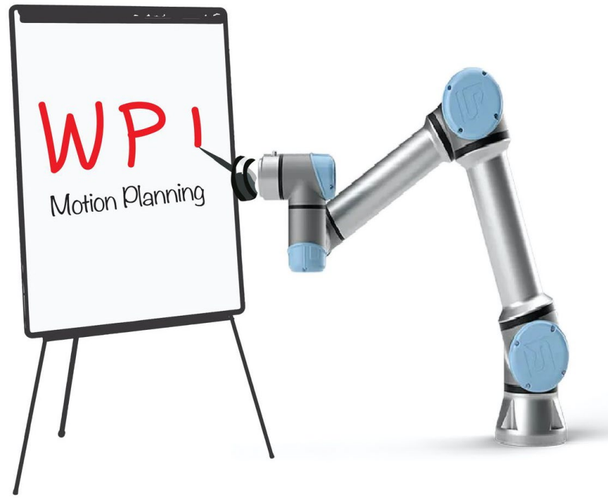
RBE550
Motion Planning
Learning and Motion Planning II



Constantinos Chamzas
www.cchamzas.com
www.elpislab.org

Last Lecture Overview

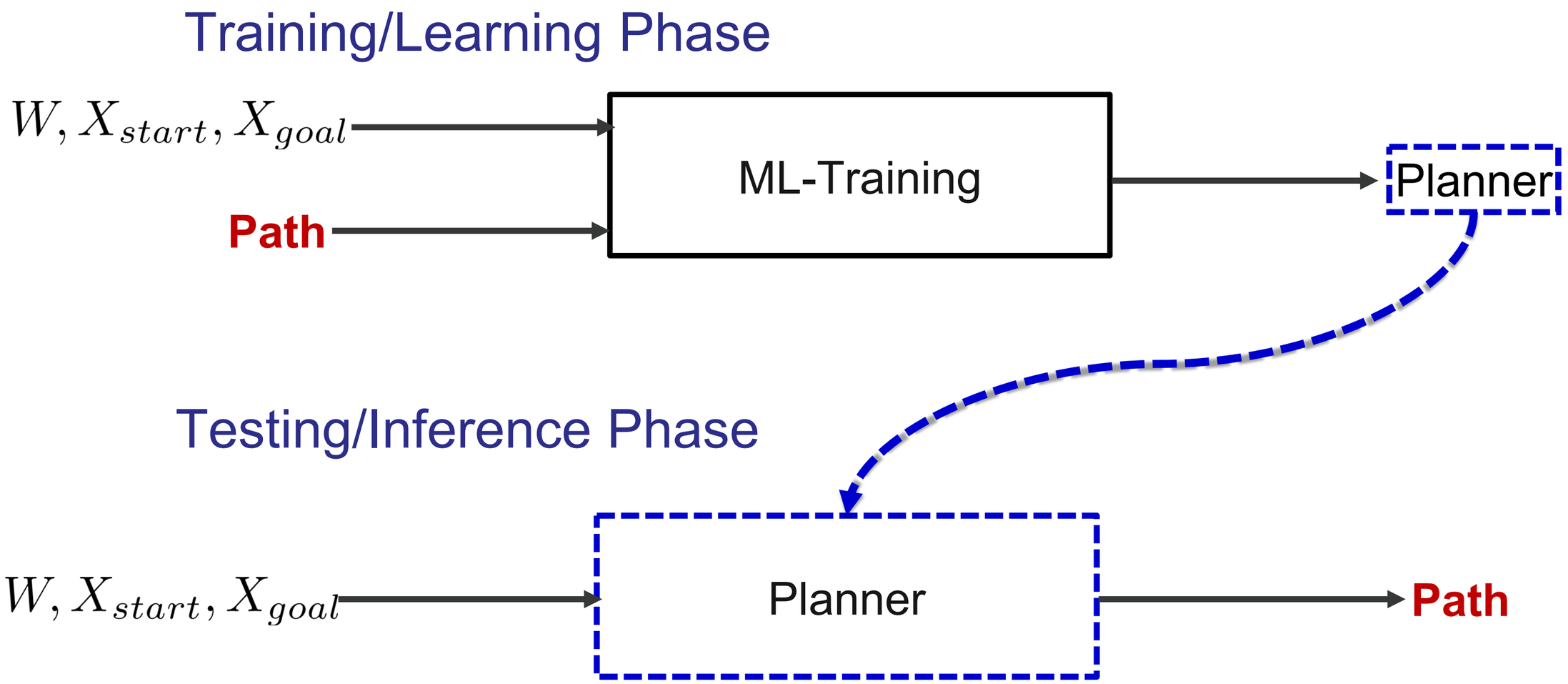
- Why Combine Learning and Motion Planning?
  – To improve planning efficiency in challenging problems

- Learning For Motion Planning Archetypes
  - Retrieve and Repair, Biased Sampling

- Discussed 4 Papers
  - Lightning(2013) Thunder (2015), Rep Sampling (2017), Rep (Roadmap) (2019)

Today: We will focus only on Biased Sampling Methods

Machine Learning and Motion Planning



Biasing the Sampling Distribution

Algorithm 1: General Sampling Based Planner

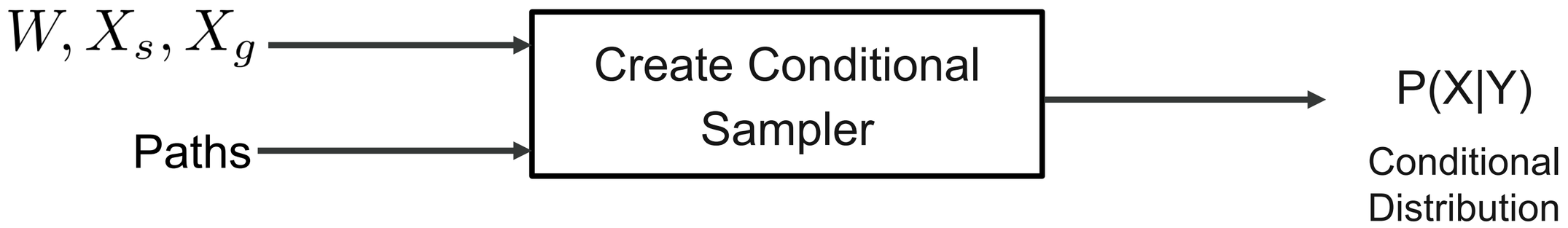
Input : Number of iterations N
Output : Graph structure G

```
1 while i ≤ N or solutionFound() do
2 |   x ∼ Uniform()
3 |   update(G, x)
4 end
5 return G
```

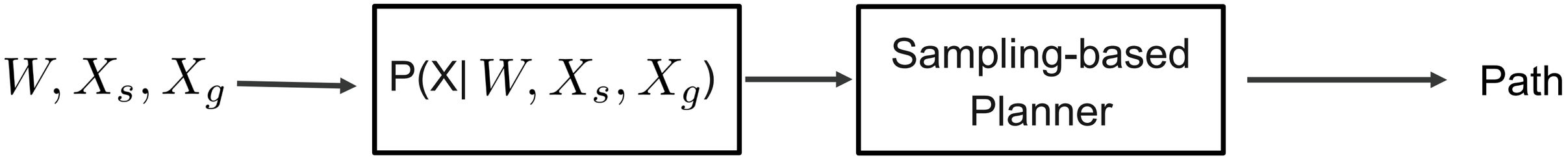
- Theoretical Insights supporting biased sampling
- Any sampling-based planner can be used

Biased Samplers Archetype

Training/Learning Phase



Testing/Inference



Learning Archetypes (Methodologies)

Learning Archetypes	Corresponding Papers
1. Retrieve and Repair	Lightning[1], **Thunder [2],** ERT[3], Sim-Obstacles[4], Traj-Pred[5]
2. Biased Samplers	**Rep-Sampling[6]**,**Rep-Roadmaps[7],** AWS[8], SPARK2D[9], CVAE[10], FLAME[11], FIRE[12]

Today’s Agenda

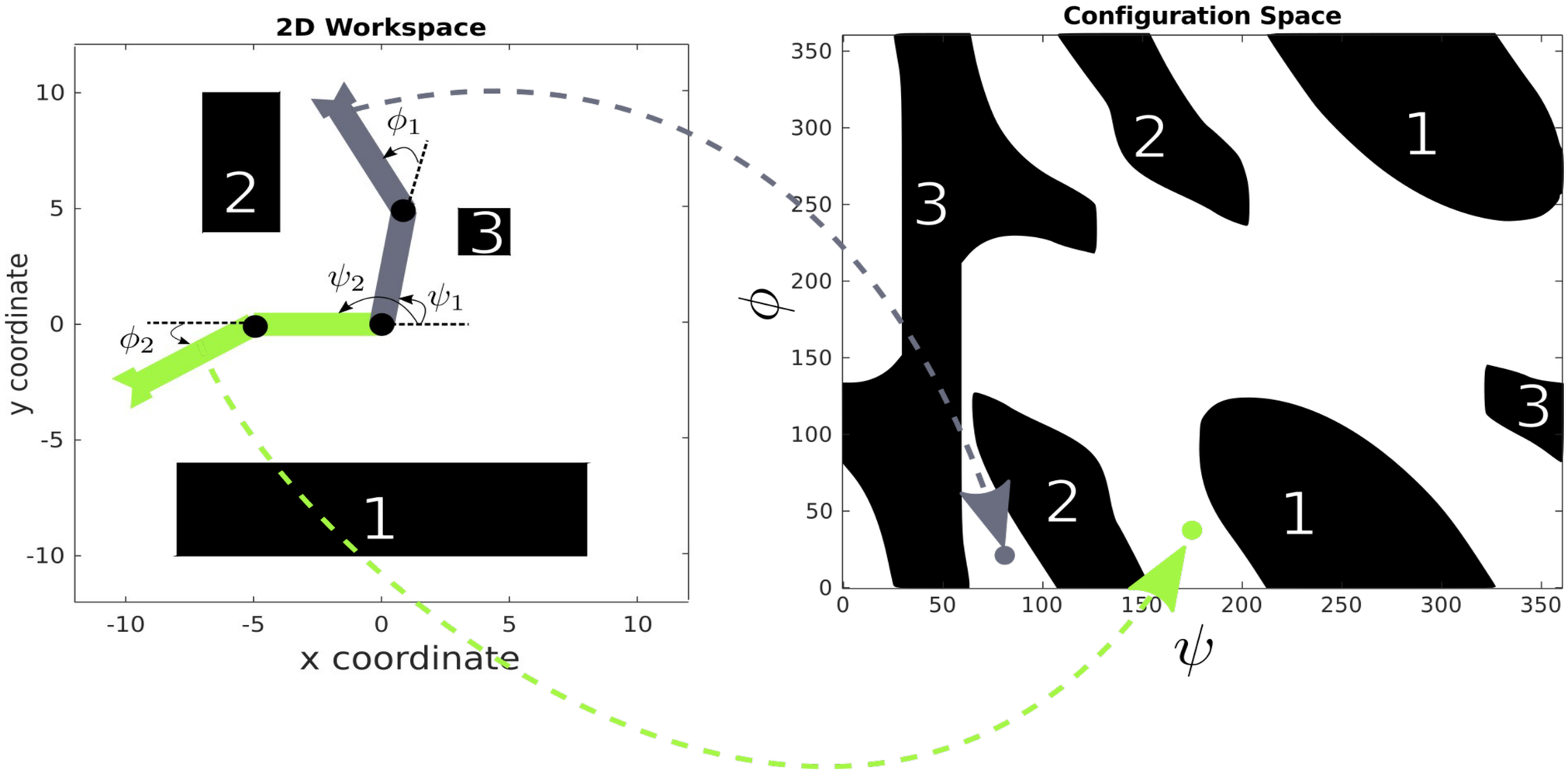
- Important Considerations
  - Locality of narrow passages
  - Discontinuity

- Biased Samplers (Cont.)
  - 2D Workspaces
  - 3D Workspaces
  - Workplace, Start and Goal

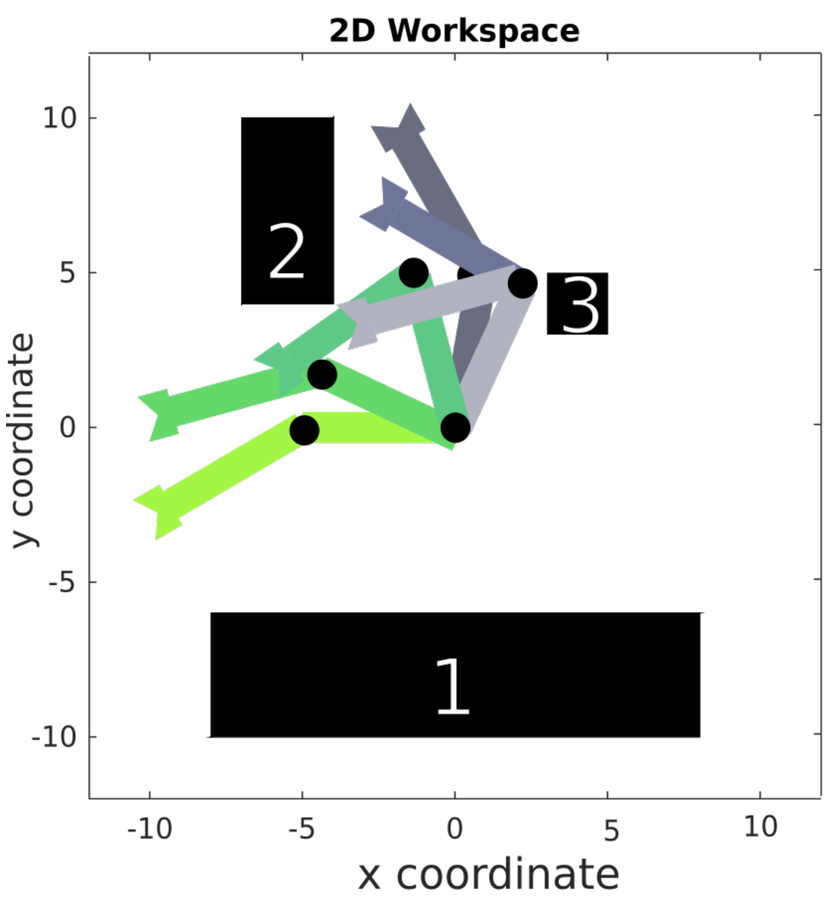
Learning Archetypes (Methodologies)

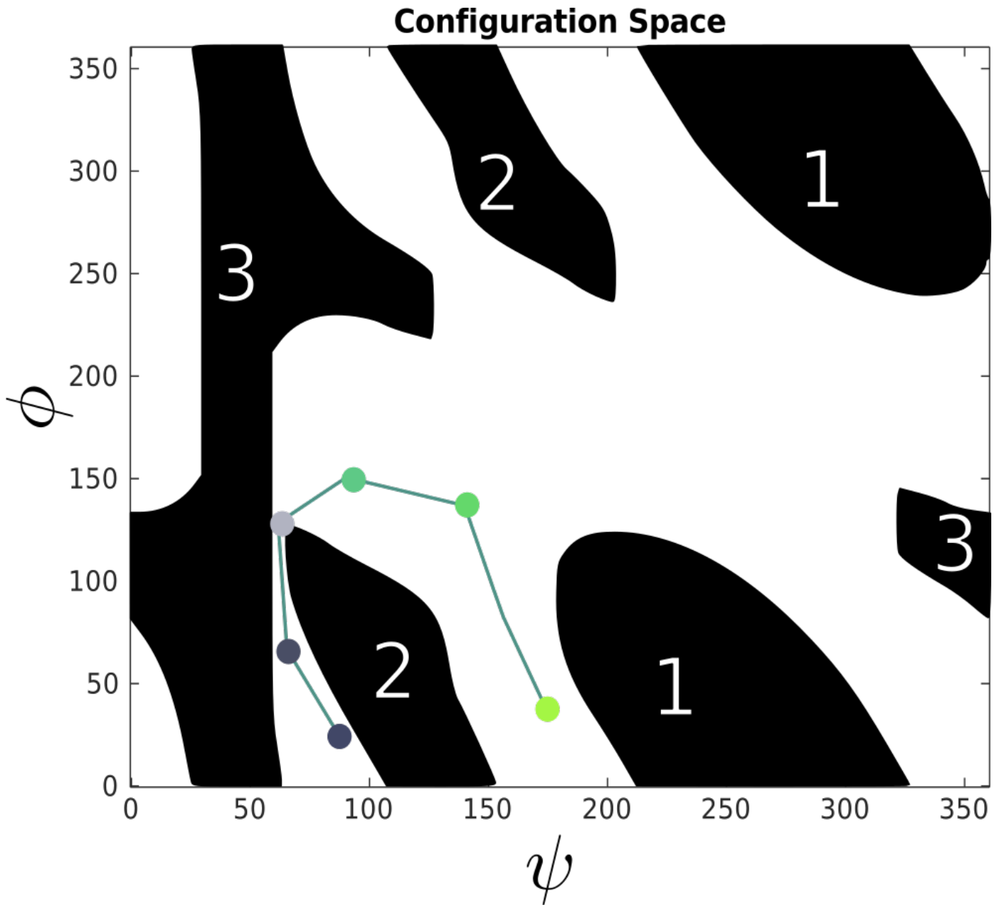
Learning Archetypes	Corresponding Papers
1. Retrieve and Repair	Lightning[1], Thunder [2], ERT[3], Sim-Obstacles[4], Traj-Pred[5]
2. Biased Samplers	Rep-Sampling[6],Rep-Roadmaps[7], AWS[8], **SPARK2D[9]**, **CVAE[10]**, **FLAME[11]**, **FIRE[12]**

A 2D Manipulator Problem



A 2D Manipulator Path/Solution

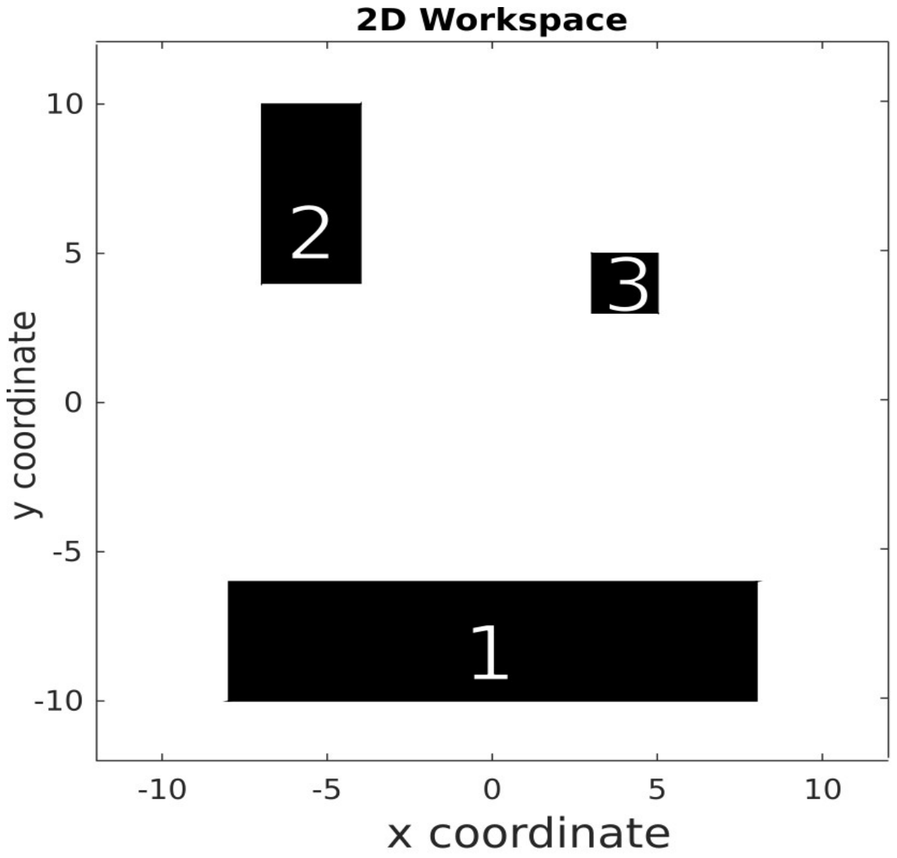


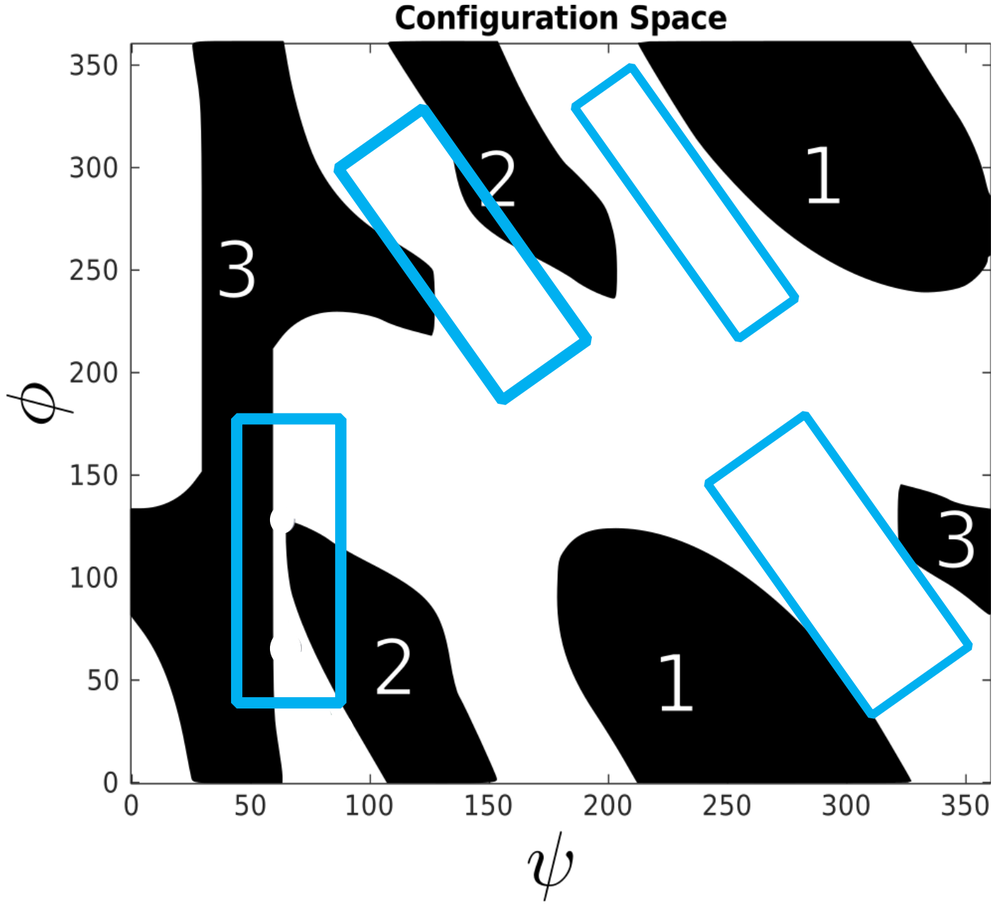


Conditioned Biased Sampling Distributions

- P(x): Learning Only Invariants
- P(x| Start, Goal): Sampling conditioned on start and goal
- **P(x| W): Sampling biased from workspace features**
- P (x| Start, Goal, W): Sampling leveraging both workspace, start and goal information

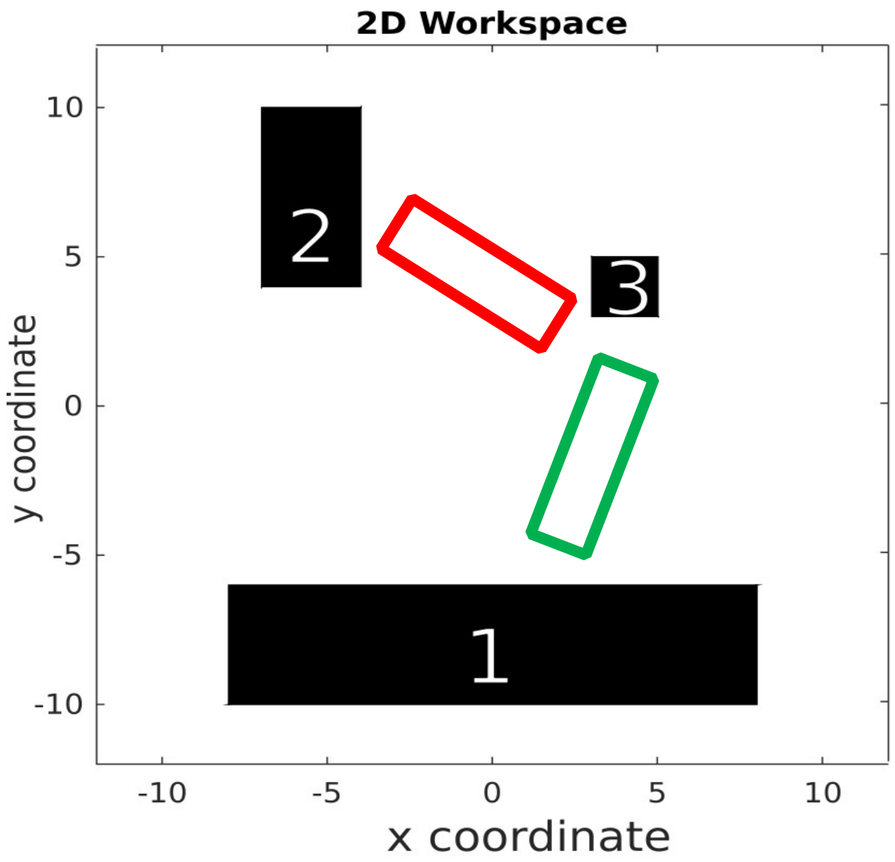
Using only workspace features

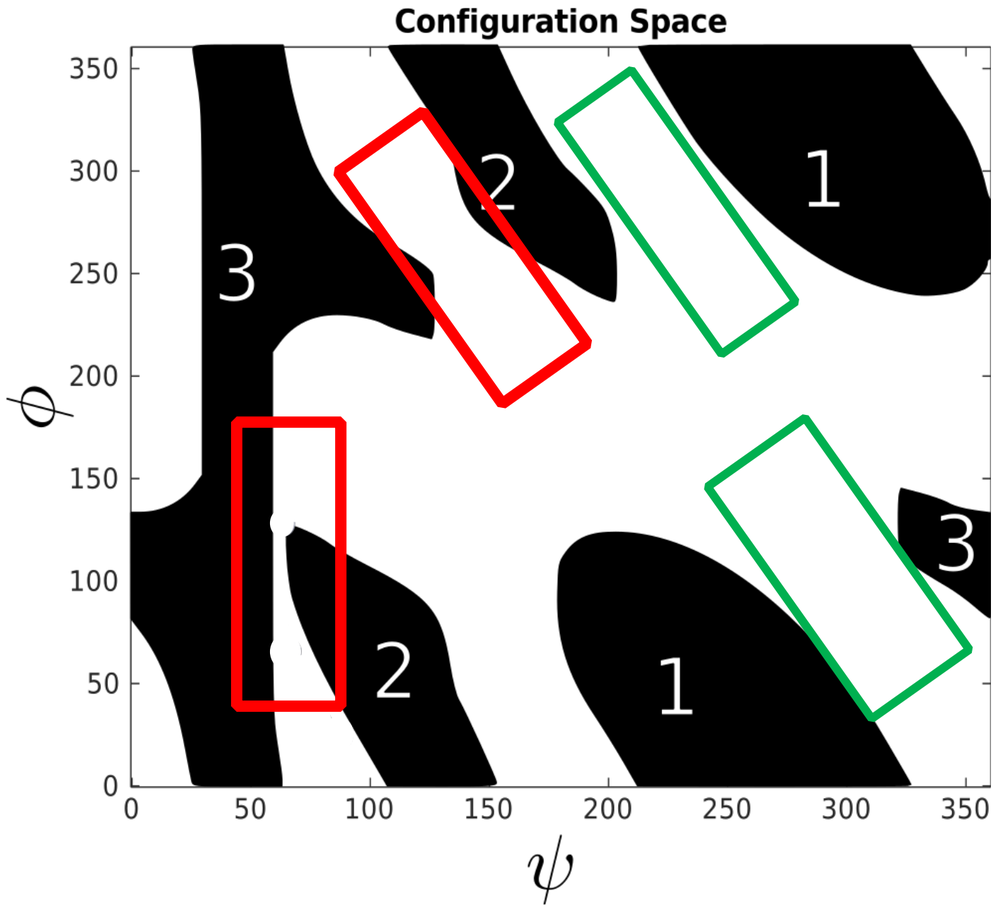




Since we don’t have start/goal information we will sample on **all** narrow passages

