

# CSE 478 Robot Autonomy

## Lazy Search

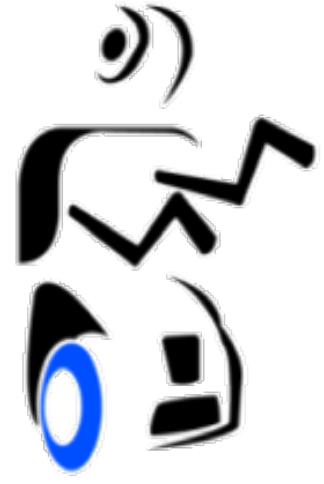
Siddhartha Srinivasa (siddh@)  
Abhishek Gupta (abhgupta@)

TAs:  
Rohan Baijal (rbaijal@)  
Sidhartha Talia (sidtalia@)  
Christopher Tan (tan7271@)  
Helen Wang (yiruwang@)

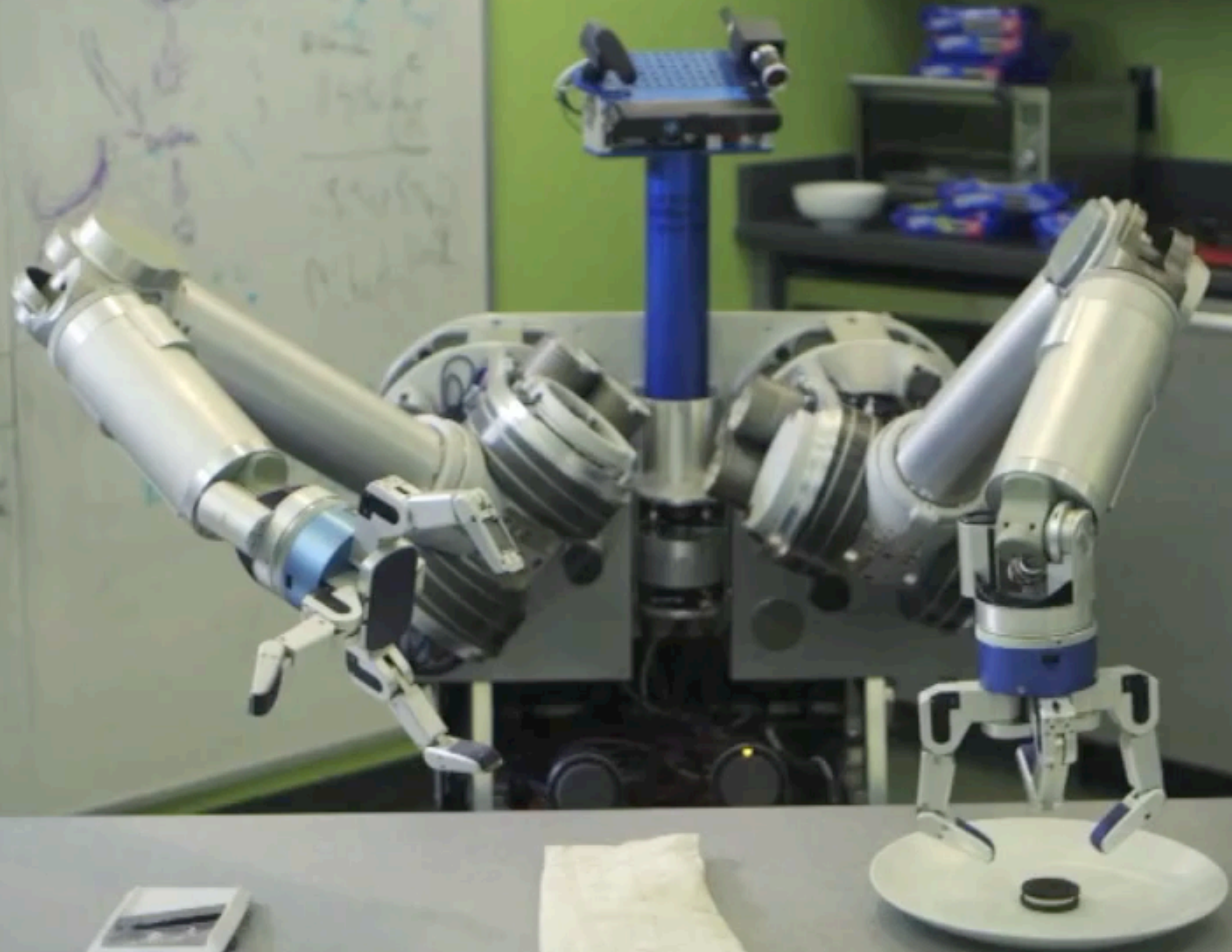
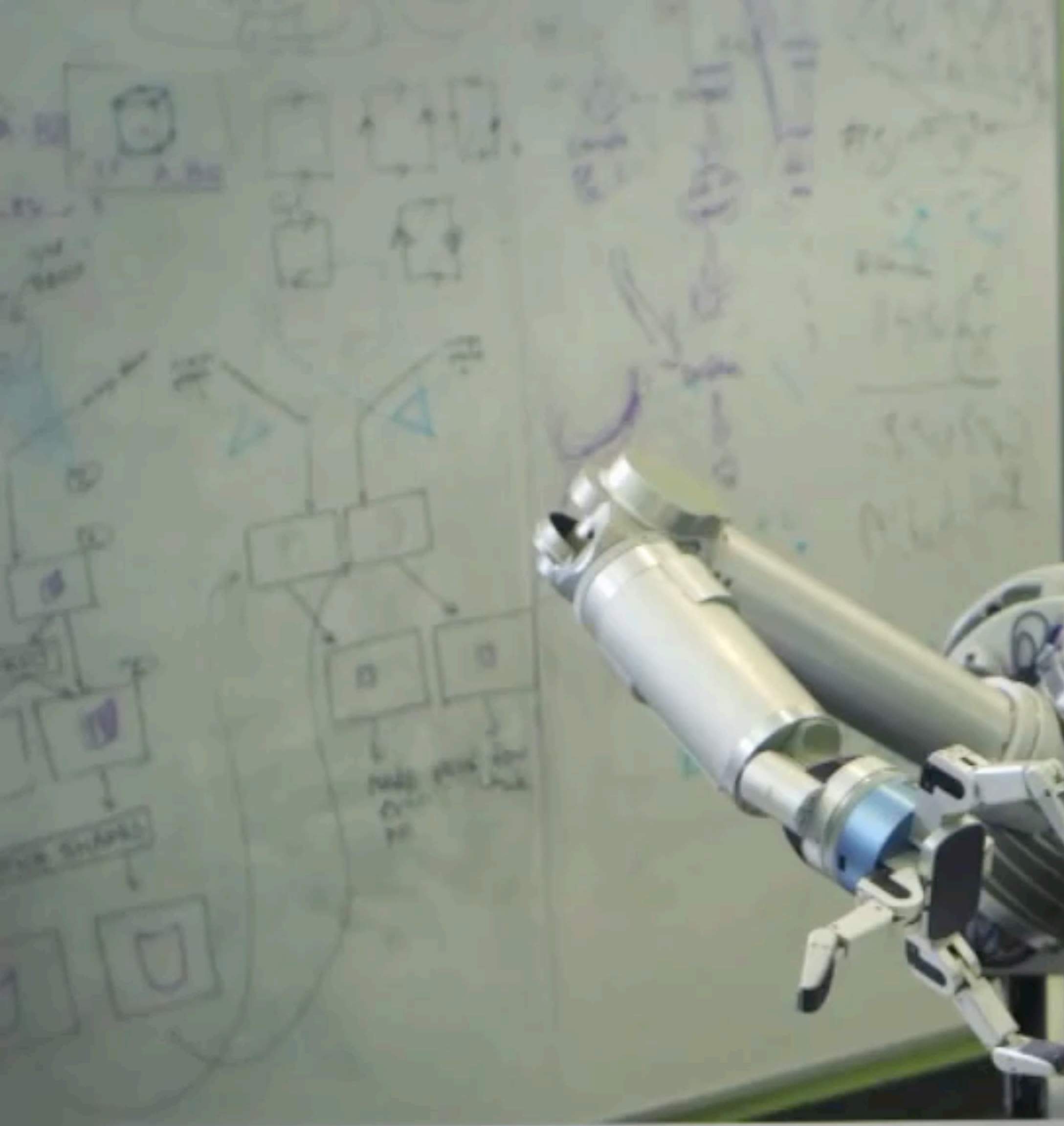


# Upcoming Deliverables

- Planning project demos: May 16, 2024 1:30 - 3:30 pm
- Discussion post 7: May 16, 2024 @ 11:59 pm
- Quiz - Planning and Learning: Monday, May 20, 2024 in class
- Planning Writeup: Monday, May 20, 2024 @ 11:59 pm
- Final Project Demos & Writeup: May 31 (Time TBD)



# Motion Planning





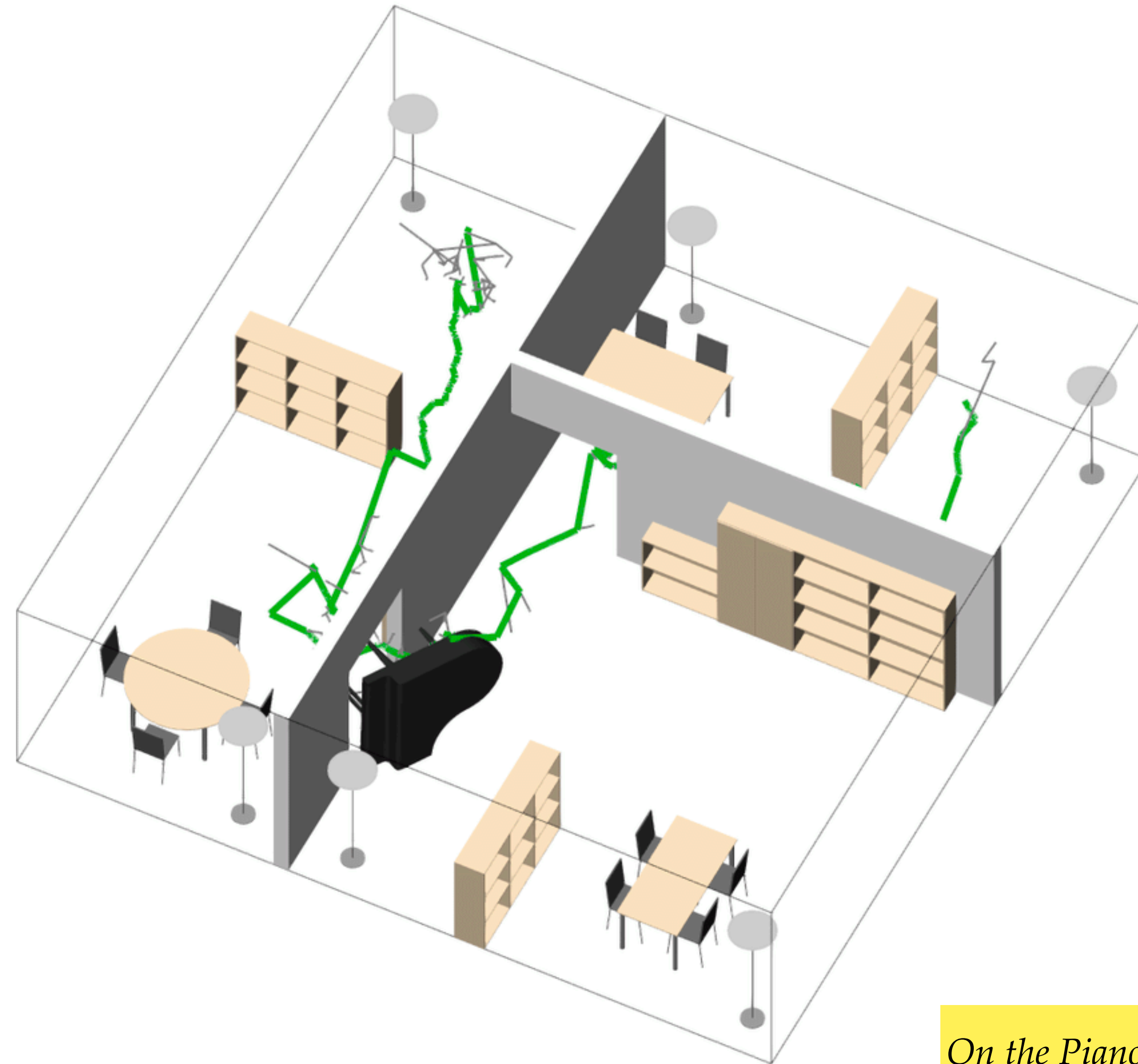




**10-100X**

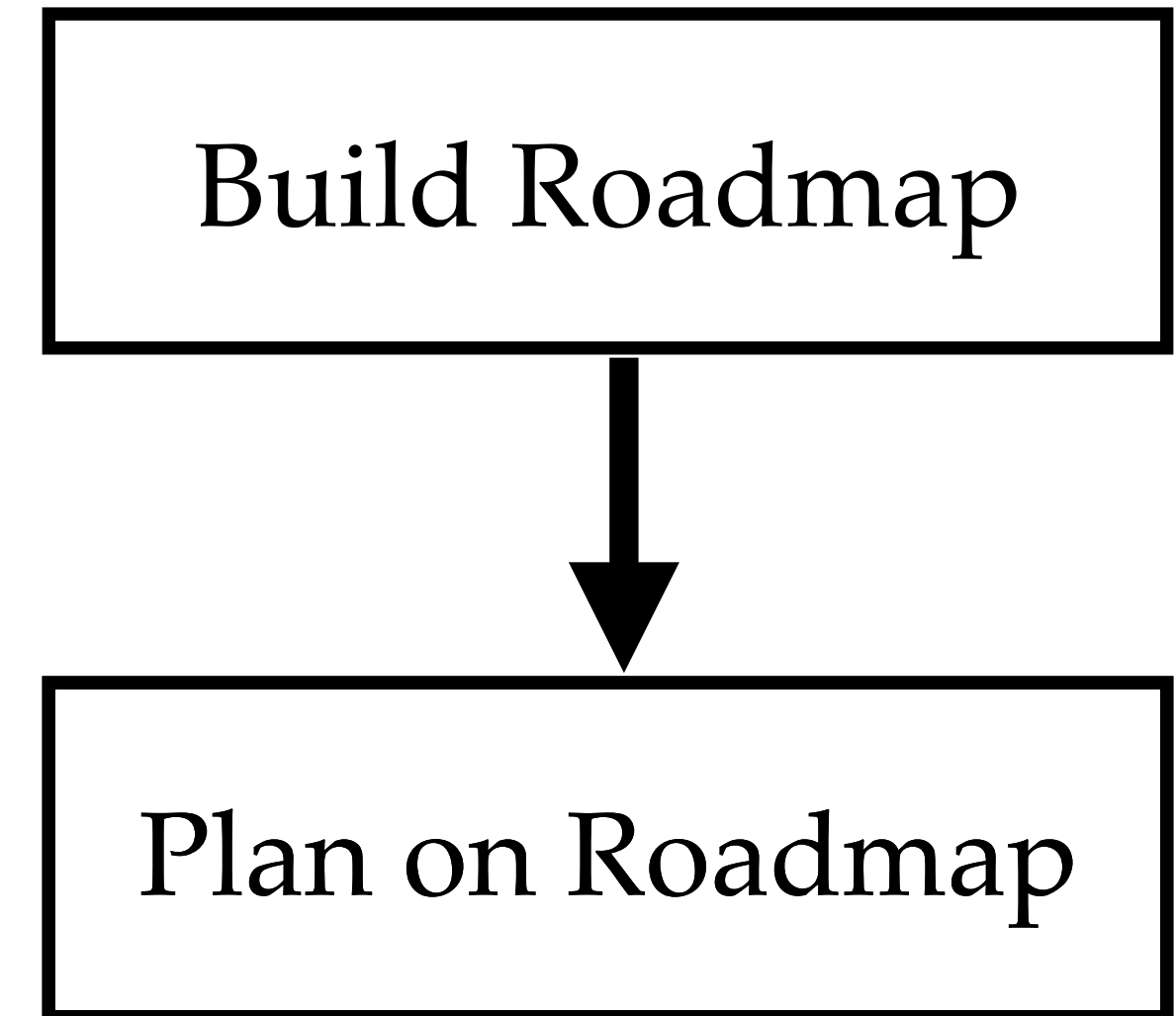
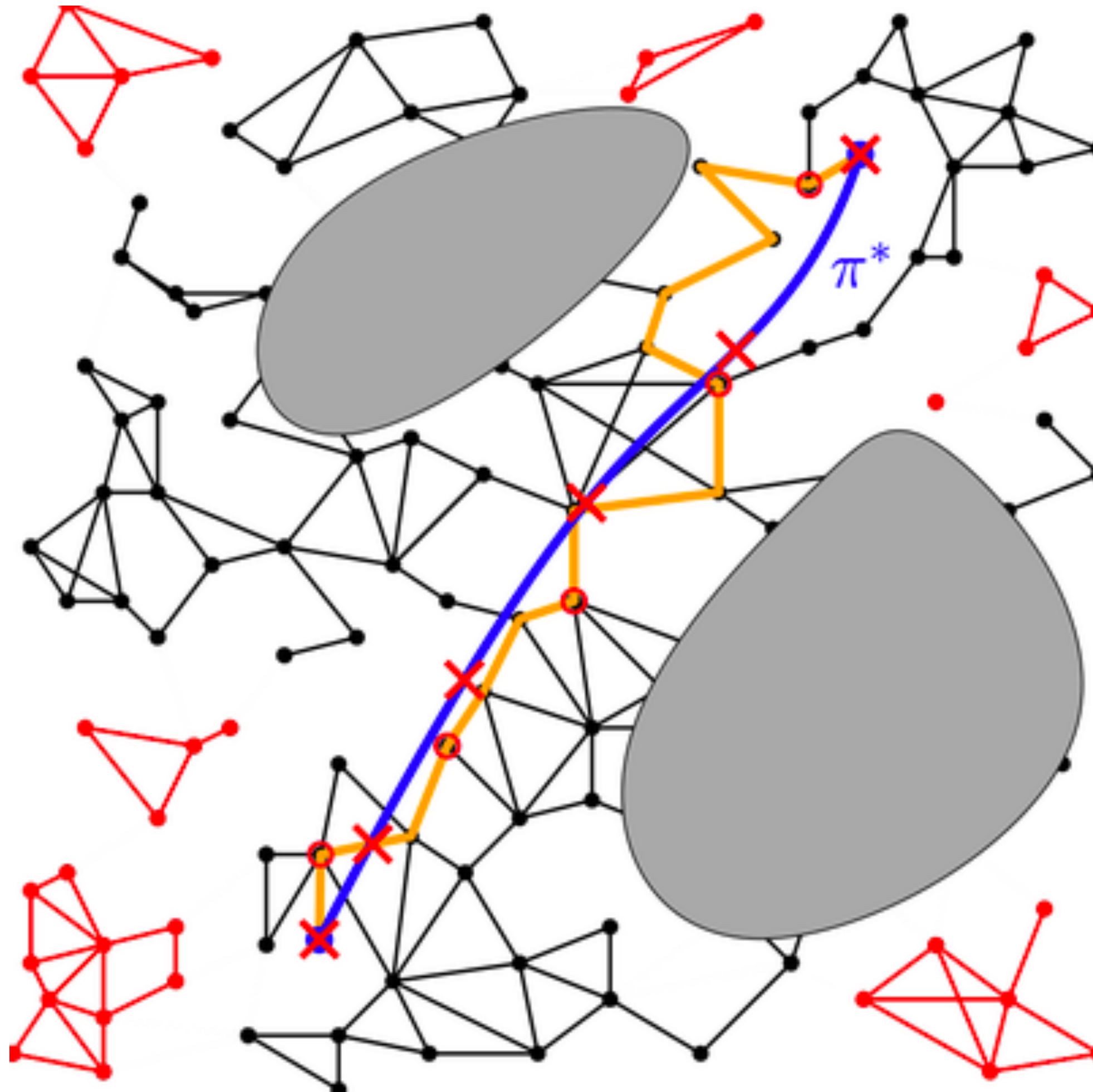
**Improvement**

# The Piano Movers' Problem



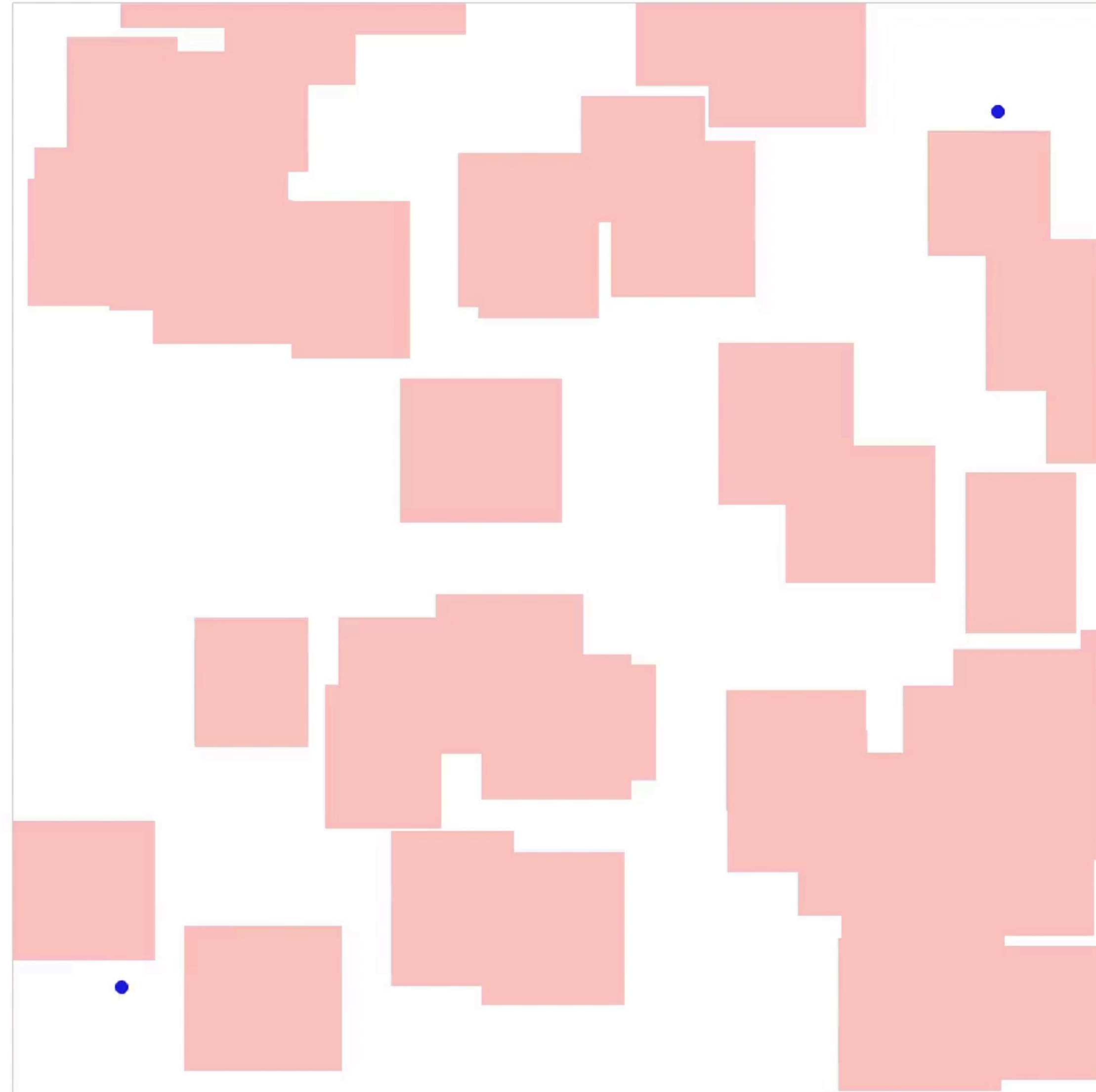
*On the Piano Movers problem. I-III, Schwartz and Sharir, Comm. on Pure and Applied Math., 1983*

# Roadmaps



*Probabilistic roadmaps for path planning in high-dimensional configuration spaces, Kavraki et al., IEEE TRO, 1996.*

# A\* Search



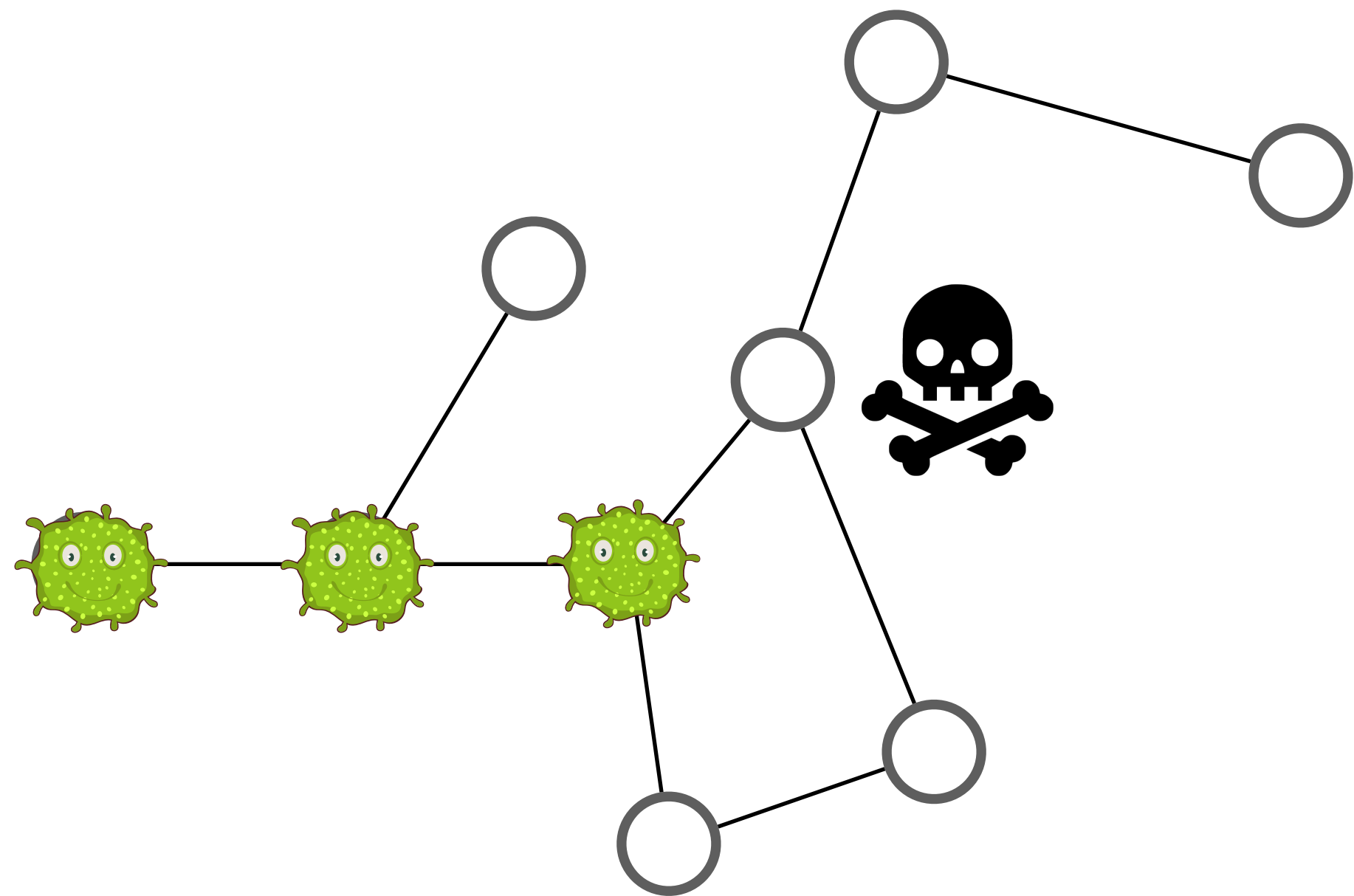
# A\* Search



**OPTIMAL!!**

Is it optimal over something we care about?

