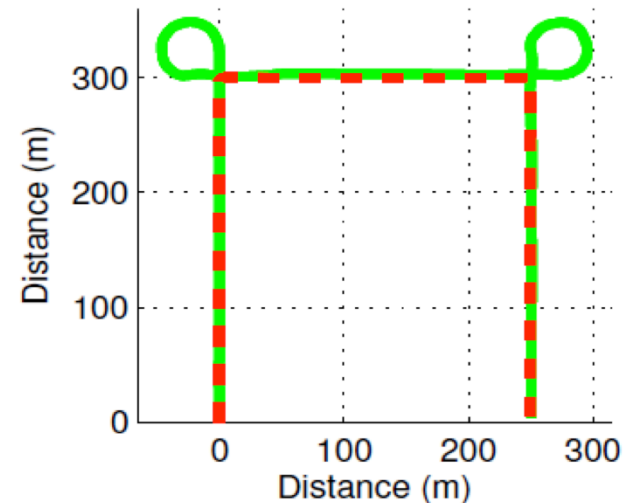
Non-Holonomic Robot Controller



- Turning strategies



Greedy Waypoint Controller

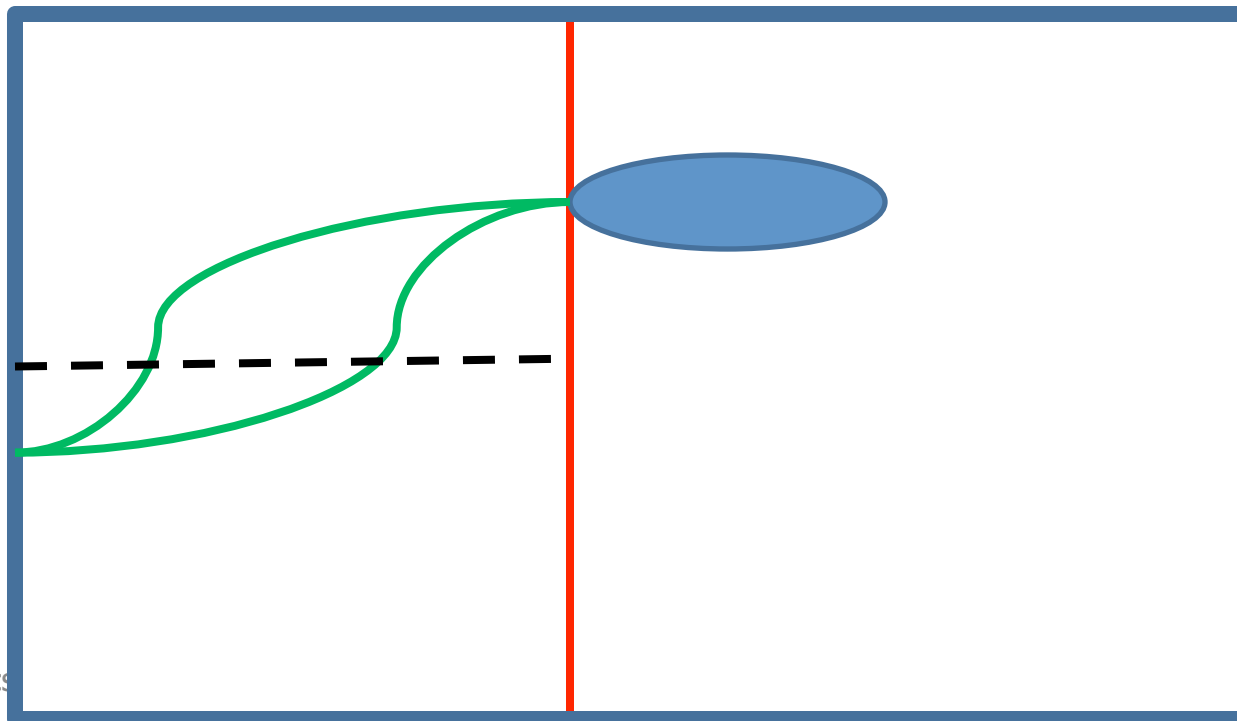


Curlicue Controller



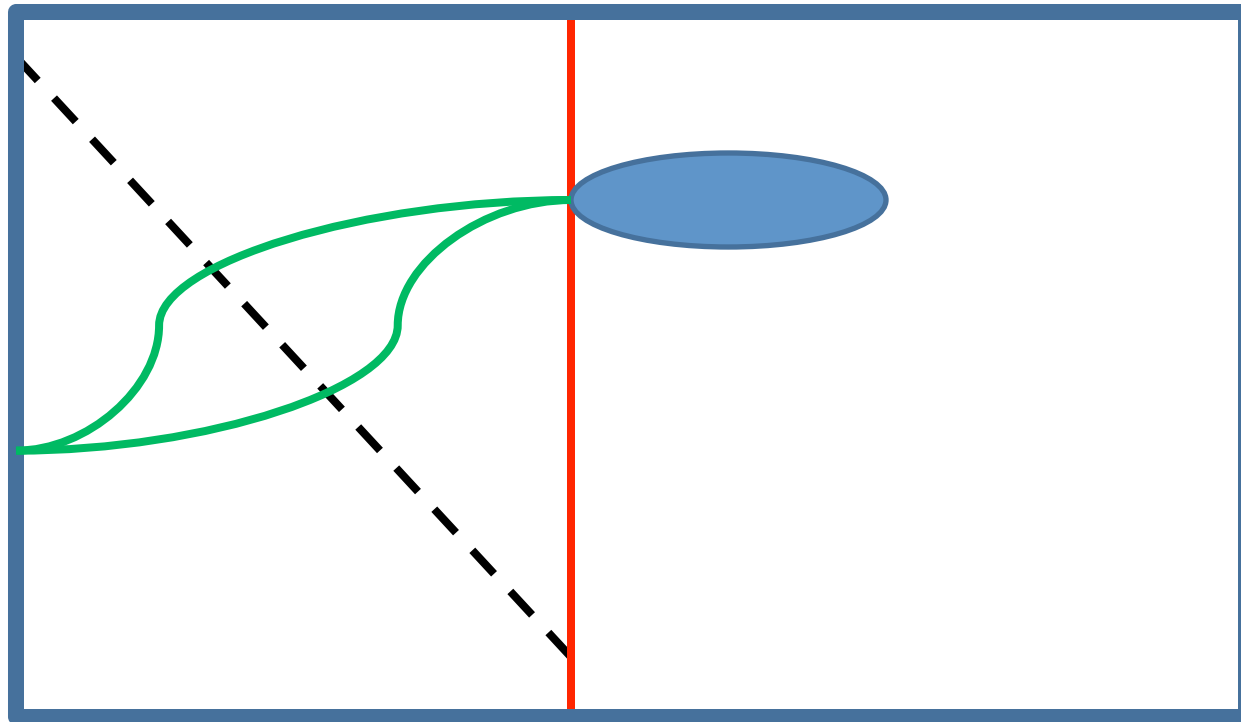
Chinese Postman Problem

- The solution of the CPP guarantees that no edge is doubled more than once
- That means some cells have to be traversed twice
- Cells that have to be traversed/covered are divided in half



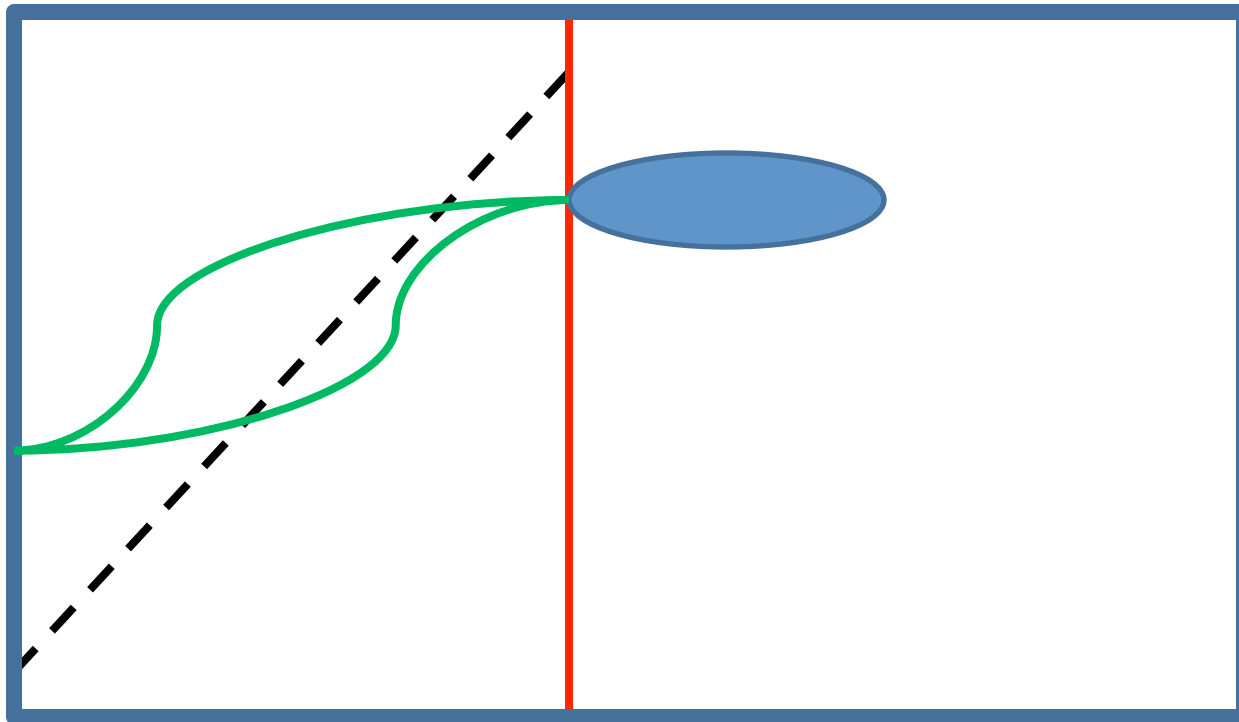
Double Coverage of a Single Cell

- By dividing the cell diagonally we control the beginning and end of the coverage



Double Coverage of a Single Cell

- By dividing the cell diagonally we control the beginning and end of the coverage

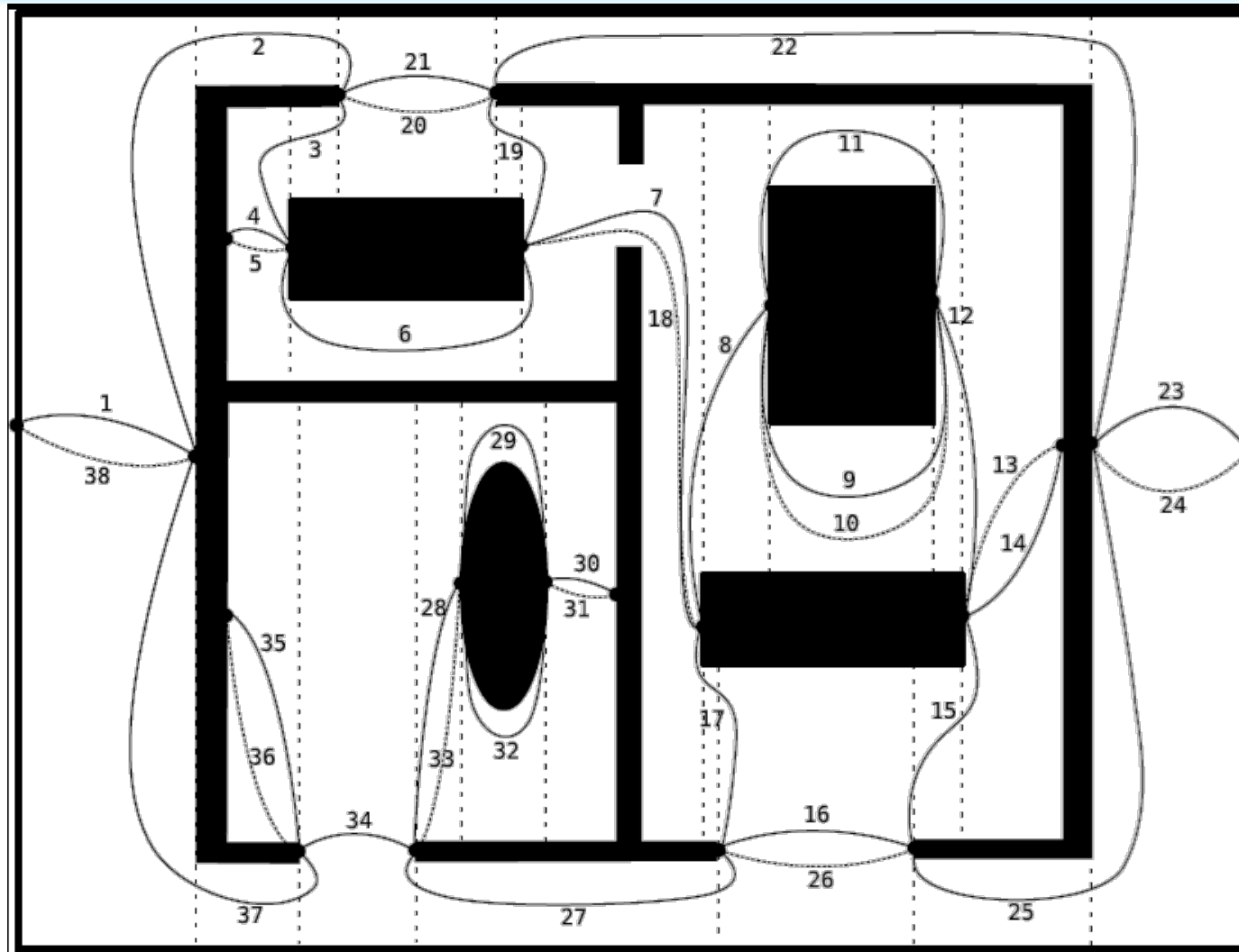


Efficient Coverage Algorithm

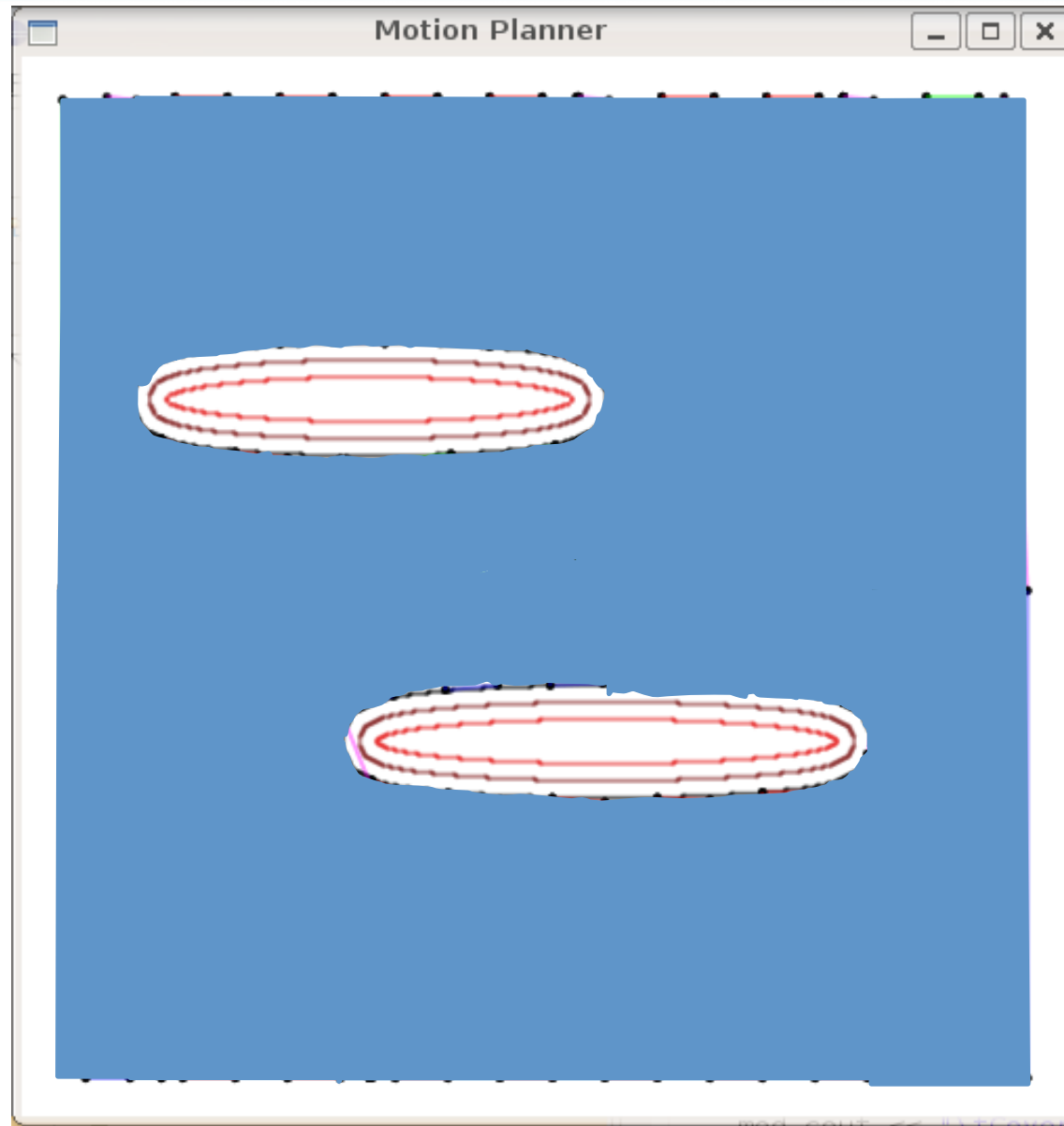
- Given a known environment:
 - Calculate the Boustrophedon decomposition
 - Construct the Reeb graph
 - Use the Reeb graph as input to the Chinese Postman Problem (CPP)
 - Use the solution of the CPP to find a minimum cost cycle traversing every edge of the Reeb graph
 - For every doubled edge divide the corresponding cell in half
 - Traverse the Reeb graph by covering each cell in order



