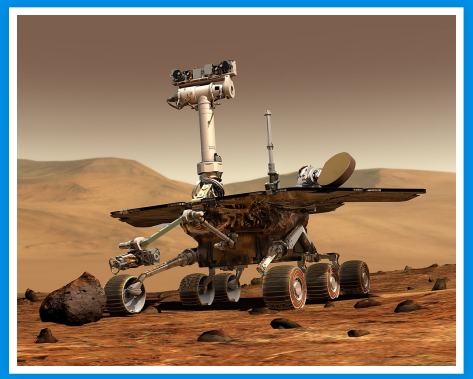
FINDING OPTIMAL ROBOT MOTION

BRYCE WILLEY

NOVEMBER 16TH, 2017

RICE



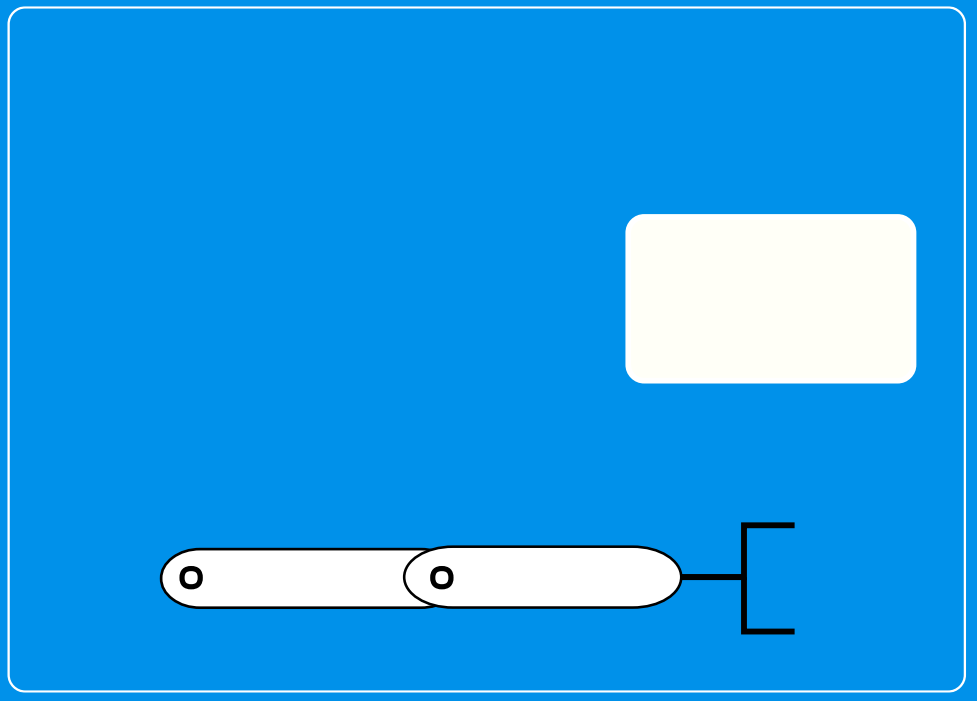
Robots can...

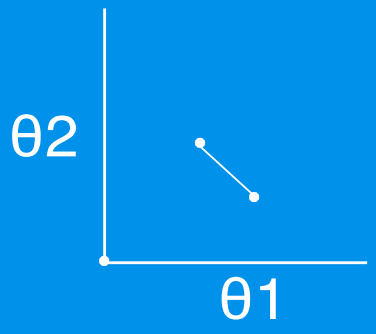
- assist people with disabilities/immobility
- complete tasks that are unsafe or impossible for humans

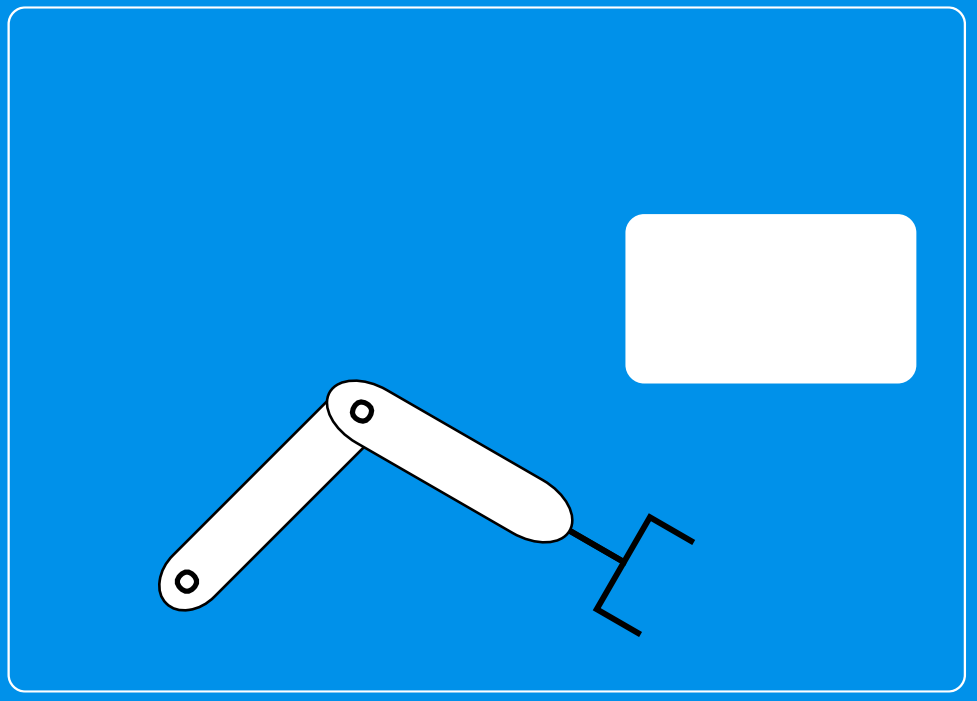
Automating these tasks means allows widespread deployment

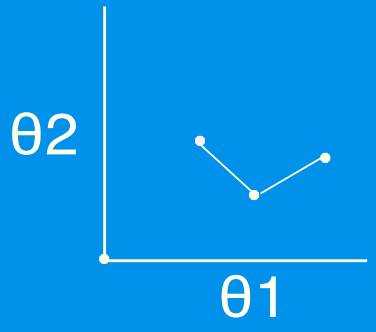
Motion planning is essential to automation.