# A\* Search: Amoebas!



## Optimal Substructure

$$f(a) < f(b) \implies f(a \circ x) < f(b \circ x) \forall x$$

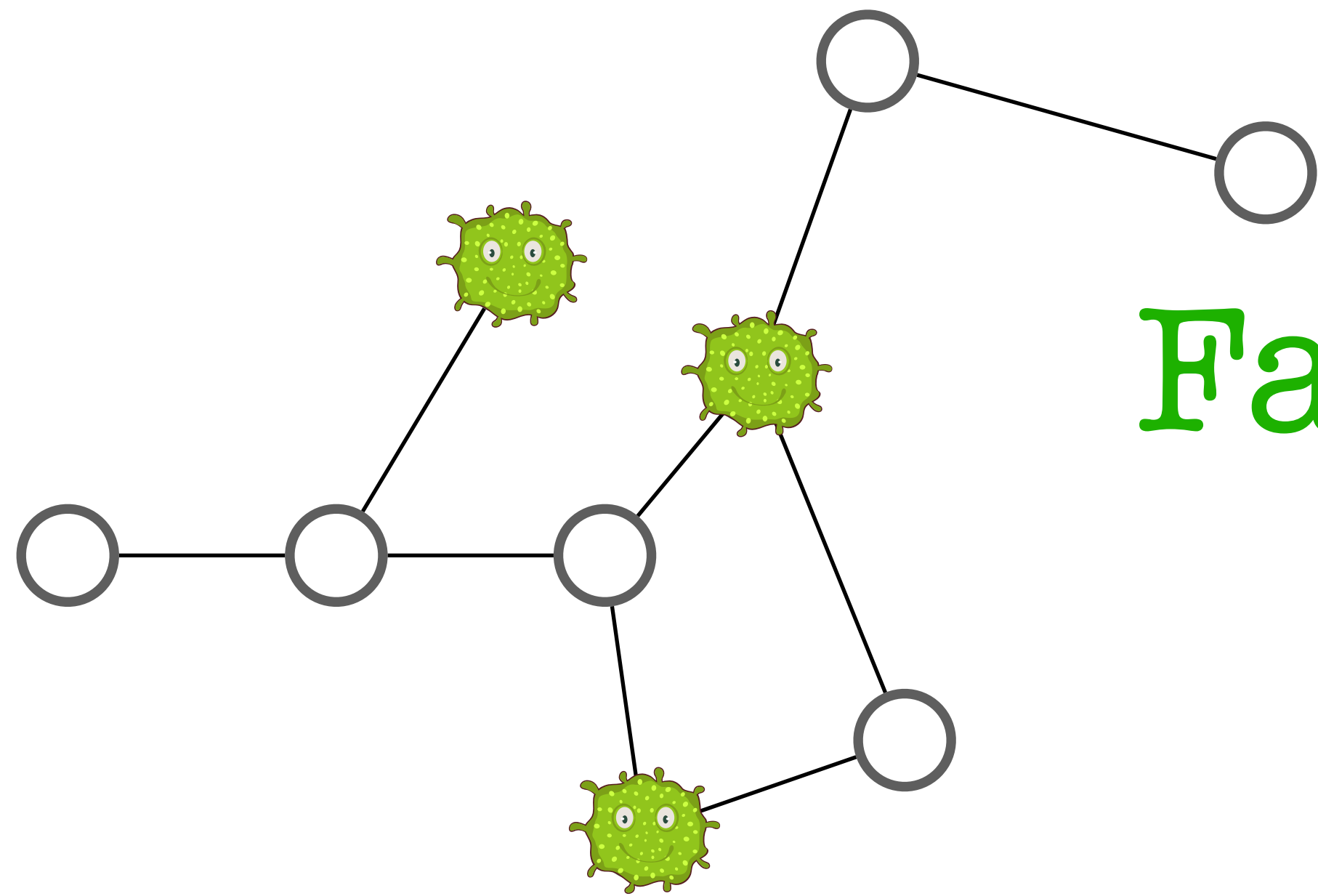
*You will never catch up.*

## Bellman Condition

$$f^*(a) = \min_{x \in \text{SUCC}} \{c(a, x) + f^*(x)\}$$

*Be best, locally.*

# A\* Search: Favoritism



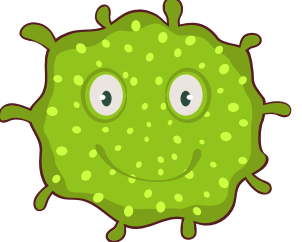
Optimism in the  
Face of Uncertainty (OFU)

$$\min_{x \in \text{open}} g(x) + h(x)$$



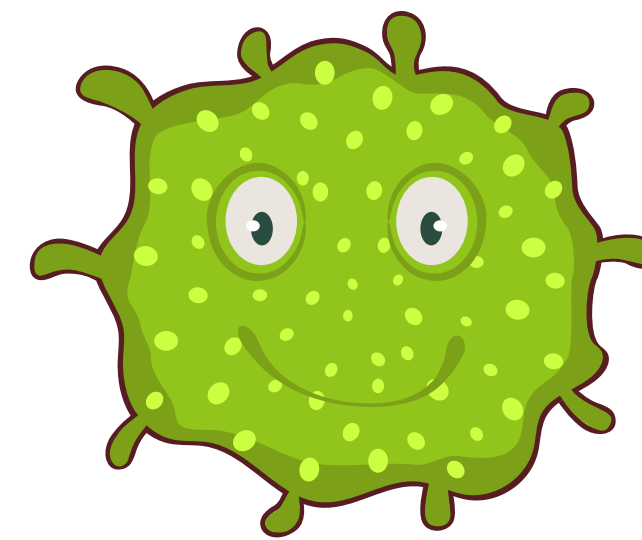
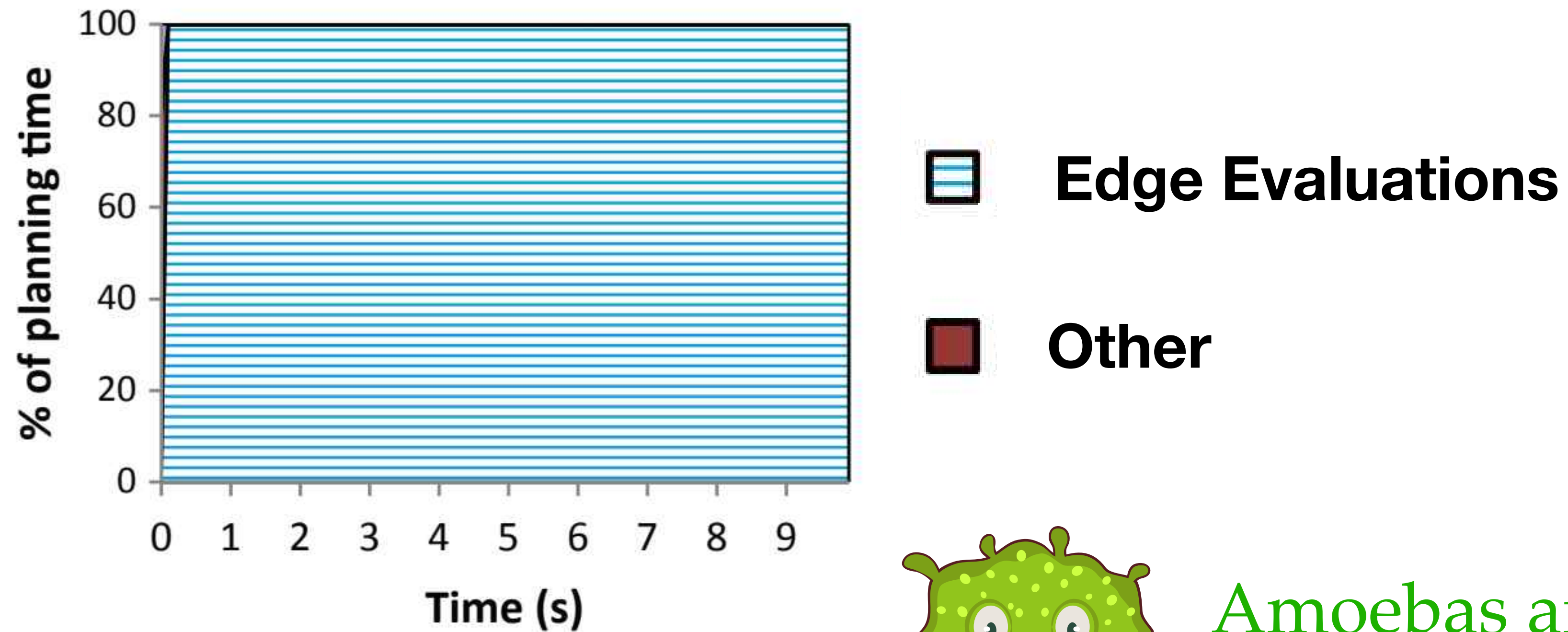
*Always be optimistic under uncertainty.  
You'll either be correct,  
or learn something important if you're wrong.*

A\* Search is **Optimal** ...

Expands the Fewest Number of Vertices 

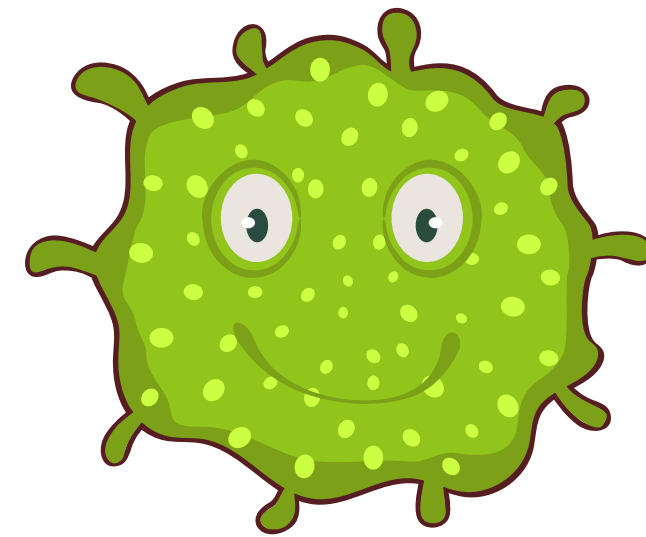
But is this what we  
*really* want in Motion Planning?

# Edge Evaluation **Dominates** Planning Time



Amoebas are Cheap  
Slime is Expensive

Is there a Search Algorithm  
that **Minimizes**  
the Number of Edge Evaluations?



I don't care about amoebas.  
What algorithm minimizes slime?

# LazySP

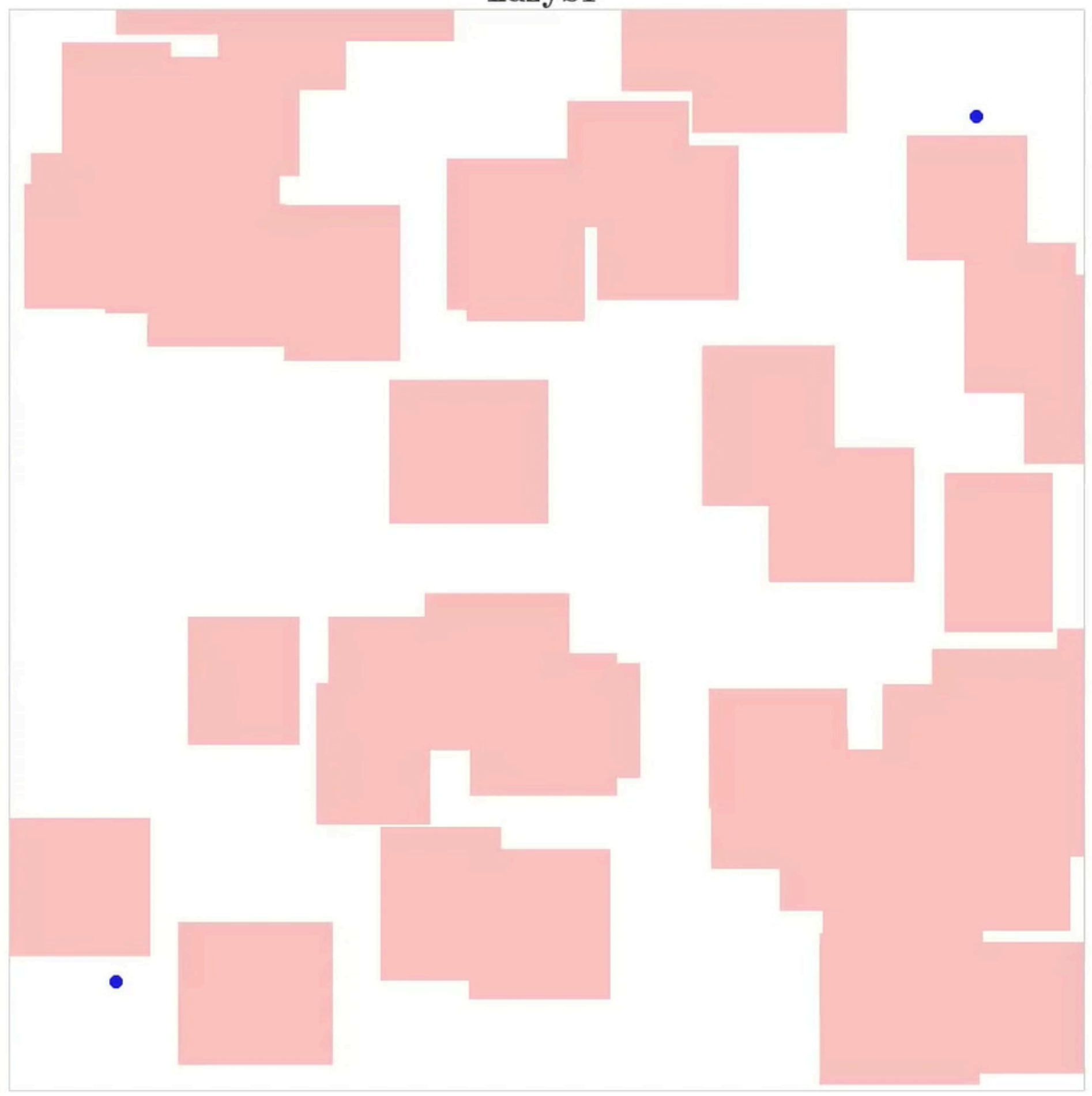
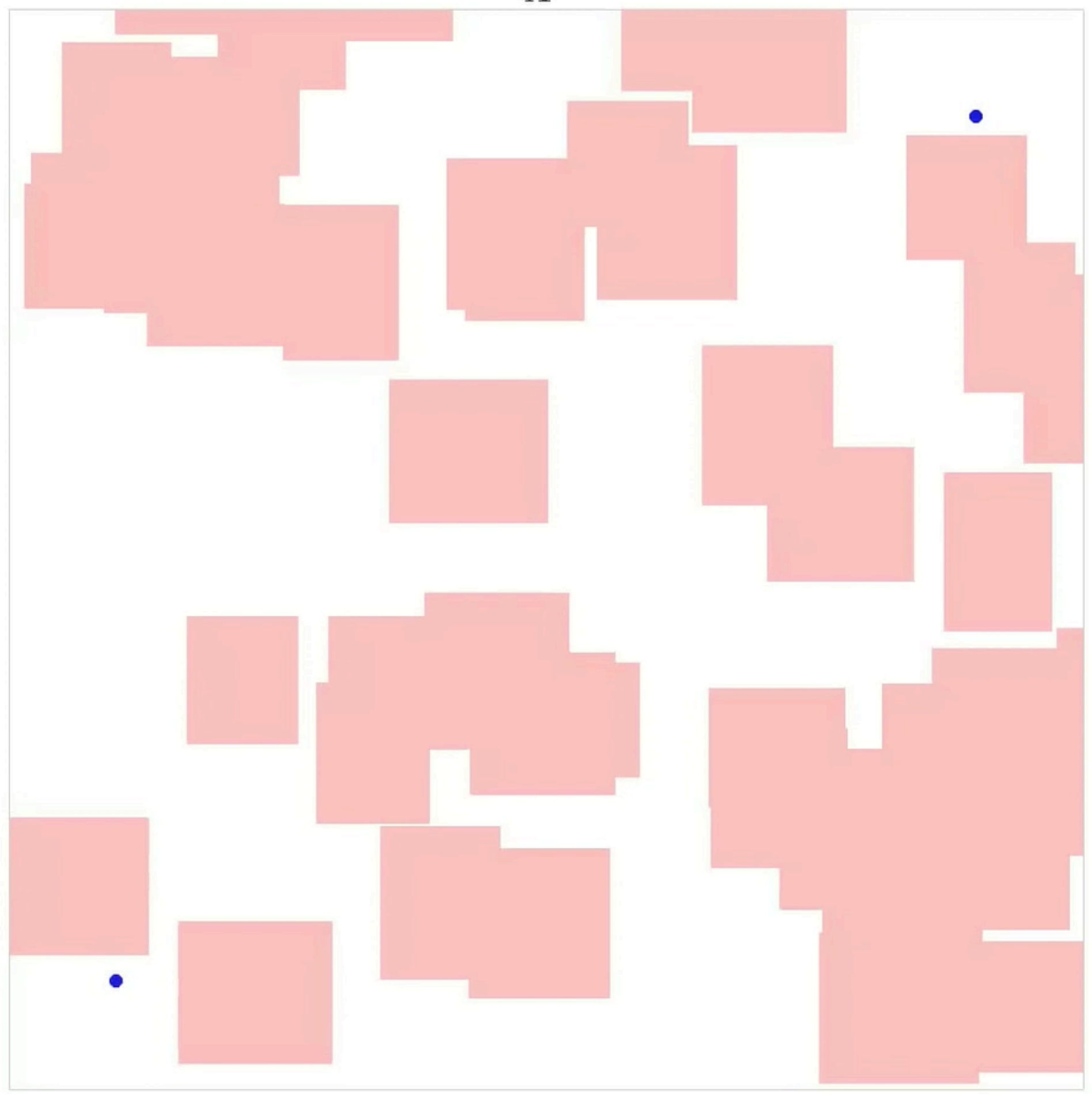
ICAPS 2018, 2019 [Best Conference Paper Award Winner]