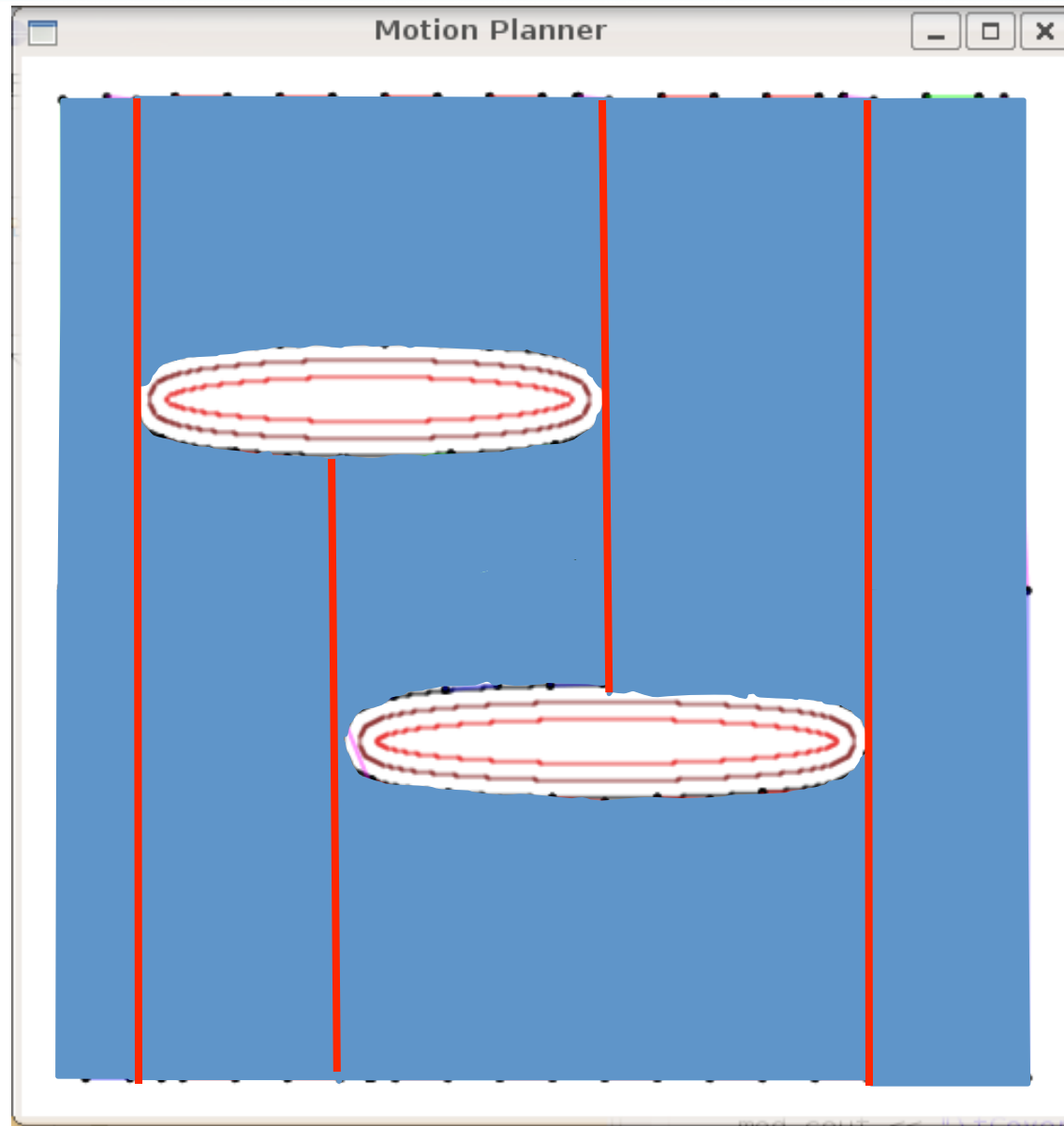
Traversal order of the Reeb graph



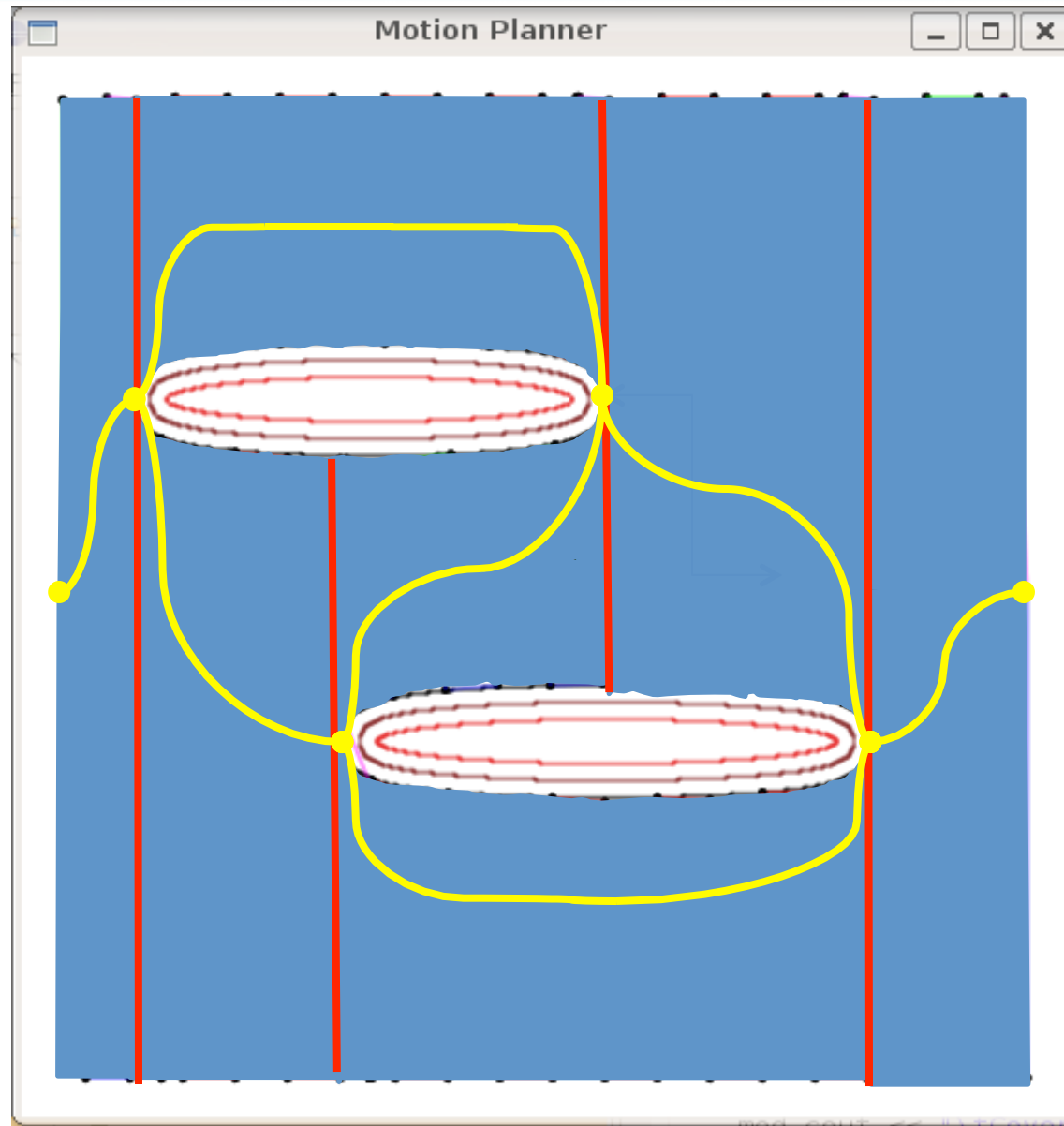
Example



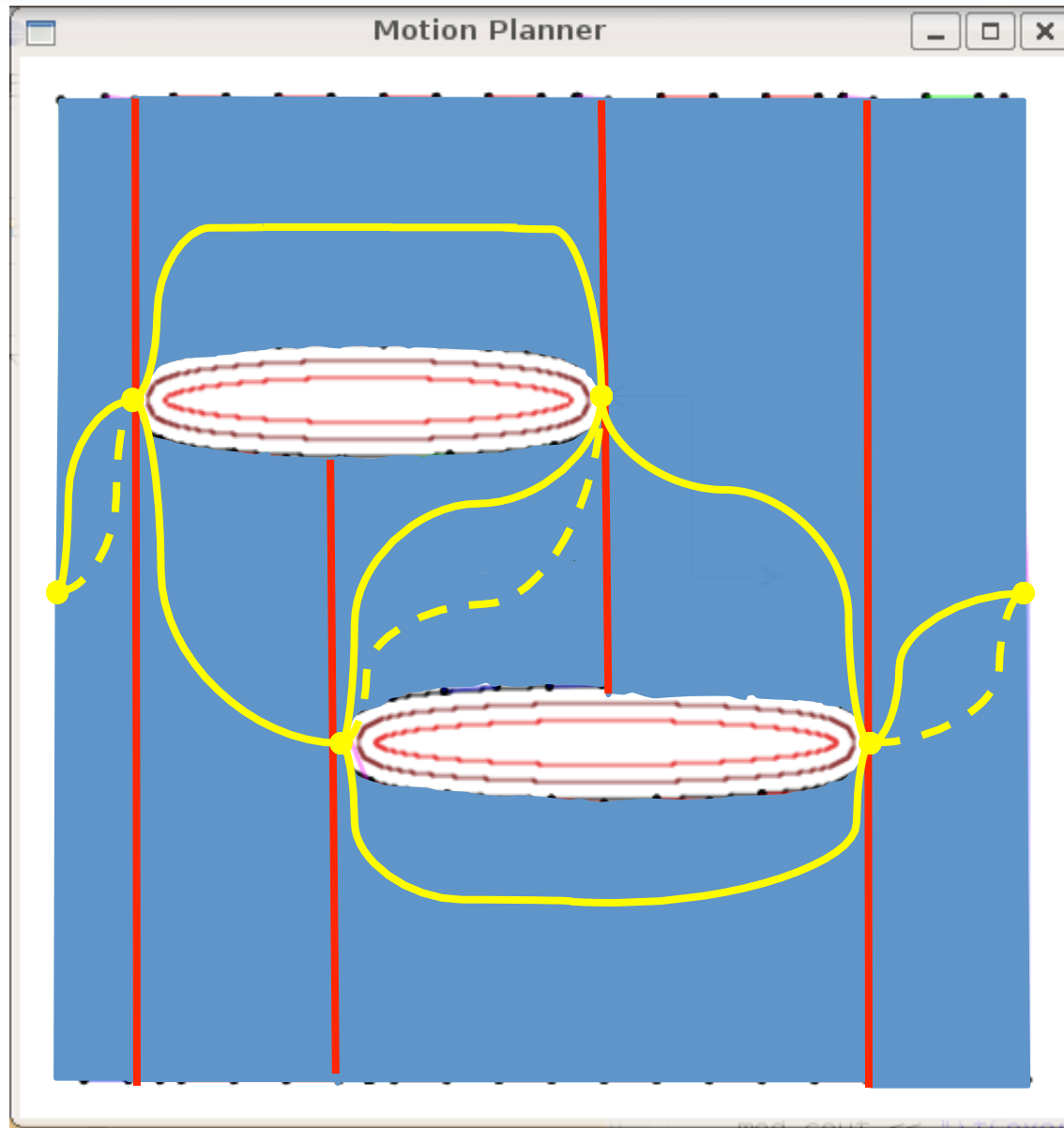
Example: Boustrophedon Decomposition



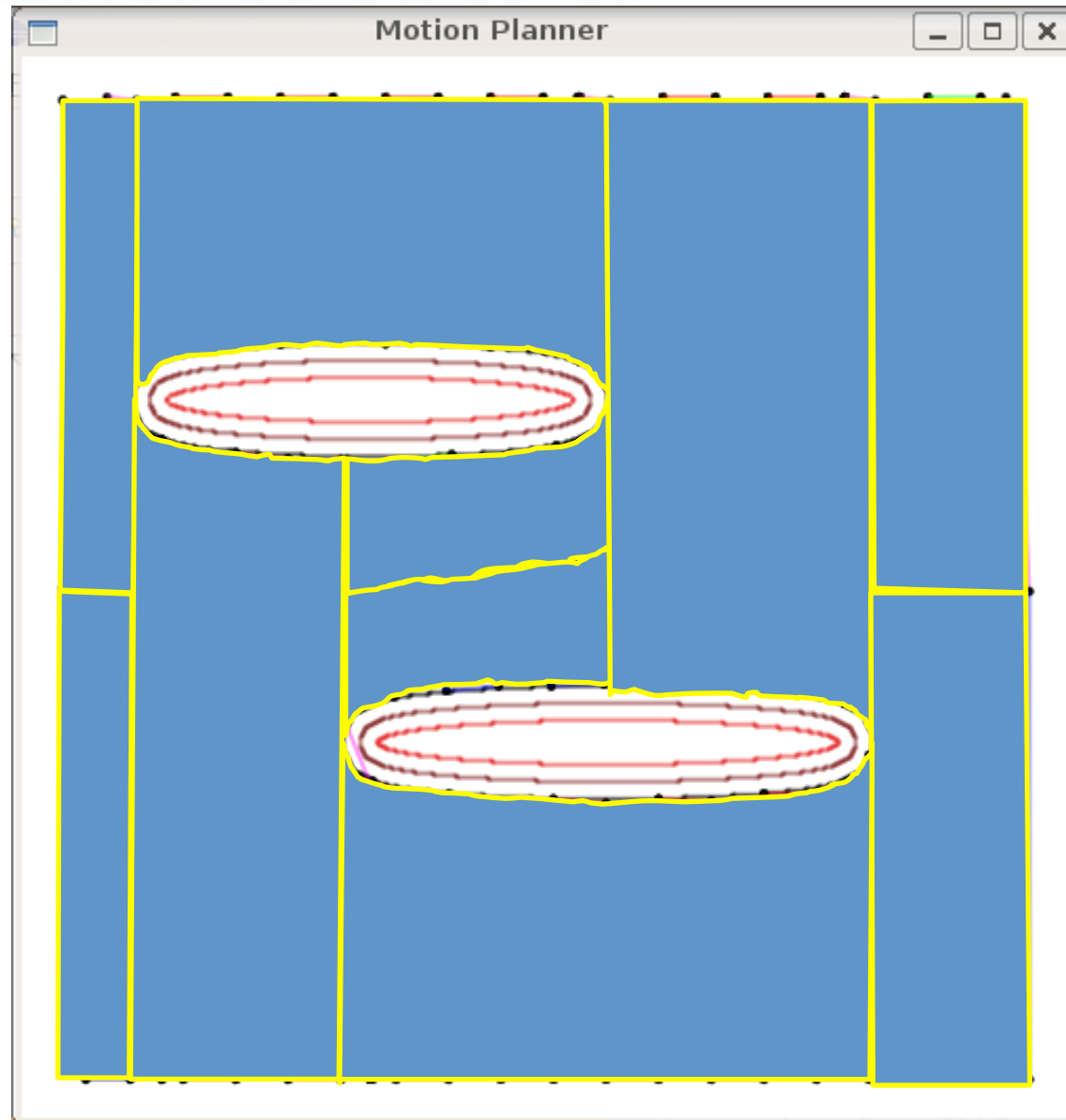
Example: Reeb Graph



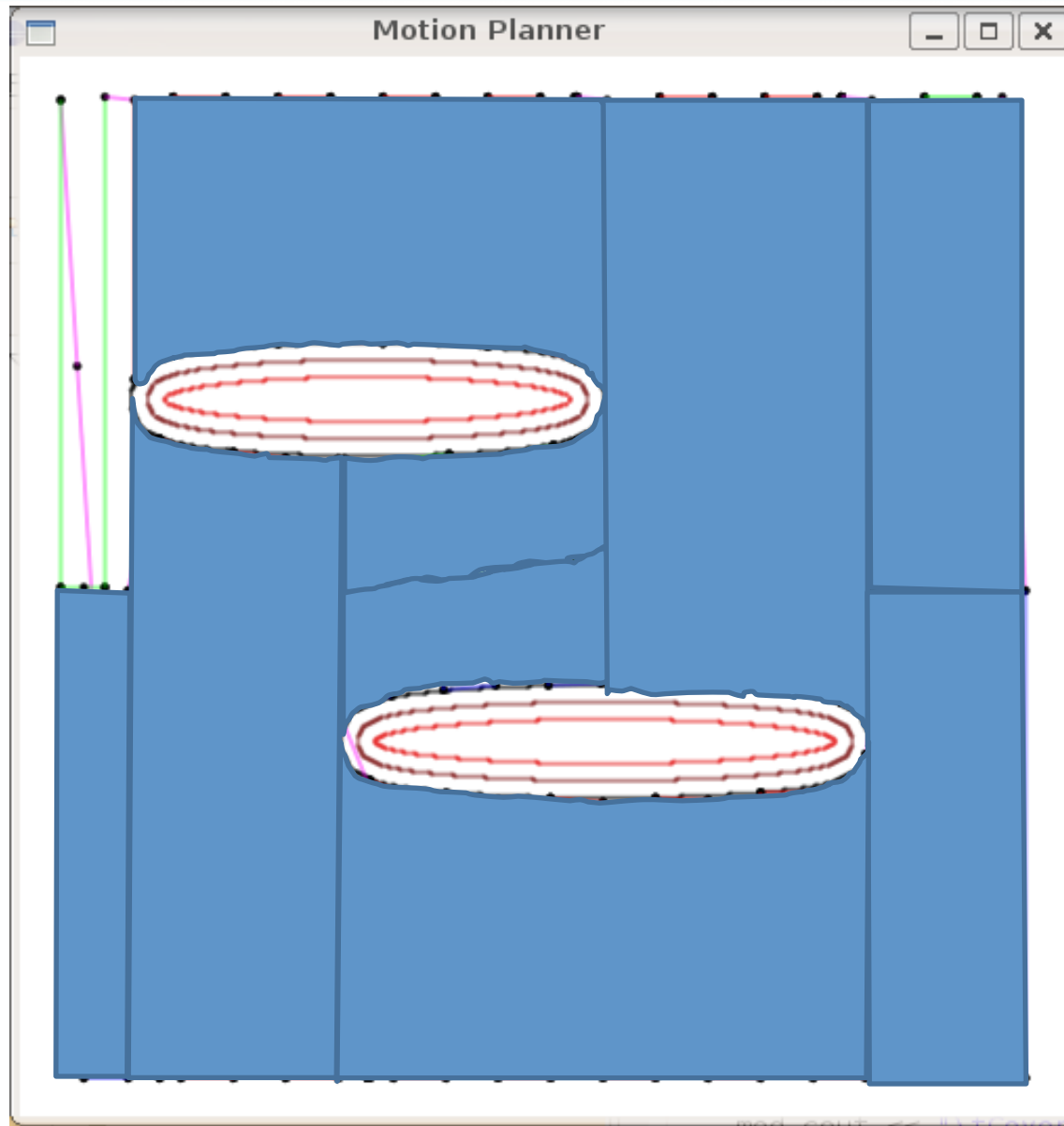
Example: CPP solution



