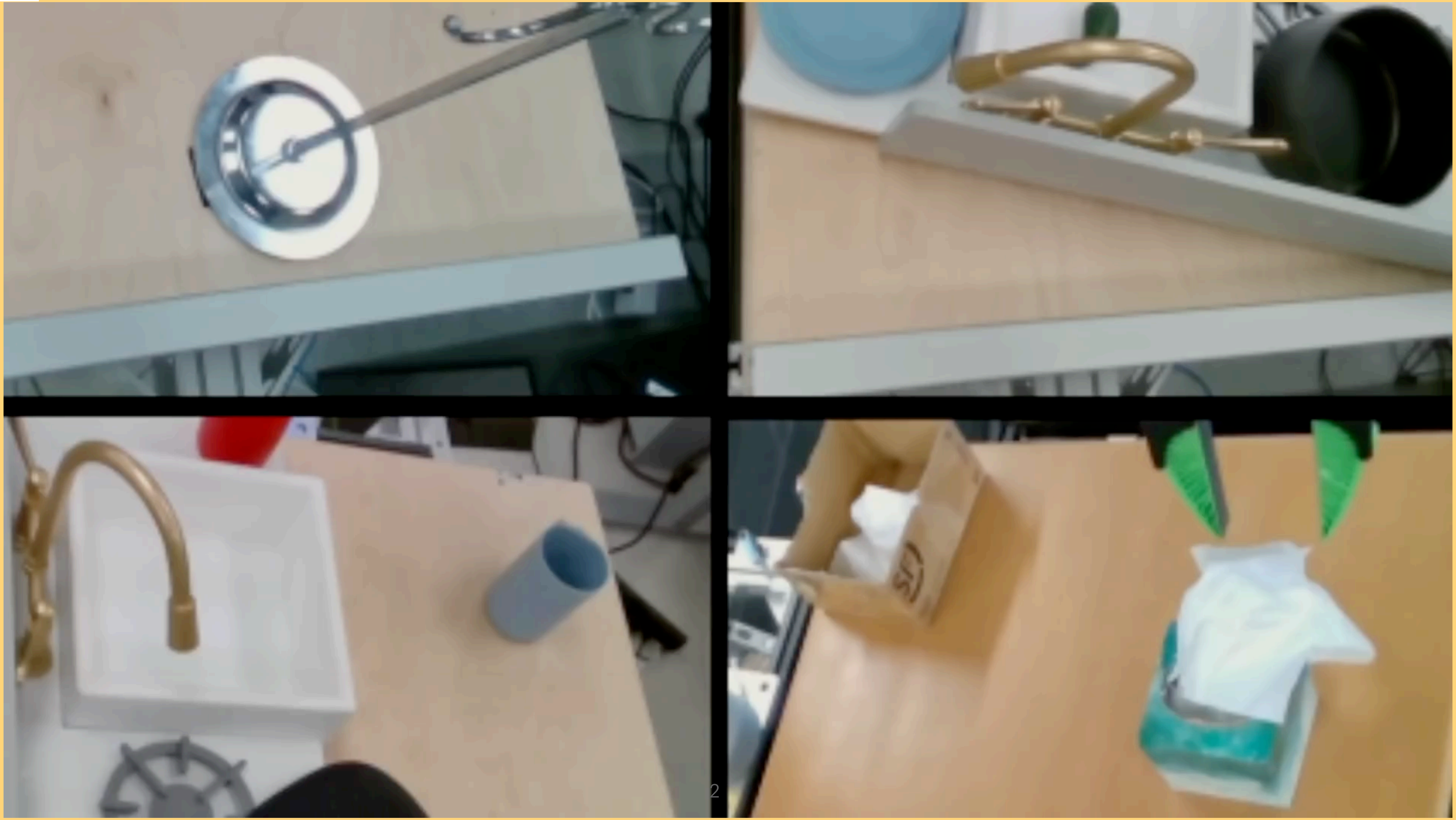
Continuous-Token Behavior Cloning: Pitfalls and Promises

Max Simchowitz (MIT ⟶ CMU)

joint w/ Adam Block (MIT ⟶ Columbia) Dan Pfrommer, Russ Tedrake, Ali Jadbabie (MIT) + Thomas Zhang, Boyuan Chen, Allen Ren et others..

(not my work)

Behavior Cloning



Understanding Behavior Cloning

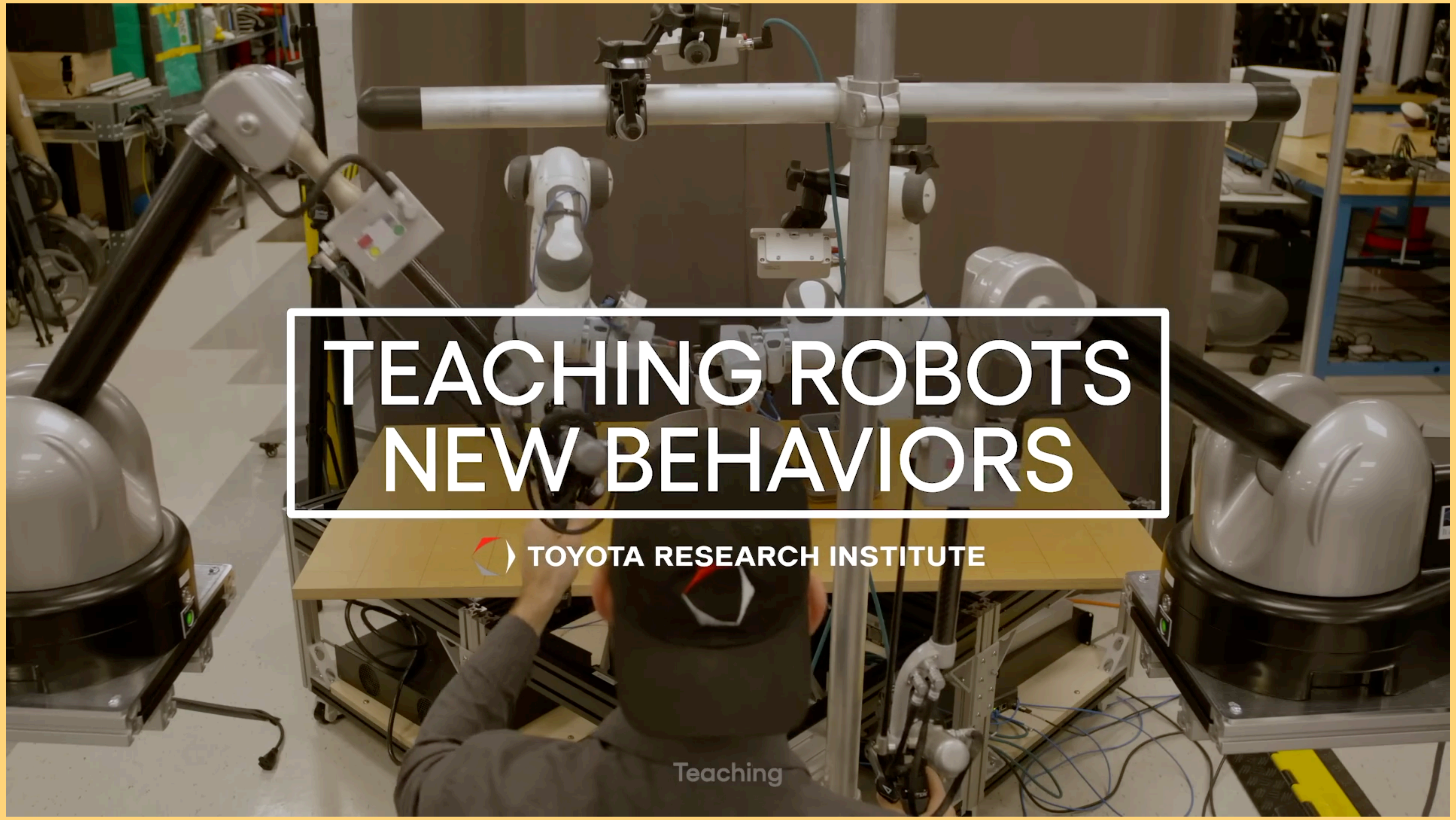
a building block for modern robot learning

Diffusion Policy

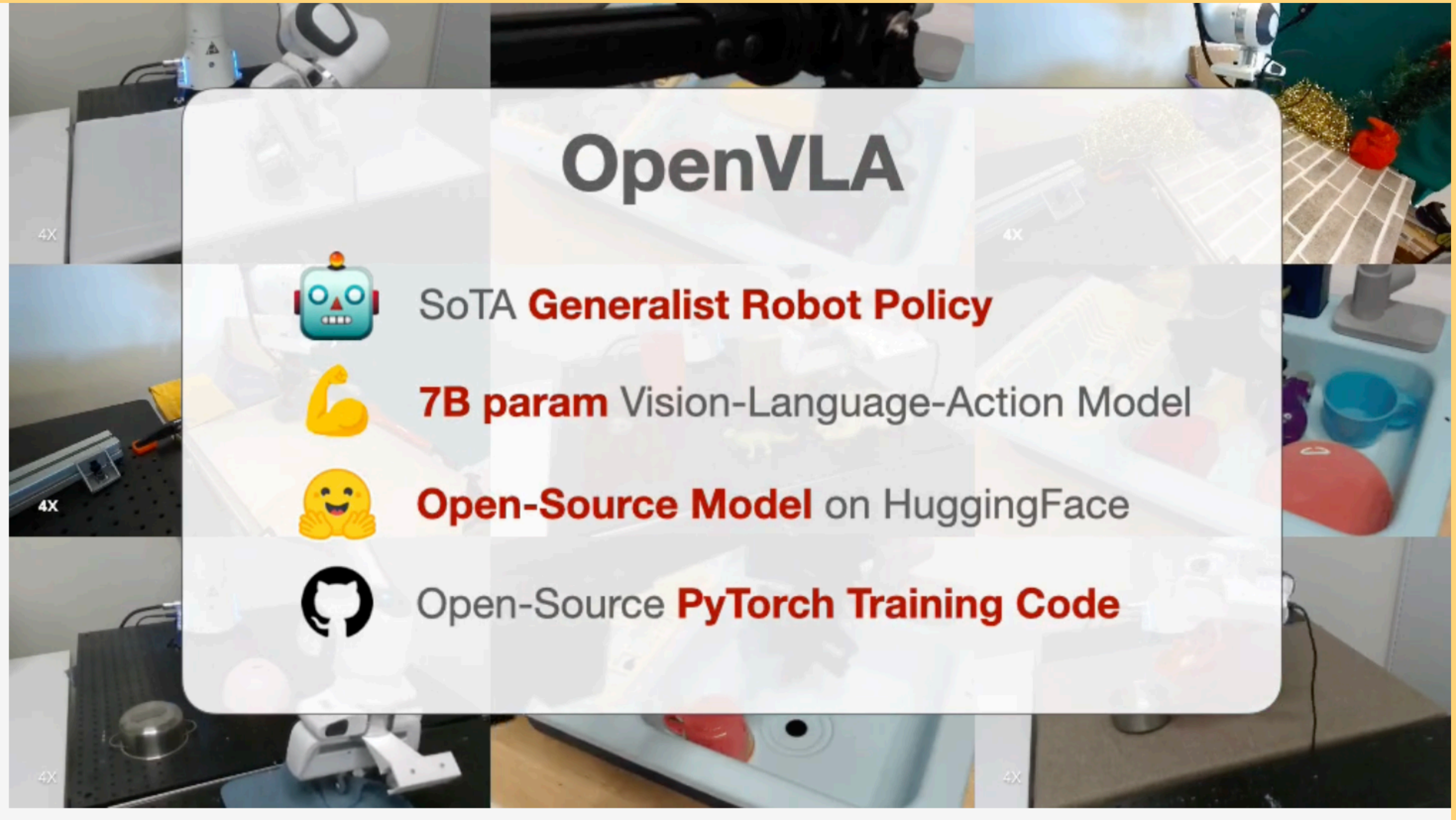
Robotic Transformer 2 (RT-2)

Octo

Dobb·E



Video Language Action Models

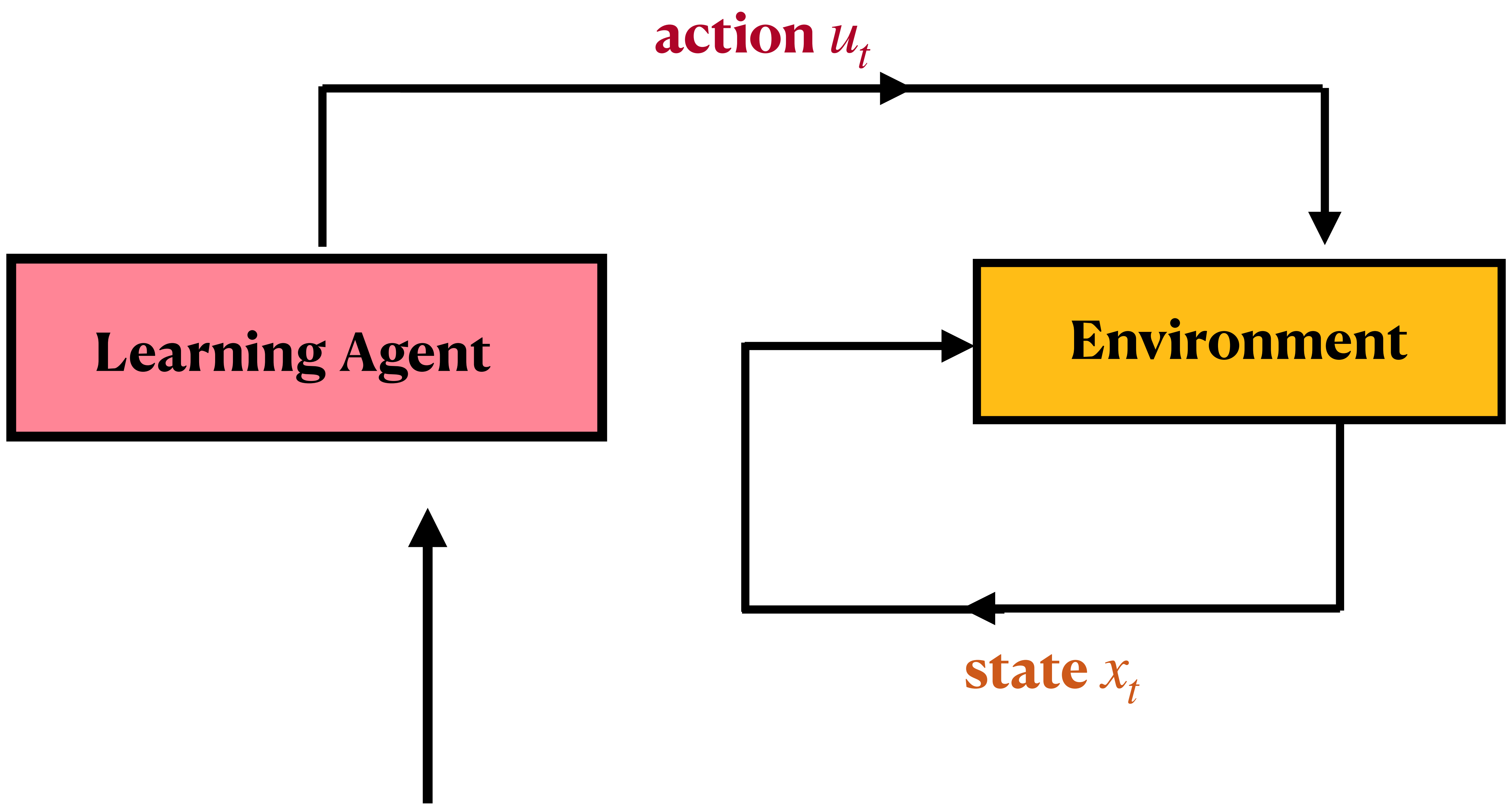


towards 'continuous tokens'

