

Trajectory Rollout Algorithm

Course 4, Module 6, Lesson 3



UNIVERSITY OF TORONTO
FACULTY OF APPLIED SCIENCE & ENGINEERING

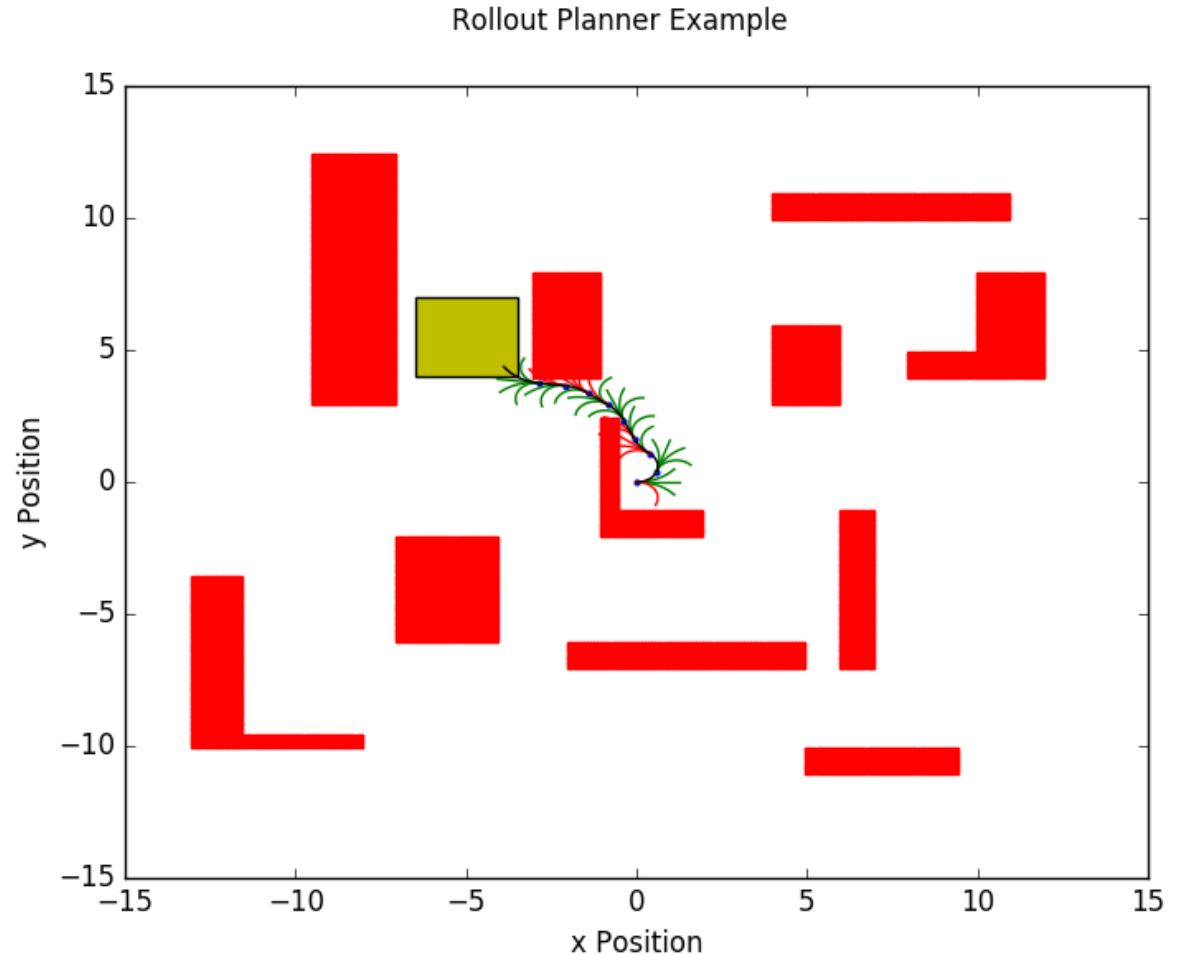
Learning Objectives

- Implement the trajectory rollout algorithm
 - Trajectory propagation
 - Collision checking
 - Path selection
- Understand the concept of receding horizon planning

Trajectory Rollout Planner

- Uses trajectory propagation to generate candidate set of trajectories
- Among collision-free trajectories, select trajectory that makes the most progress to goal

Take the obstacle information and determine collision-free paths



Trajectory Set Generation

- Each trajectory corresponds to a fixed control input to our model
 - Typically uniformly sampled across range of possible inputs
- More sampled trajectories leads to more maneuverability
- Fewer sampled trajectories improves computation time

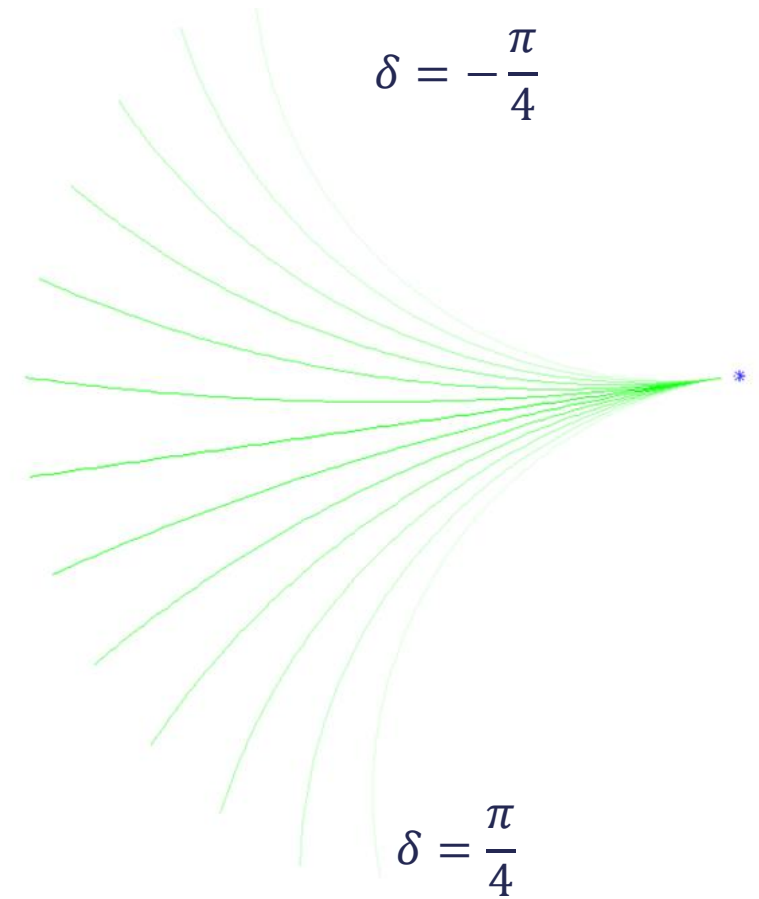
Trajectory Propagation

- Holding the velocity constant and varying the steering angle gives candidate set of trajectories