**First Provably Edge-Optimal A\*-like Search Algorithm**

Number of Edges Evaluated

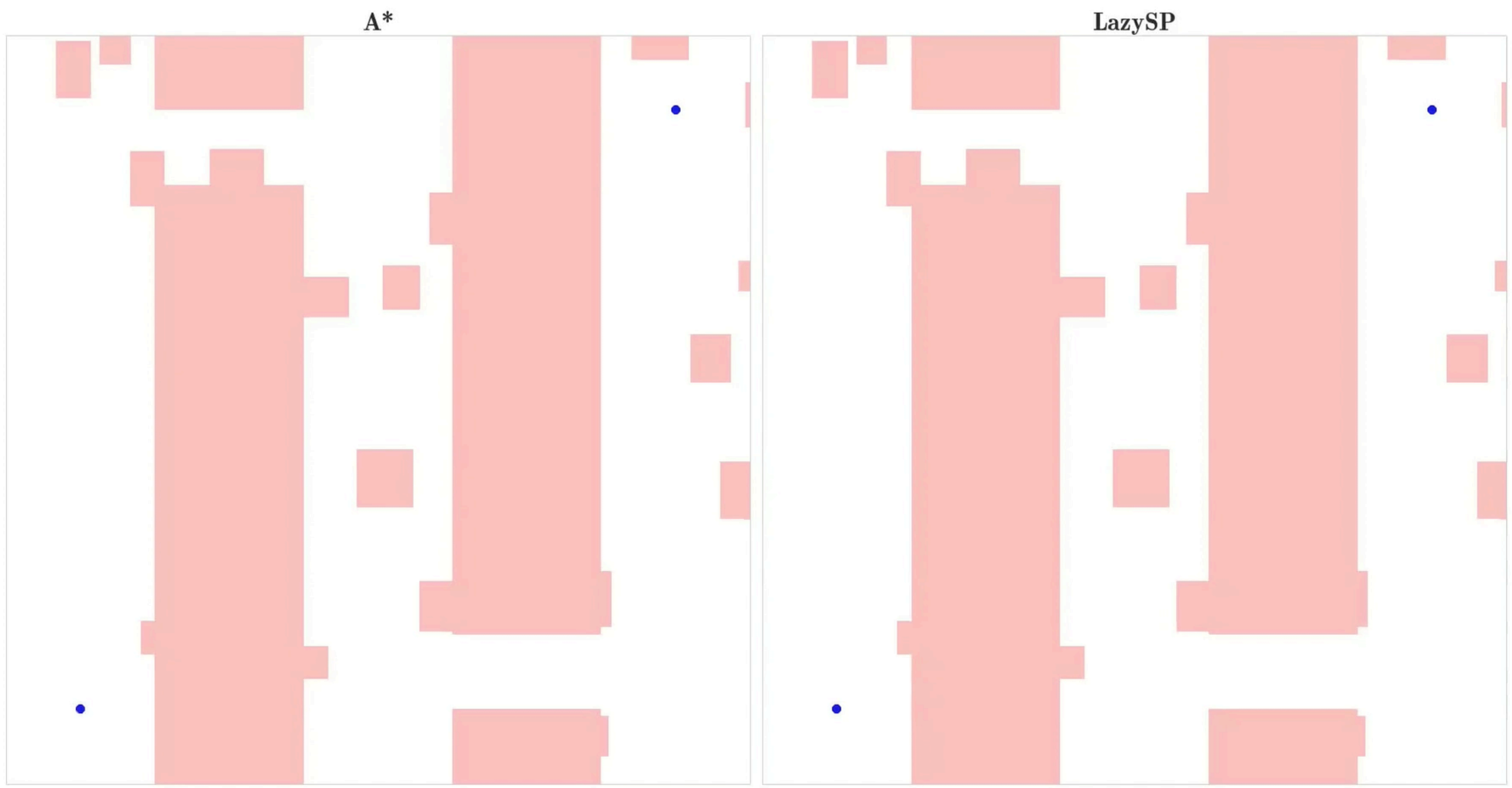
A\*

LazySP





Number of Edges Evaluated

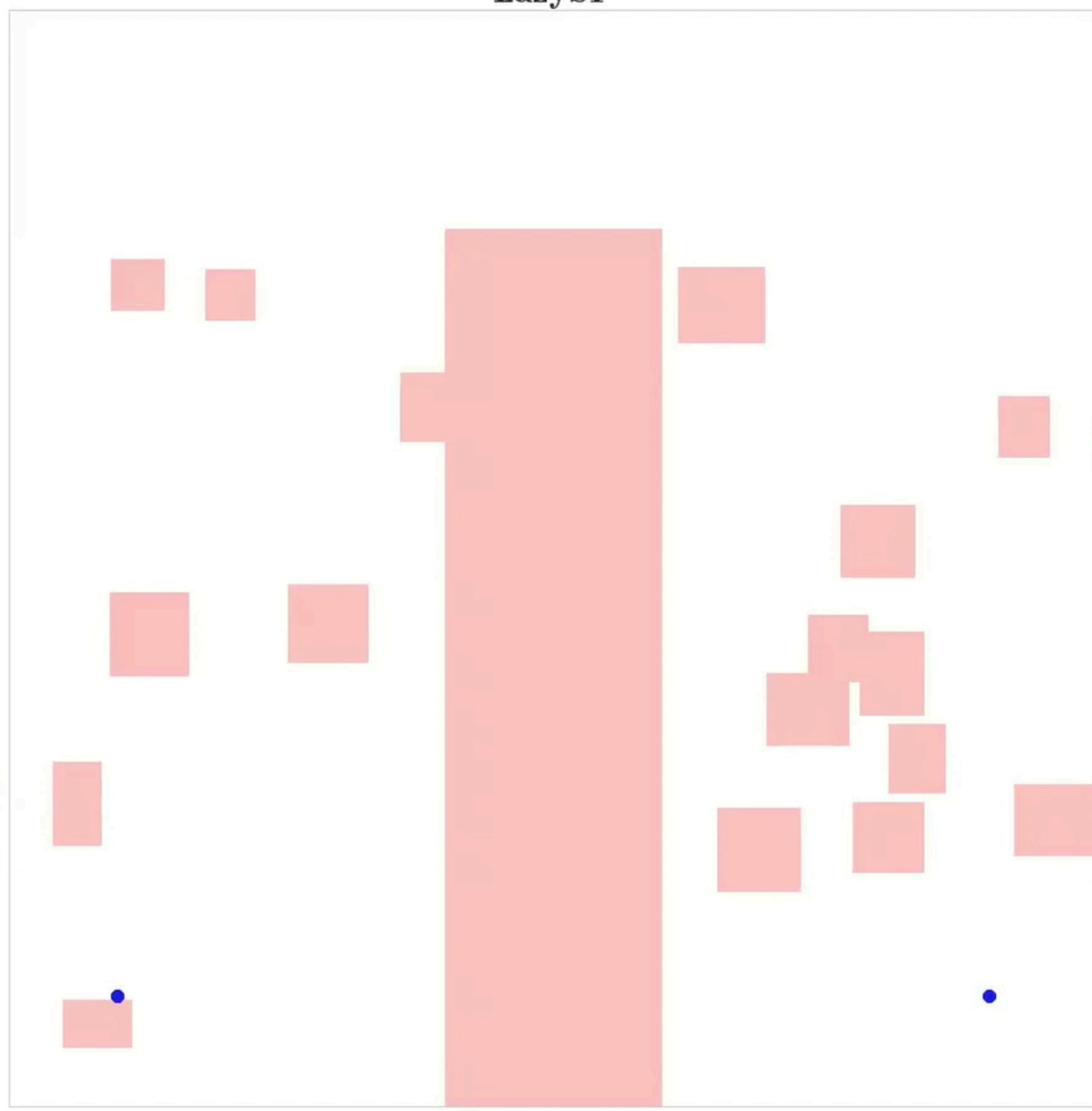
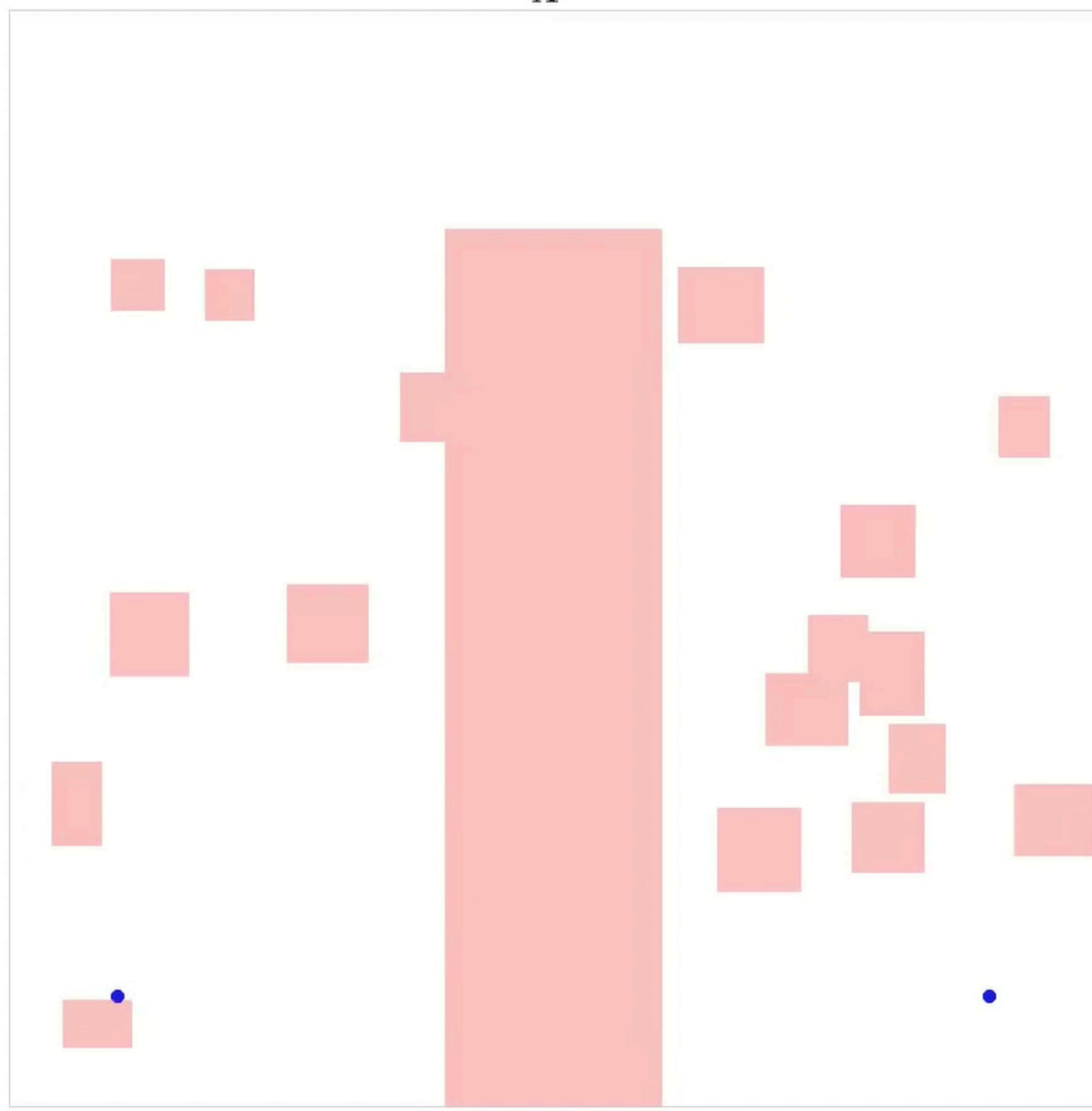




Number of Edges Evaluated

A\*

LazySP





# LazySP

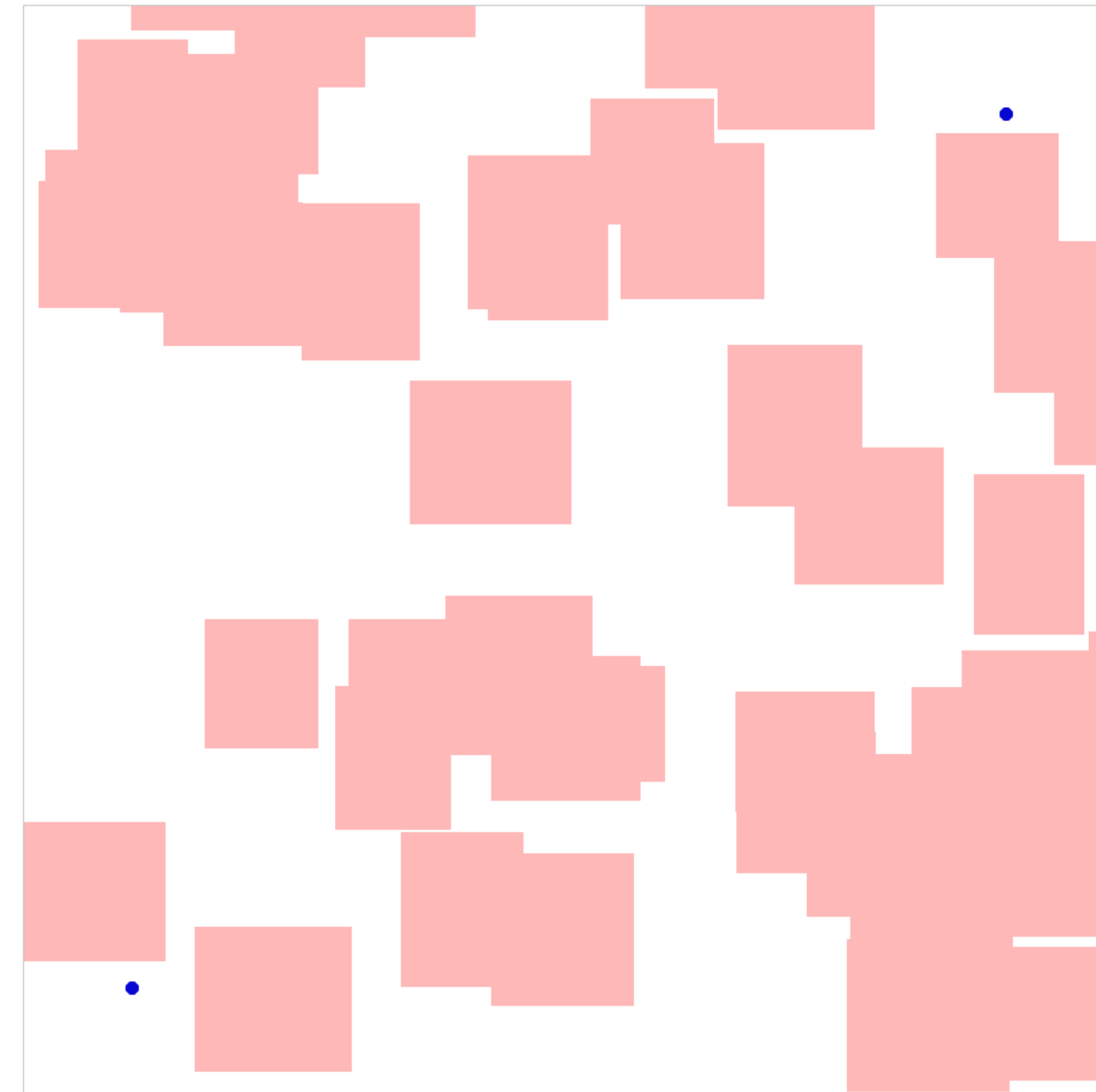
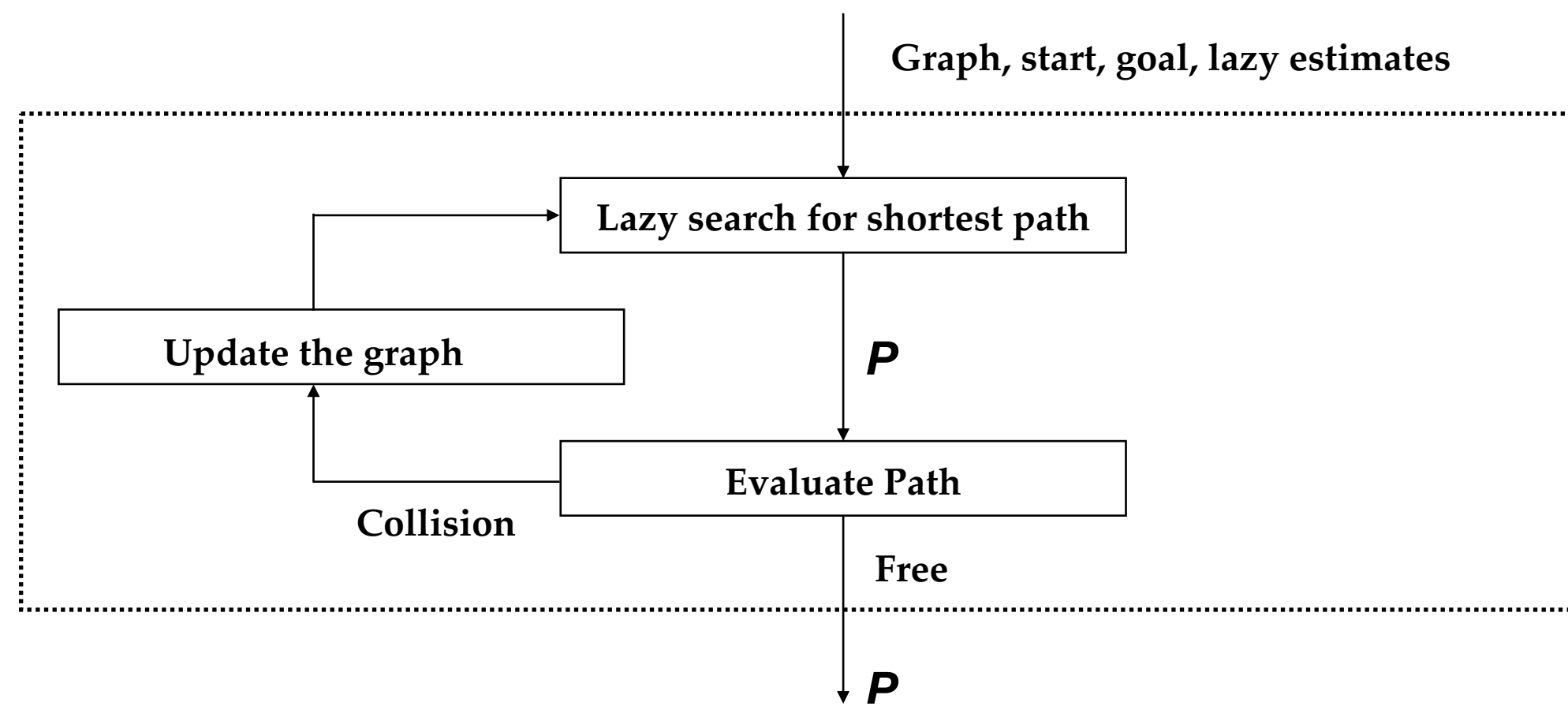
Greedy Best-first Search over **Paths**



*To find the shortest path,  
eliminate all shorter paths!*

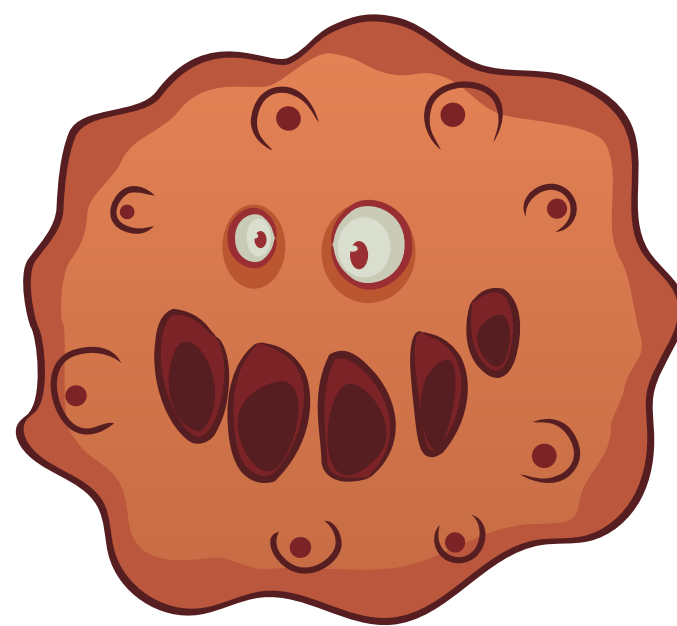
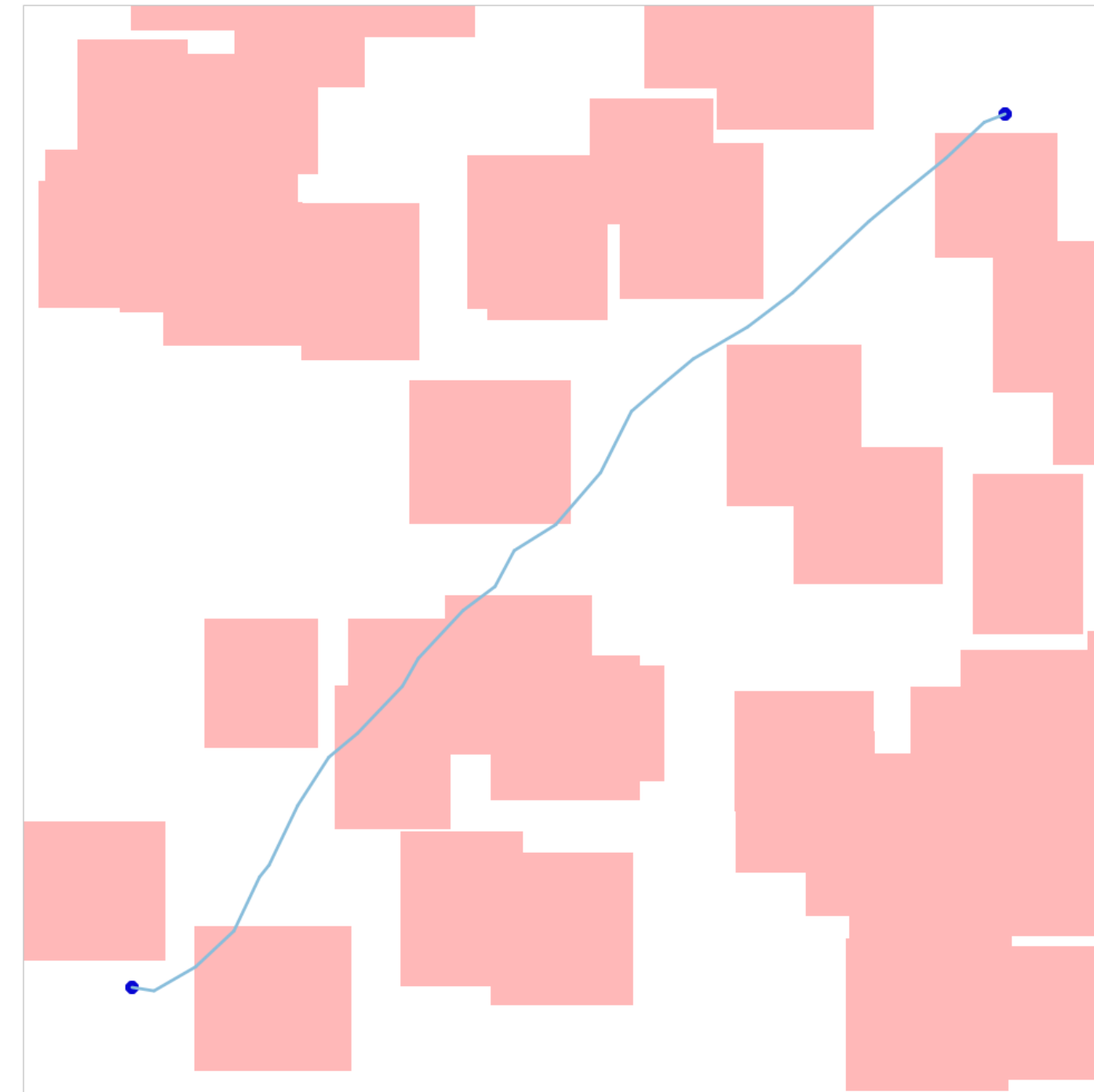
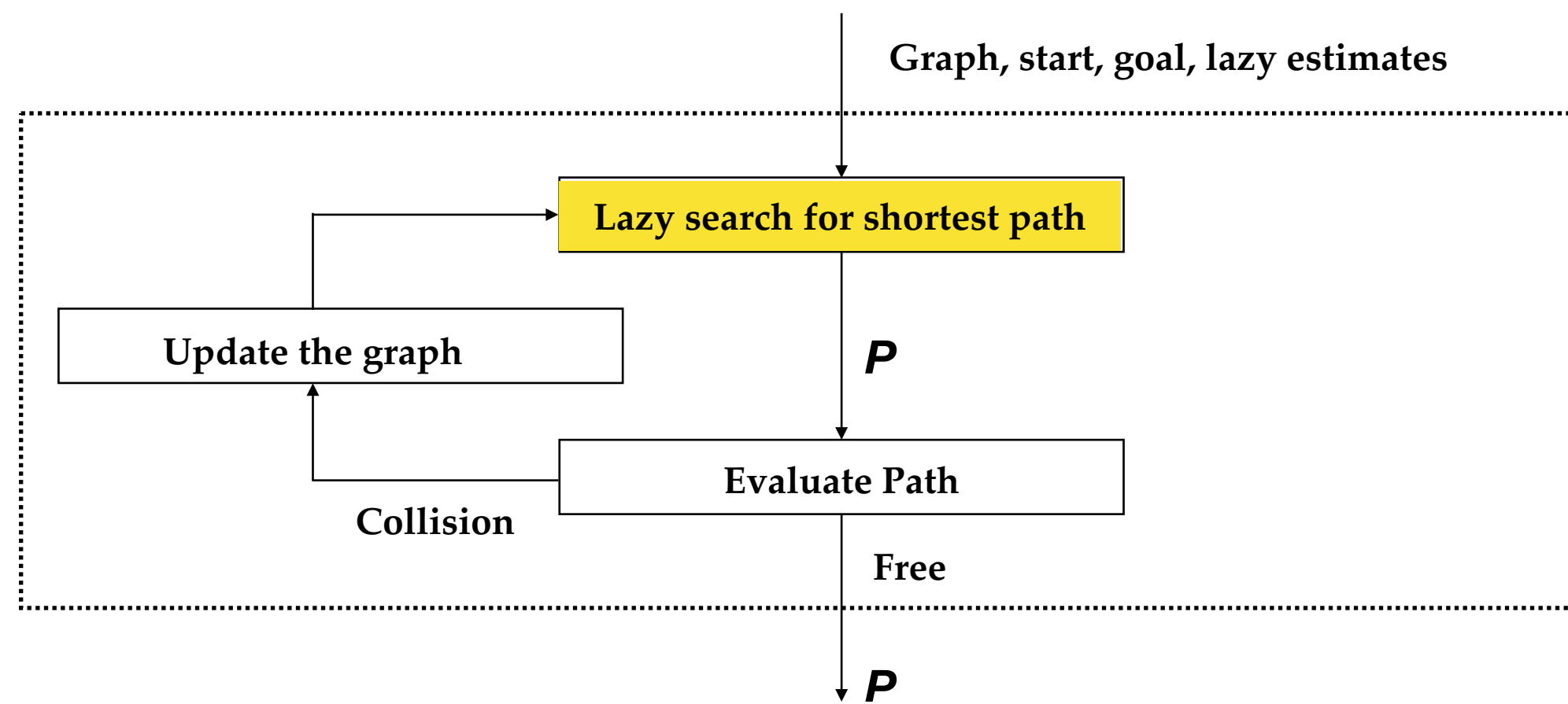
# LazySP

OFU on Steroids!



# LazySP

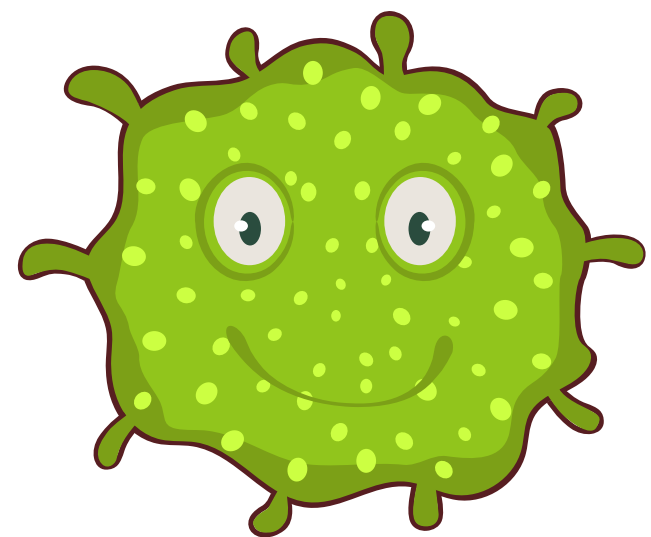
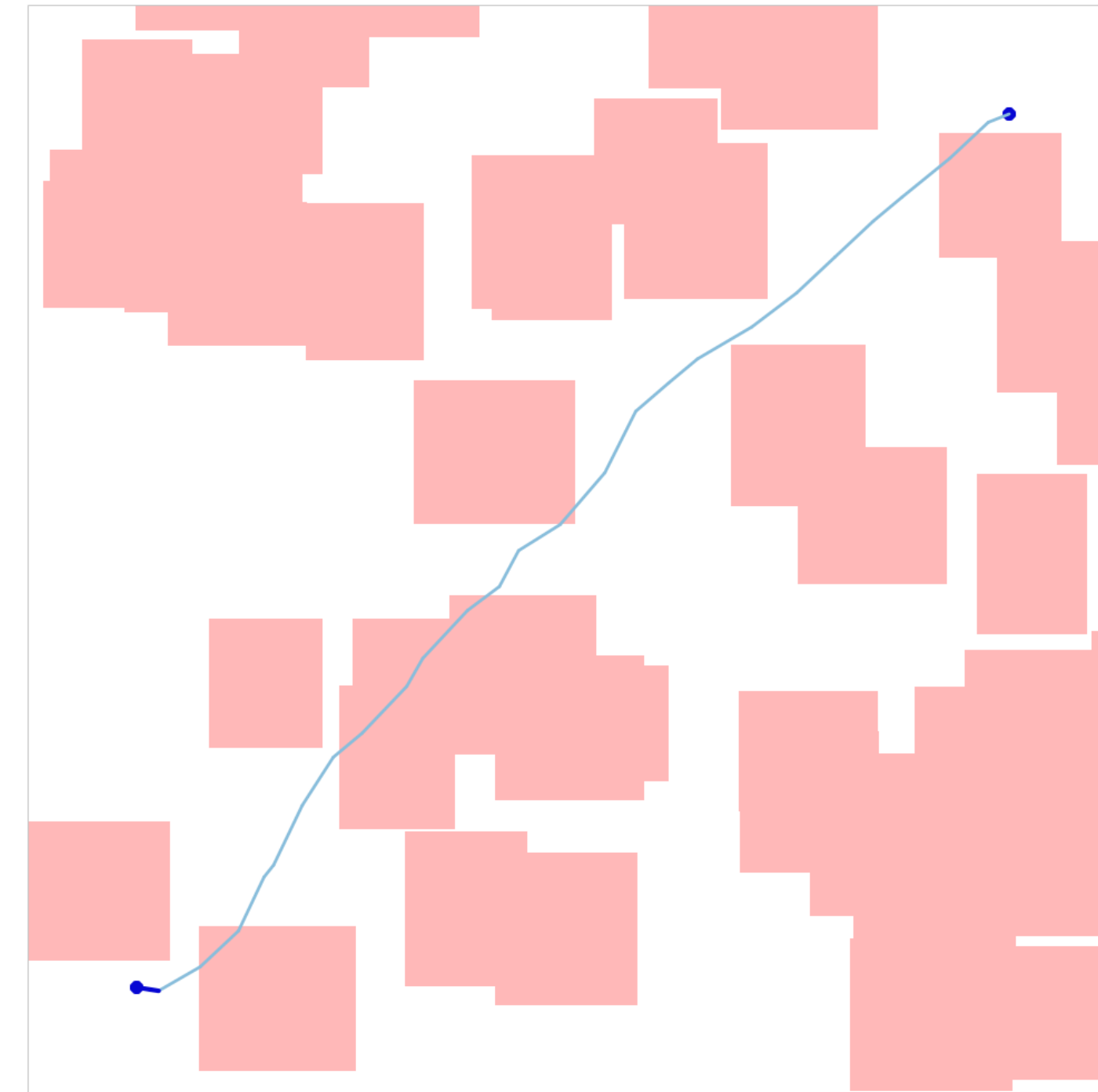
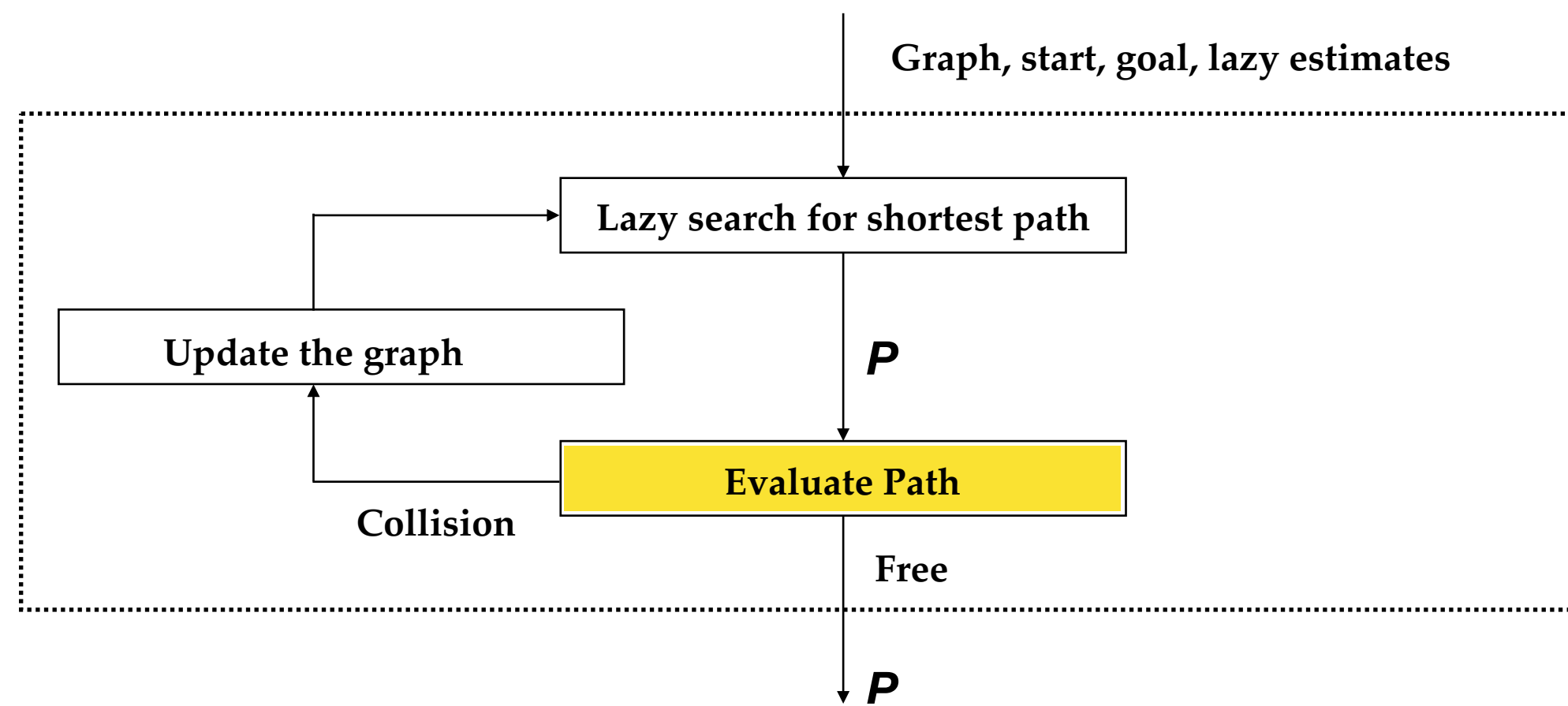
OFU on Steroids!