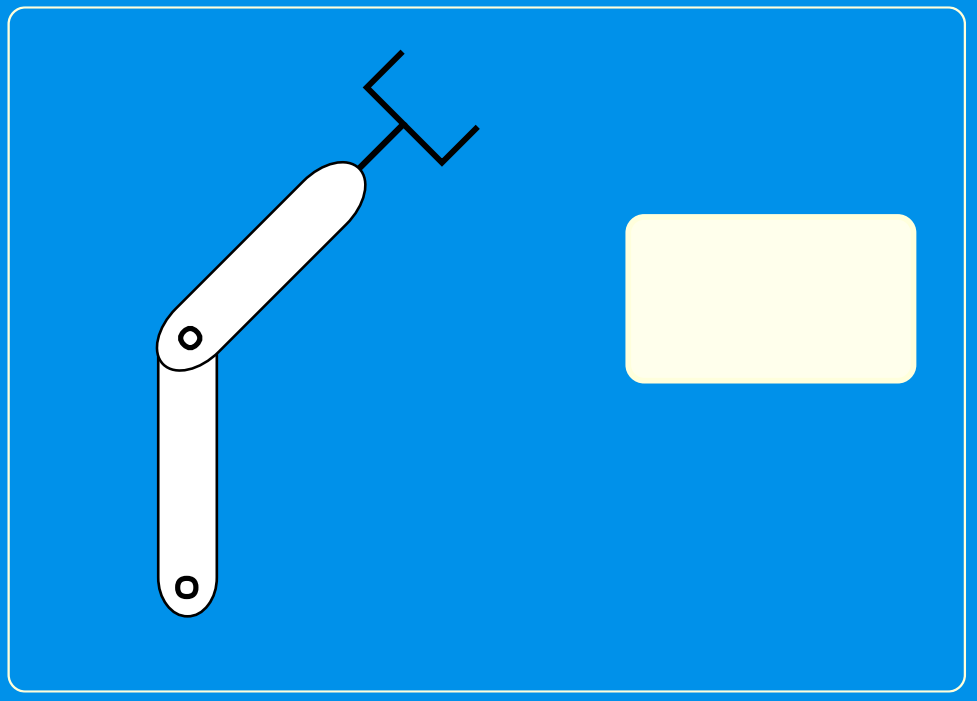

θ2
θ1




θ2
θ1

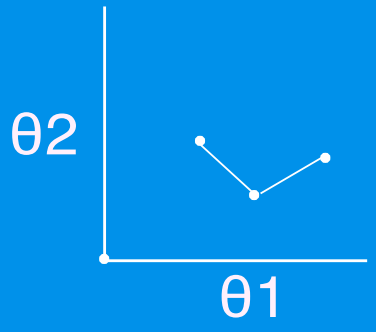


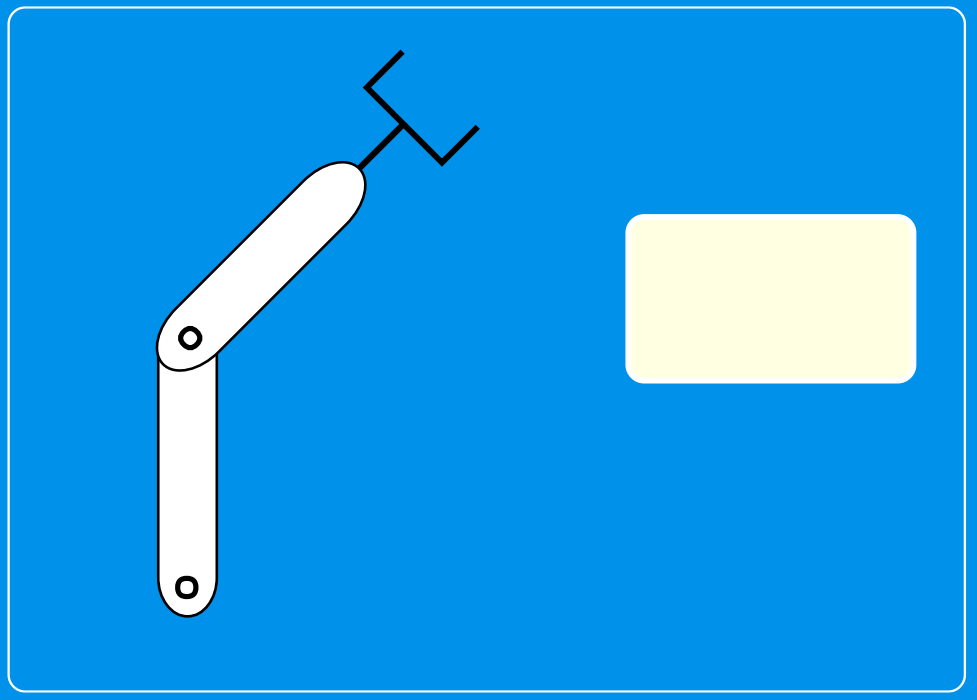

θ2
θ1



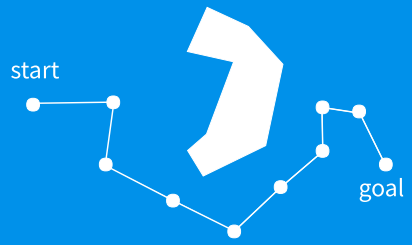
Configuration

Space



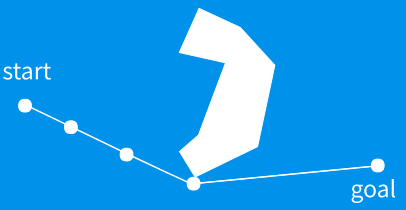


FEASIBLE MOTION PLANNING

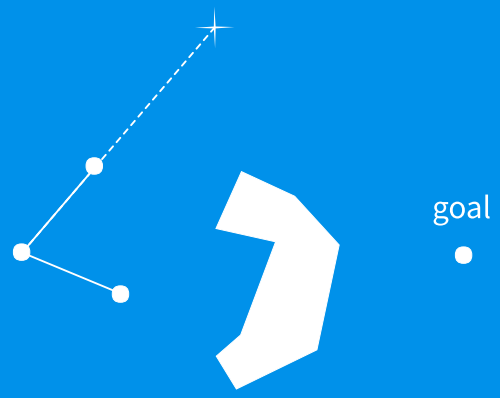


- Finds a connected path from start to goal
- Doesn't collide with any obstacles

OPTIMAL MOTION PLANNING



- Assigns a cost to every path
- Finds a feasible path that minimizes the cost

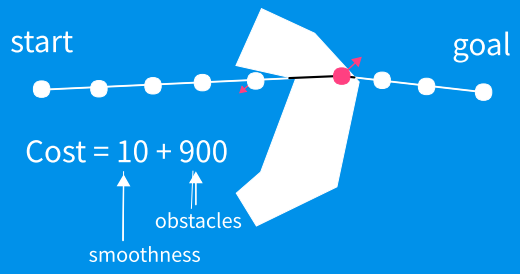


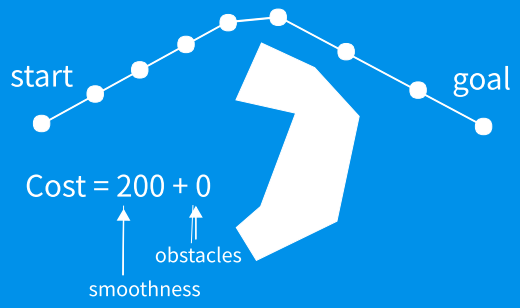
PROS

- If a solution exists, it will be found
- Very reliable
- Can find optimal paths

CONS

- Must smooth the path after finding it
- Finding optimal paths is time consuming





PROS

- Fast! (5-10x by our experiments)
- Quality, smooth paths

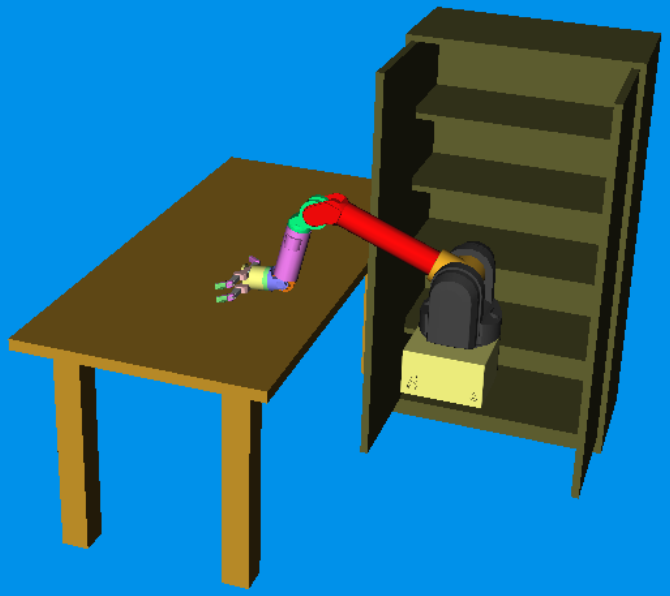
CONS

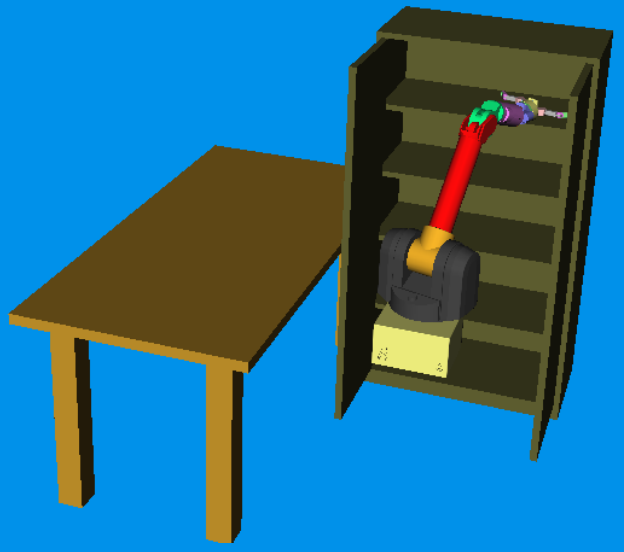
- No guarantee of finding a feasible path
- Not as reliable as sampling planners

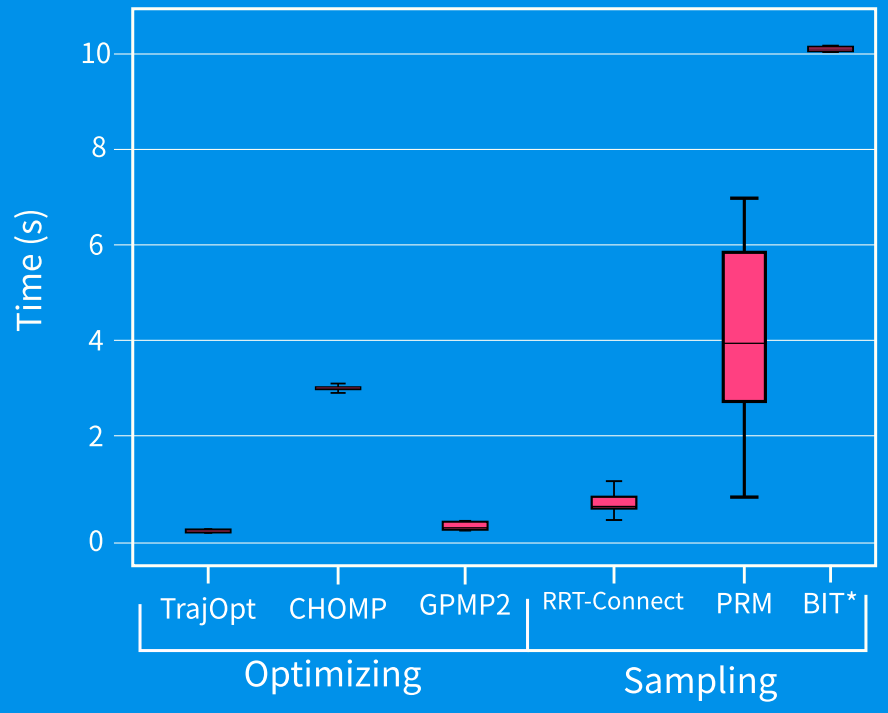
Currently, no rigorous comparisons in the literature