Control Systems

$$x_{t+1} = f(x_t, u_t)$$

system state — control input

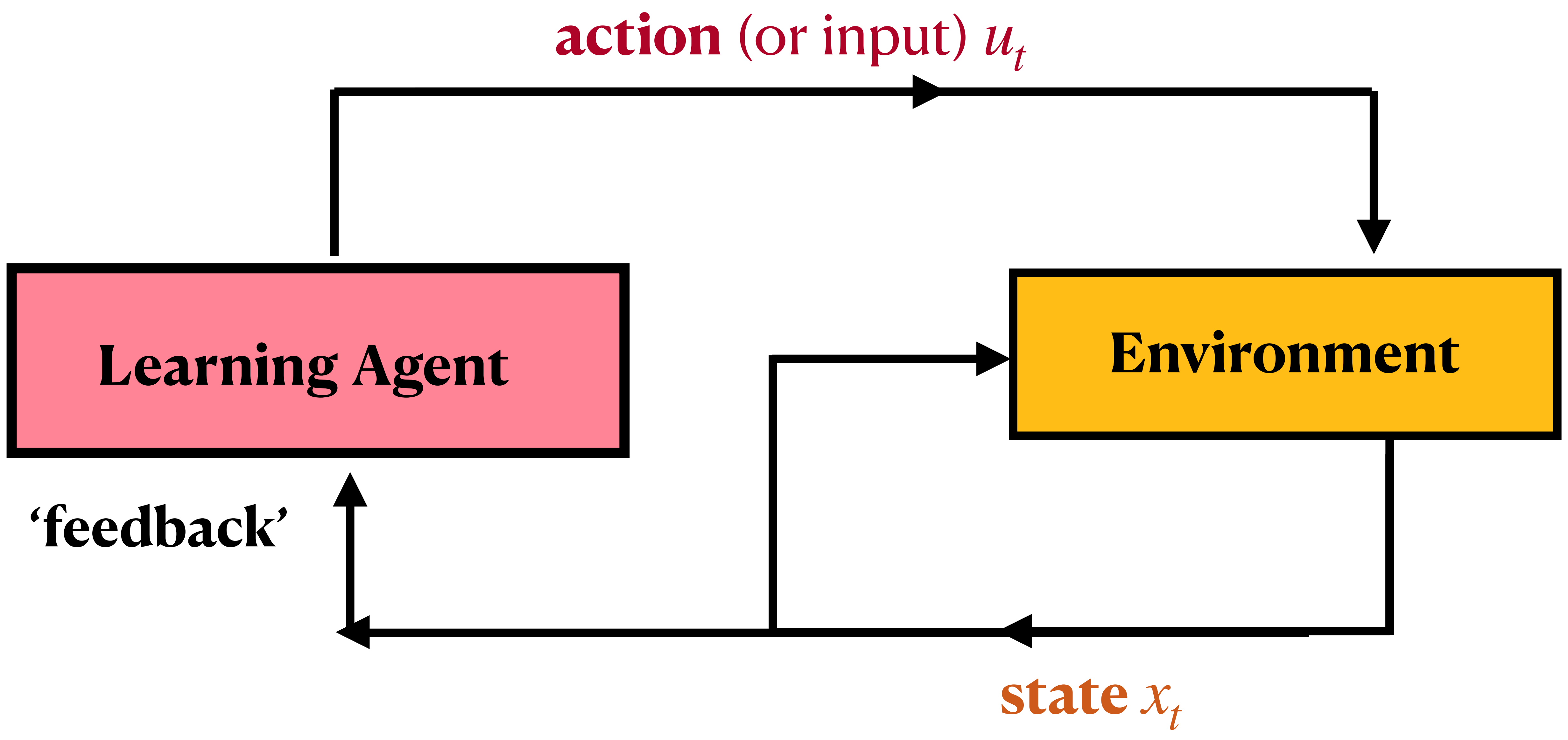


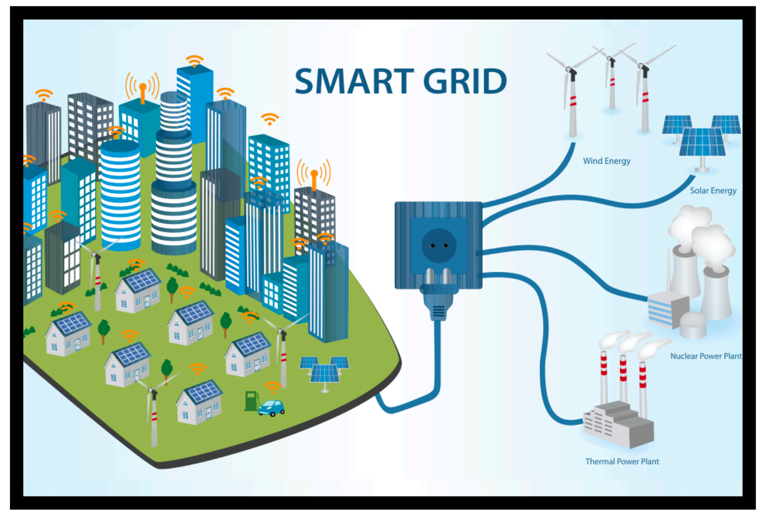
continuous state & action

*no partial observability

Control Systems

$$x_{t+1} = f(x_t, u_t)$$



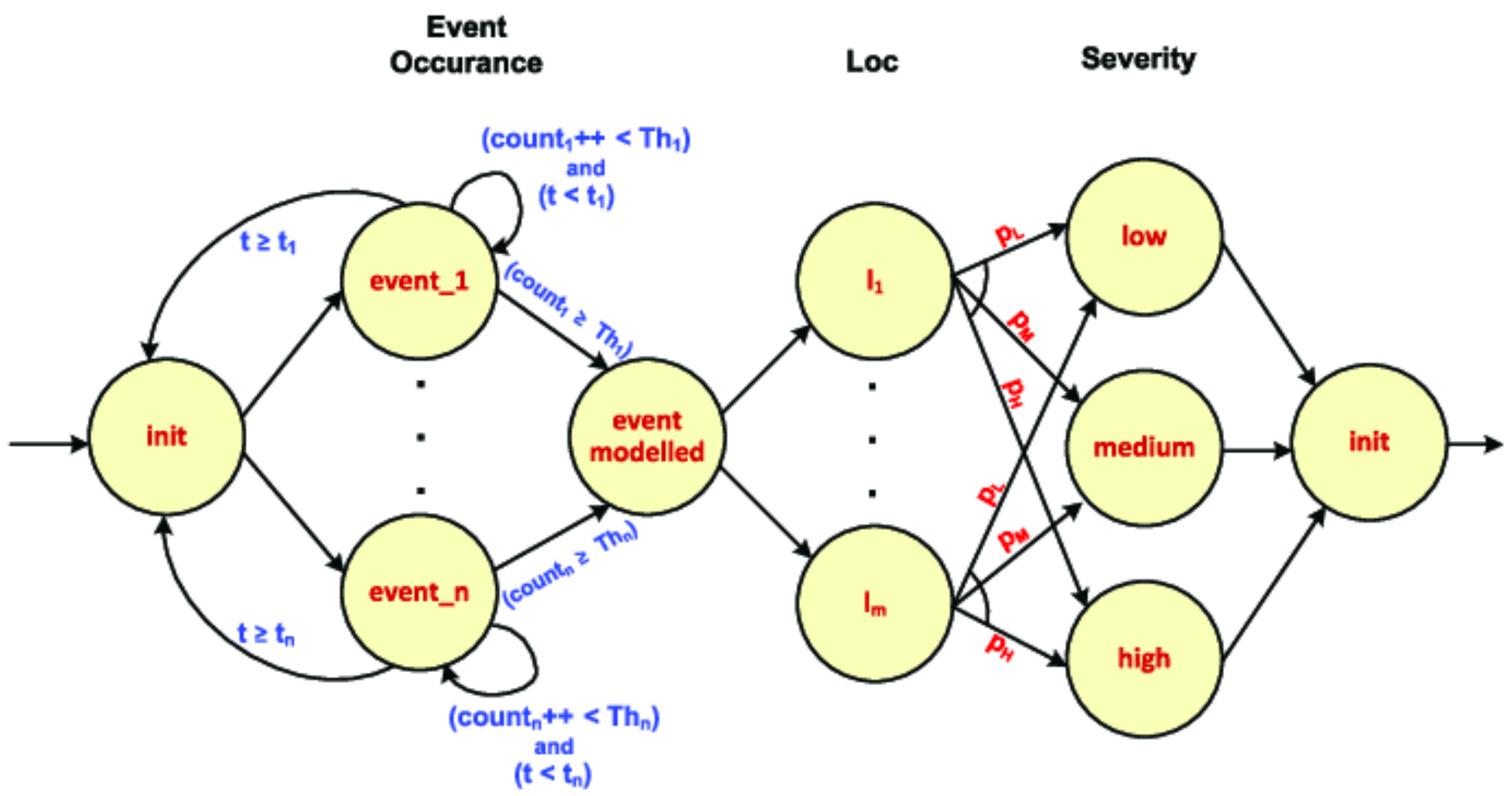


abstraction reflects physical laws.

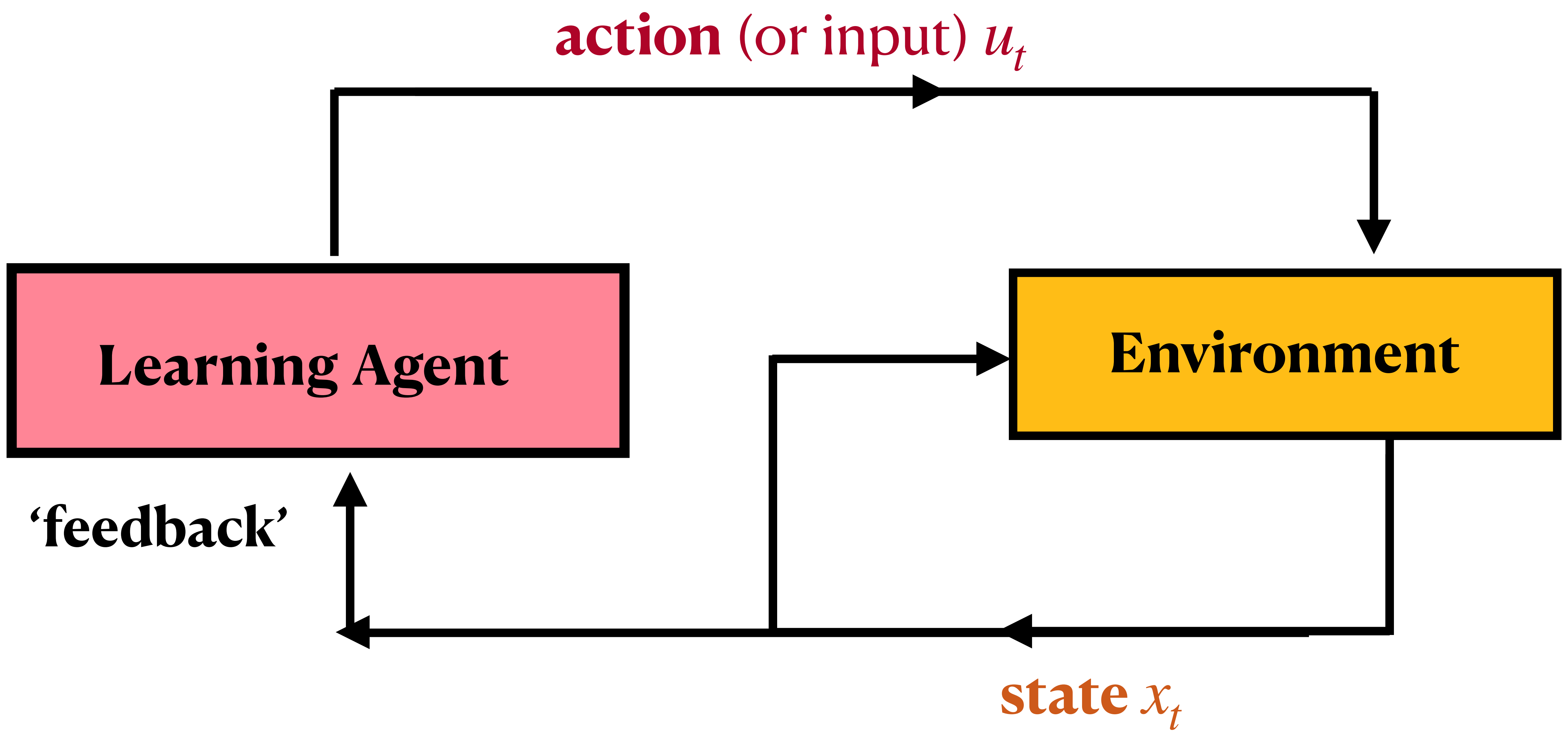
**no partial observability*

Control Systems

$$x_{t+1} = f(x_t, u_t)$$



network routing, AlphaGo, etc ...

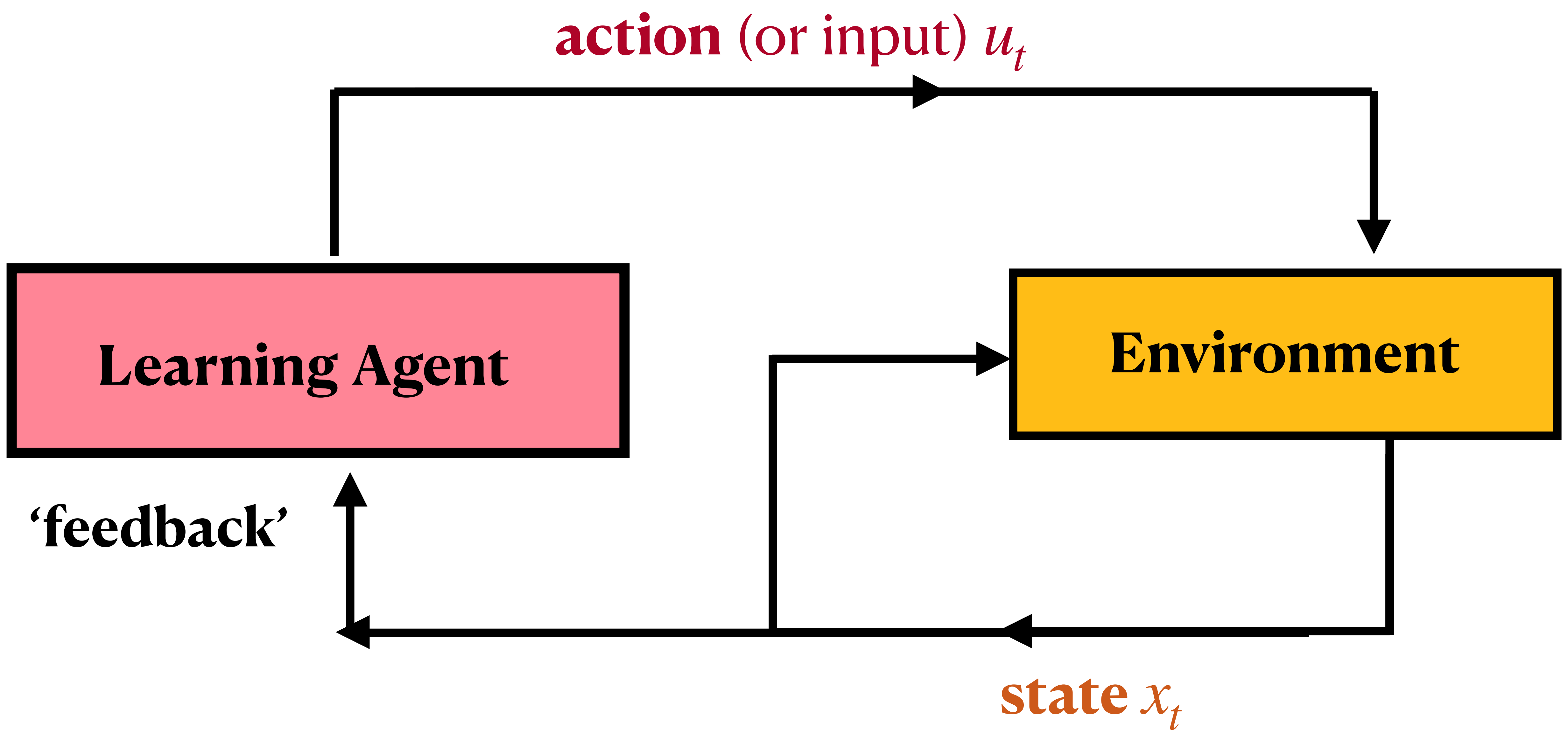


abstraction reflects physical laws.