Send out the Ghost Amoebas

# LazySP

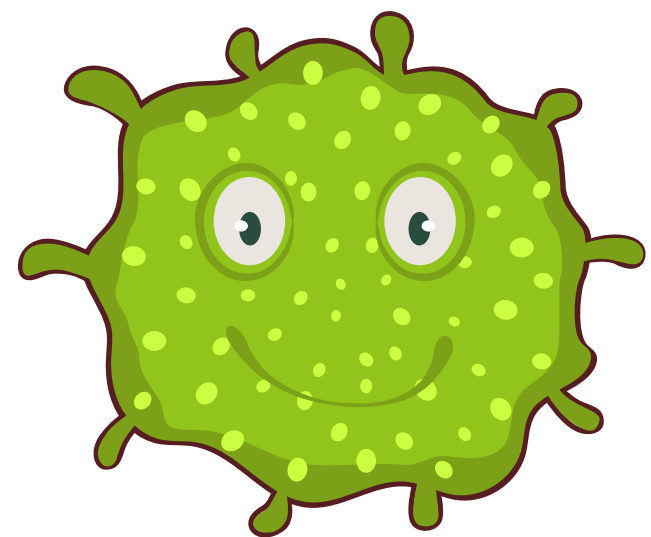
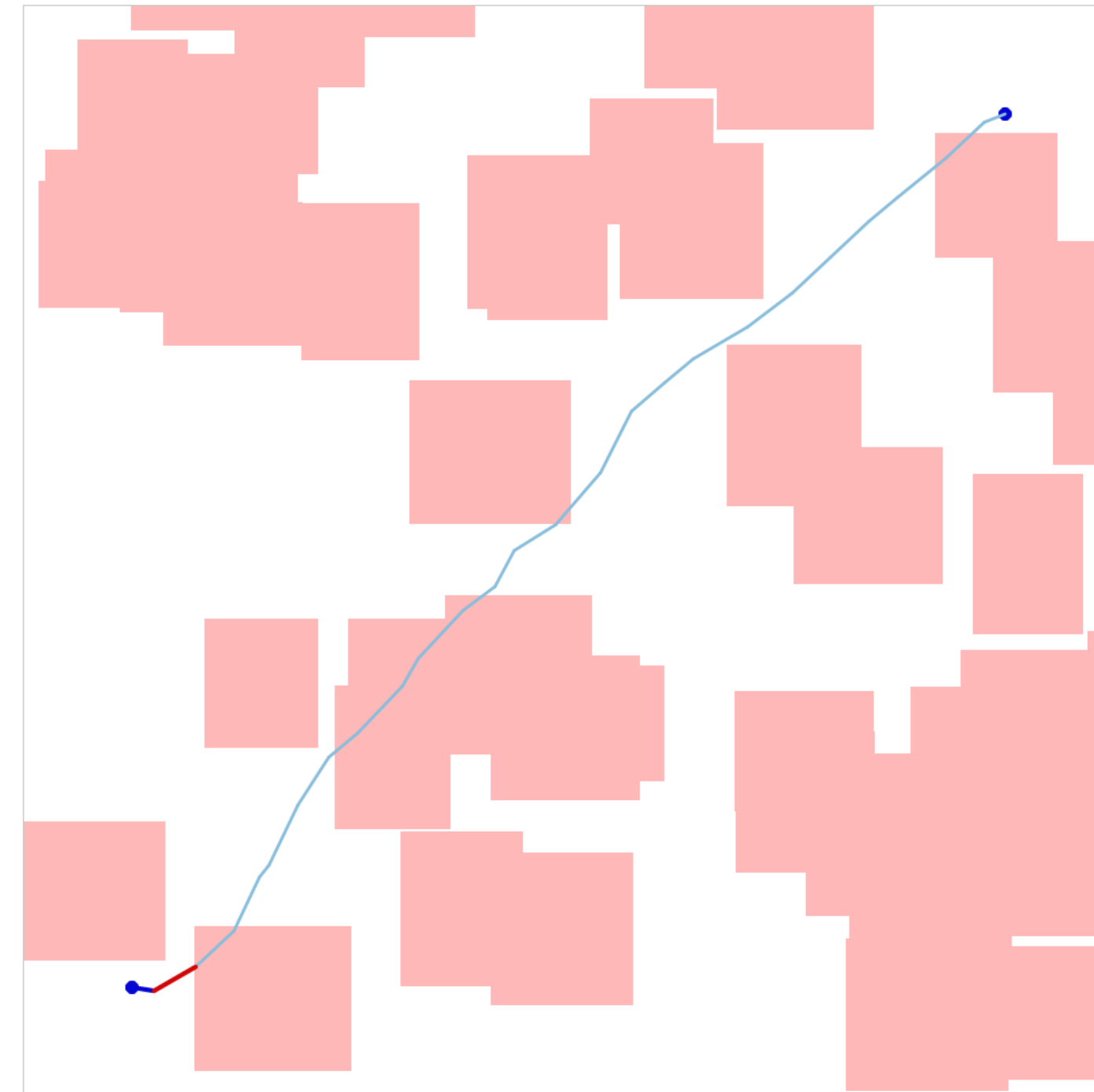
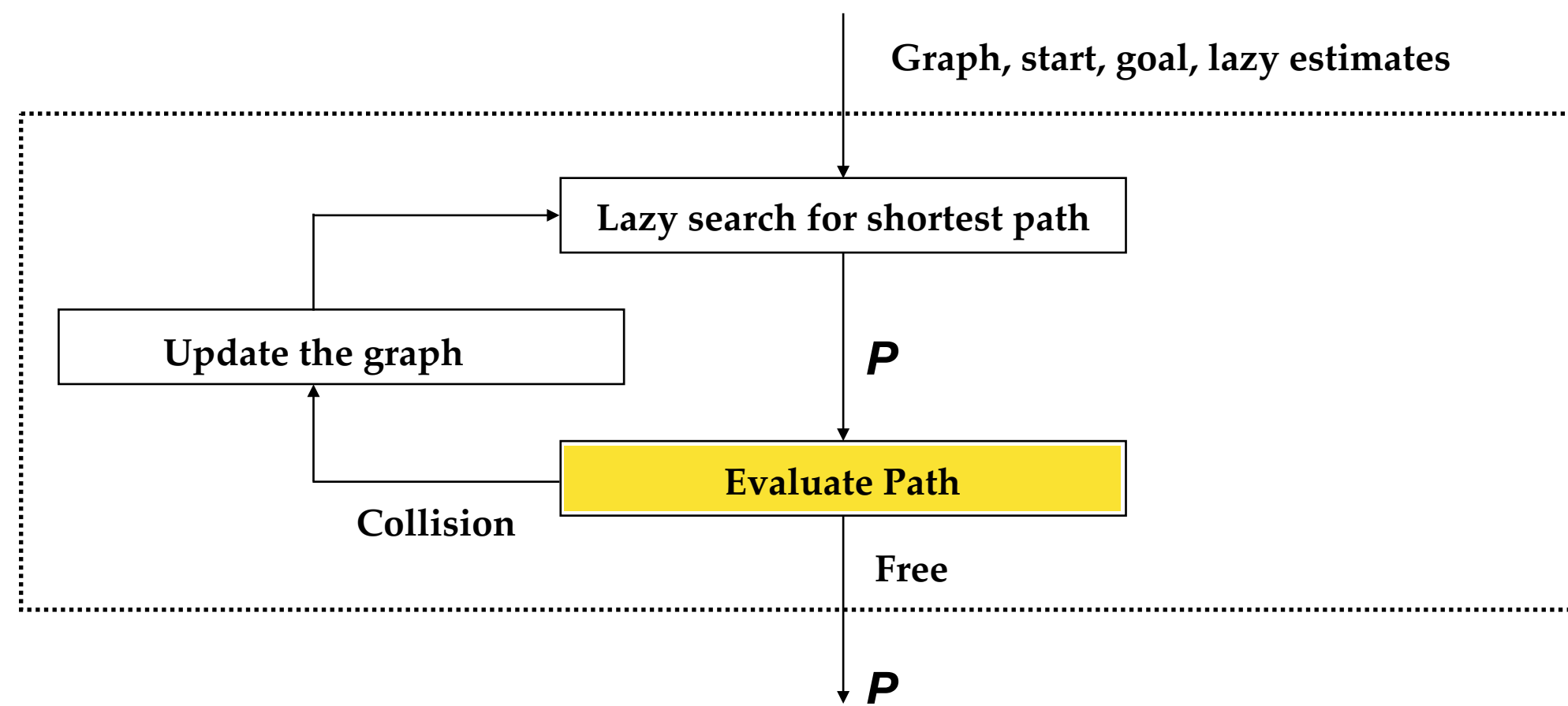
OFU on Steroids!



Only Slime Known Shortest Paths

# LazySP

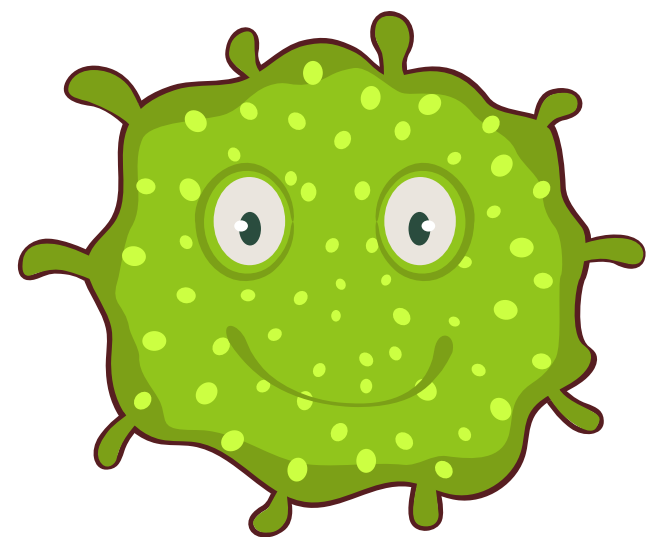
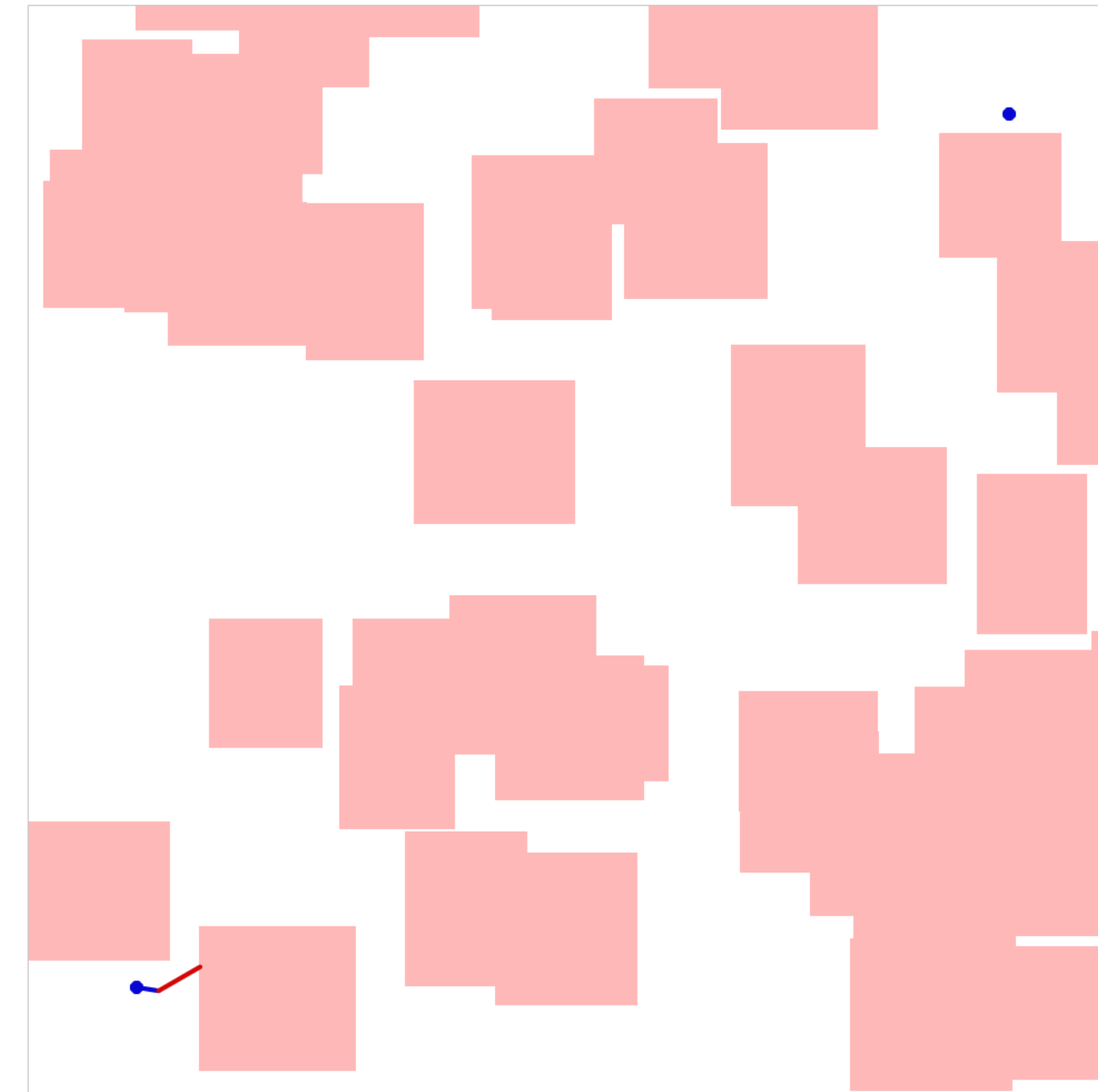
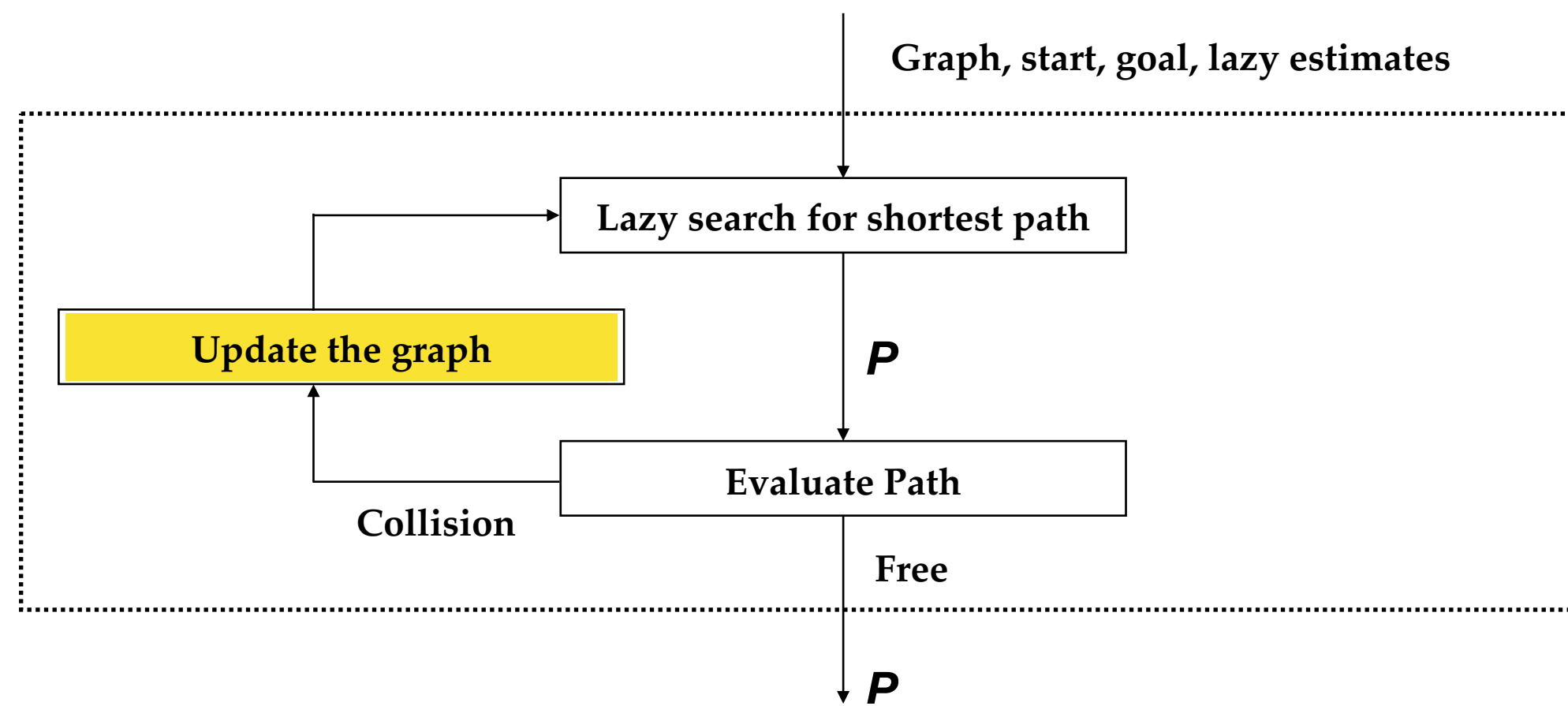
OFU on Steroids!



Only Slime Known Shortest Paths

# LazySP

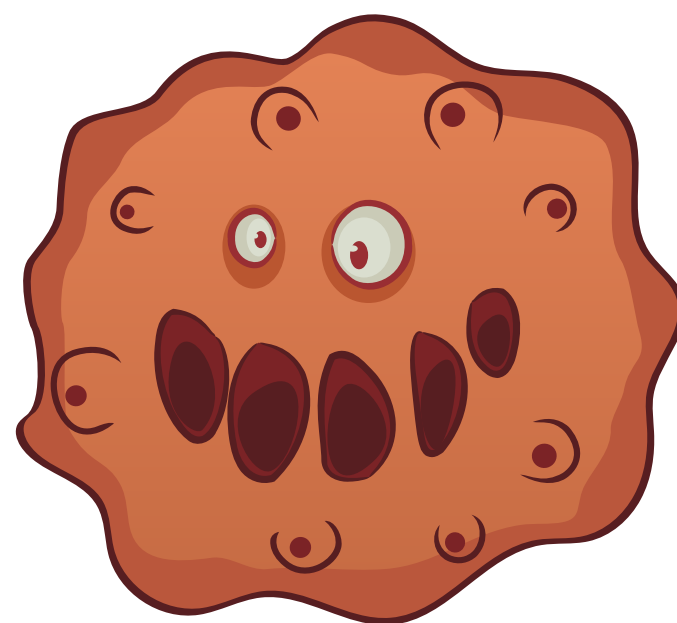
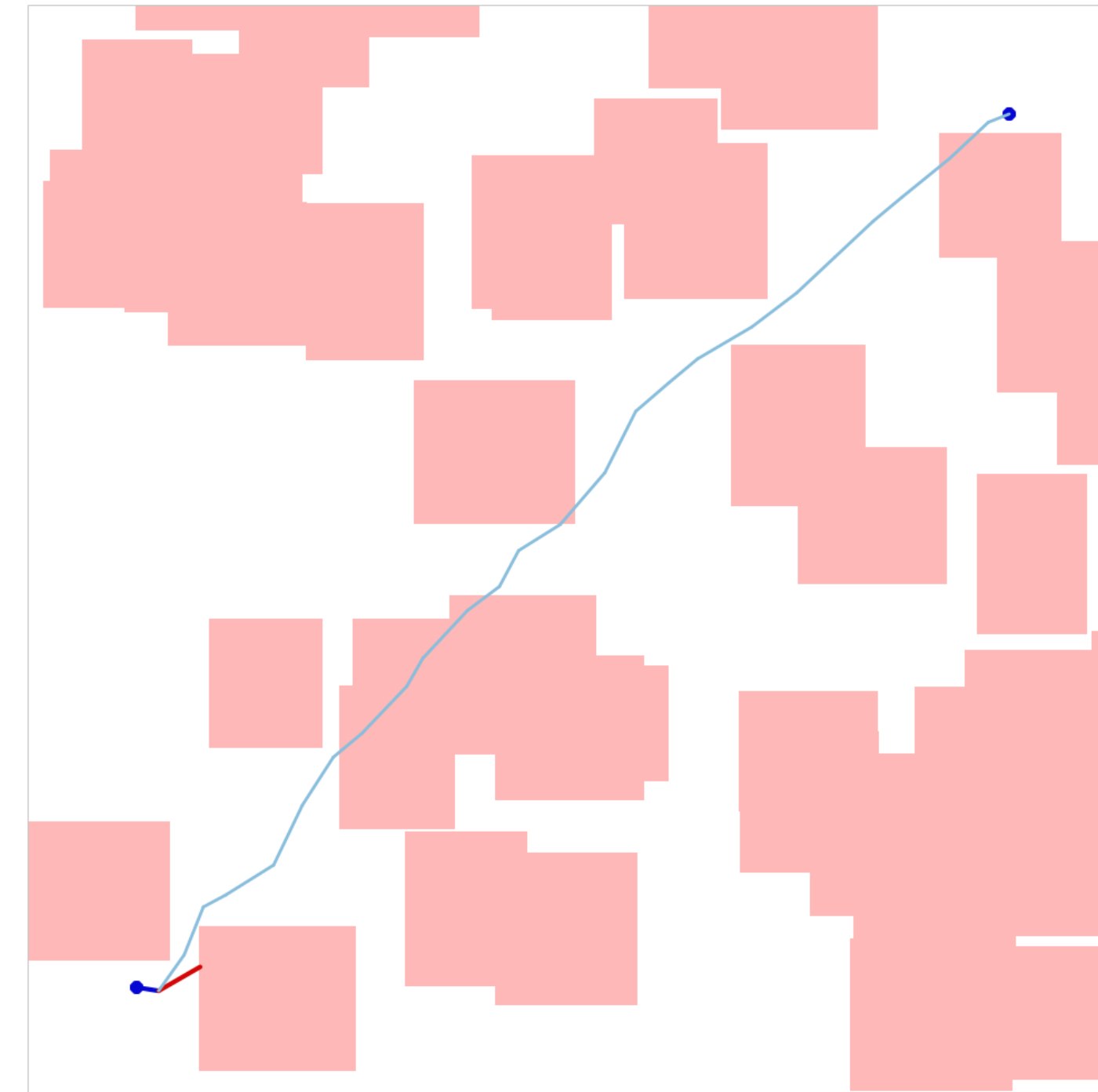
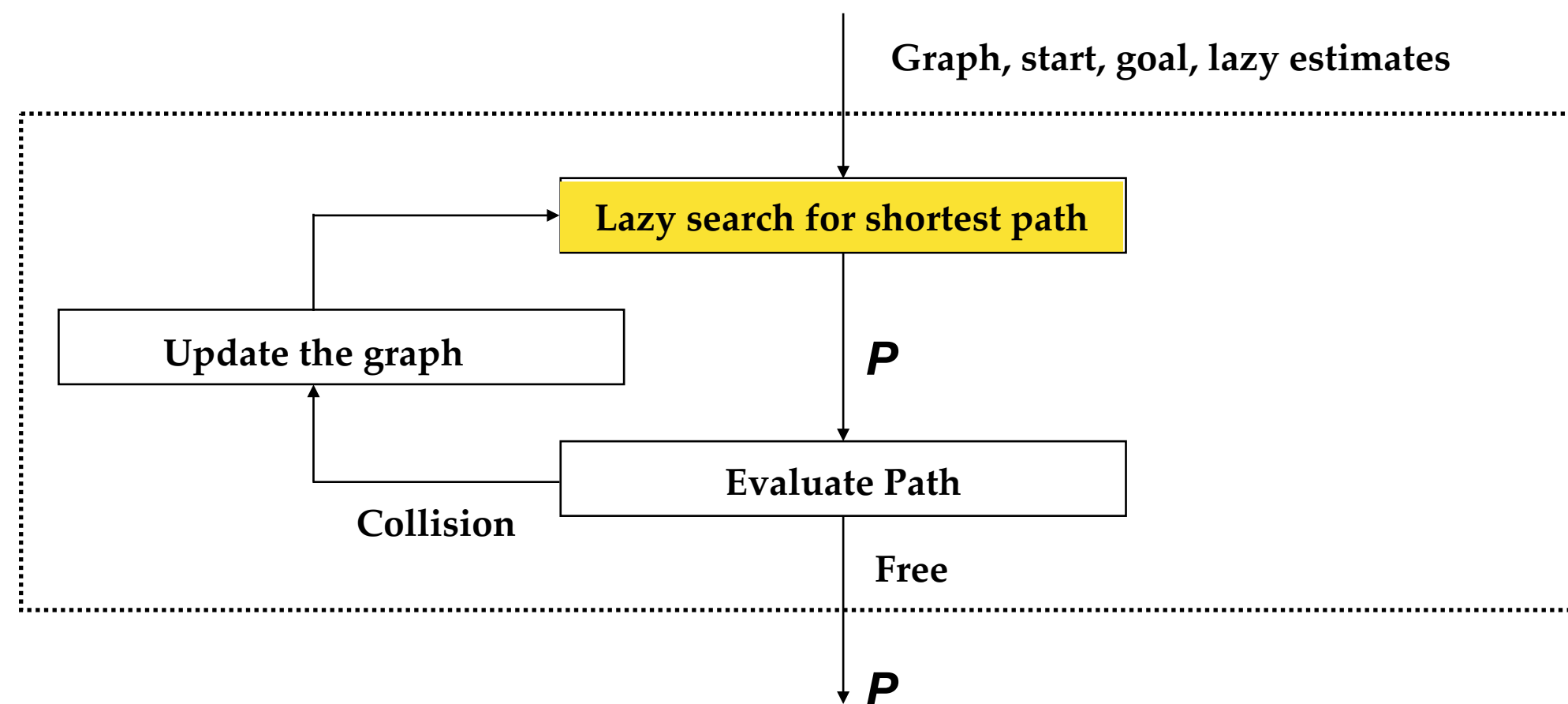
OFU on Steroids!