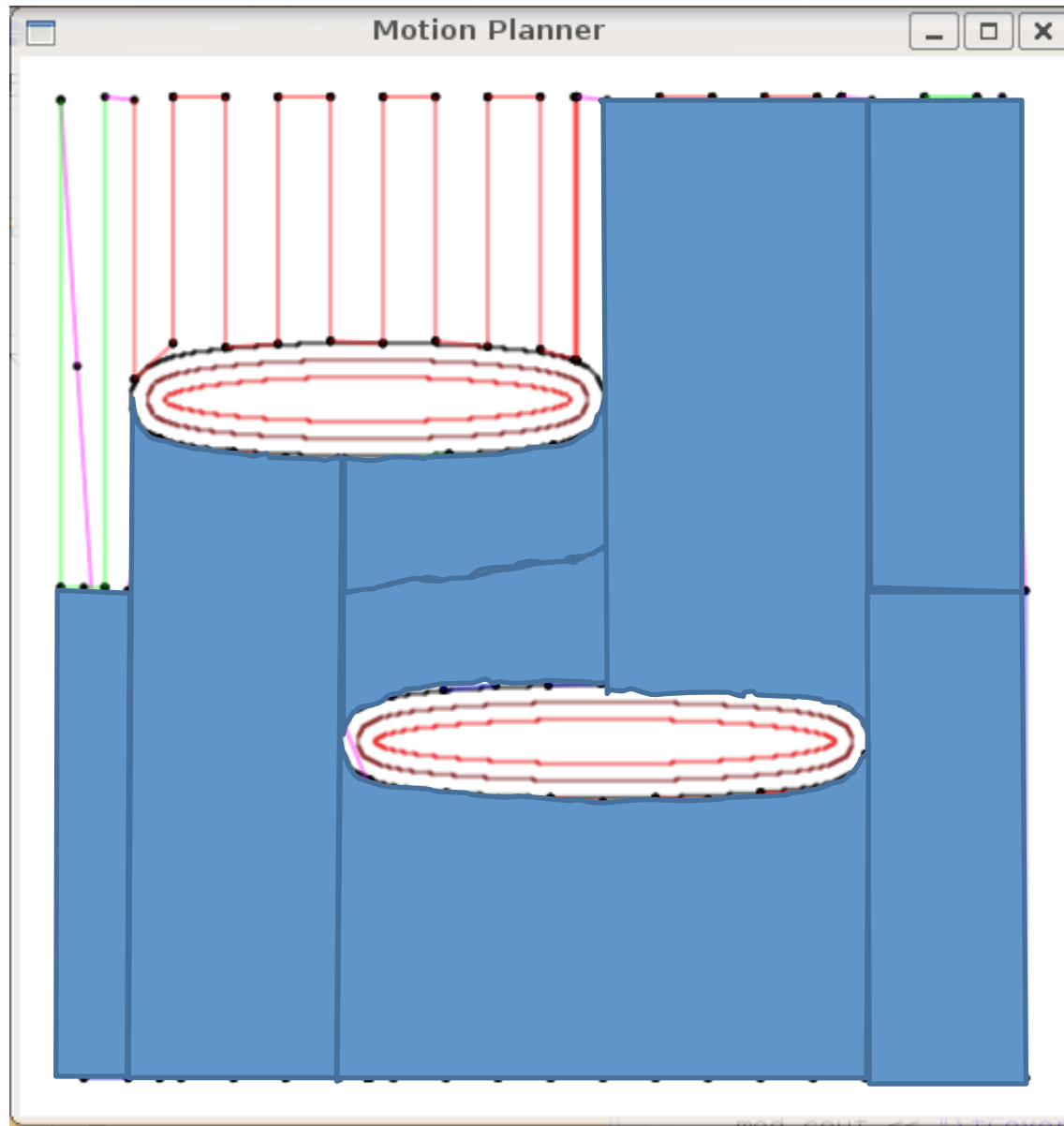
Example



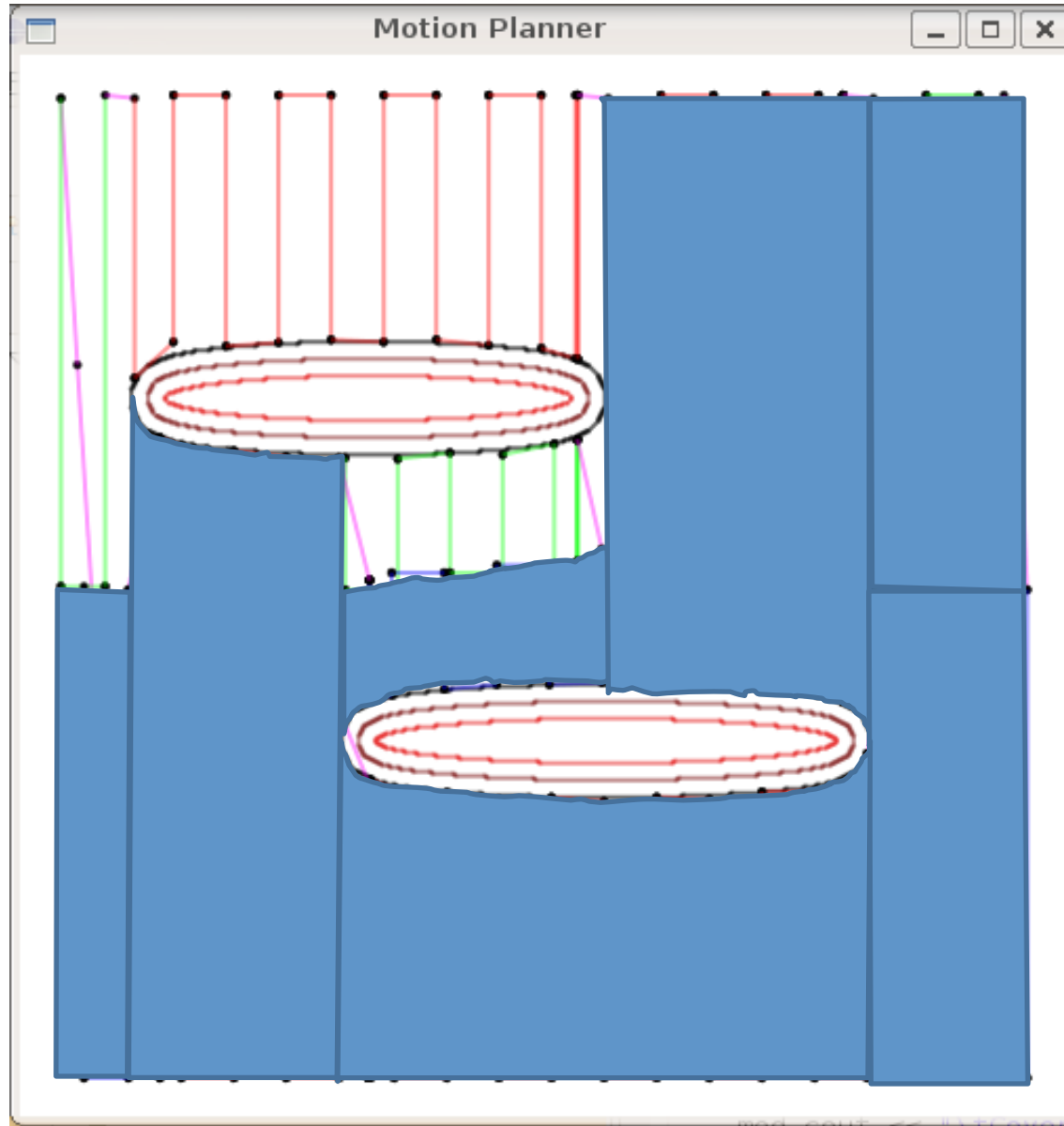
Example



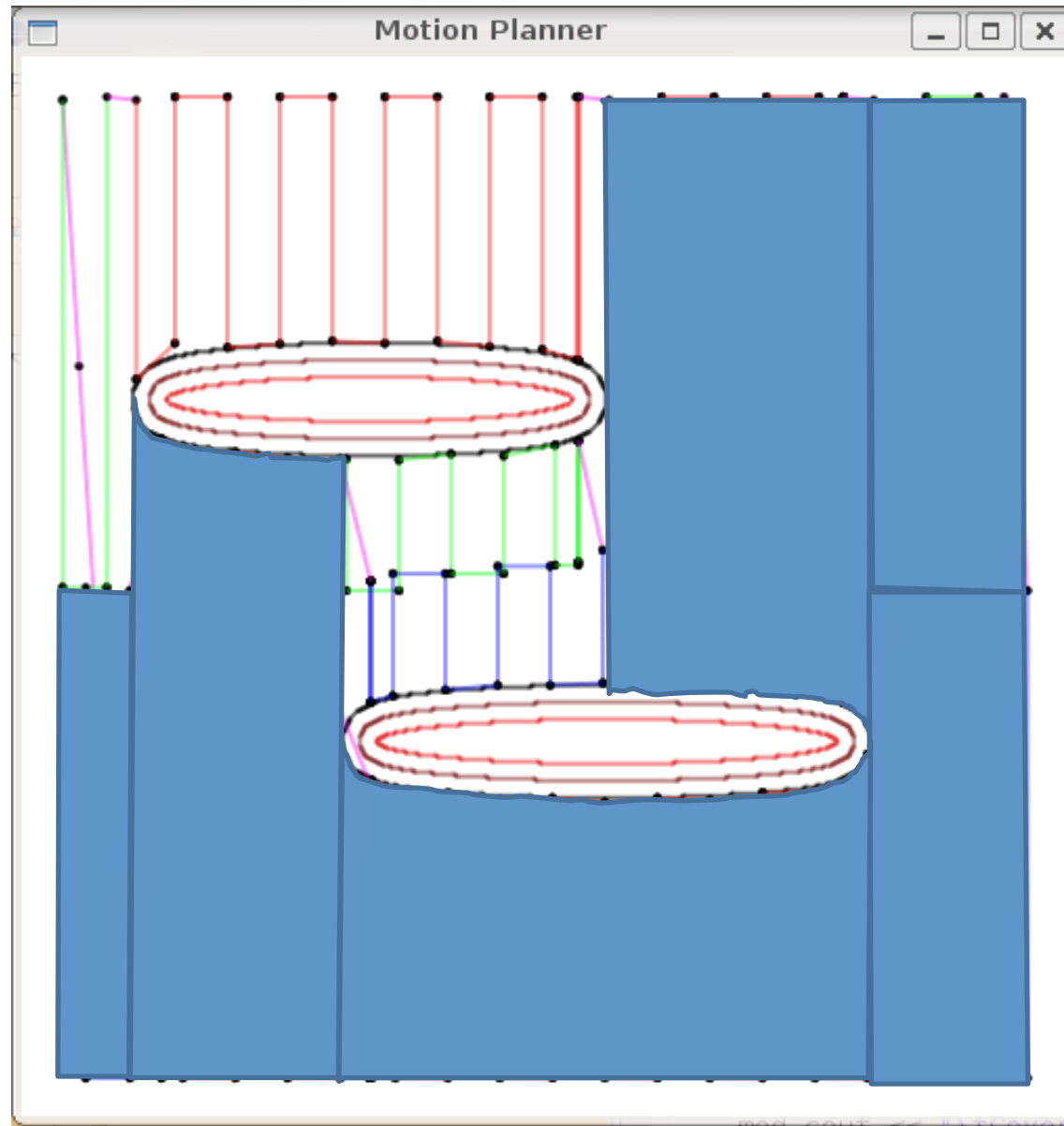
Example



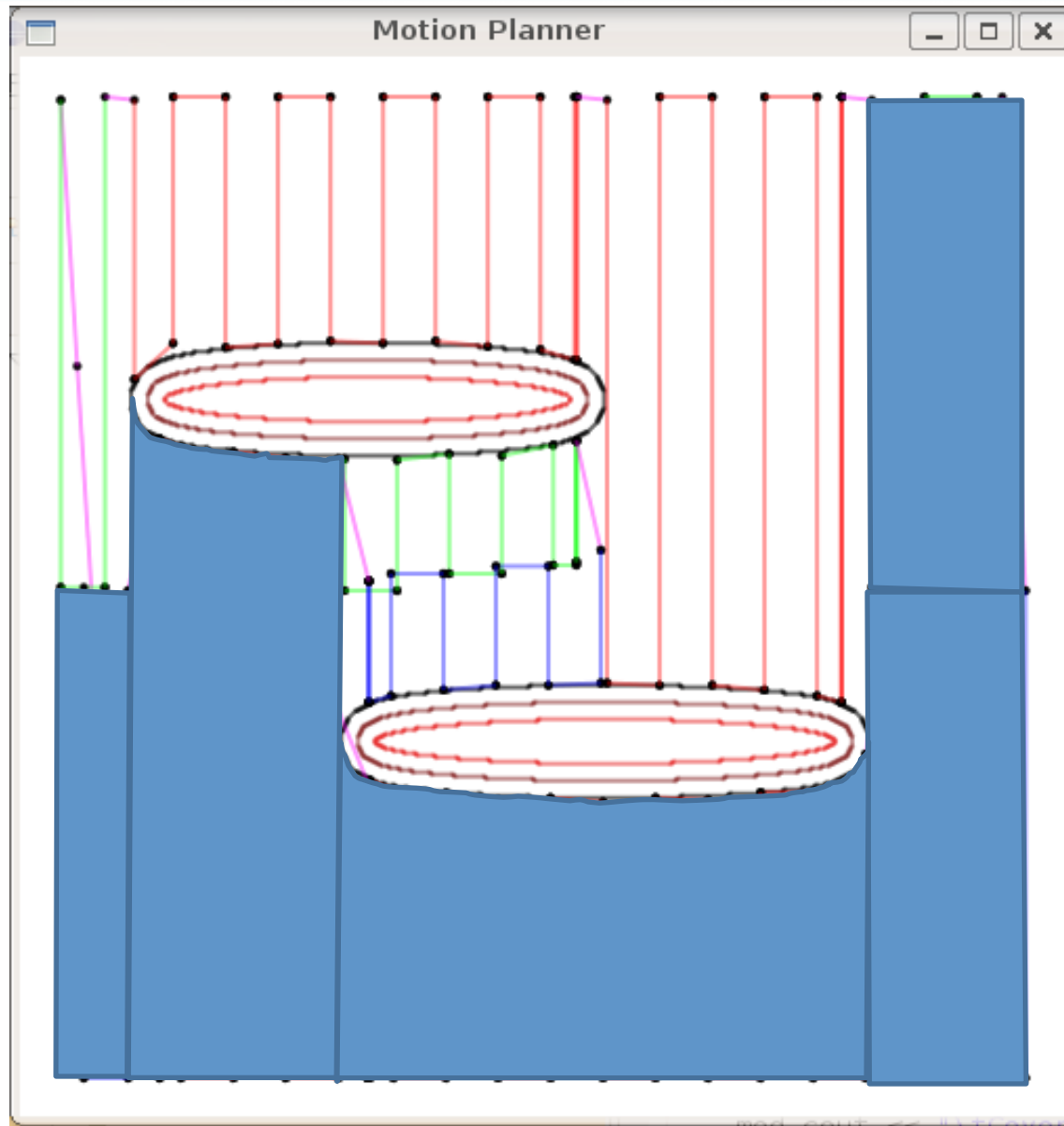
Example



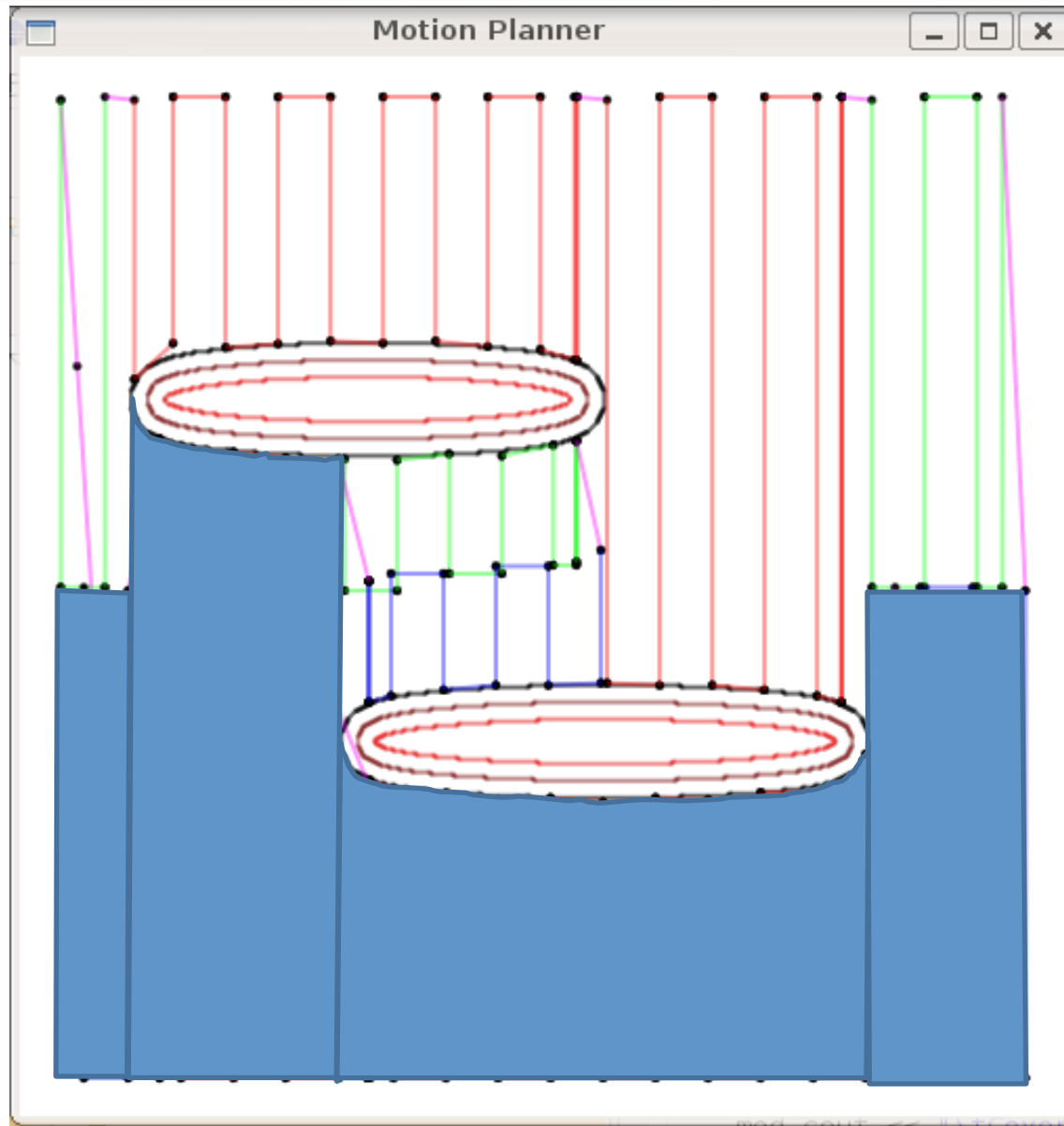
Example



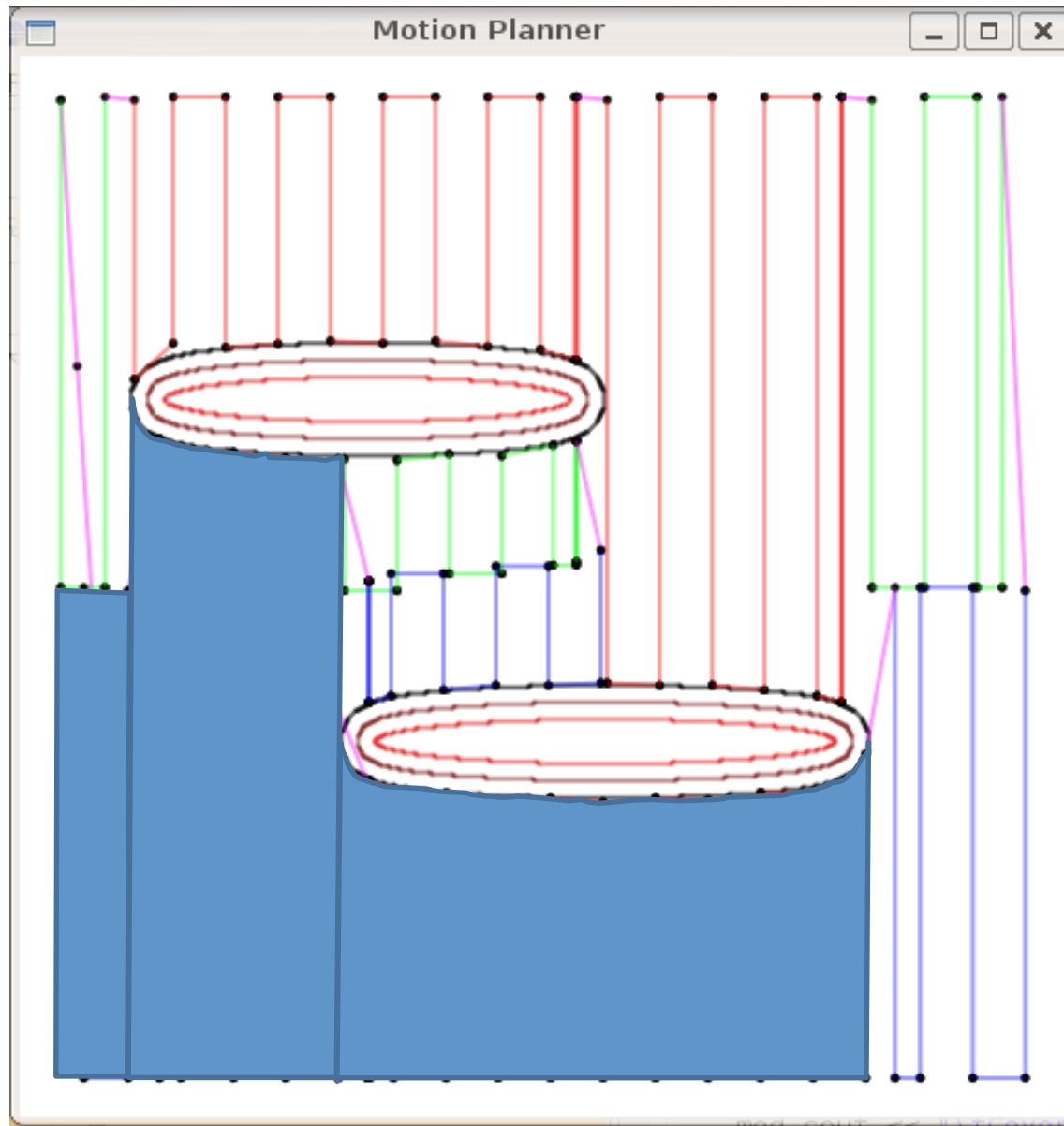
Example



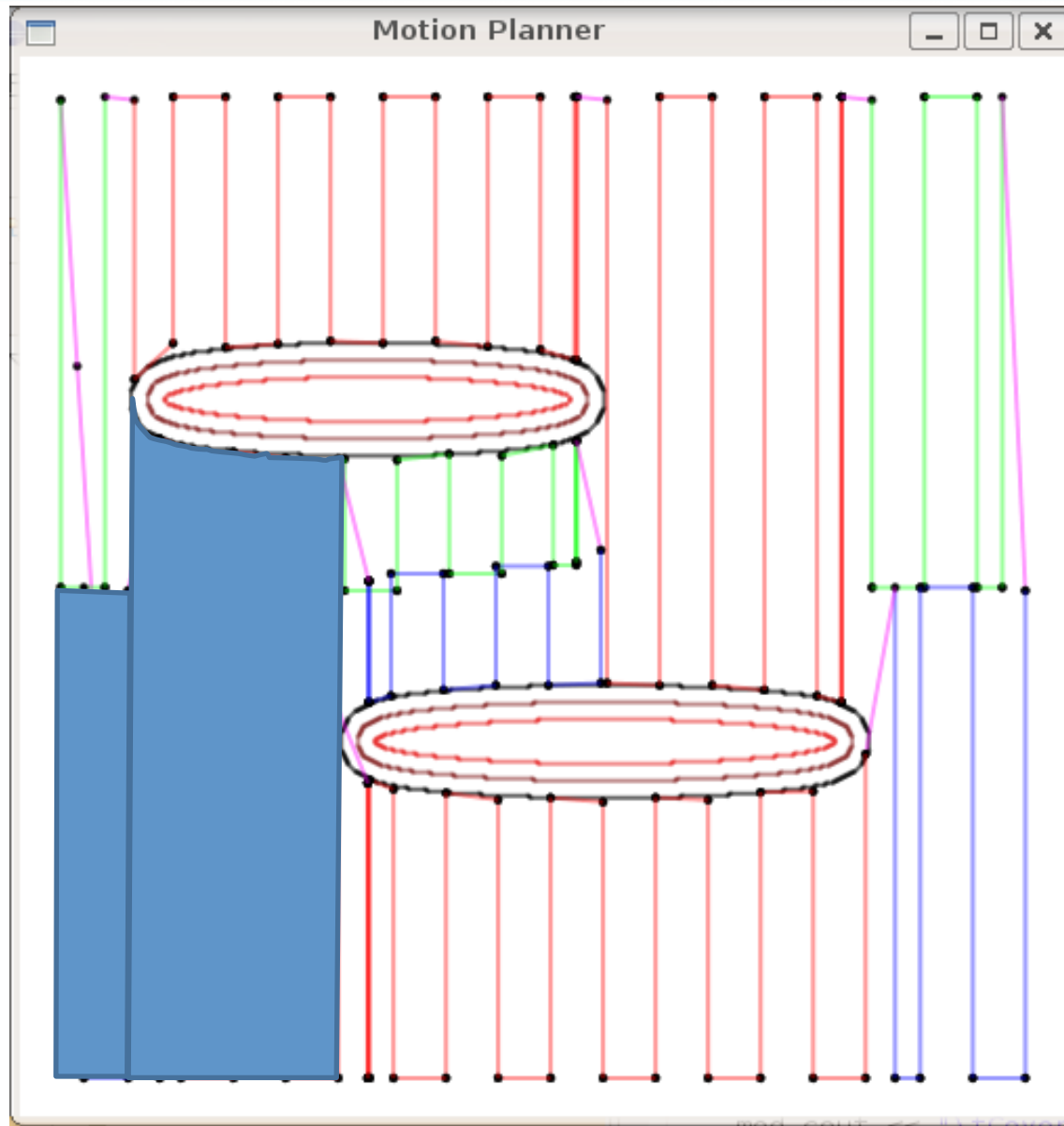