*no partial observability

Control Systems

$$x_{t+1} = f(x_t, u_t)$$



Defn. a 'policy' maps {state x_t} $\mapsto$ actions u_t

$x_{t+1} = f(x_t, u_t)$

nonlinear

Behavior Cloning

states x_t = state of robot + object

inputs u_t = robot **action**

'continuous tokens'

**caveat: focus on states*

Themes

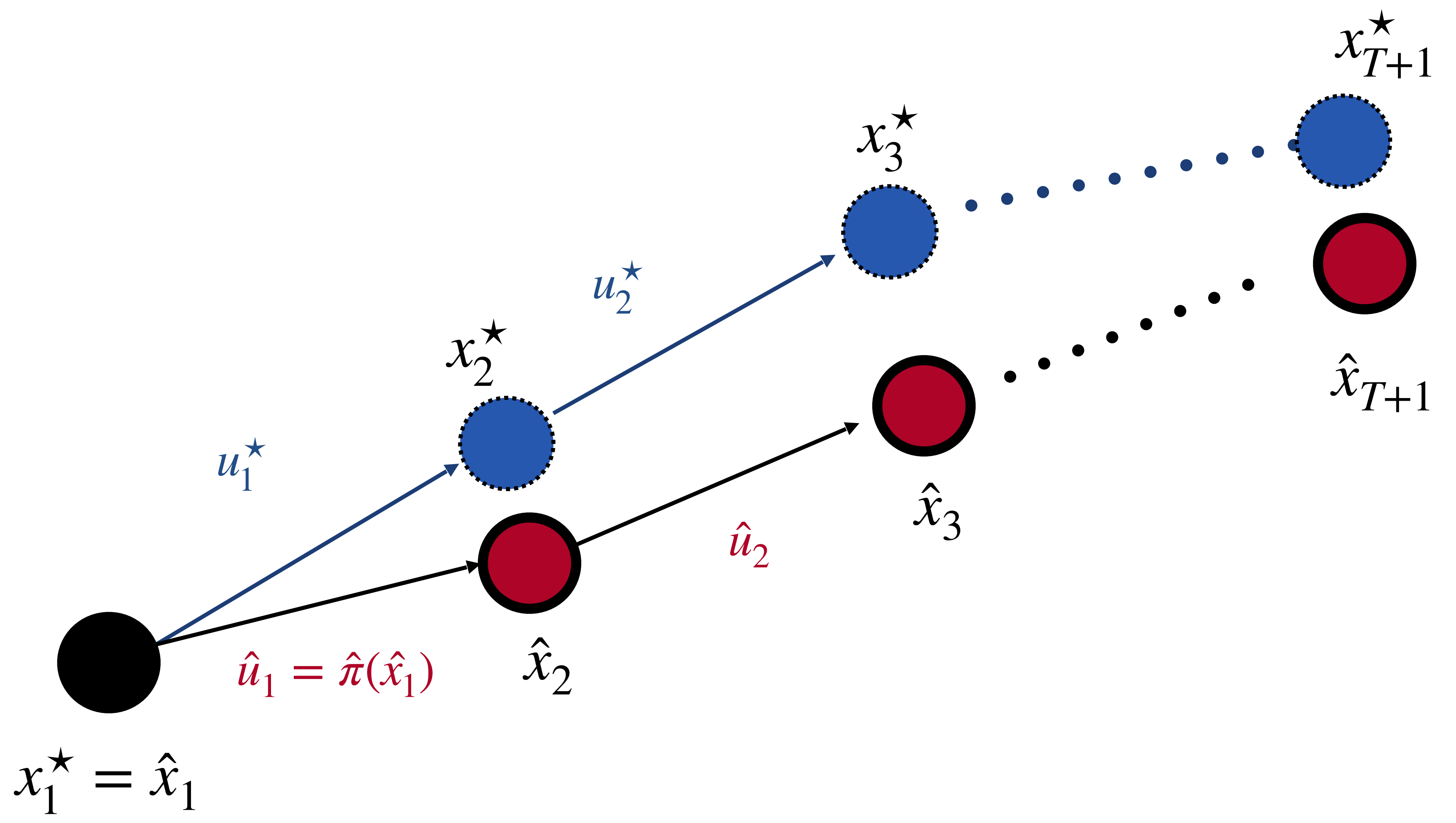
1. How things go out-of-distribution
2. How this differs from discrete tokens
3. Theoretical Guarantees
4. Some applications

$x_{t+1} = f(x_t, u_t)$

Behavior Cloning

demonstrator $u^\star \sim \pi^\star(x^\star)$

robot imitation.



Algorithm Template:

(1) Collect **N** expert demonstrations $(x^\star_{1:T}, u^\star_{1:T})$

(2) train a **predictive model** to predict

$$\hat{\pi}(u \mid x) \approx \mathbb{P}[u^\star_t = u \mid x^\star_t = x]$$

1. **supervised learning** from **demonstration**
2. no **reward model** (given or inferred).

disclaimer: other approaches exist.

Behavior Cloning meets Generative Models

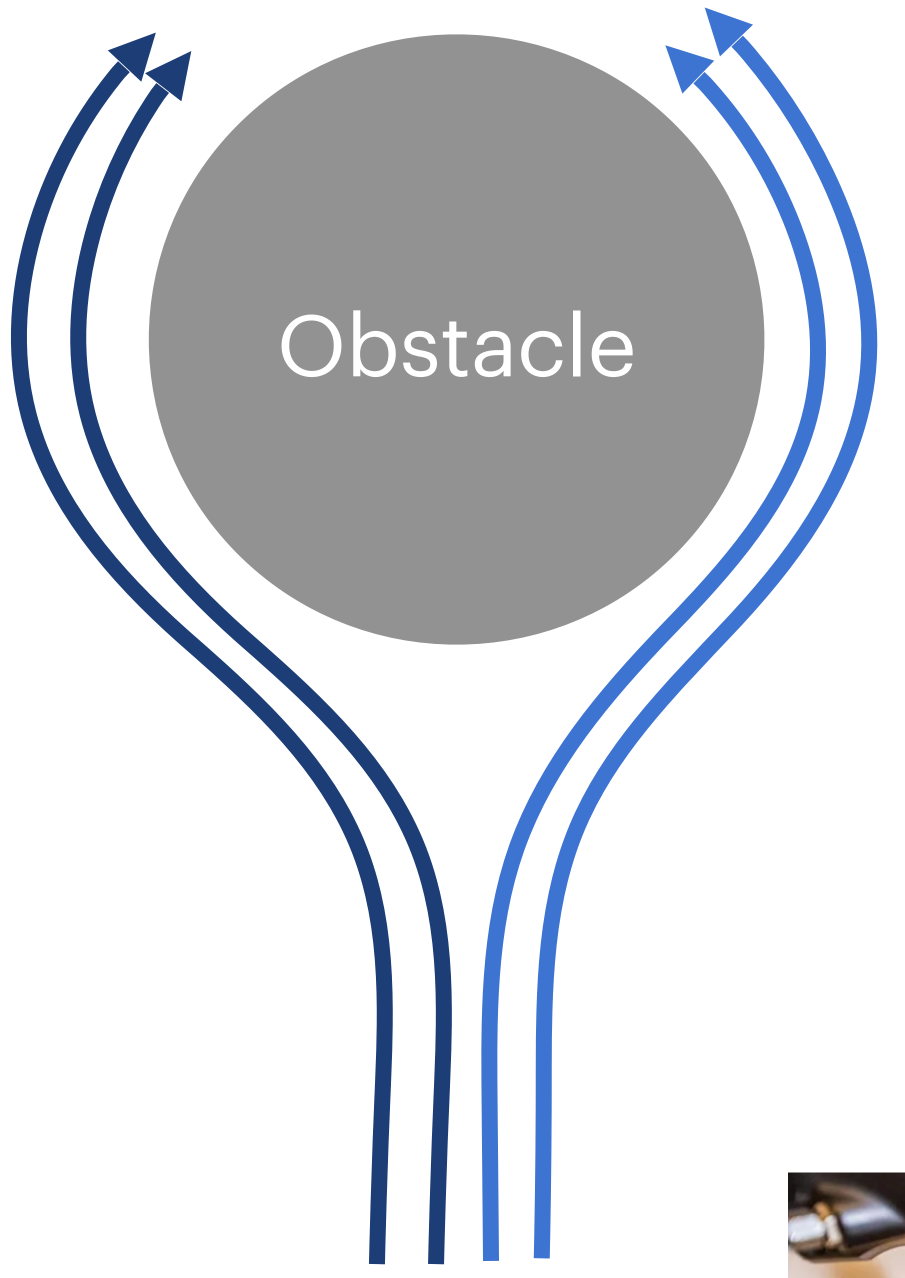
(as supervised learners)

Algorithm Template:

(1) Collect **N** expert demonstrations $(x^\star_{1:T}, u^\star_{1:T})$

(2) train a **generative model** to predict

$$\hat{\pi}(u \mid x) \approx \mathbb{P}[u^\star_t = u \mid x^\star_t = x]$$



*multi-modal **expert** data*

Left Mode

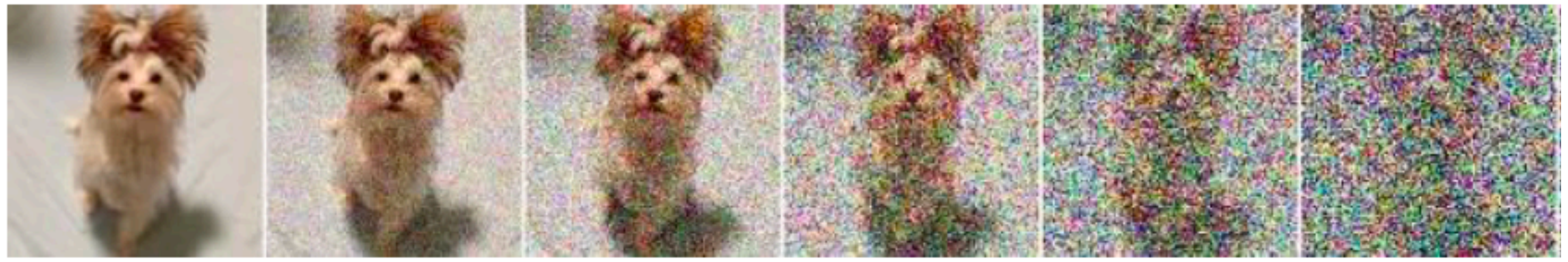
Right Mode

$\pi : x \mapsto u \sim P(x)$

'conditional sampling'

$\pi : x \mapsto f(x) + \text{ noise}$

'mean parametrization'



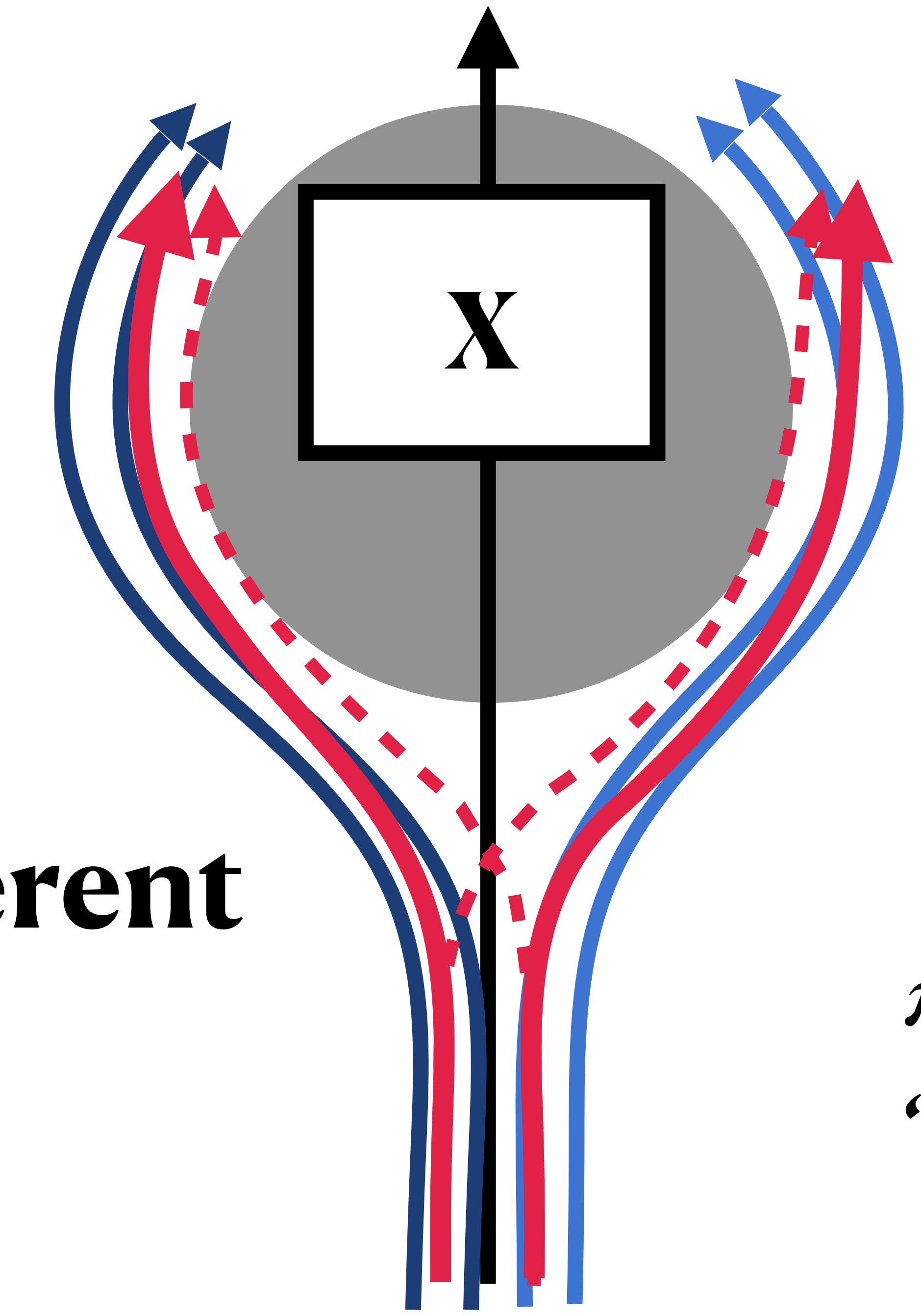
(e.g. Diffusion)

multiple strategies

Behavior Cloning meets Generative Models

Hypothesis 1: Condition sampling models can fit complex data distributions (‘realizability’)

Hypothesis 2: Condition sampling allow for different out-of-distribution inductive biases.



multi-modal ***expert*** *data*

Left Mode

Right Mode

$\pi : x \mapsto f(x) +$ noise

‘mean parametrization’

$\pi : x \mapsto u \sim P(x)$

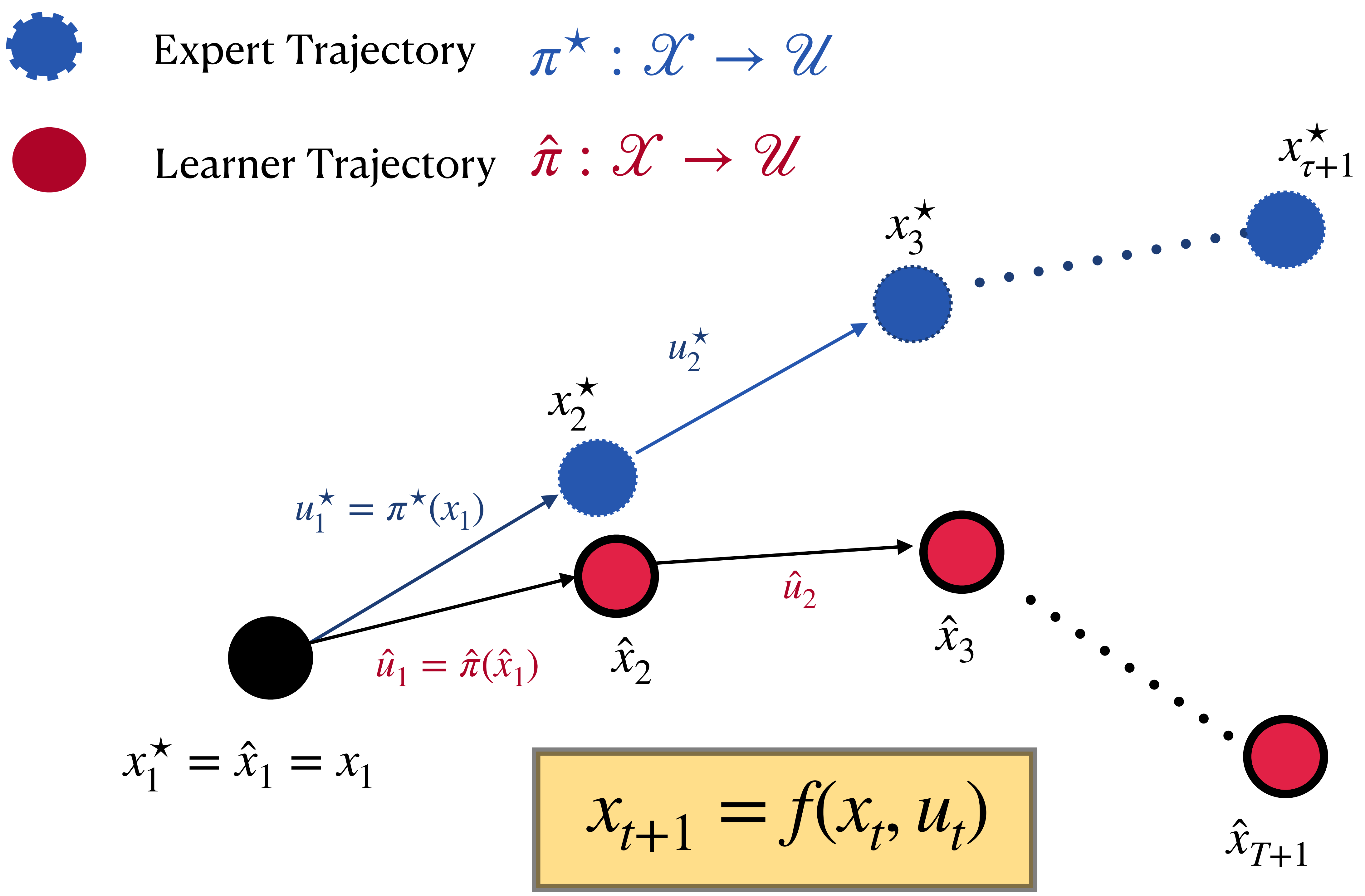
‘conditional sampling’