Only Slime Known Shortest Paths

# LazySP

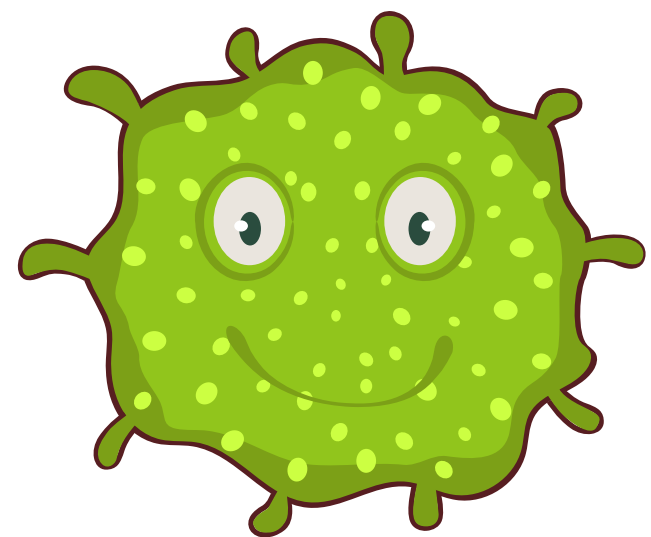
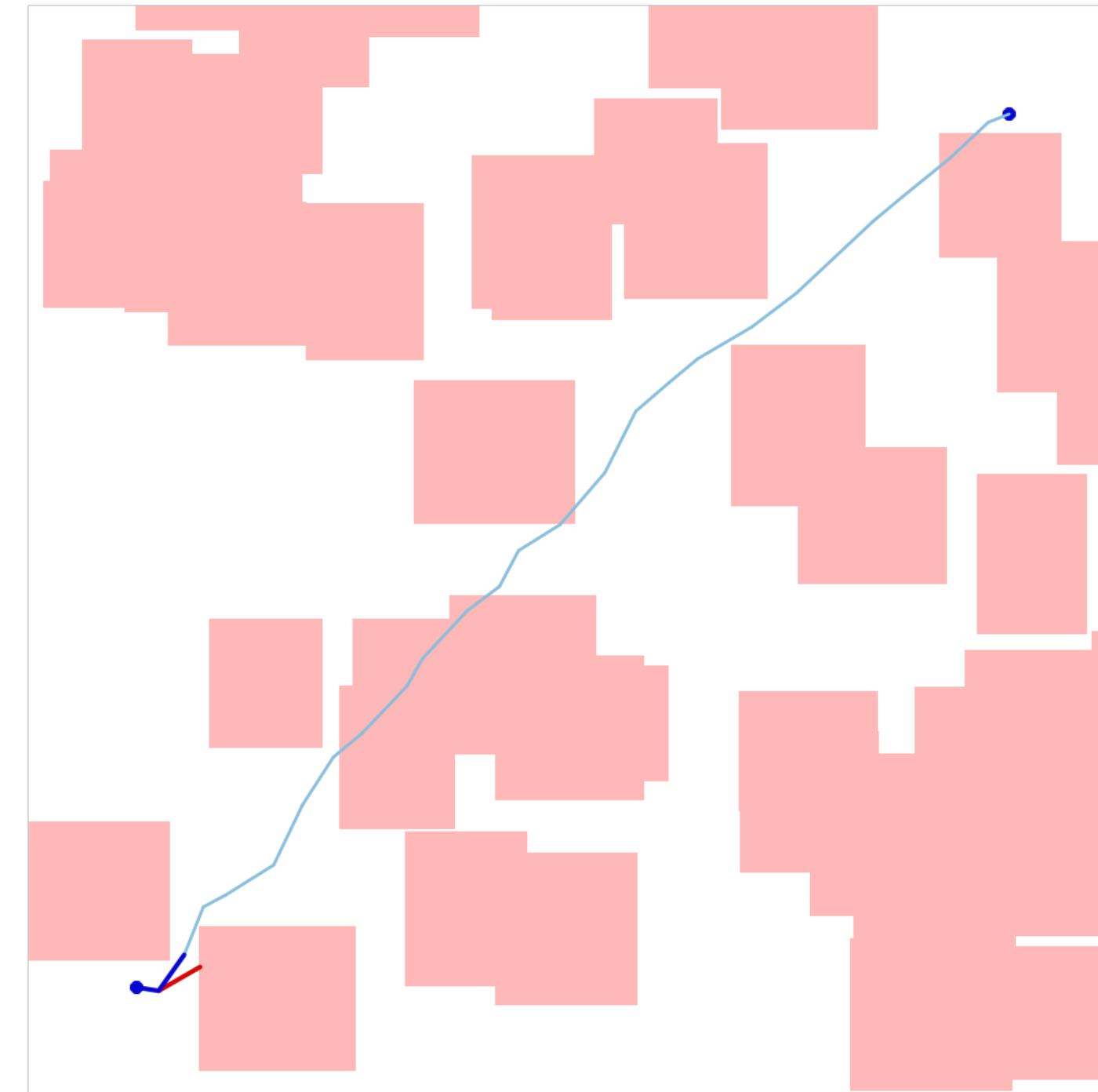
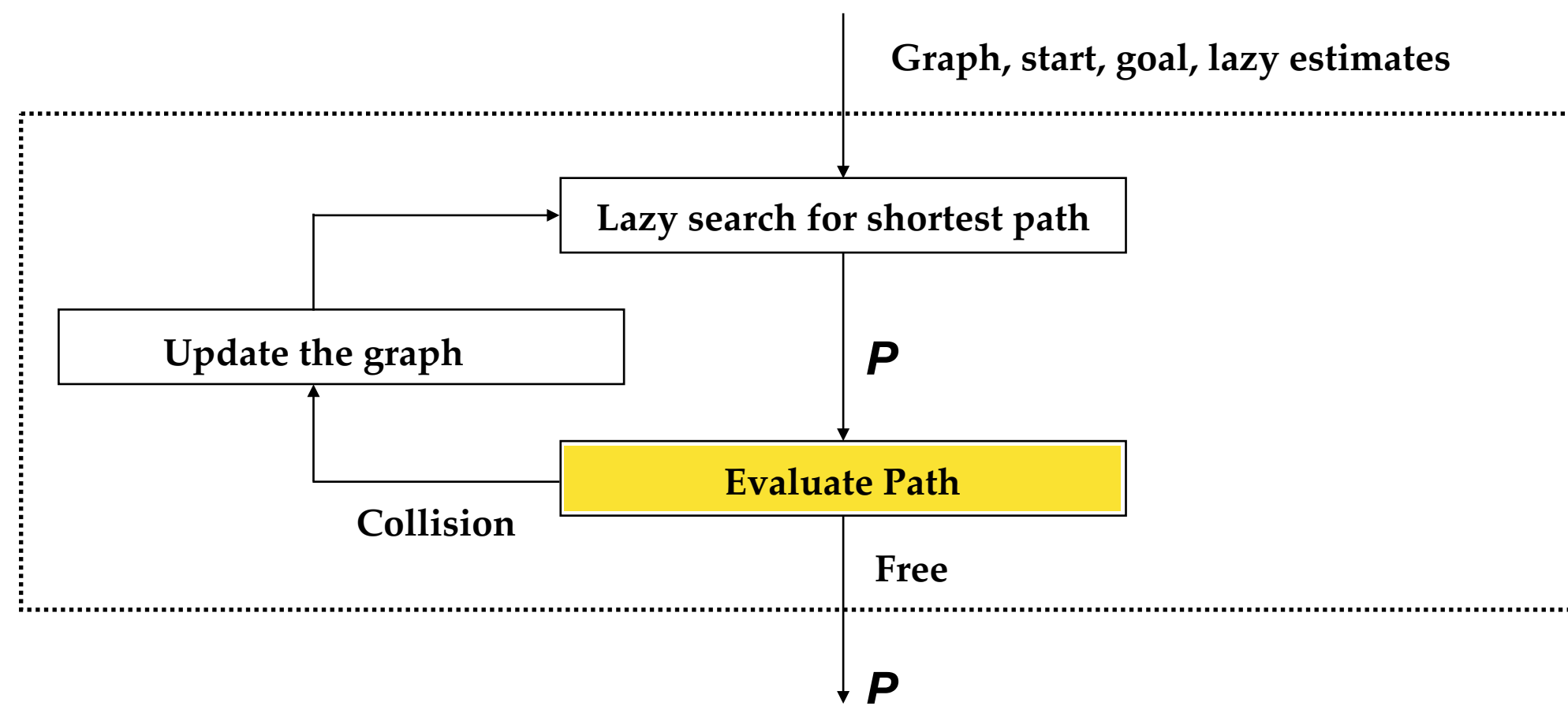
OFU on Steroids!



Send out the Ghost Amoebas

# LazySP

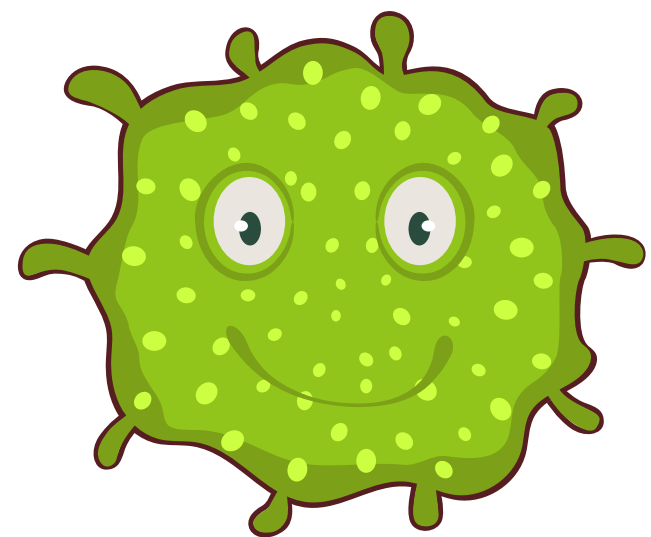
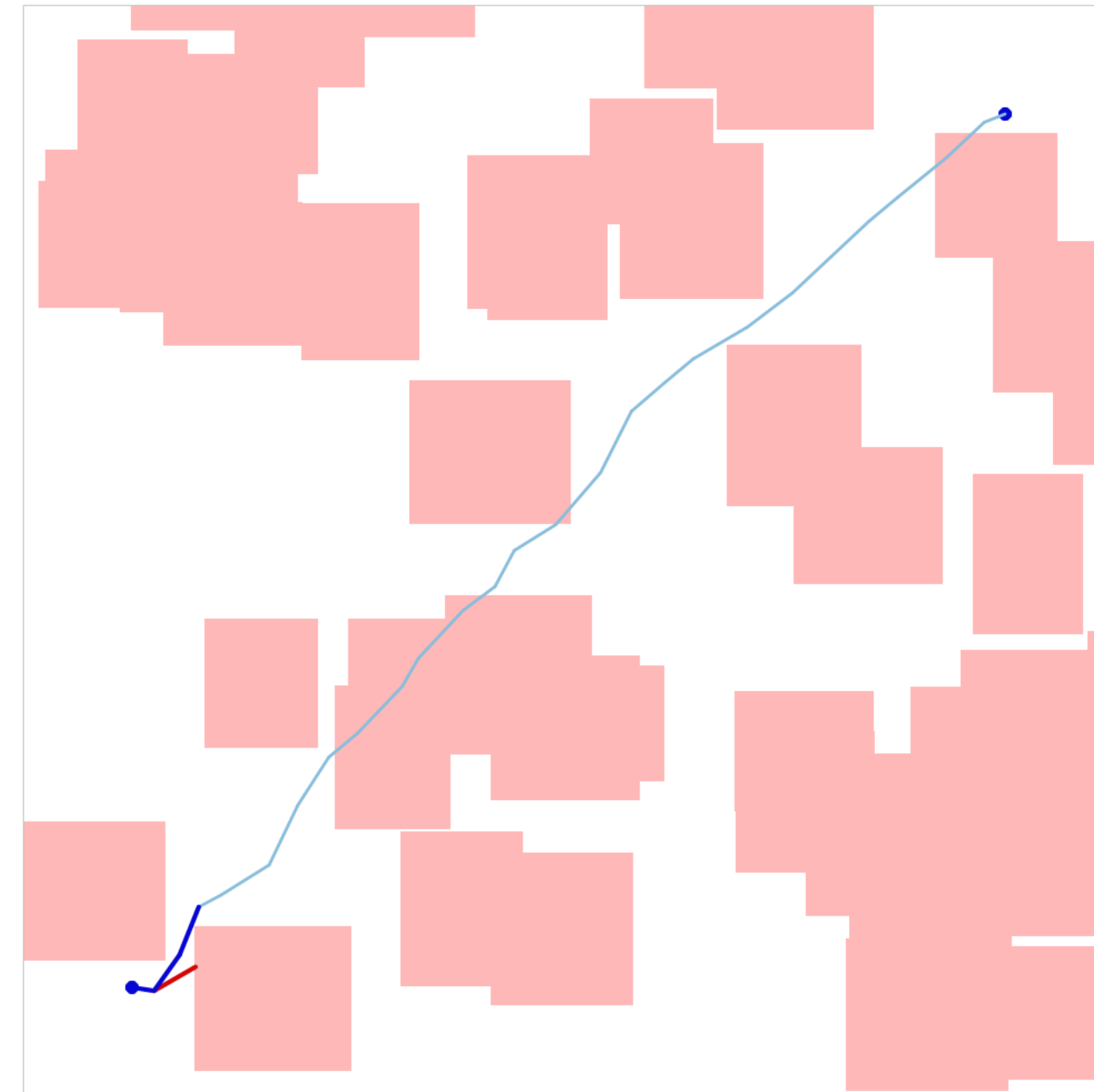
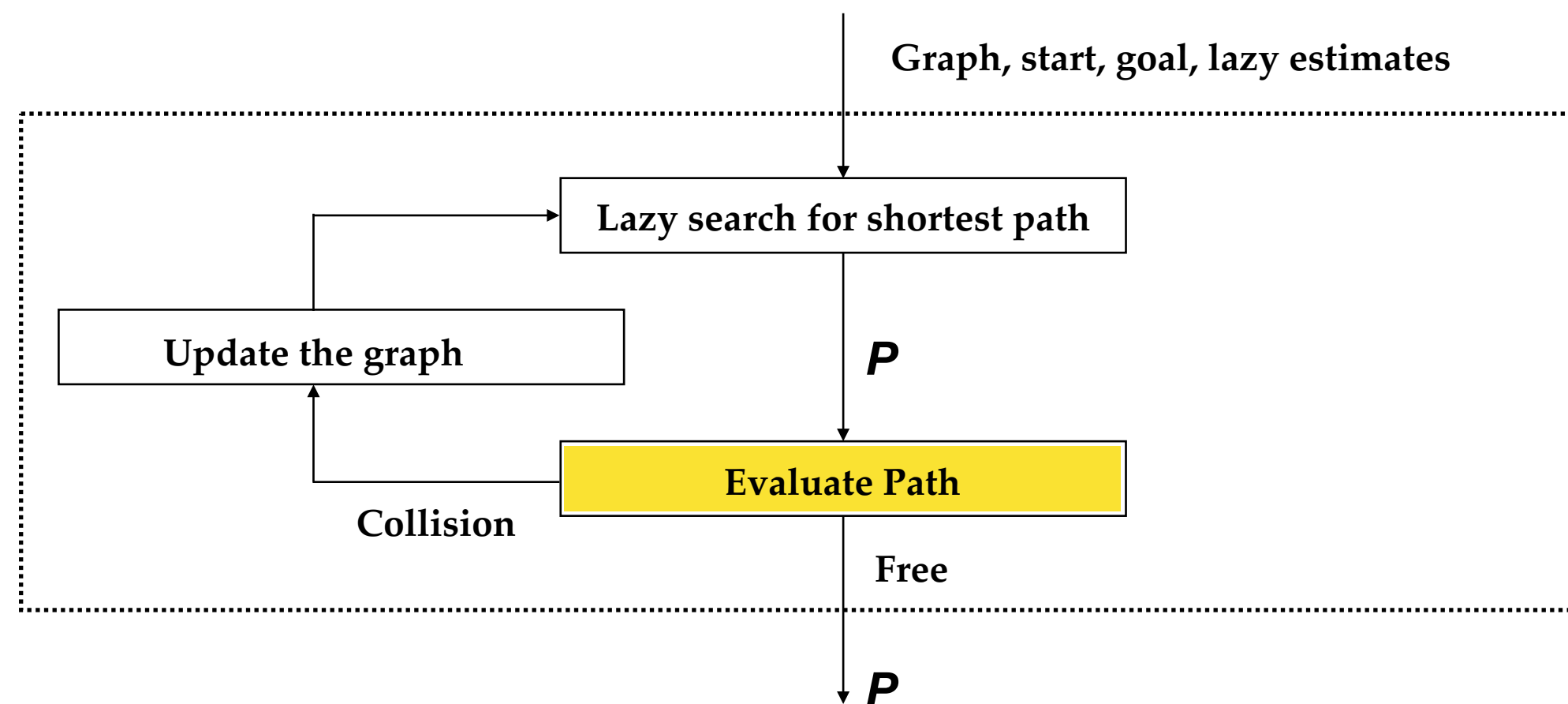
OFU on Steroids!



Only Slime Known Shortest Paths

# LazySP

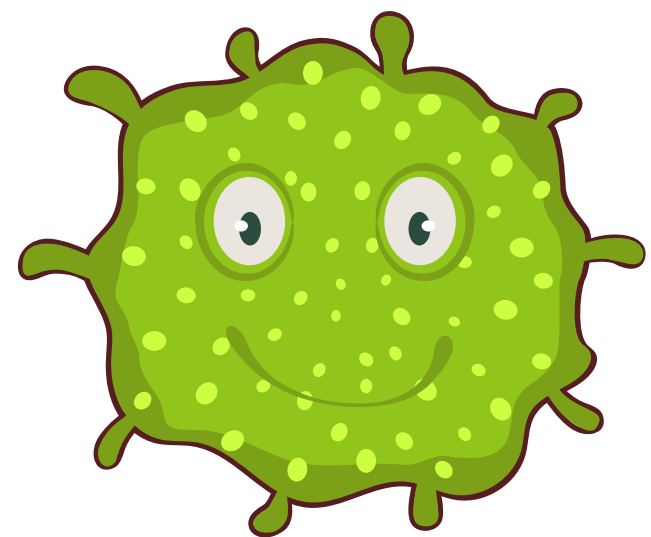
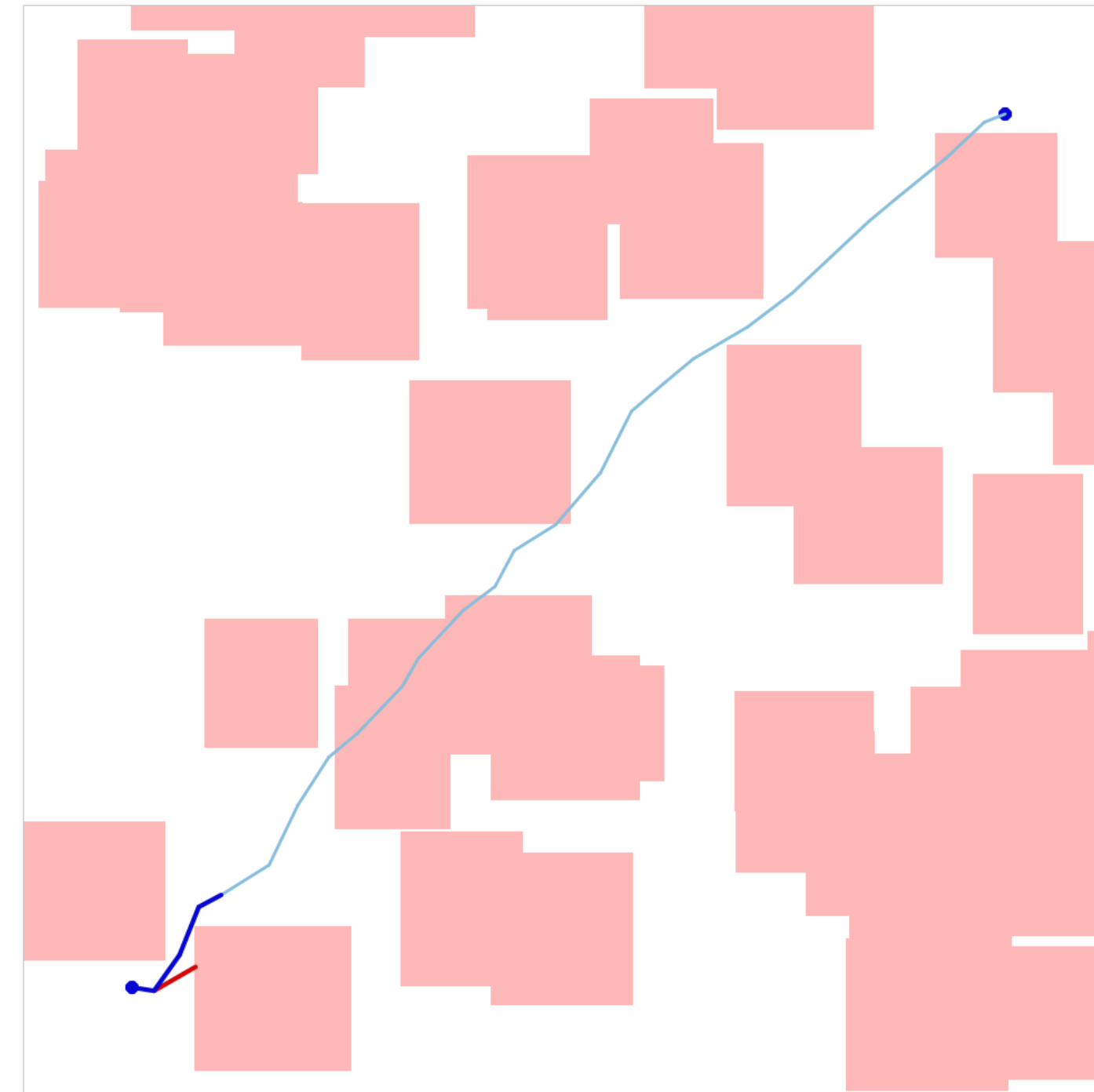
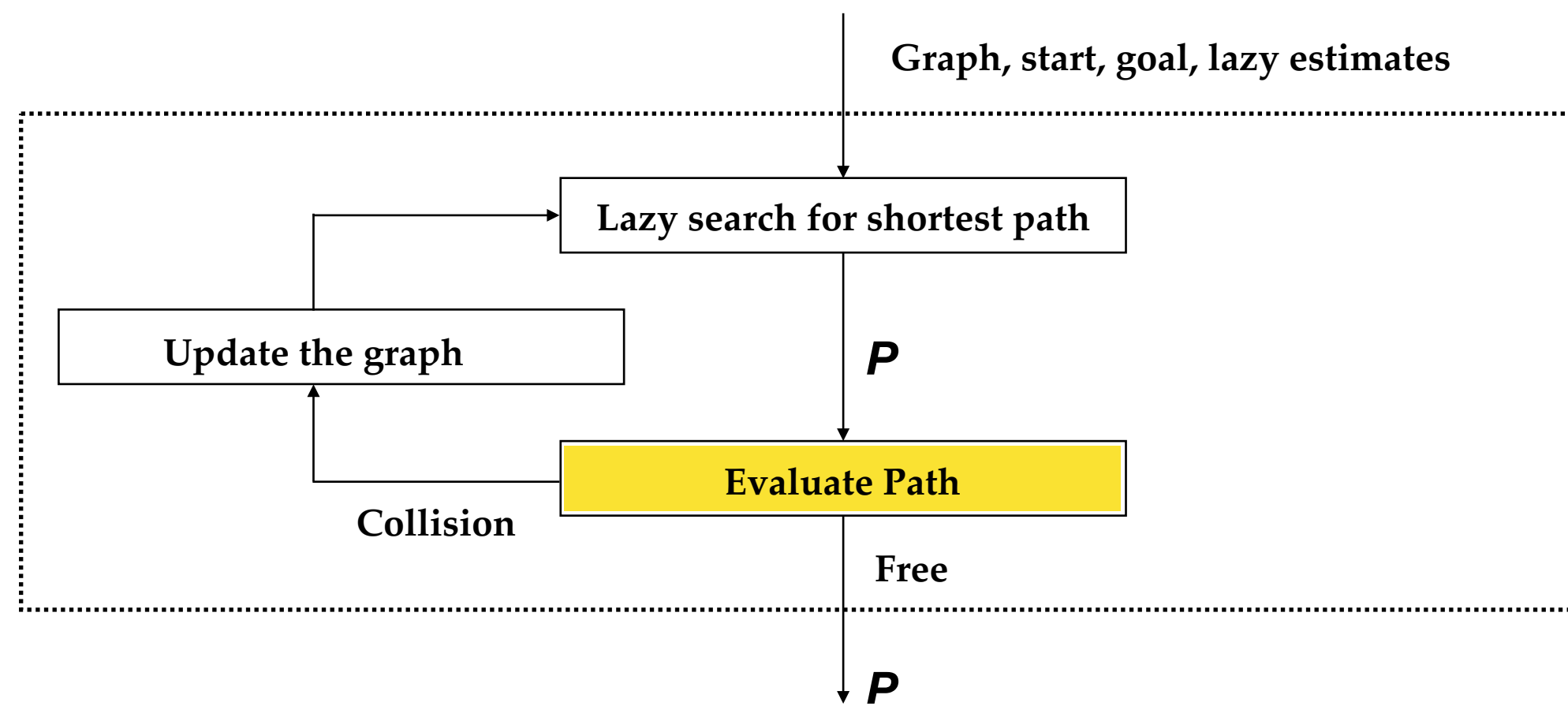
OFU on Steroids!



