

Mobile Robot Exploration

...

Bill Hunt, Annie Hernandez & Trevor Rocks

Goal & Challenges

Goal: Map an unknown closed environment autonomously, within twenty minutes.

Challenges:

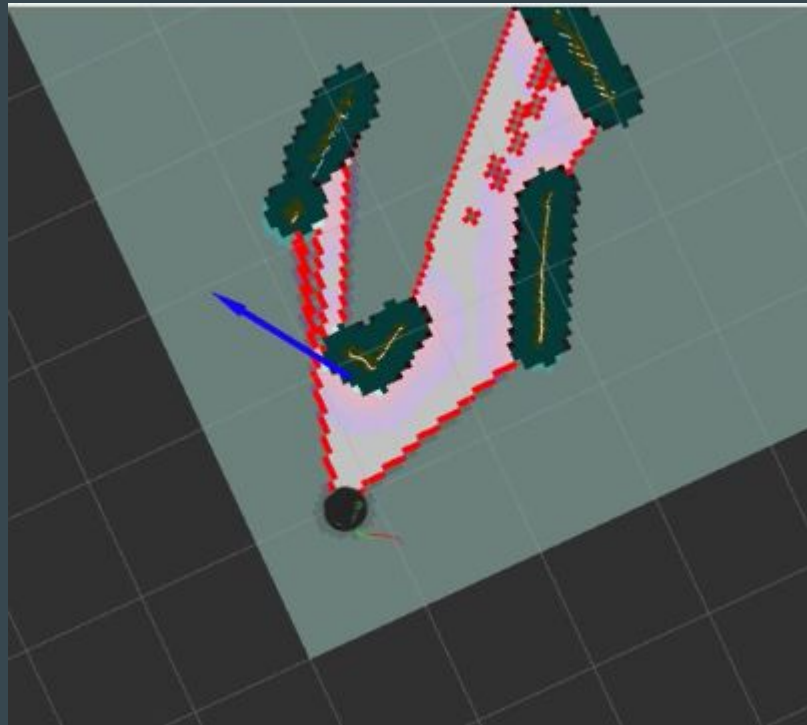
- SLAM and A* (handled by GMapping)
- Frontier detection
- Frontier grouping (blobbing)
- Goal selection
- Completion detection
- Fault detection and recovery



http://cdn.shopify.com/s/files/1/0084/1842/products/Back_grande.png?v=1351283525

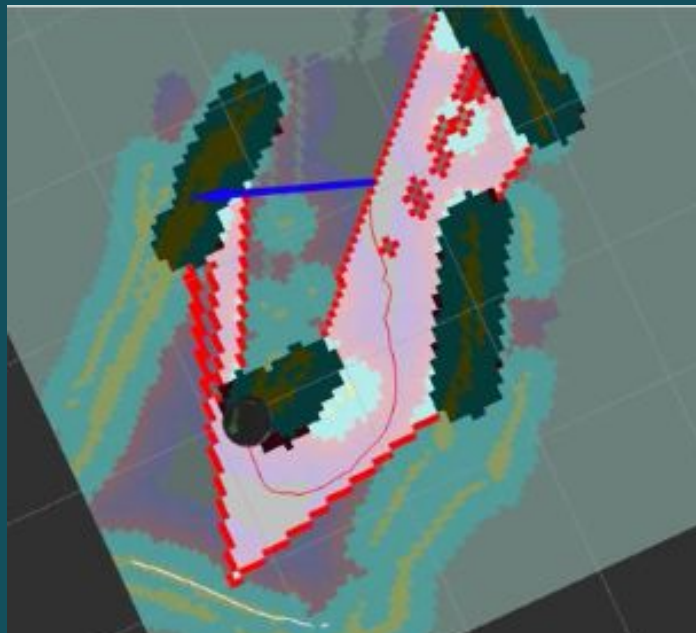
Frontier Detection and Blobbing

1. Expand obstacles in map to account for robot dimensions.
2. Wavefront out from robot to determine frontiers (boundaries of known map).
3. Group frontiers into “blobs” based on contiguity.



Goal Selection

- Here, the robot has found a usable goal: the centroid of a frontier.
- GMapping's path planning and navigation are used to navigate to the waypoint.
- Our node is subscribed to *move_base/result*, so we are notified when GMapping reaches a goal or experiences a failure.



Fault Detection and Recovery

- Sometimes GMapping cannot reach a normally-generated goal.
- When enough failures occur, our node reanalyzes the occupancy grid using a less zealous obstacle expansion algorithm.
- Instead of choosing the centroid of the largest frontier, we find a reachable point *in the global costmap* that is nearby the ideal centroid target.
- We target a random orientation at each goal location, to promote stochastic map growth.

```
if tryClosest:
    reachableCentroid = getReachableTarget(costGrid, centroid, localPose)
    target = mapToPoseCoordinates(reachableCentroid, grid.info.resolution, grid.info.origin)
else:
    target = mapToPoseCoordinates(centroid, grid.info.resolution, grid.info.origin)

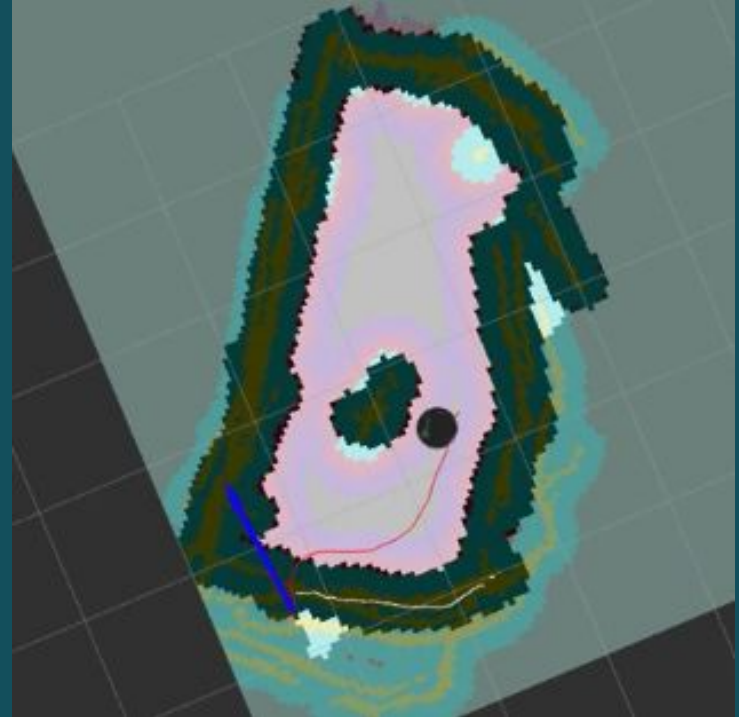
target.pose.orientation.x = 0
target.pose.orientation.y = 0
target.pose.orientation.z = 1
target.pose.orientation.w = random.uniform(0.0, 0.99)
```

Final Map

- Completed outer perimeter (very small residual frontiers are allowed).
- Entire interior area explored
- Obstacle within interior is well defined and topology of space is accurate.
- We don't terminate in the fault condition; the robot must recover and *then* check for termination criteria.

When exploration is complete, the user sees a bash prompt:

```
Waiting for map update...
Handling cost map
Handling map
Computing frontiers.
Map Complete
```



Video of operation

We are happy to answer any questions that you may have.

Thank you for a fantastic and interesting course!

Link to video: <https://www.youtube.com/watch?v=V5FJzfqFjpA>