$$x_n = \sum_{i=0}^{n-1} v_i \cos(\theta_i) \Delta t = x_{n-1} + v_{n-1} \cos(\theta_{n-1}) \Delta t$$

$$y_n = \sum_{i=0}^{n-1} v_i \sin(\theta_i) \Delta t = y_{n-1} + v_{n-1} \sin(\theta_{n-1}) \Delta t$$

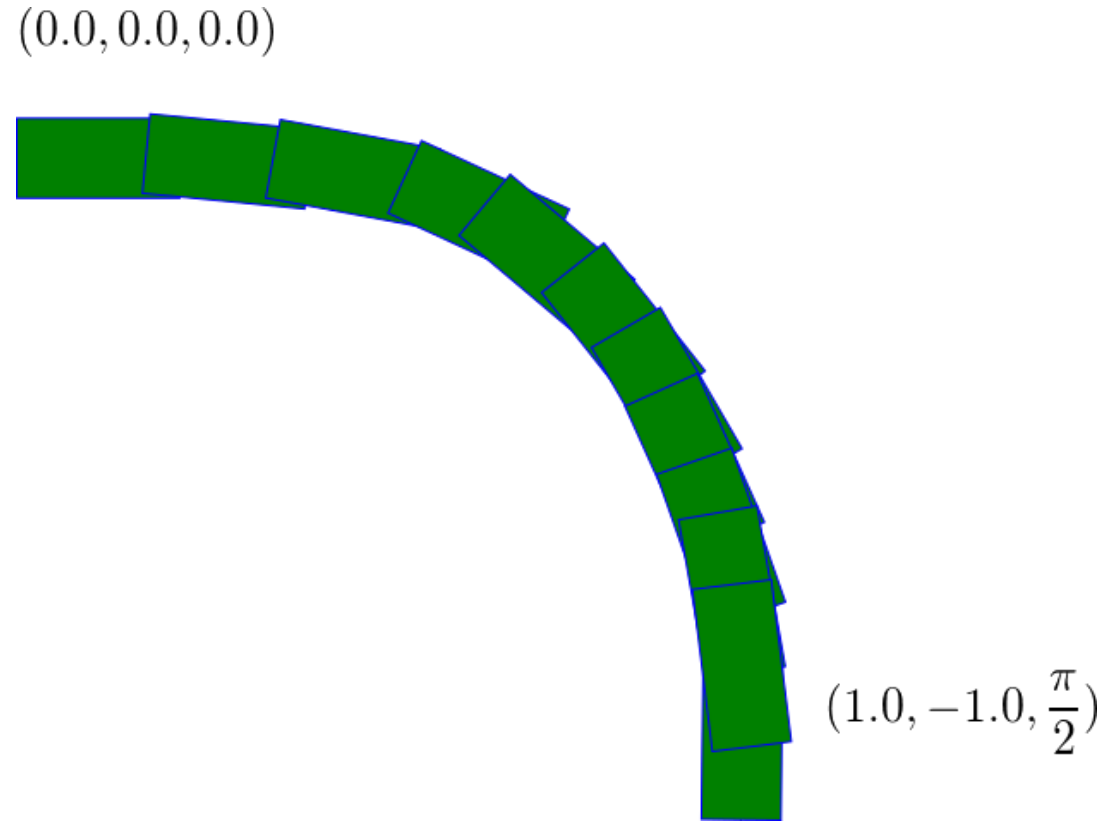
$$\theta_n = \sum_{i=0}^{n-1} \frac{v_i \tan(\delta_i)}{L} \Delta t = \theta_{n-1} + \frac{v_i \tan(\delta_i)}{L} \Delta t$$



Swath Based Collision Checking

- Swath is generated for each candidate trajectory
- Collision checking is performed for each point in the swath using the occupancy grid

$$S = \cup_{p \in P} F(x(p), y(p), \theta(p))$$



Objective Function

- Rewarding progress to a goal point is the ultimate goal of motion planning

$$J = \|x_n - x_{goal}\|$$

Distance to goal

$$J = \alpha_1 \|x_n - x_{goal}\| + \alpha_2 \sum_{i=1}^n \kappa_i^2 + \alpha_3 \sum_{i=1}^n \|x_i - P_{center}(x_i)\|$$