Only Slime Known Shortest Paths

# LazySP

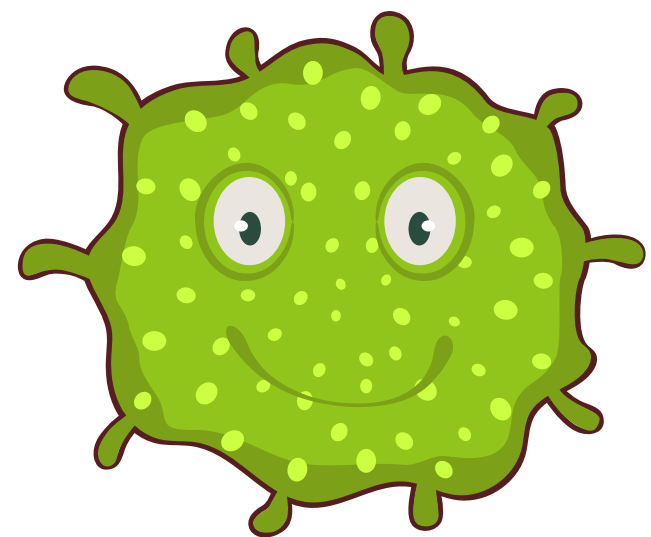
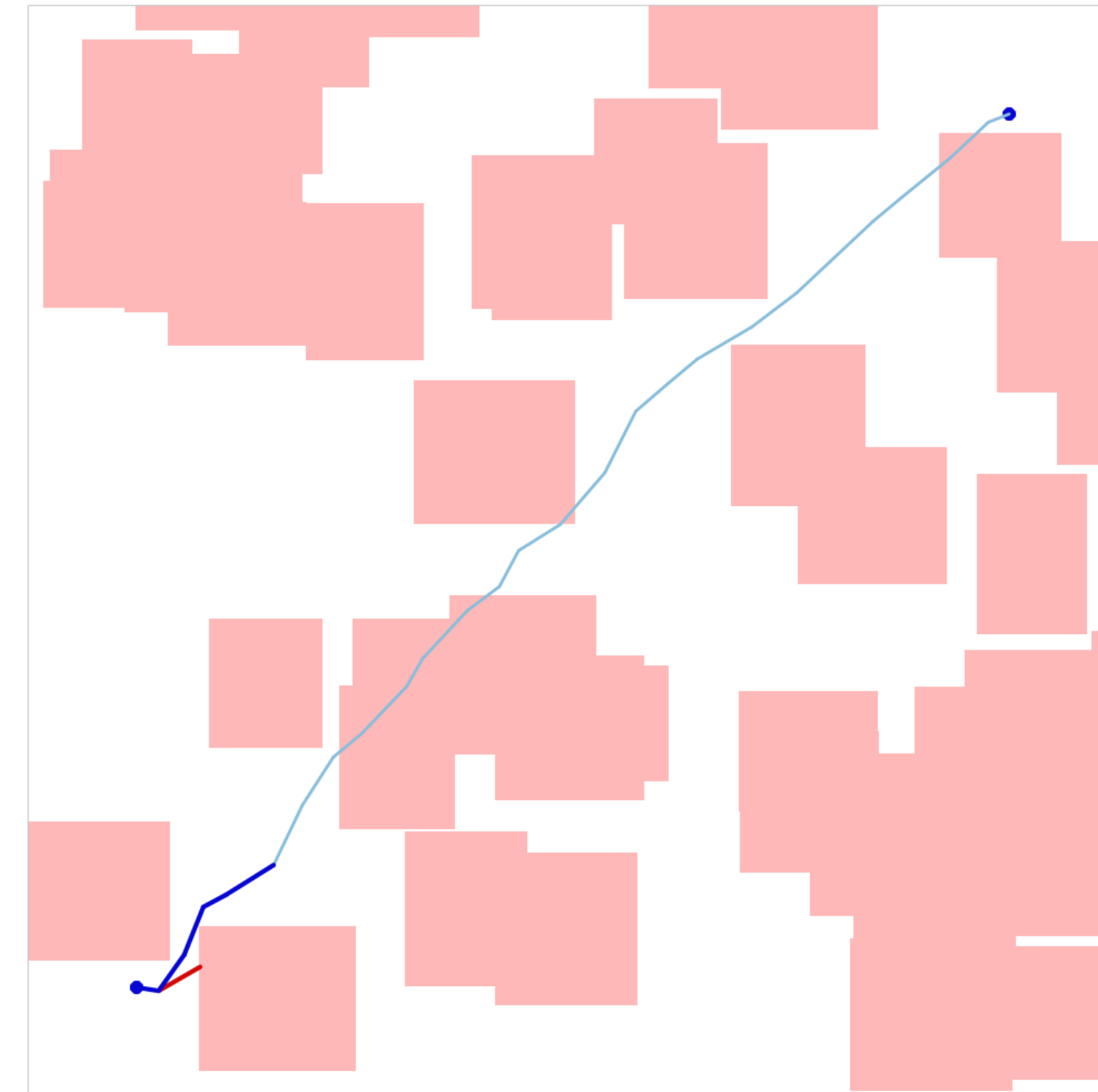
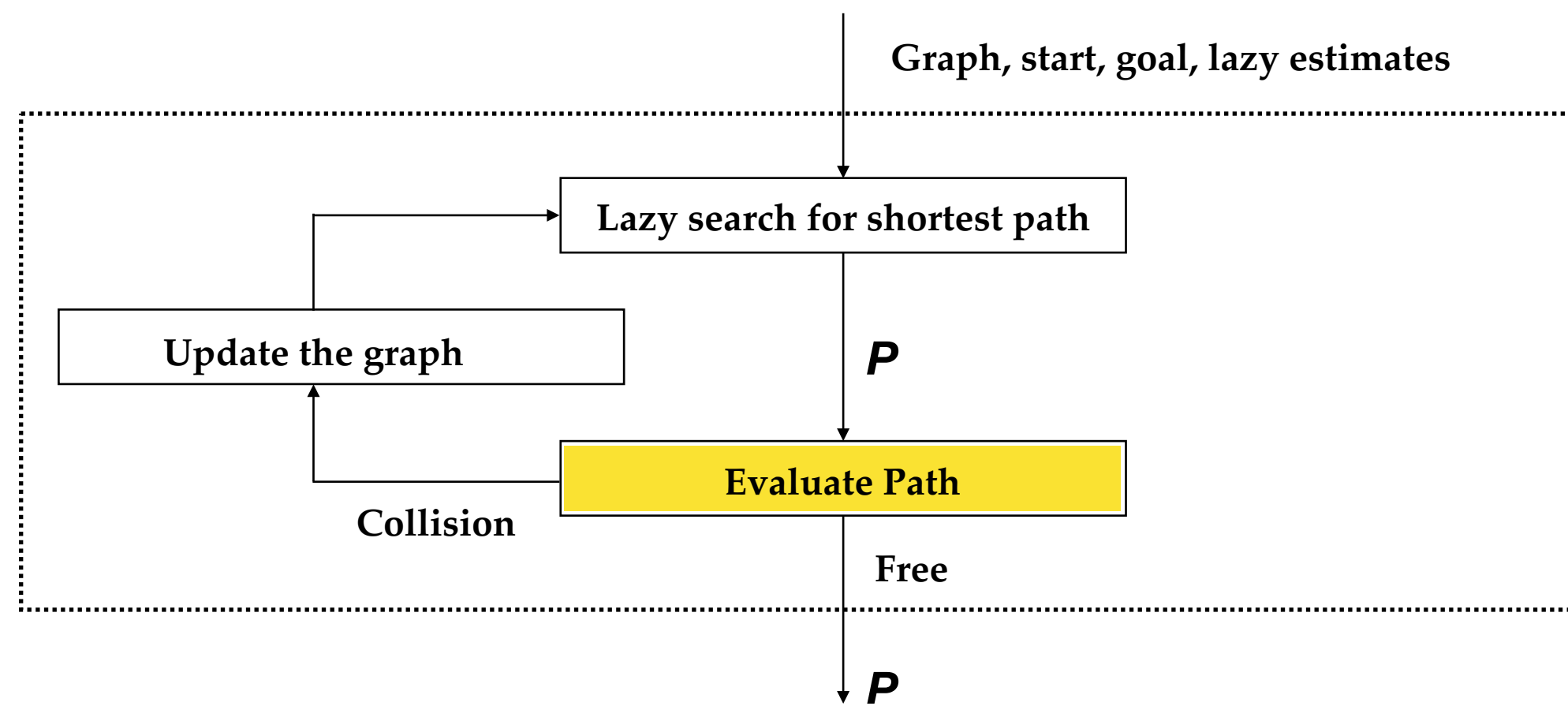
OFU on Steroids!



Only Slime Known Shortest Paths

# LazySP

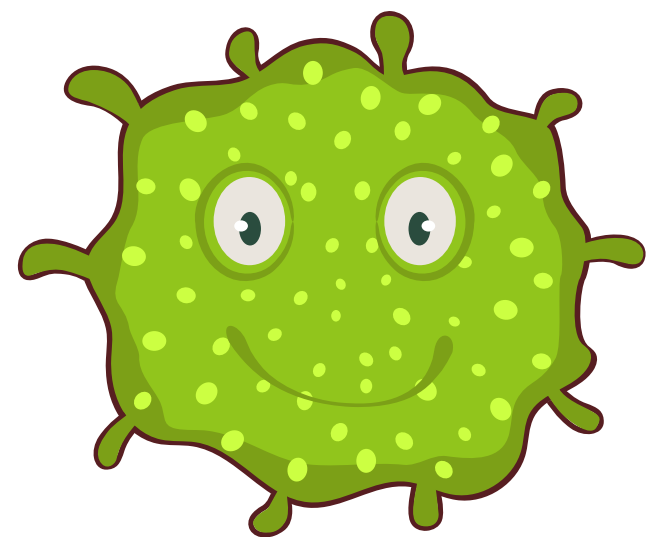
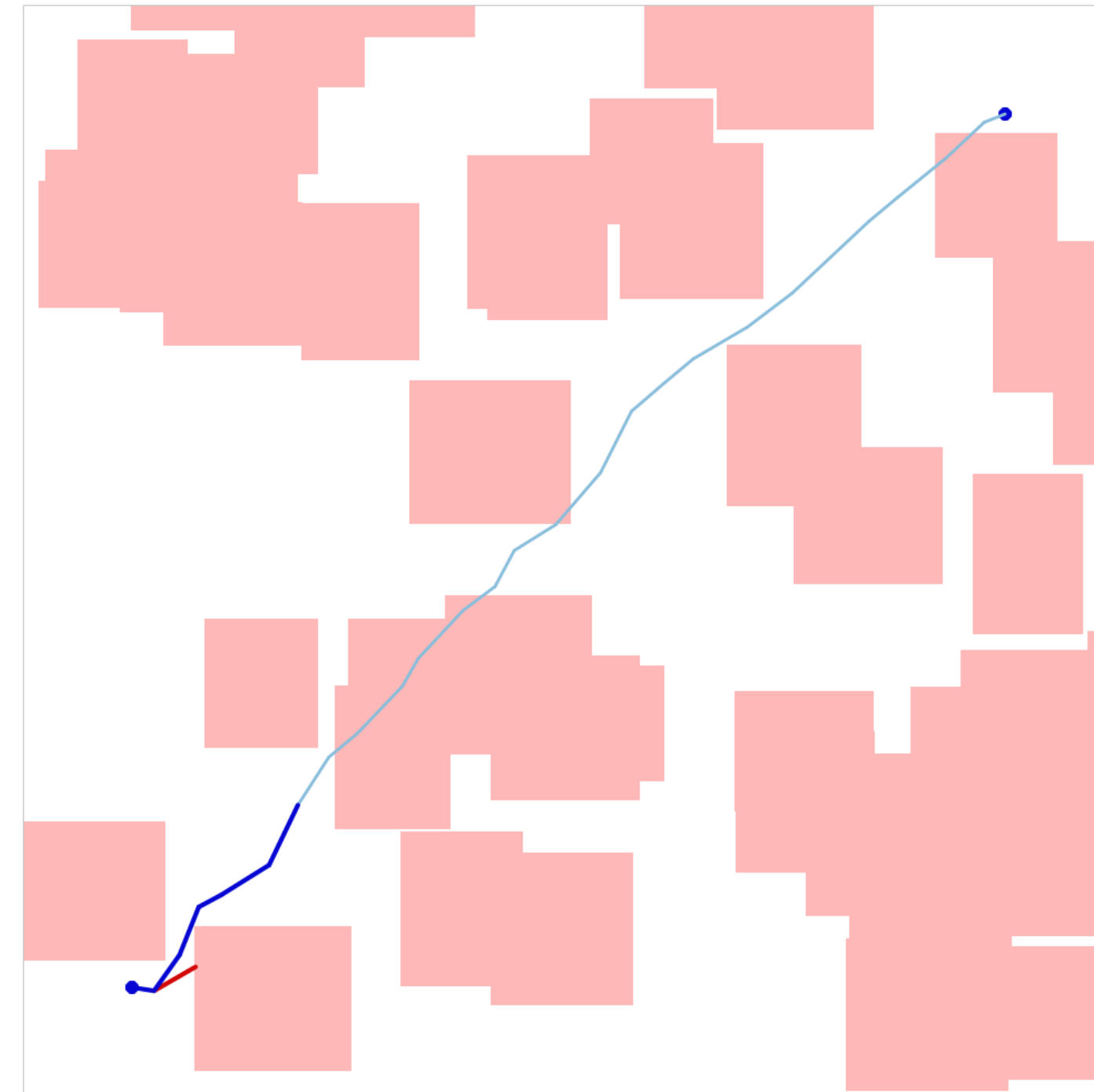
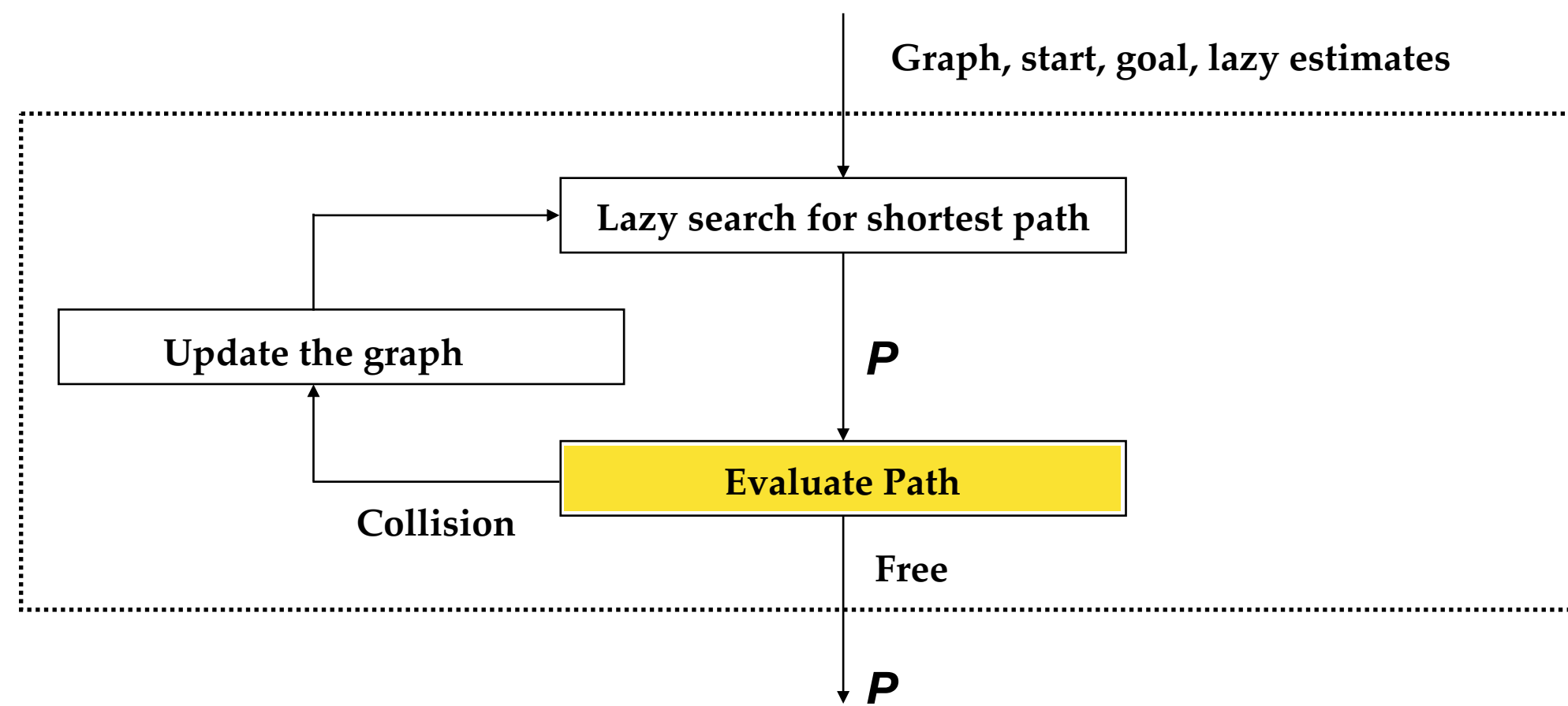
OFU on Steroids!



Only Slime Known Shortest Paths

# LazySP

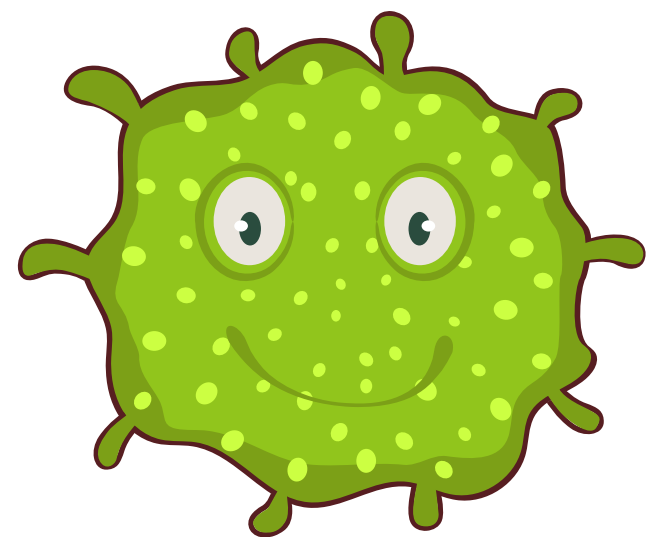
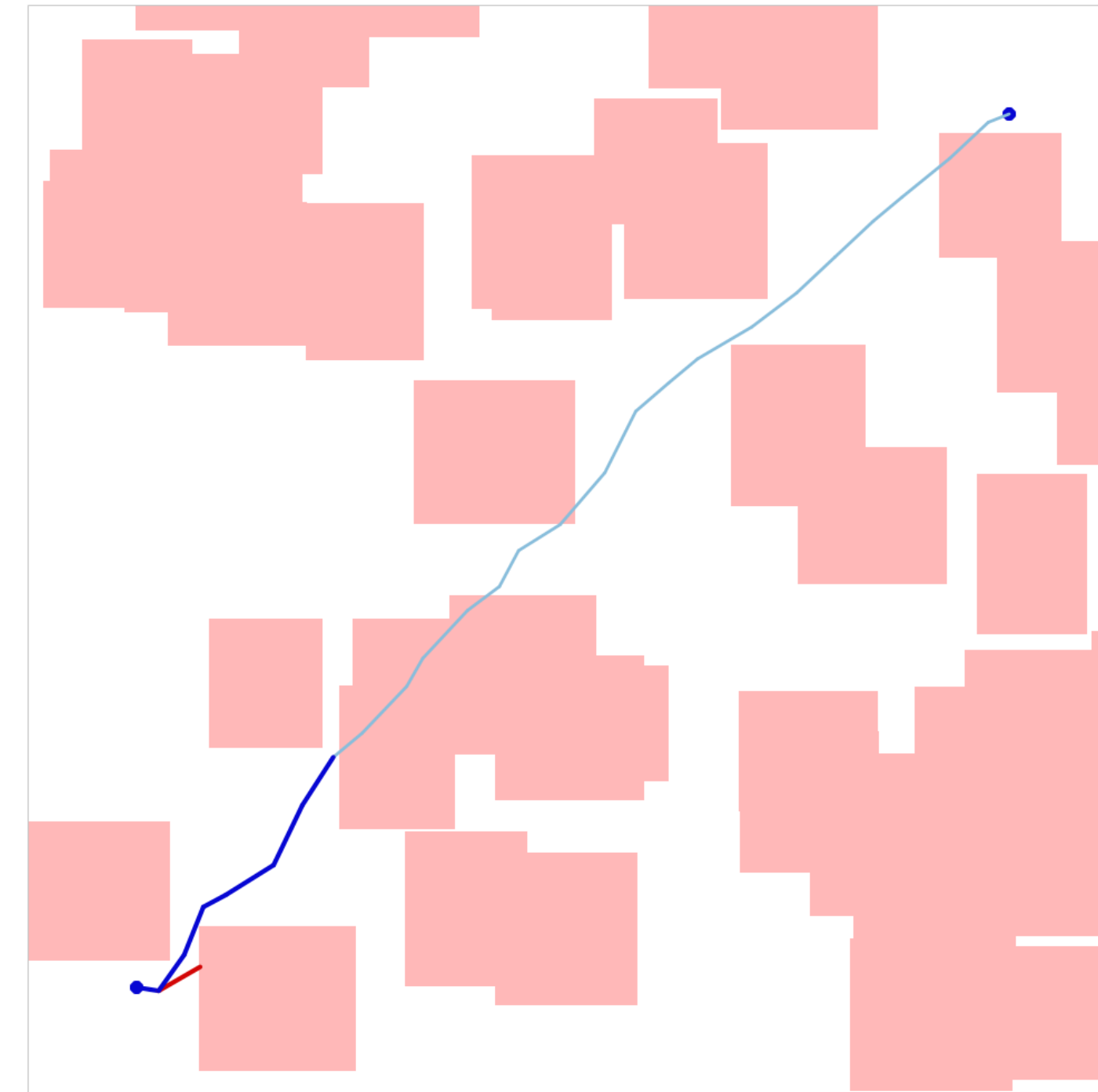
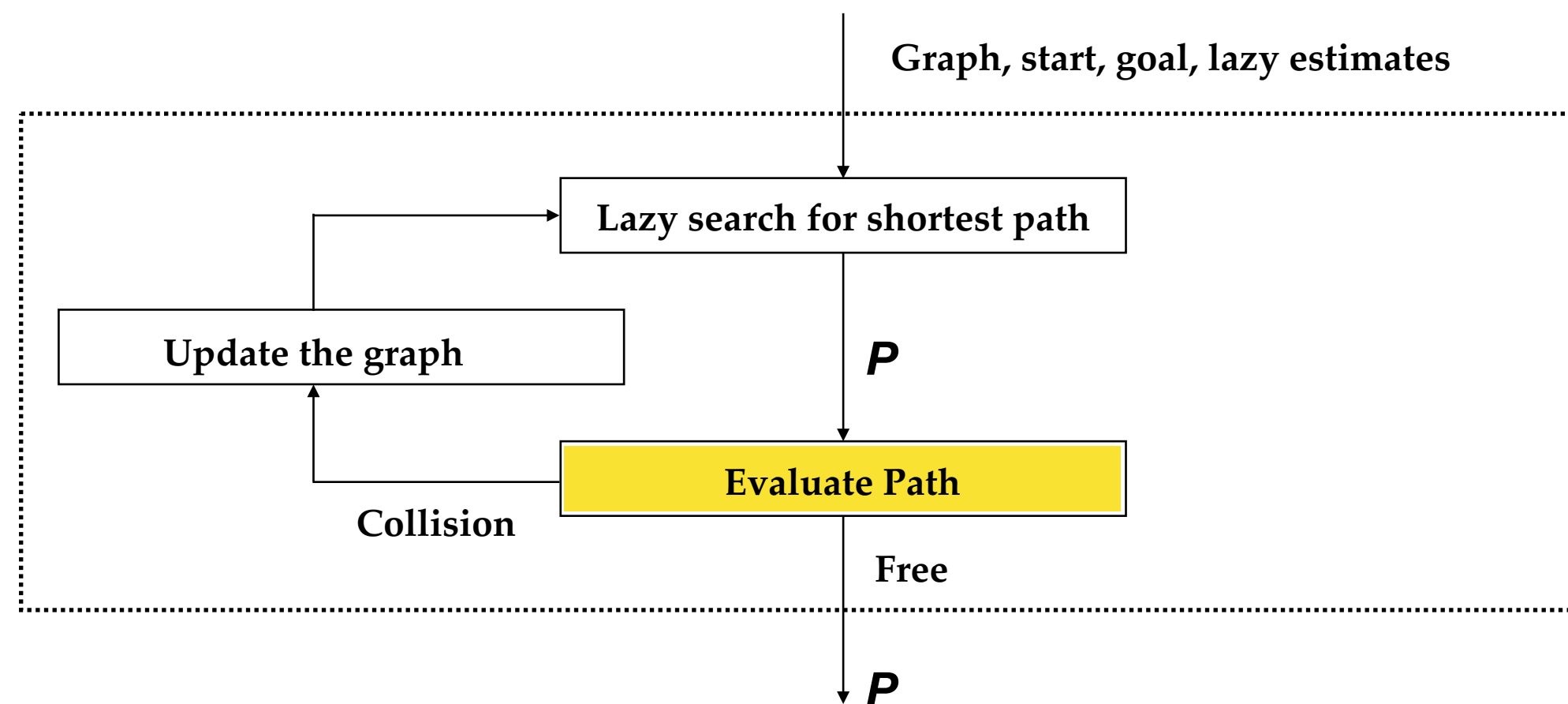
OFU on Steroids!