Example



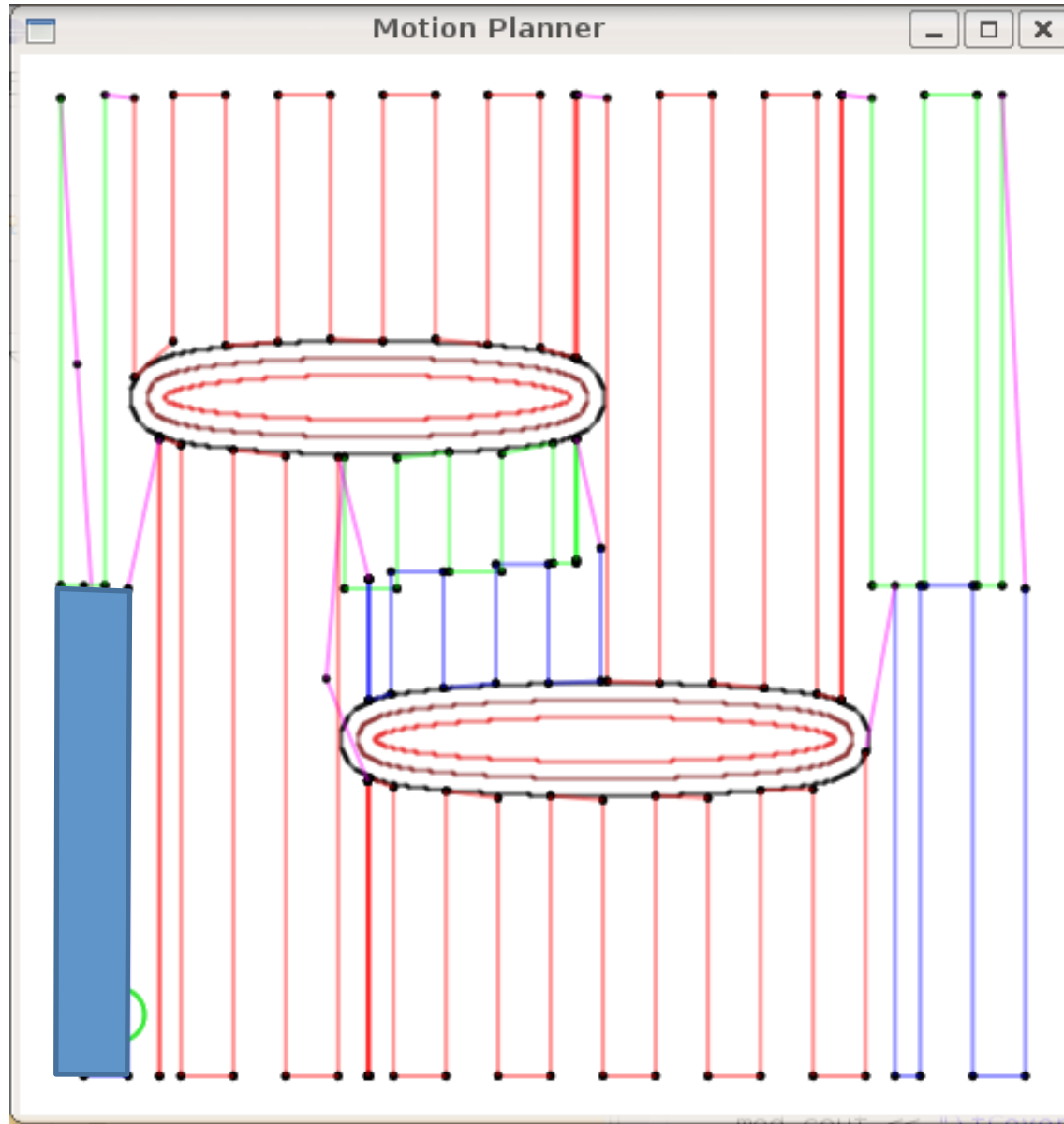
Example



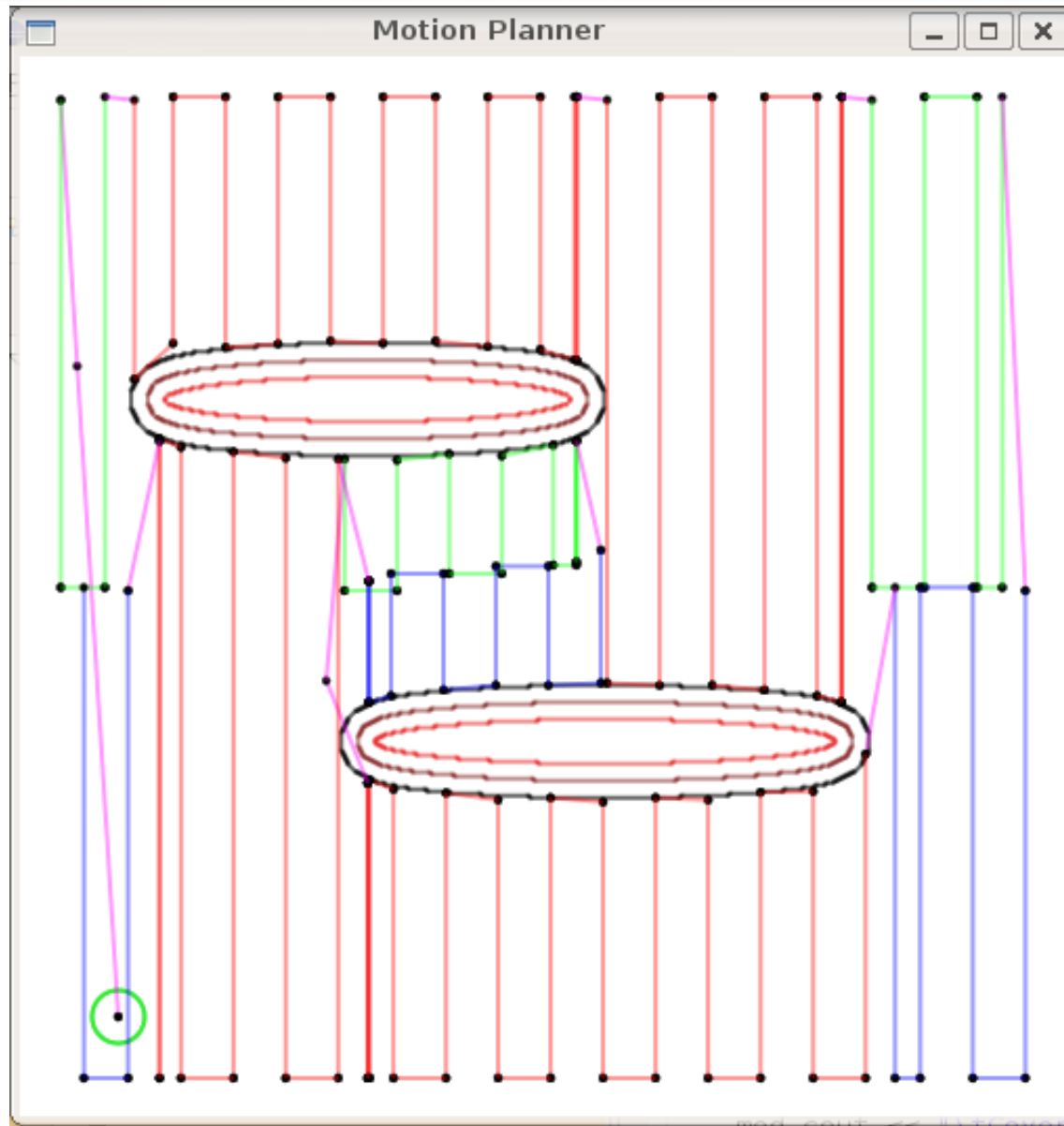
Example



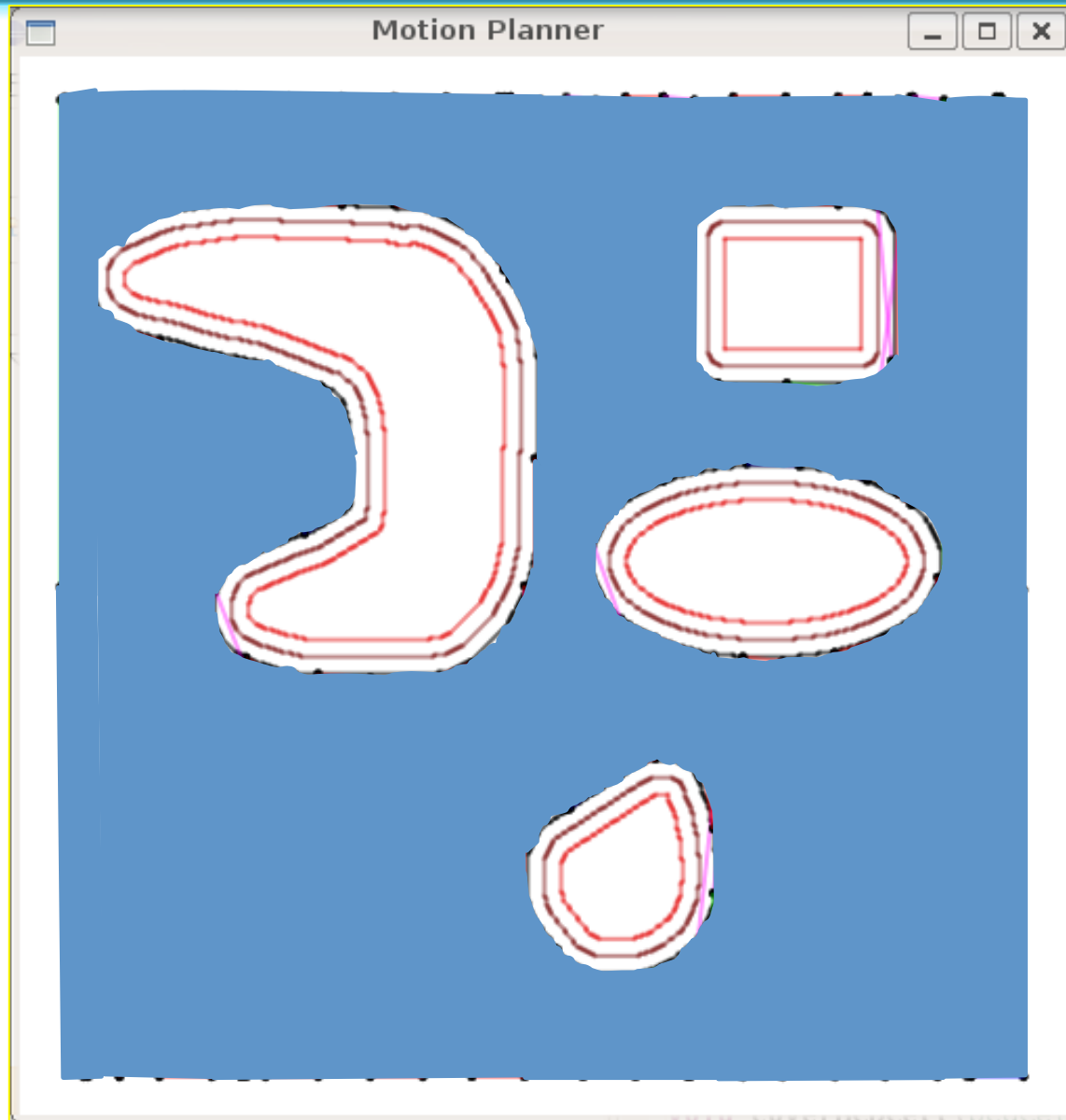
Example



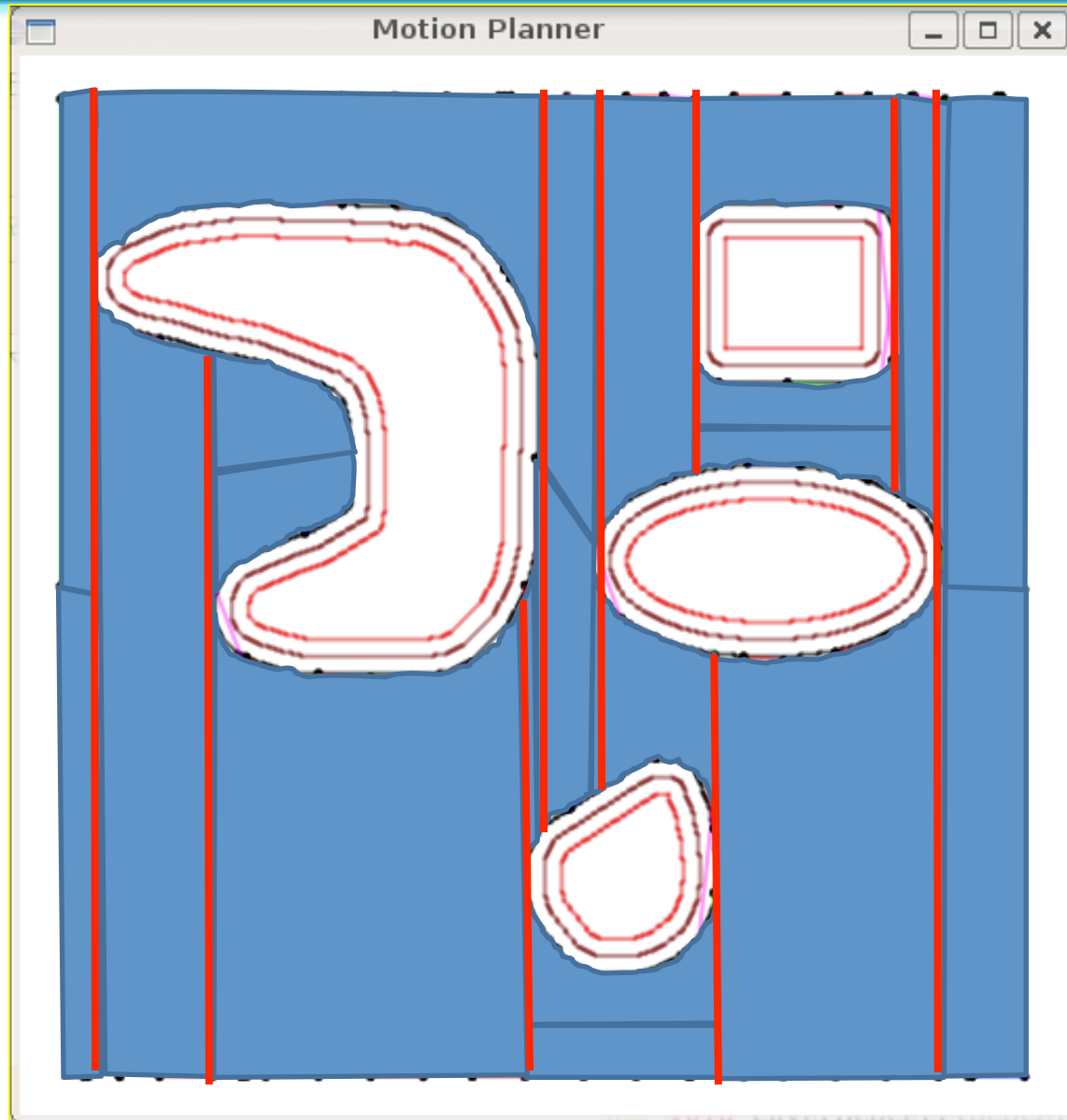
Example



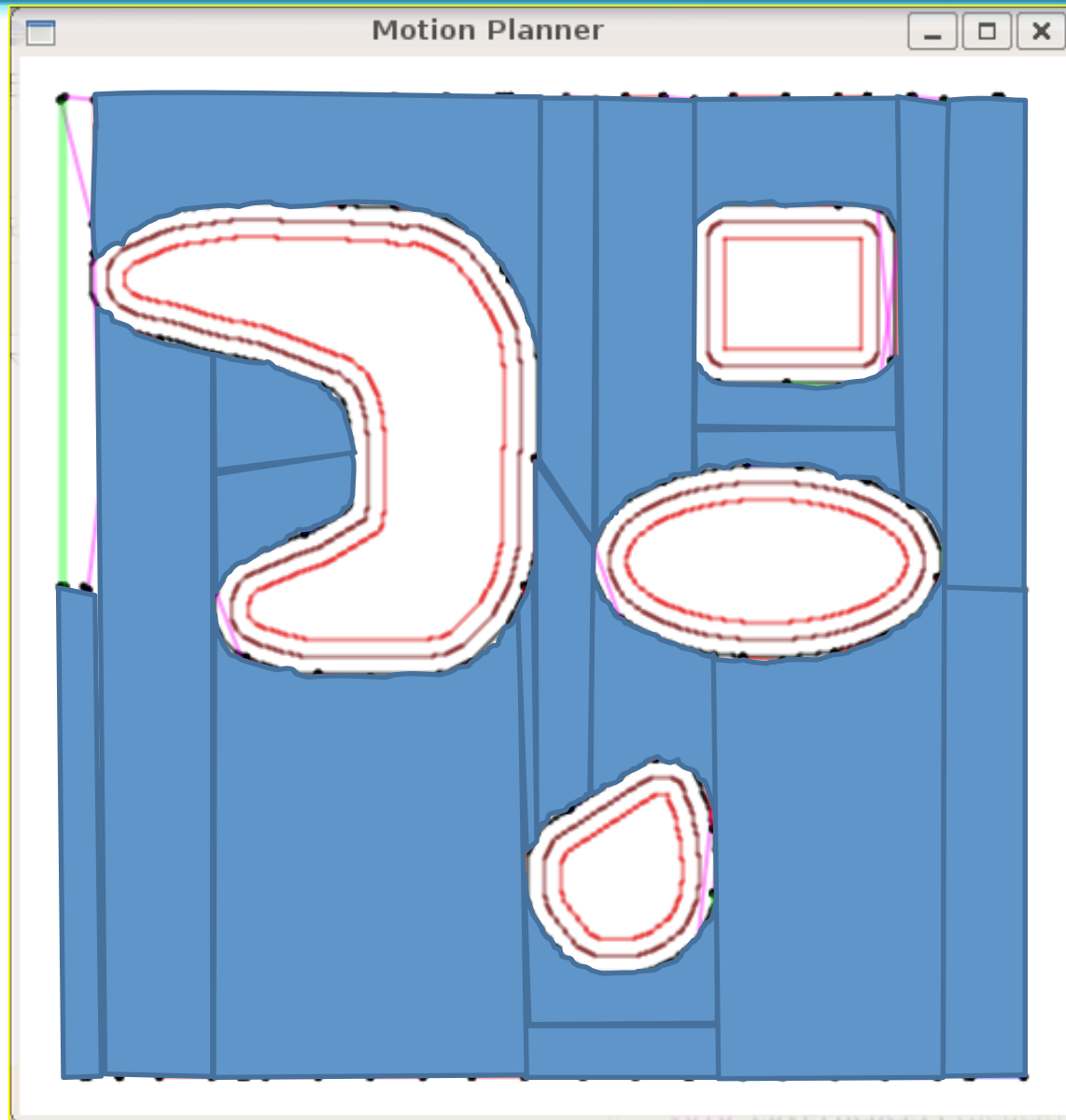
Example 2



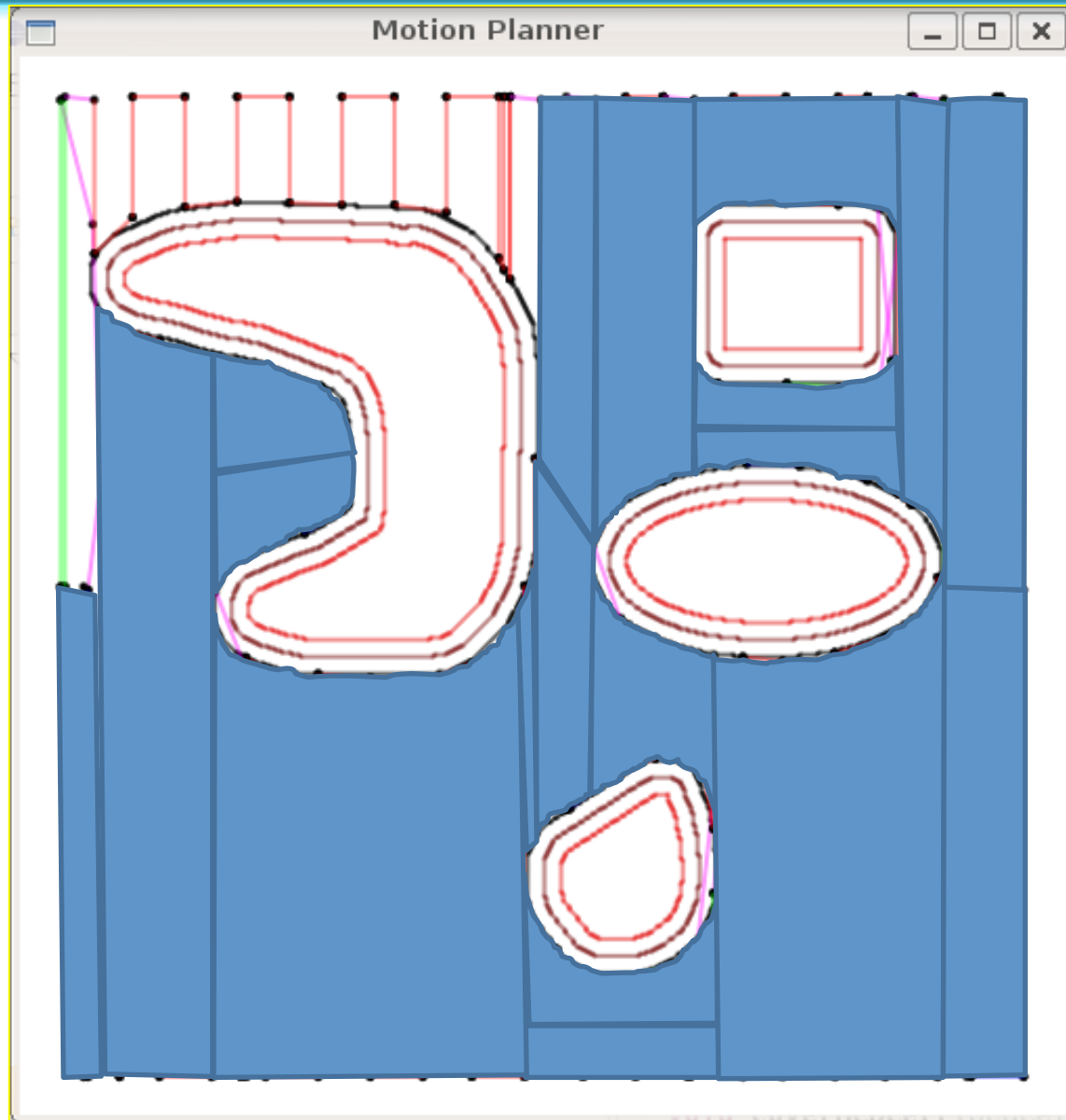
Example 2 Boustrophedon Decomp.