Curvature

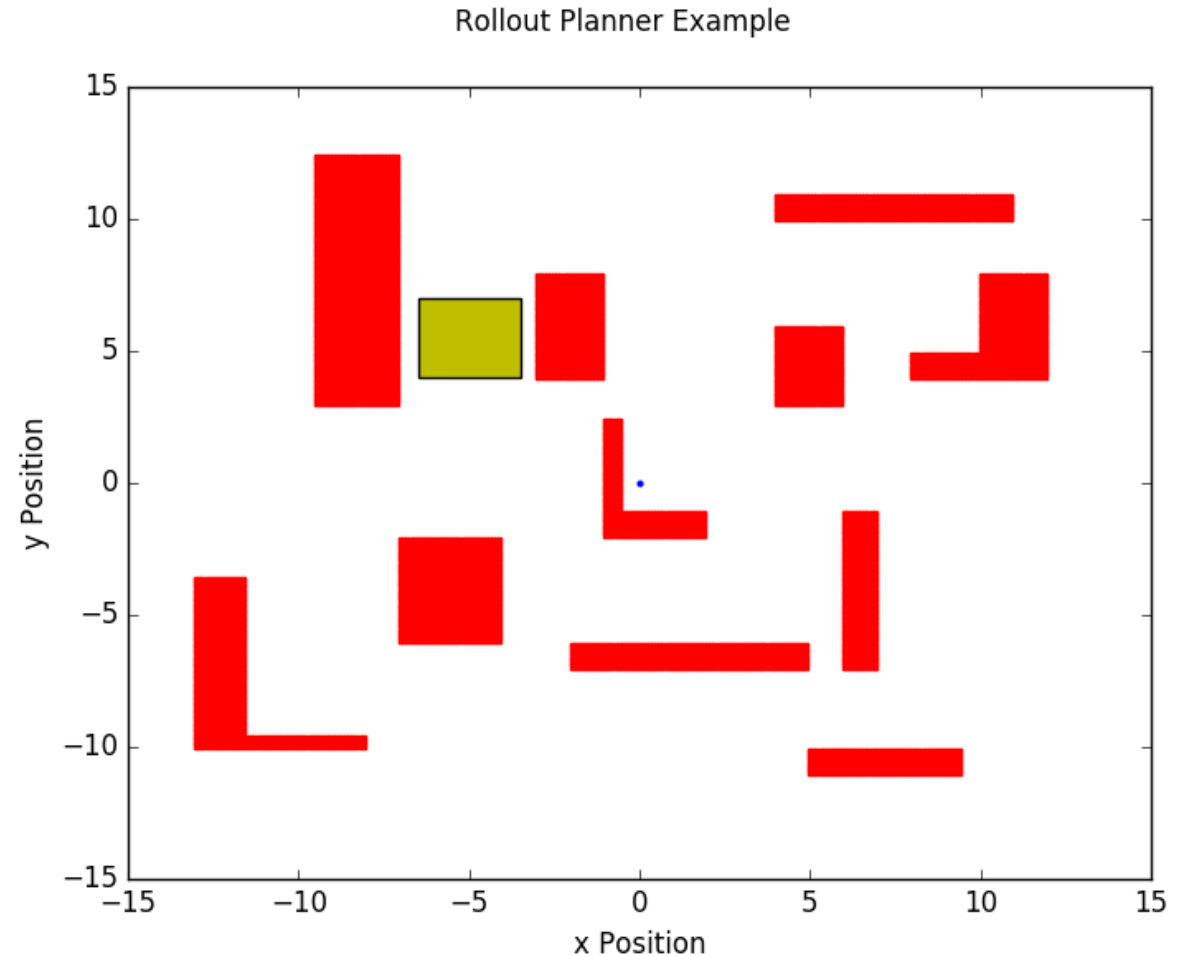
Deviation from centerline

+ ... distance to nearest obstacle

Example

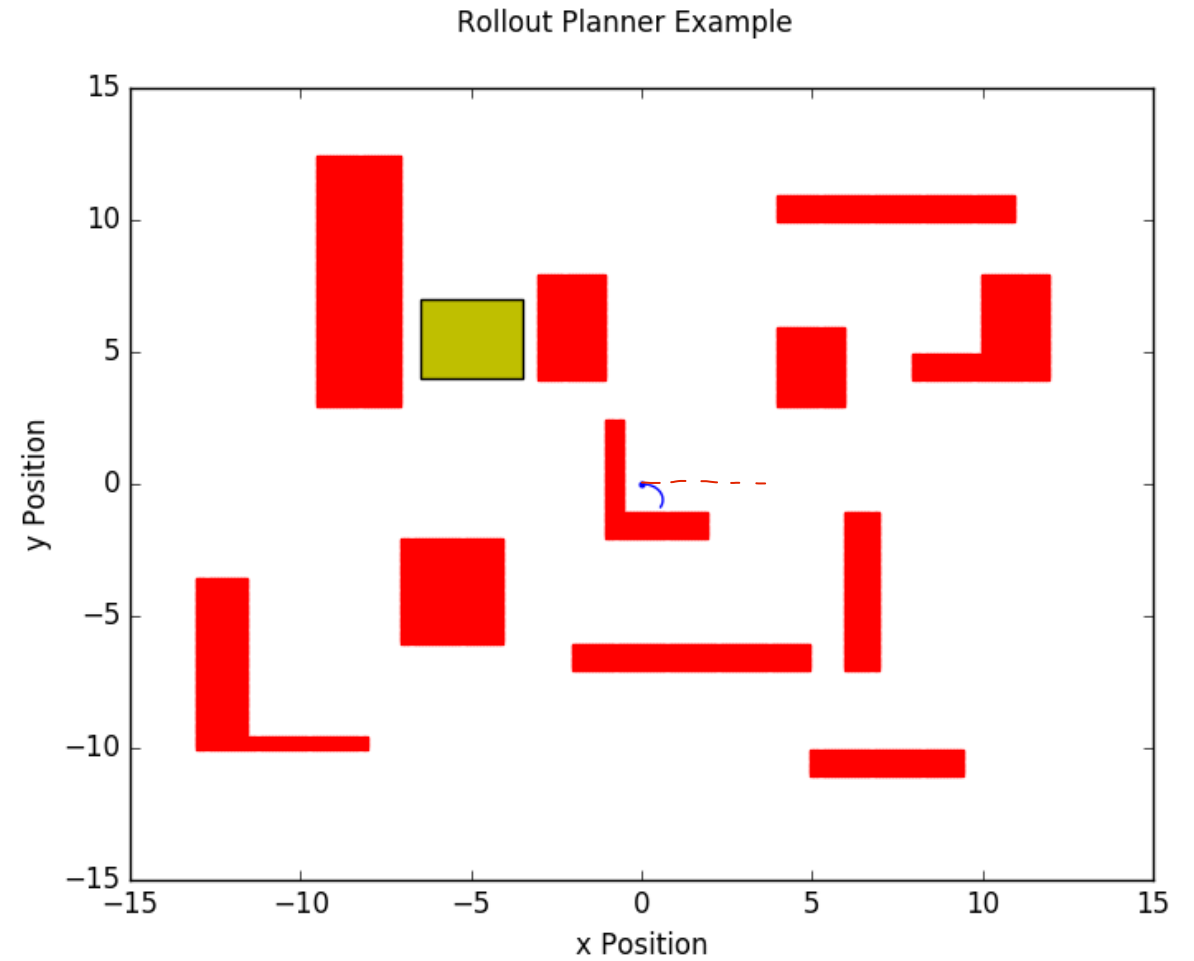
- Steering angle bounded by $|\delta| \leq \frac{\pi}{4}$ *$\frac{\pi}{8}$ step size*
- Time discretization of 0.1s, with a 2s planning horizon
- Want to reach the gold region while avoiding red obstacles