How **distribution shift** arises

Goal: Make **trajectory distance small**

$$\mathrm{dist}(x^\star_{1:T}, \hat{x}_{1:T}) = \max_t \|x^\star_t - \hat{x}_t\|$$

*(**deterministic** policies, in expectation over initial condition)*



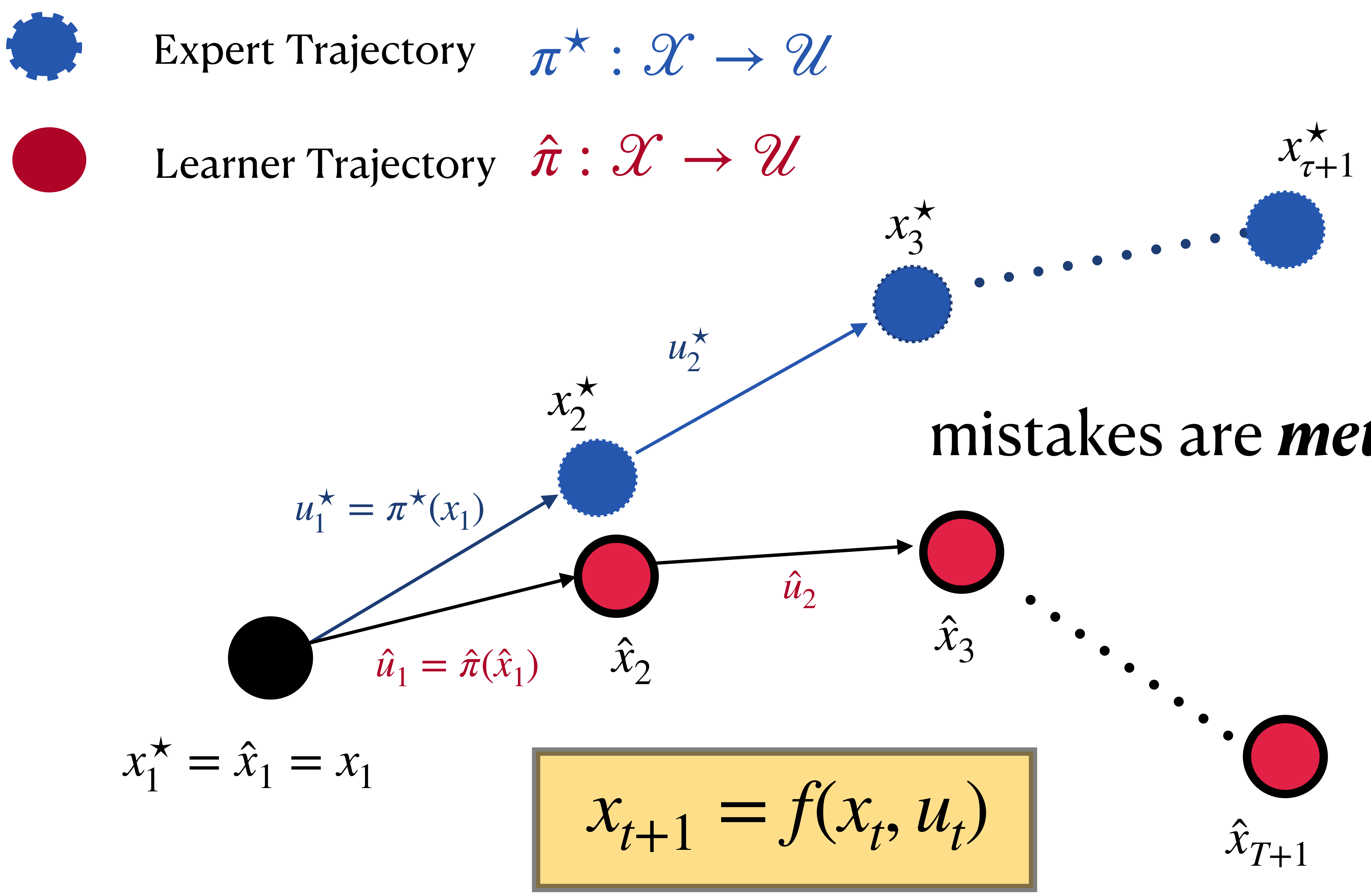
Challenge A: Error accumulates over time steps, *possibly **exponentially** in **horizon***!

Challenge B: After error has accumulated, we are now **out of distribution.**

How this differs from discrete tokens

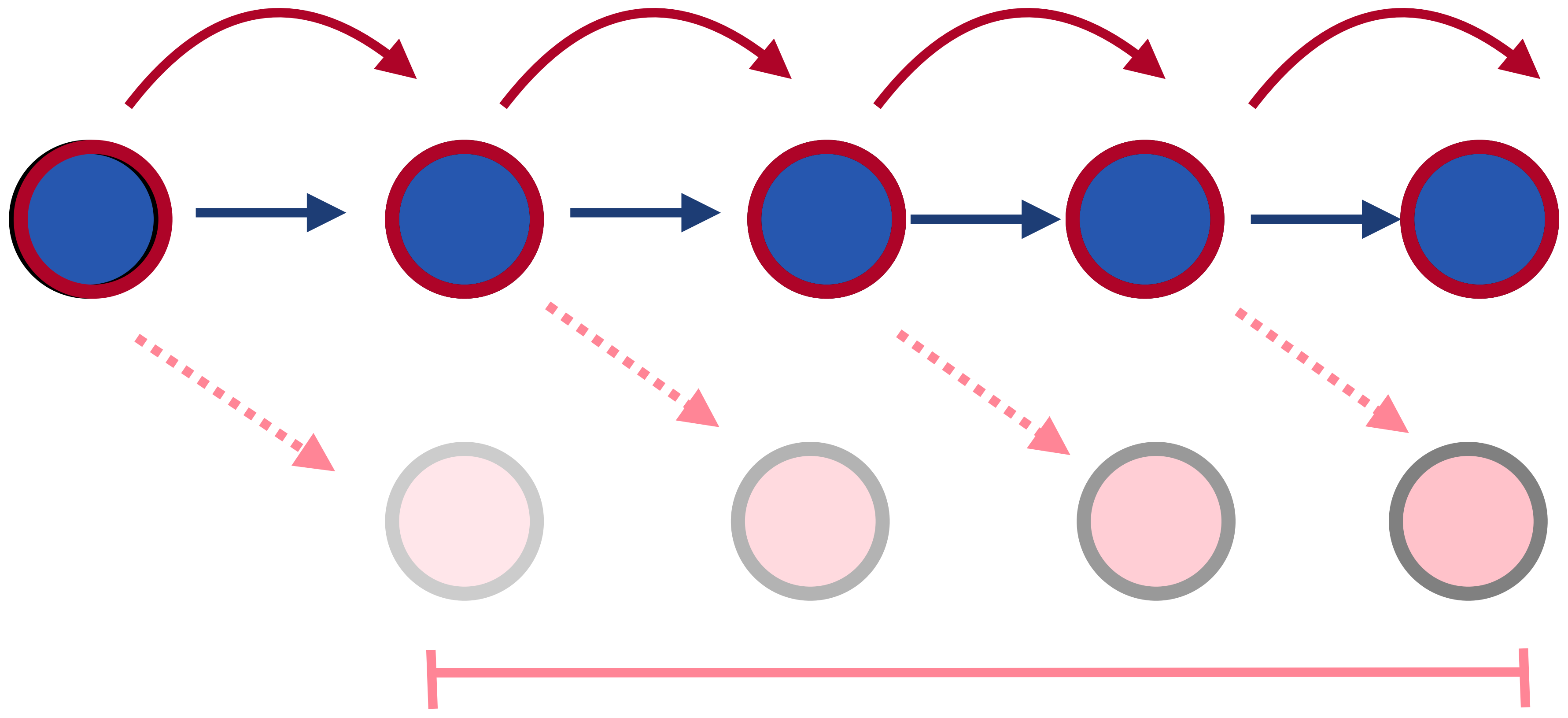
Goal: Make **trajectory distance small**

$$\mathrm{dist}(x^\star_{1:T}, \hat{x}_{1:T}) = \max_t \|x^\star_t - \hat{x}_t\|$$



mistakes are ***metric.***

Contrast: Discrete Behavior Cloning



probabilistic mistakes *accumulate* ***at most linearly.***

(e.g. **DAGGER,** see also Foster '24 et al.)

$x_{t+1} = f(x_t, u_t)$

Schematic of Results

Theorem 1 *(informal)*: With **generative-model policies** (conditional sampling), we can imitate **without exponentially compounding error** in **contractive systems**.

Theorem 2 *(informal)*: If **we know the dynamics**, there is a **reduction** to learning in contractive systems

Theorem 3 *(informal)*: If **we don't known the dynamics**, learning is **hard**, even in "incrementally stable" but non-contractive systems.

$x_{t+1} = f(x_t, u_t)$

*all new results

Schematic of Results

Theorem 1 *(informal)*: With **generative-model policies** (conditional sampling), we can imitate **without exponentially compounding error** in **contractive systems**.

Theorem 2 *(informal)*: If **we know the dynamics**, there is a **reduction** to learning in contractive systems

Theorem 3 *(informal)*: If **we don't known the dynamics**, learning is **hard**, even in "incrementally stable systems."

$x_{t+1} = f(x_t, u_t)$

*all new results

Schematic of Results

Theorem 1 *(informal)*: With **generative-model policies** (conditional sampling), we can imitate **without exponentially compounding error** in **contractive systems**.

A. Introduce contractive systems

B. Show just fitting the expert data isn't enough.

C. Introduce an inductive bias, TVC, guarantees imitation.

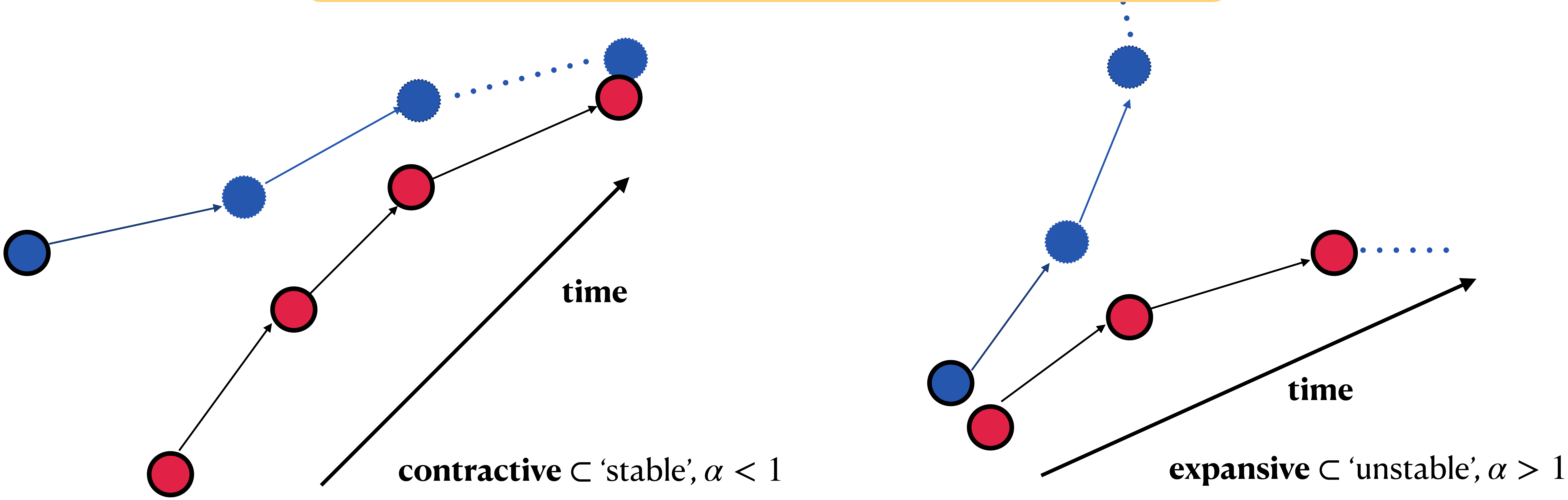
D. Given an algorithmic recommendation to ensure TVC.

$x_{t+1} = f(x_t, u_t)$

Contractive Systems

Definition: We will say a system is (α, β)-**contractive** if

$$\|f(x', u') - f(x, u)\| \leq \alpha \|x - x'\| + \beta \|u - u'\|$$



$x_{t+1} = f(x_t, u_t)$

Contractive Systems

Definition: We will say a system is (α, β)-**contractive** if

$$\|f(x', u') - f(x, u)\| \leq \alpha \|x - x'\| + \beta \|u - u'\|$$

Lemma: If dynamics are (α, β)-**contractive**, given two sequences $(x^\star_{1:T}, u^\star_{1:T})$, $(\hat{x}_{1:T}, \hat{u}_{1:T})$ with $x^\star_1 = \hat{x}_1$, and if $\alpha < 1$, we get

$$\max_{1 \leq t \leq T} \|x^\star_t - \hat{x}_t\| \leq \frac{\beta}{1 - \alpha} \max_{1 \leq t \leq T} \|u^\star_t - \hat{u}_t\|$$

special case of 'stability'

$x_{t+1} = f(x_t, u_t)$

Is Low Training Error Enough?

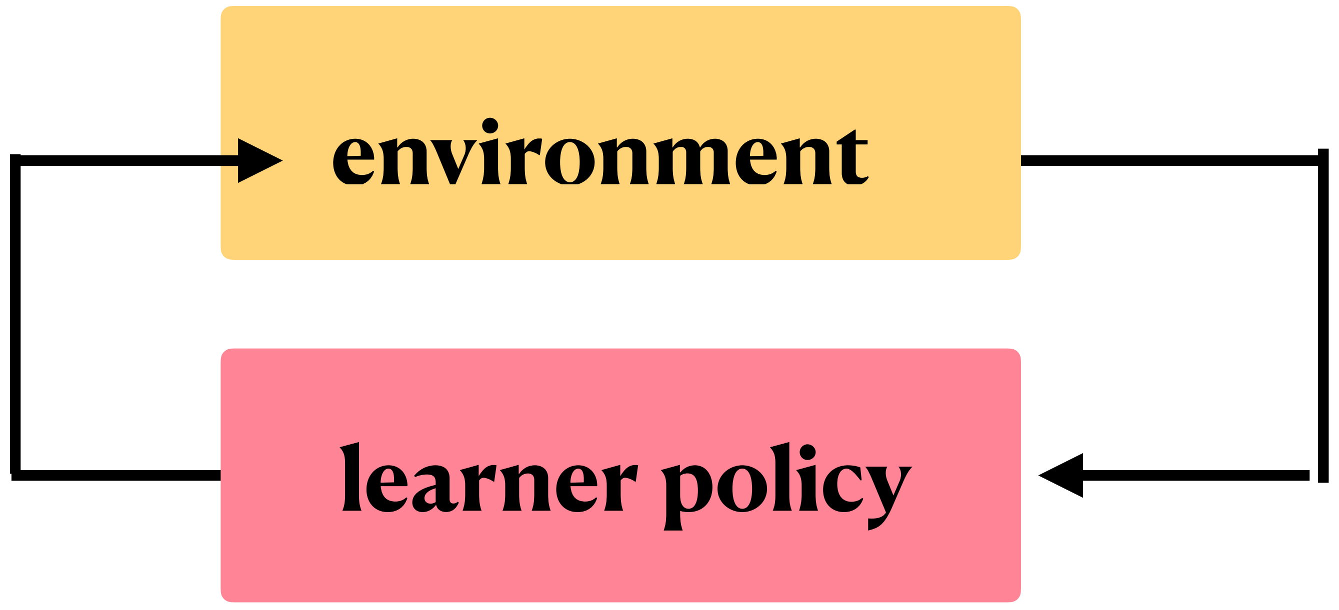
Example *(Contractive, Scalar Dynamics)*:

(a) $f(x, u) = .9x + u$

(b) $\pi^\star(x) = 0$

(c) *training data: "0"-trajectory* $x_1^\star = x_2^\star = \ldots = 0$

'**feedback**': $f(x, \hat{\pi}(x))$



inductive bias creates 'feedback'

Bad Learner Policy: $\hat{\pi}^{\text{Bad}}(x) = .15x + \epsilon$

(a) *For all training x,* $\pi^\star(x) - \hat{\pi}^{\text{Bad}}(x) = \epsilon$

(b) *On deployment,* $\hat{x}_t \geq (1.05)^t \epsilon = e^{\Omega(t)} \cdot \epsilon$

$x_{t+1} = f(x_t, u_t)$

Is Low Training Error Enough?