Only Slime Known Shortest Paths

# LazySP

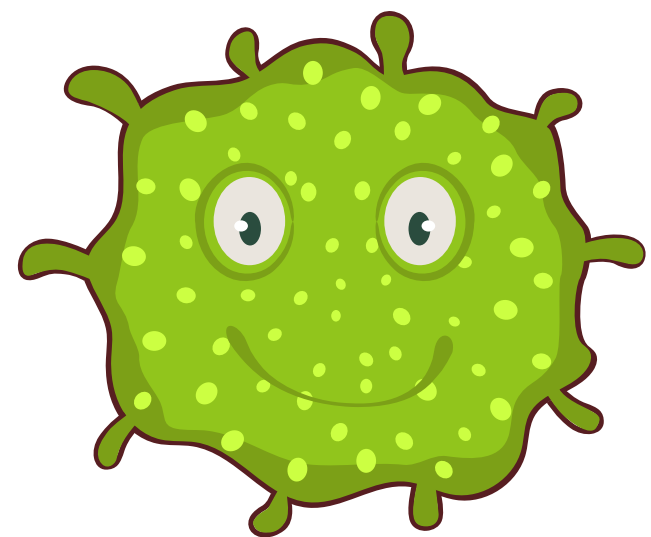
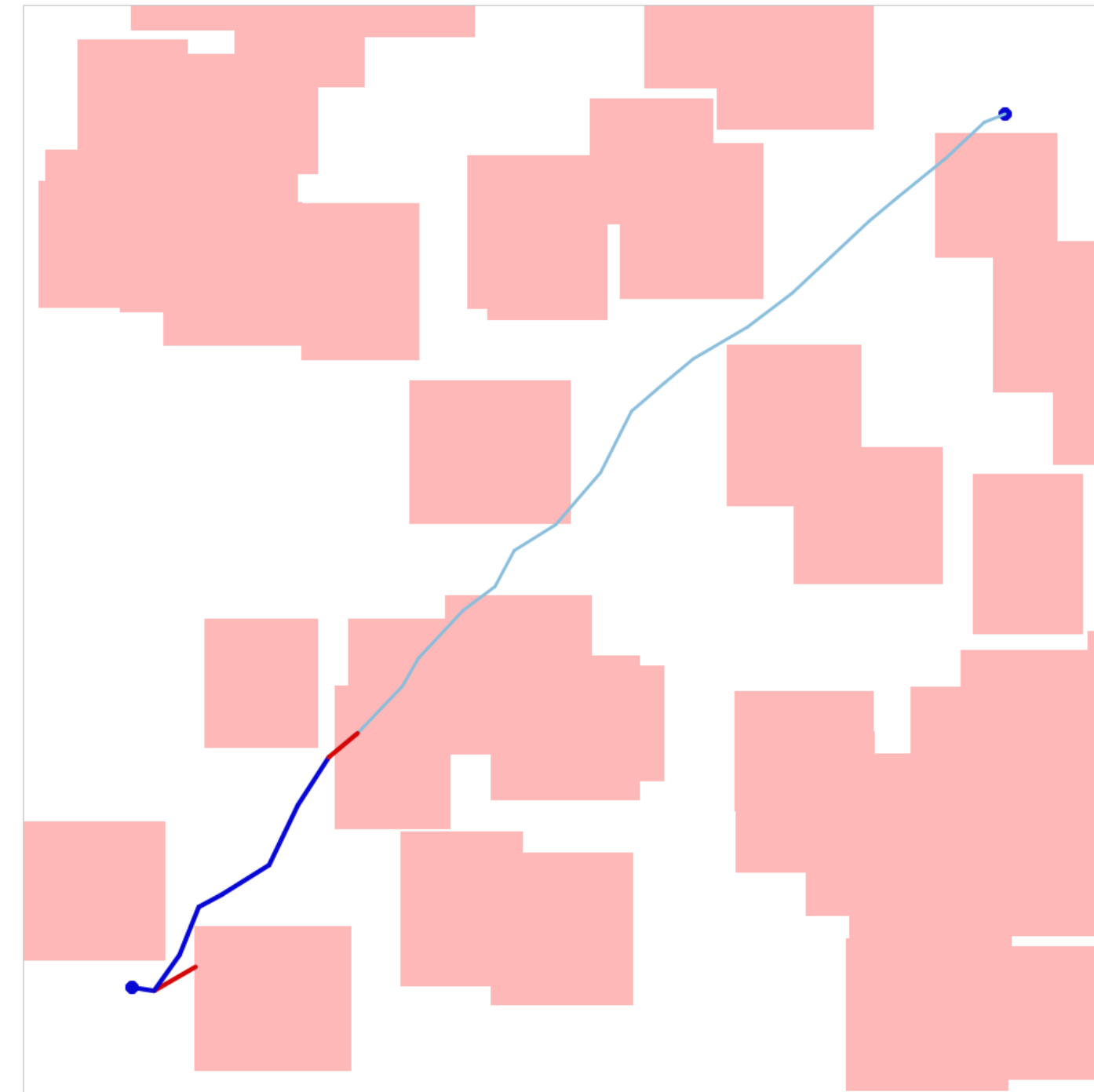
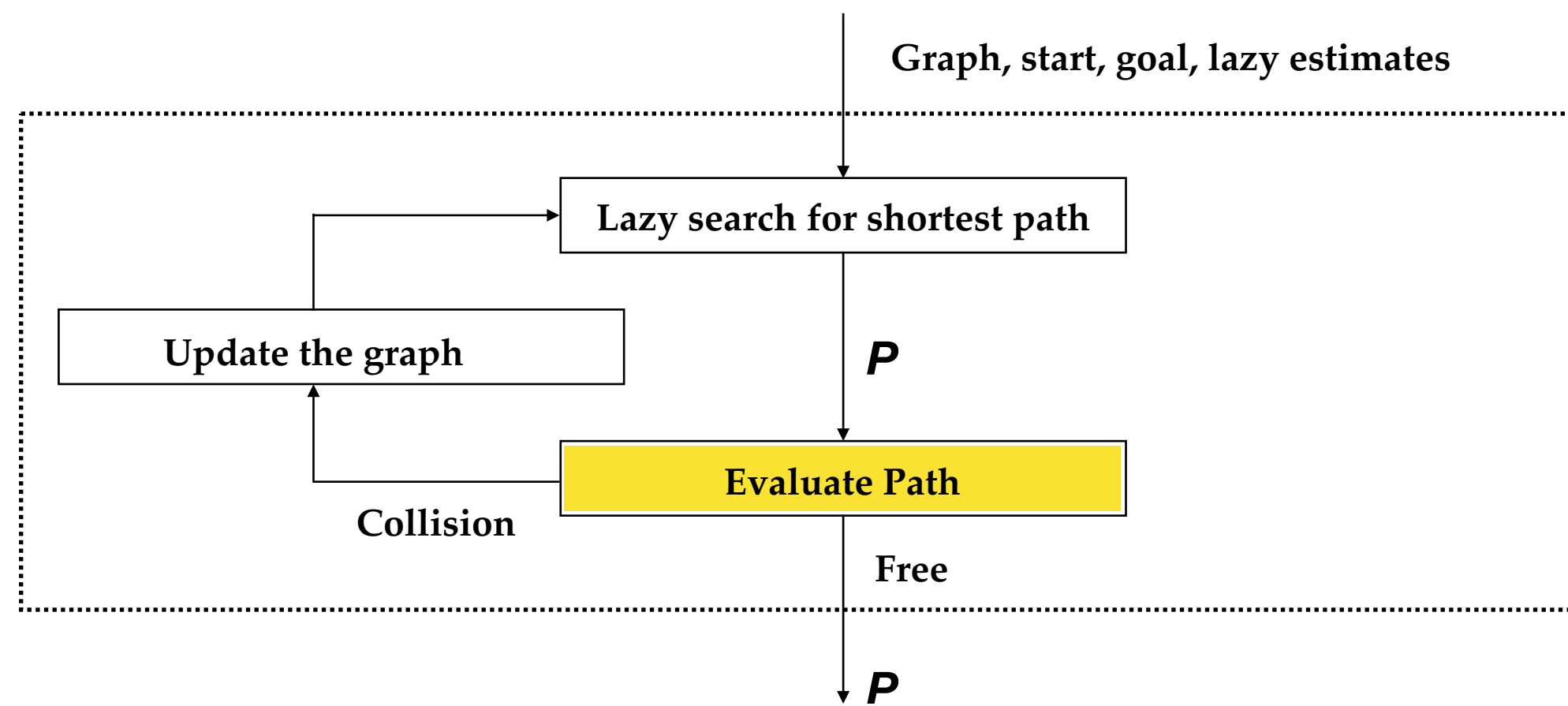
OFU on Steroids!



Only Slime Known Shortest Paths

# LazySP

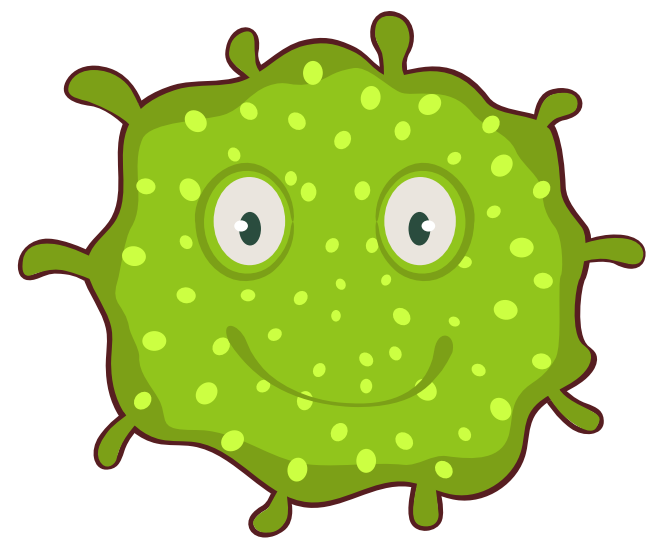
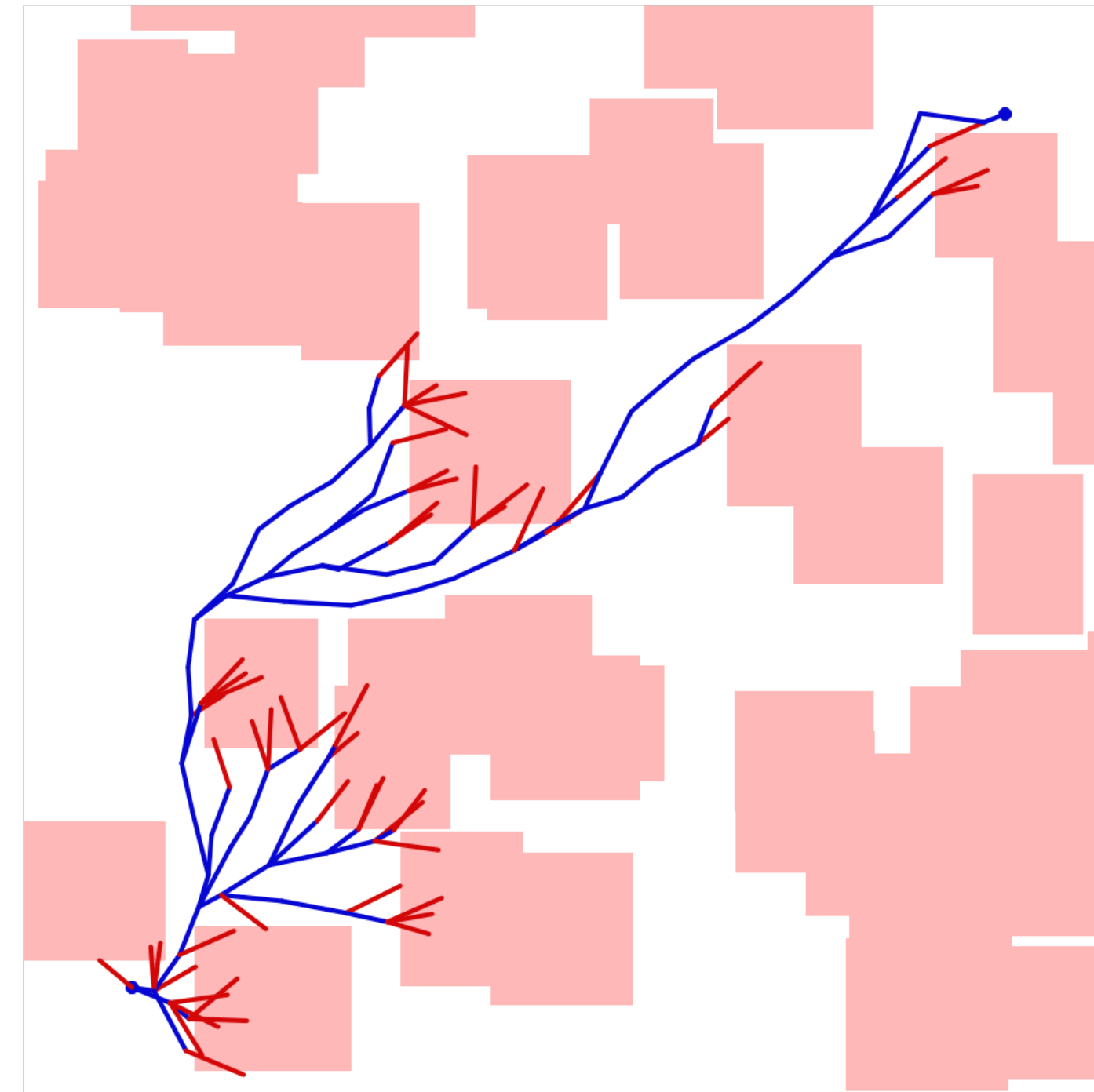
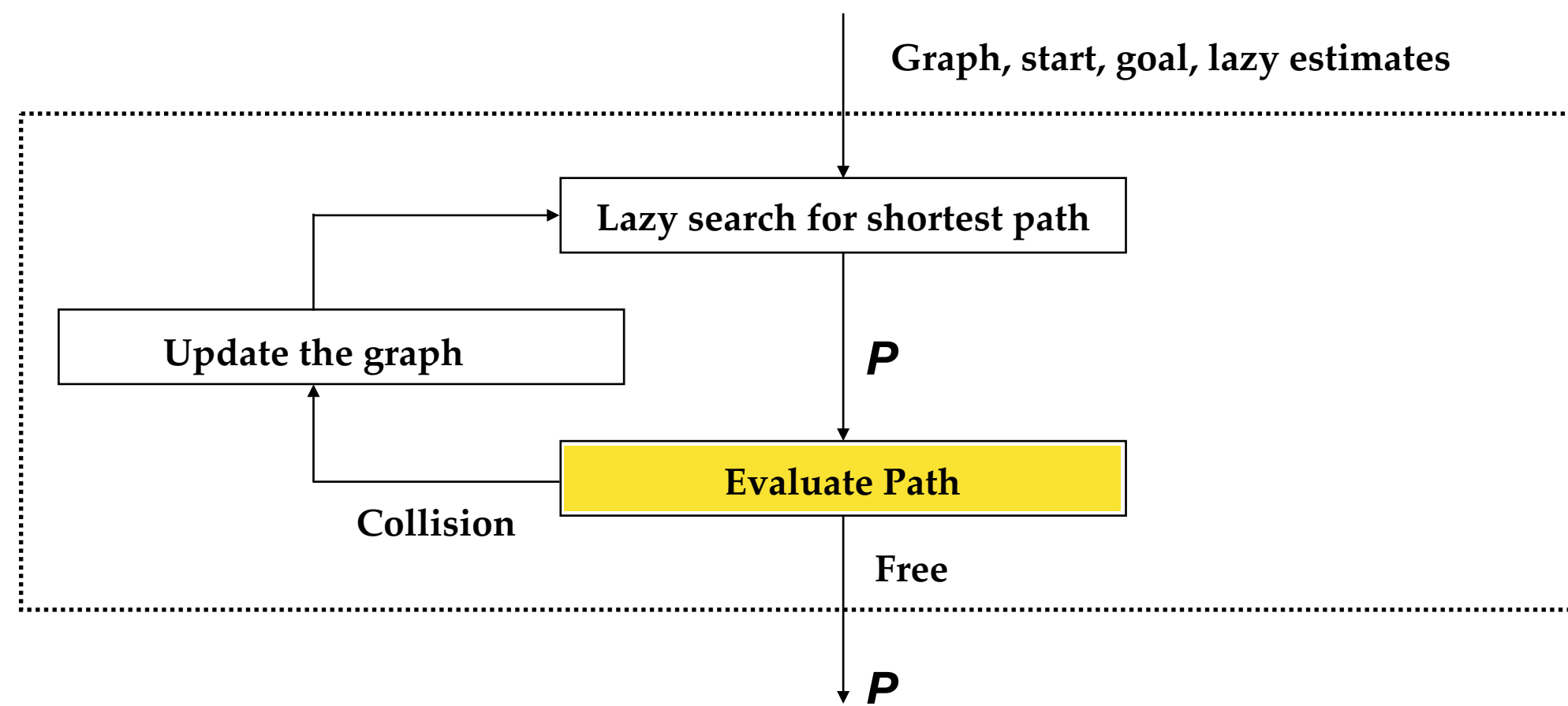
OFU on Steroids!



Only Slime Known Shortest Paths

# LazySP

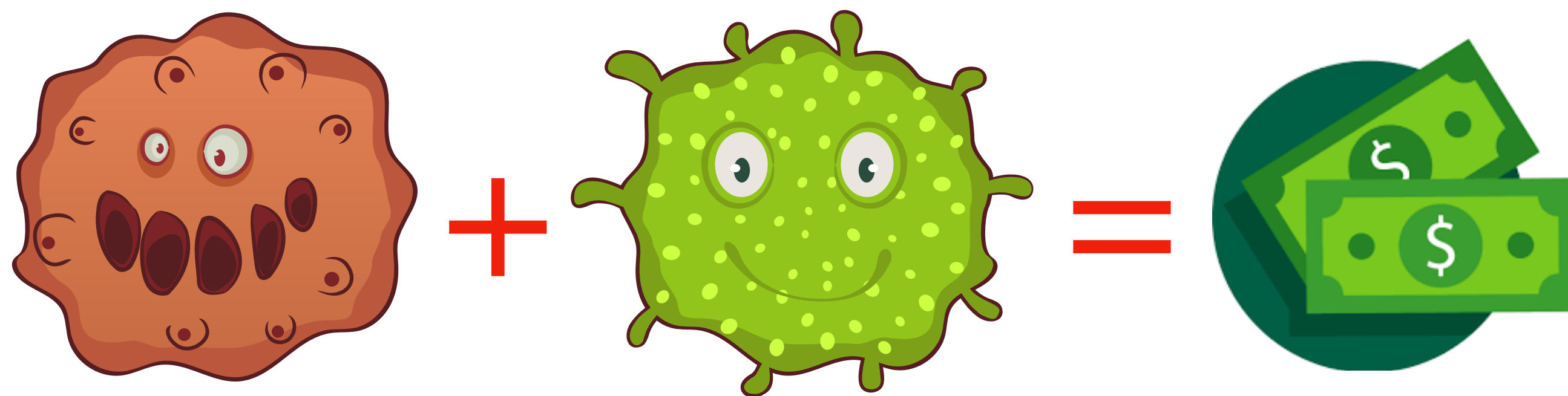
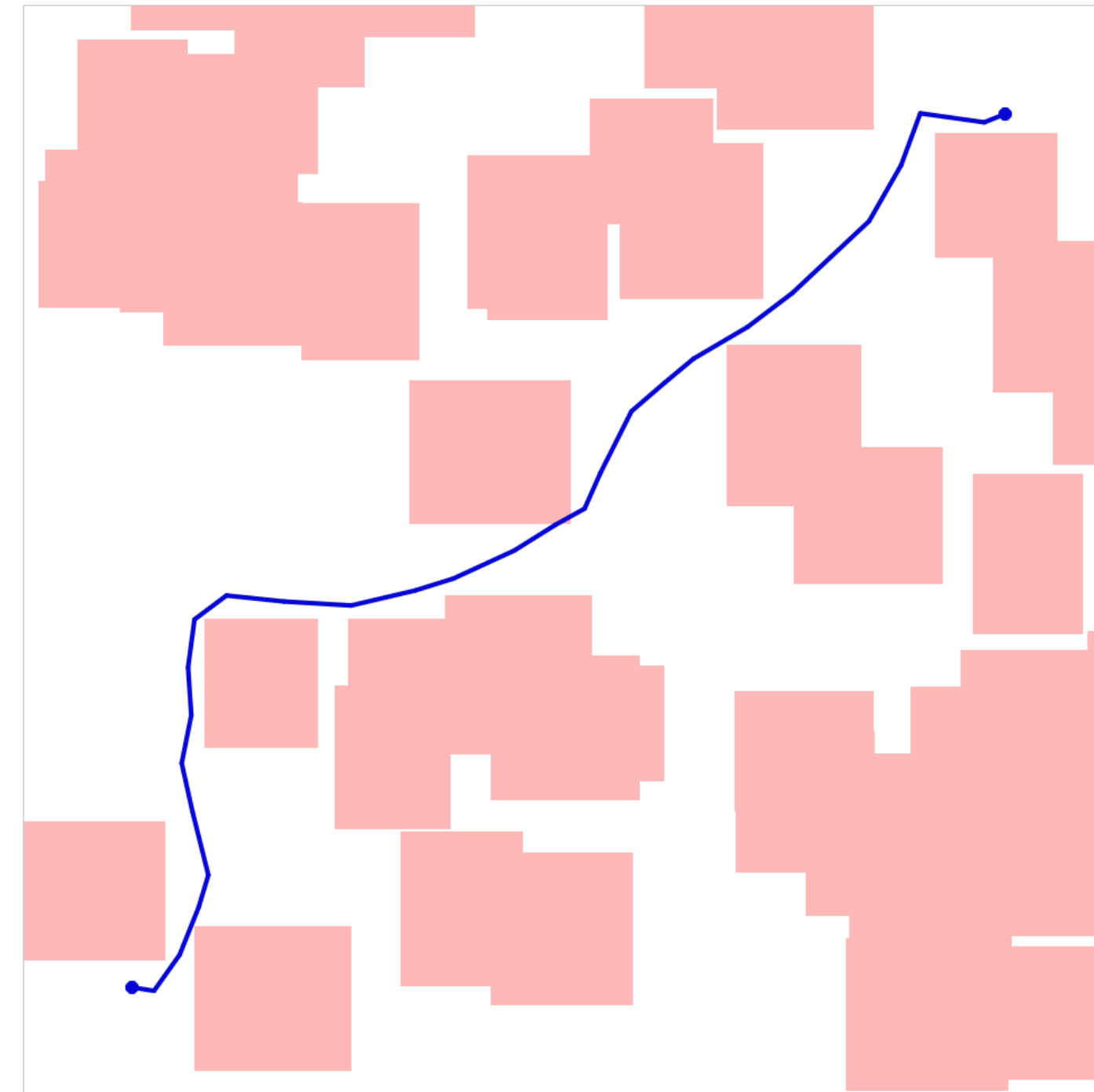
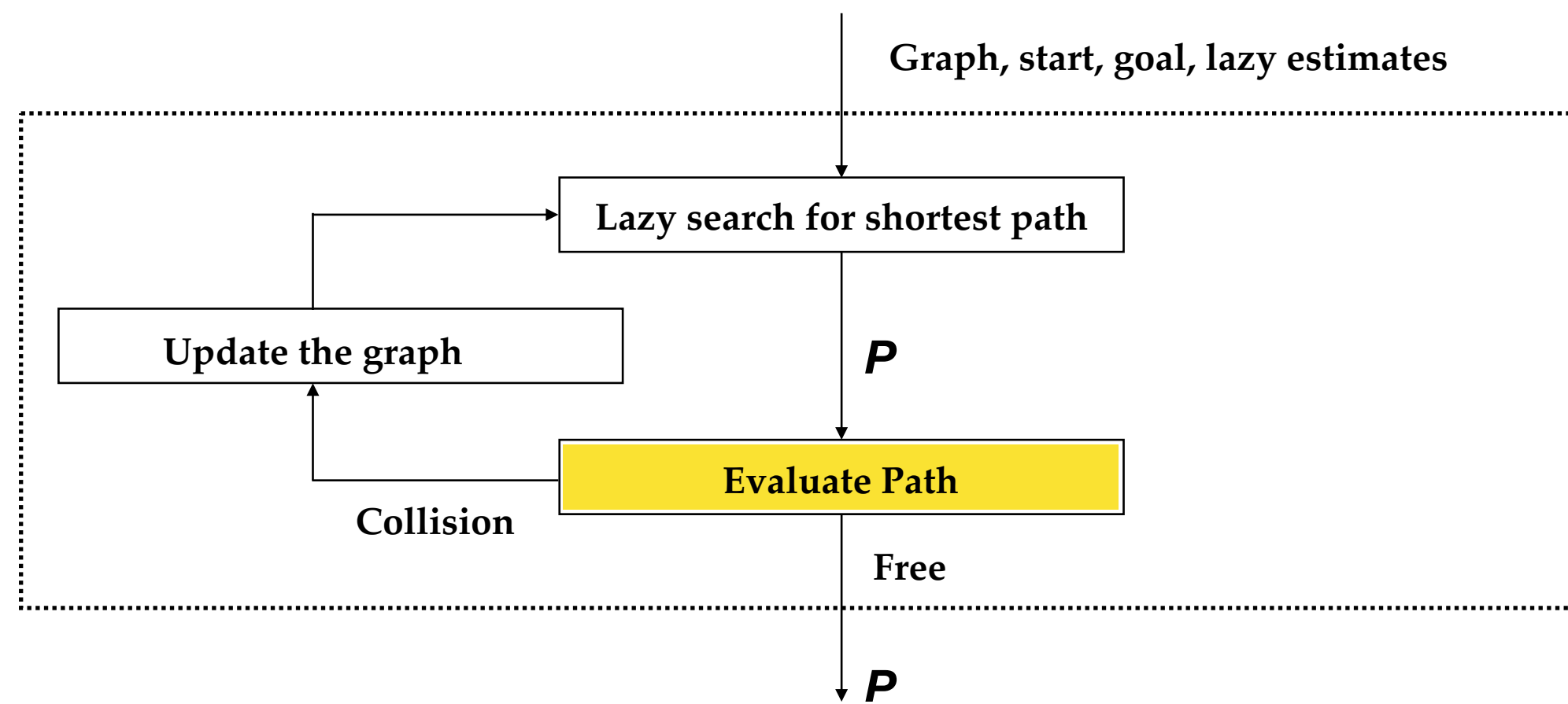
OFU on Steroids!