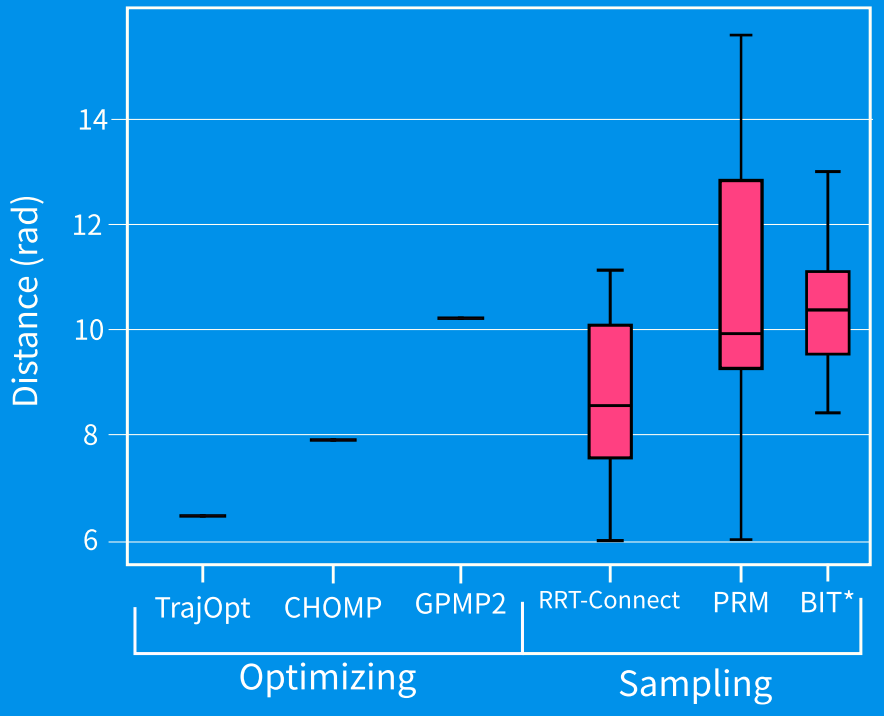
Optimization planners do special 'tricks' to speed up planning

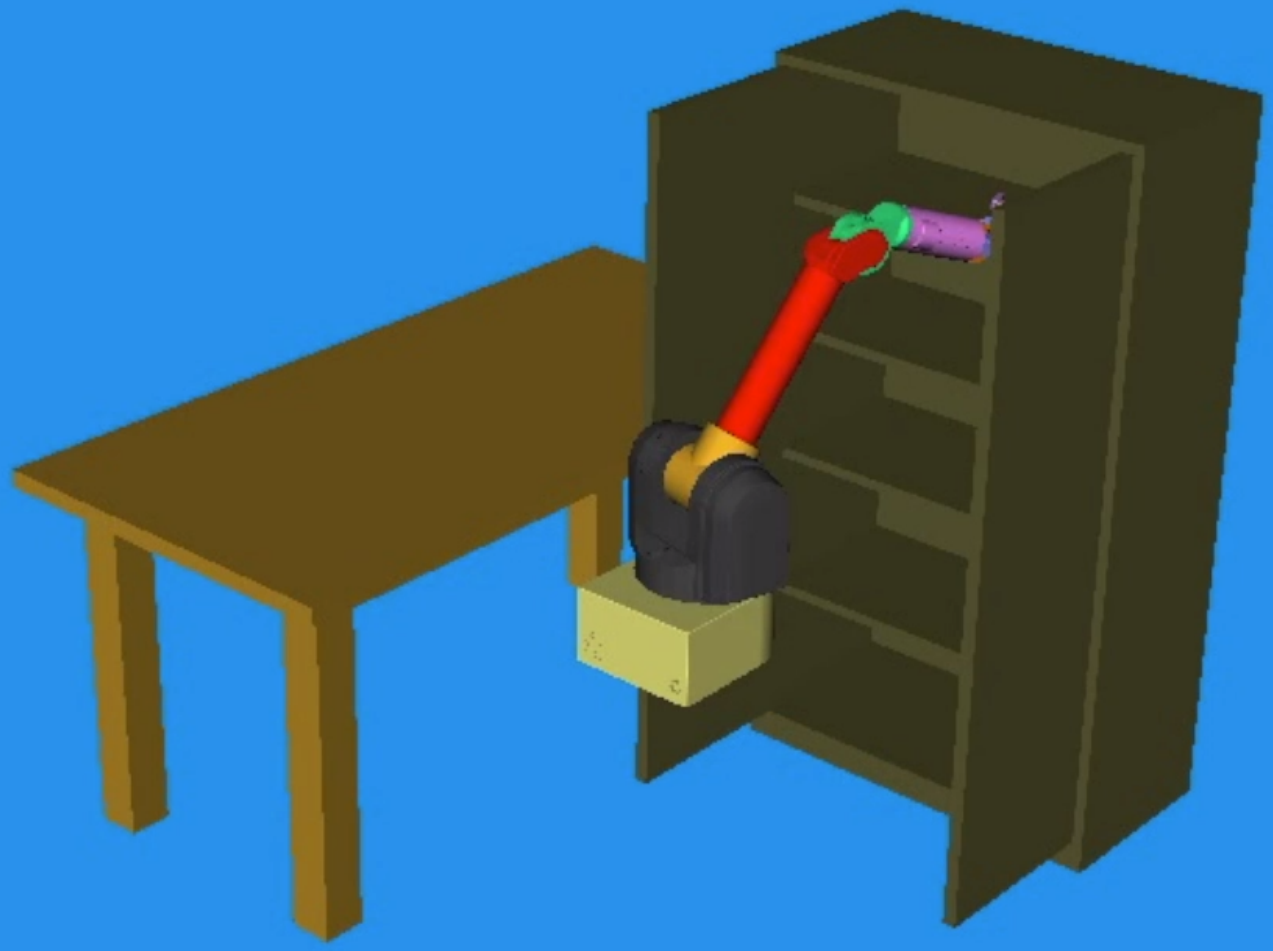
What really makes optimization planners faster?