Example *(Contractive, Scalar Dynamics):*

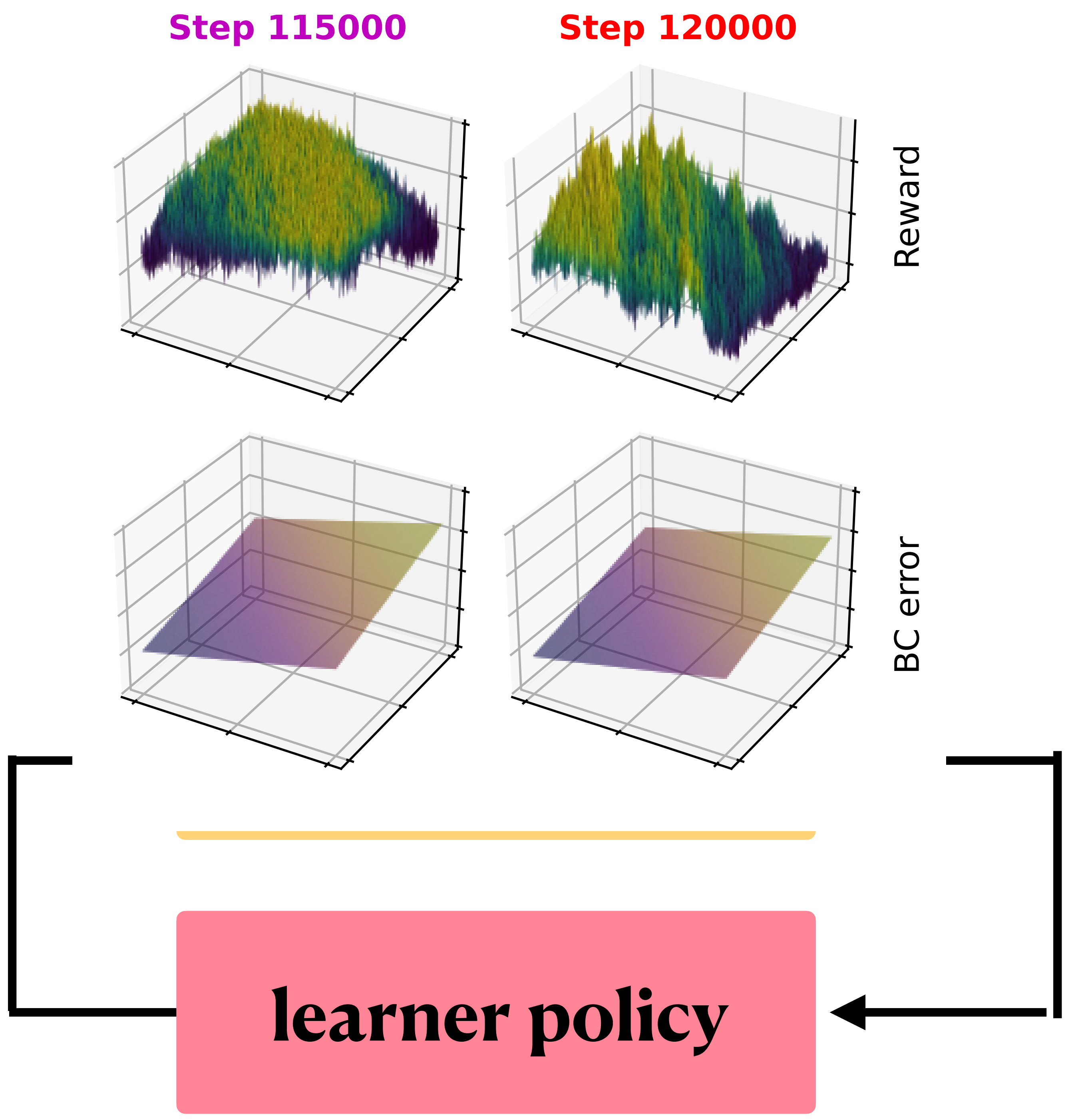
(a) $f(x, u) = .9x + u$

(b) $\pi^\star(x) = 0$

(c) *training data: "0"-trajectory* $x_1^\star = x_2^\star = \ldots = 0$

Bad Learner Policy: $\hat{\pi}^{\text{Bad}}(x) = \boxed{.15x} + \epsilon$

Butterfly Effects of SGD, Block '24



inductive bias creates 'feedback'

can be improved by better data coverage ....

A different inductive bias.

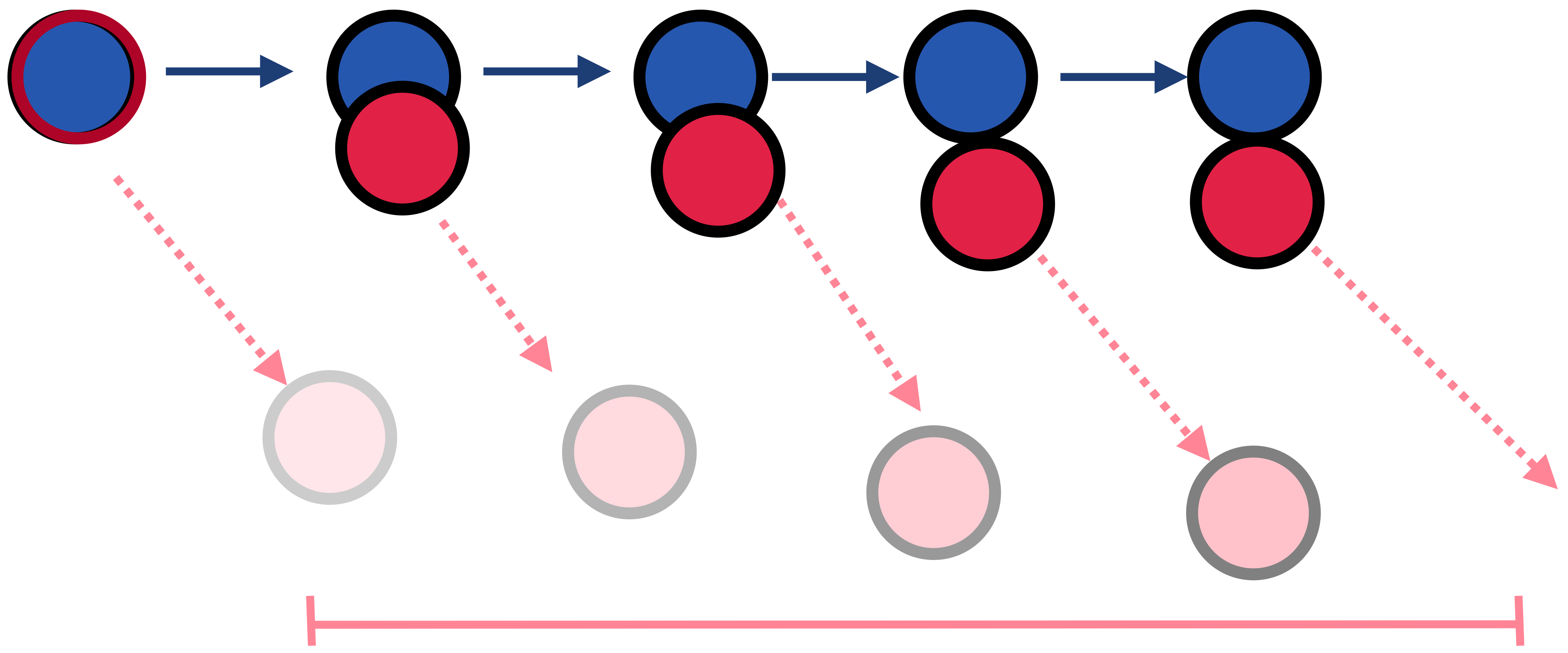
Example *(Contractive, Scalar Dynamics):*

$f(x, u) = .9x + u, \pi^\star(x) = 0$

Not-So-Bad Learner Policy: $\hat{\pi}^{\mathrm{NSB}}(x) = \mathrm{Bernoulli}(\min\{1, .15x\}) + \epsilon$

(a) *For all training x,* $\pi^\star(x) - \hat{\pi}(x) = \epsilon$

(b) *On deployment,* $\hat{x}_t \leq O(\epsilon)$ w.p. $1 - O(t\epsilon)$



probabilistic mistakes *accumulate* ***at most linearly.***

‘Discrete Token Error’ ?

Example *(Contractive, Scalar Dynamics)*:

$f(x, u) = .9x + u, \pi^\star(x) = 0$

Not-So-Bad Learner Policy: $\hat{\pi}^{\mathrm{NSB}}(x) = \mathrm{Bernoulli}(\min\{1, .15x\}) + \epsilon$

↑

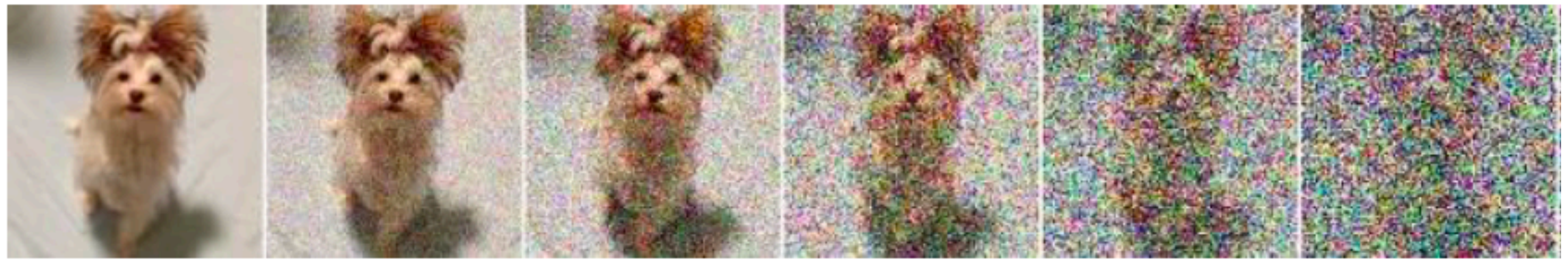
convert ‘metric mistakes’ into ‘probabilistic mistakes’

‘Discrete Token Error’?

Example *(Contractive, Scalar Dynamics):*

$f(x, u) = .9x + u, \pi^\star(x) = 0$

Not-So-Bad Learner Policy: $\hat{\pi}^{\mathrm{NSB}}(x) = \mathrm{Bernoulli}(\min\{1, .15x\}) + \epsilon$



Generative models

For small enough **x,** $\hat{\pi}^{\mathrm{Bad}}(x) = \mathbb{E}[\hat{\pi}^{\mathrm{NSB}}(x)] = .15x + \epsilon$ **is the OG bad policy.**

Total Variation Continuity

Definition: We say $\pi(x)$ is **L-TVC** if $\mathrm{TV}(\pi(x), \pi(x')) \leq L\|x - x'\|$

$$\mathrm{TV}(P, Q) := \inf_{(X_P, X_Q)\sim\mu} \Pr\left[X_P \neq X_Q\right]$$

Example 1: $\hat{\pi}^{\mathrm{NSB}}(x) = \mathrm{Bernoulli}(\min\{1, .15x\}) + \epsilon$ **is** $L = .15$ **TVC**

Example 2: $\hat{\pi}^{\mathrm{Bad}}(x) = \mathbb{E}[\hat{\pi}^{\mathrm{NSB}}(x)] = .15x + \epsilon$ **is not TVC**

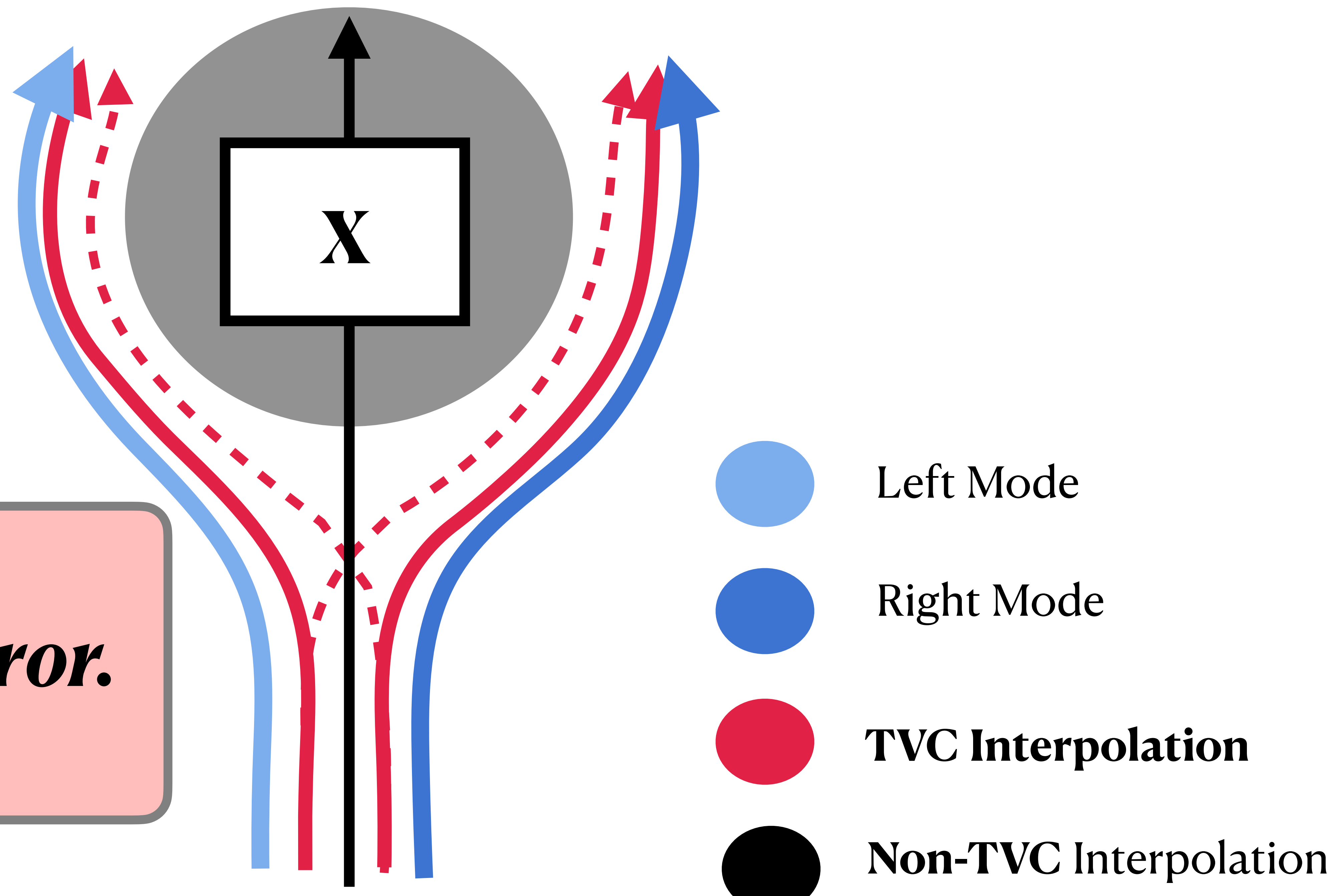
Total Variation Continuity

Definition: We say $\pi(x)$ is **L-TVC** if $\mathrm{TV}(\pi(x), \pi(x')) \leq L\|x - x'\|$

$$\mathrm{TV}(P, Q) := \inf_{(X_P, X_Q)\sim\mu} \Pr\left[X_P \neq X_Q\right]$$

TVC is the opposite of **mode-collapse**

We will show *TVC Policies have low execution error.*



*contractive dynamics

$x_{t+1} = f(x_t, u_t)$

Problem Definition

Definition: Let **P** , **Q** be two distribution on the same normed space. We define

$$\mathrm{TV}_{\epsilon}(P, Q) := \inf_{(X_P, X_Q) \sim \mu} \Pr\left[\|X_P - X_Q\| > \epsilon \right]$$