Consideration 1: Locality of narrows passages

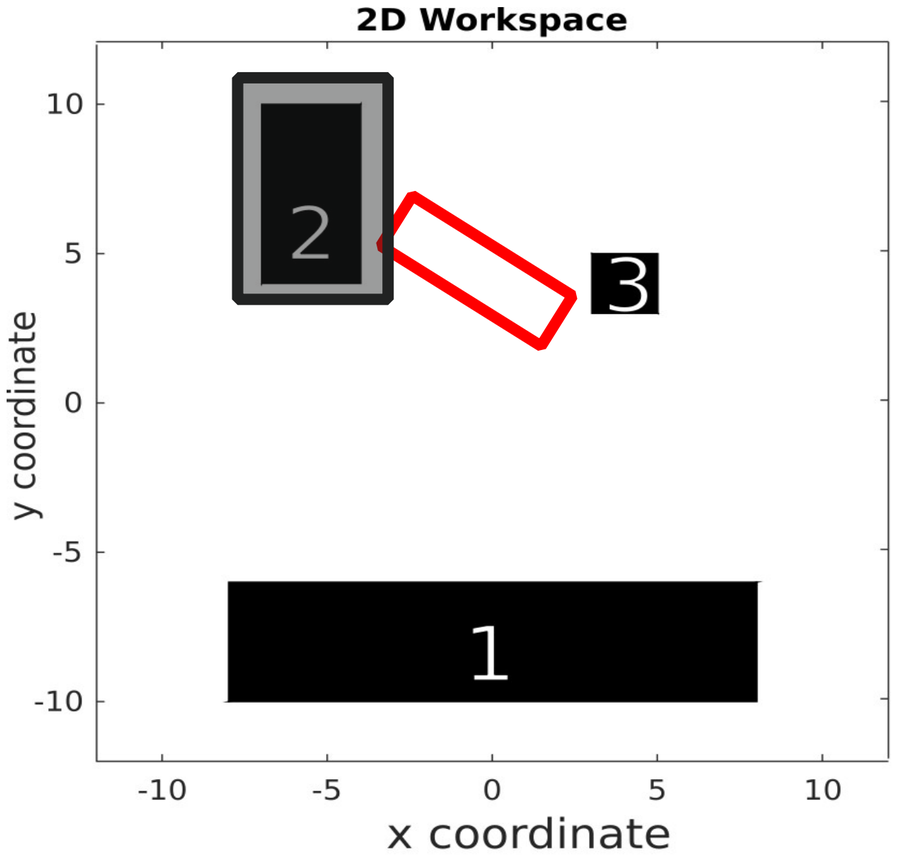


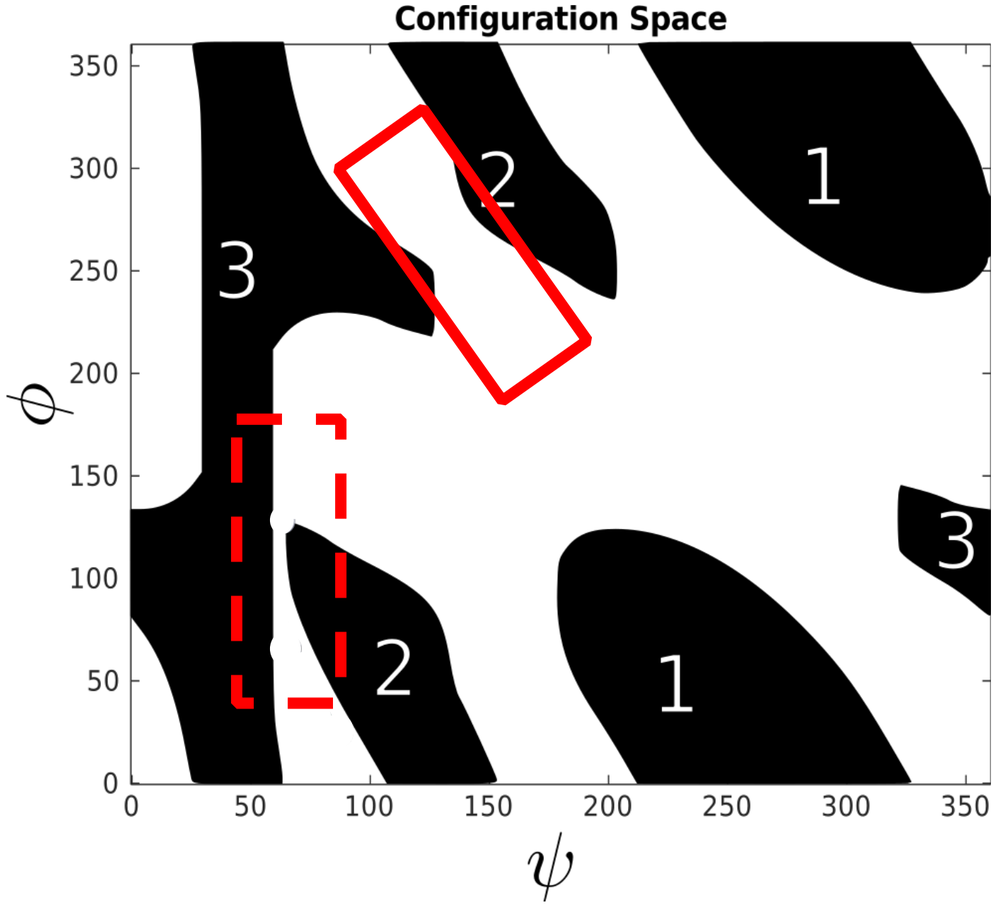


Local workspace areas create different narrow passages !

Consideration 2: Discontinuity

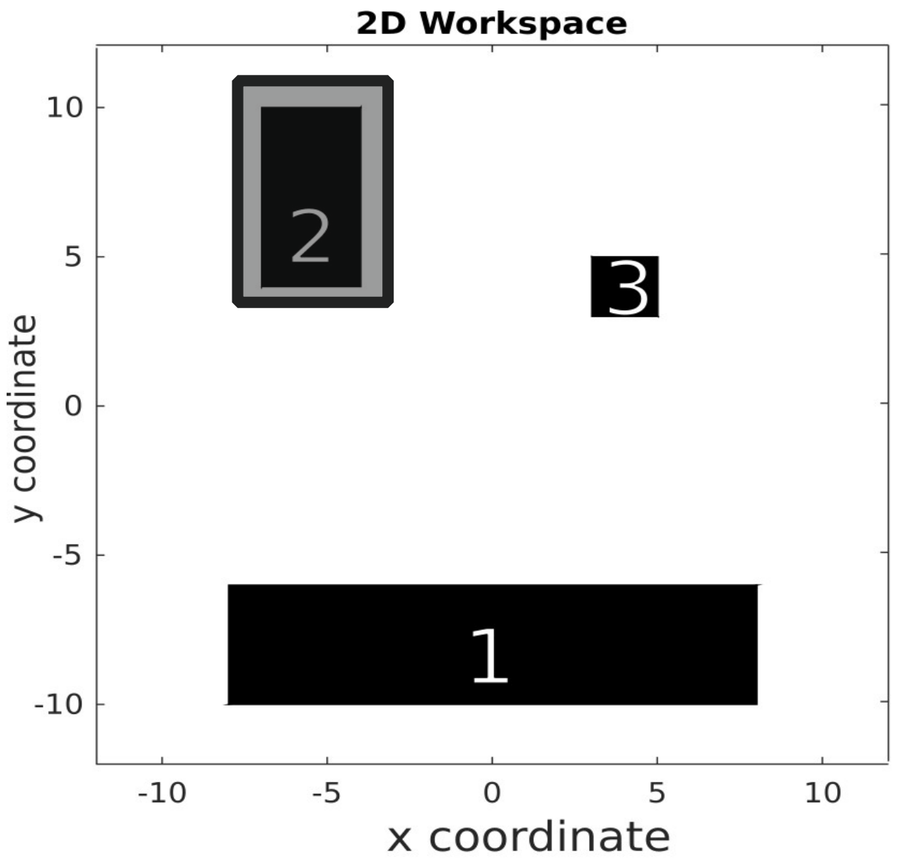
Lets make obstacle 2 a bit larger

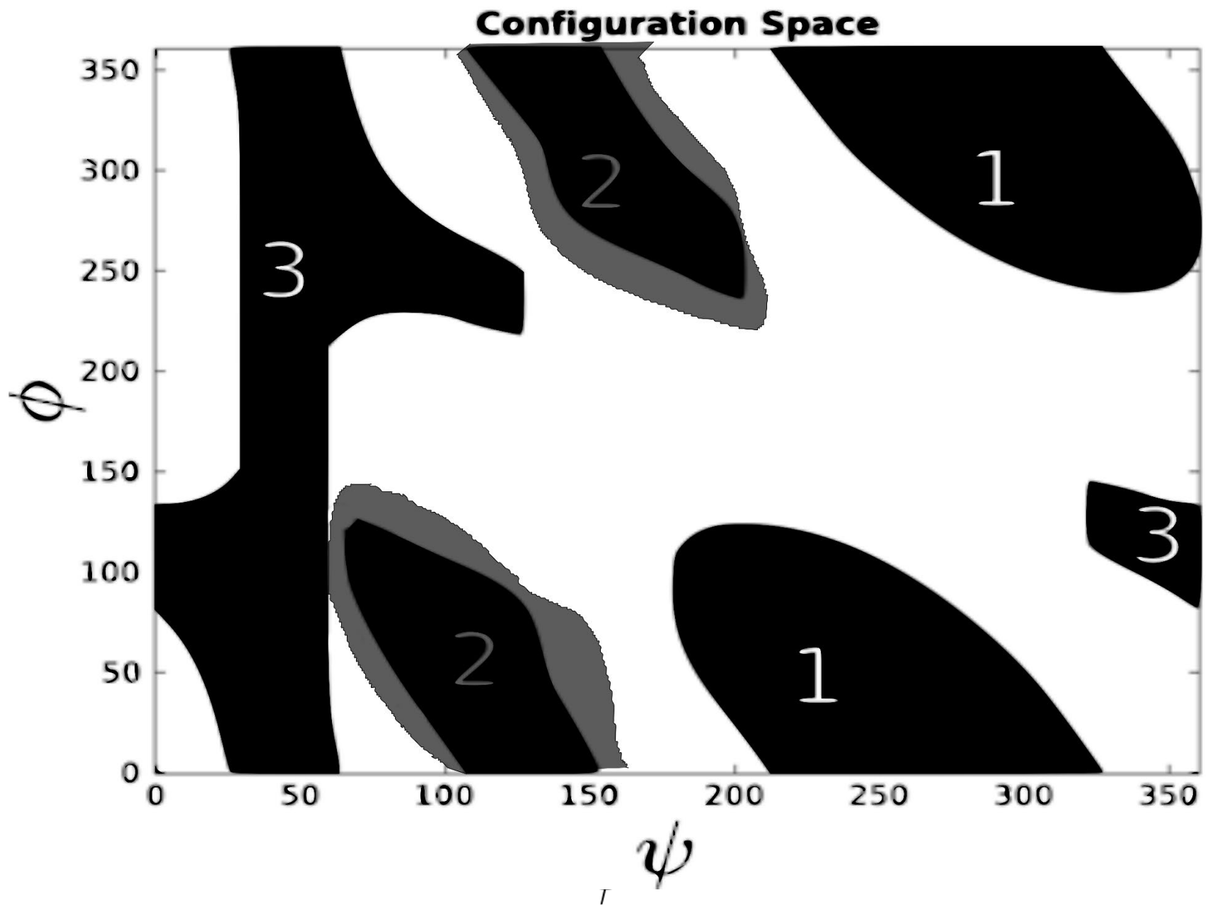




What will happen to the narrow passages in the c-space?

Consideration 2: Discontinuity





Small changes in the workspace can affect C-Space solutions a lot

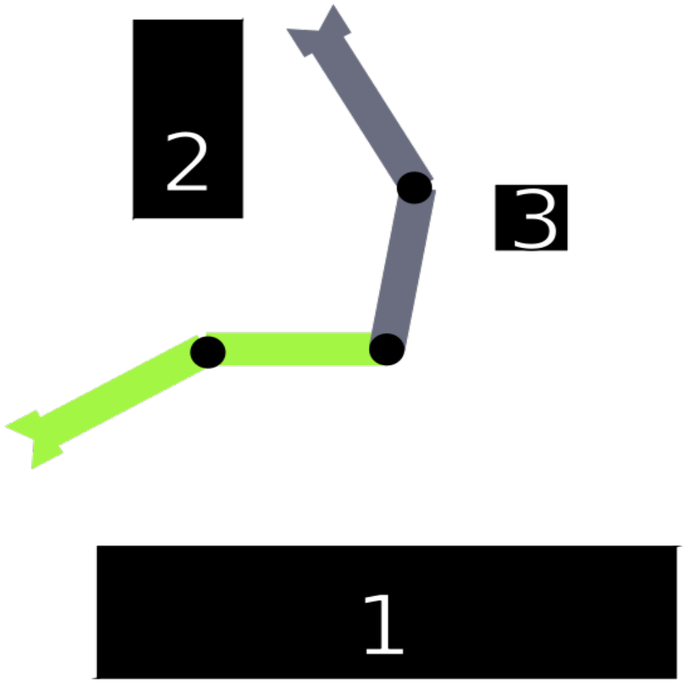
Today’s Agenda

- Important Considerations
  - Locality of narrow passages
  - Discontinuity

- Biased Samplers (Cont.)
  - **2D Workspaces**
  - 3D Workspaces
  - Workplace, Start and Goal

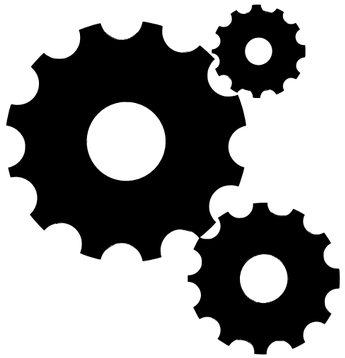
Learning a biased sampling distribution (Workspace)

Input



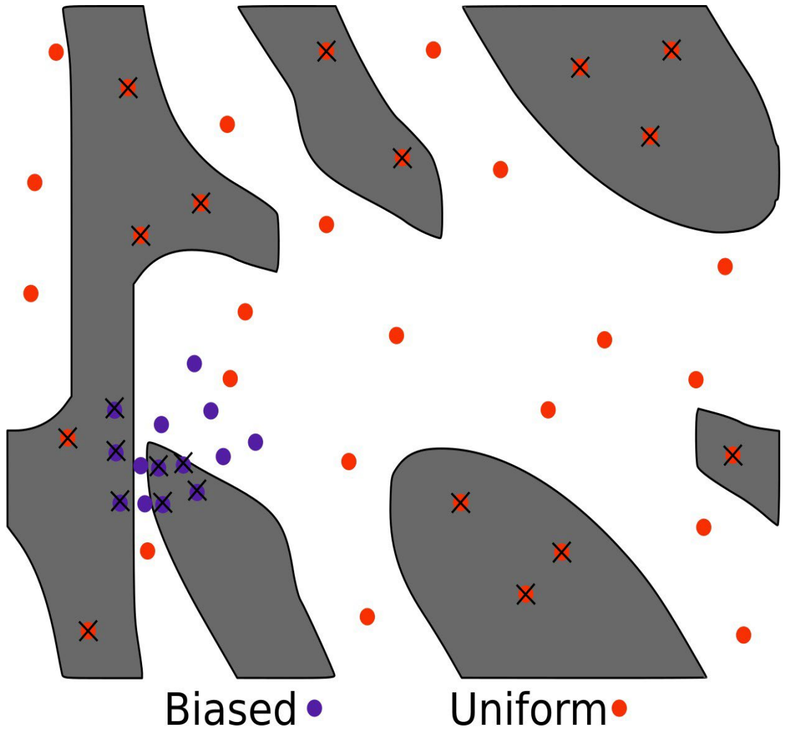
Workspace W

Learning Model



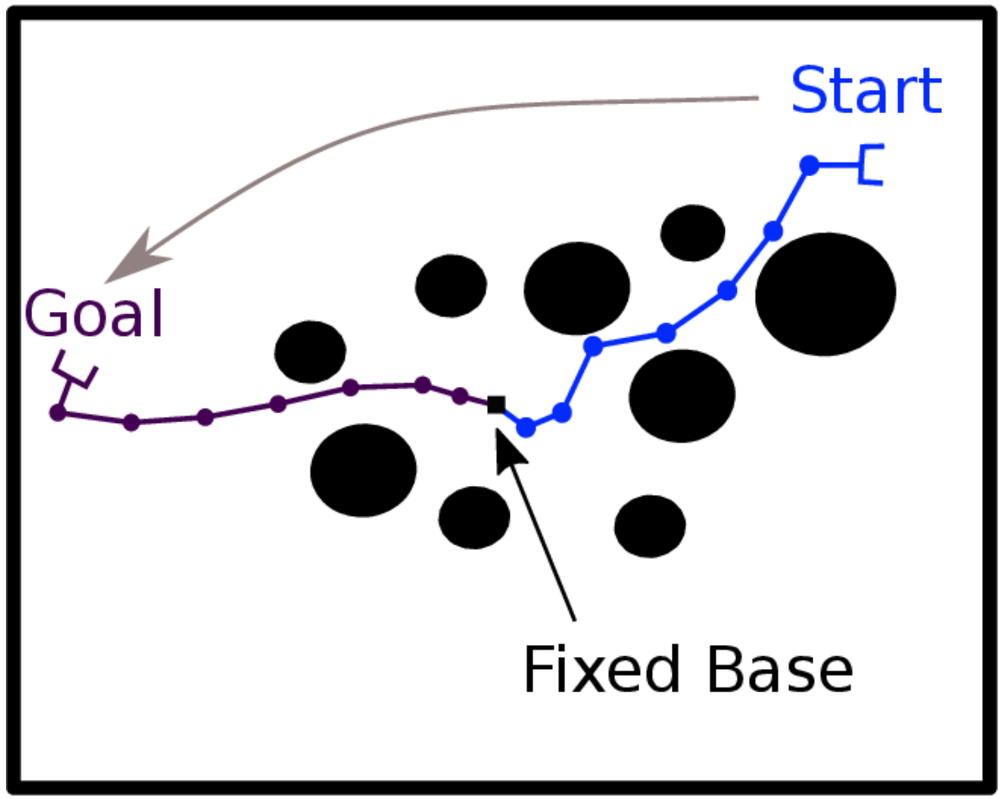
$Map: [W] \rightarrow X$

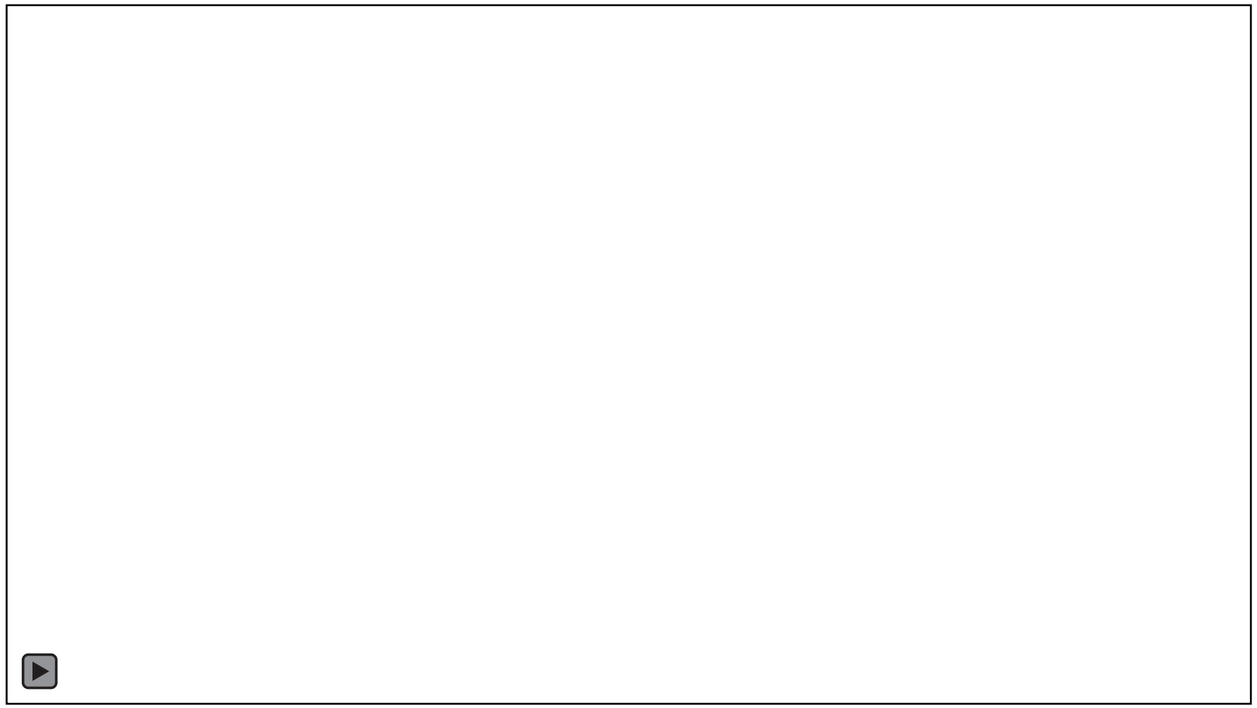
Output



Sampling Dist. $\mathrm{P}(X|W)$

A 2D workspace challenging problem



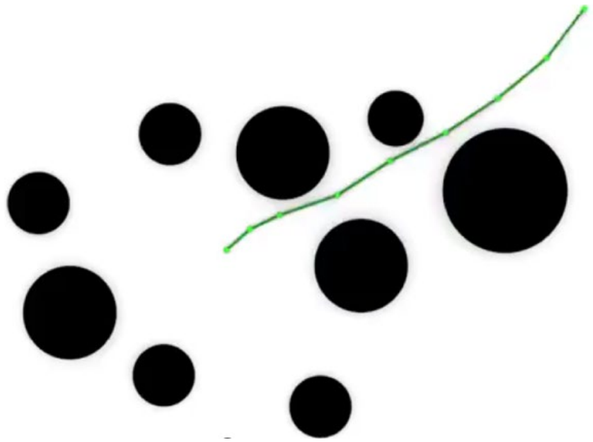


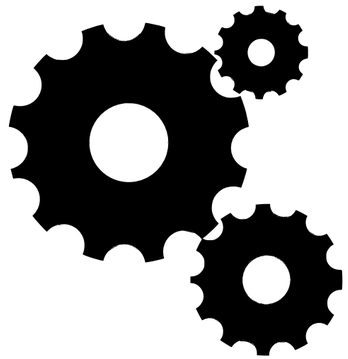
Problem Formulation

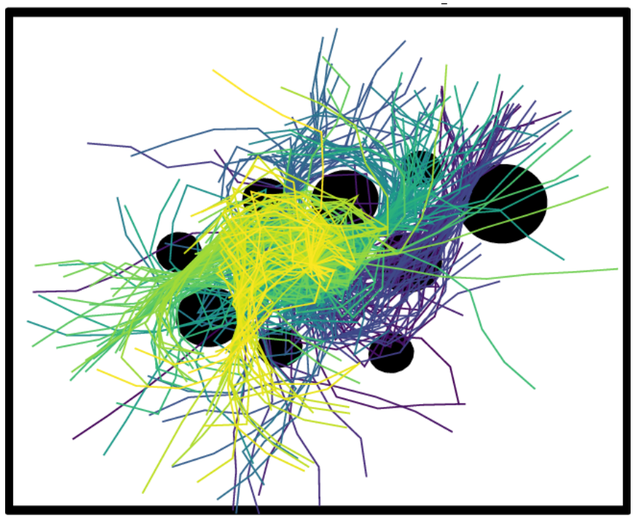
Input

Learning Model

Output





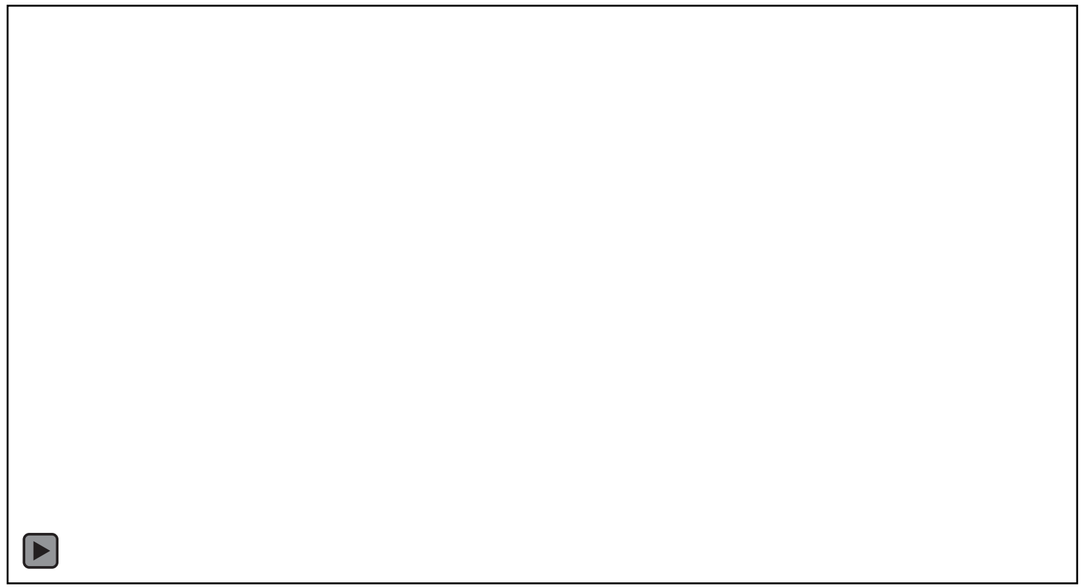


Workspace W

$Map: [W] \rightarrow X$

P(X|W)

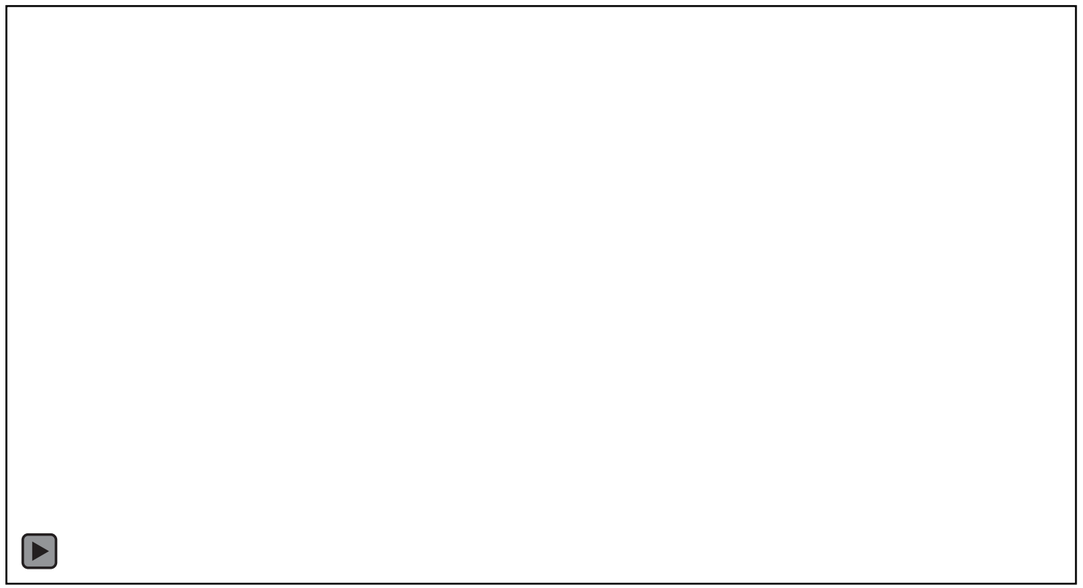
Dealing with Locality (Consideration 1)

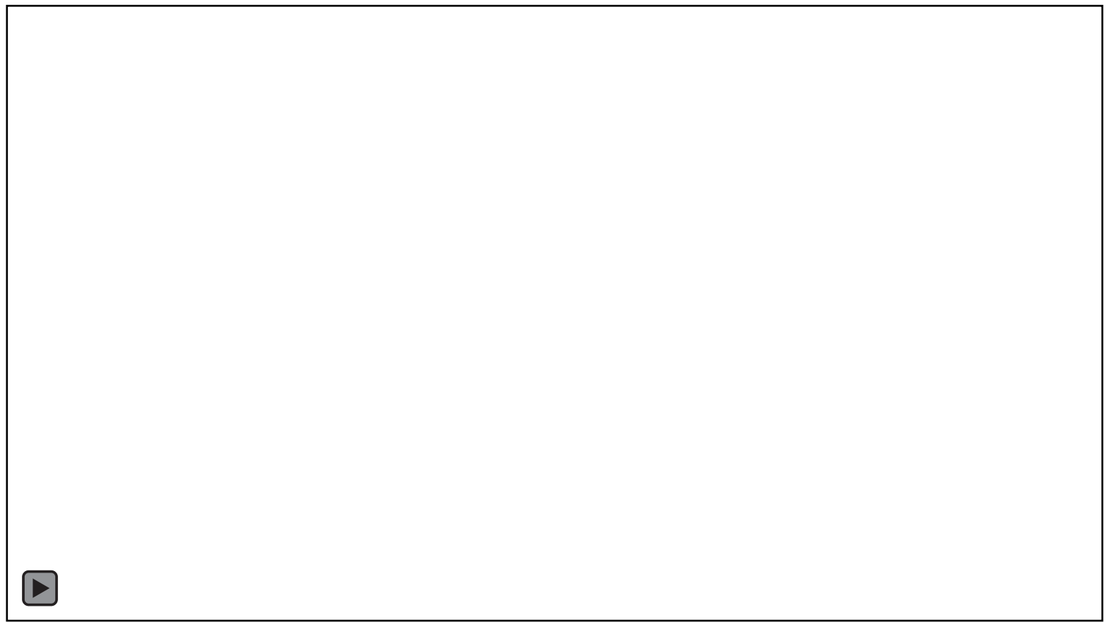


A potential solution:

We will only consider pairs of obstacles (local workspaces)

Dealing with Discontinuity (Consideration 2)





Problem:

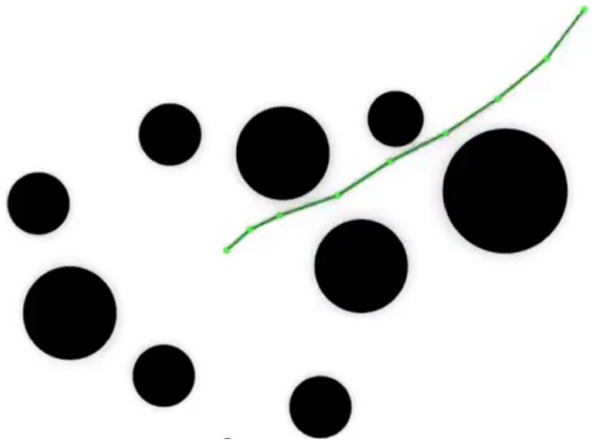
Small changes in the workspace can affect C-Space solutions a lot

Dealing with Discontinuity (Consideration 2)

Input

Learning Model

Output

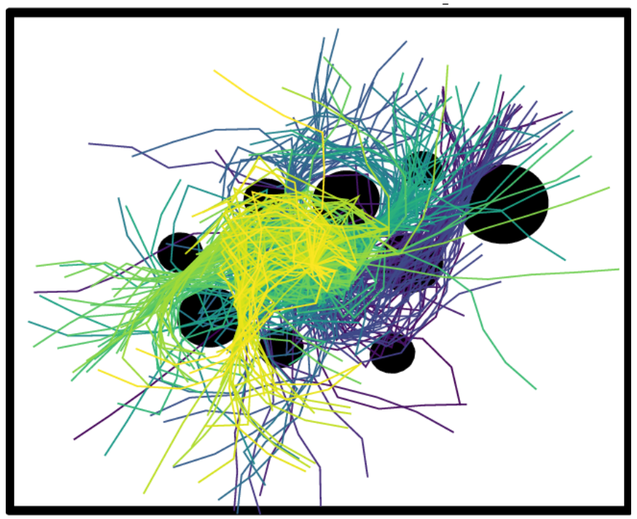


Workspace W

Database/Library of Experiences

$d(W', W) \rightarrow [0, 1]$

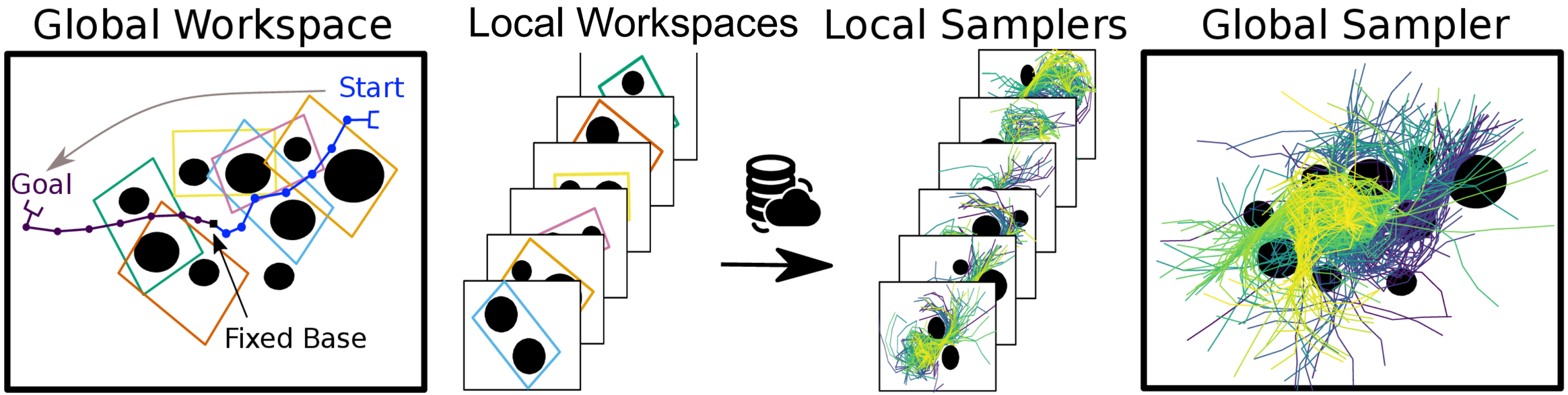
$W' : P'(X)$



P(X|W)

A potential solution: retrieval methods are discontinuous/multimodal by nature

SPARK2D Framework

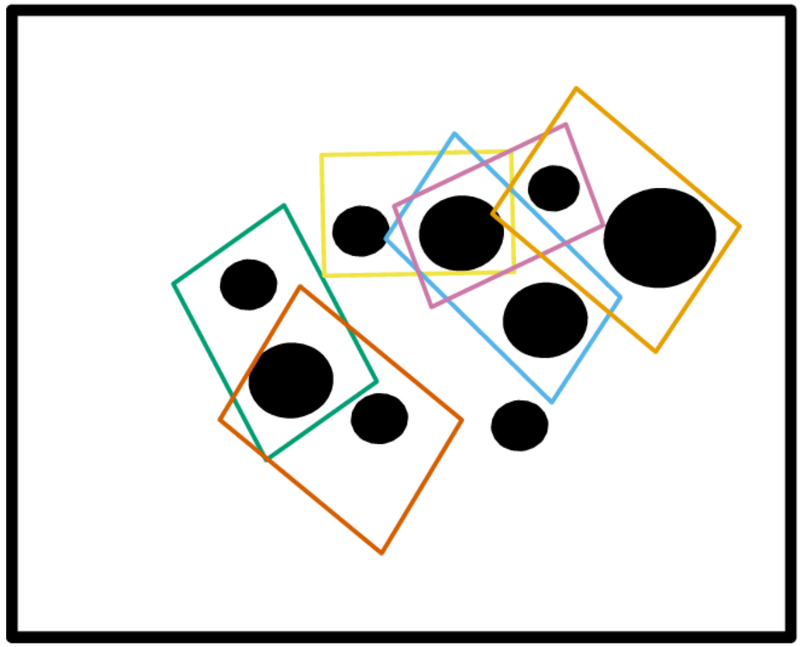


1. Decompose to local primitives
2. Creating a local sampler for every local workspaces [OFFLINE]
3. Retrieve local samplers given the local workspaces [ONLINE]
4. Synthesize a global sampler from the local samplers

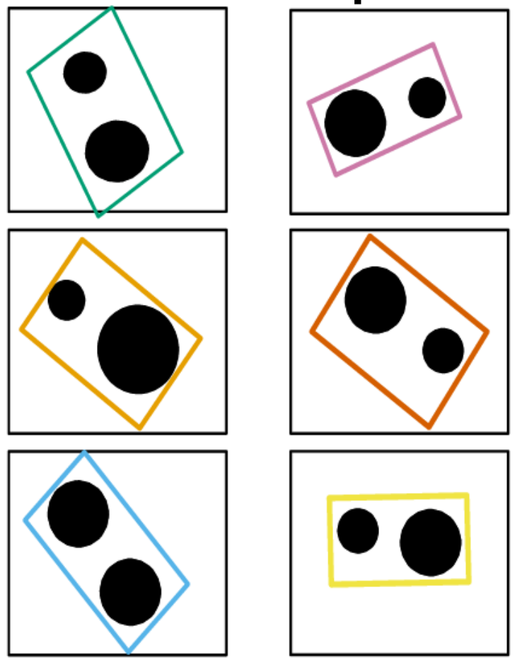
[9] Chamzas et al "Learning from Local Experiences for Efficient Motion Planning", *ICRA*, 2019

1) Decomposition to local primitives

Global Workspace



Local Workspaces



We represent each local primitive with 6 numbers

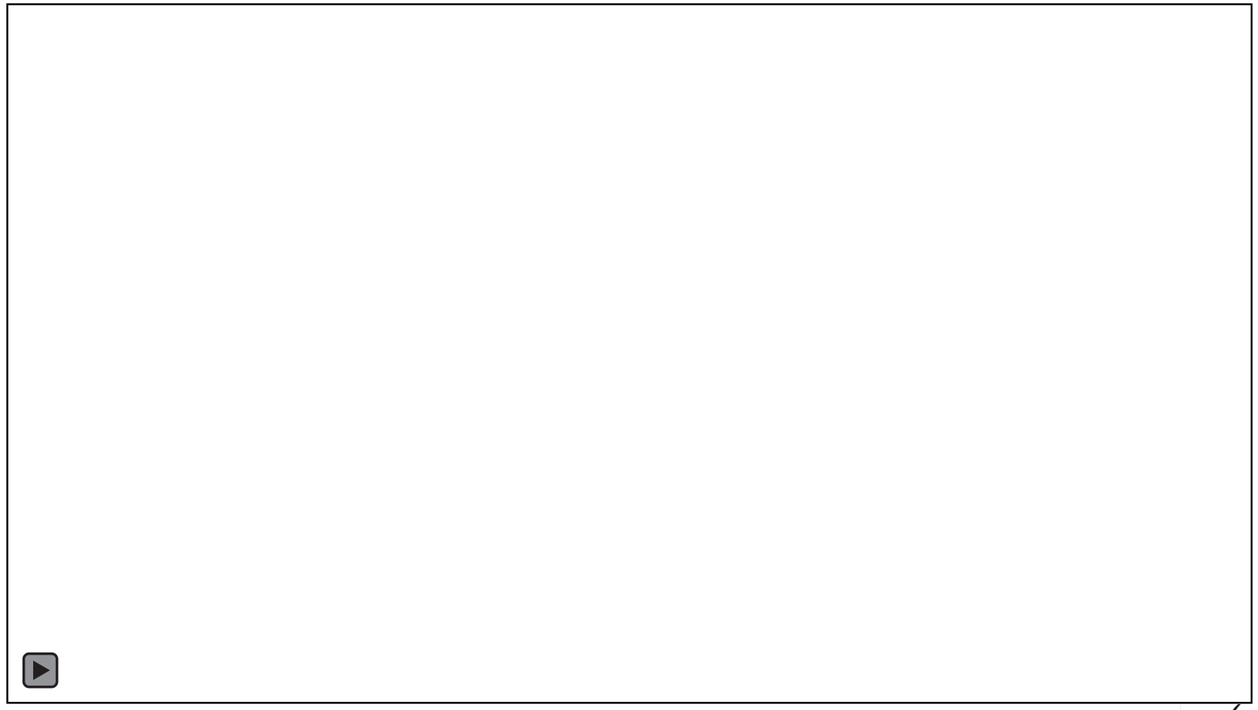
$$lw = \{x_a, y_a, r_a, x_b, y_b, r_b\}$$

Local primitives space is 6 dimensional

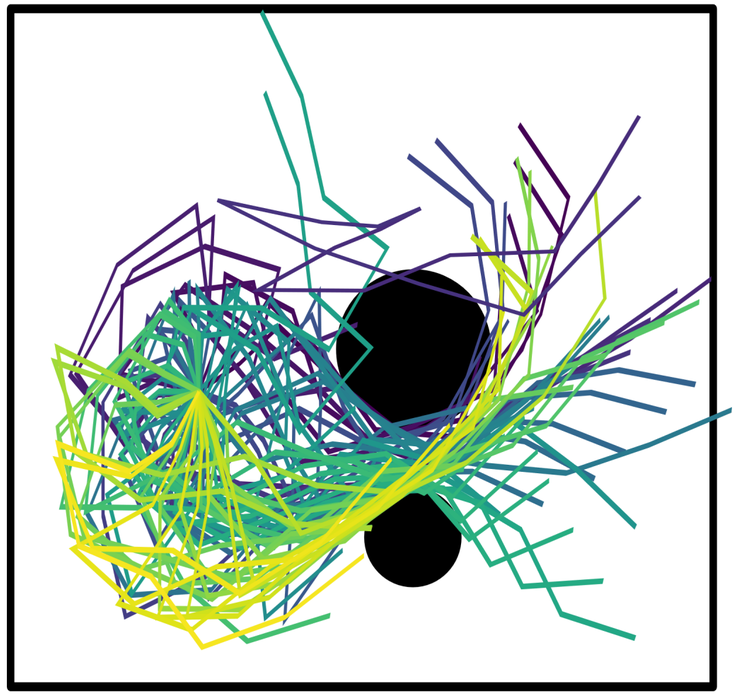
[9] Chamzas et al “Learning from Local Experiences for Efficient Motion Planning”, *ICRA*, 2019

2) Creating Local Samplers

Create a representative set of samples for each local workspace



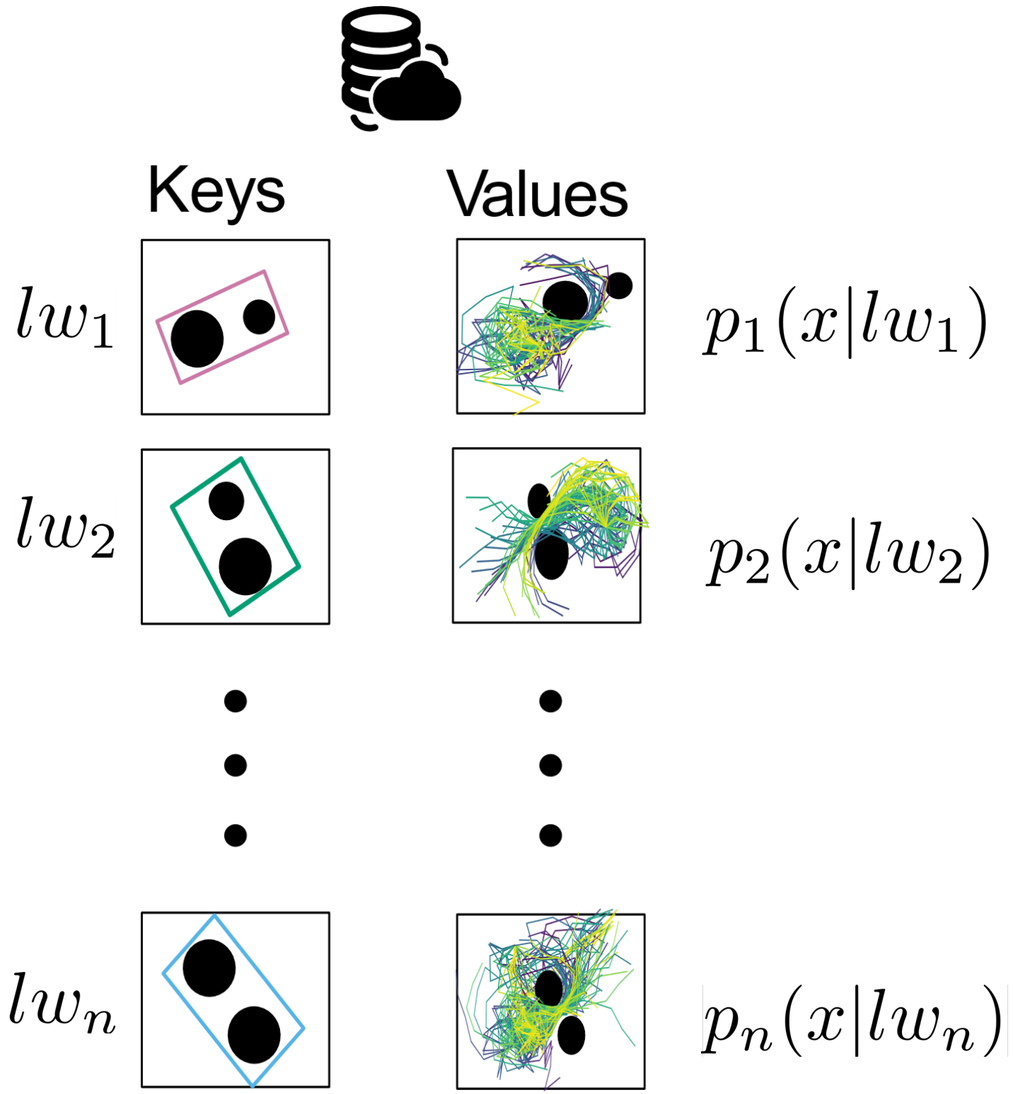
Fit a Gaussian Mixture Model using the samples of the solution



$$p(x|lw) = GMM(\mu, \Sigma) = \frac{1}{M}\sum_{j=1}^{M} N(q_j, \Sigma_j)$$

[9] Chamzas et al "Learning from Local Experiences for Efficient Motion Planning", *ICRA*, 2019

3) Retrieve a Local sampler given local primitive

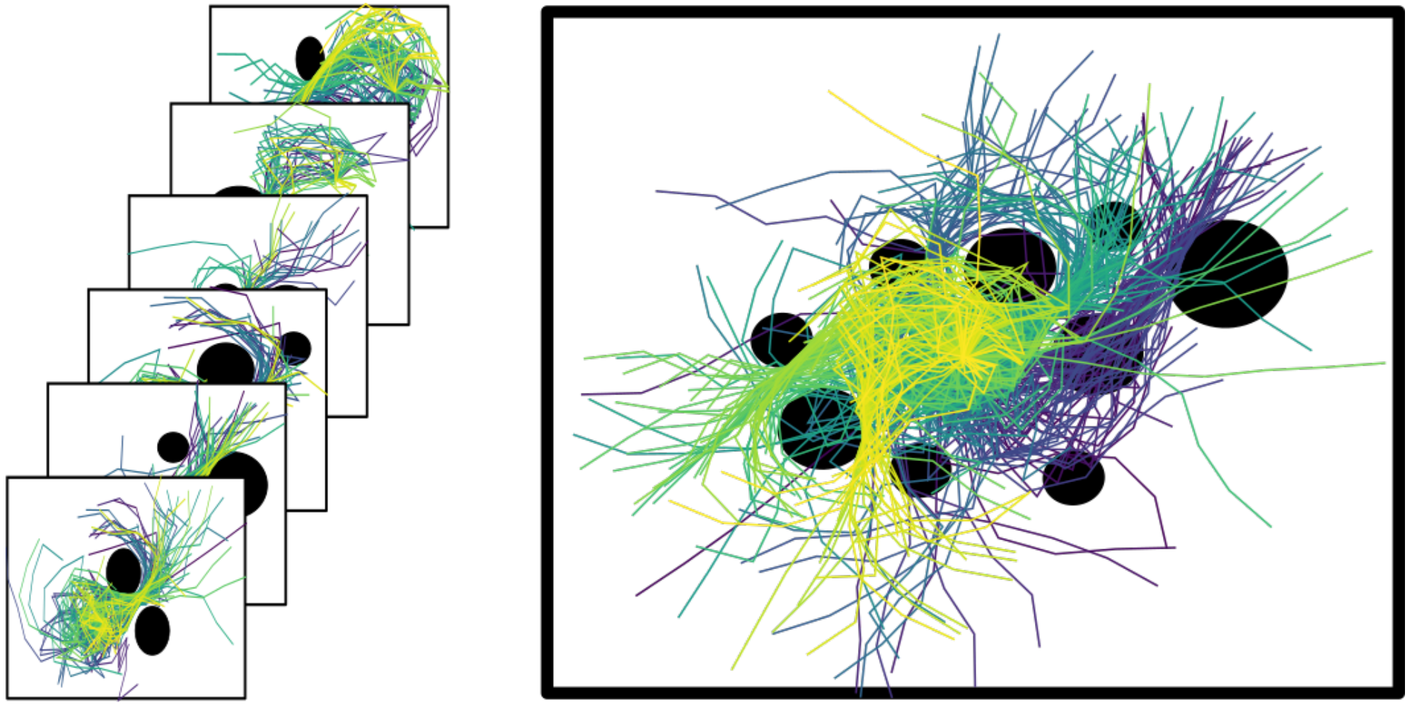


We define a simple similarity function

$$k(lw_i, lw_j) = (lw_i - lw_j)^T (lw_i - lw_j)$$

[9] Chamzas et al "Learning from Local Experiences for Efficient Motion Planning", *ICRA*, 2019

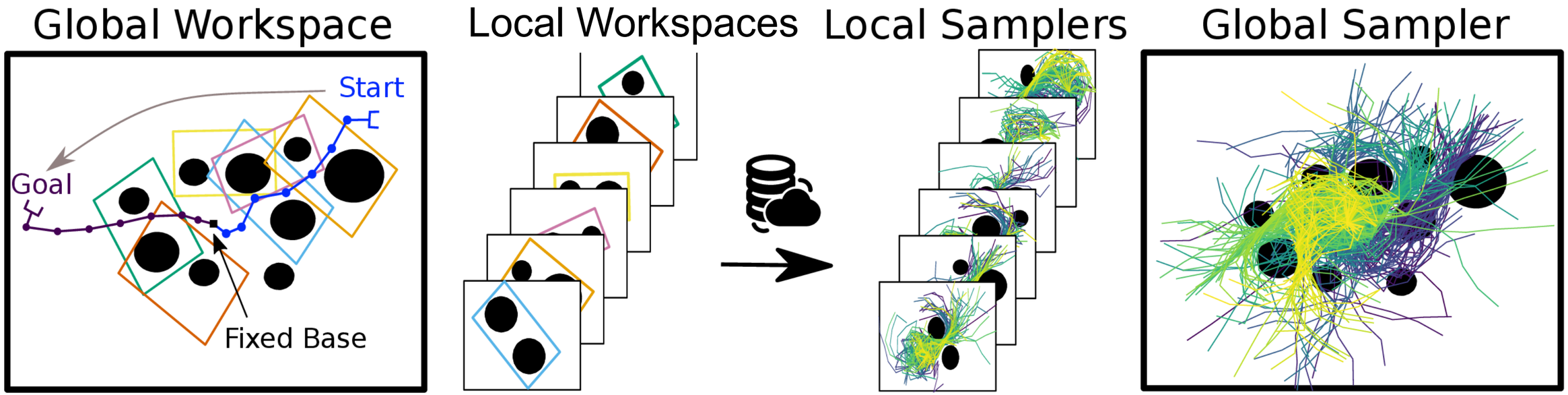
4) Synthesize the local samplers to a global sampler



We simply create a bigger GMM

[9] Chamzas et al “Learning from Local Experiences for Efficient Motion Planning”, *ICRA*, 2019

SPARK2D Framework

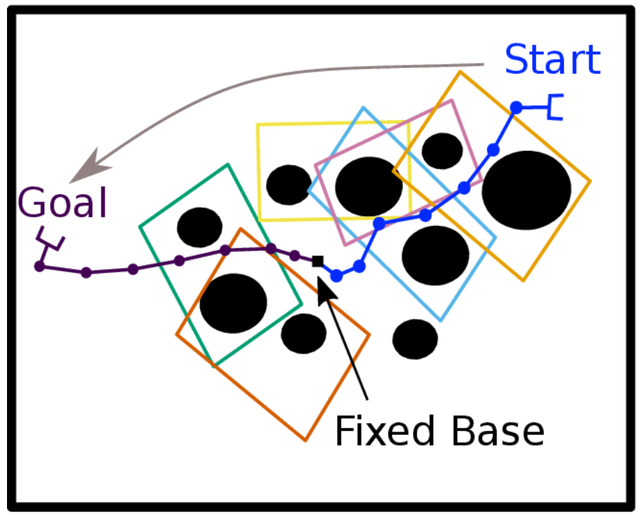


1. Decompose to local primitives
2. Creating a local sampler for every local workspaces [OFFLINE]
3. Retrieve local samplers given the local workspaces [ONLINE]
4. Synthesize a global sampler from the local samplers

[9] Chamzas et al "Learning from Local Experiences for Efficient Motion Planning", *ICRA*, 2019

A neural network baseline

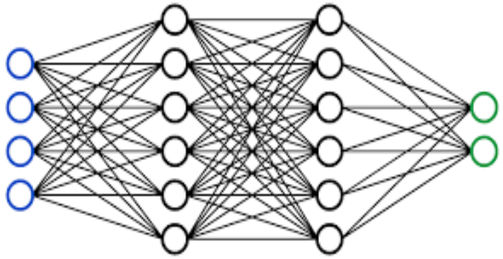
Global Workspace



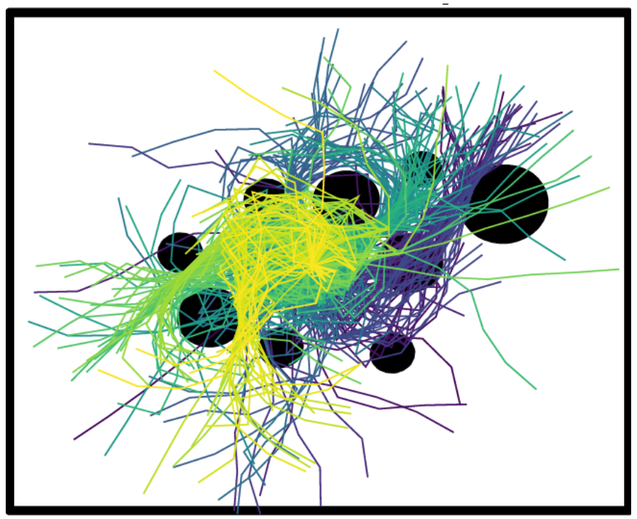
Workspace W

Learning Model

Neural Network (CVAE)



Output

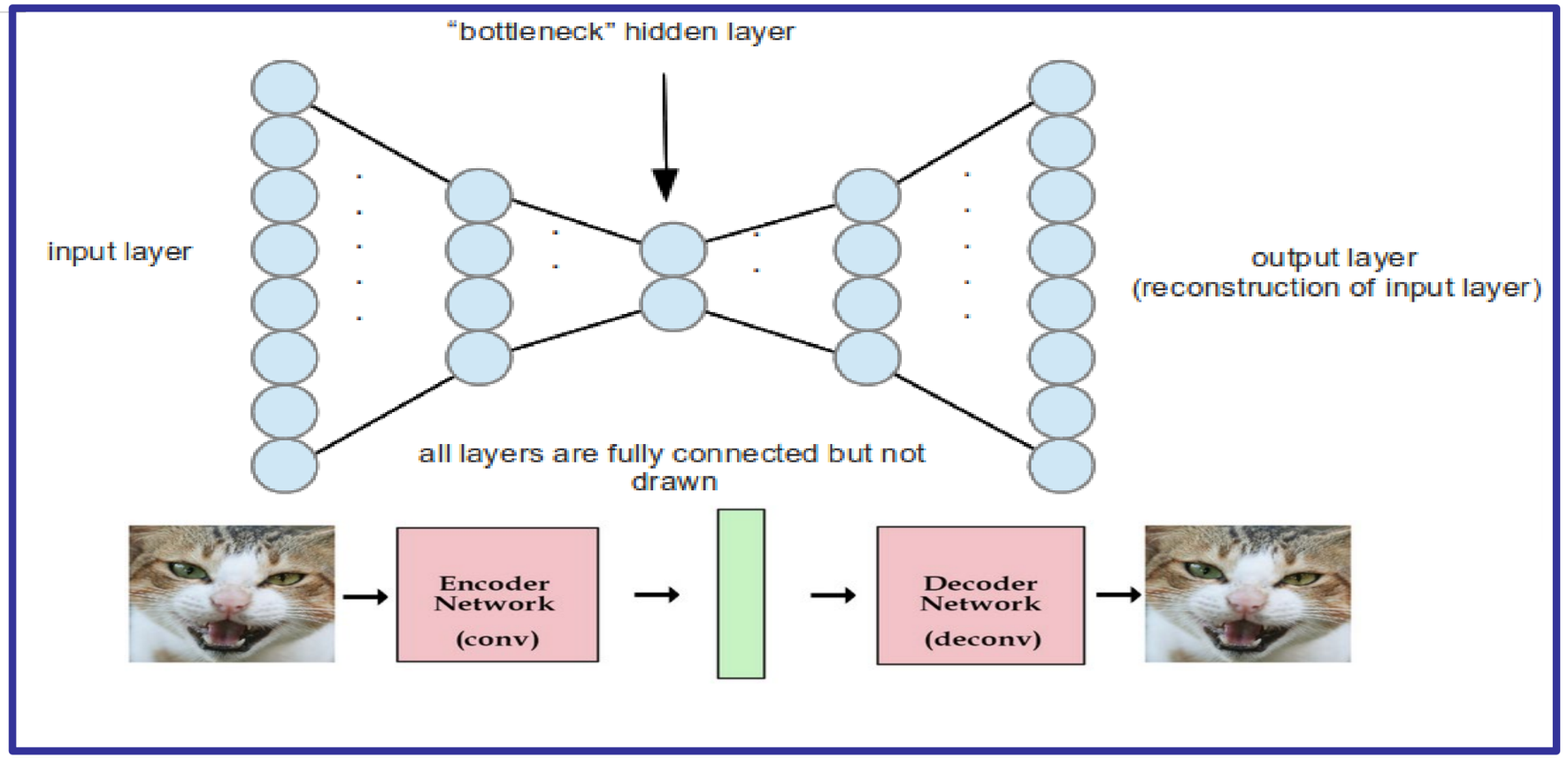


P(X|W)

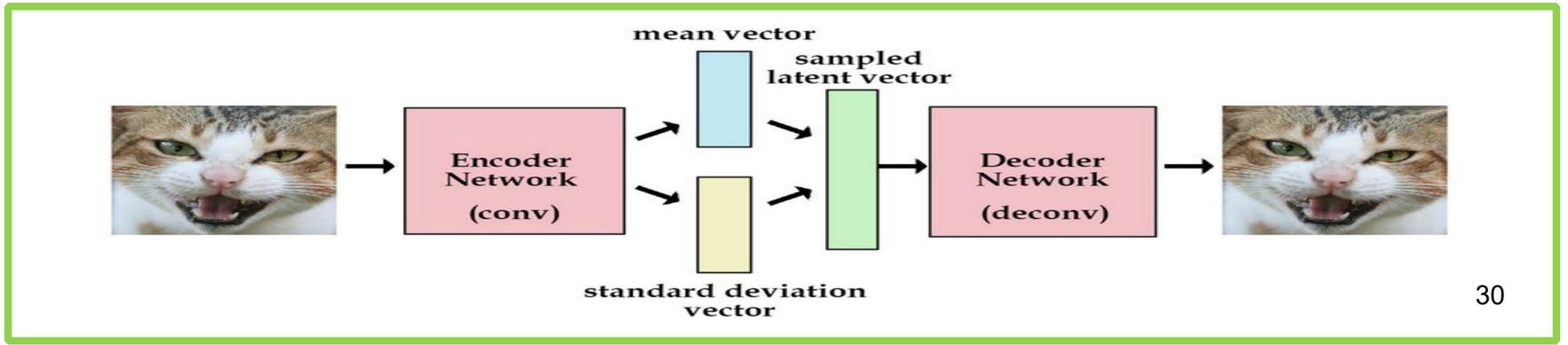
Variational Autoencoders

How to learn a sampling distribution with a neural network?