Velocity: 0.5 m/s



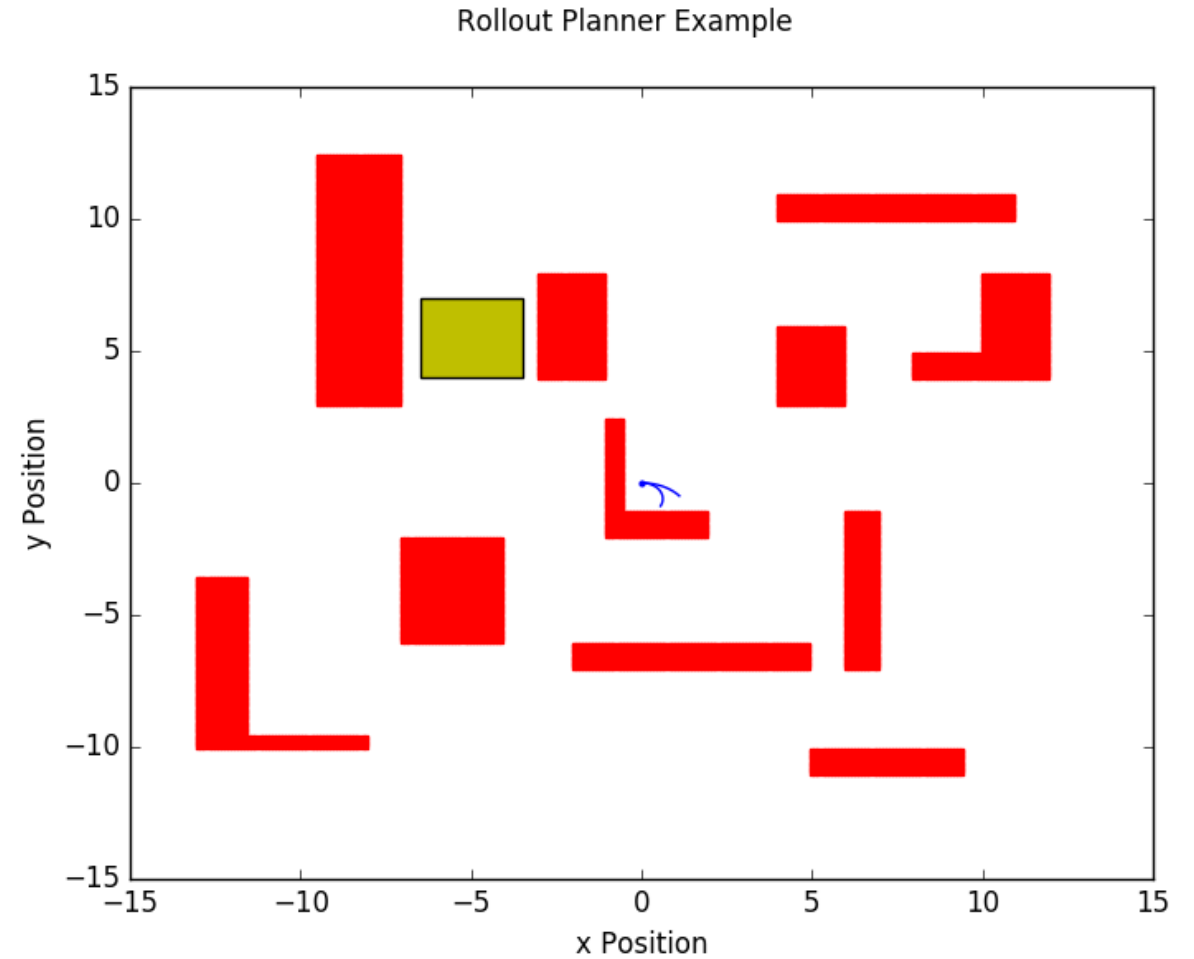
Example

- First trajectory corresponds to $\delta = -\frac{\pi}{4}$



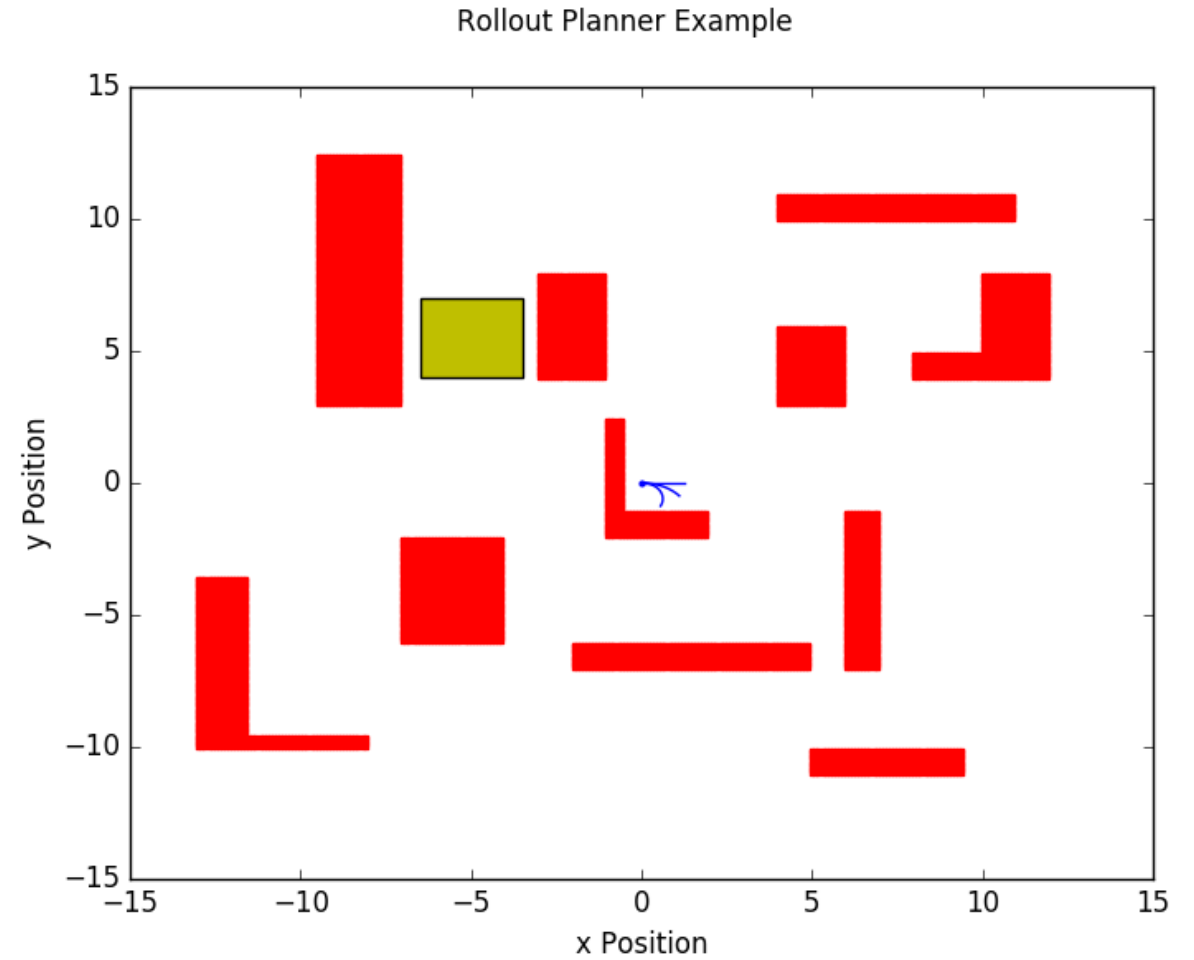
Example

- Second trajectory corresponds to $\delta = -\frac{\pi}{8}$