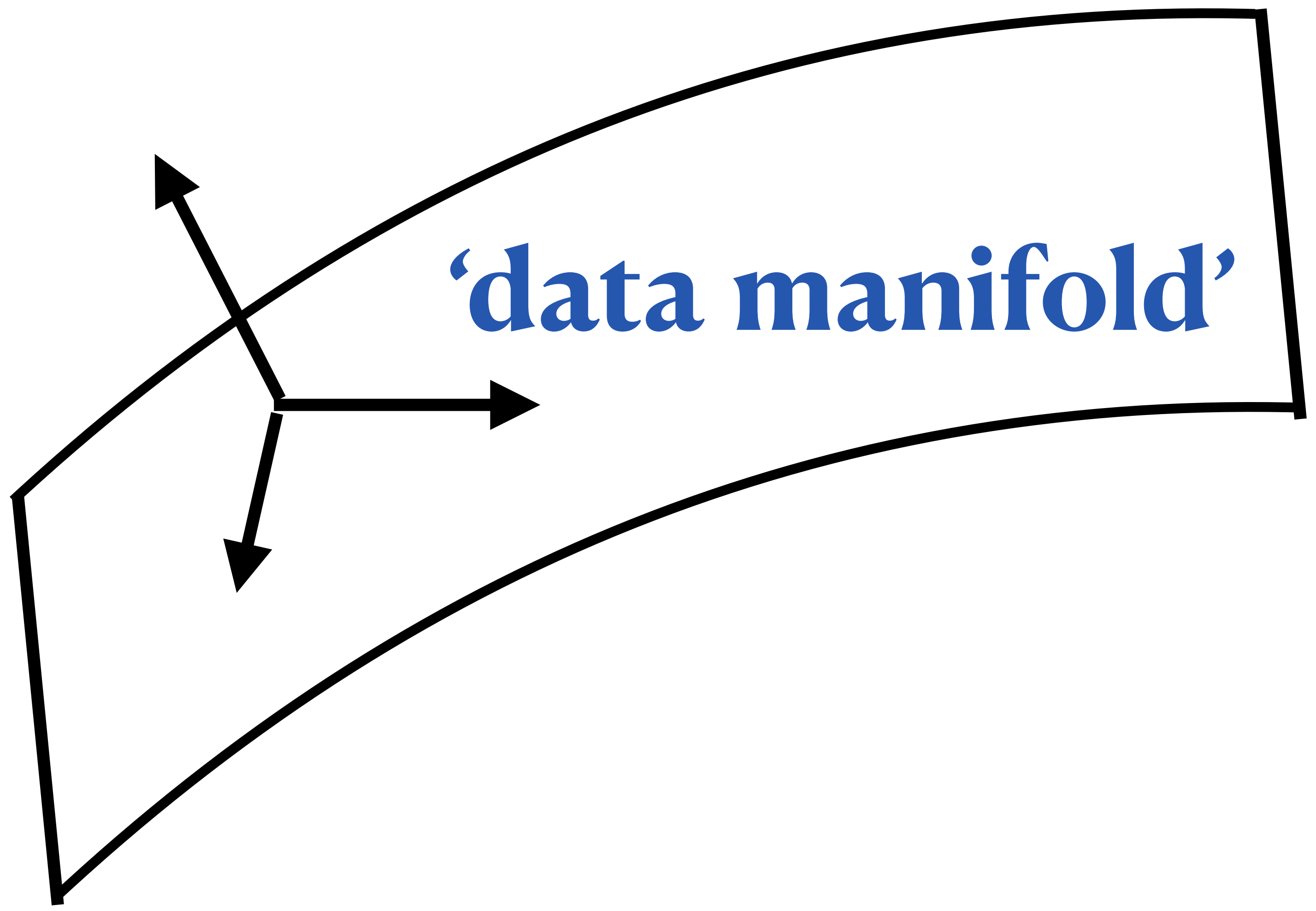
1. 'Optimal Transport Distance', reduces to regular **TV** for $\epsilon = 0$
2. A way of measuring distance between **continuous-valued R.V.s**
3. Like **Wasserstein,** but easier to work with for imitation learning

$x_{t+1} = f(x_t, u_t)$

Problem Definition

Training Error: Suppose we get trajectories $(x_1^\star, u_1^\star, x_2^\star, u_2^\star, \ldots, x_H^\star, u_H^\star)$, $u_t^\star \sim \pi^\star(x_t^\star)$

$$\mathrm{D}_{\mathrm{train},\epsilon}(\hat{\pi} \| \pi^\star) := \max_t \mathbb{E}_{x_t^\star} \mathrm{TV}_\epsilon(\pi^\star(x_t^\star), \hat{\pi}(x_t^\star))$$ **(can be made small w/ DDPM)**



(exact TV not tractable)

$x_{t+1} = f(x_t, u_t)$

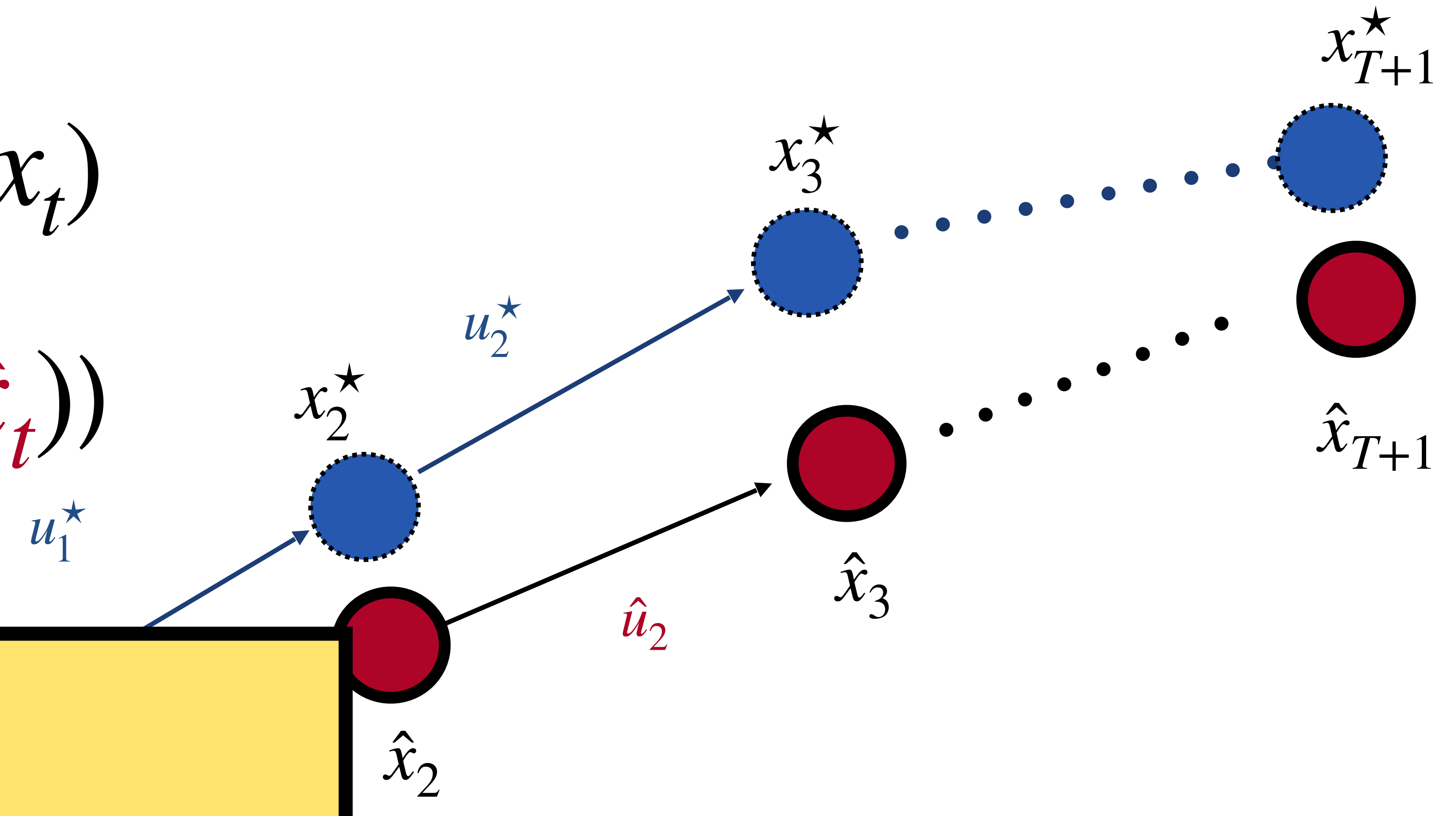
Problem Definition

Training Error: Suppose we get trajectories $(x_1^\star, u_1^\star, x_2^\star, u_2^\star, \ldots, x_H^\star, u_H^\star)$, $u_t^\star \sim \pi^\star(x_t^\star)$

$$\mathrm{D}_{\mathrm{train},\epsilon}(\hat{\pi} \,\|\, \pi^\star) := \max_t \mathbb{E}_{x_t^\star} \mathrm{TV}_\epsilon(\pi^\star(x_t^\star), \hat{\pi}(x_t^\star))$$

Test Error: We roll out $(\hat{x}_1, \hat{u}_1, \hat{x}_2, \hat{u}_2, \ldots, \hat{x}_H, \hat{u}_H)$, $\hat{u}_t \sim \hat{\pi}(x_t)$

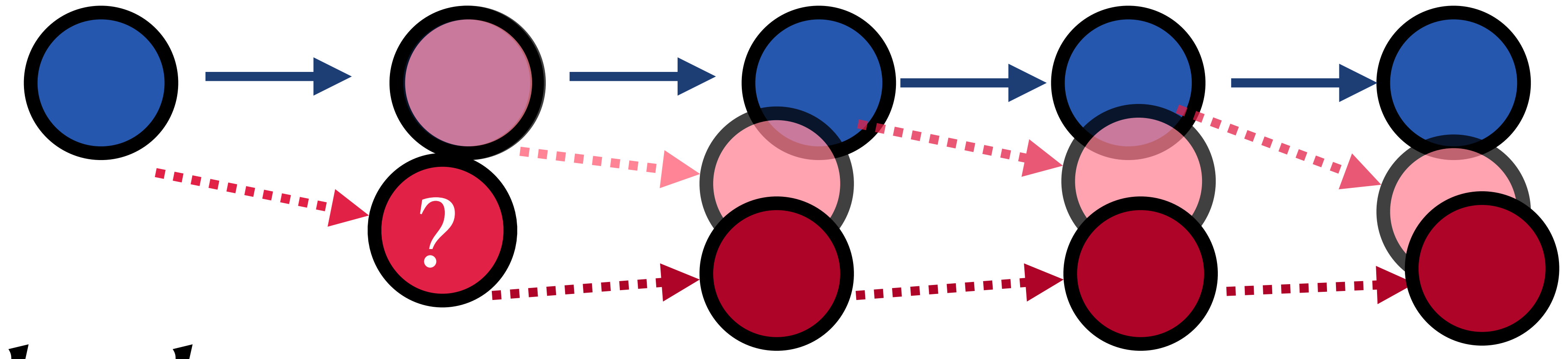
$$\mathrm{D}_{\mathrm{test},\epsilon}(\hat{\pi} \,\|\, \pi^\star) := \max_t \mathrm{TV}_\epsilon(\mathbf{Law}(x_t^\star), \mathbf{Law}(\hat{x}_t))$$



Goal: $D_{\mathrm{test},\epsilon} \leq \mathrm{poly}(H) \cdot D_{\mathrm{train},\epsilon'}$

A First Guarantee

Theorem: If $\hat{\pi}$ is **L-TVC**,

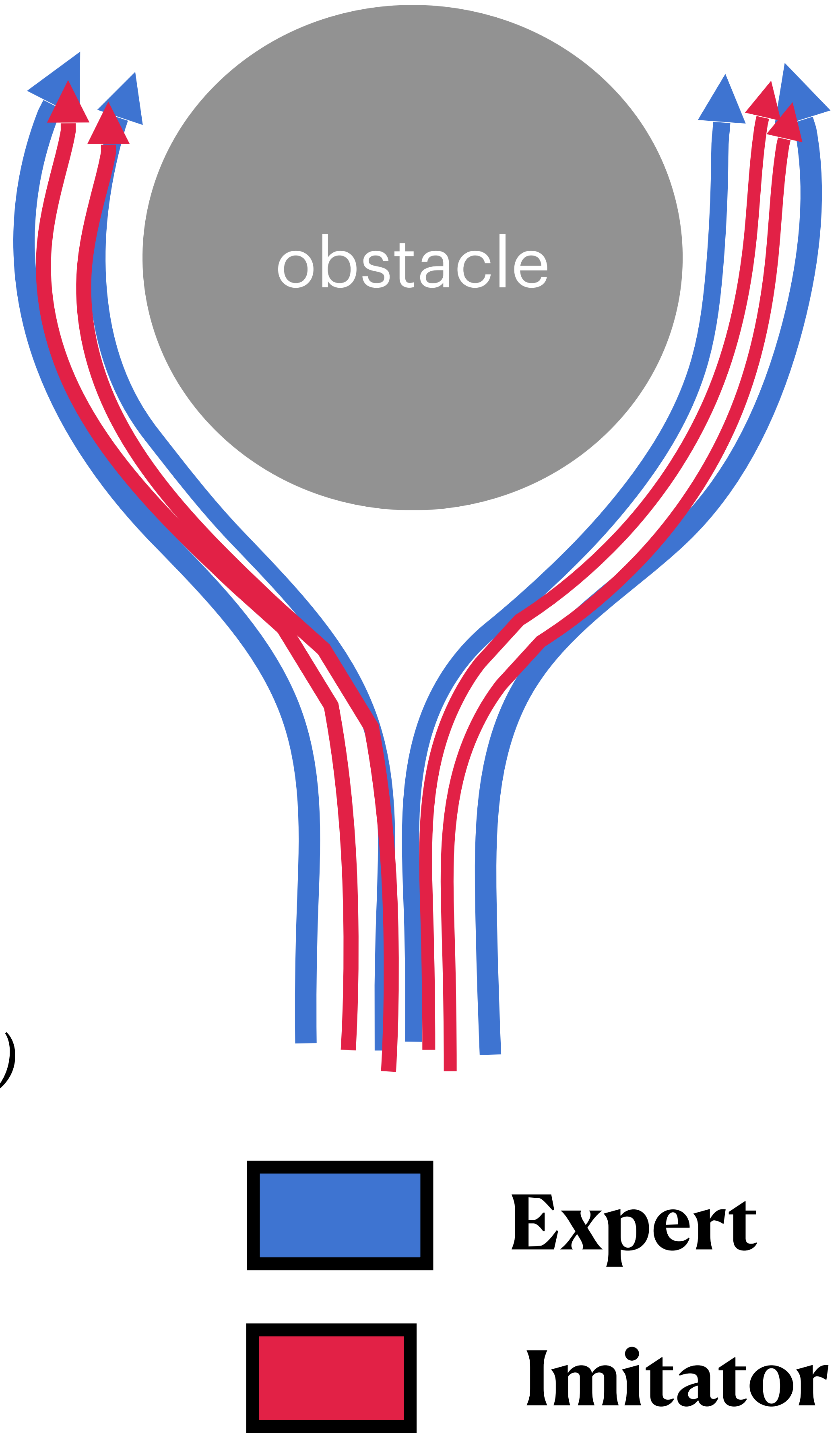


Proof Sketch:

*(1) TVC implies **coupling s.t.*** $\mathbb{P}[\hat{u}_t \sim \hat{\pi}(\hat{x}_t) \neq \hat{u}'_t \sim \hat{\pi}(x^\star_t)] \leq L\epsilon$ *(change of measure)*

(2) Supervised Learning ensures that $\hat{u}'_t \sim \hat{\pi}(\hat{x}^\star_t) \approx \pi^\star(x^\star_t)$

*(3) Contractive of dynamics implies errors **compound** by at most **c**-factor*