Autoencoder

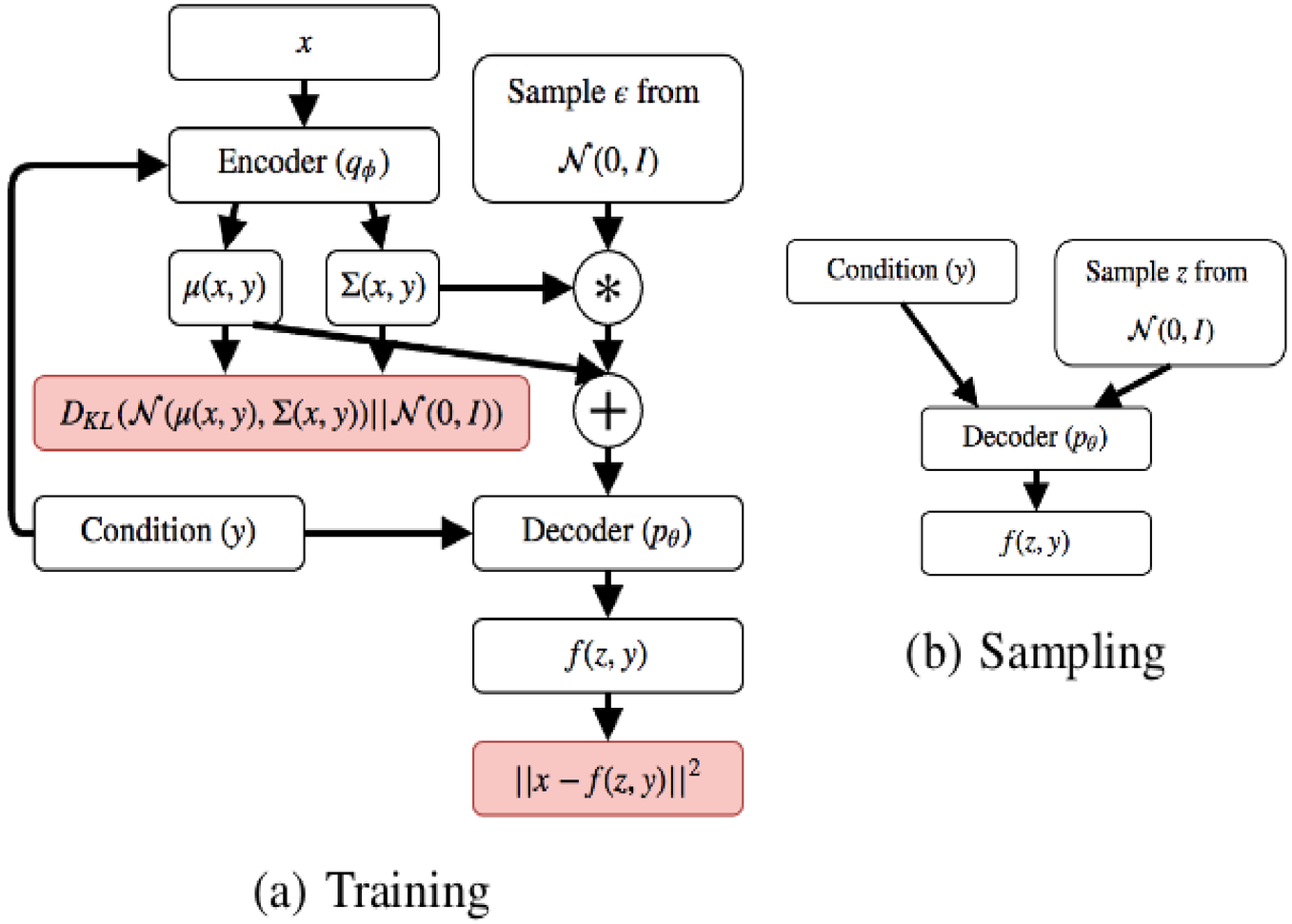


Variational Autoencoder



Conditional Variational Autoencoder (CVAE) [14]

Key Idea :
Lean a sampling distribution from demonstrations and use it to bias sampling of any sampling-based planner



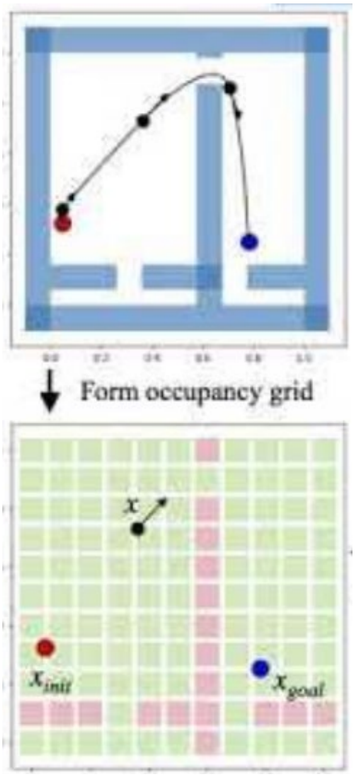
[10] Ichter et al ``Learning sampling distributions for robot motion planning'', *ICRA*, 2018.

A neural network baseline

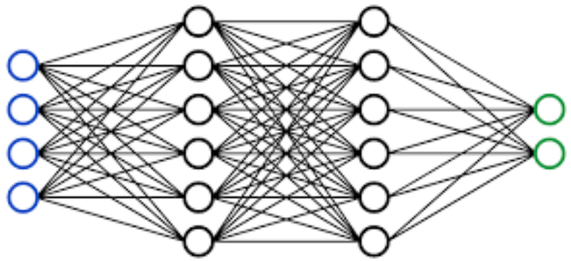
Input

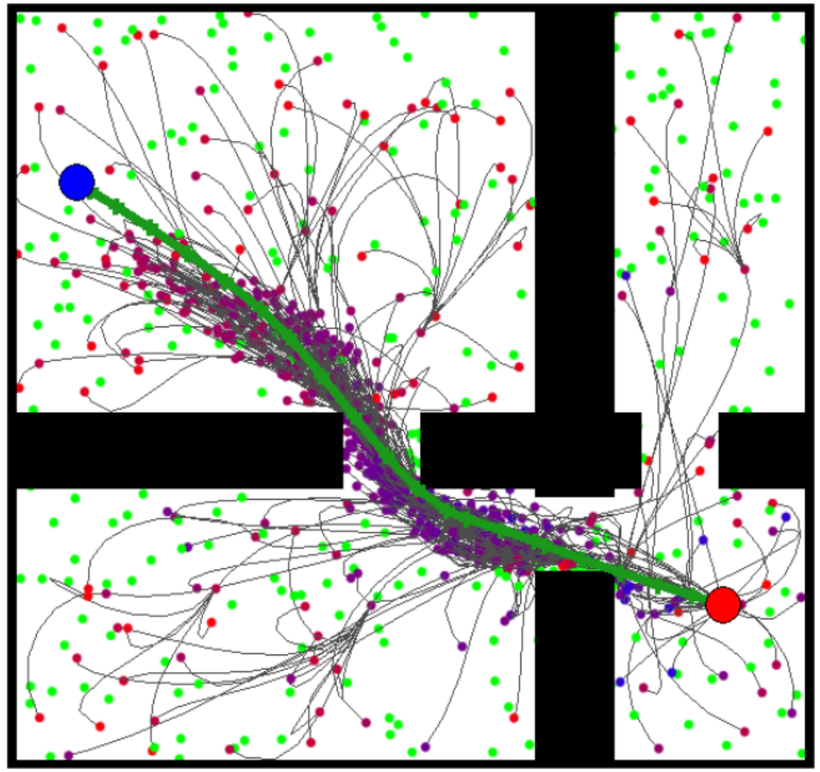
Learning Model

Output



Conditional Variational Autoencoder
(CVAE)





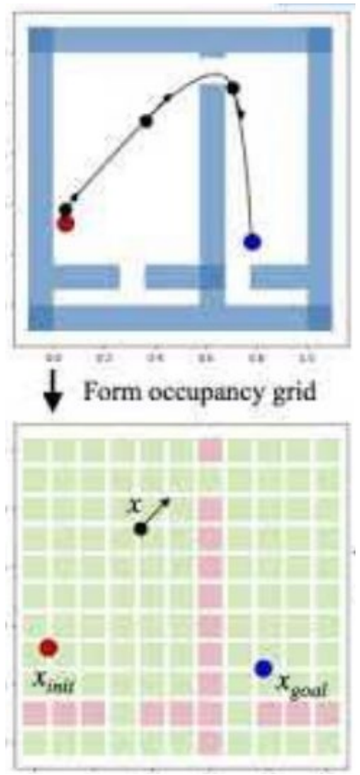
[10] Ichter et al ``Learning sampling distributions for robot motion planning'', *ICRA*, 2018.

A neural network baseline (CVAE)

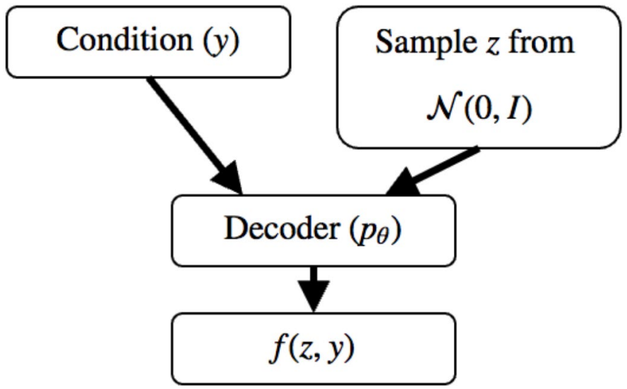
Input

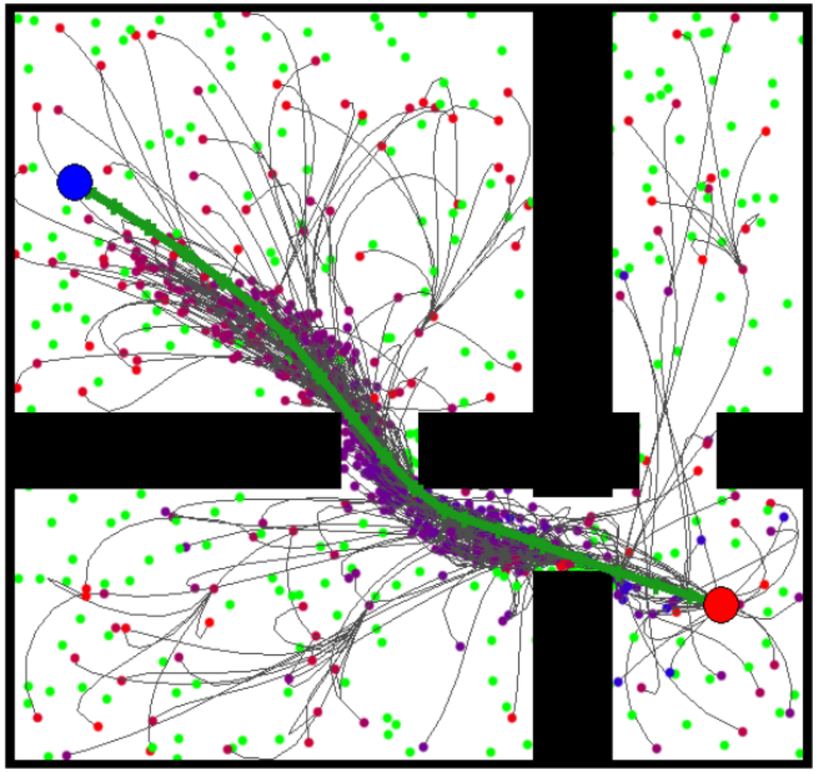
Learning Model

Output



Conditional Variational Autoencoder (CVAE)





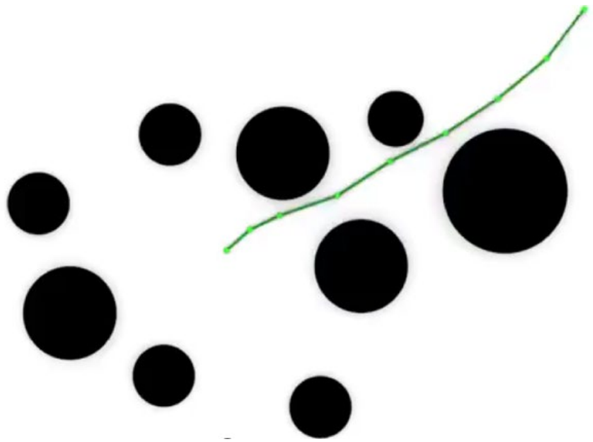
[10] Ichter et al ``Learning sampling distributions for robot motion planning'', *ICRA*, 2018.

A neural network baseline (CVAE)

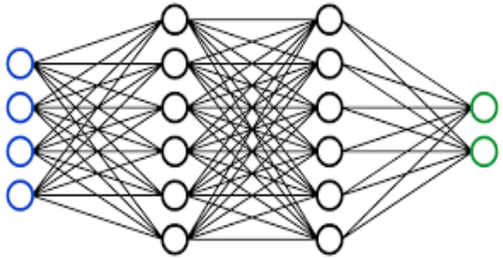
Input

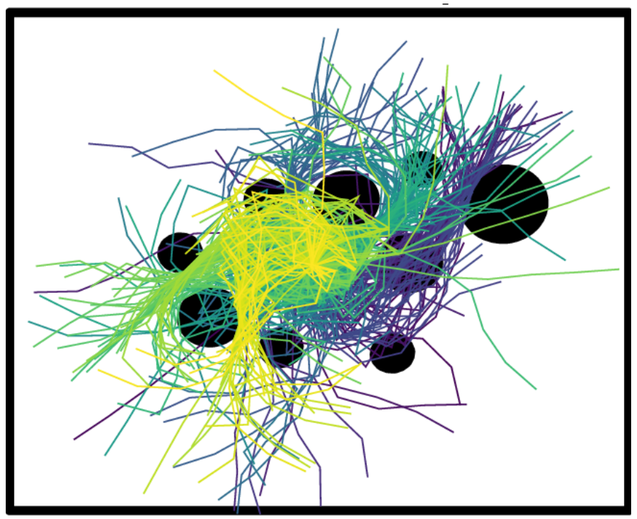
Learning Model

Output



Neural Network (CVAE)





Workspace W

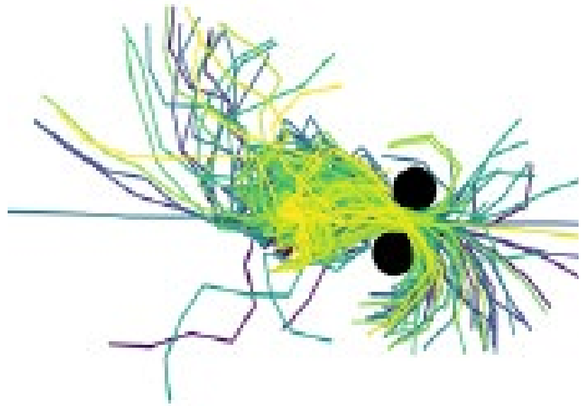
$$Lift : W \rightarrow X$$

P(X|W)

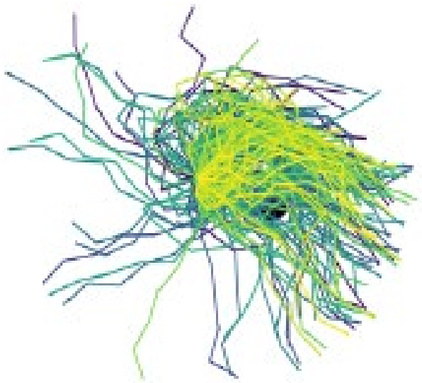
Does not consider discontinuities !

Qualitative comparison SPARK and CVAE

Problem



Model A



Model B

Which ones is SPARK and which one is CVAE

Which model corresponds to which method

Model A - SPARK, Model B - CVAE

Model A - CVAE, Model B - SPARK

Model A - CVAE, Model B - CVAE

Model A -SPARK, Model B - SPARK

Which model corresponds to which method

Model A - SPARK, Model B - CVAE

0%

Model A - CVAE, Model B - SPARK

0%

Model A - CVAE, Model B - CVAE

0%

Model A -SPARK, Model B - SPARK

0%

Which model corresponds to which method

Model A - SPARK, Model B - CVAE

0%

Model A - CVAE, Model B - SPARK

0%

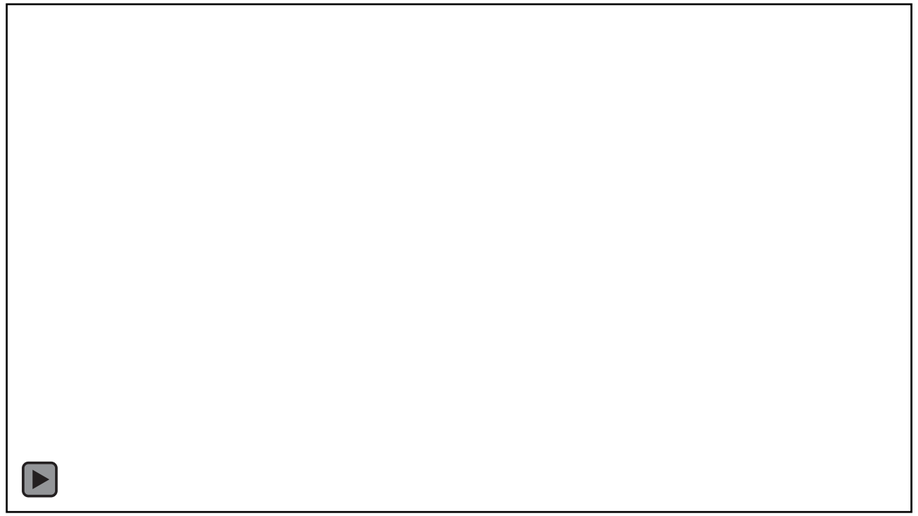
Model A - CVAE, Model B - CVAE

0%

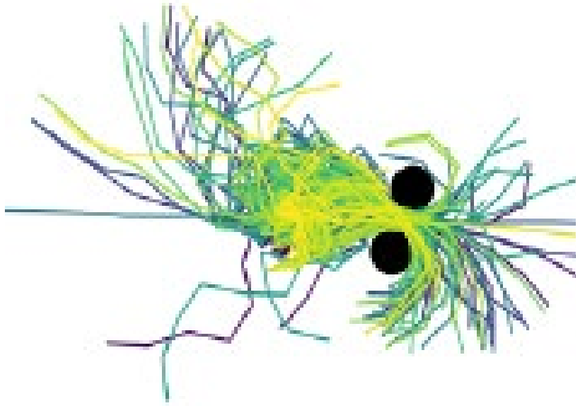
Model A -SPARK, Model B - SPARK

0%

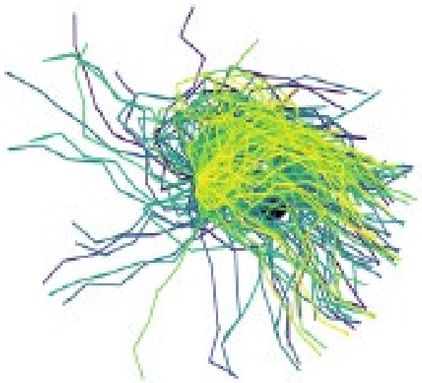
Qualitative comparison SPARK and CVAE



Problem



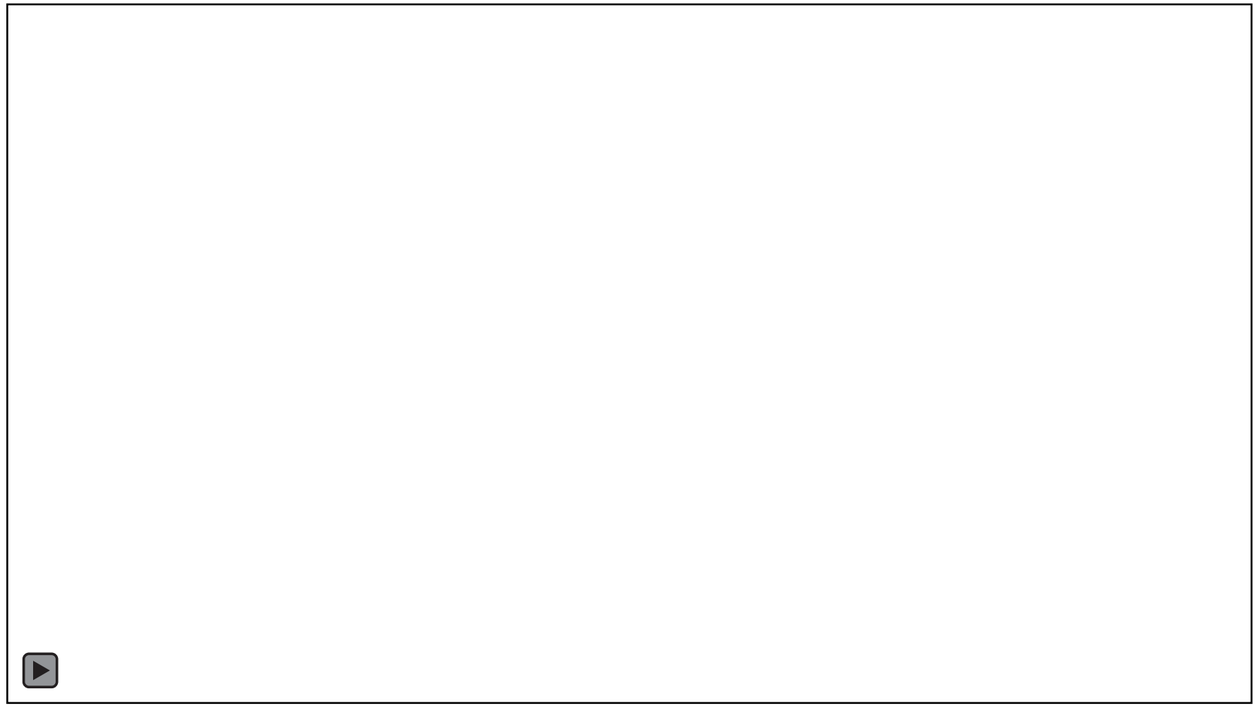
SPARK samples

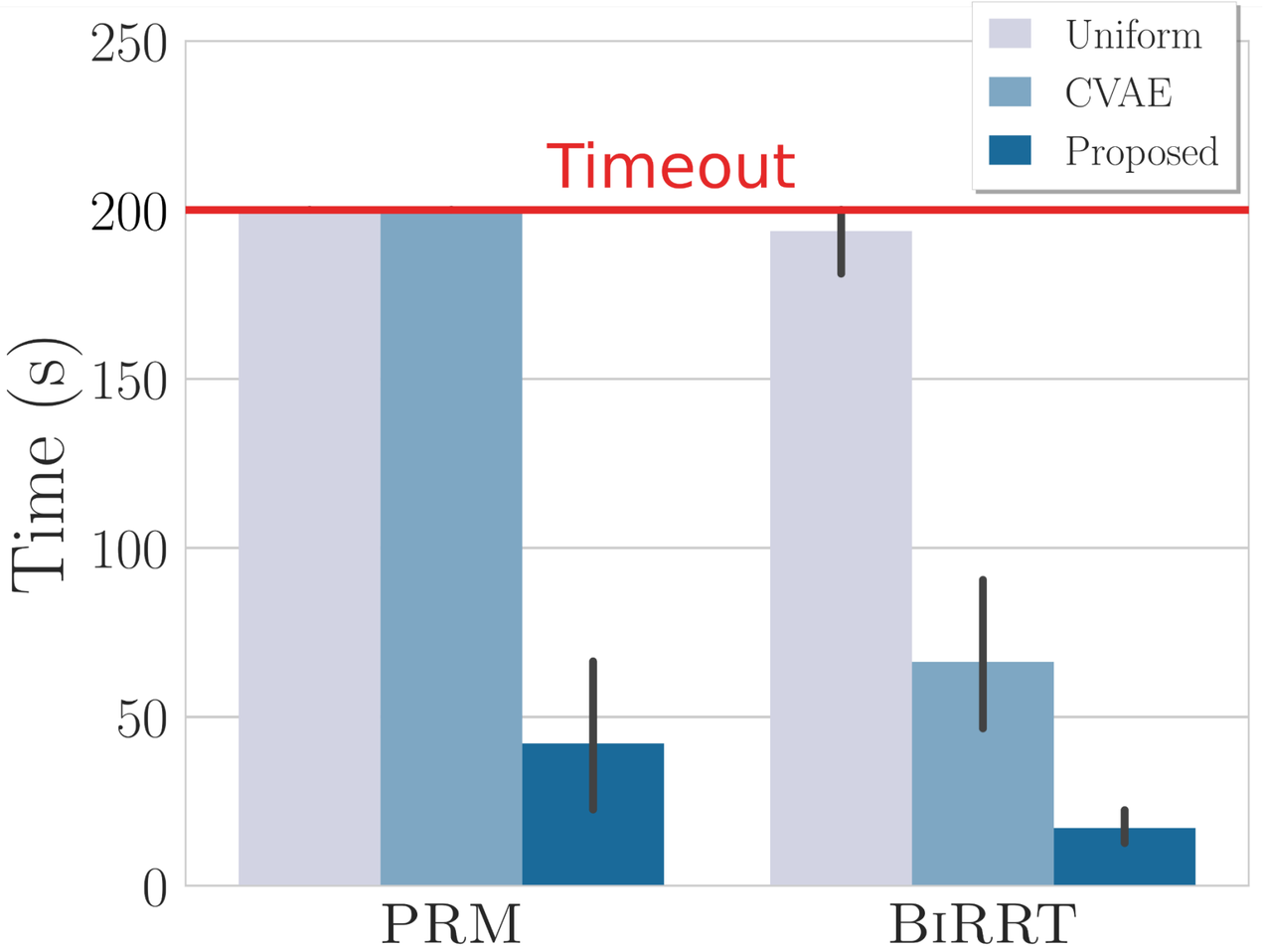


CVAE samples

Discontinuity seems to affect CVAE samples

Quantitative comparison SPARK and CVAE





SPARD2D Overview

Advantages:

- Solved intractable problems
- Decomposition enables generalization with few samples

Disadvantages:

- Tested in a geometrically simple case
- Obstacles known beforehand
- Retrieval was trivial with exhaustive database