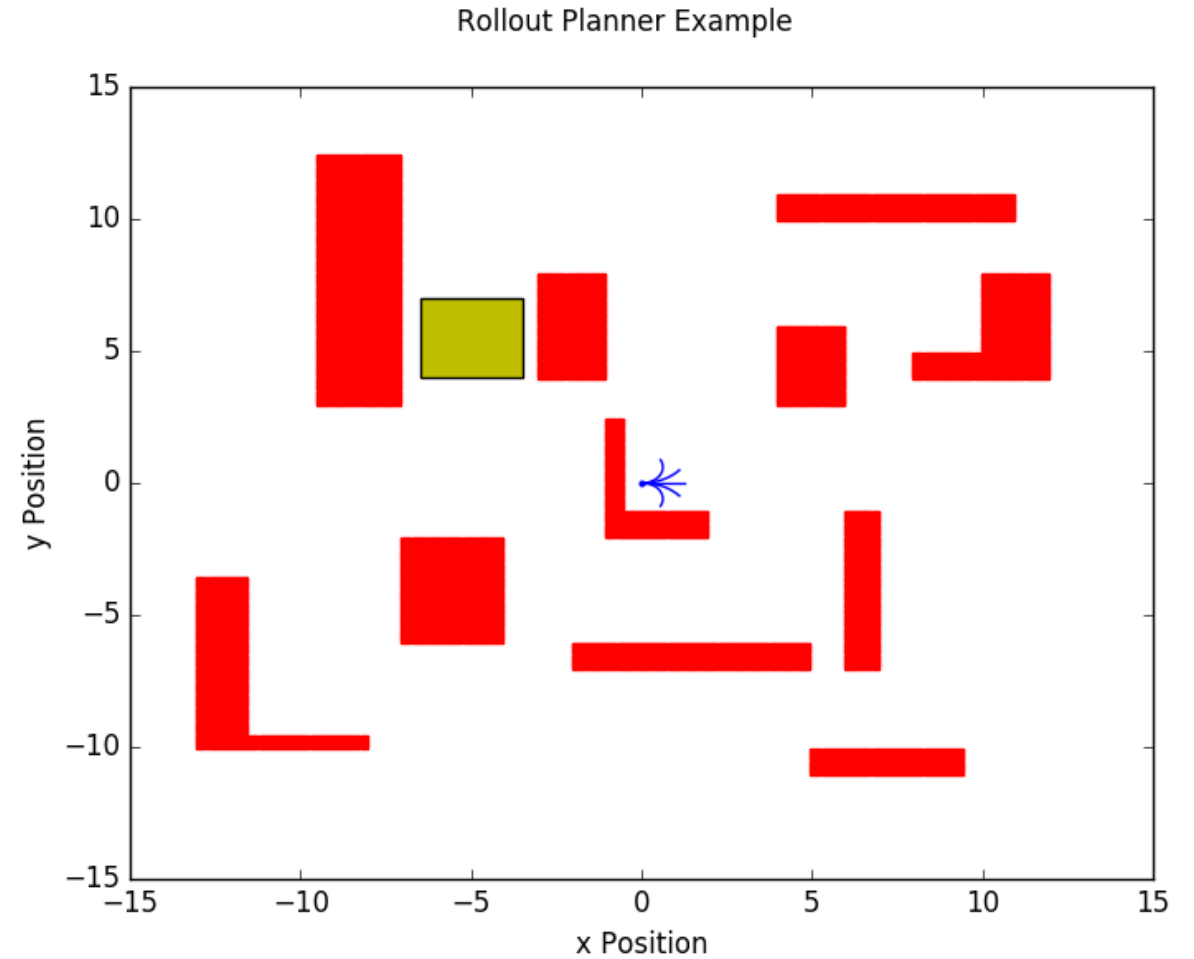
Example

- Straight line trajectory corresponds to $\delta = 0$



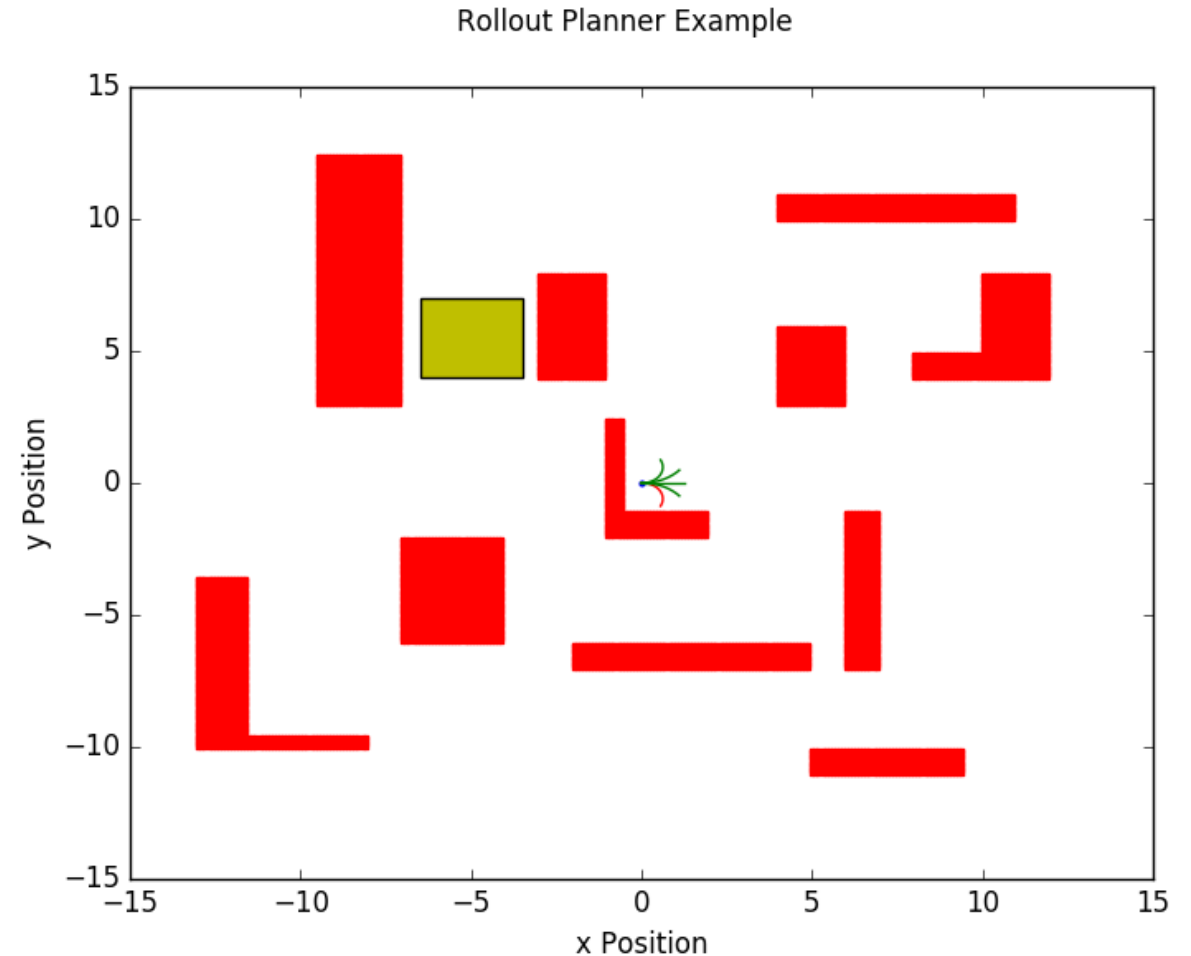
Example

- Positive steering angle trajectories are symmetrical with the negative steering angle trajectories