A Recap

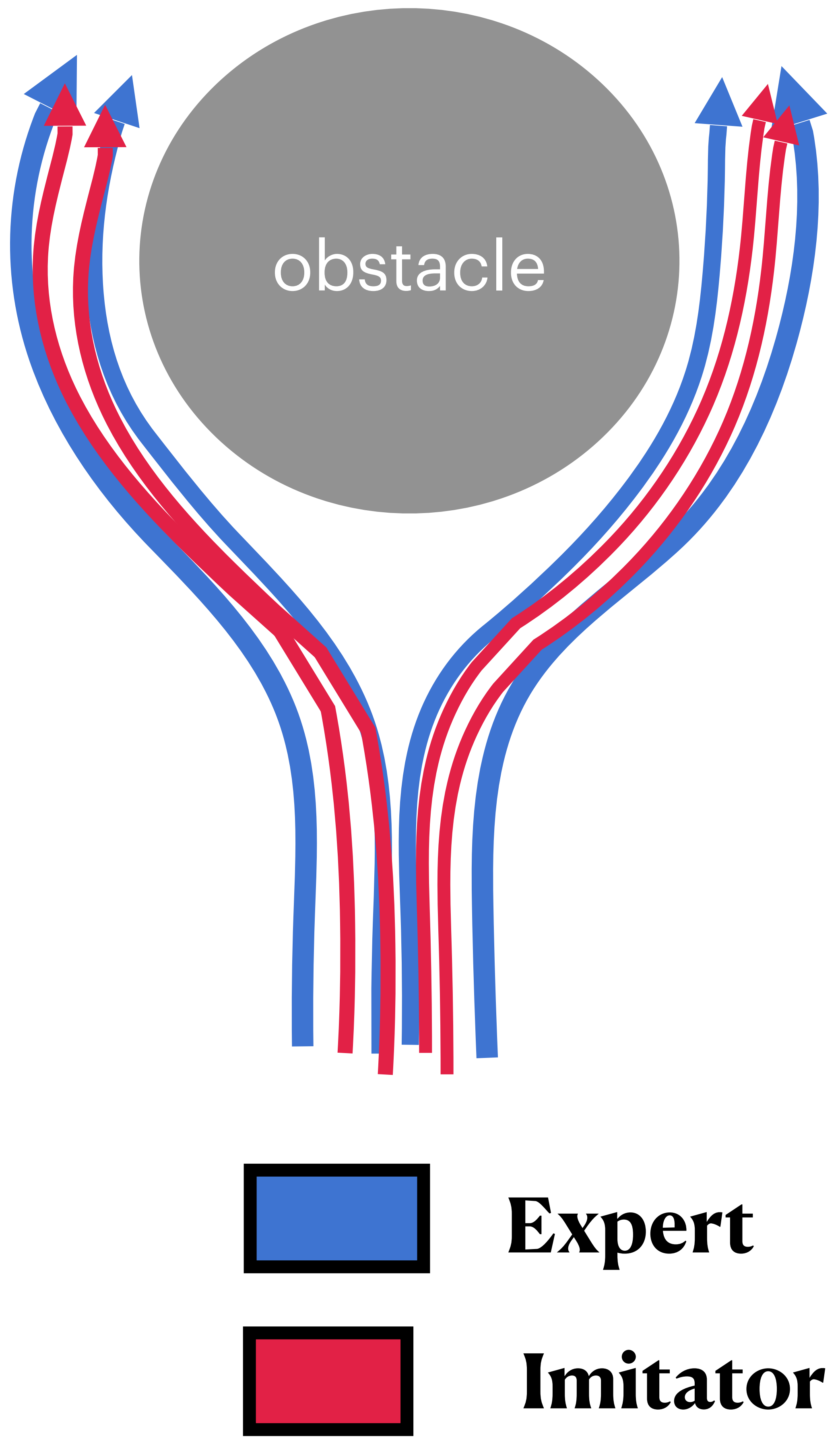
Theorem: If $\hat{\pi}$ is **L-TVC**, and system is $(1 - c^{-1}, O(1))$ contractive

$$D_{\text{test},\epsilon}(\hat{\pi} \| \pi^\star) \leq O(cLH) \cdot D_{\text{train},\epsilon/c}(\hat{\pi} \| \pi^\star)$$

(1) **Distribution Shift can be bad in continuous-state BC**

(2) **TVC** + **Contractive Dynamics*** gets us around the issue

TVC is a nice inductive bias. **By how do we get it?**



Replica Noising.

TVC via Noising

Elementary Lemma: Let $\hat{\pi} : x \in \mathbb{R}^d \mapsto \Delta\left(\mathcal{U}\right)$

Define **smoothed policy** $\hat{\pi}_\sigma : x \mapsto \hat{\pi} \circ \mathcal{N}(x, \sigma^2 \mathbf{I})$

Then $\hat{\pi}_\sigma$ is $(1/2\sigma)$ - **TVC**

Proof: $\mathrm{TV}(\hat{\pi}_\sigma(x), \hat{\pi}_\sigma(x')) \leq \mathrm{TV}(\mathcal{N}(x, \sigma^2 \mathbf{I}), \mathcal{N}(x', \sigma^2 \mathbf{I}))$ **(Data Processing)**

$$\leq \left(\frac{1}{2}\mathrm{KL}(\mathcal{N}(x, \sigma^2 \mathbf{I}), \mathcal{N}(x', \sigma^2 \mathbf{I}))\right)^{1/2}$$ **(Pinsker)**

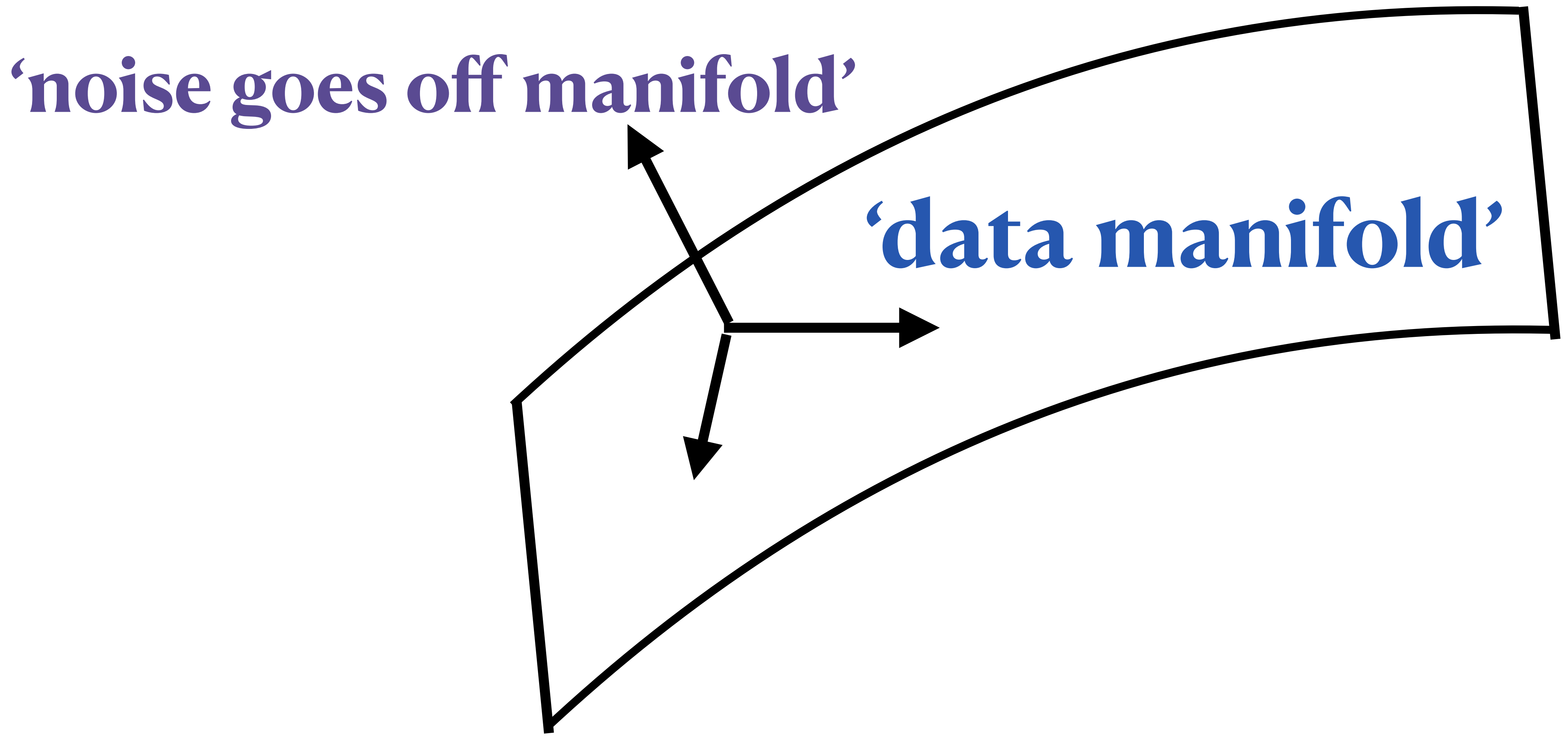
$$= \frac{1}{2\sigma}\|x - x'\|$$ **(Stat Class)**

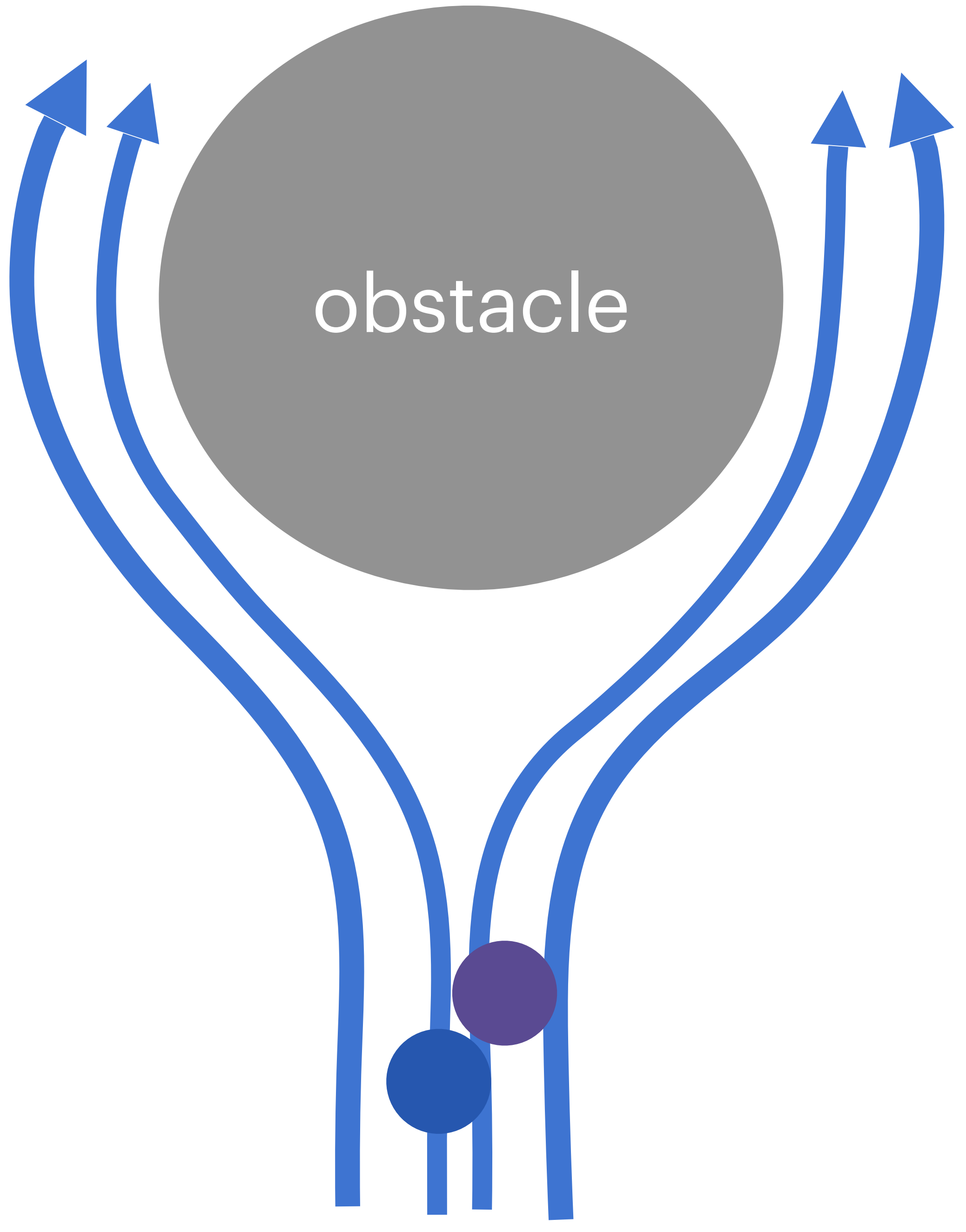
TVC via Noising

Smoothed policy $\hat{\pi}_\sigma : x \mapsto \hat{\pi}(x+\sigma w)$ is $(1/2\sigma)$-**TVC**

1. **Nothing new here** - we know noising gives robustness

2. This might be a **terrible idea**:





Noise might knock me off modes

Replica Noising

Algorithm

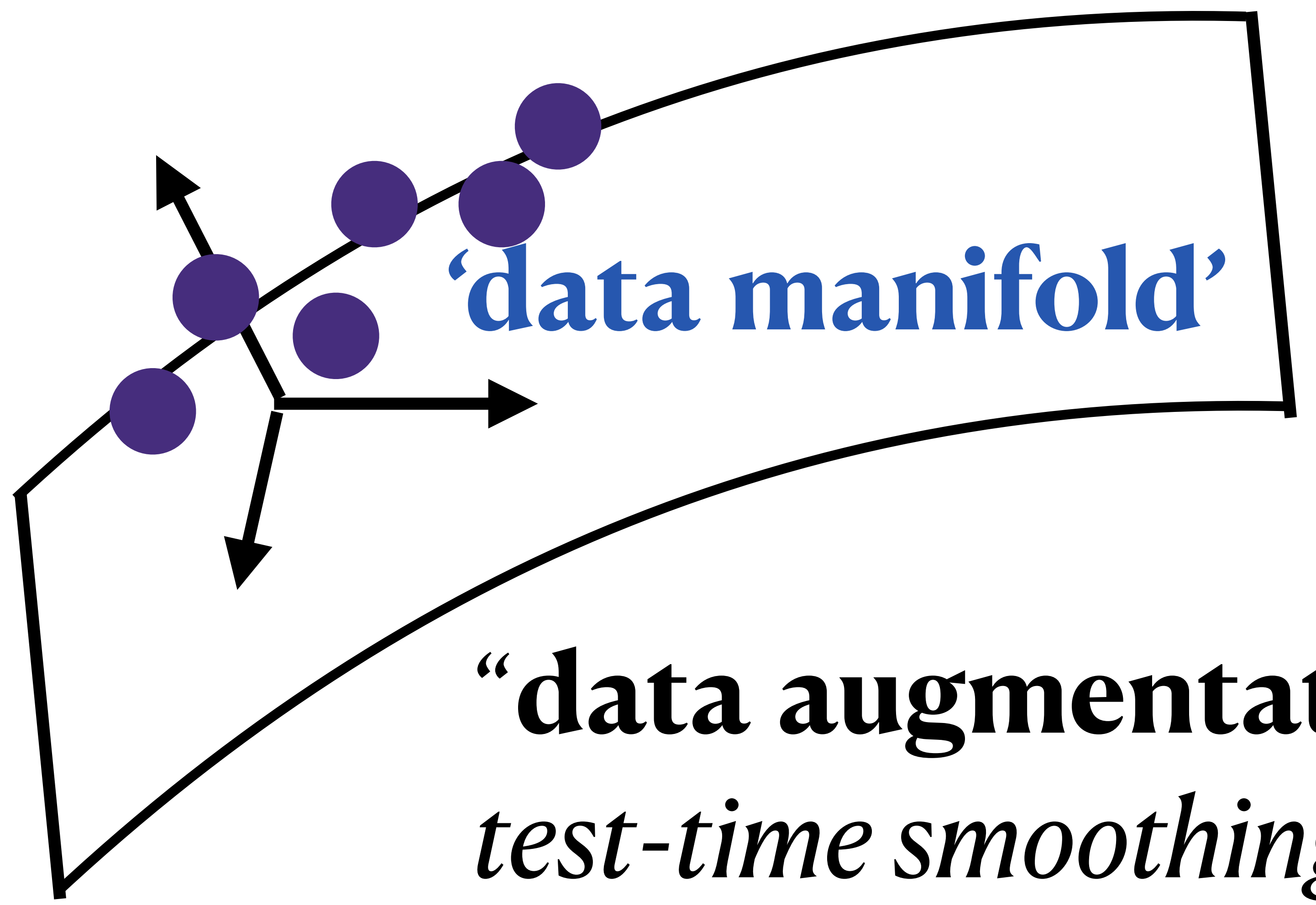
(1) Collect demonstrations $\{x^\star, u^\star \sim \pi^\star(x^\star)\}$

(2) Train **policy** (e.g. Diffusion)

$$\hat{\pi}(x^\star + \sigma w) \approx \mathbb{P}[u^\star \mid x^\star + \sigma w]$$