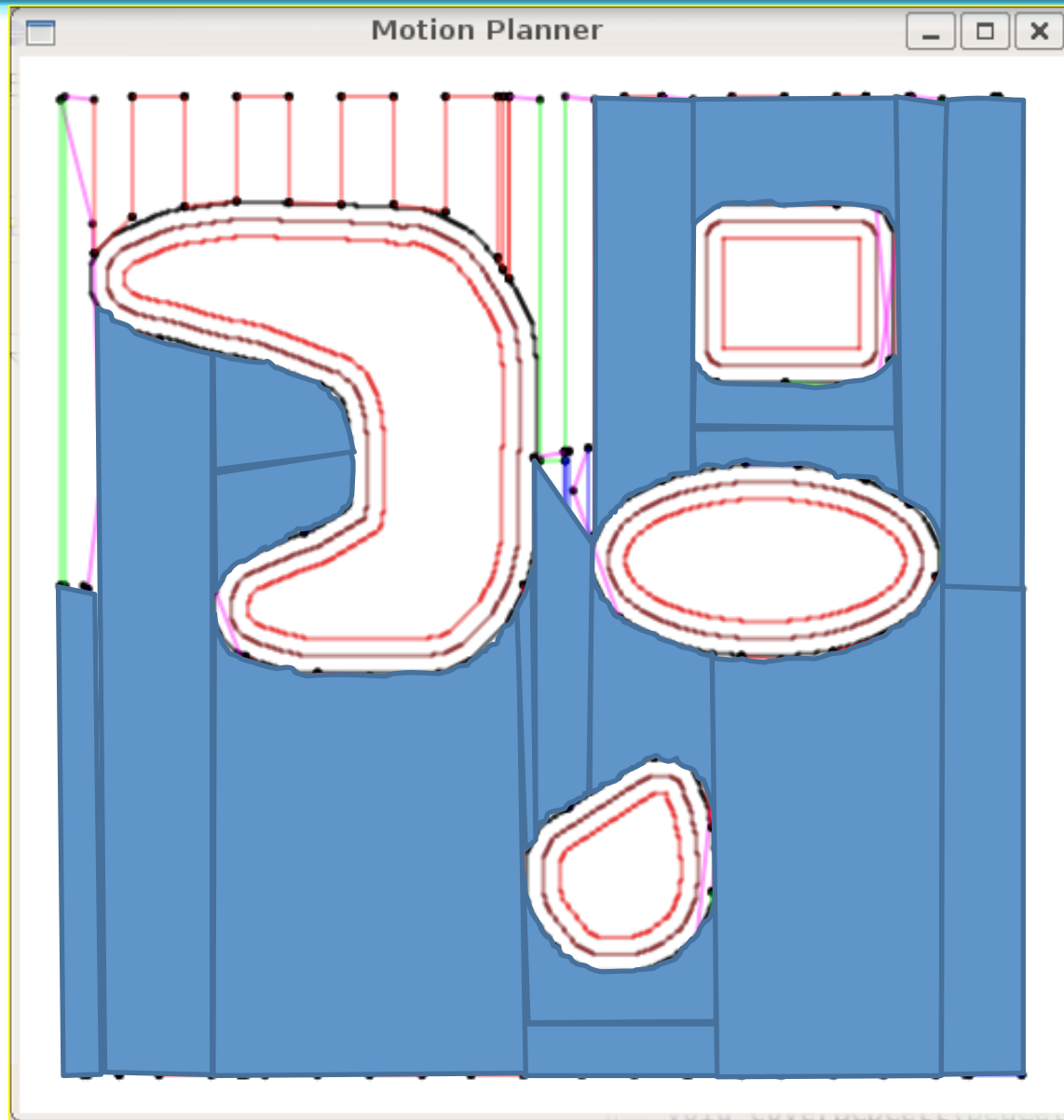
Example 2



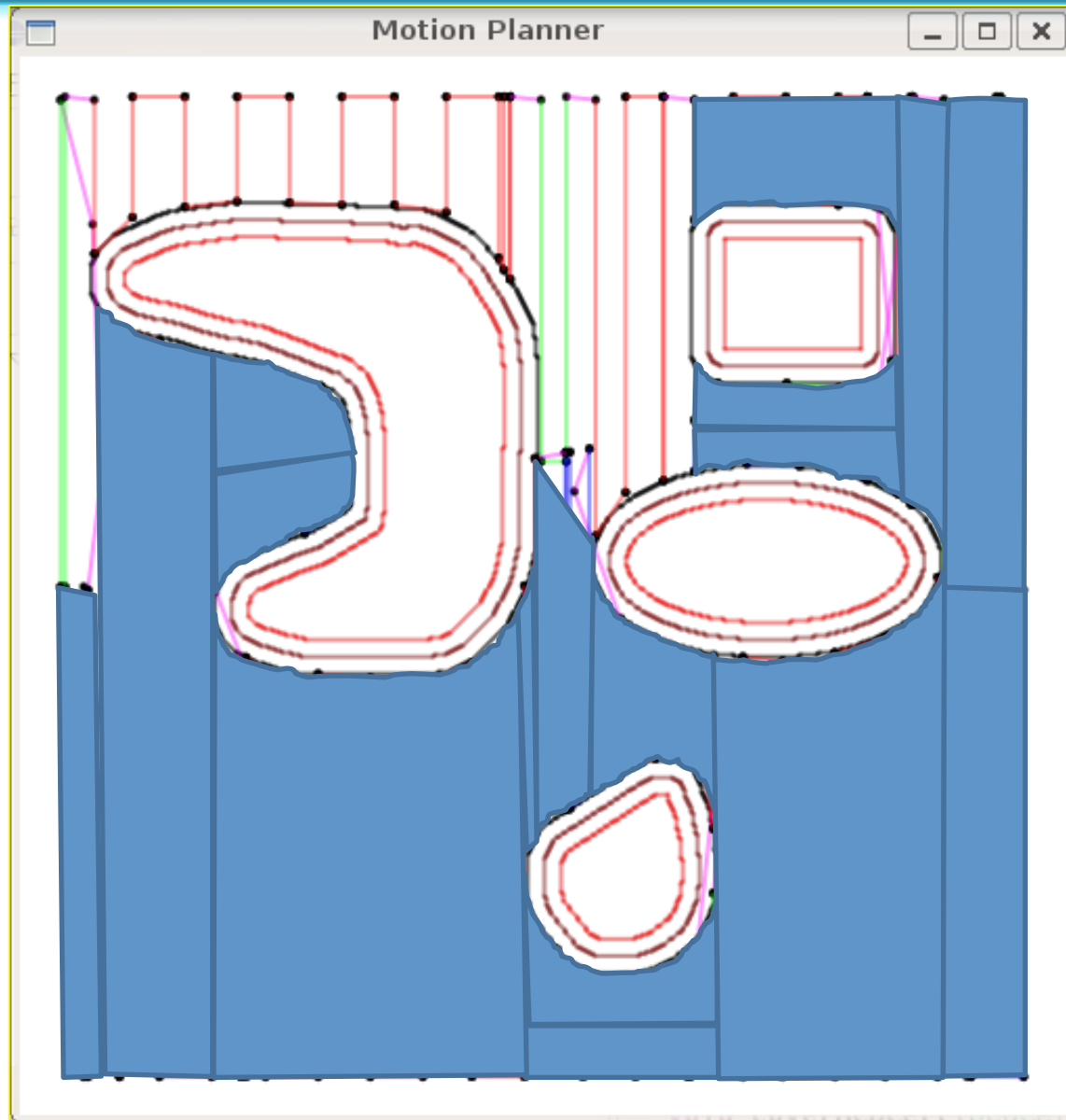
Example 2



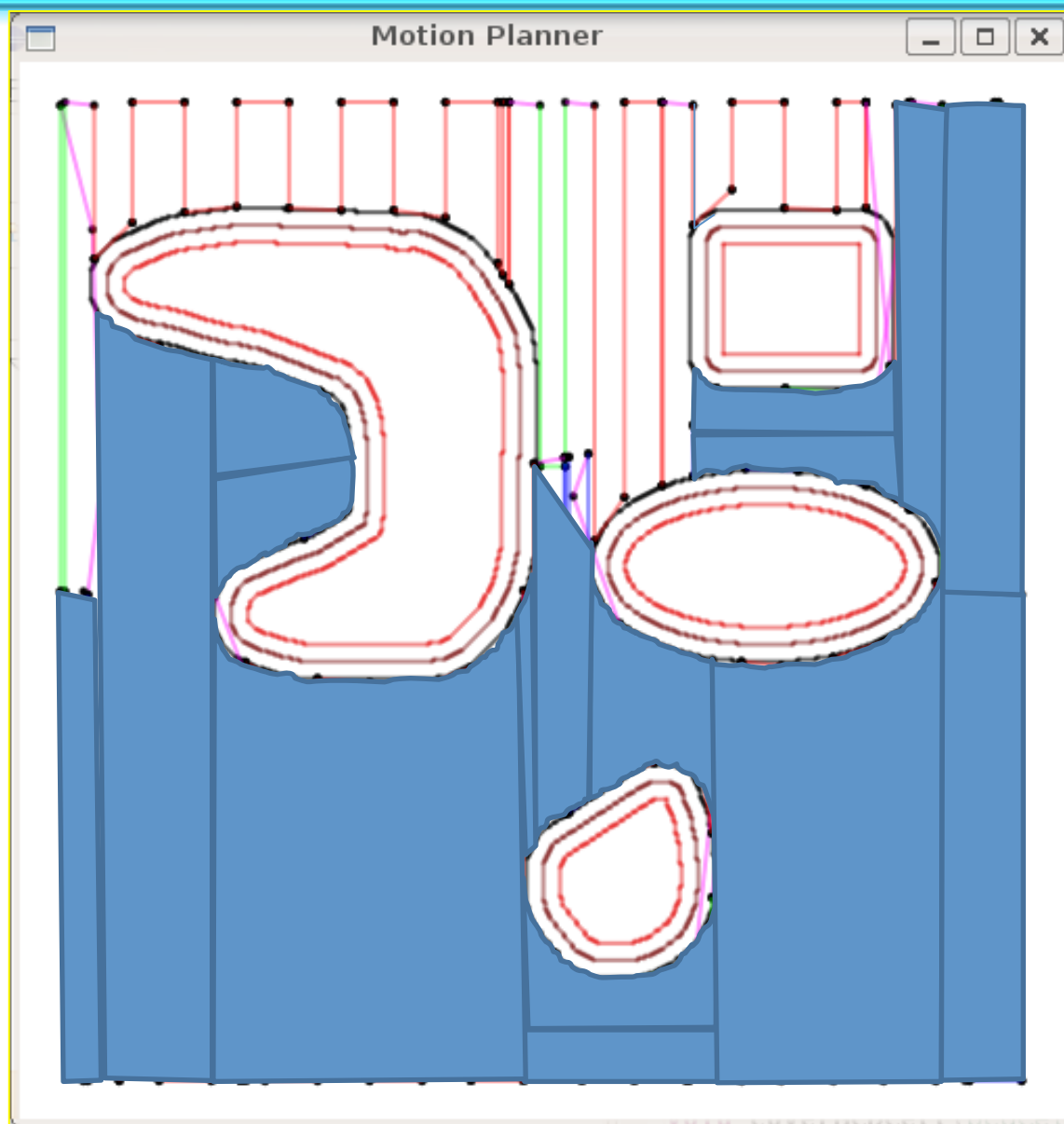
Example 2



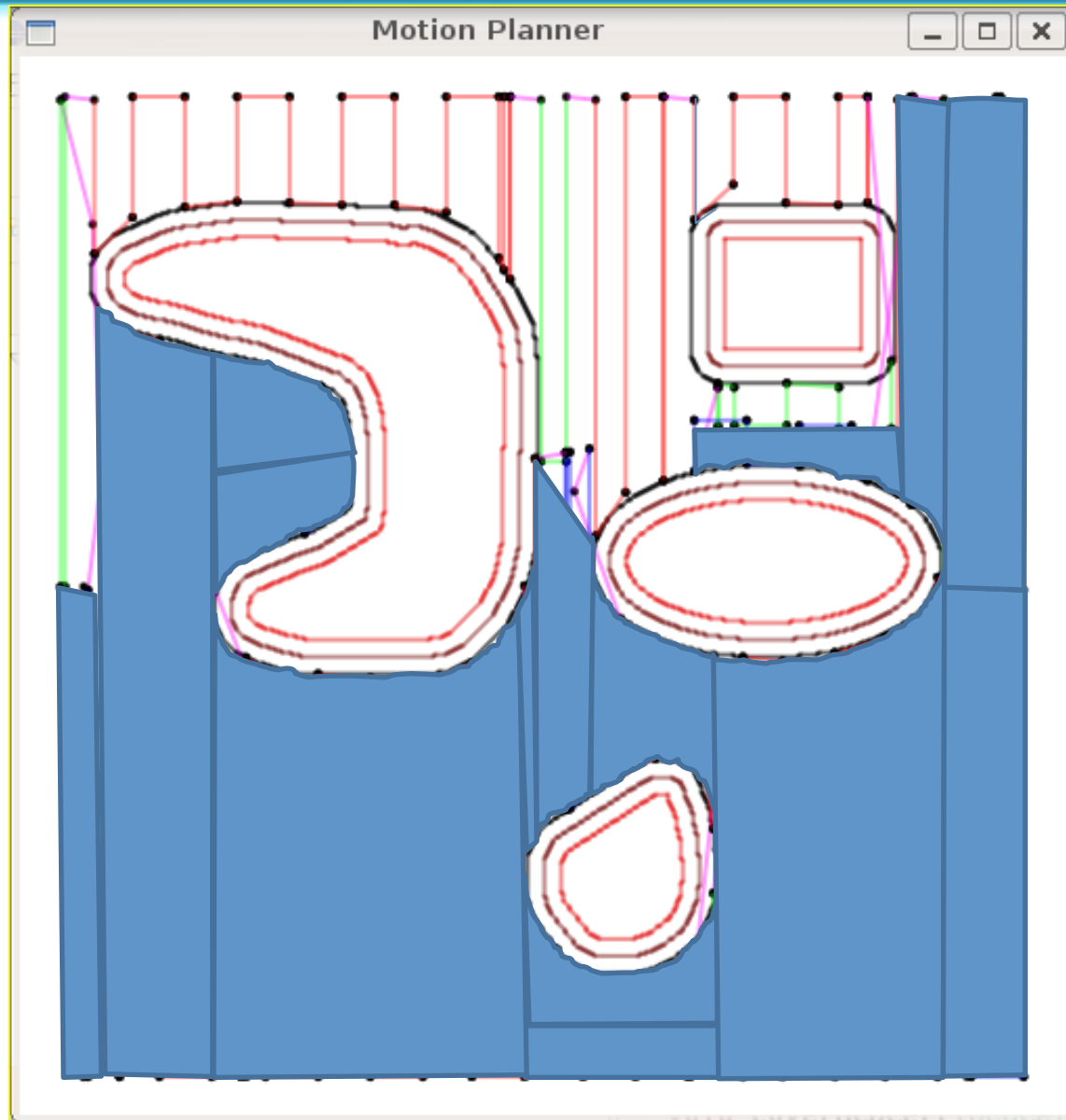
Example 2



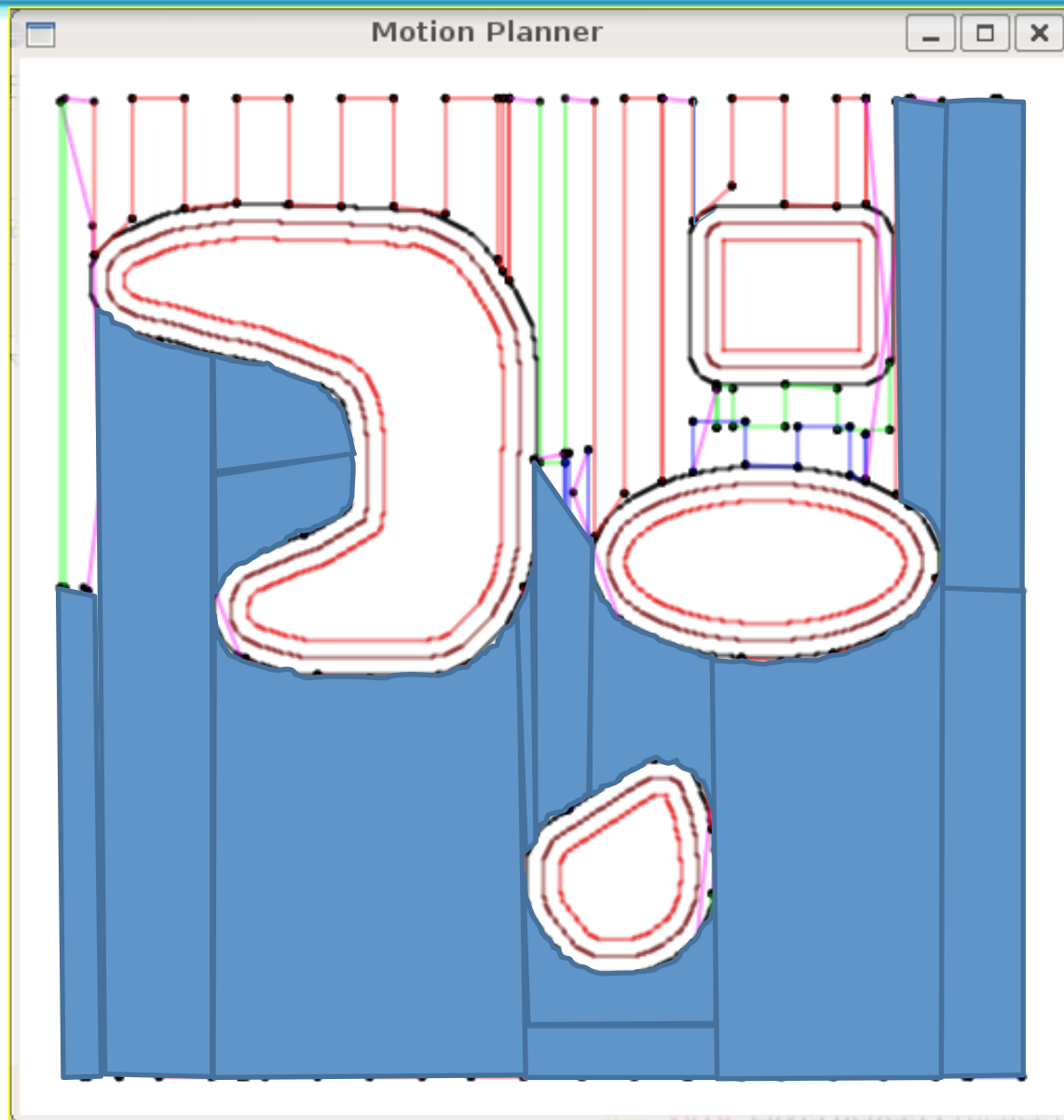
Example 2



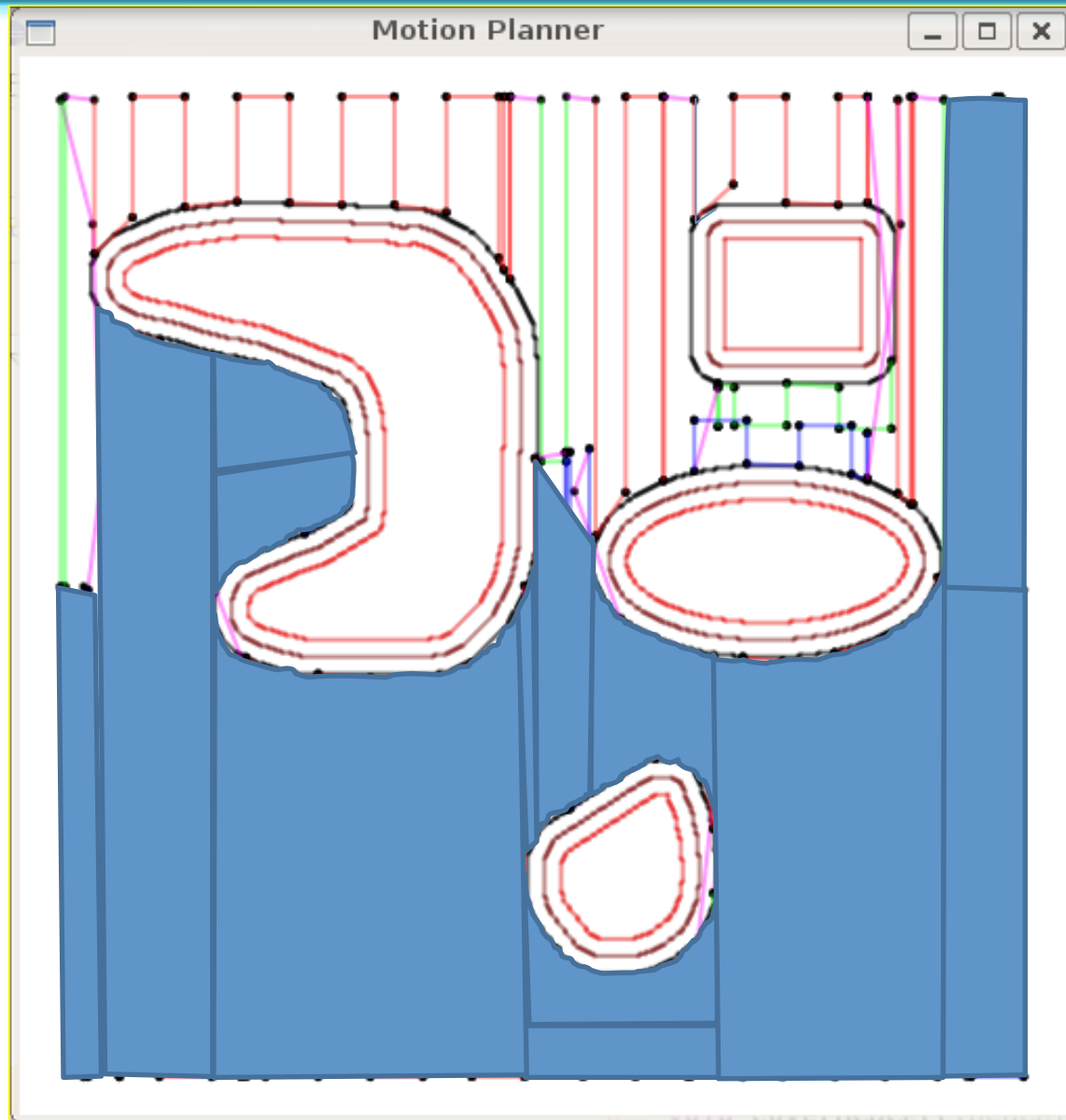
Example 2



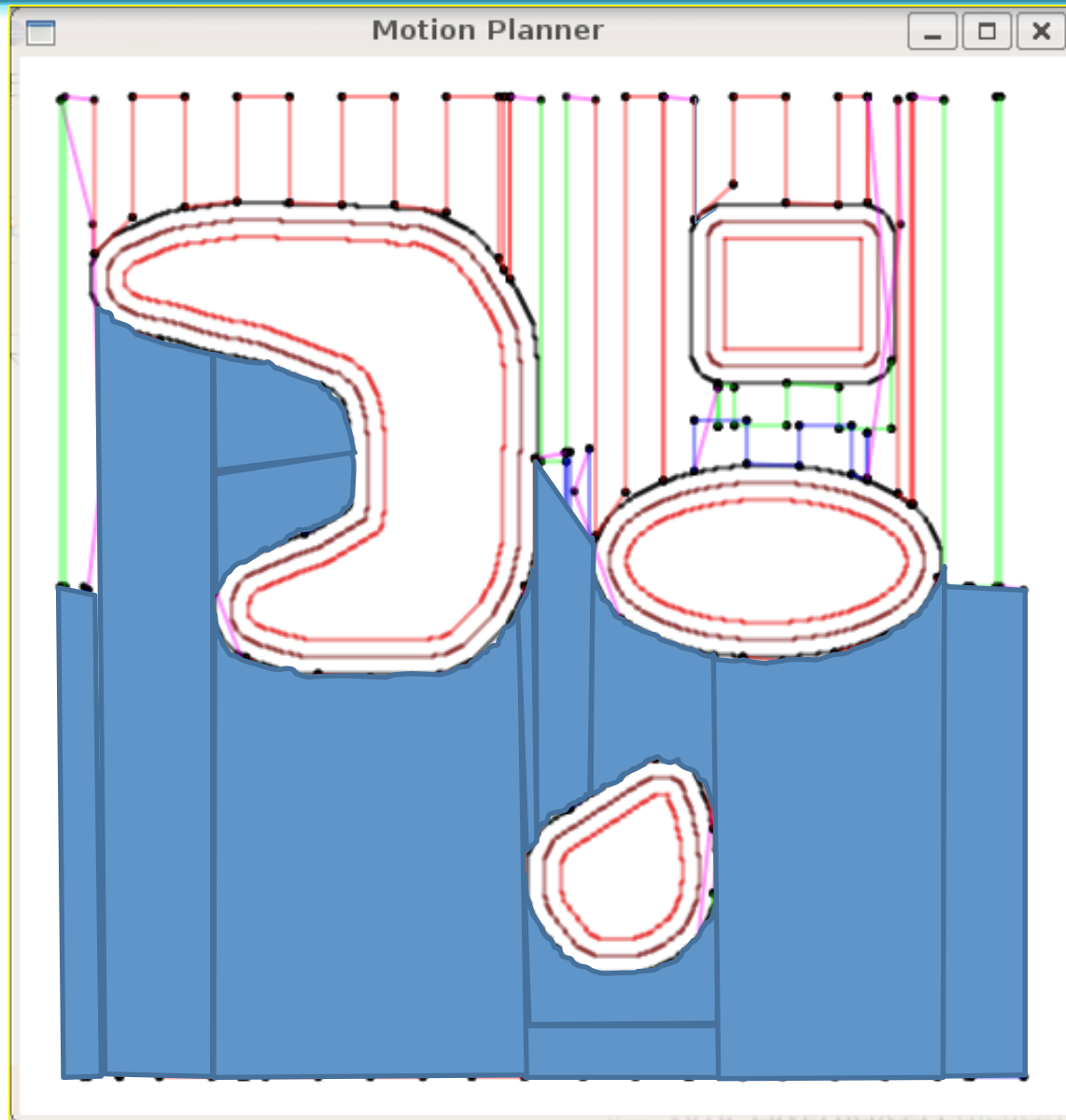
Example 2



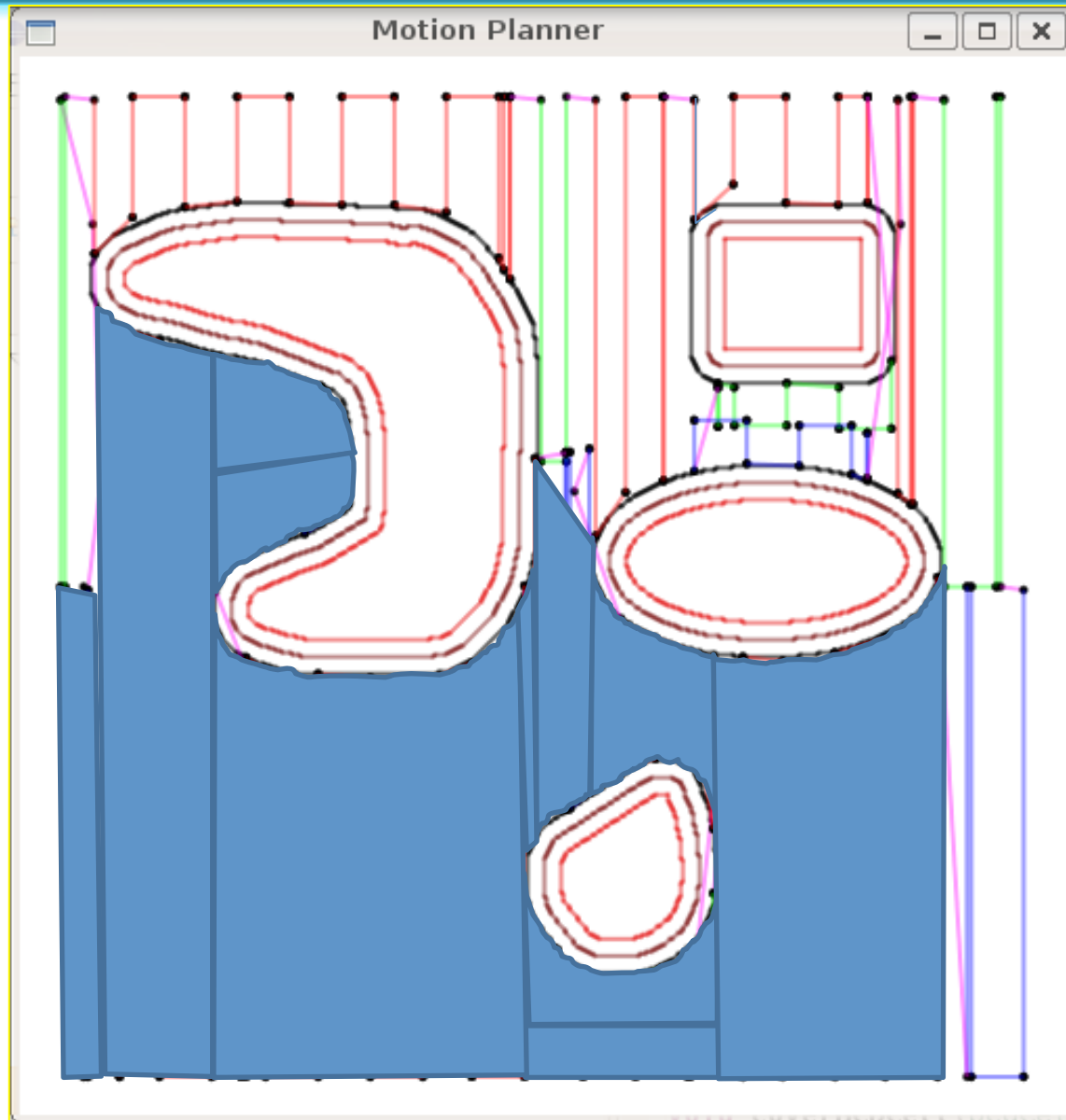