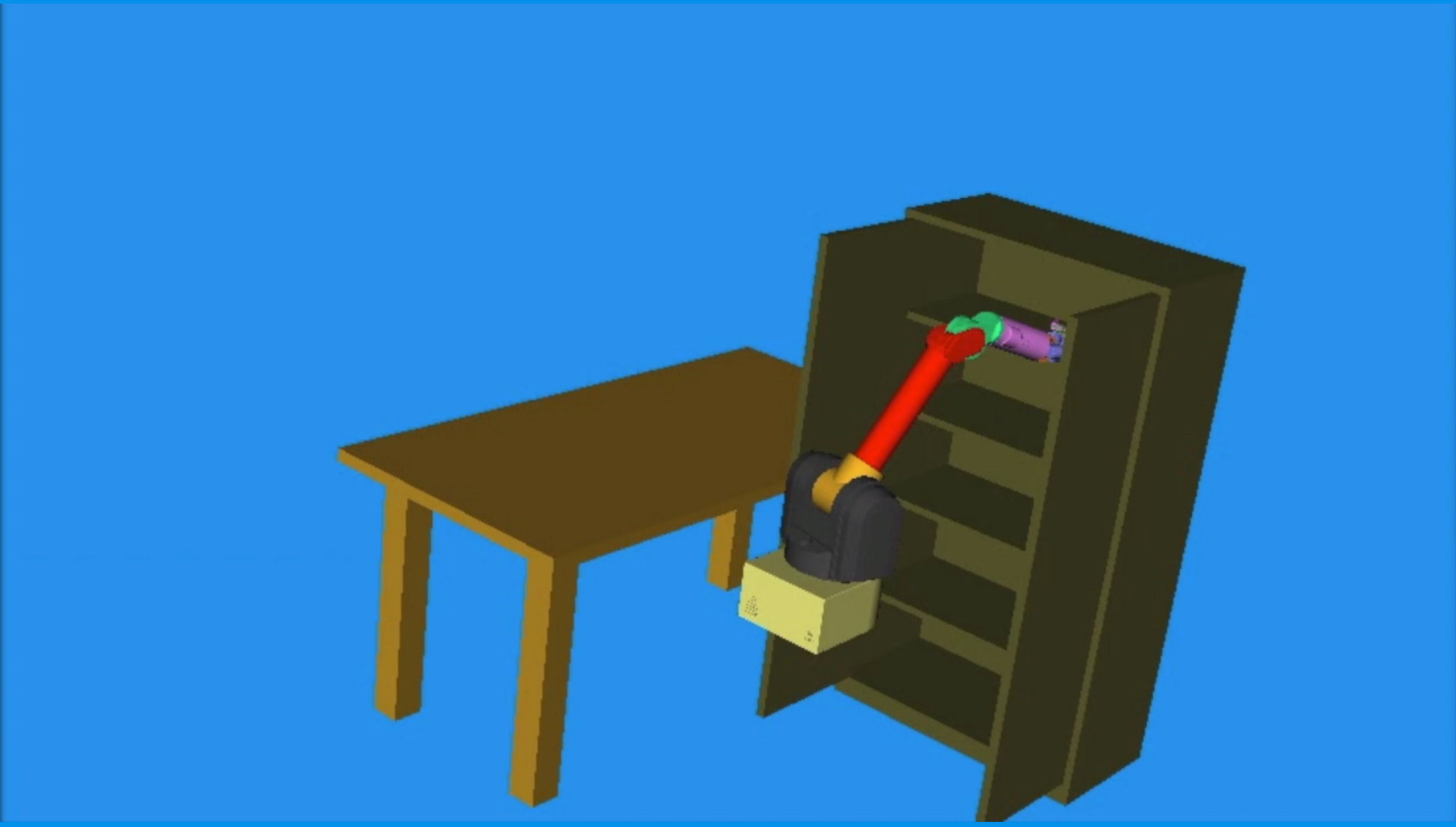


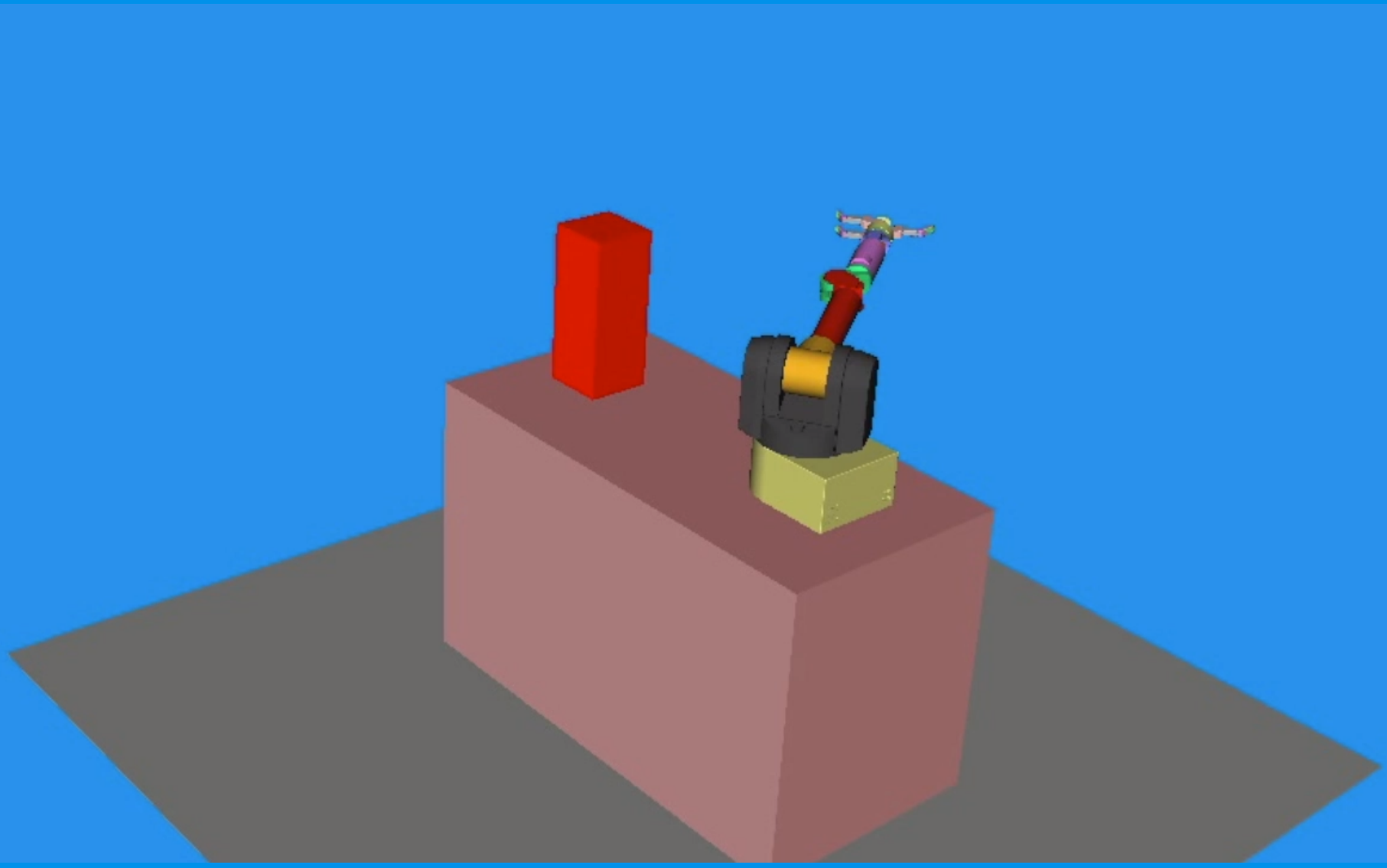



Time (s)
10
8
6
4
2
0
TrajOpt
CHOMP
GPMP2
RRT-Connect
PRM
BIT*
Optimizing
Sampling


Distance (rad)
14
12
10
8
6
TrajOpt
CHOMP
GPMP2
RRT-Connect
PRM
BIT*
Optimizing
Sampling



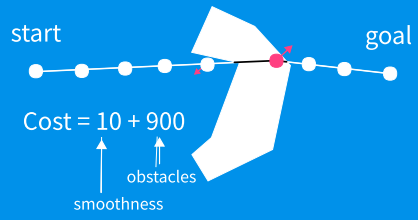




Abstract the 'tricks' from the planners

Test on a wider variety of planning problems

Use sampling and optimization together: quality motion and faster convergence



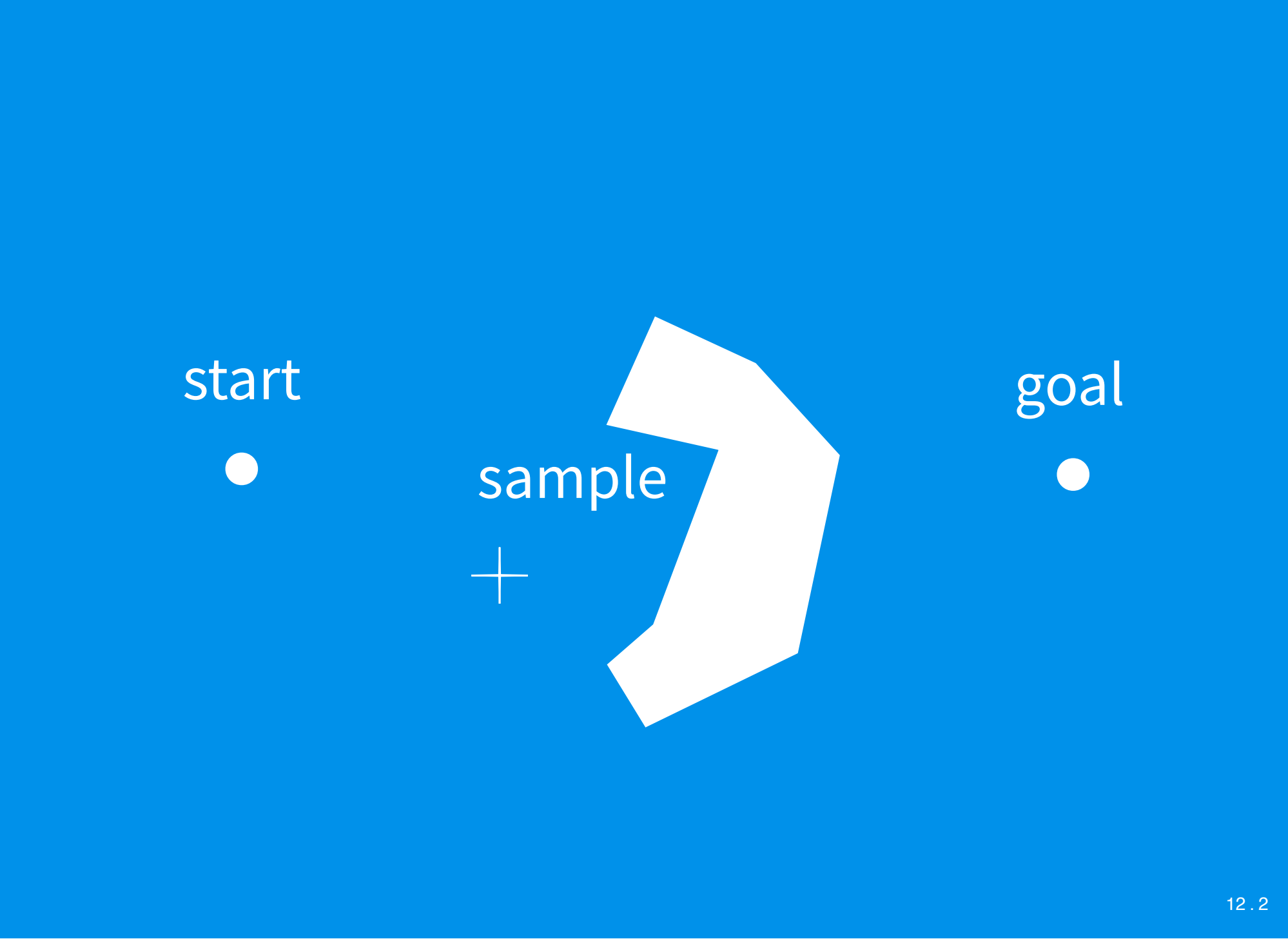
Sampling planners and optimization planners both have pros and cons