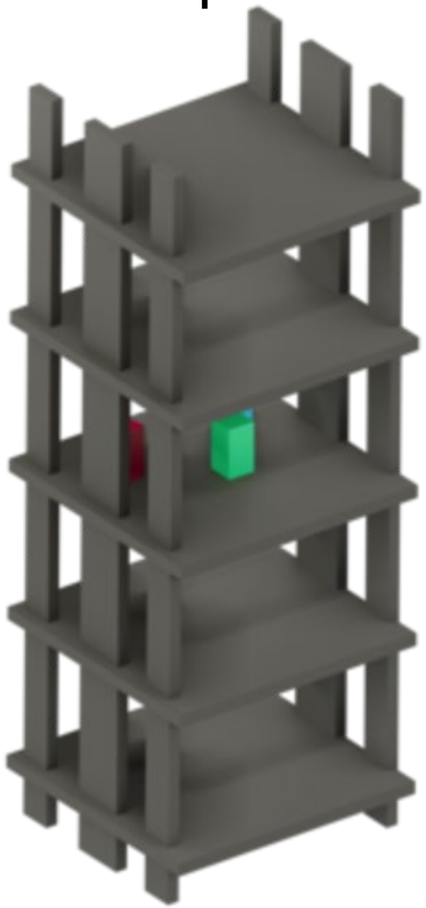
Today’s Agenda

- Important Considerations
  - Locality of narrow passages
  - Discontinuity

- Biased Samplers (Cont.)
  - 2D Workspaces
  - **3D Workspaces**
  - Workplace, Start and Goal

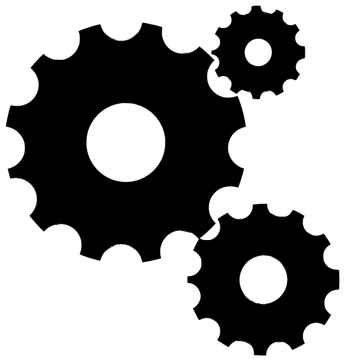
Learn a sampling distribution in 3D workspaces

Input



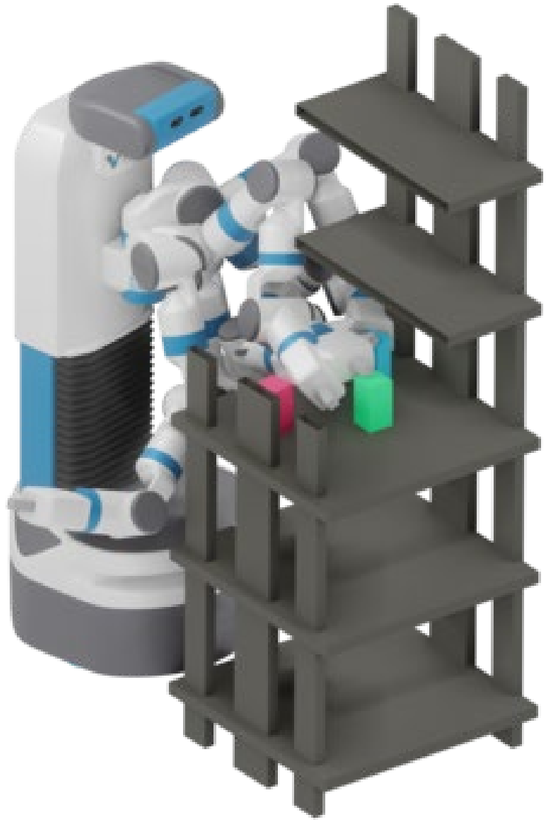
Workspace W

Learning Model



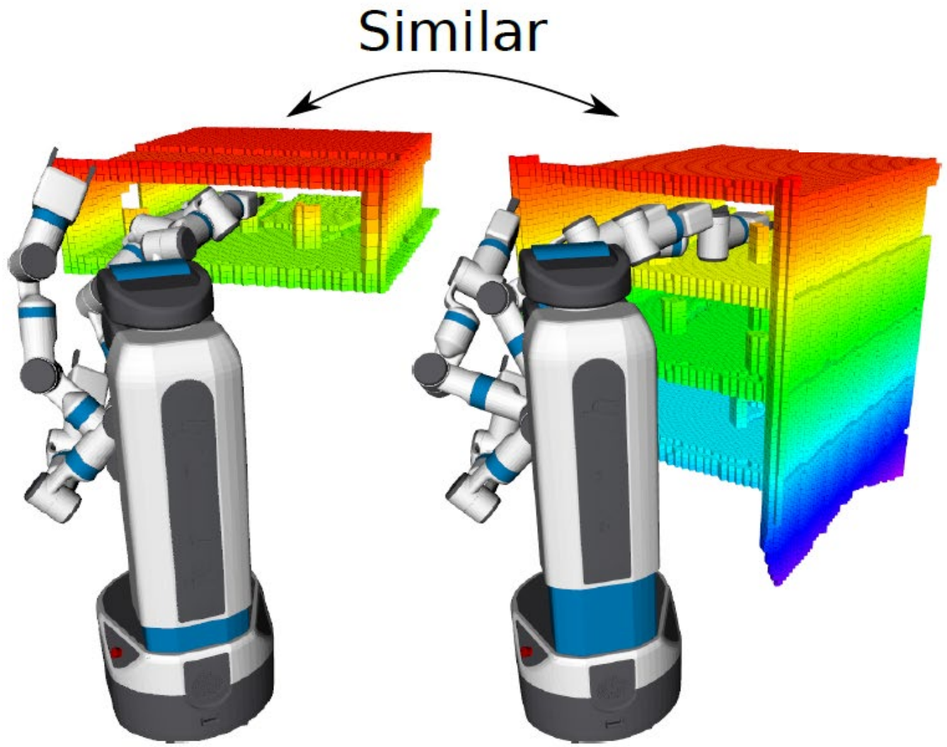
$Map: [W] \rightarrow X$

Output



Sampling Dist. $\mathrm{P}(X|W)$

The Learning Considerations in 3D



Locality (M1&M2):

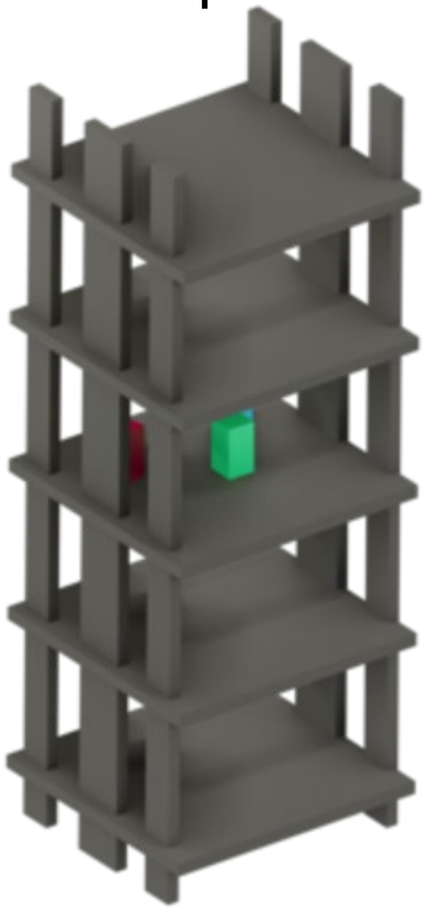
- **similar** path solutions (output)
- workspaces **locally** similar (input)

Discontinuous (M2& M3):

- **dissimilar** solution paths(output)
- very similar (input), due to **small** changes

Use an experience DB to store/retrieve samplers

Input



Workspace W

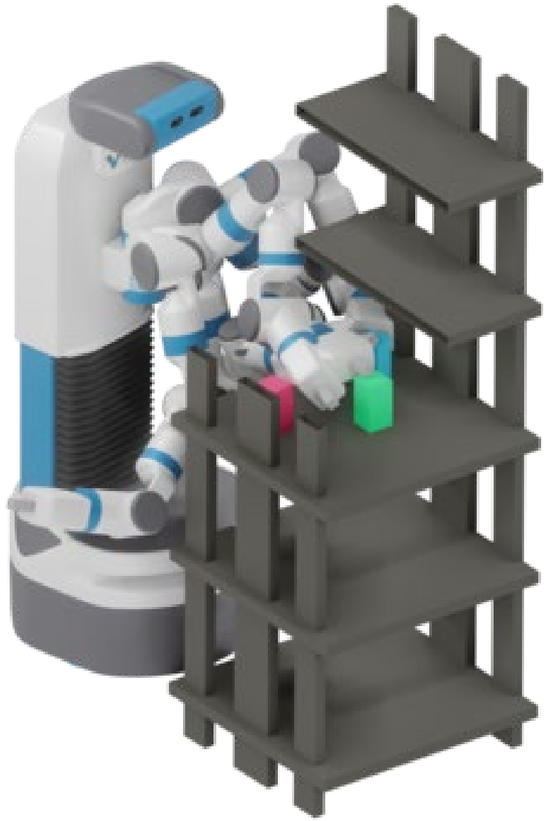
Non-Parametric Model

Database/Library of Experiences

$d(W', W) \rightarrow [0, 1]$

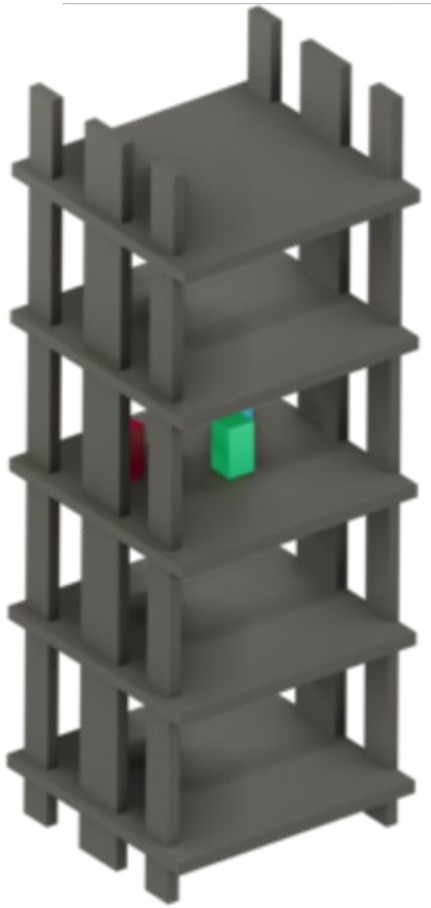
$W' : P'(X)$

Output



Sampling Dist. $\mathrm{P}(X|W)$

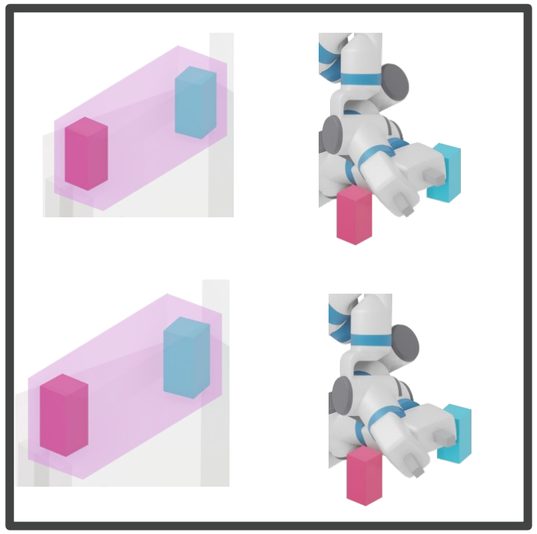
Proposed framework pipeline



Workspace W

1. Decompose

2. Create

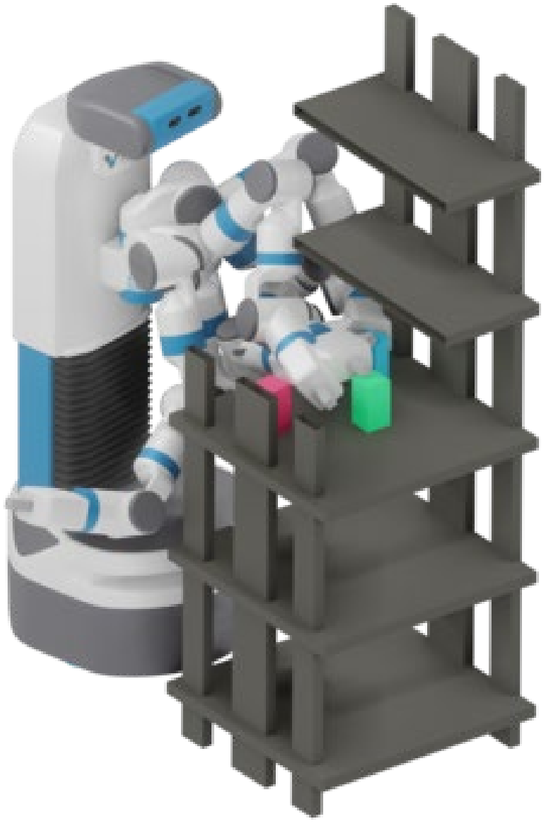


Local workspaces

Local samplers

3. Retrieve

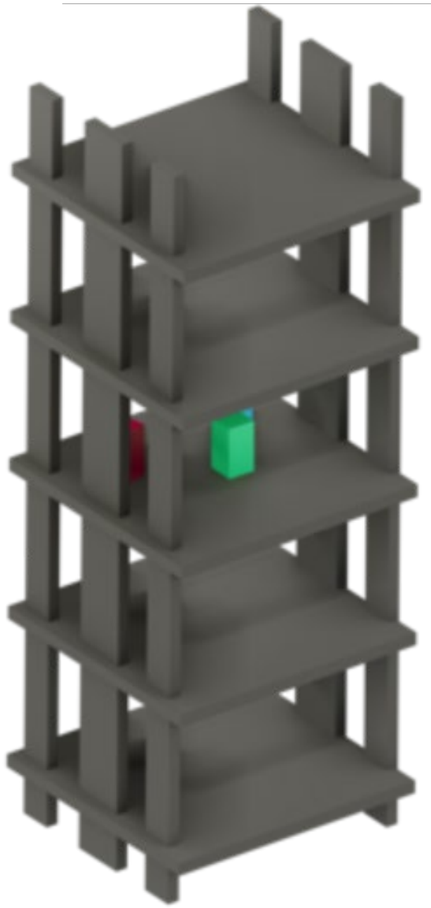
4. Synthesize



Sampling Dist. $\mathrm{P}(X|W)$

[11] Chamzas et al "Learning Sampling Distributions Using Local 3D Workspace Decompositions for Motion Planning in High Dimensions", *ICRA*, 2021

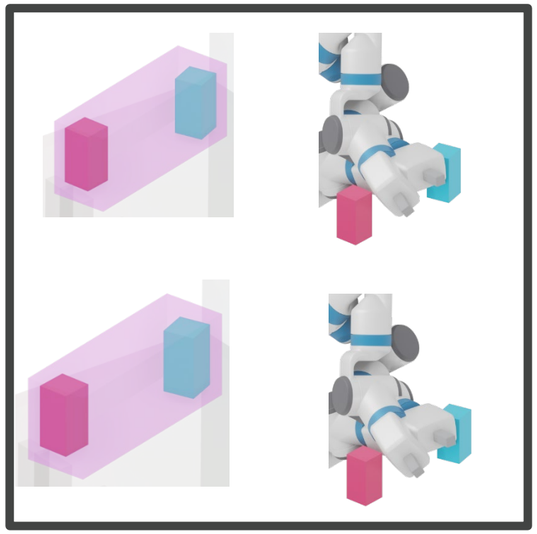
Proposed framework pipeline



Workspace W

1. Decompose

2. Create

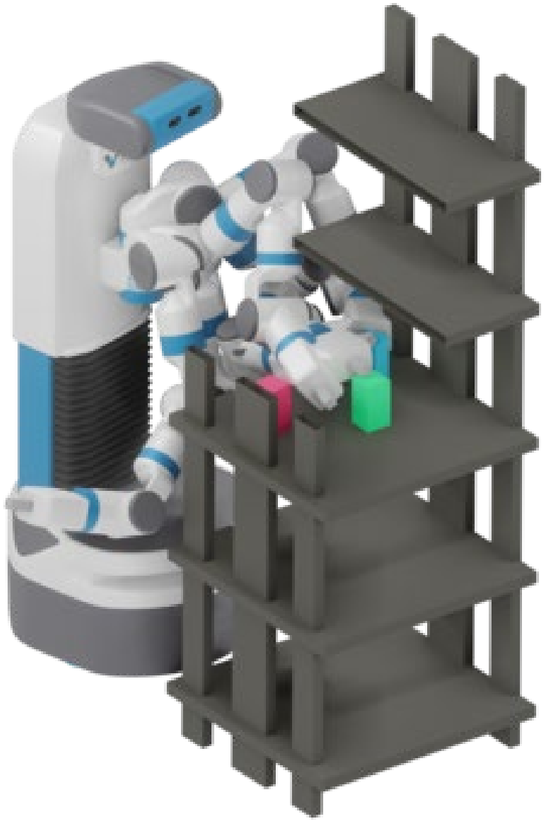


Local workspaces

Local samplers

3. Retrieve

4. Synthesize



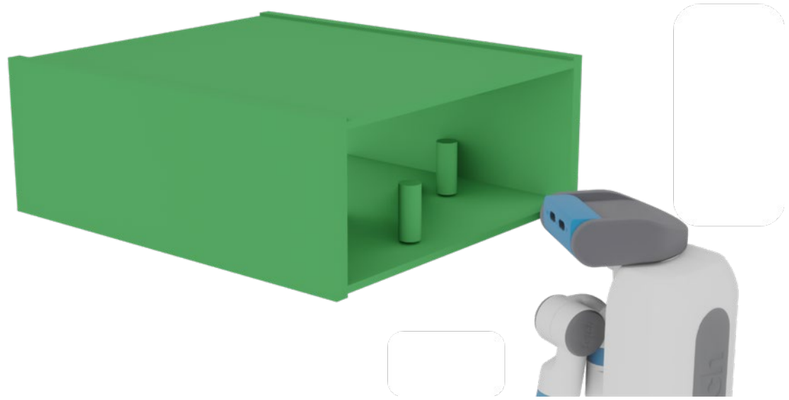
Sampling Dist. $\mathrm{P}(X|W)$

Learning (Training)

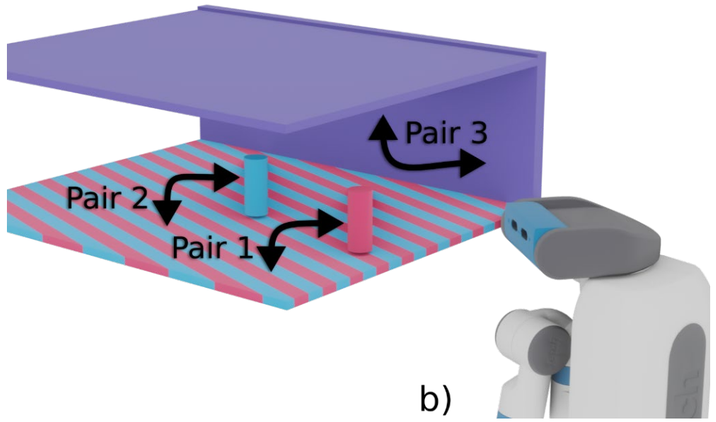
Inference (Testing)

[11] Chamzas et al "Learning Sampling Distributions Using Local 3D Workspace Decompositions for Motion Planning in High Dimensions", *ICRA*, 2021

1) Decompose to Local Workspaces (Learning & Inference)

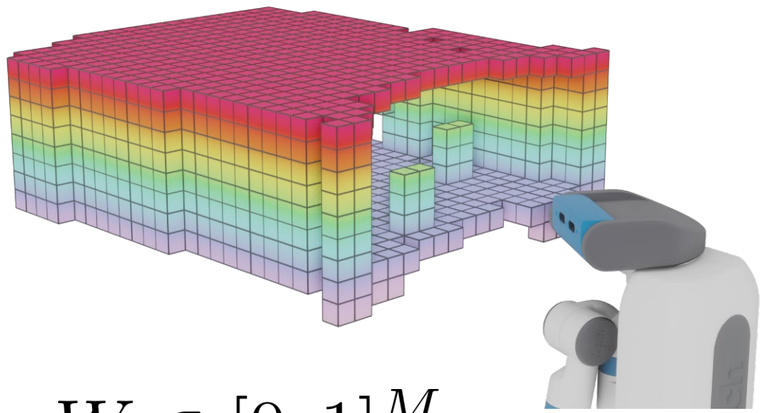


Geometric Decomposition

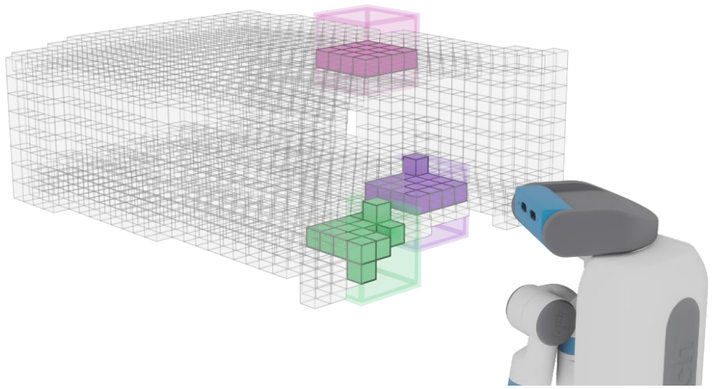


$$W = \{o : o = (pose, geom)\}$$

$$lw = \{x_a, q_a, s_a, x_b, q_b, s_b\}$$



Sensed Decomposition

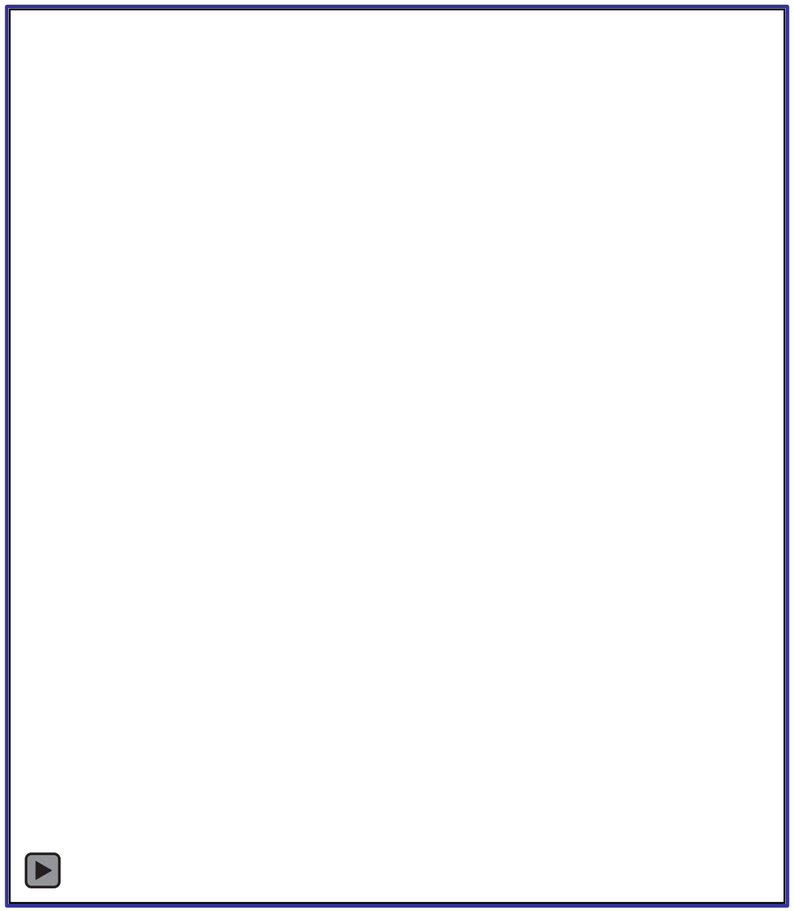


$$W \in [0,1]^M$$

$$lw = \{x_i, b_i\}, b_i \in [0,1]^{64}$$

[11] Chamzas et al "Learning Sampling Distributions Using Local 3D Workspace Decompositions for Motion Planning in High Dimensions", *ICRA*, 2021

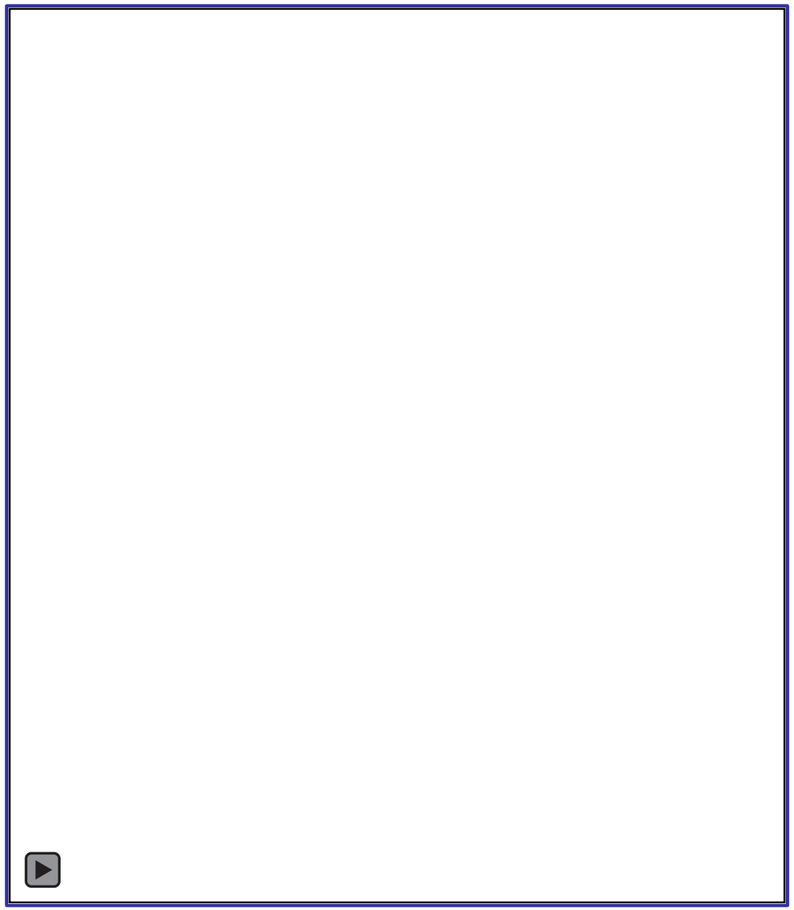
2) Create local workspaces/samplers (Learning)



A workspace and a corresponding solution path

[11] Chamzas et al “Learning Sampling Distributions Using Local 3D Workspace Decompositions for Motion Planning in High Dimensions”, *ICRA*, 2021

2) Create local workspaces/samplers (Learning)

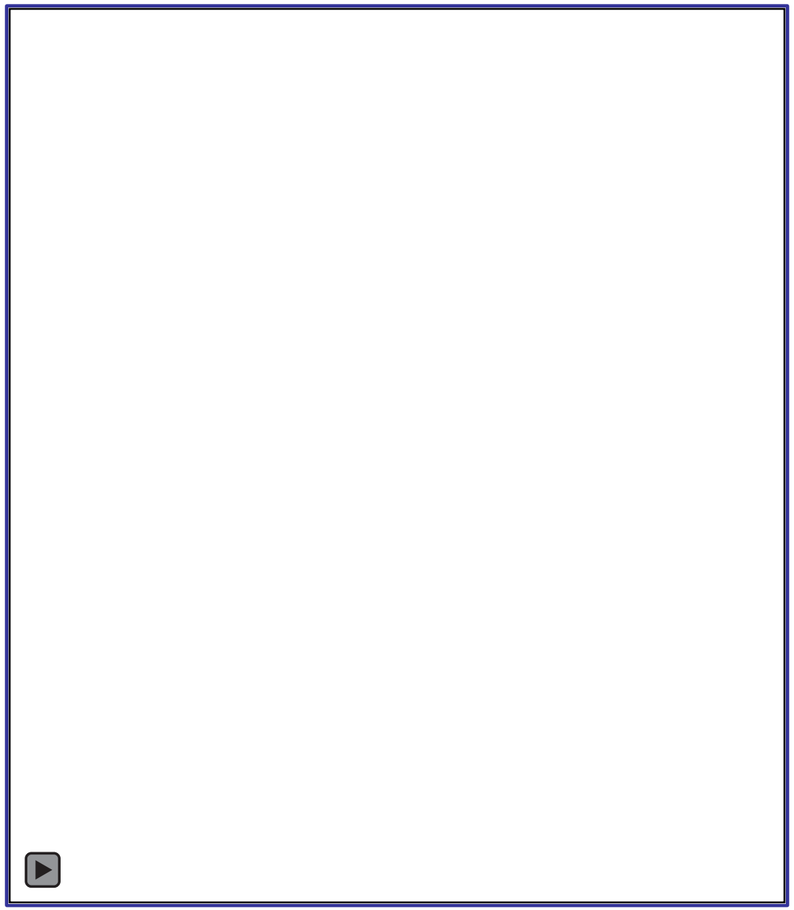


Critical samples in geometric workspace

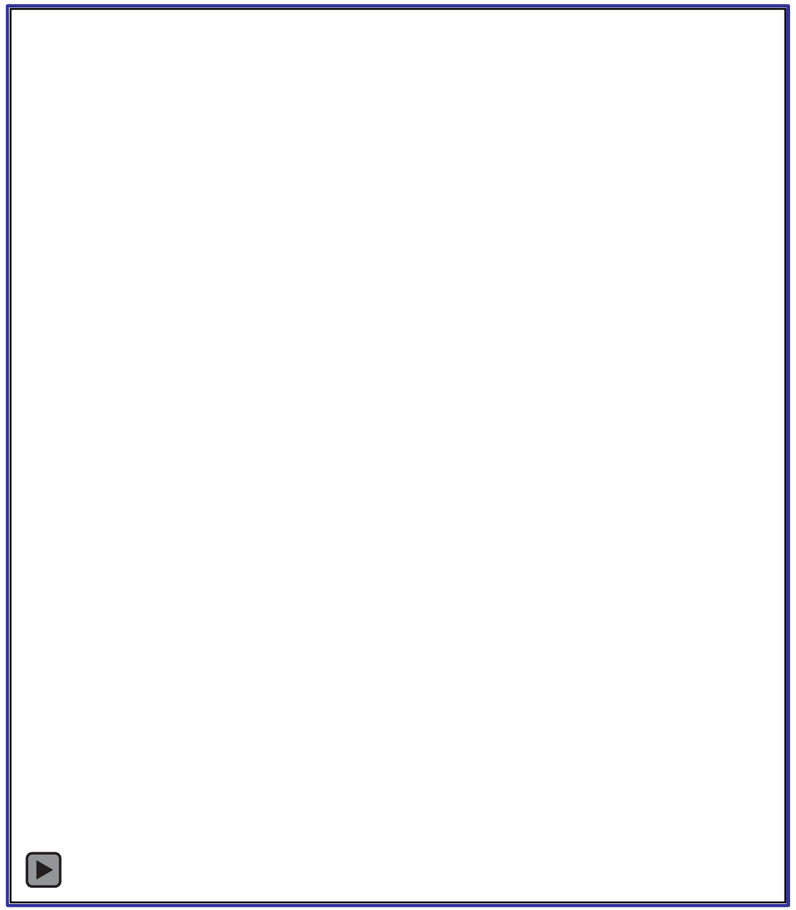
a) Associate critical samples with local workspaces