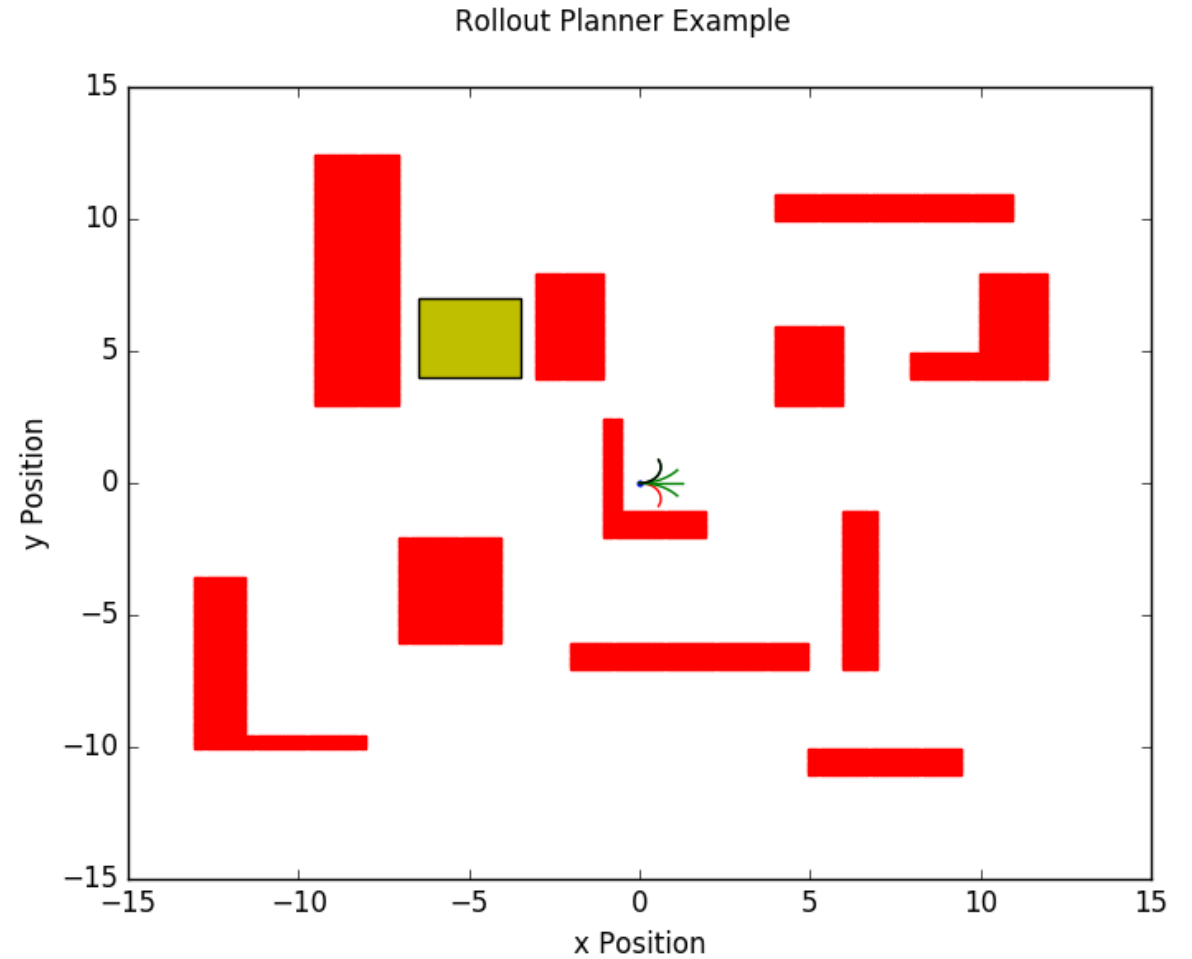
Example

- Using the vehicle footprint, we compute the swath for each trajectory
- Any trajectory whose swath intersects an obstacle is coloured red to mark a collision
- Green denotes a collision-free trajectory