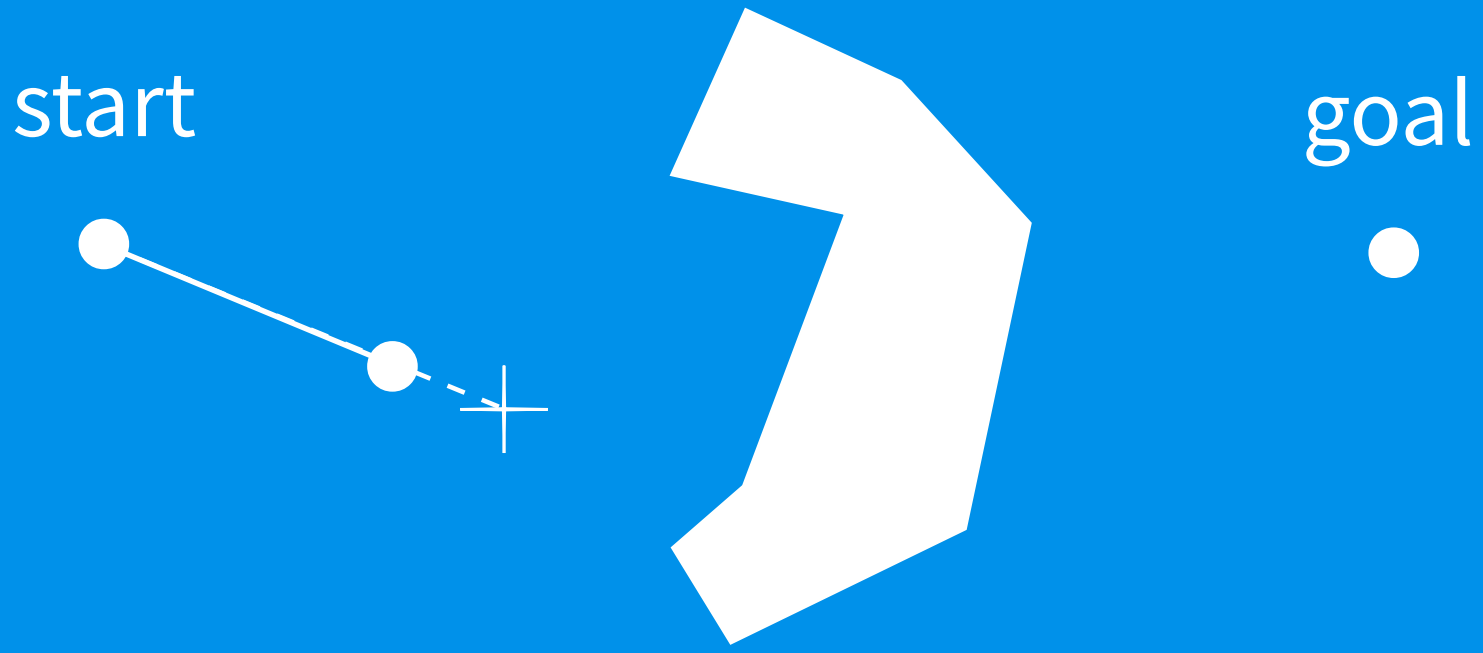
Optimization Planners are still not reliable enough to be used safely