Example 2



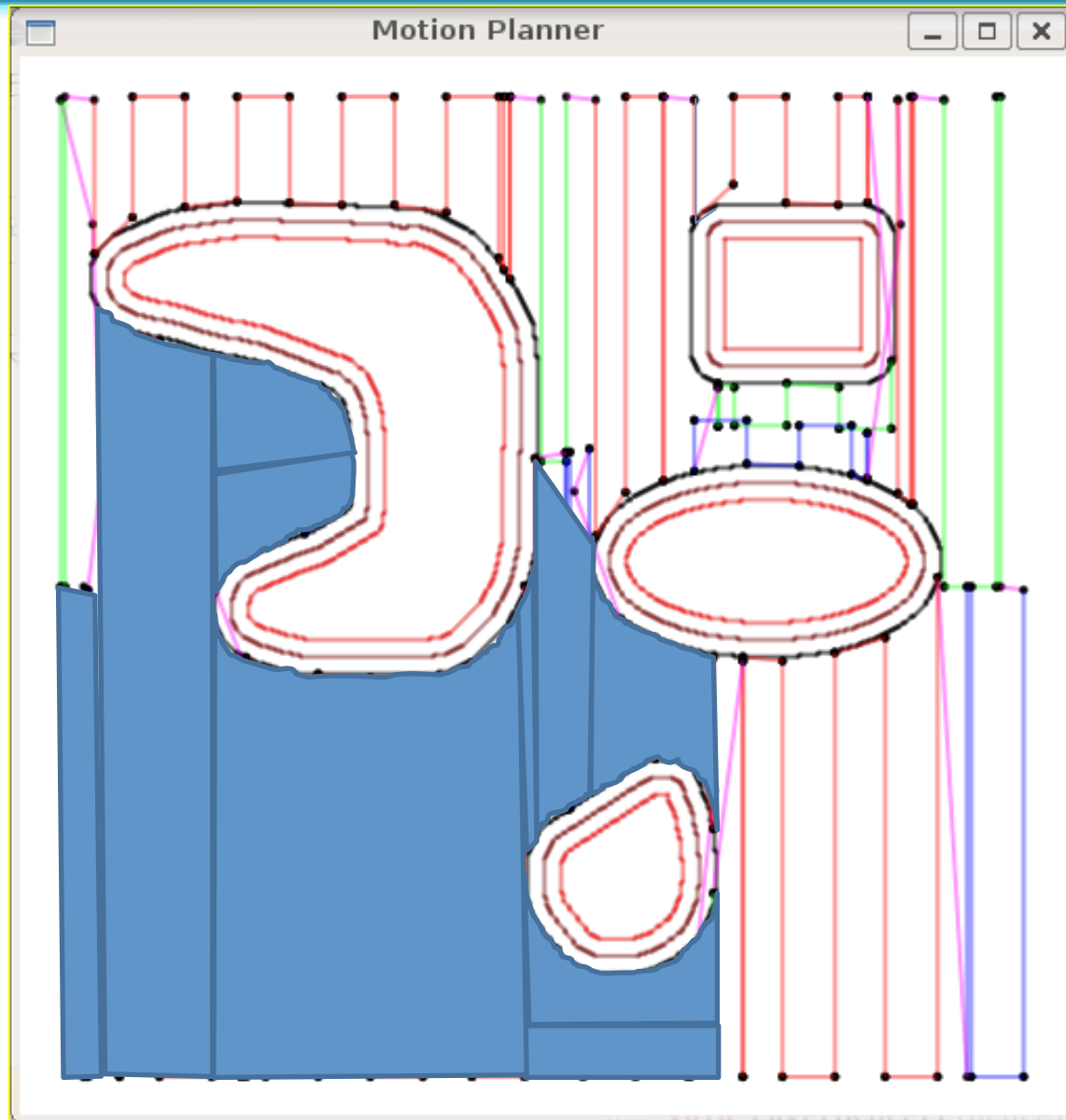
Example 2



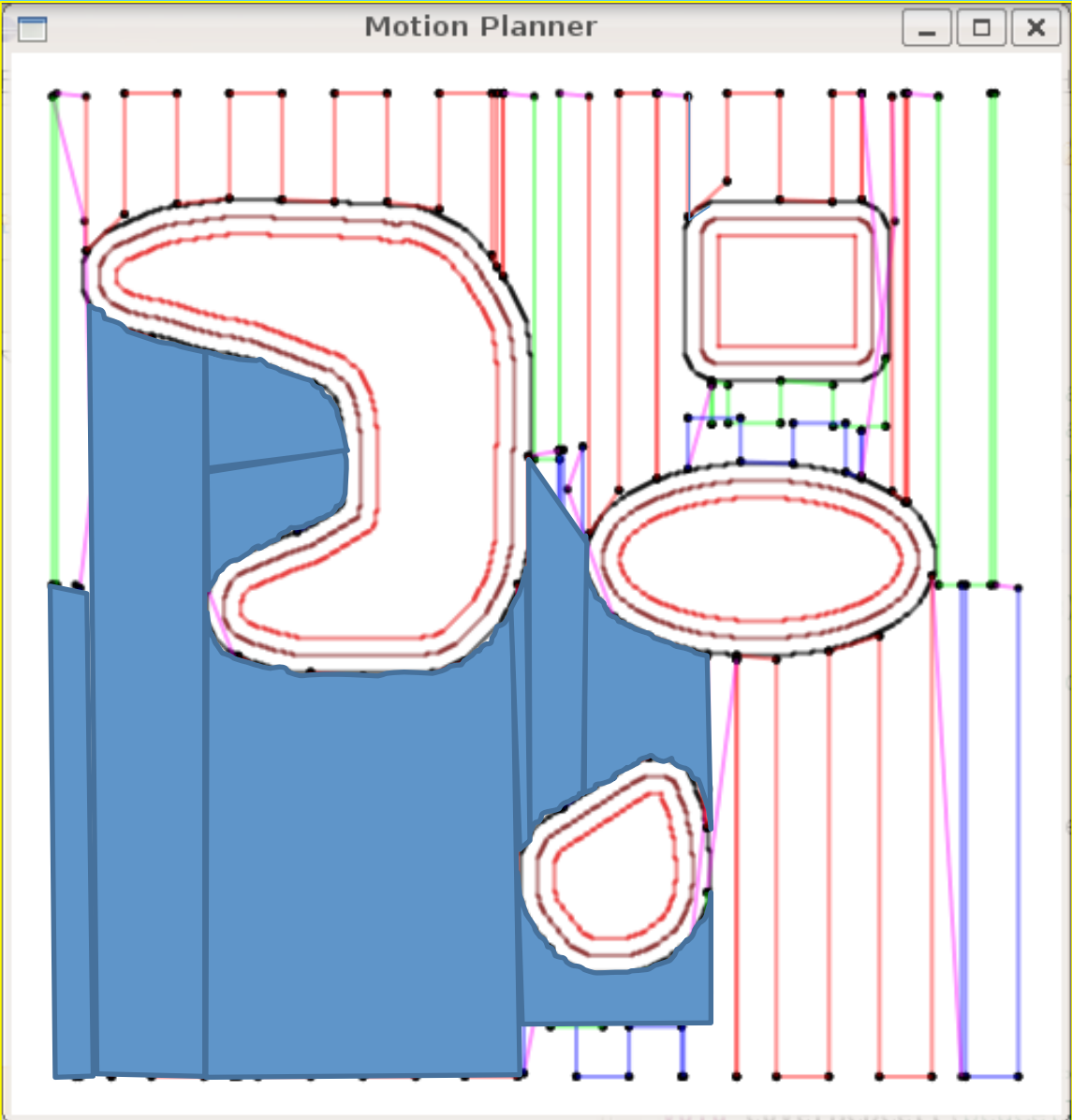
Example 2



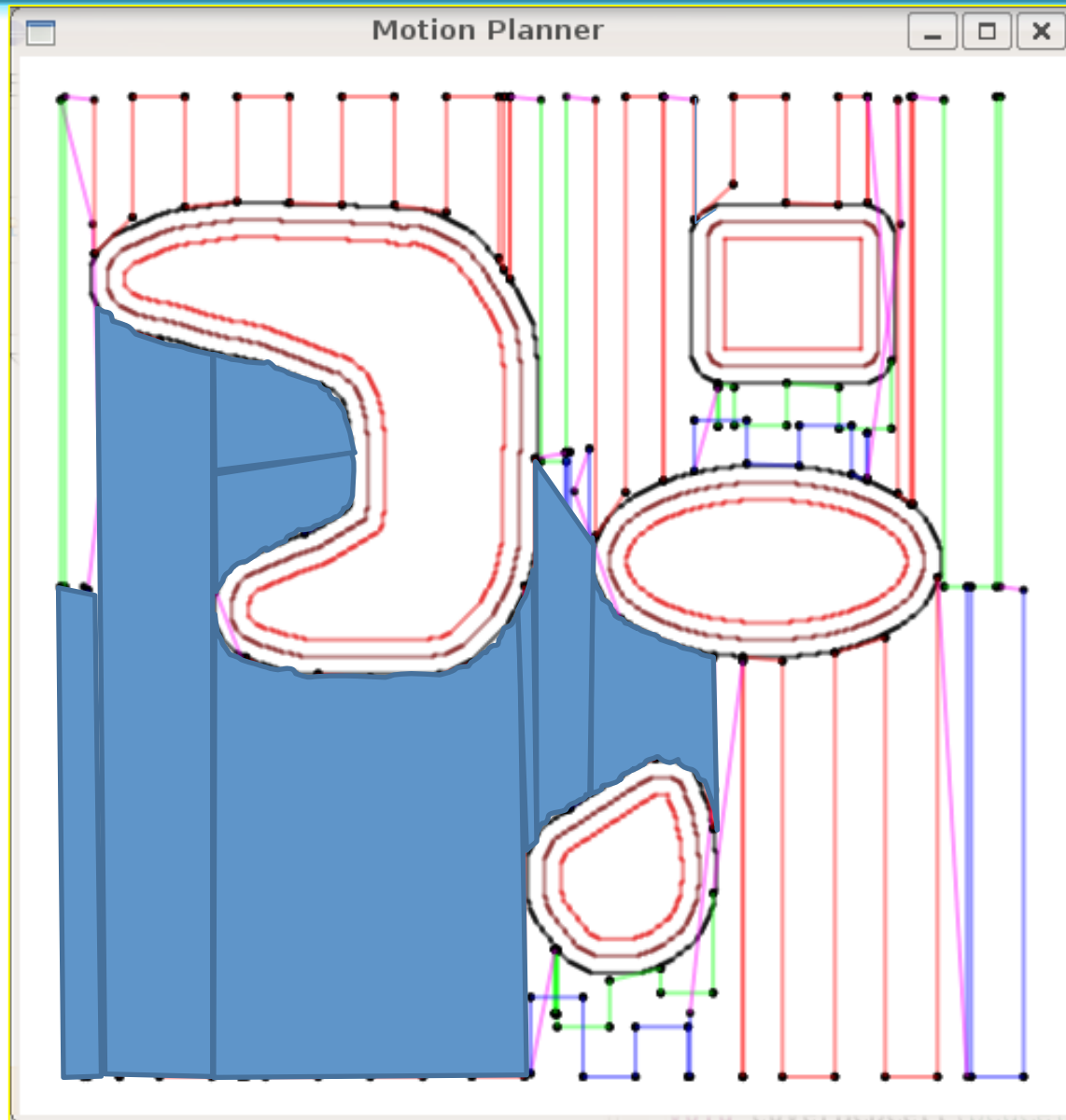
Example 2



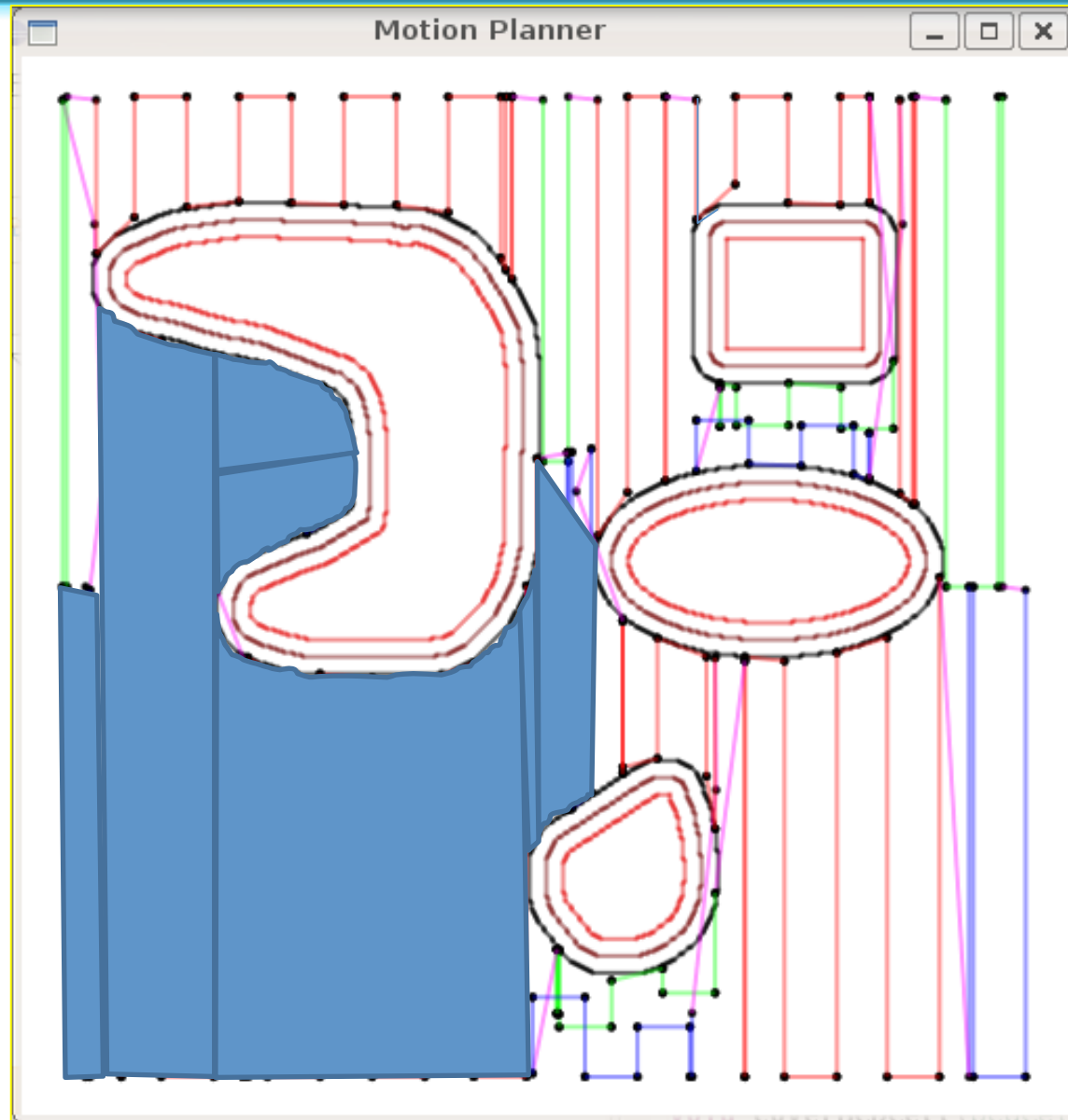
Example 2



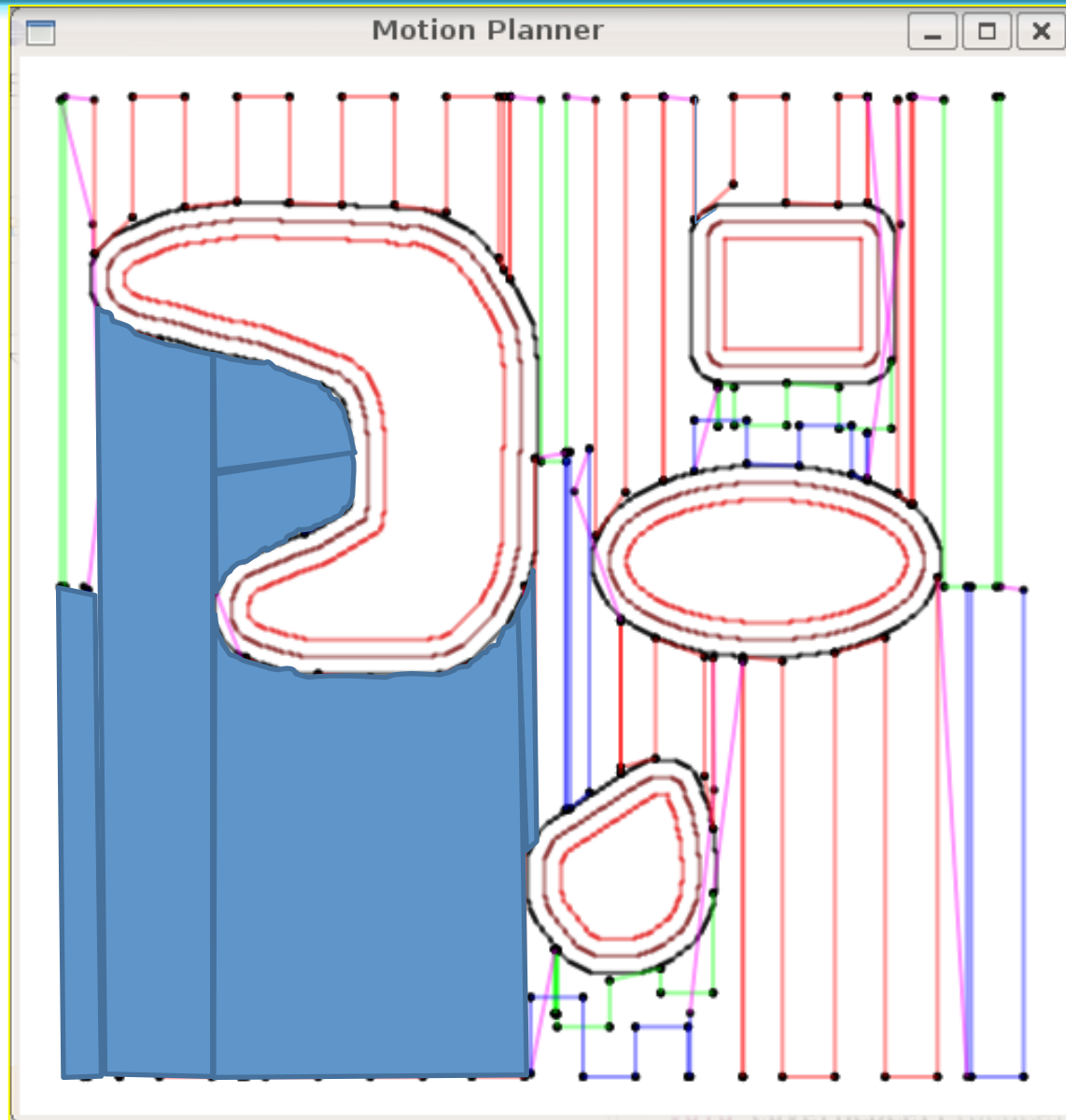
Example 2



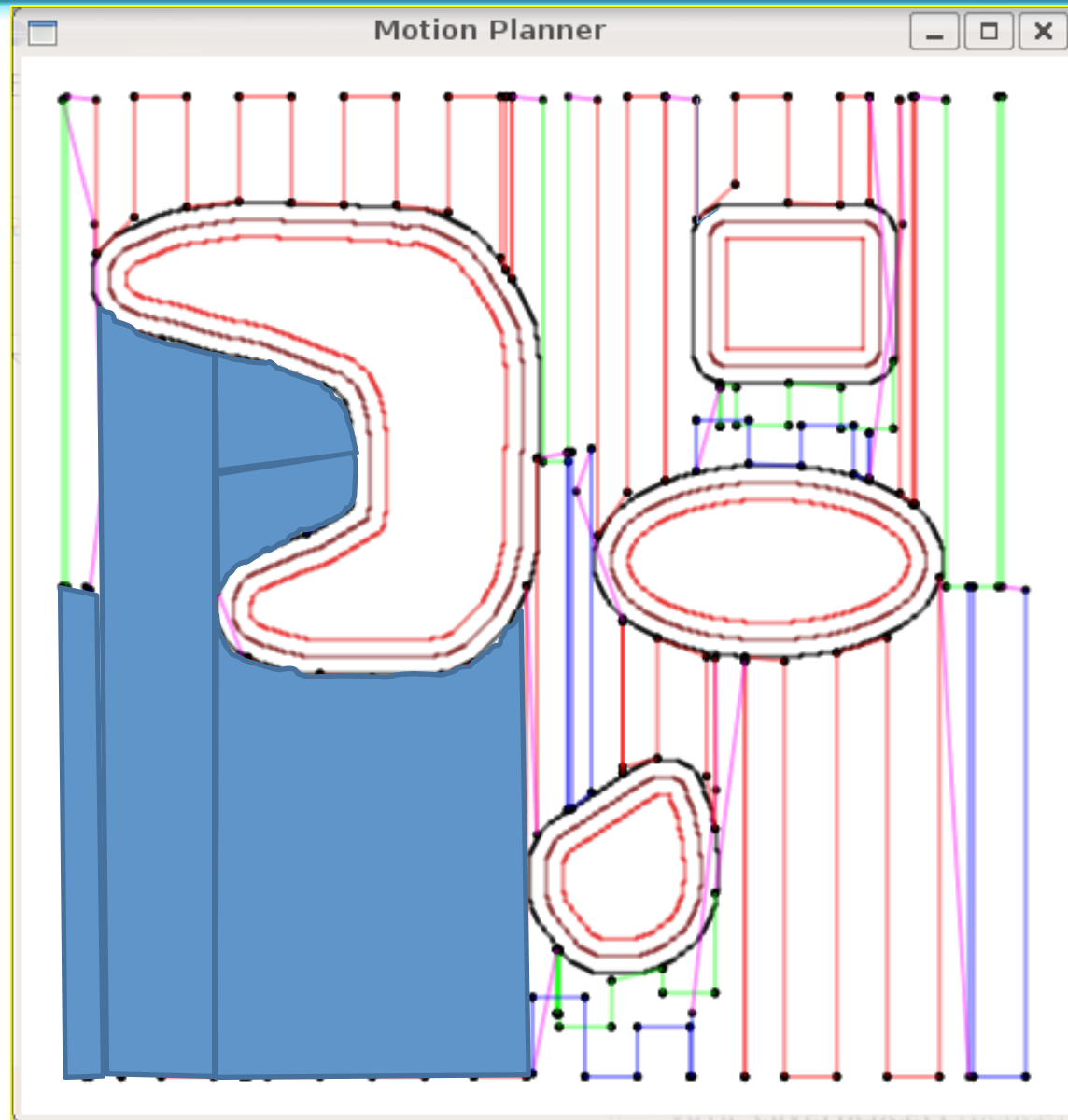
Example 2



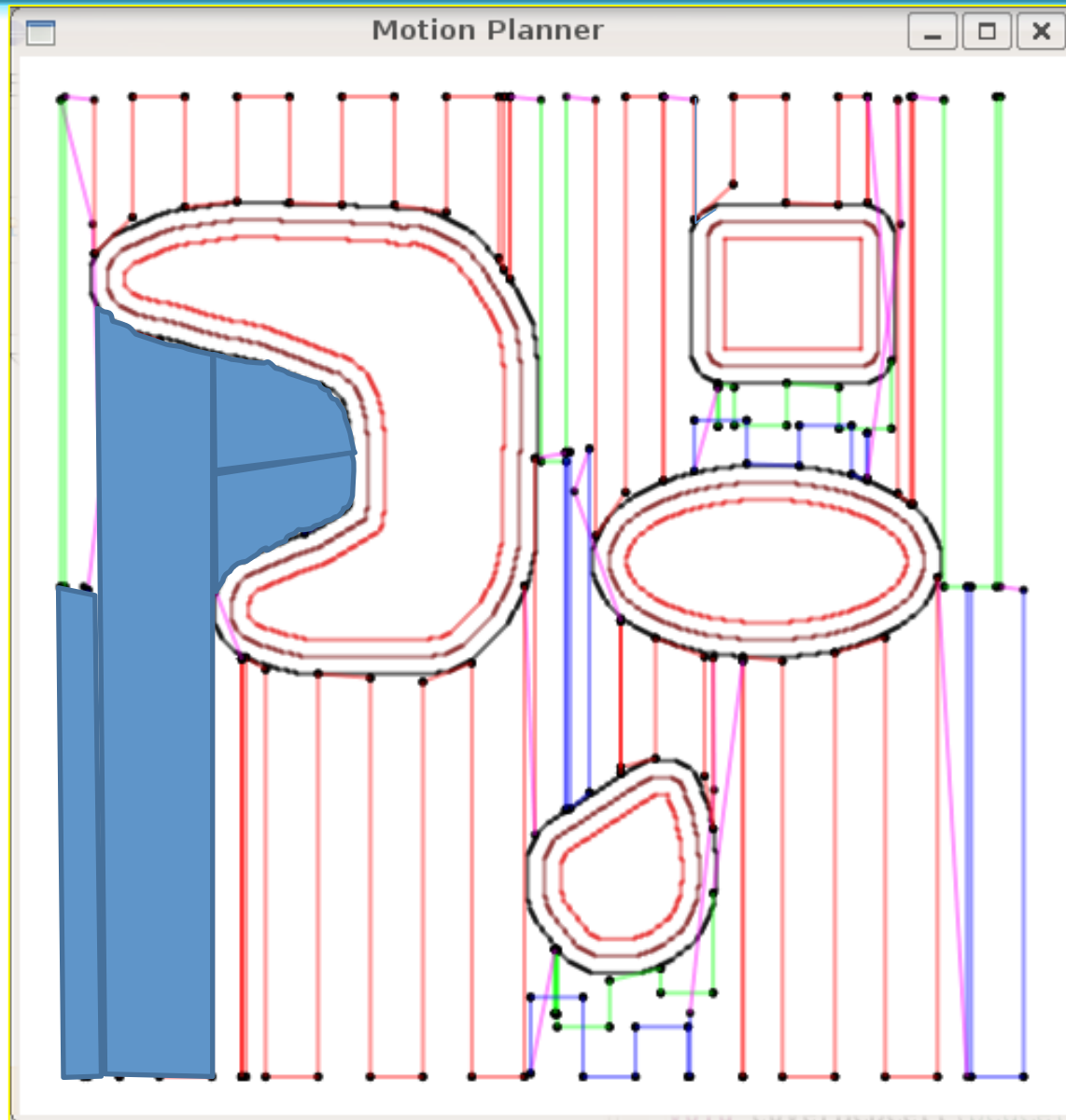
Example 2