[11] Chamzas et al “Learning Sampling Distributions Using Local 3D Workspace Decompositions for Motion Planning in High Dimensions”, *ICRA*, 2021

2) Create local workspaces/samplers (Learning)



Critical samples in geometric workspace

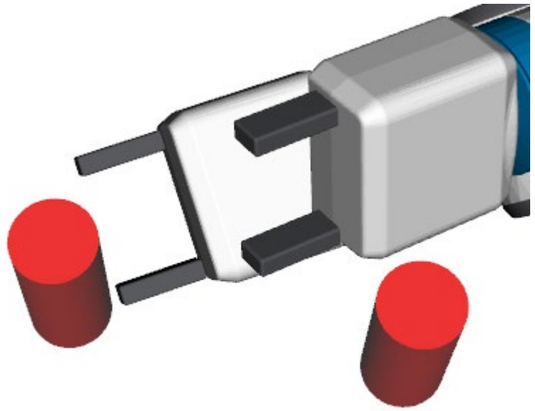


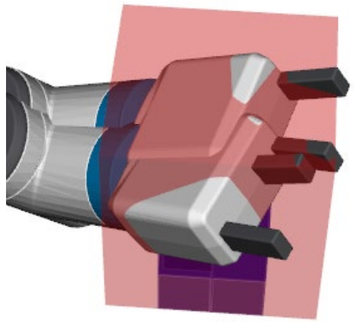
Critical samples in sensed workspace

a) Associate critical samples with local workspaces

[11] Chamzas et al "Learning Sampling Distributions Using Local 3D Workspace Decompositions for Motion Planning in High Dimensions", *ICRA*, 2021

2) Create local workspaces/samplers (Learning)





Local workspaces (lw) are shown in red with their associated critical samples (x)

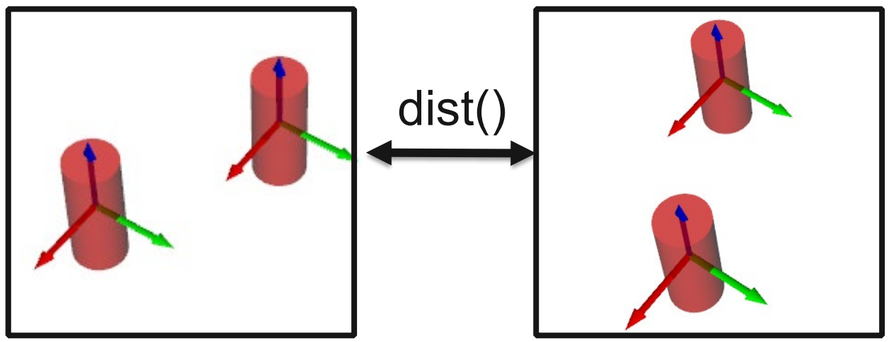
a) Associate critical samples with local workspaces

$$p(x|lw) = GMM(\mu, \Sigma) = \frac{1}{M}\sum_{j=1}^{M} N(q_j, \Sigma_j)$$

b) Fit critical samples with a GMM for each local workspace

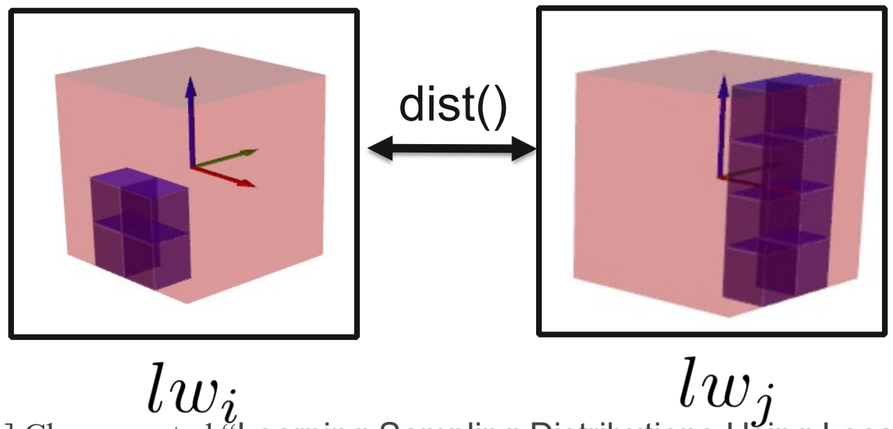
[11] Chamzas et al "Learning Sampling Distributions Using Local 3D Workspace Decompositions for Motion Planning in High Dimensions", *ICRA*, 2021

3) Retrieve local samplers (Inference)



$$d_o(o_i, o_j) = w_x \|x_i - x_j\| + w_r(1 - \langle q_i, q_j \rangle) + w_s \|s_i - s_j\|$$

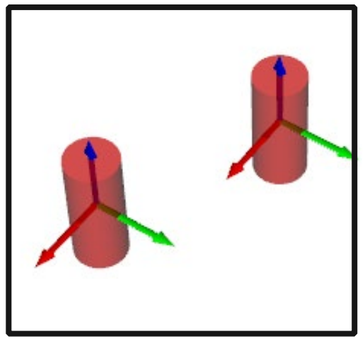
$$d(lw_i, lw_j) = \min \begin{cases} d_o(o_i^a, o_j^a) + d_o(o_i^b, o_j^b) \\ d_o(o_i^a, o_j^b) + d_o(o_i^b, o_j^a) \end{cases}$$



$$d(lw_i, lw_j) = \begin{cases} 0 & x_i = x_j, b_i = b_j, \\ \infty & \text{otherwise} \end{cases}$$

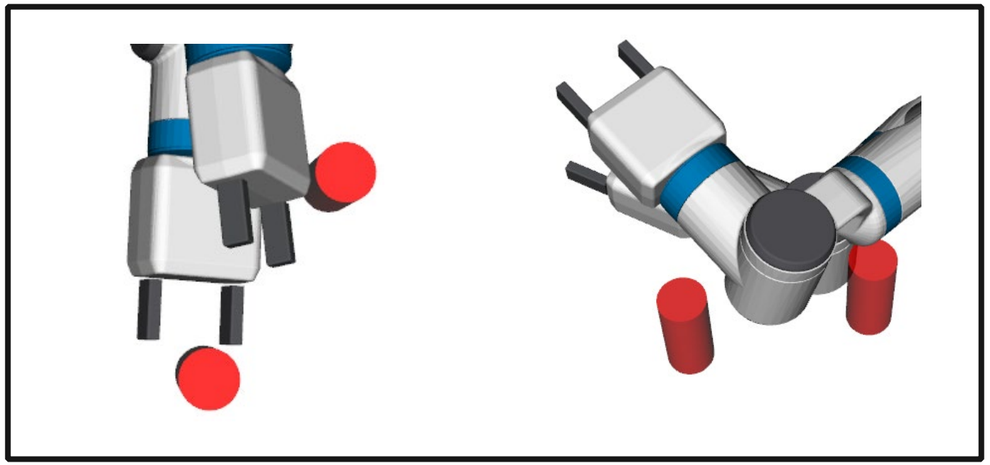
[11] Chamzas et al "Learning Sampling Distributions Using Local 3D Workspace Decompositions for Motion Planning in High Dimensions", *ICRA*, 2021

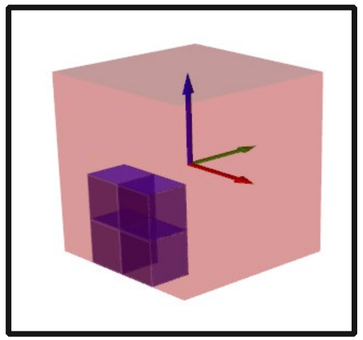
3) Retrieve local samplers (Inference)



Fixed-radius NN search

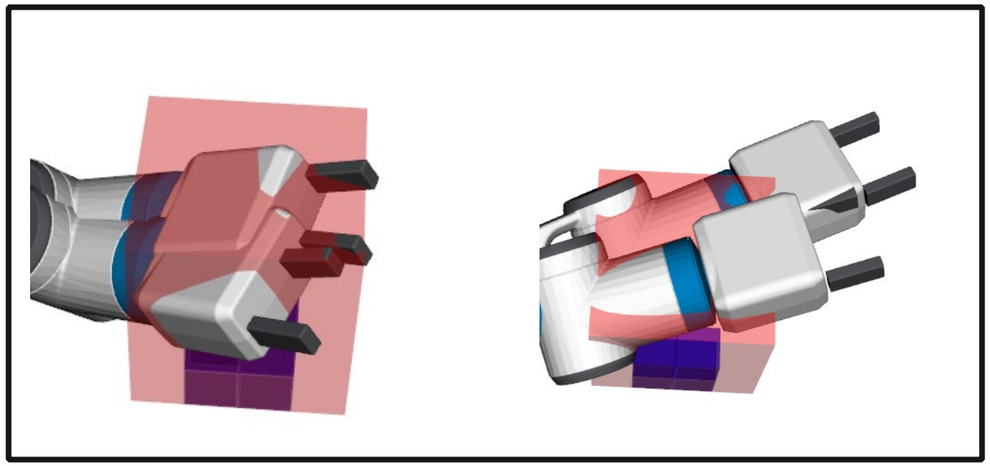
Retrieve





Fixed-radius NN search

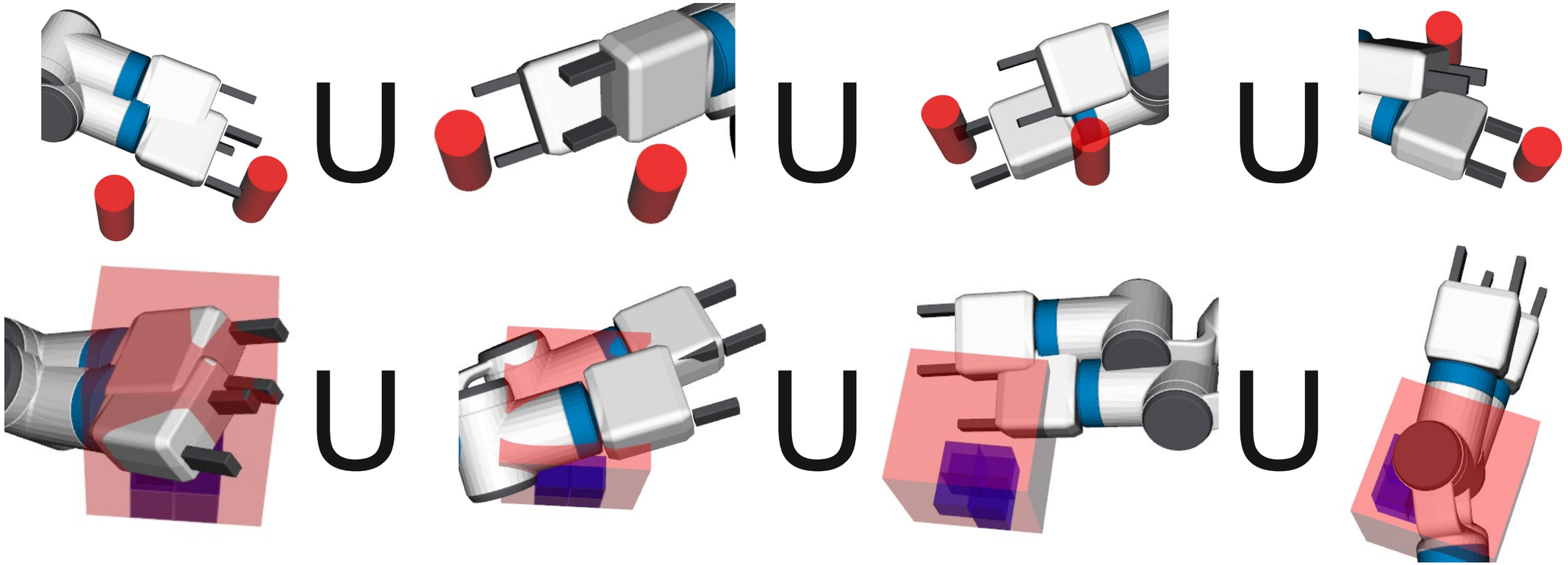
Retrieve



We used the GNAT structure which has logarithmic retrieval time

[11] Chamzas et al "Learning Sampling Distributions Using Local 3D Workspace Decompositions for Motion Planning in High Dimensions", *ICRA*, 2021

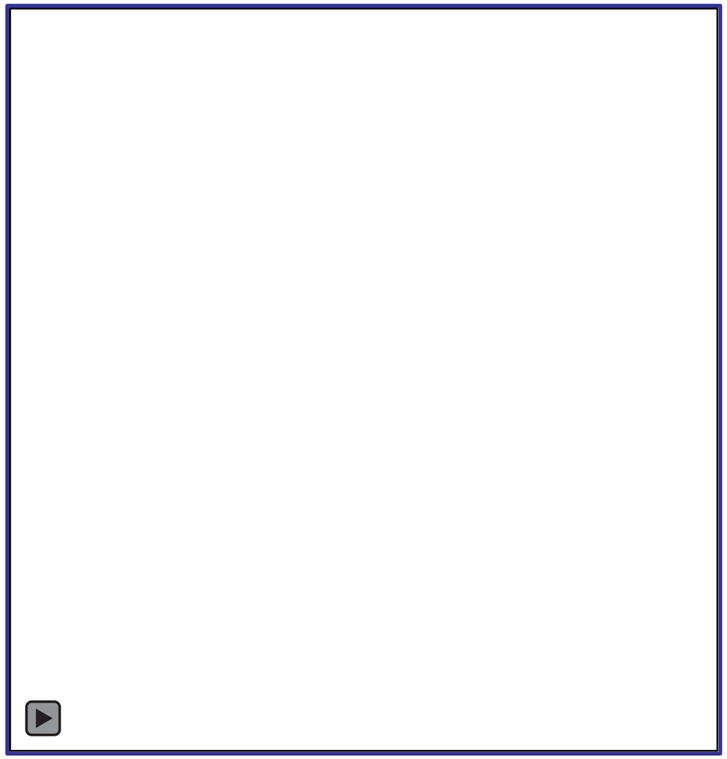
4) Synthesize local samplers (Testing)

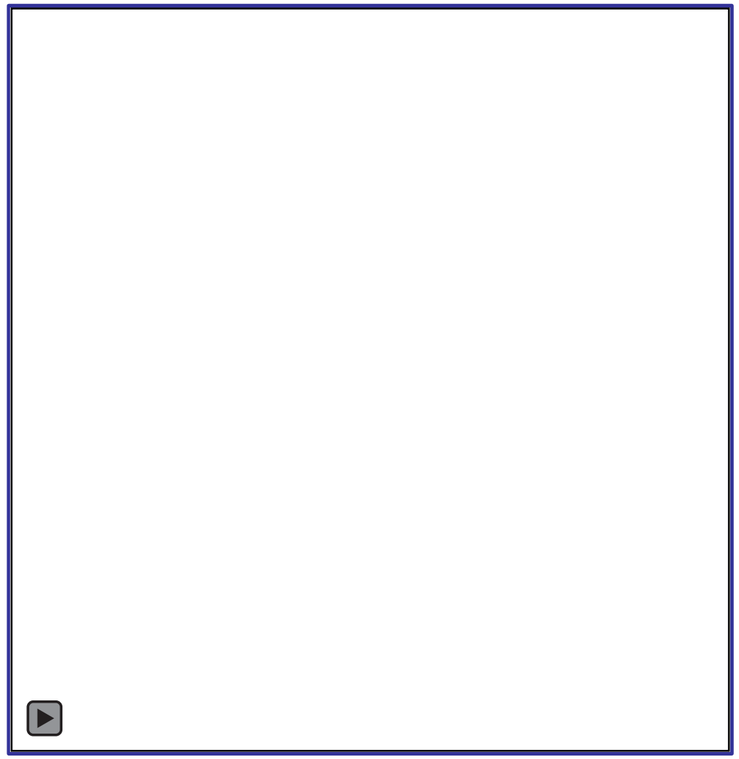


$$p(x|W) = \frac{1}{K}\sum_{i=1}^{K}\frac{1}{M_i}\sum_{j=1}^{M_i}\mathcal{N}(q_{ij}, \Sigma_{ij})$$

[11] Chamzas et al "Learning Sampling Distributions Using Local 3D Workspace Decompositions for Motion Planning in High Dimensions", *ICRA*, 2021

Inference with proposed methods SPARK & FLAME





Retrieved local workspaces are shown in **red**

SPARK works with geometrically known environments

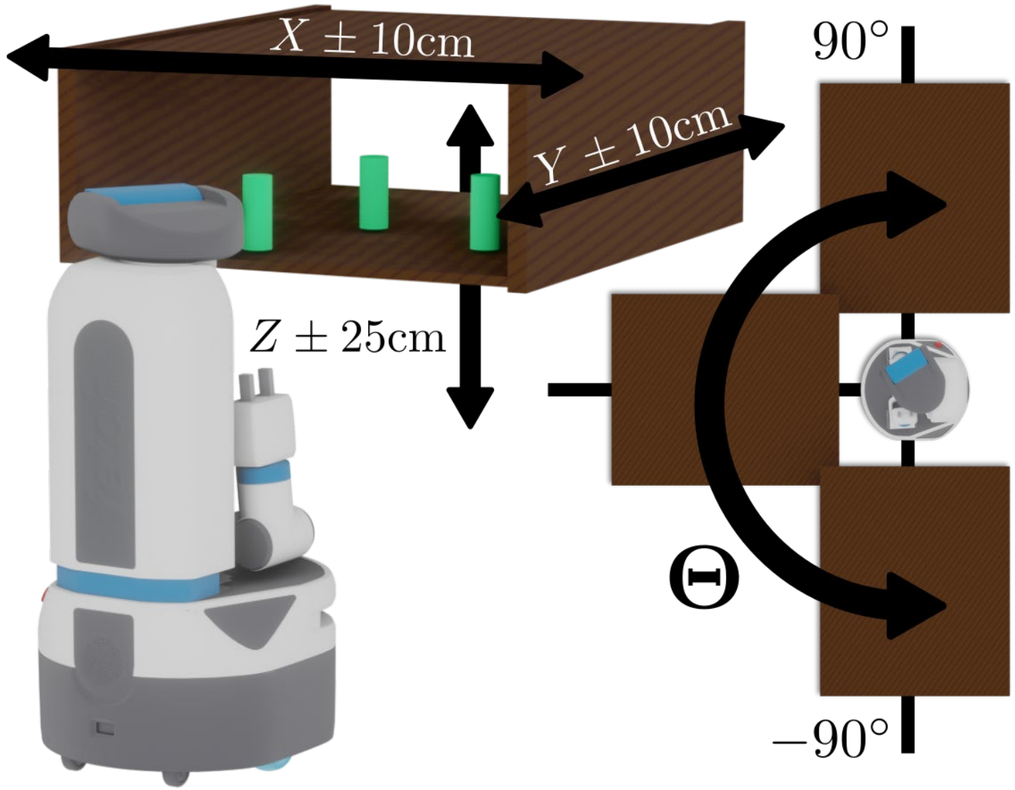
FLAME works with sensed environments

[11] Chamzas et al "Learning Sampling Distributions Using Local 3D Workspace Decompositions for Motion Planning in High Dimensions", *ICRA*, 2021

Experiments

1. **Robustness to environment variation**
2. Generalization to different tasks
3. Real robot application

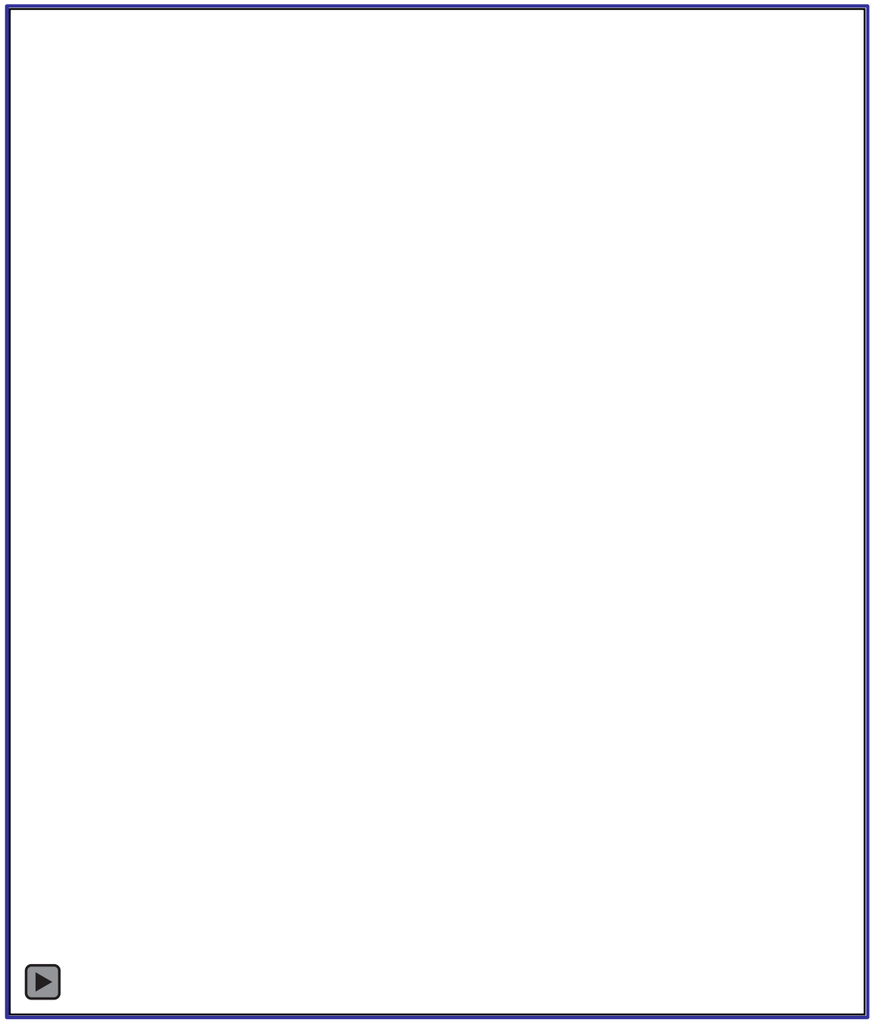
SPARK/FLAME are robust to workspace variations



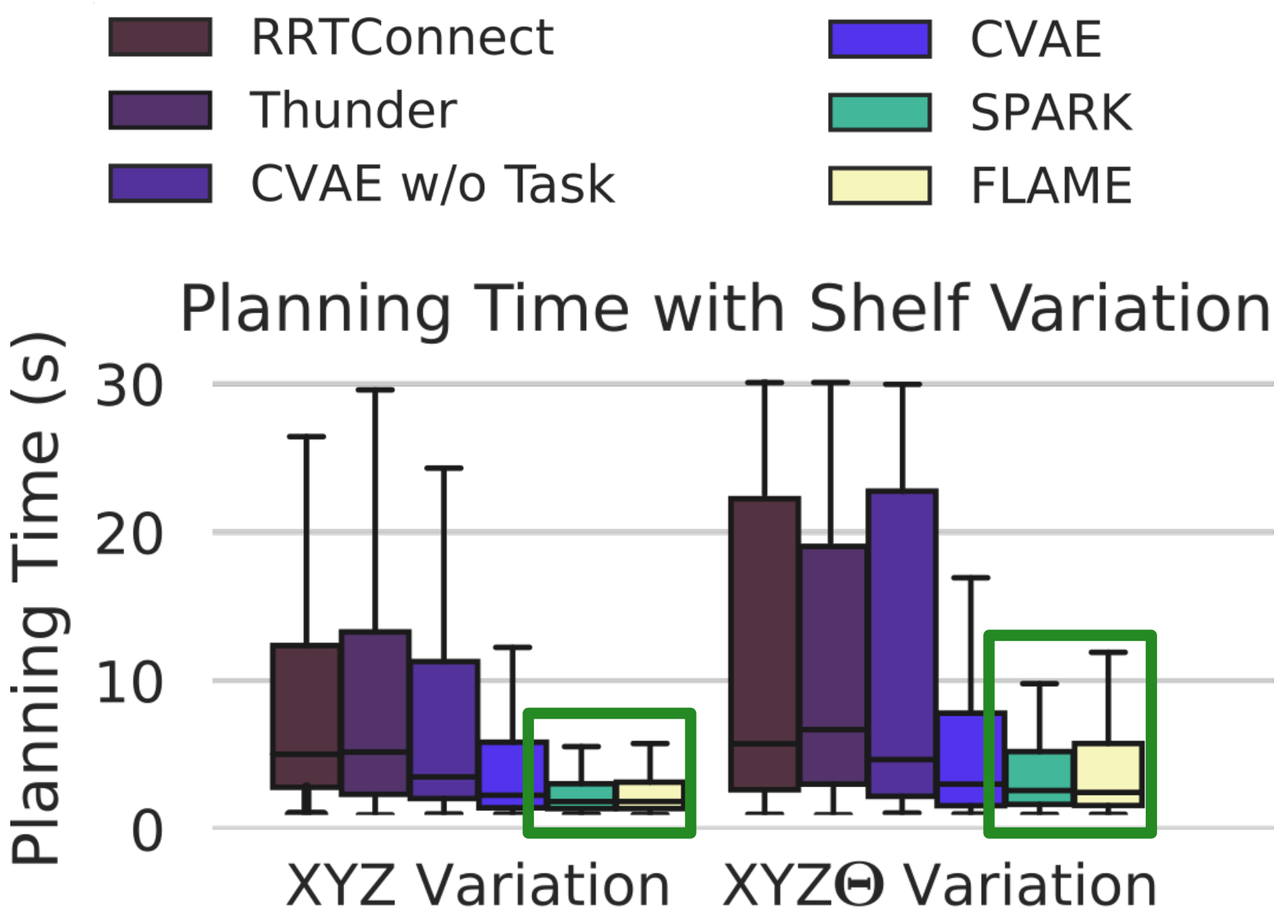
Realistic variation of shelf workspaces

[11] Chamzas et al "Learning Sampling Distributions Using Local 3D Workspace Decompositions for Motion Planning in High Dimensions", *ICRA*, 2021