Optimal Slime!

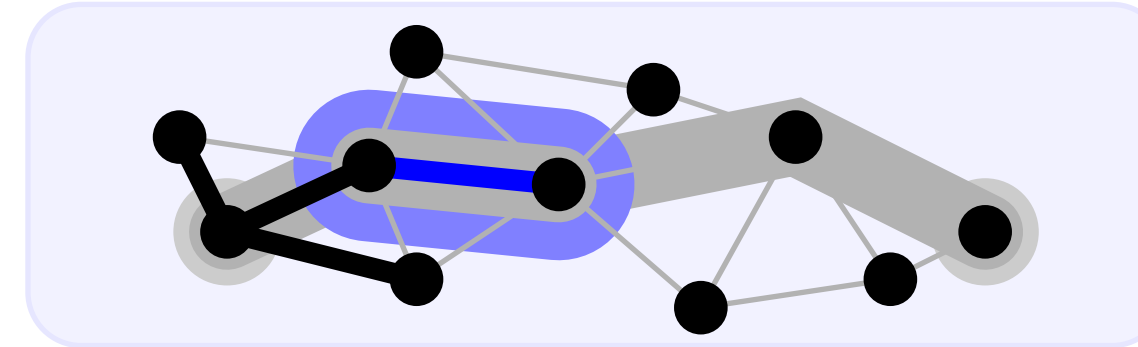
# LazySP

OFU on Steroids!

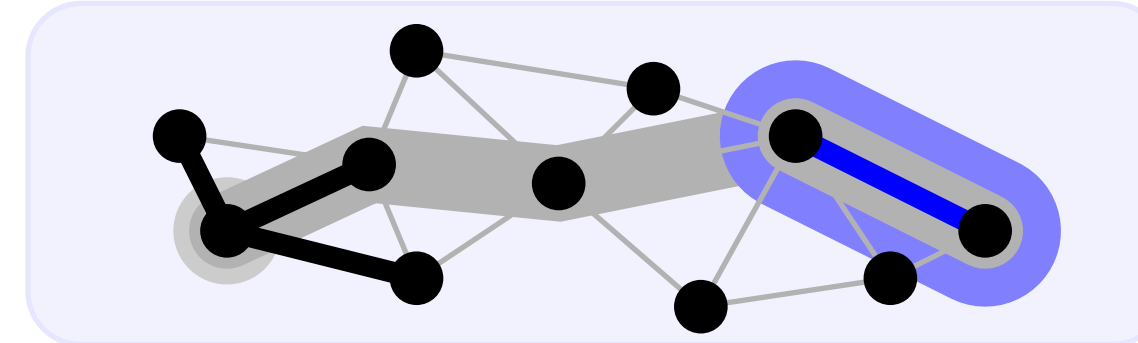


# Edge Selectors

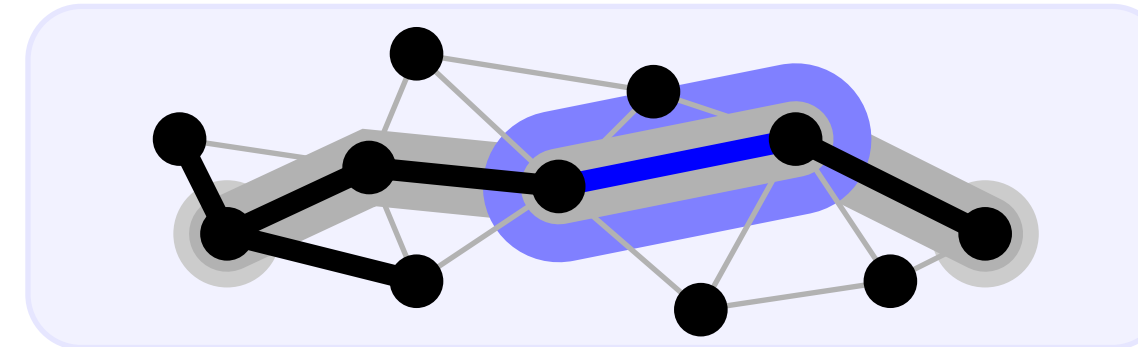
**Forward**  
(first unevaluated edge)



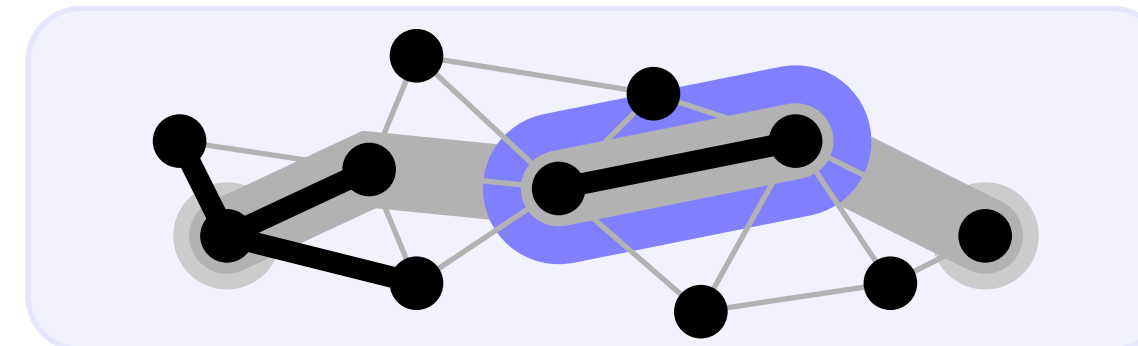
**Reverse**  
(last unevaluated edge)



**Alternate**  
(alternate Forward and Reverse)



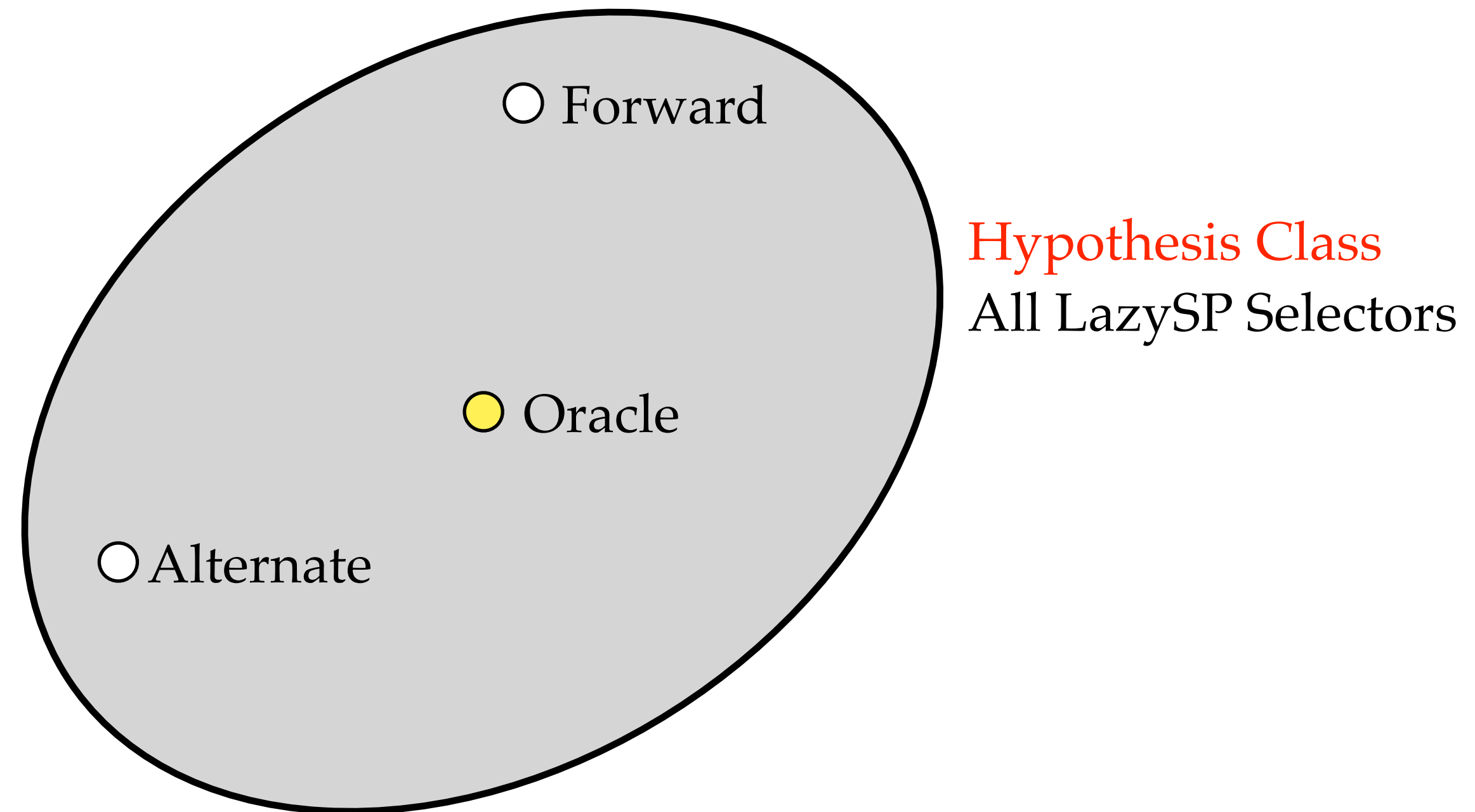
**Bisect**  
(furthest from an unevaluated edge)



# The Realizability Assumption

Can we Learn to  
Imitate the Oracle?

*Leveraging experience in lazy search, Bhardwaj et al., RSS 2019.*



The Oracle is a LazySP Selector!

*The Provable Virtue of Laziness in Motion Planning, Hagtalab et al., ICAPS 2018.*

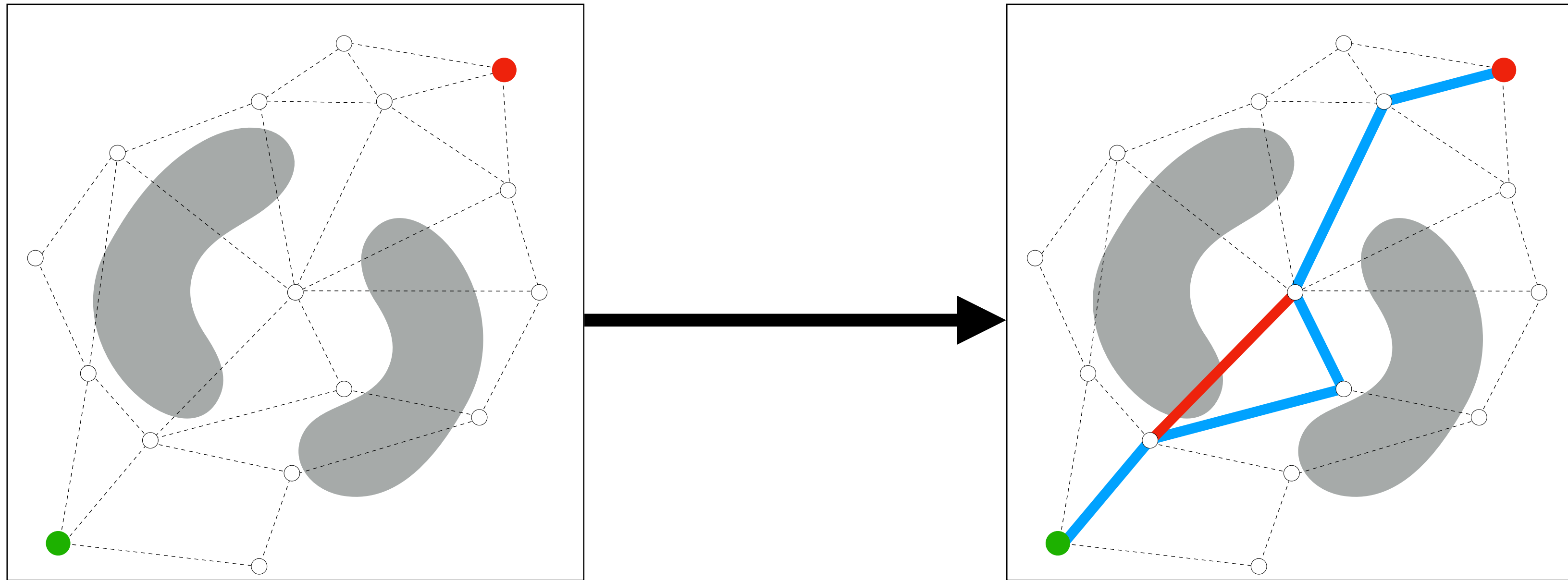
Is there a Search Algorithm  
that **Minimizes**  
the Number of Edge Evaluations?

# LazySP

ICAPS 2018 [Best Conference Paper Award Winner]

**First Provably Edge-Optimal A\*-like Search Algorithm**

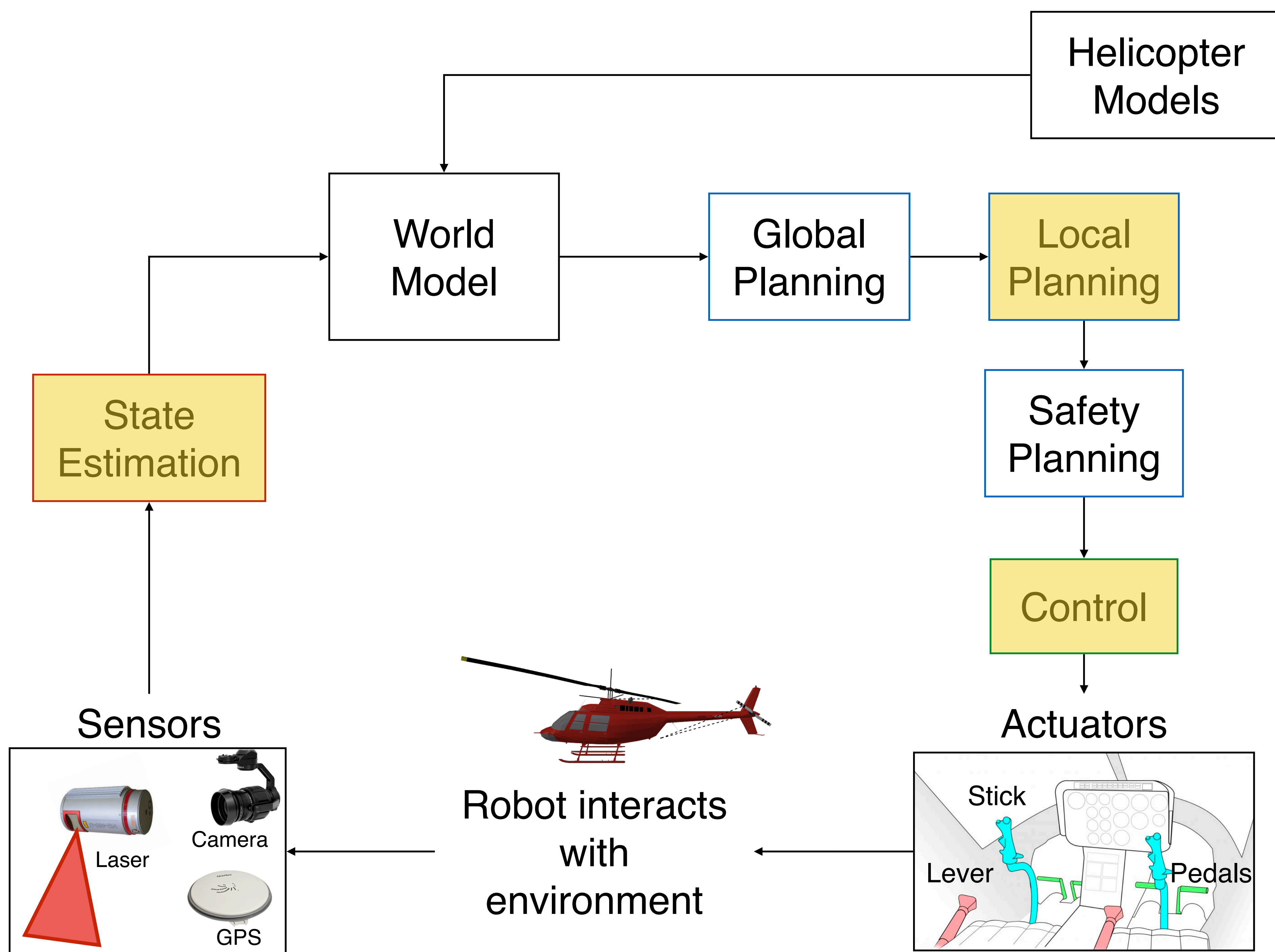
# Sampling-Based Motion Planning



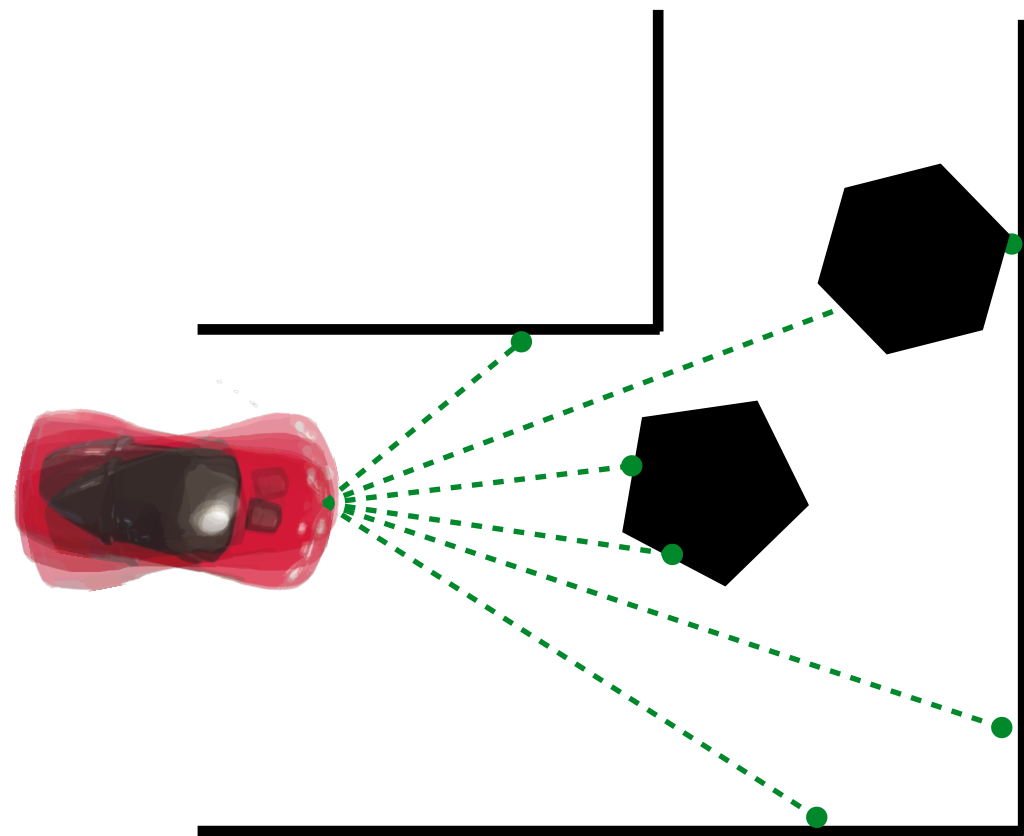
**CREATE**

**SEARCH**

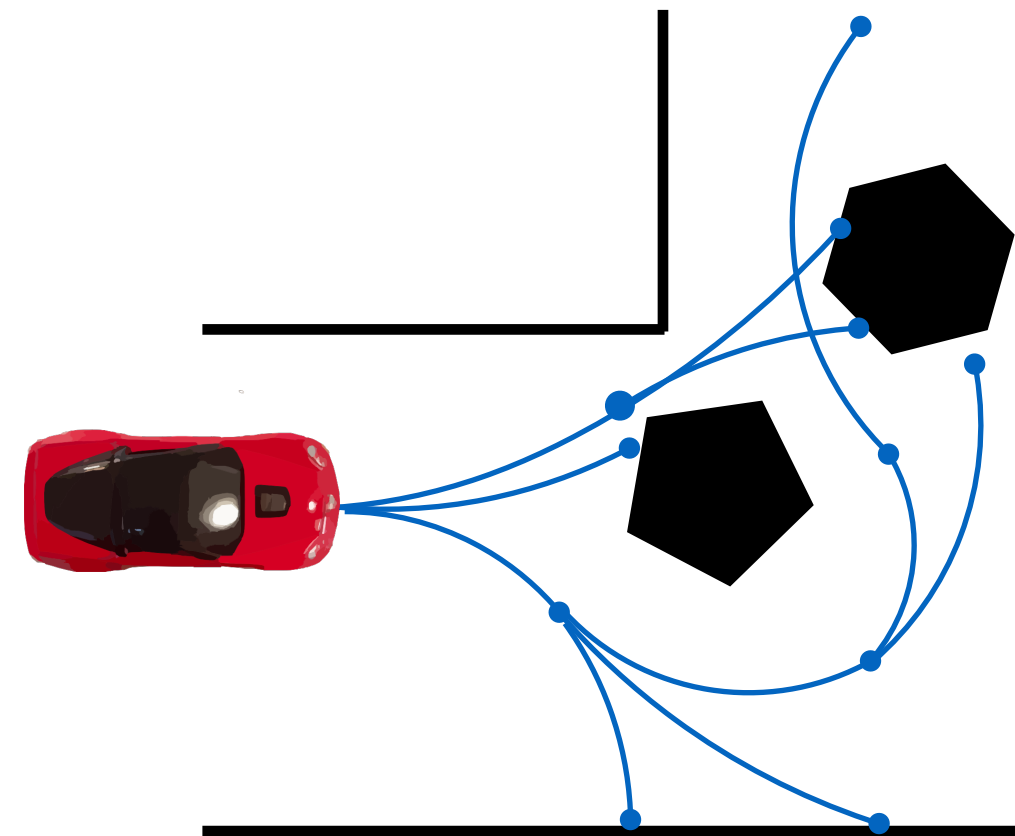
**INTERLEAVE**



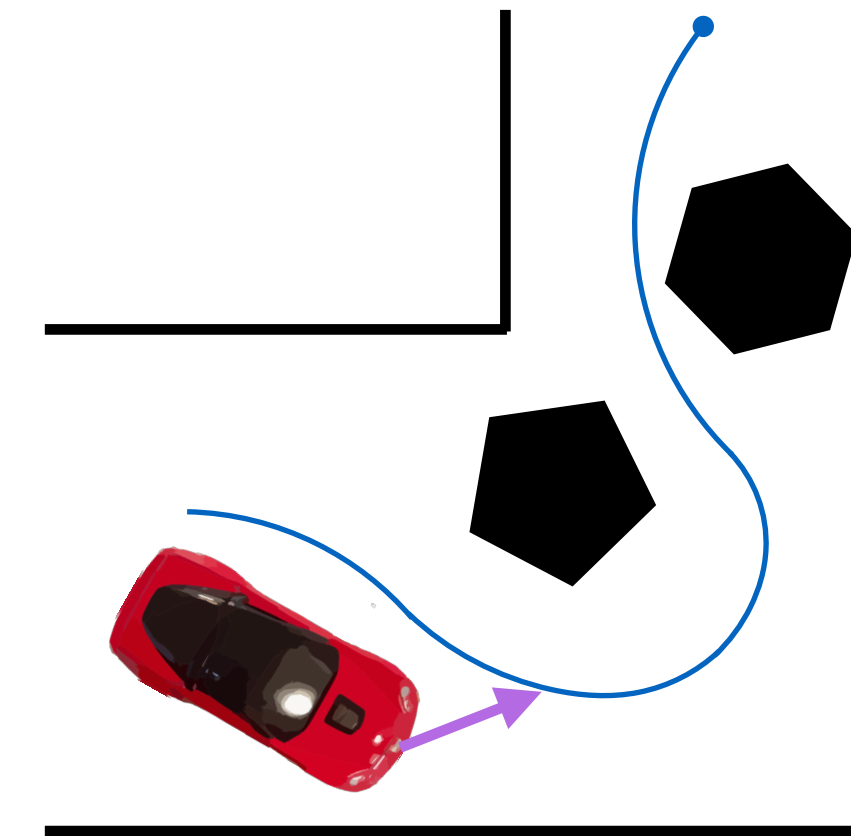
**Estimate  
state**



**Plan a  
sequence of  
motions**



**Control  
robot to  
follow plan**



# CSE 478 Robot Autonomy

## Lazy Search

Siddhartha Srinivasa (siddh@)  
Abhishek Gupta (abhgupta@)

TAs:  
Rohan Baijal (rbaijal@)  
Sidhartha Talia (sidtalia@)  
Christopher Tan (tan7271@)  
Helen Wang (yiruwang@)