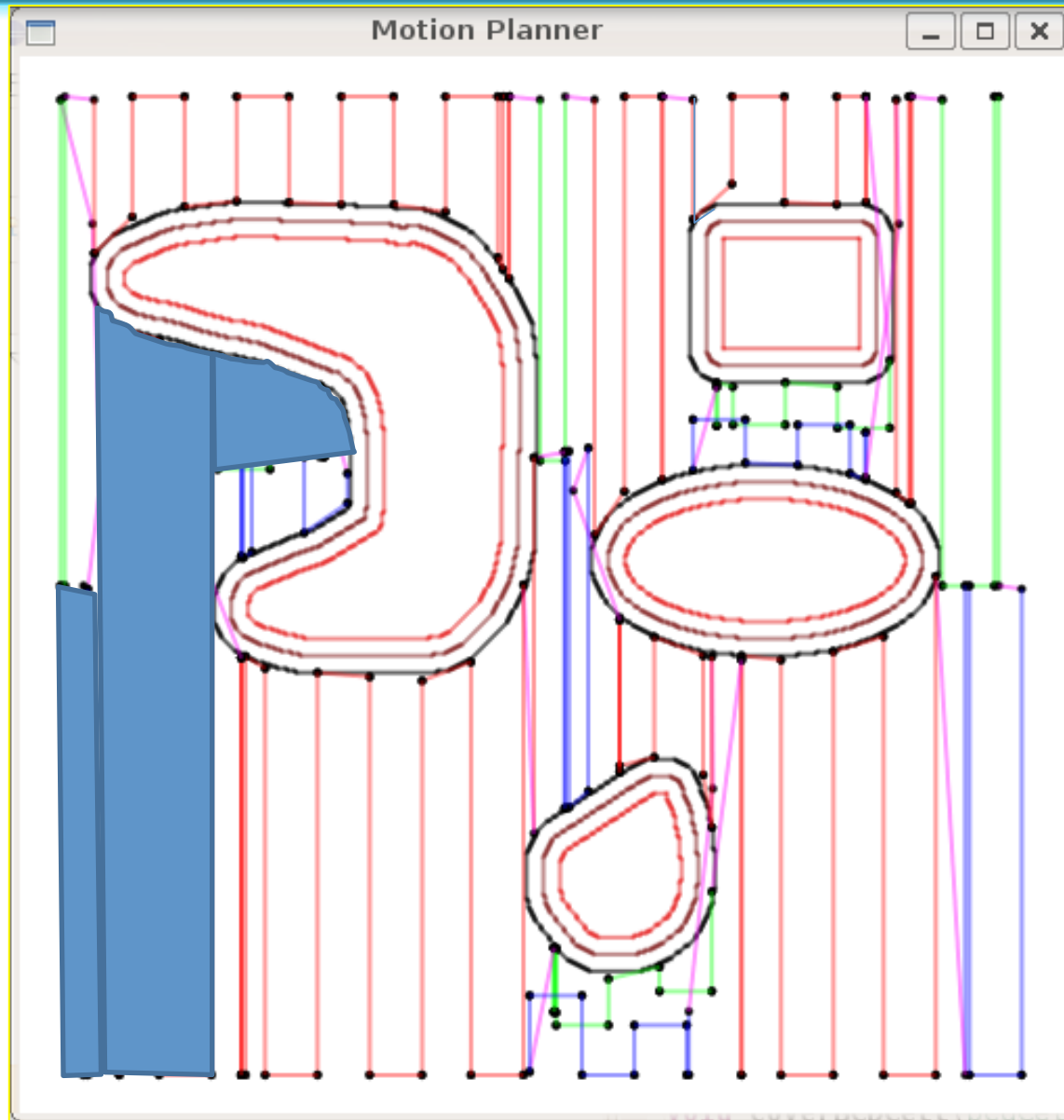
Example 2



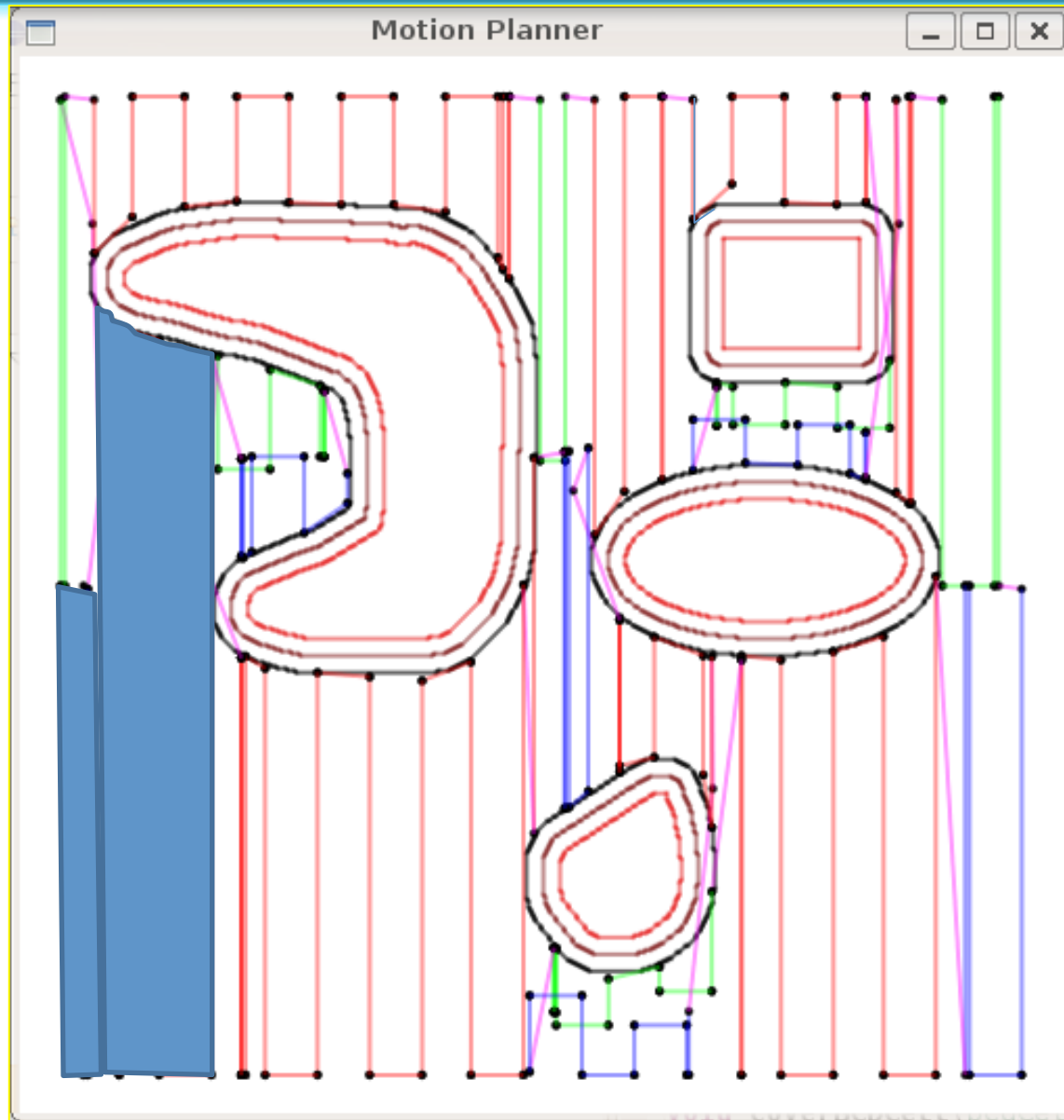
Example 2



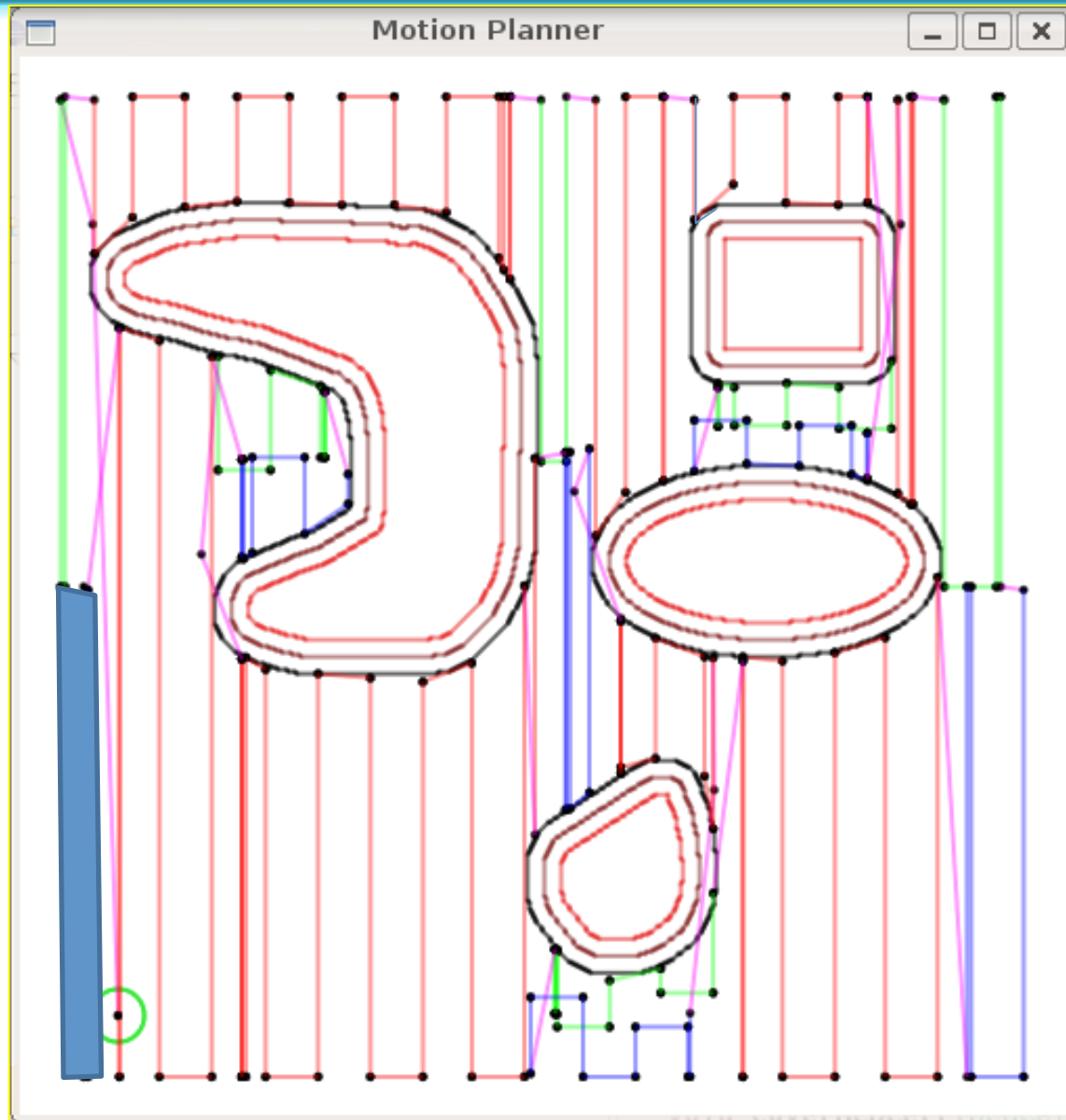
Example 2



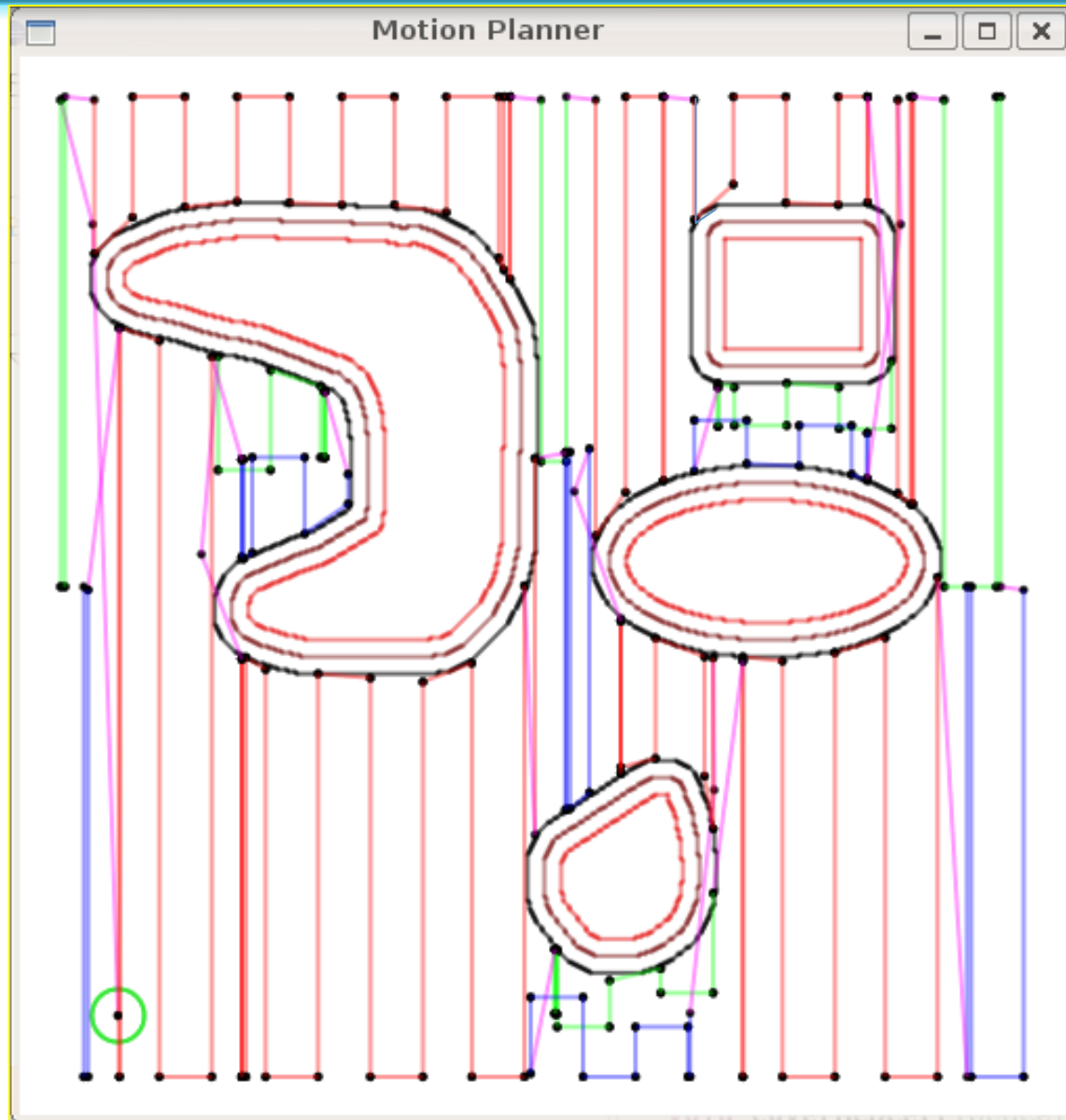
Example 2



Example 2



Example 2



UAV-Efficient Coverage



UAV-Efficient Coverage

- UAVs non-holonomic constraints require special trajectory planning
- 120 Km of flight during coverage