SPARK/FLAME are robust to workspace variations



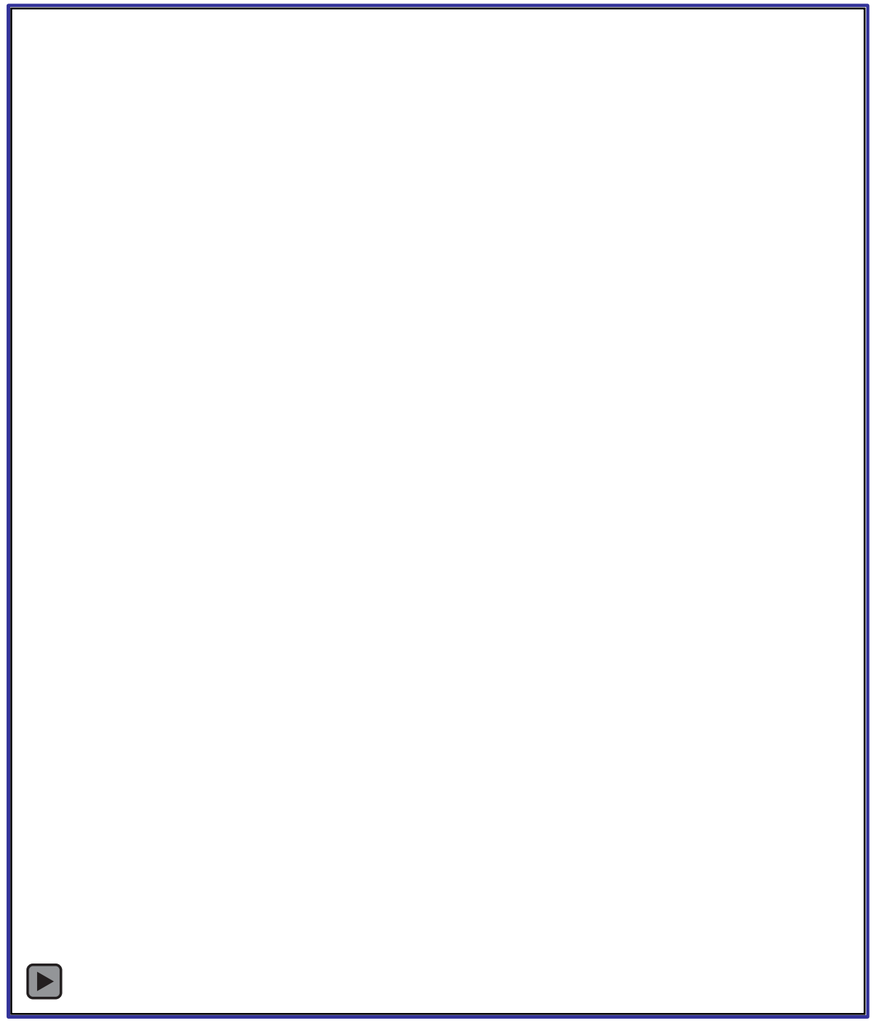
Realistic variation of shelf workspaces



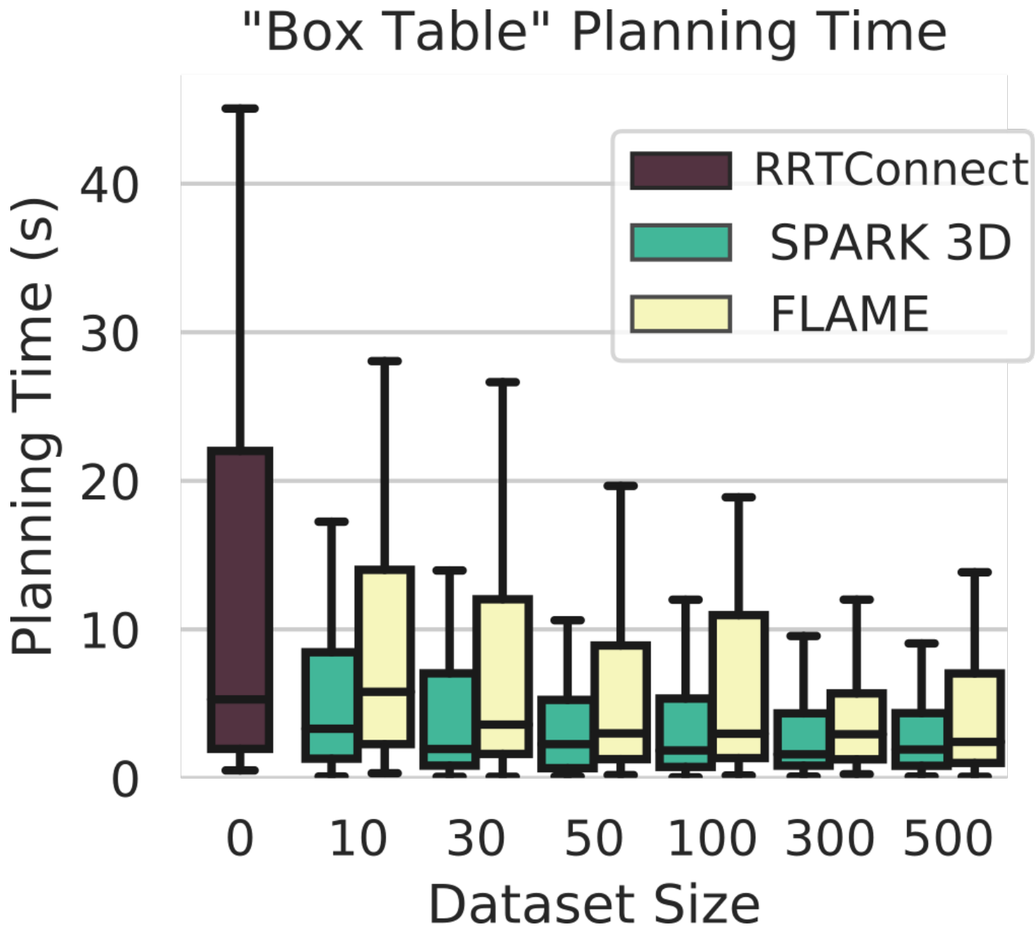
As axes of variation increases we maintain our advantage

[11] Chamzas et al "Learning Sampling Distributions Using Local 3D Workspace Decompositions for Motion Planning in High Dimensions", *ICRA*, 2021

SPARK/FLAME are robust to workspace variations



Realistic variation of table workspaces

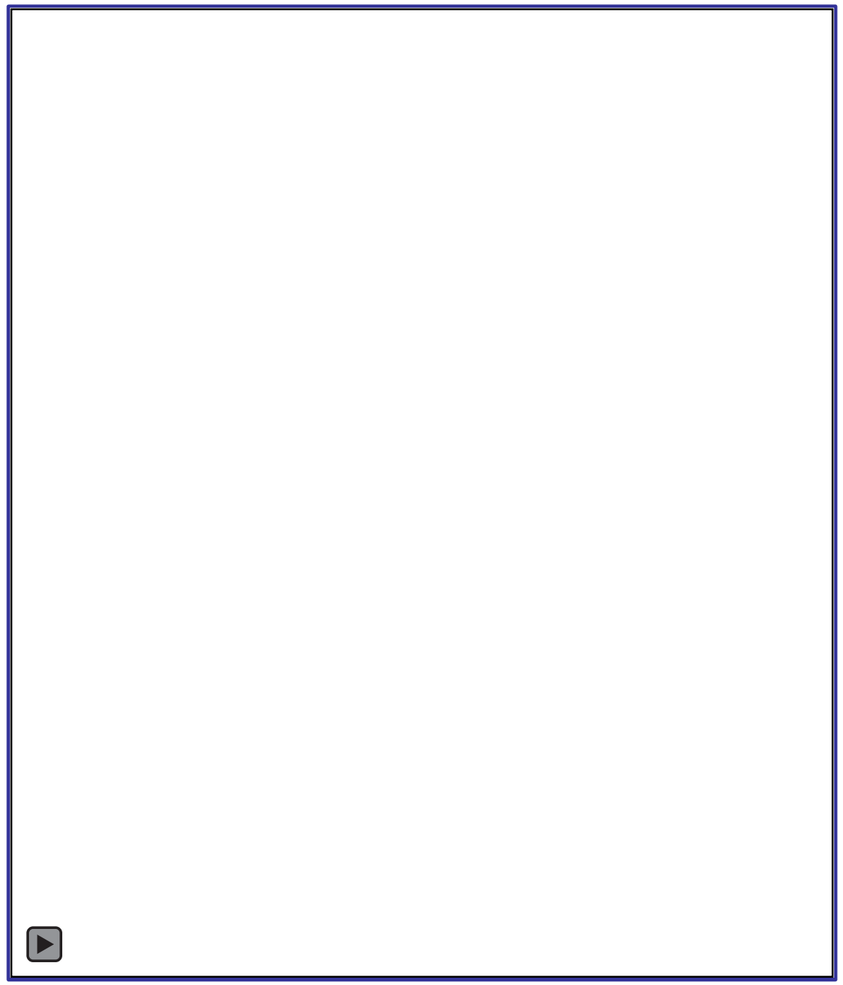


As number of experiences increase the performance improves

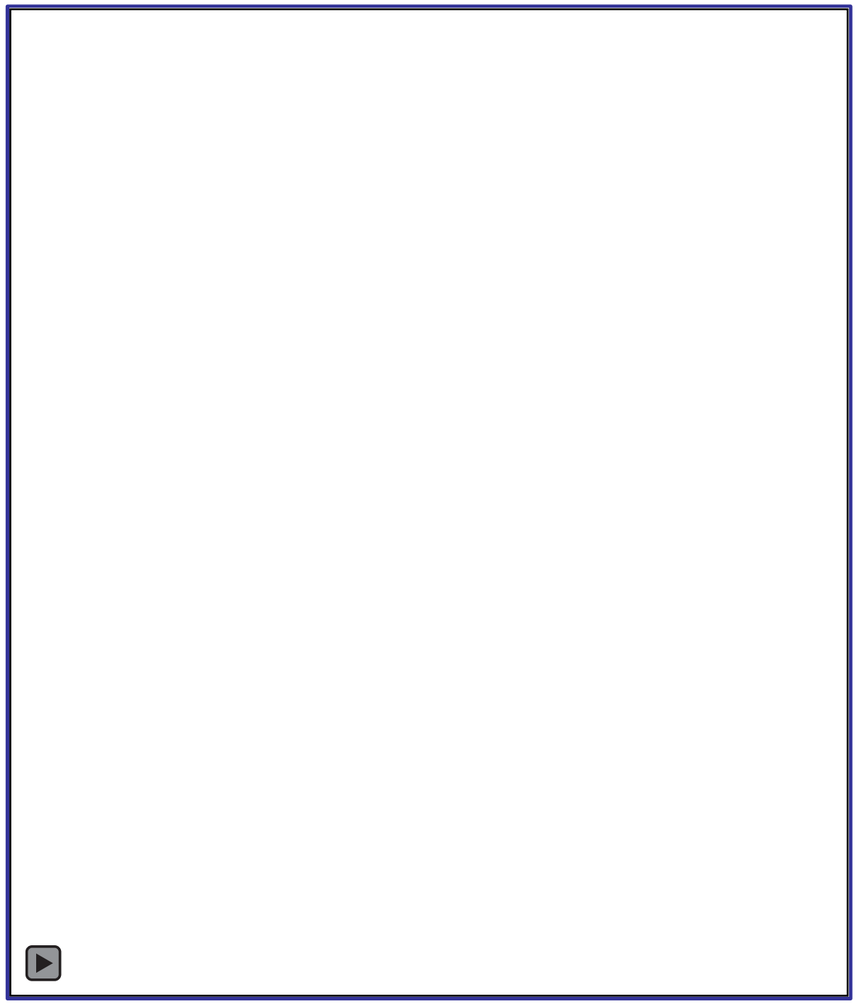
Experiments

1. Robustness to environment variation
2. **Generalization to different tasks**
3. Real robot application

SPARK/FLAME: Generalization to a different task



Pick Task

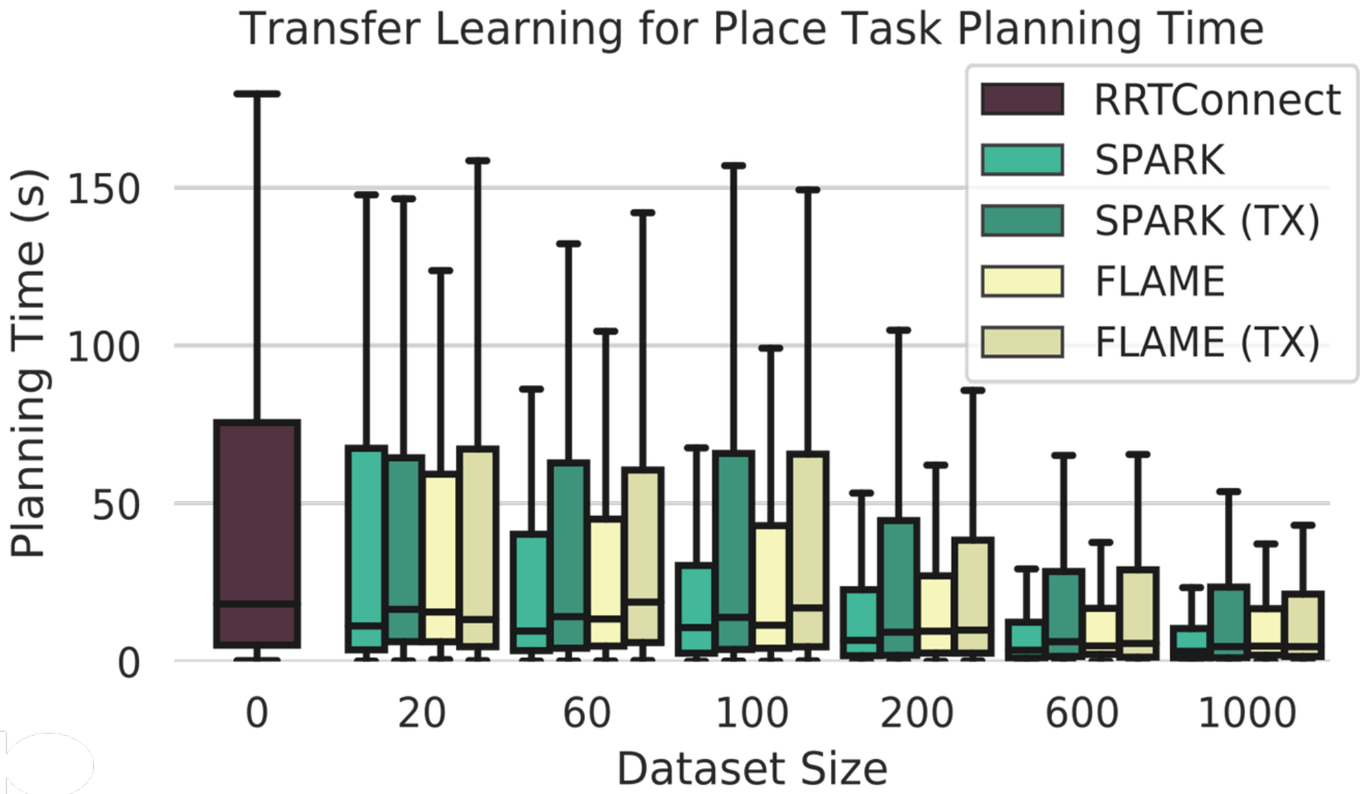


Place Task

[11] Chamzas et al "Learning Sampling Distributions Using Local 3D Workspace Decompositions for Motion Planning in High Dimensions", *ICRA*, 2021

SPARK/FLAME: Generalization from pick to place

Method	Train Task	Test Task
SPARK	Place	Place
FLAME	Place	Place
SPARK (TX)	Pick	Place
FLAME (TX)	Pick	Place

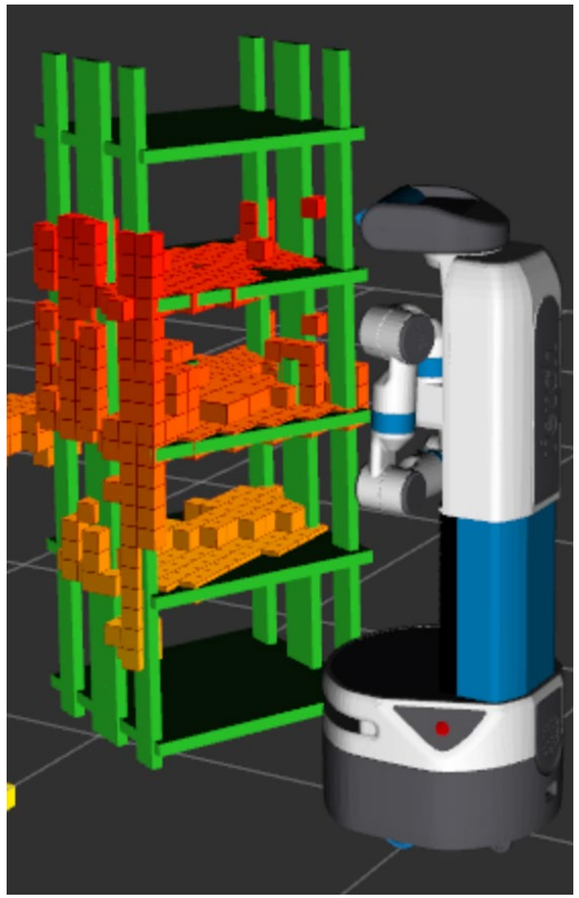


While trained only on pick tasks we generalize to the place task as well

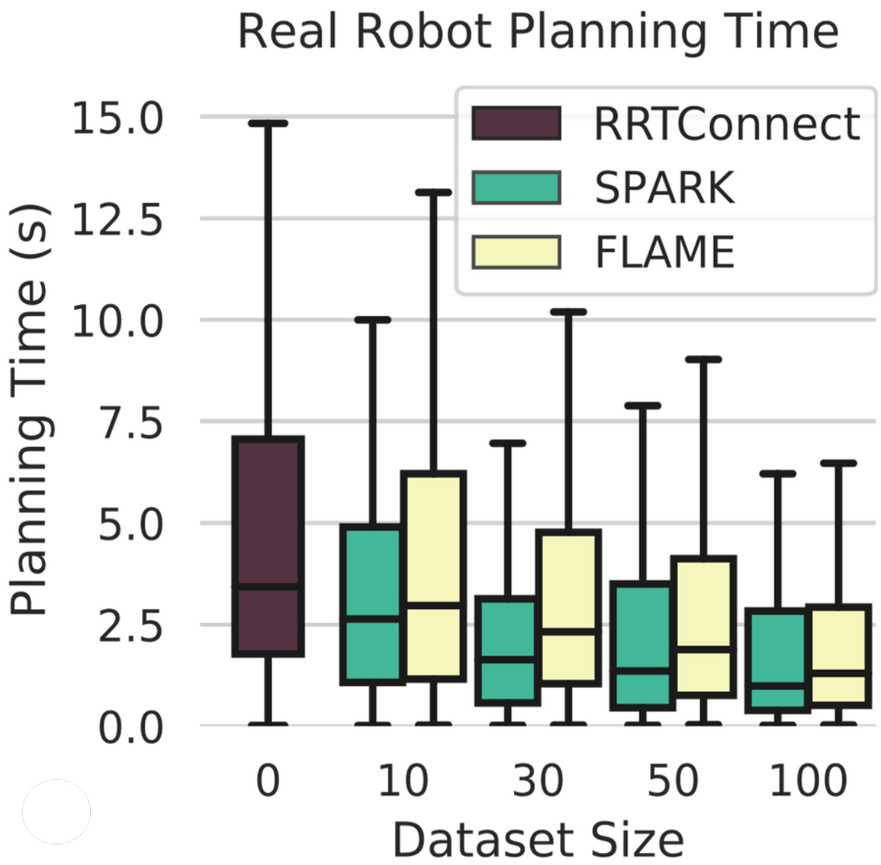
Experiments

1. Robustness to environment variation
2. Generalization to different Tasks
3. **Real robot application**

SPARK/FLAME: Real Fetch experiment

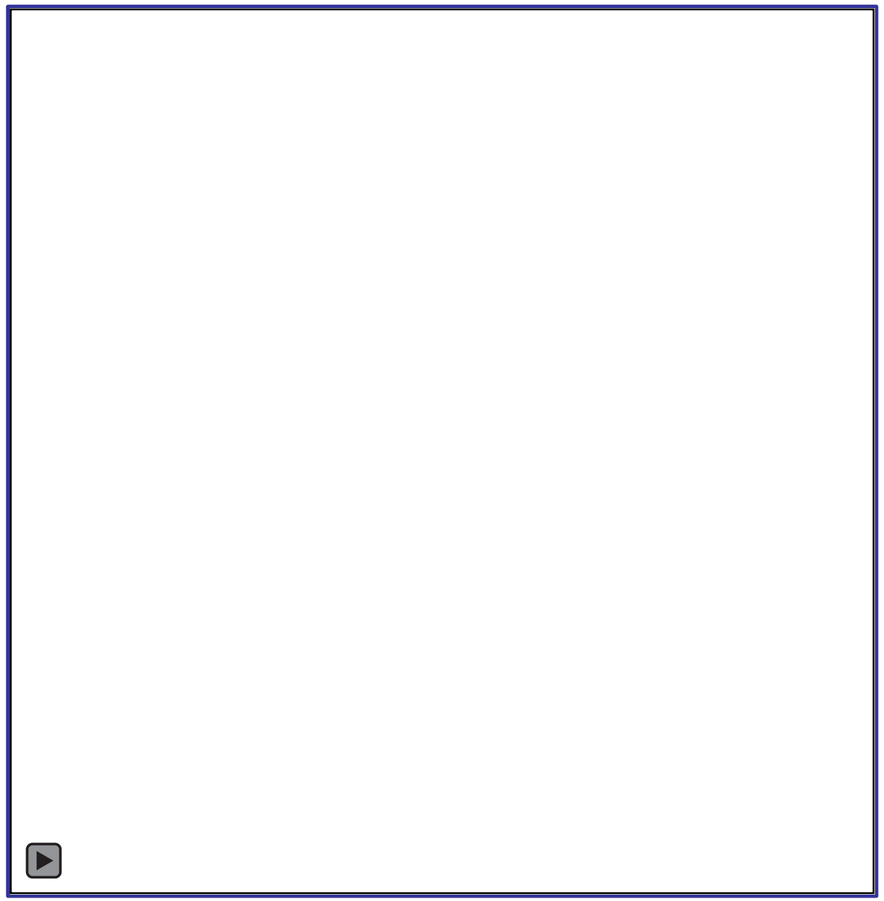


Application in a real setting



[11] Chamzas et al "Learning Sampling Distributions Using Local 3D Workspace Decompositions for Motion Planning in High Dimensions", *ICRA*, 2021

SPARK/FLAME: Real Fetch experiment

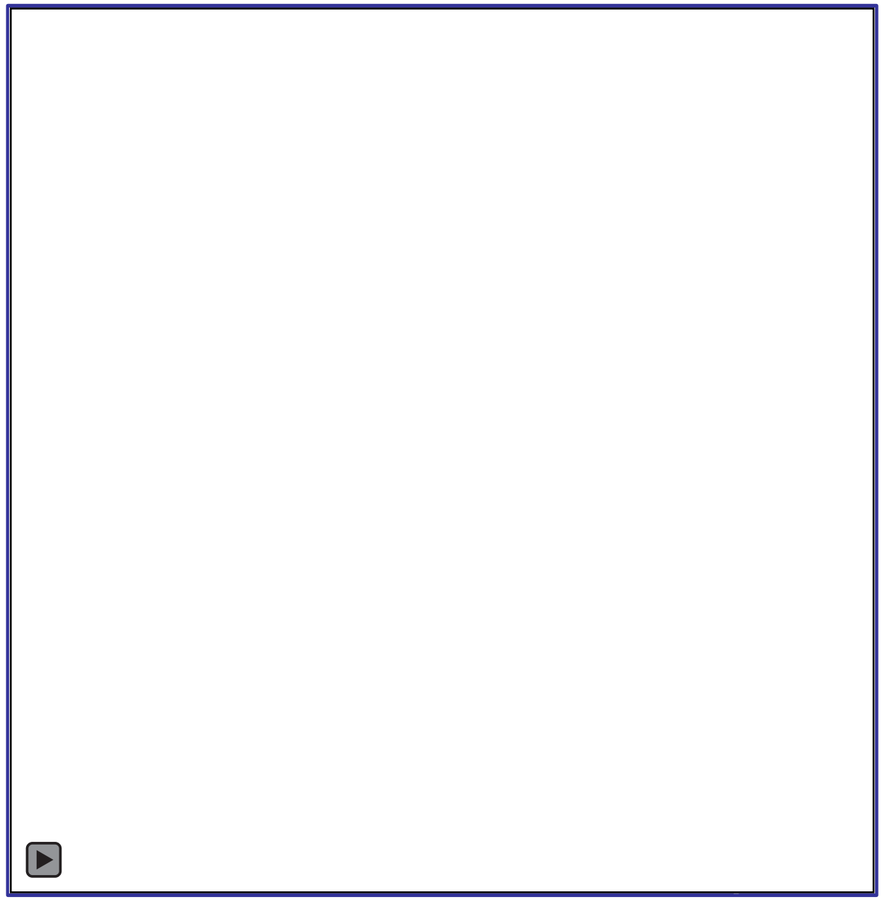


Example solution path
of pick and tuck

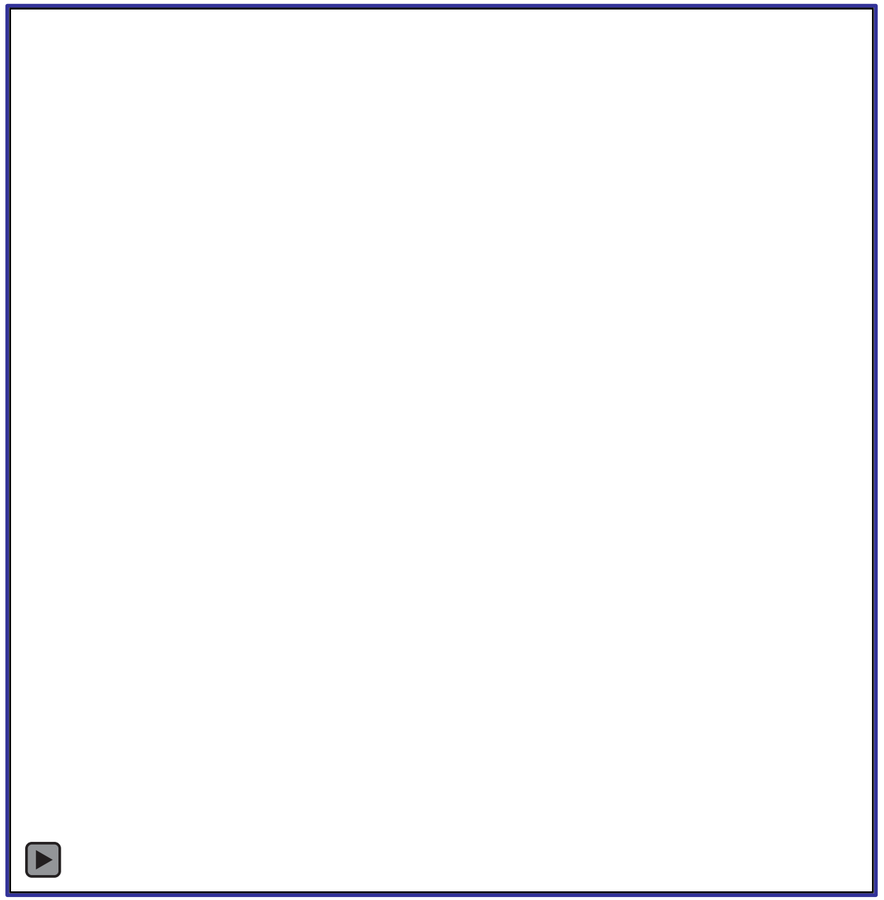
Method	Planning time
SPARK	**~3.4** faster
FLAME	**~2.6** faster

[11] Chamzas et al “Learning Sampling Distributions Using Local 3D Workspace Decompositions for Motion Planning in High Dimensions”, *ICRA*, 2021

SPARK/FLAME: Real Fetch experiment



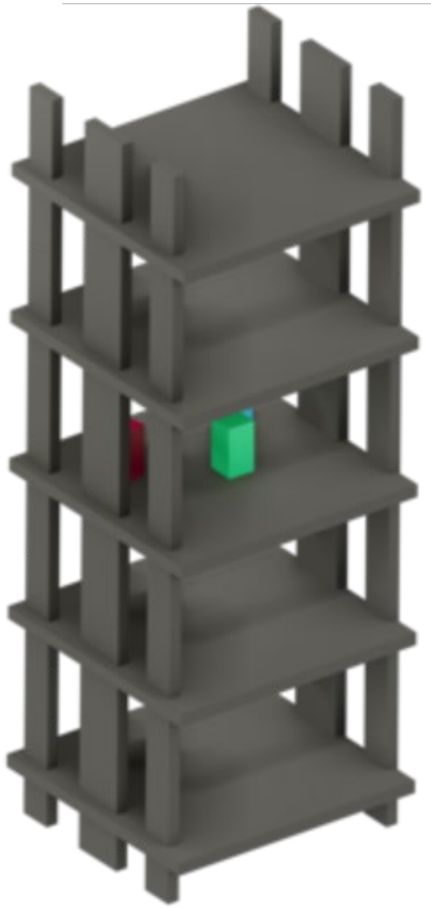
RRT-Connect returns only approximate solution in 2s



Solution path

[11] Chamzas et al “Learning Sampling Distributions Using Local 3D Workspace Decompositions for Motion Planning in High Dimensions”, *ICRA*, 2021