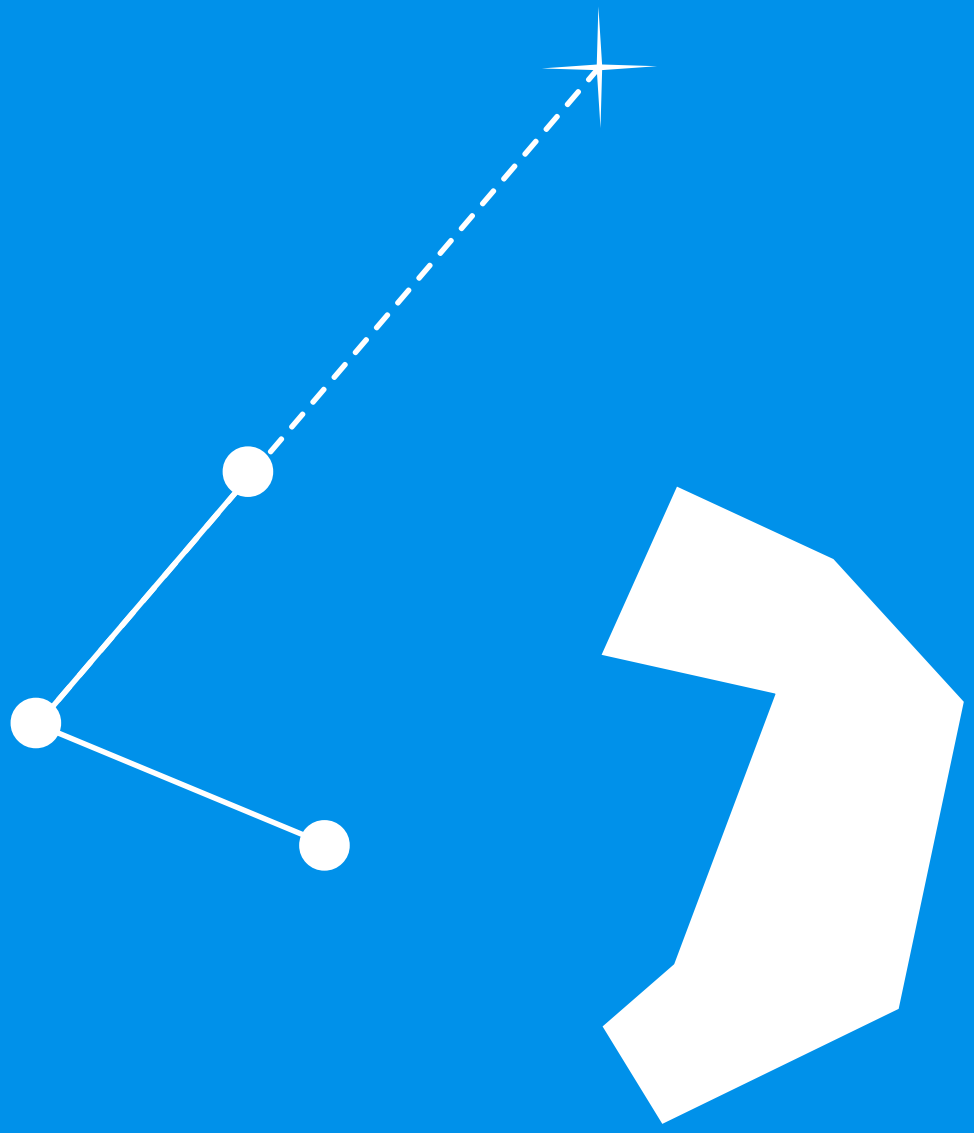
BONUS ANIMATIONS

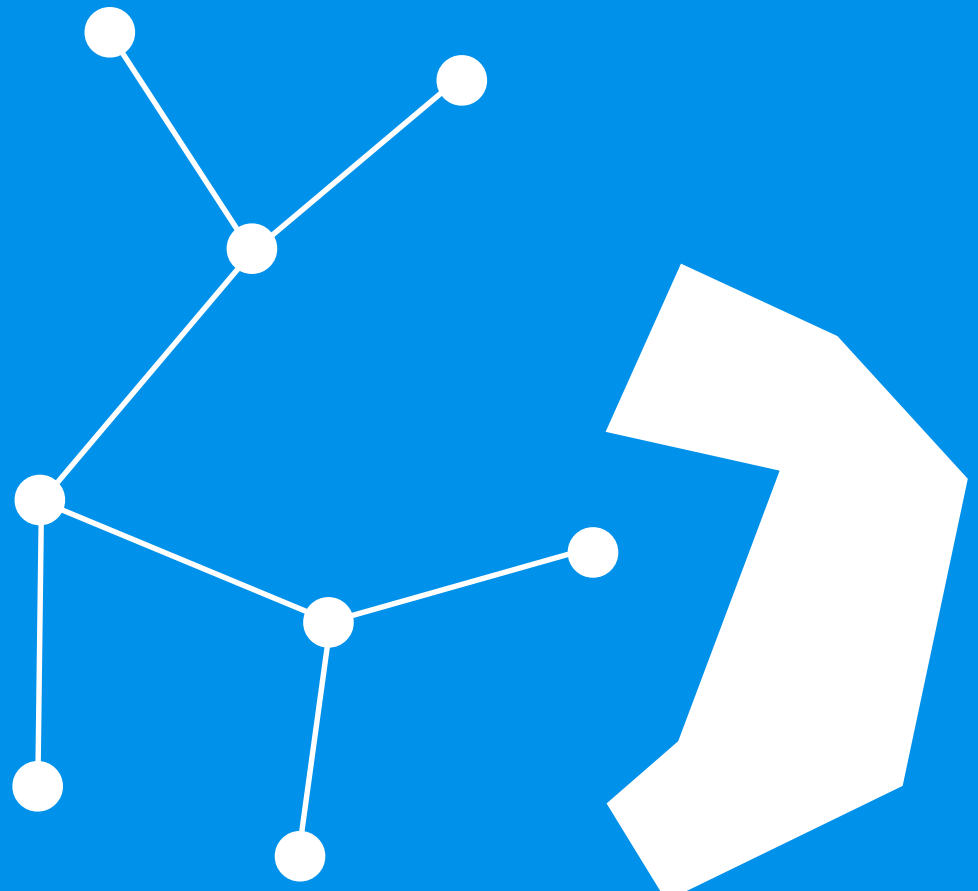
start

goal

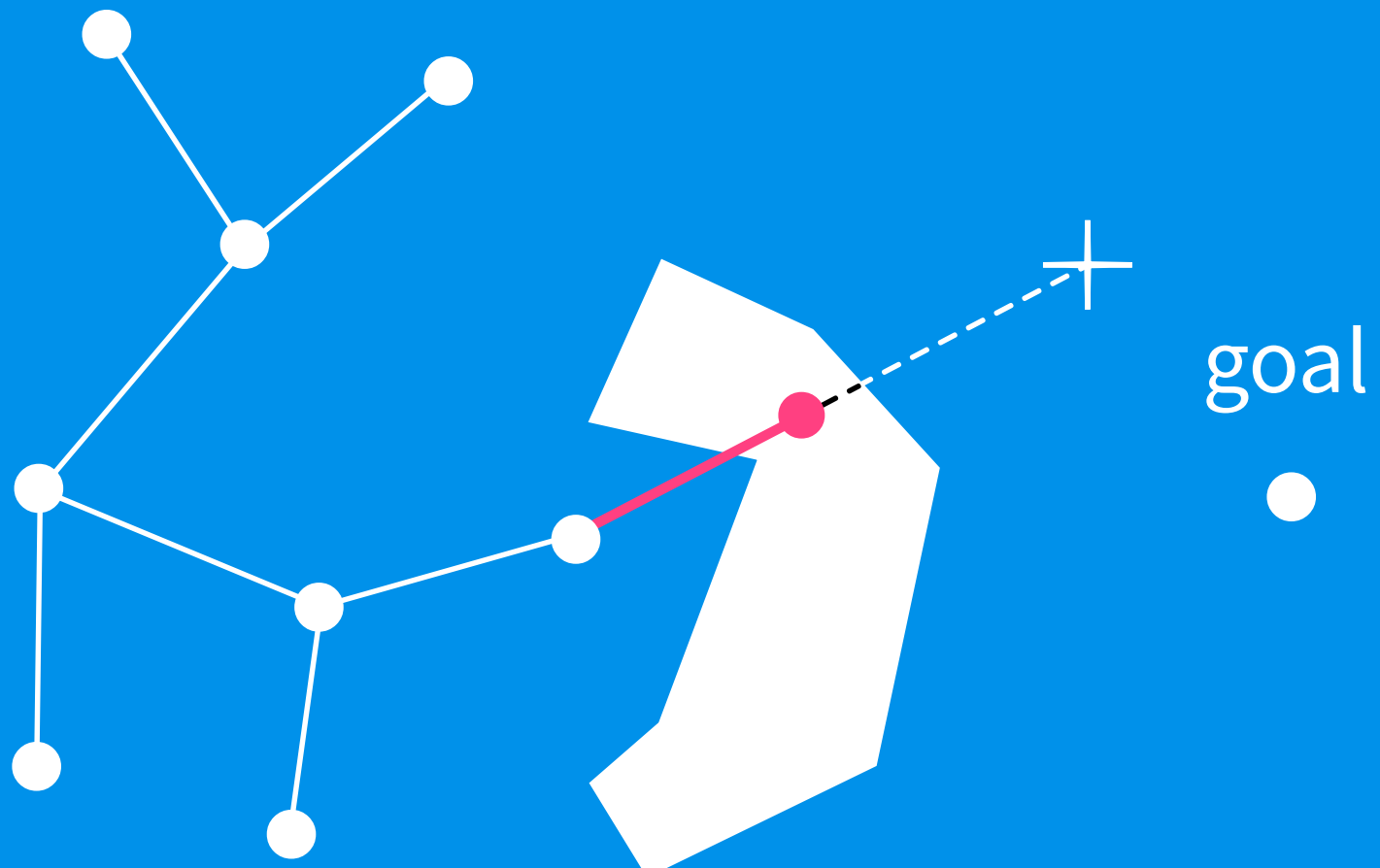

start
sample
goal


start
goal

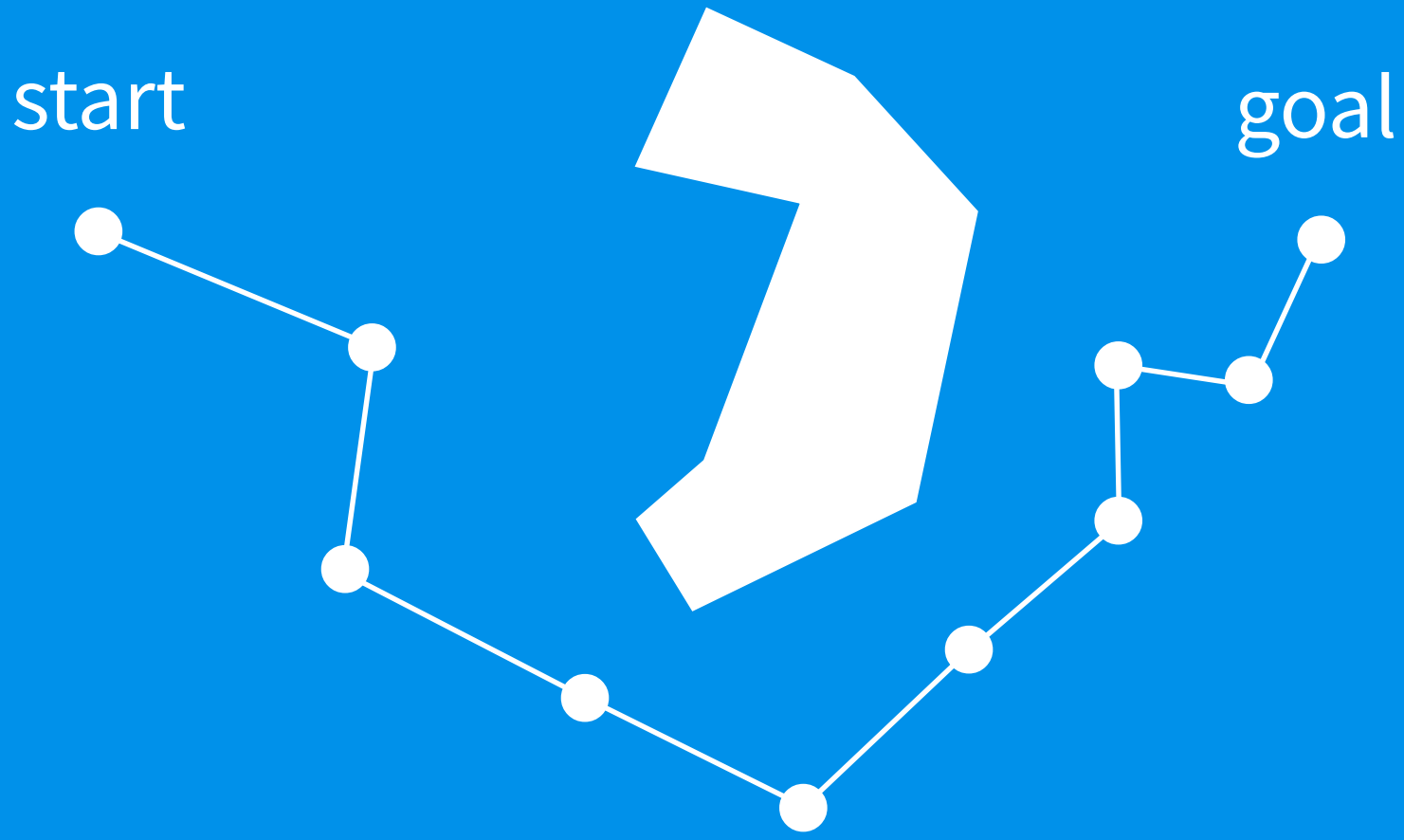

goal



goal