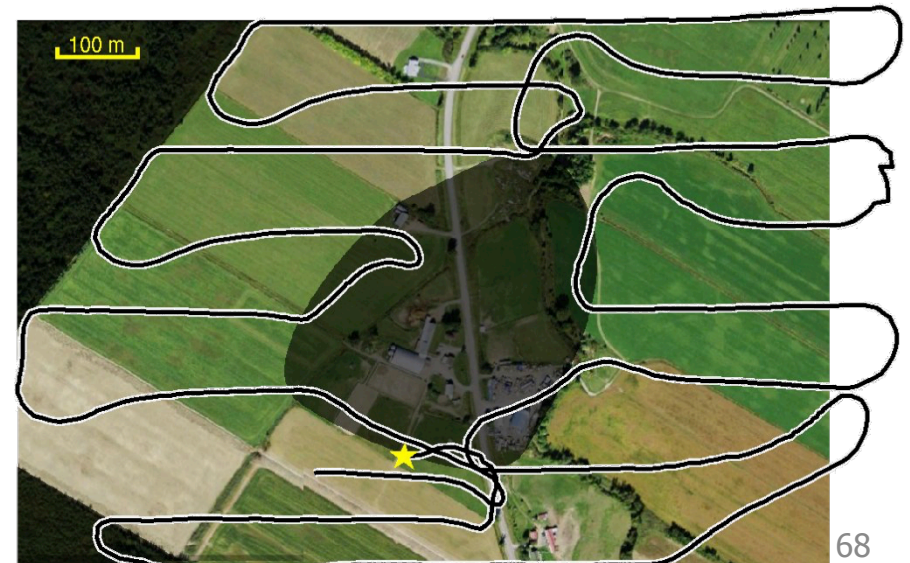
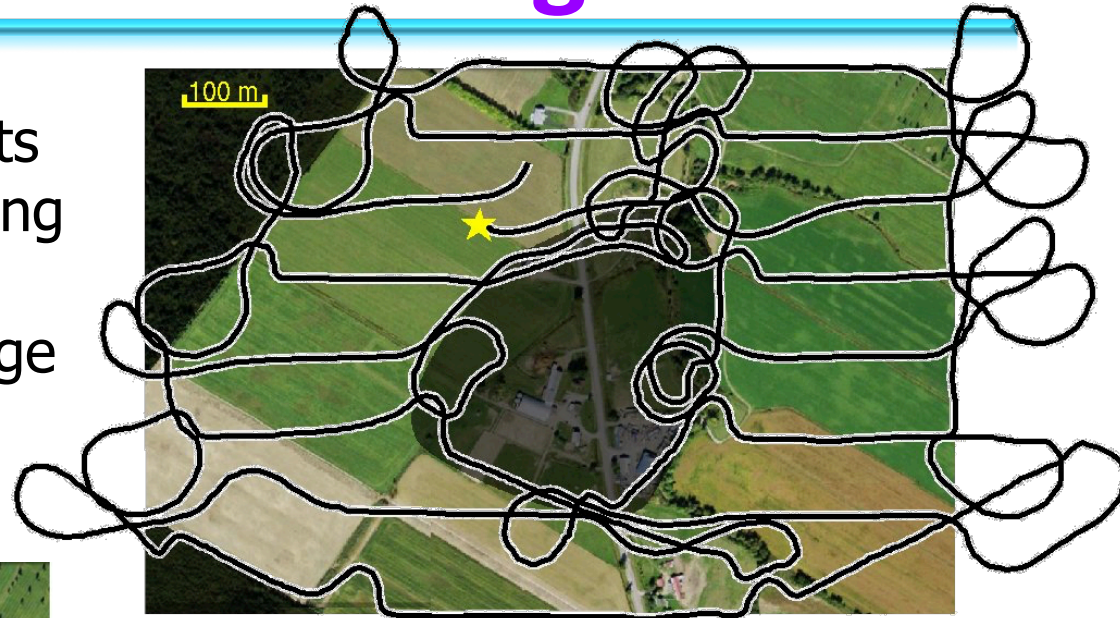


Image Mosaic



Video at ICRA 2011

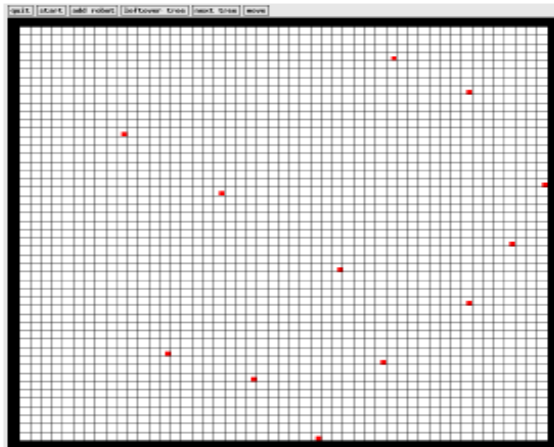
Complete Optimal Terrain Coverage using an Unmanned Aerial Vehicle

Anqi Xu
Chatavut Viriyasuthee
Ioannis Rekleitis

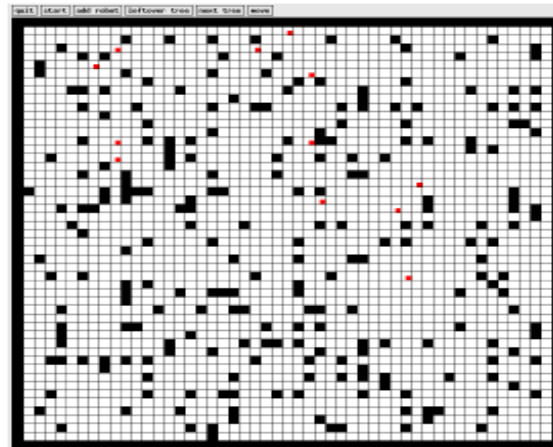


Coverage of Known Worlds

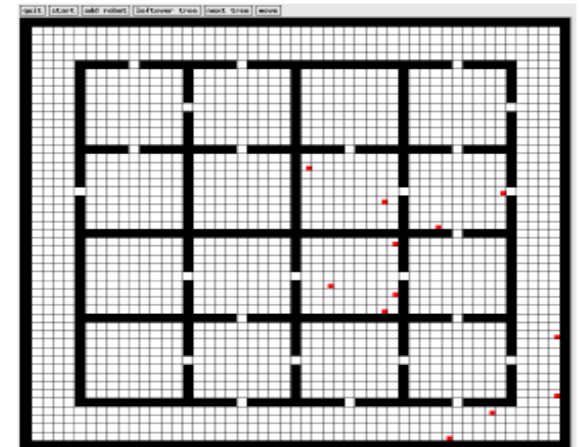
Empty Terrain



Outdoor-Like Terrain



Indoor-Like Terrain

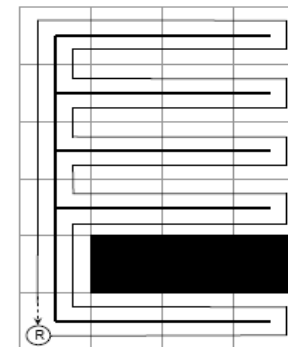


From: X. Zheng and S. Koenig. Robot Coverage of Terrain with Non-Uniform Traversability. In Proc. of the IEEE Int. Conf. on Intelligent Robots and Systems (IROS), pg. 3757-3764, 2007

STC

cover time = 682

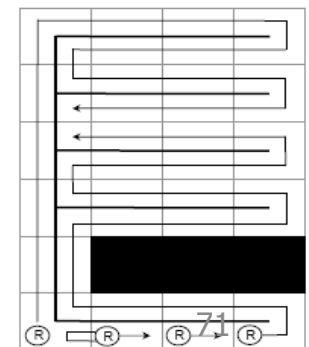
cover and return time = 688



MSTC

cover time = 332

cover and return time = 394



Cell-Decomposition Methods

Two families of methods:

- Exact cell decomposition
- Approximate cell decomposition

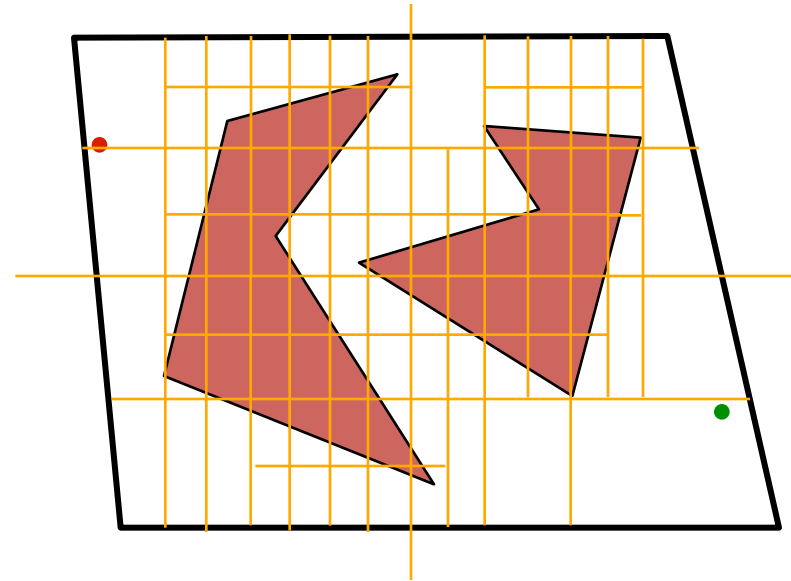
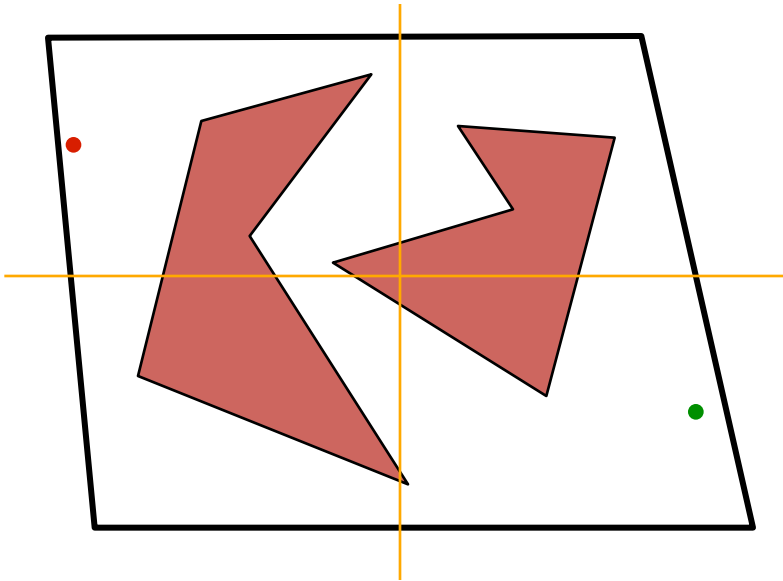
F is represented by a collection of non-overlapping cells whose union is contained in F

Examples: quadtree, octree, 2^n -tree



further decomposing...

- Approximate cell decomposition

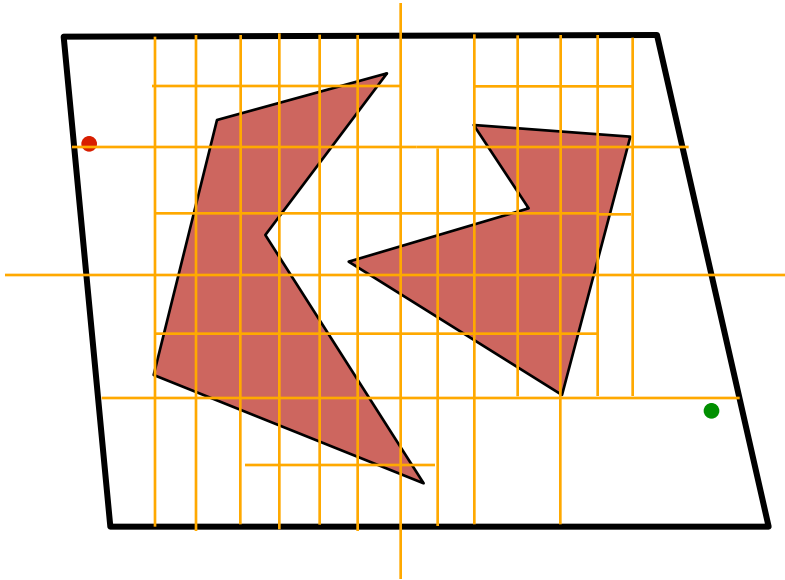


Quadtree:

recursively subdivides each *mixed* obstacle/free (sub)region into four quarters...

further decomposing...

- Approximate cell decomposition

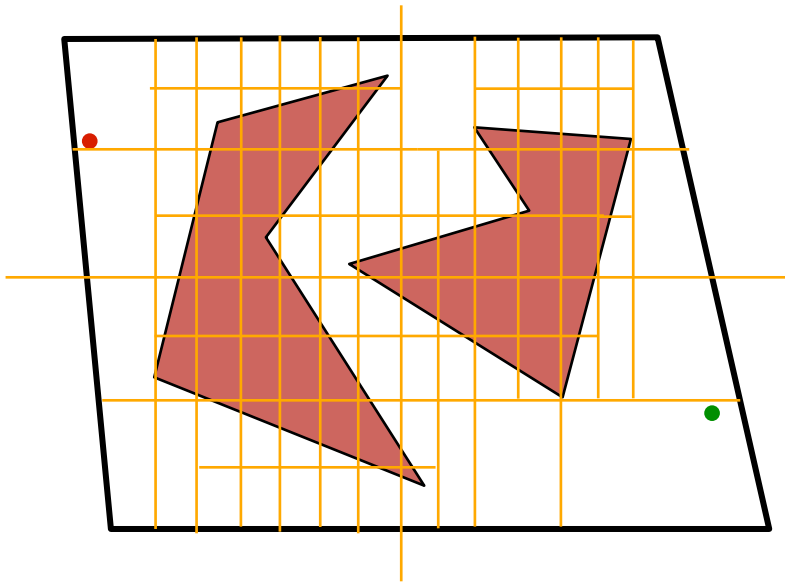


Quadtree:

recursively subdivides each *mixed* obstacle/free (sub)region into four quarters...

further decomposing...

- Approximate cell decomposition

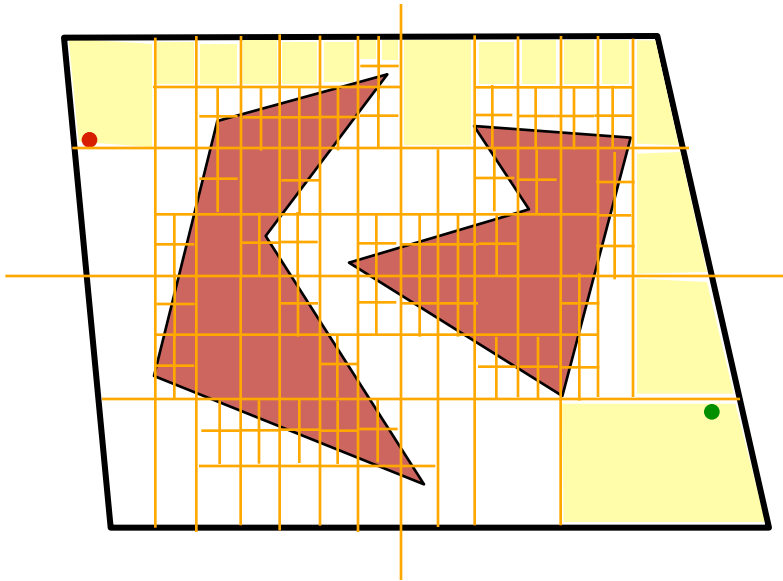


Quadtree:

recursively subdivides each *mixed* obstacle/free (sub)region into four quarters...

further decomposing...

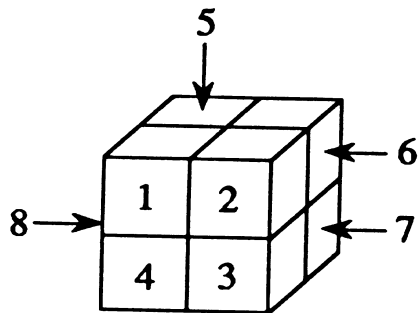
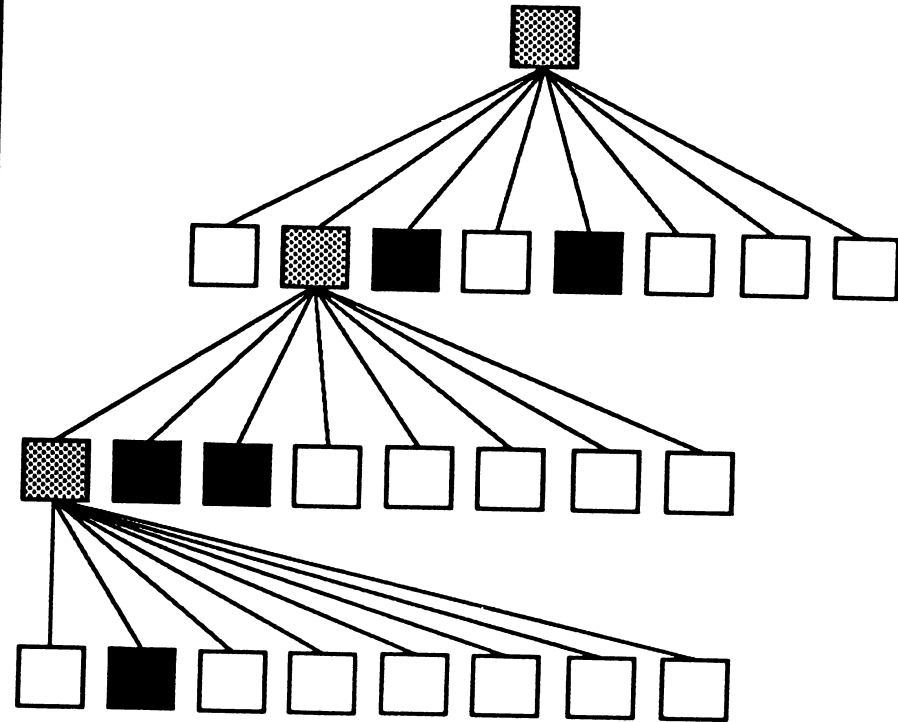
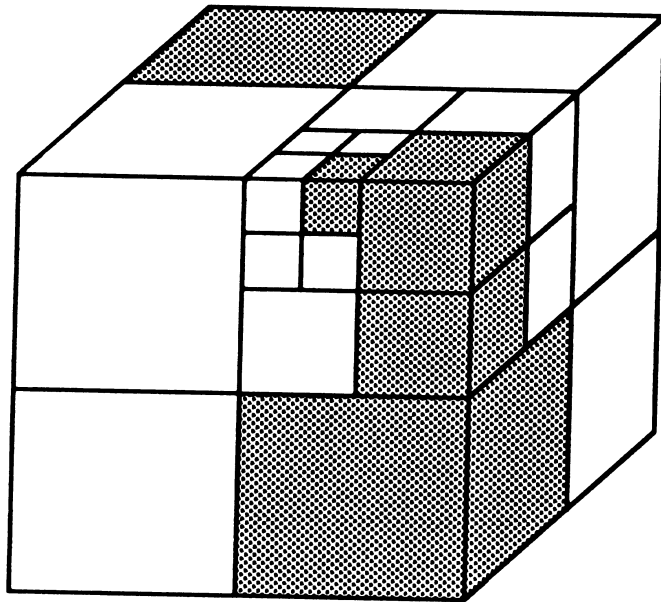
- Approximate cell decomposition



Quadtree

Again, use a graph-search algorithm to find a path from the start to goal

Octree Decomposition



 EMPTY cell  MIXED cell  FULL cell