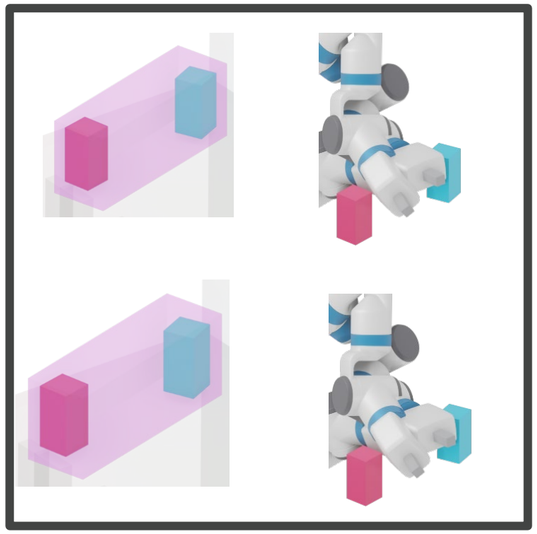
Limitations of framework



Workspace W

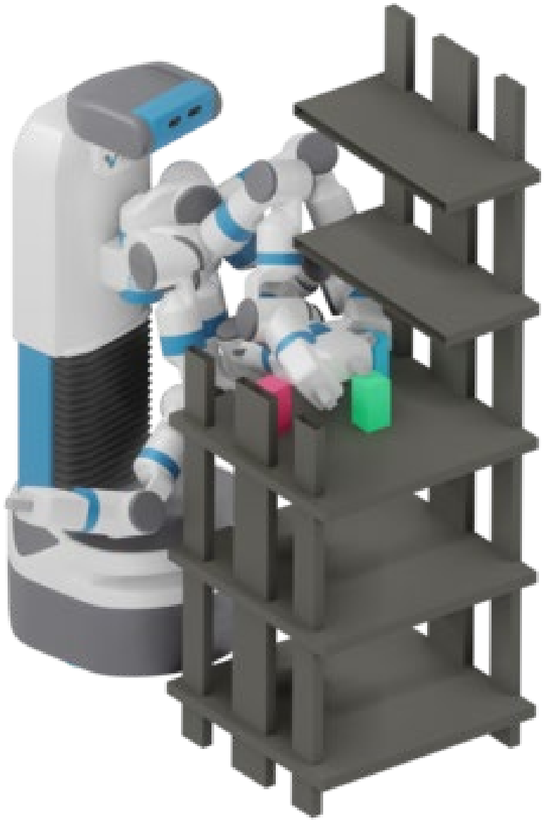
1. Decompose

2. Create



3. Retrieve (Hard Coded)

4. Synthesize



Sampling Dist. $\mathrm{P}(X|W)$

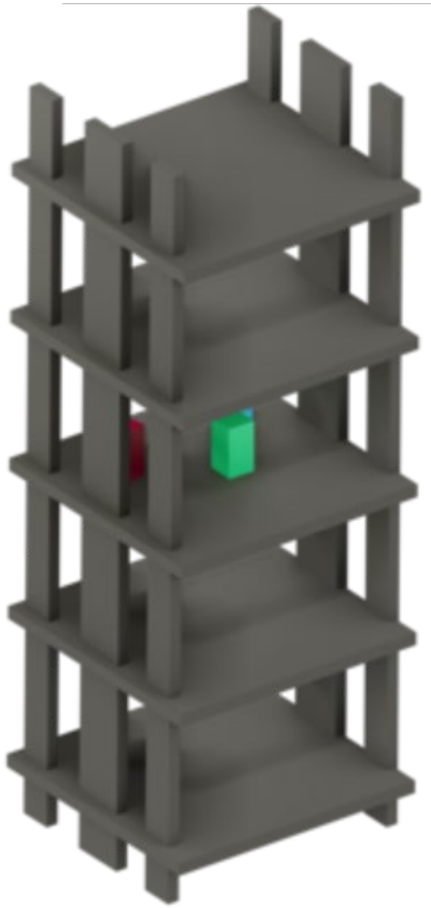
No Start, Goal

Today’s Agenda

- Important Considerations
  - Locality of narrow passages
  - Discontinuity

- Biased Samplers (Cont.)
  - 2D Workspaces
  - 3D Workspaces
  - **Workplace, Start and Goal**

Learning the similarity function

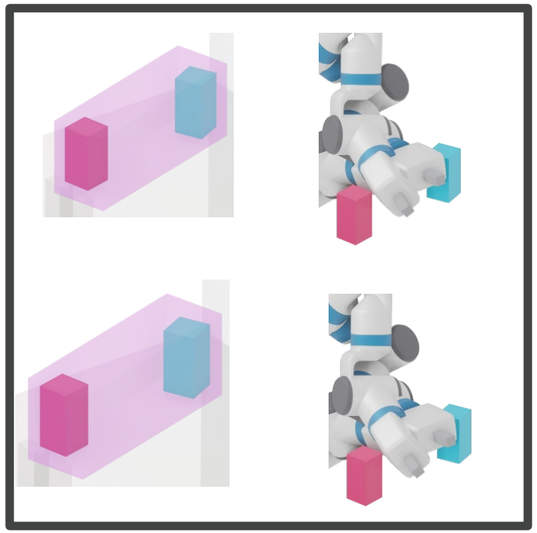


Workspace W

Start S / Goal G

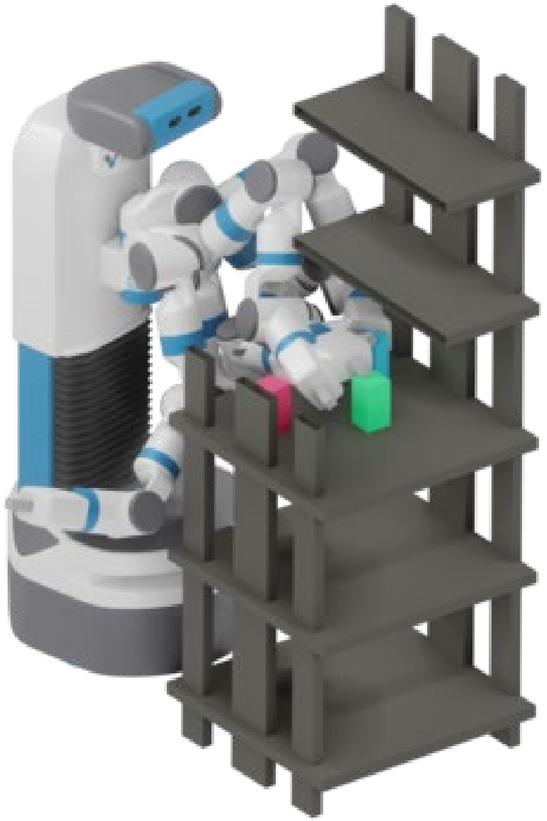
1. Decompose

2. Create



3. Retrieve **(Learn)**

4. Synthesize

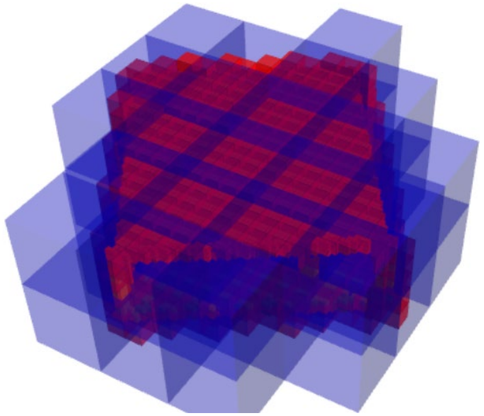


Sampling Dist.

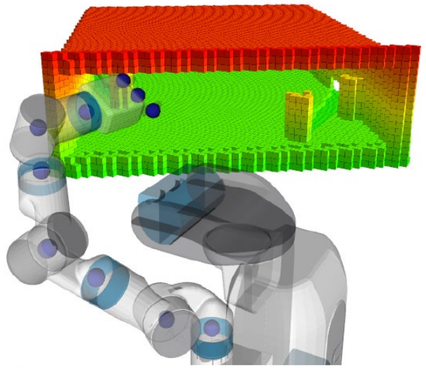
$P(X|W, S, G)$

Chamzas et al *ICRA* 2022

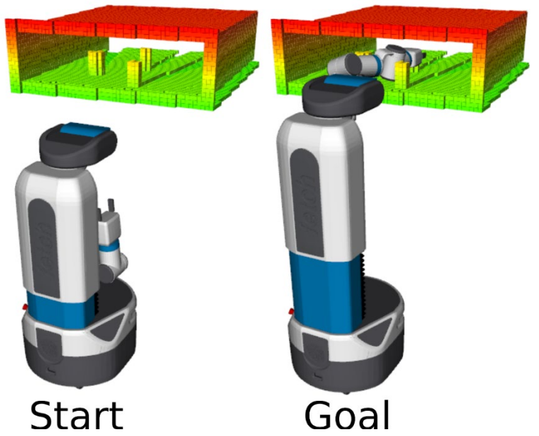
Local representations and similarity datasets



Local Features



Intermediate Features



Global Features

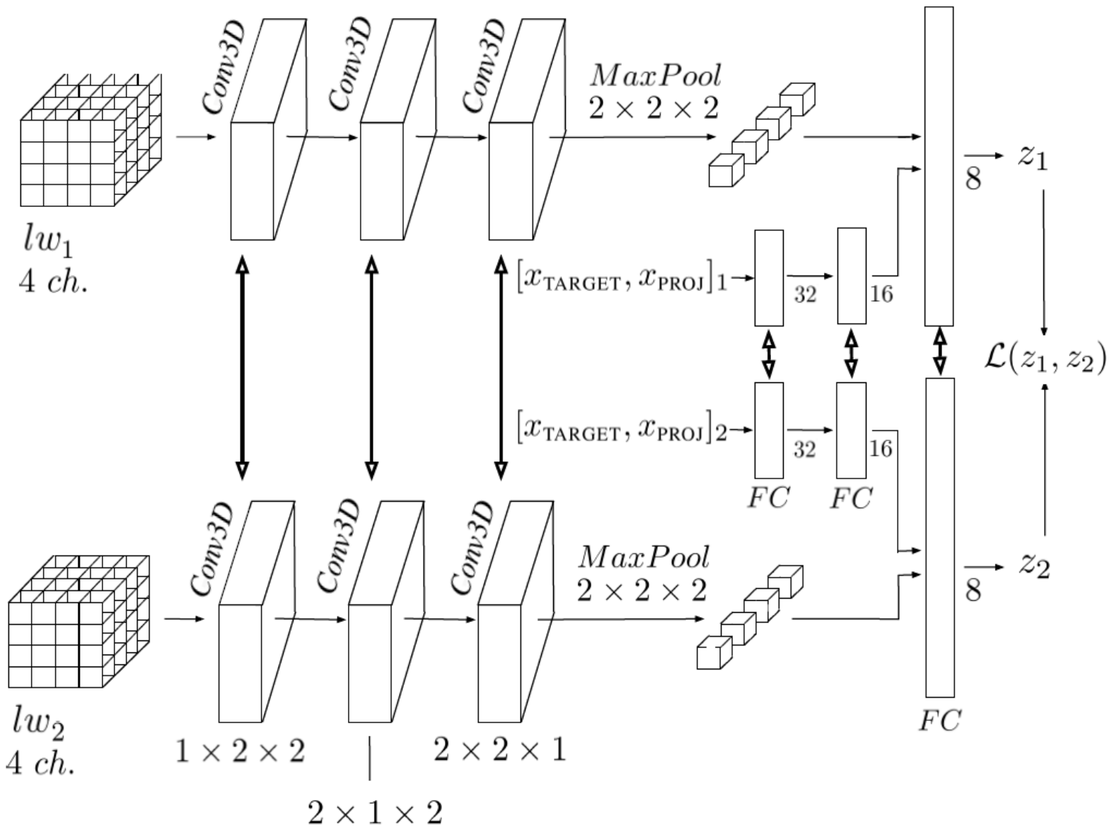
Generating a dataset of similar/dissimilar pairs of local workspaces

Similar pairs: Locally compare past solution paths

Dissimilar pairs: Randomly sample local workspaces

[12] Chamzas et al "Learning to Retrieve Relevant Experiences for Motion Planning", *ICRA*, 2022

Learn the similarity with a Siamese Network



Siamese Network architecture

$$\mathcal{L}(\ell_i, \ell_j) = \begin{cases} \max(0, d_m - \|z_i - z_j\|^2) & \text{if } \langle \ell_j, \ell_i \rangle \in \mathcal{DS} \\ \|z_i - z_j\|^2 & \text{if } \langle \ell_j, \ell_i \rangle \in \mathcal{S} \end{cases}$$

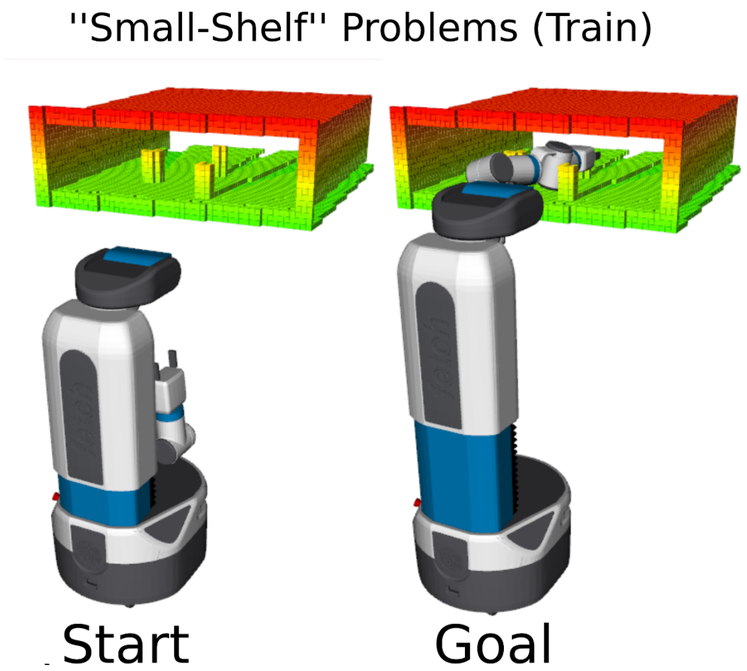
Contrastive loss function

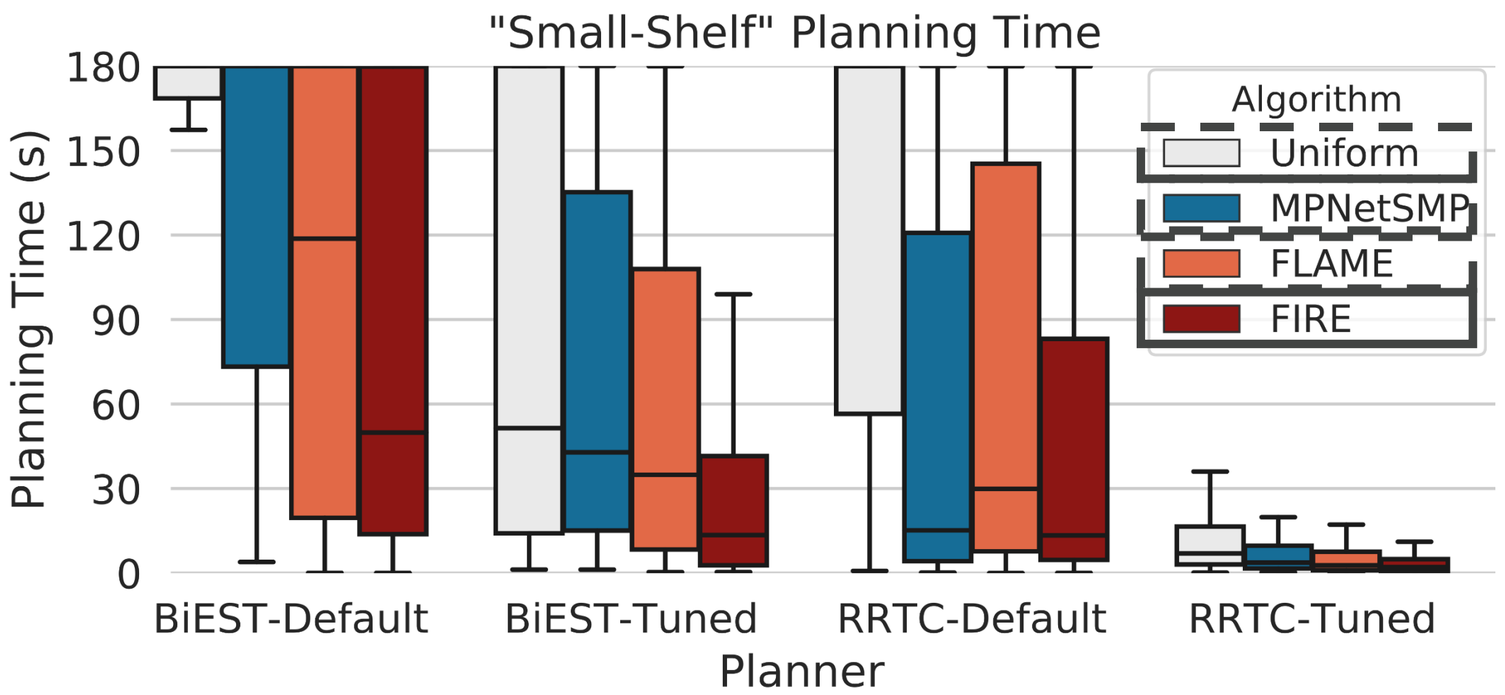
$$\text{SIM}(\ell_i, \ell_j) = \begin{cases} 1 & \text{if } \|z_i - z_j\|^2 < R \\ 0 & \text{otherwise} \end{cases}$$

Realized similarity function

[12] Chamzas et al "Learning to Retrieve Relevant Experiences for Motion Planning", *ICRA*, 2022

Generalizing within the training class of problems

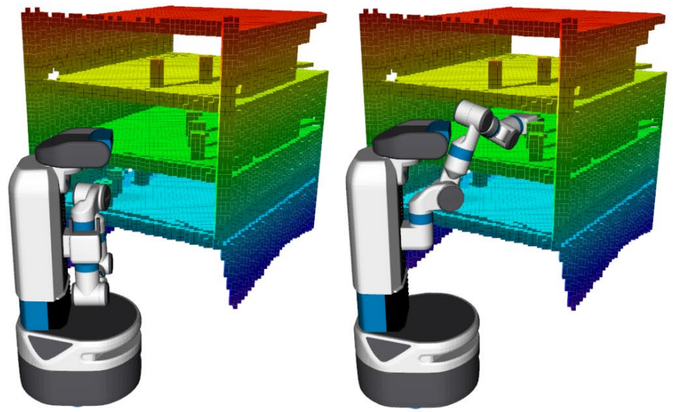




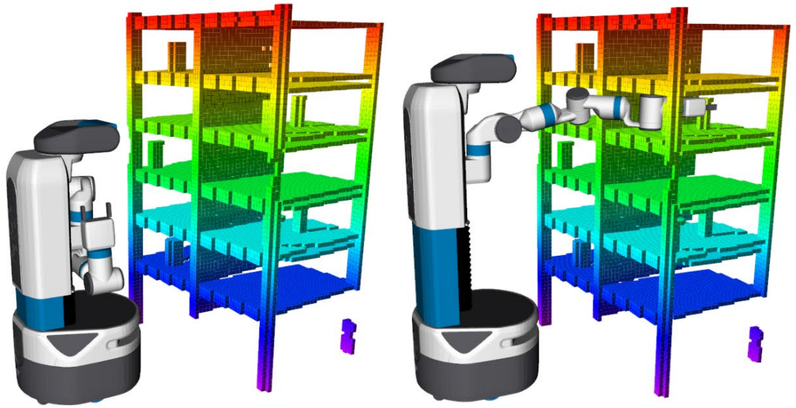
[12] Chamzas et al "Learning to Retrieve Relevant Experiences for Motion Planning", *ICRA*, 2022

Generalizing outside the training class of problems

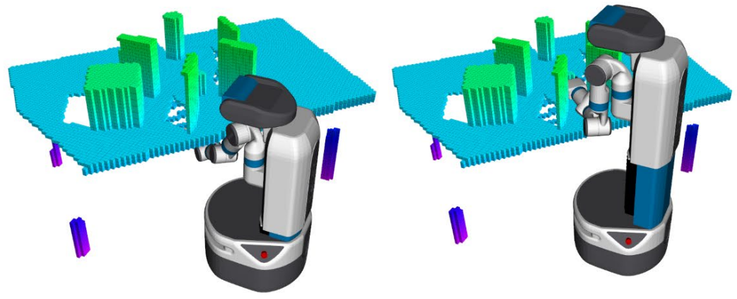
"Tall-Shelf" Dataset (Test)

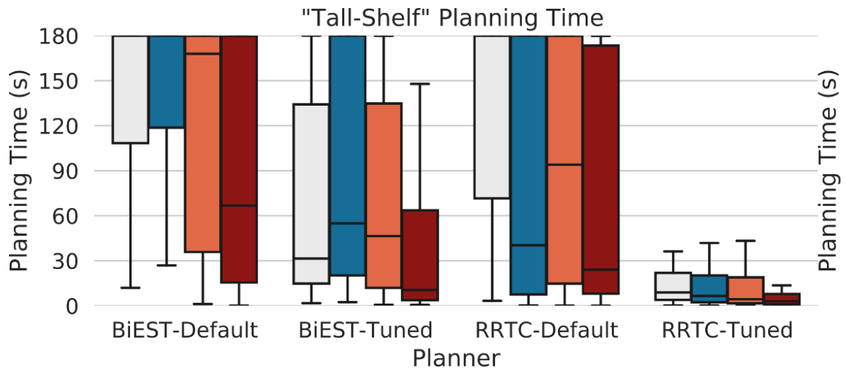


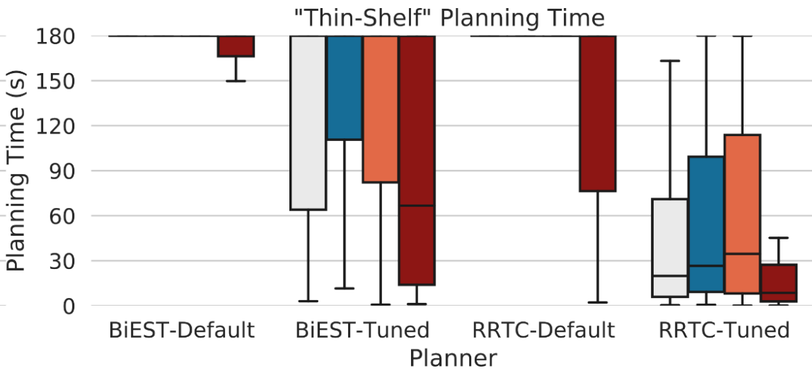
"Thin-Shelf" Dataset (Test)

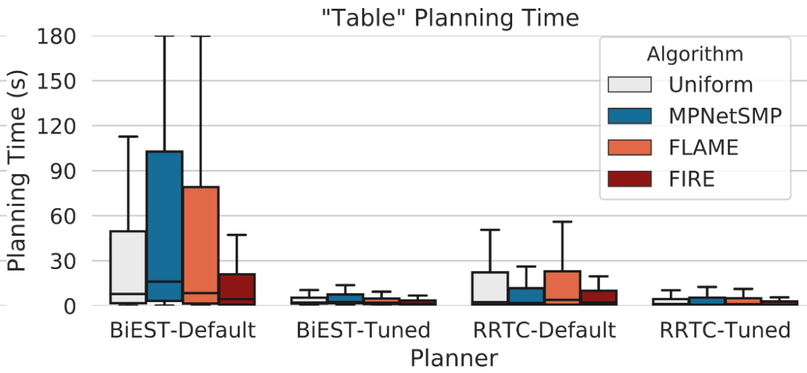


"Table" Dataset (Test)





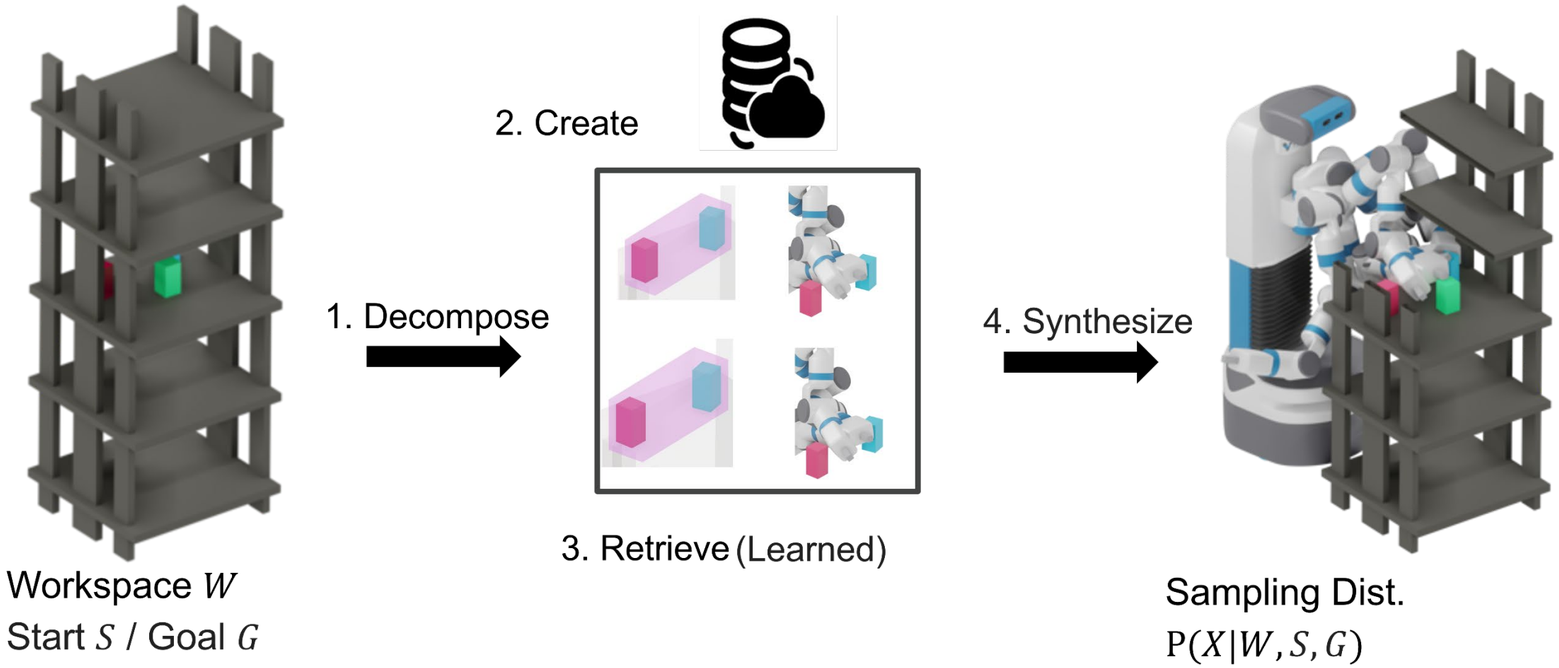




[12] Chamzas et al "Learning to Retrieve Relevant Experiences for Motion Planning", *ICRA*, 2022

Summary of SPARK+FLAME+FIRE = Pyre

- Improve planning time for high-dof robots
- Learns incrementally
- Apply to realistic robots and diverse workspaces
- Generalized to unseen tasks and environments



Open-source code: https://github.com/KavrakiLab/pyre

Chamzas et al *ICRA* 2019, 2021, 2022

Today’s Agenda

- Important Considerations
  - Locality of narrow passages
  - Discontinuity

- Biased Samplers (Cont.)
  - 2D Workspaces (SPARK2D)
  - 3D Workspaces (FLAME)
  - Workplace, Start and Goal (FIRE, CVAE)

References (Retrieve-and-Repair)

[1] Berenson, Dmitry, Pieter Abbeel, and Ken Goldberg. "A robot path planning framework that learns from experience."*International Conference on Robotics and Automation*. IEEE, 2012.

[2] Coleman, David, et al. "Experience-based planning with sparse roadmap spanners." *International Conference on Robotics and Automation (ICRA)*. IEEE, 2015.

[3] Pairet, Èric, et al. "Path Planning for Manipulation using Experience-driven Random Trees." *IEEE Robotics and Automation Letters* 6.2 (2021): 3295-3302.

[4] Lien, Jyh-Ming, and Yanyan Lu. "Planning motion in environments with similar obstacles." *Robotics: Science and systems*. 2009.

[5] Jetchev, Nikolay, and Marc Toussaint. "Fast motion planning from experience: trajectory prediction for speeding up movement generation." *Autonomous Robots* 34.1 (2013): 111-127.

References (Biased Samplers 1)

[6] Lehner, Peter, and Alin Albu-Schäffer. "Repetition sampling for efficiently planning similar constrained manipulation tasks." *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2017.

[7] Lehner, Peter, and Alin Albu-Schäffer. "The repetition roadmap for repetitive constrained motion planning." *IEEE Robotics and Automation Letters* 3.4 (2018): 3884-3891.

[8] Zucker, Matt, James Kuffner, and J. Andrew Bagnell. "Adaptive workspace biasing for sampling-based planners."*International Conference on Robotics and Automation*. IEEE, 2008.

[9] Chamzas, Constantinos, Anshumali Shrivastava, and Lydia E. Kavraki. "Using local experiences for global motion planning." *International Conference on Robotics and Automation (ICRA)*. IEEE, 2019.

References (Biased Samplers 2)

[11] Ichter, Brian, James Harrison, and Marco Pavone. "Learning sampling distributions for robot motion planning."*International Conference on Robotics and Automation (ICRA)*. IEEE, 2018.

[10] Chamzas, Constantinos, et al. "Learning sampling distributions using local 3D workspace decompositions for motion planning in high dimensions."*International Conference on Robotics and Automation (ICRA)*. IEEE, 2021

[12] Chamzas, Constantinos, et al. "Learning to retrieve relevant experiences for motion planning.“ *International Conference on Robotics and Automation (ICRA)*. IEEE, 2022.