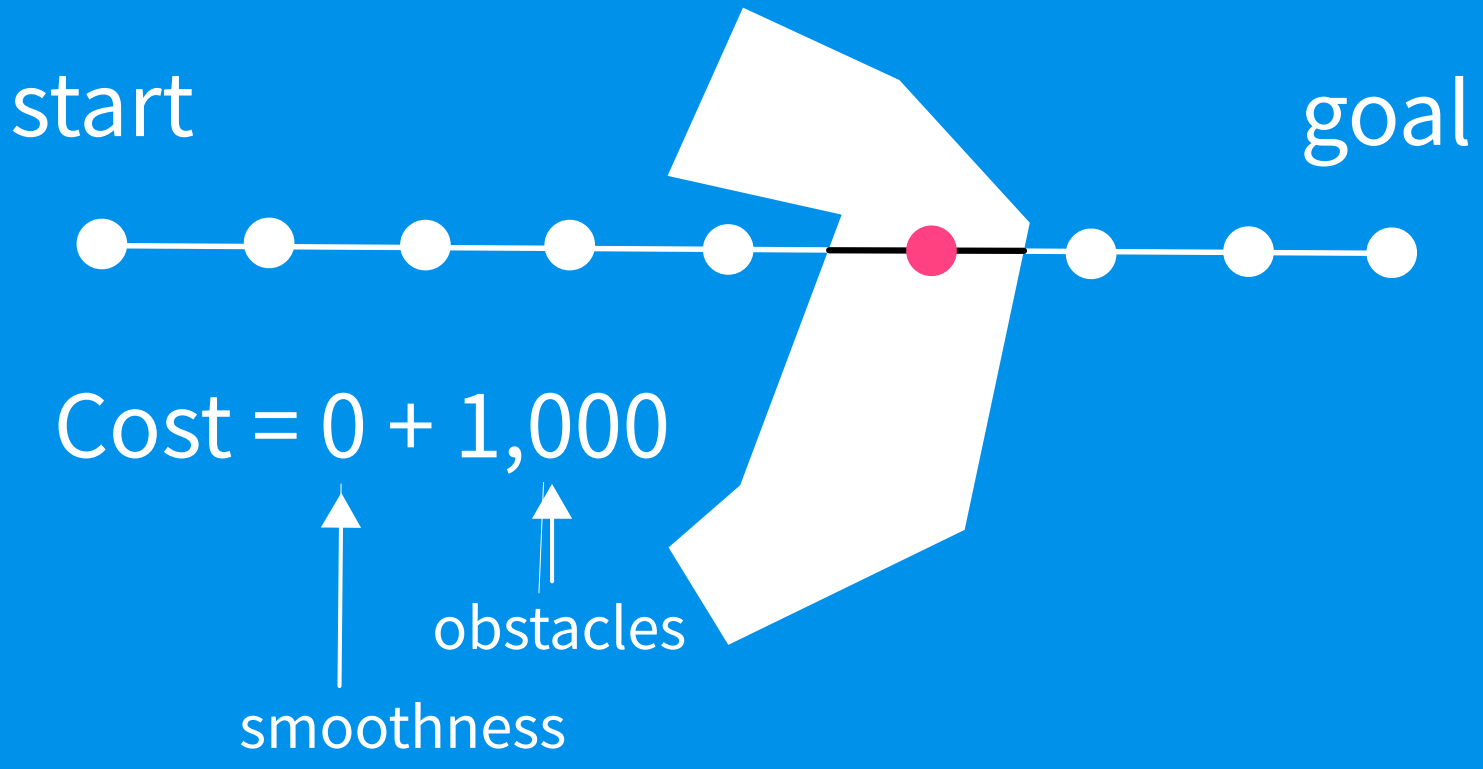

goal

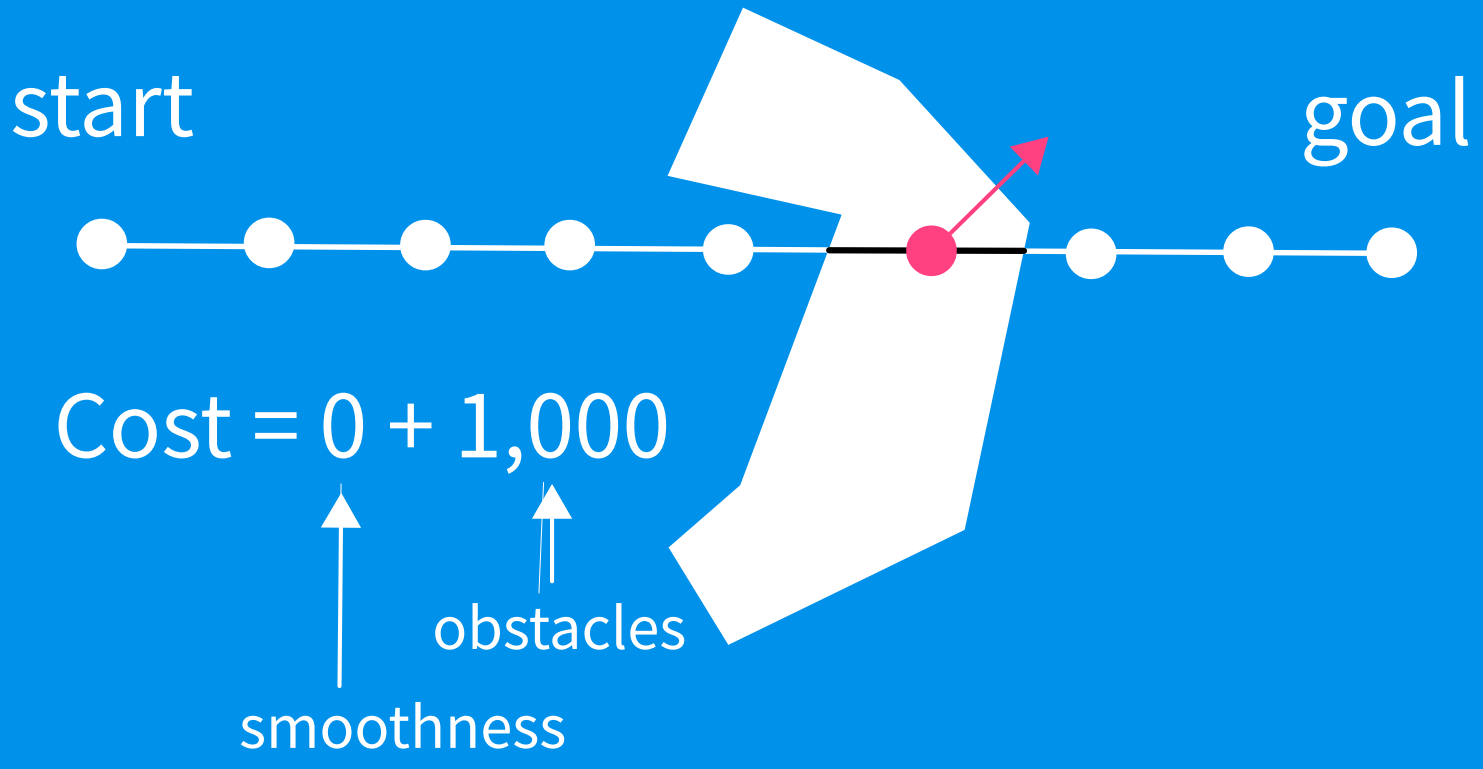

goal

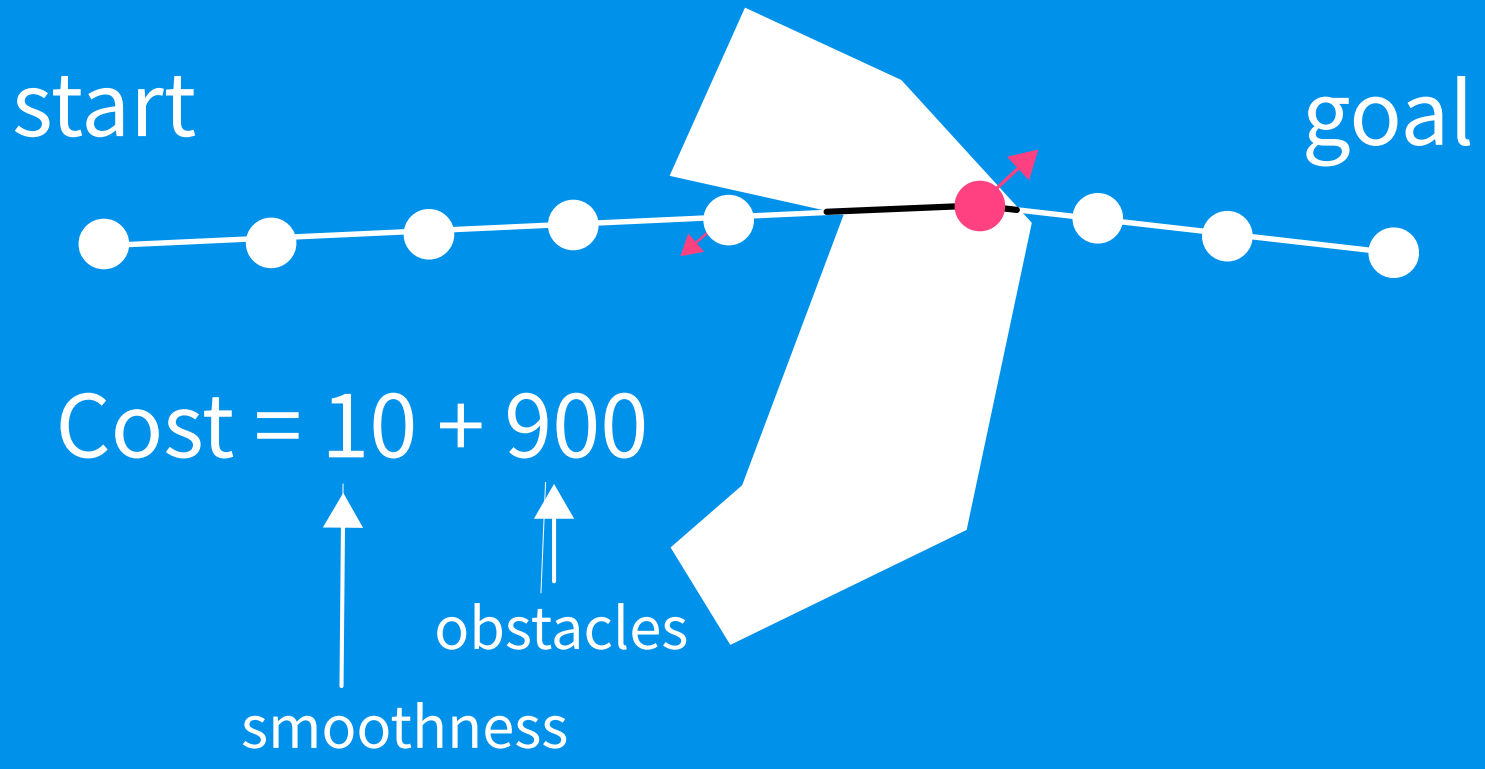

start
goal

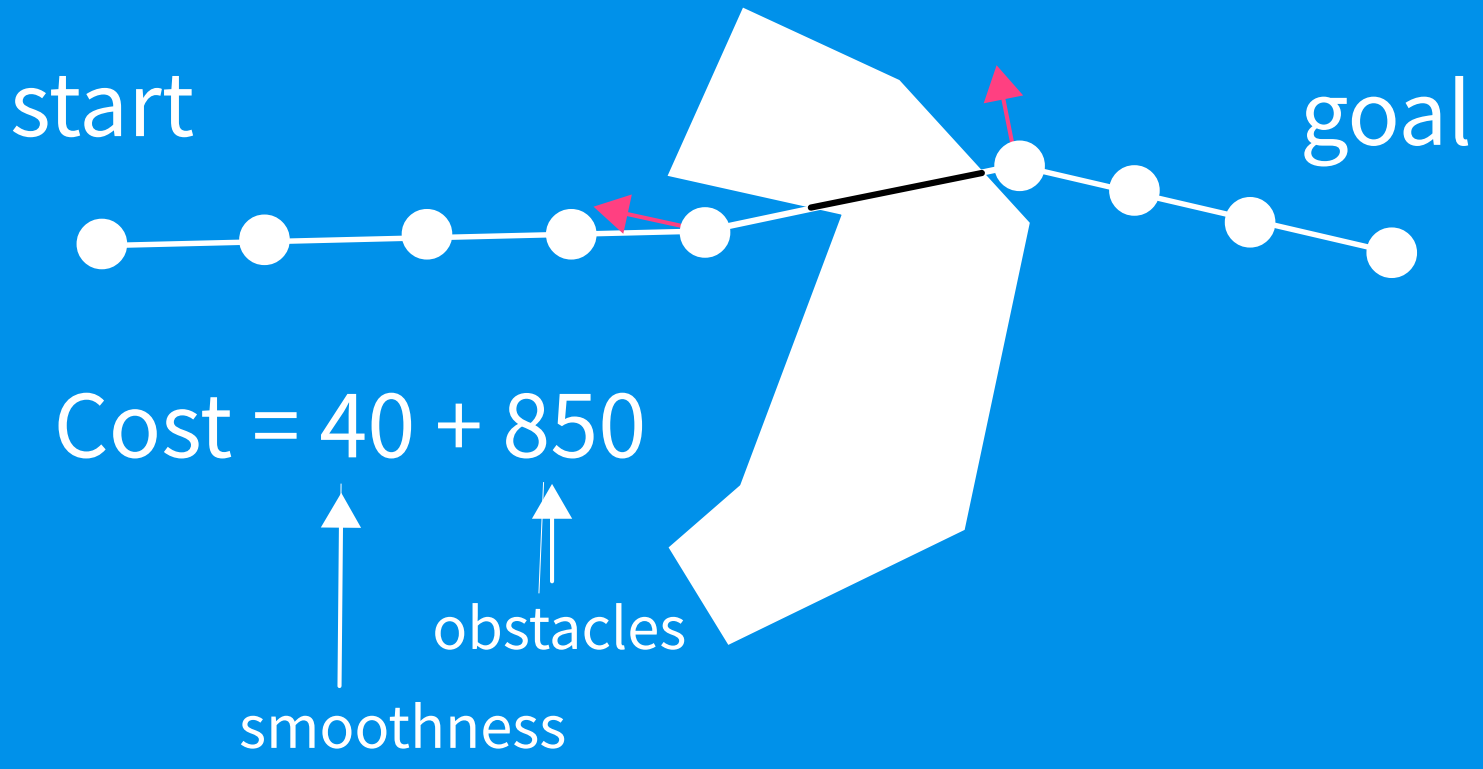
start

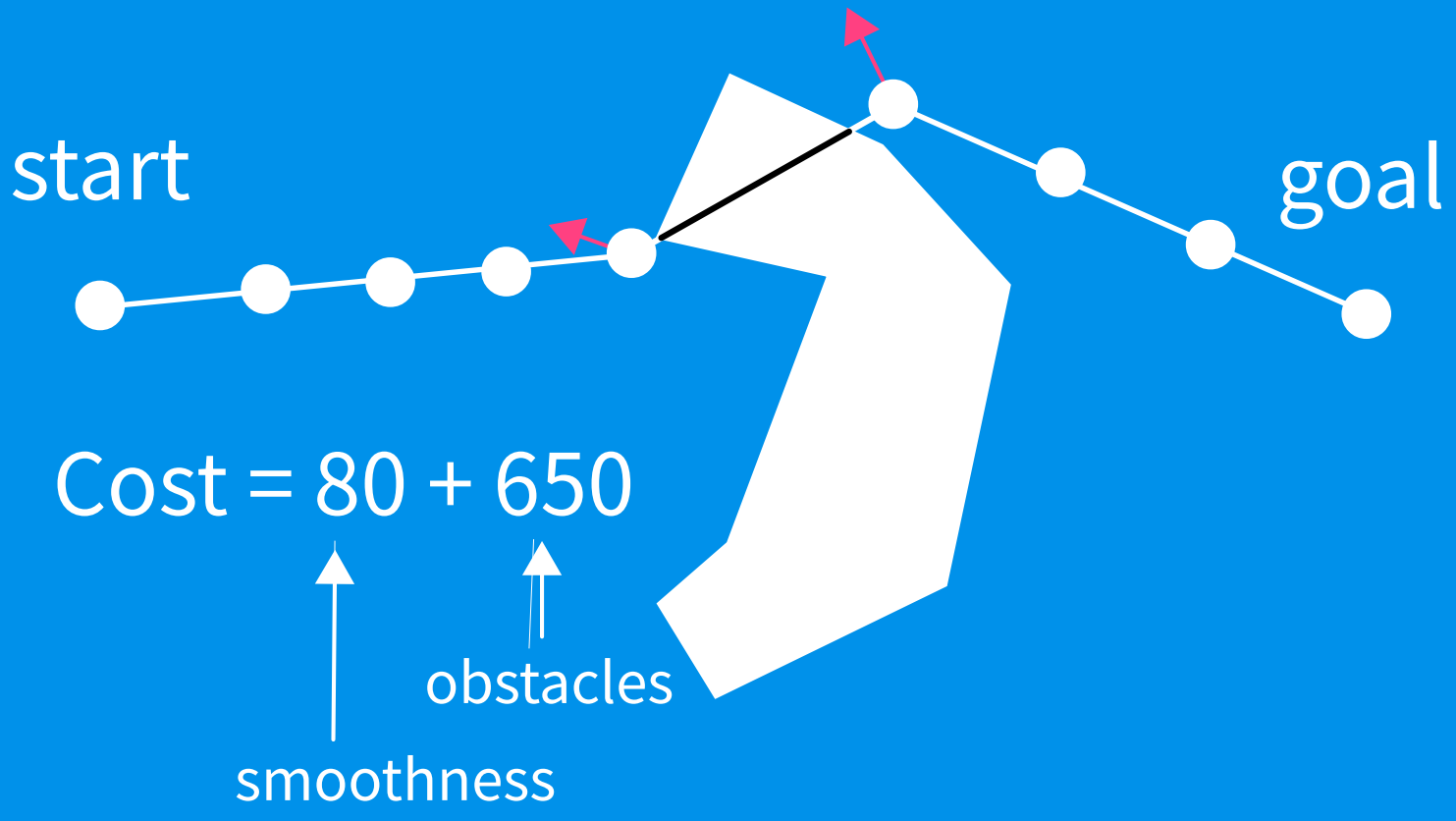
goal


start
goal
Cost = 0 + 1,000
obstacles
smoothness


start
goal
Cost = 0 + 1,000
obstacles
smoothness


start
goal
Cost = 10 + 900
obstacles
smoothness


start
goal
Cost = 40 + 850
obstacles
smoothness


start
goal
Cost = 80 + 650
obstacles
smoothness


start
goal
Cost = 200 + 0
obstacles
smoothness