Example

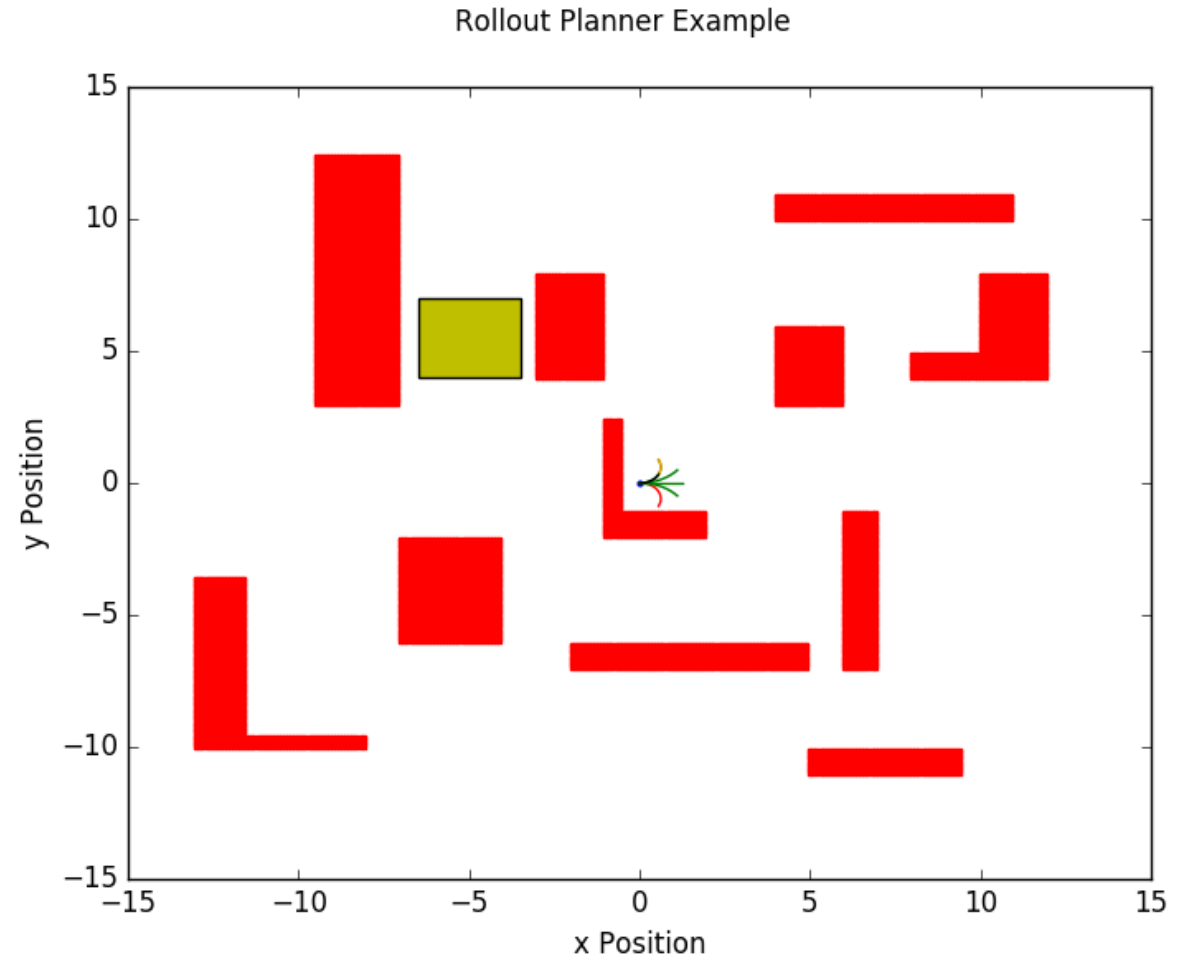
- Objective is evaluated across all collision-free trajectories
- The black path makes the most progress towards the goal region, so it is selected



Example

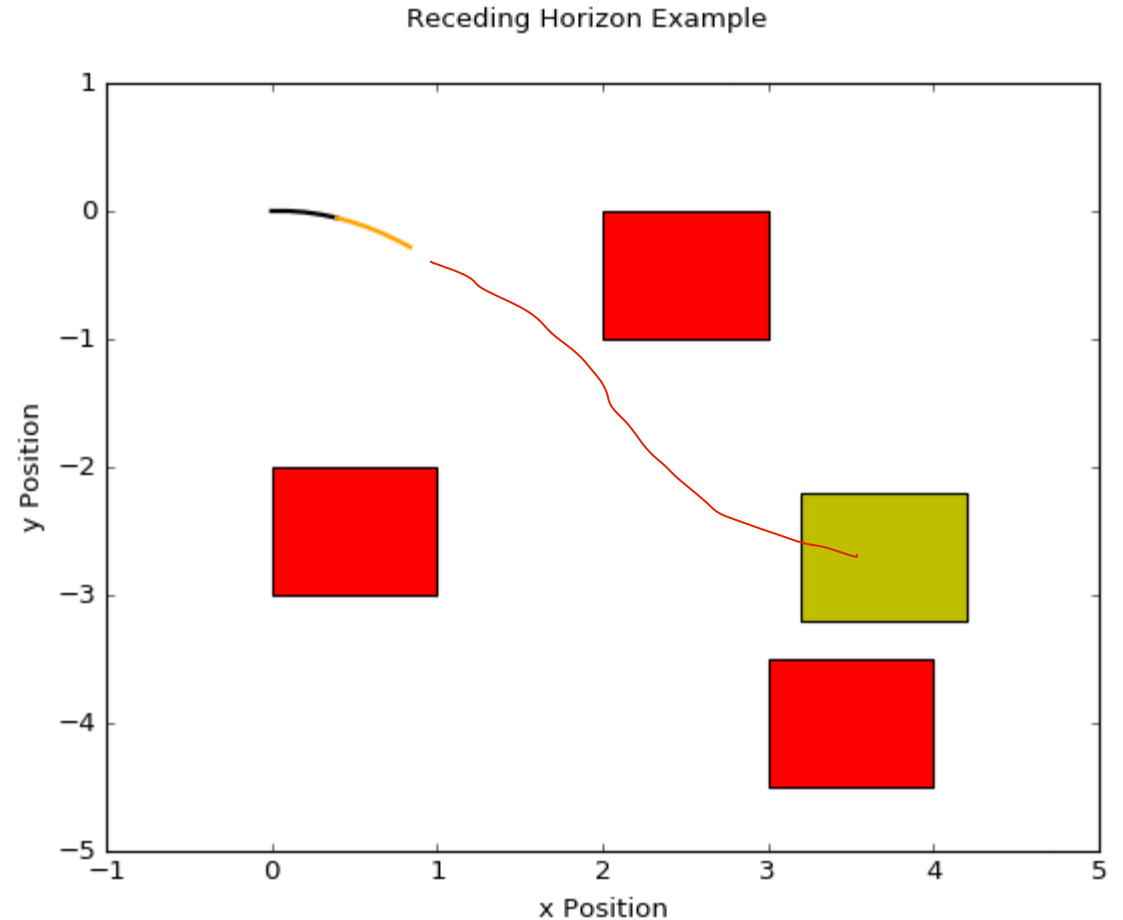
*Receding horizon
control*

- Planning cycle is shorter than trajectory length
- The orange portion of the trajectory is not executed before the next planning cycle



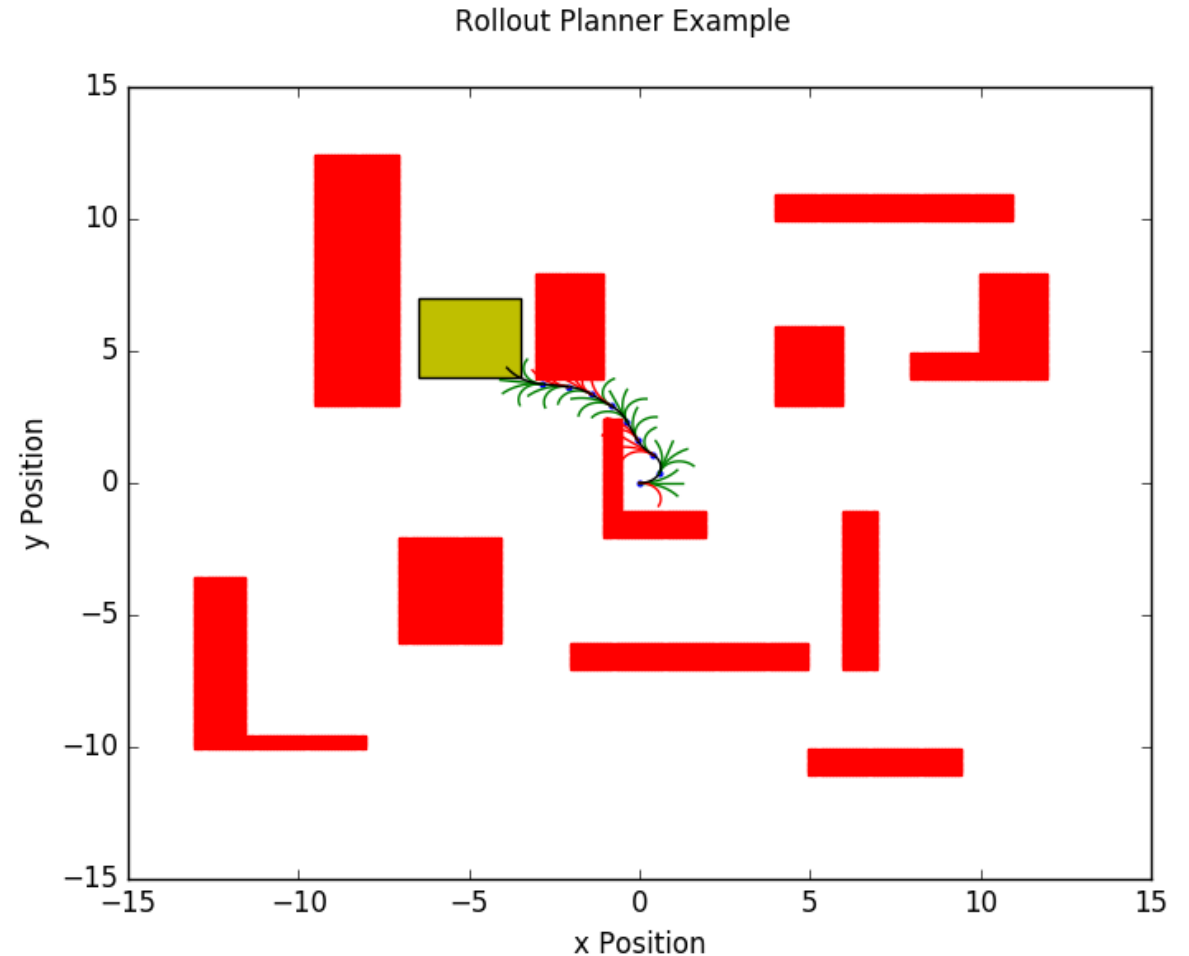
Receding Horizon Example

- Only 1s of 2s trajectory is executed at each planning iteration
- Planning horizon end time recedes towards time point when goal is reached



Example

- This process is continued until goal is reached
- This planner is greedy and sub-optimal, but is fast enough to allow for online planning



Summary

- Introduced the steps of the trajectory rollout motion planning algorithm
- Illustrated an example situation and planning solution
- Discussed receding horizon planners