(3) Deploy $\hat{\pi}_\sigma(x) = \hat{\pi}(x + \sigma w')$

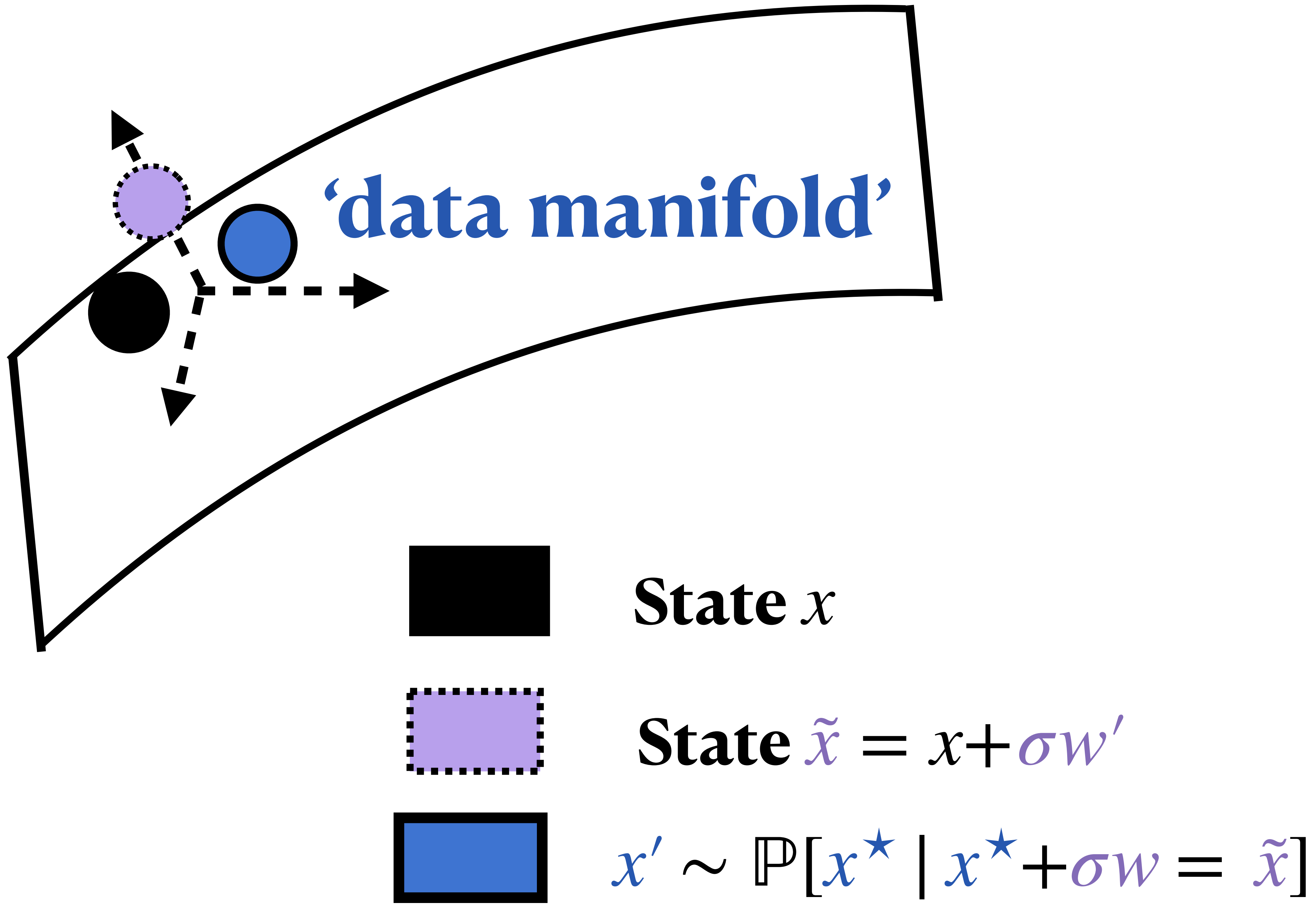
train with **same noise** as **testing**



"**data augmentation** + *test-time smoothing*"

'conditional sampling'

Replica Noising



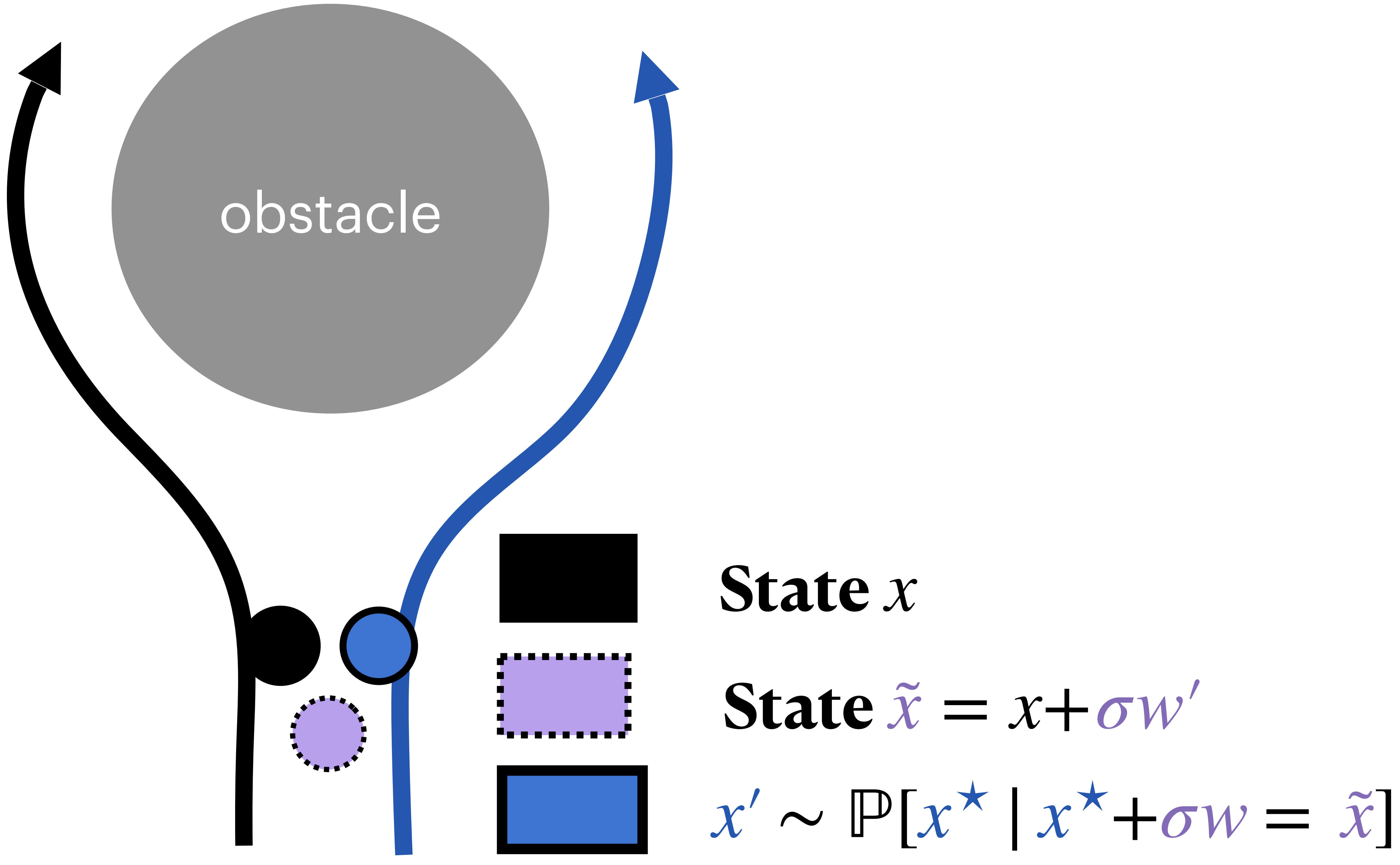
Observation: If $\hat{\pi}(x^\star + \sigma w) = \mathbb{P}[u^\star \mid x^\star + \sigma w]$ is perfect, then,

1. $\hat{\pi}(x) = \pi^\star \circ \mathbb{P}[x^\star \mid x^\star + \sigma w = x]$

2. $\hat{\pi}_\sigma(x) = \pi^\star \circ \mathbb{P}[x^\star \mid x^\star + \sigma w = x + \sigma w']$

$\mathsf{K}^{\mathrm{rep}} : \mathscr{X} \mapsto \Delta(\mathscr{X})$

Replica Noising



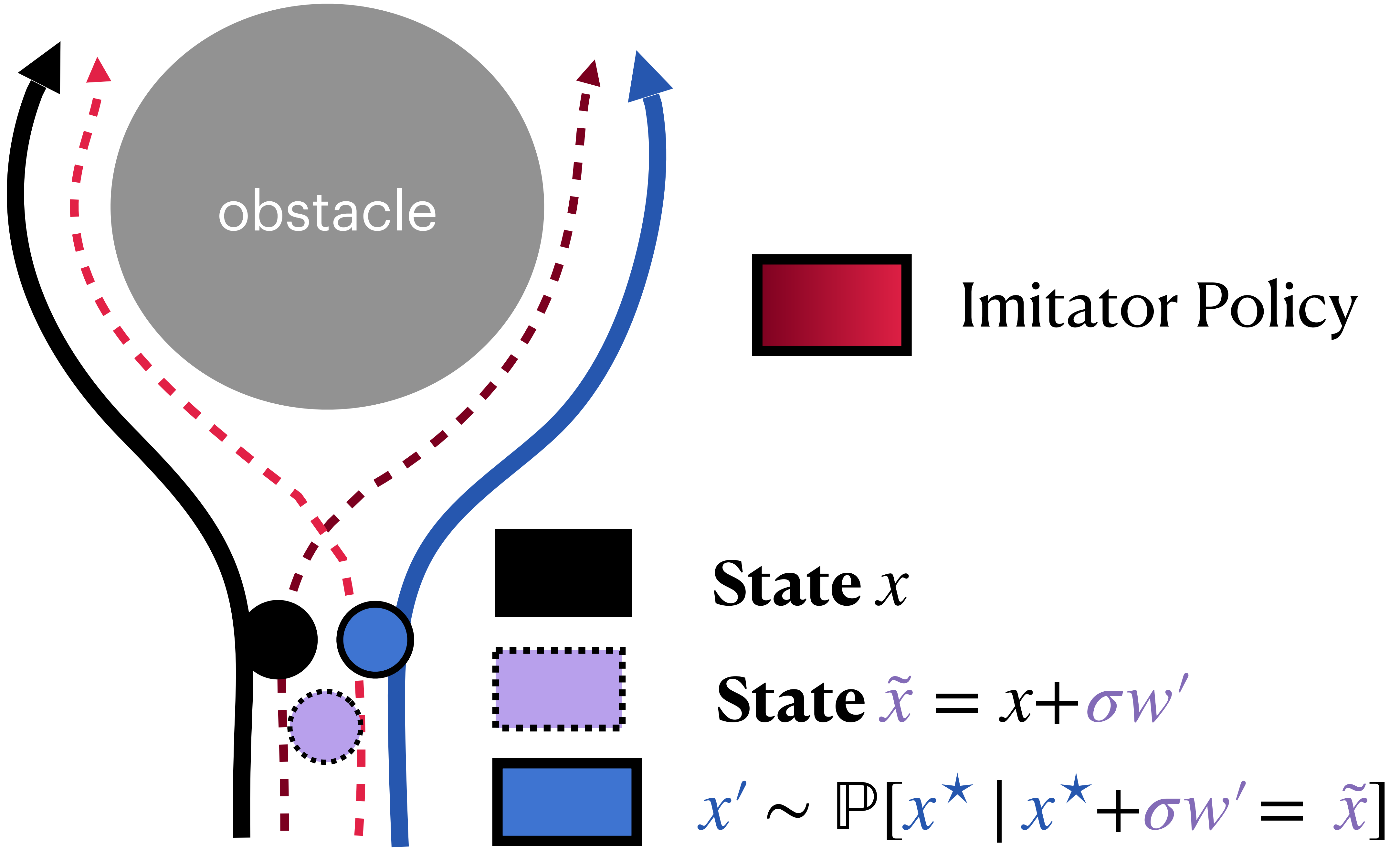
Observation: If $\hat{\pi}(x^\star + \sigma w) = \mathbb{P}[u^\star \mid x^\star + \sigma w]$ is perfect, then,

1. $\hat{\pi}(x) = \pi^\star \circ \mathbb{P}[x^\star \mid x^\star + \sigma w \; = x]$
2. $\hat{\pi}_\sigma(x) = \pi^\star \circ \mathbb{P}[x^\star \mid x^\star + \sigma w \; = x + \sigma w']$

$\mathsf{K}^{\mathrm{rep}} : \mathscr{X} \mapsto \Delta(\mathscr{X})$

proof via more complex coupling argument using the ***replica property***

Replica Noising



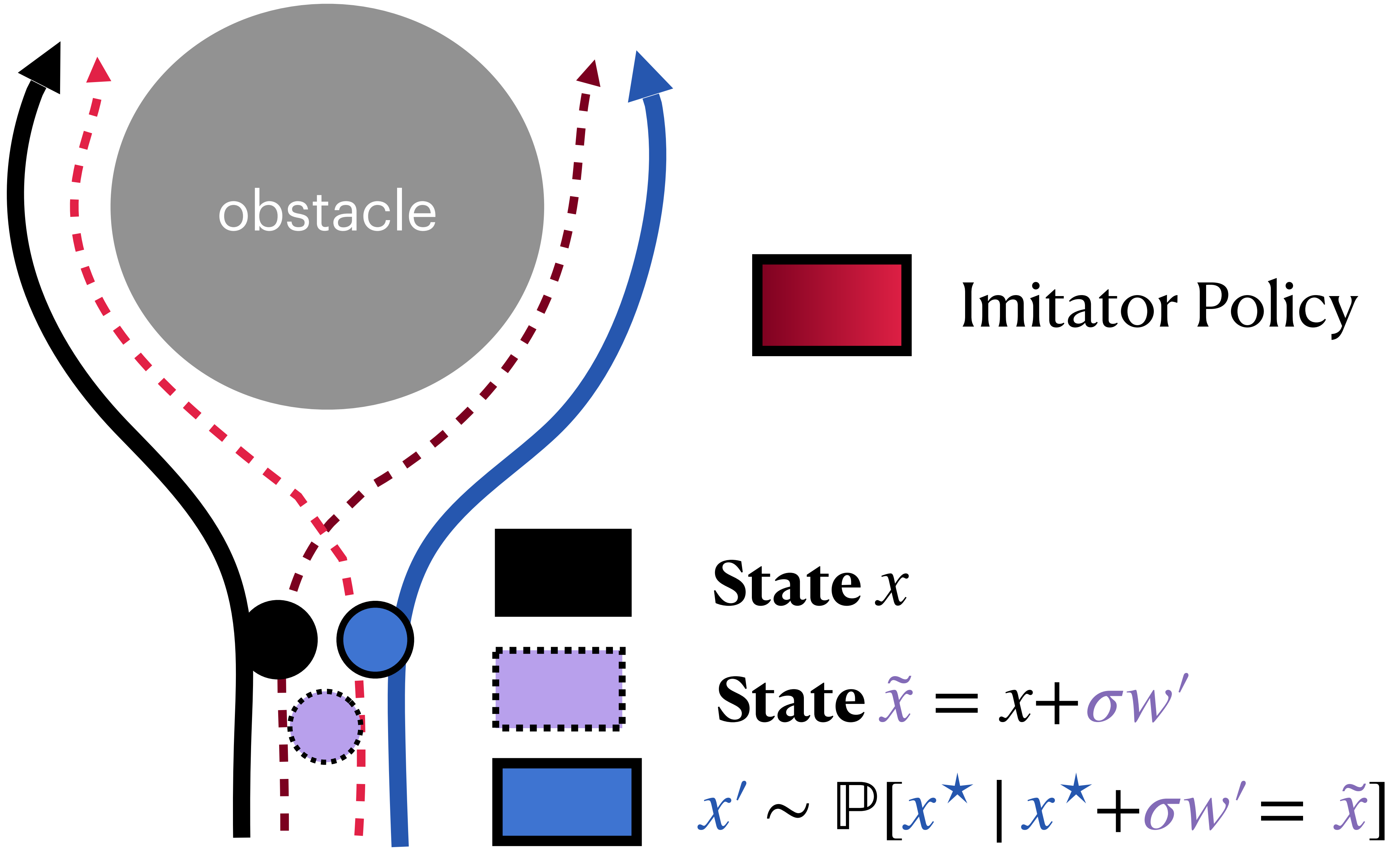
Lemma: Let $x \sim \mathrm{Law}(x^\star)$, and let $x' \sim \mathsf{K}^{\mathrm{rep}}(x)$. Then, (x, x') are

(1) **identically distributed** (and exchangeable)

(2) $\mathbb{P}[\|x - x'\| > 2\sigma\tau] \leq 2\mathbb{P}[\|w\| > \tau]$

With **perfect training,** $\hat{\pi}_\sigma(x) = \pi^\star \circ \mathsf{K}^{\mathrm{rep}}(x)$ is **unbiased** at a distributional level (and **TVC).**

Replica Noising



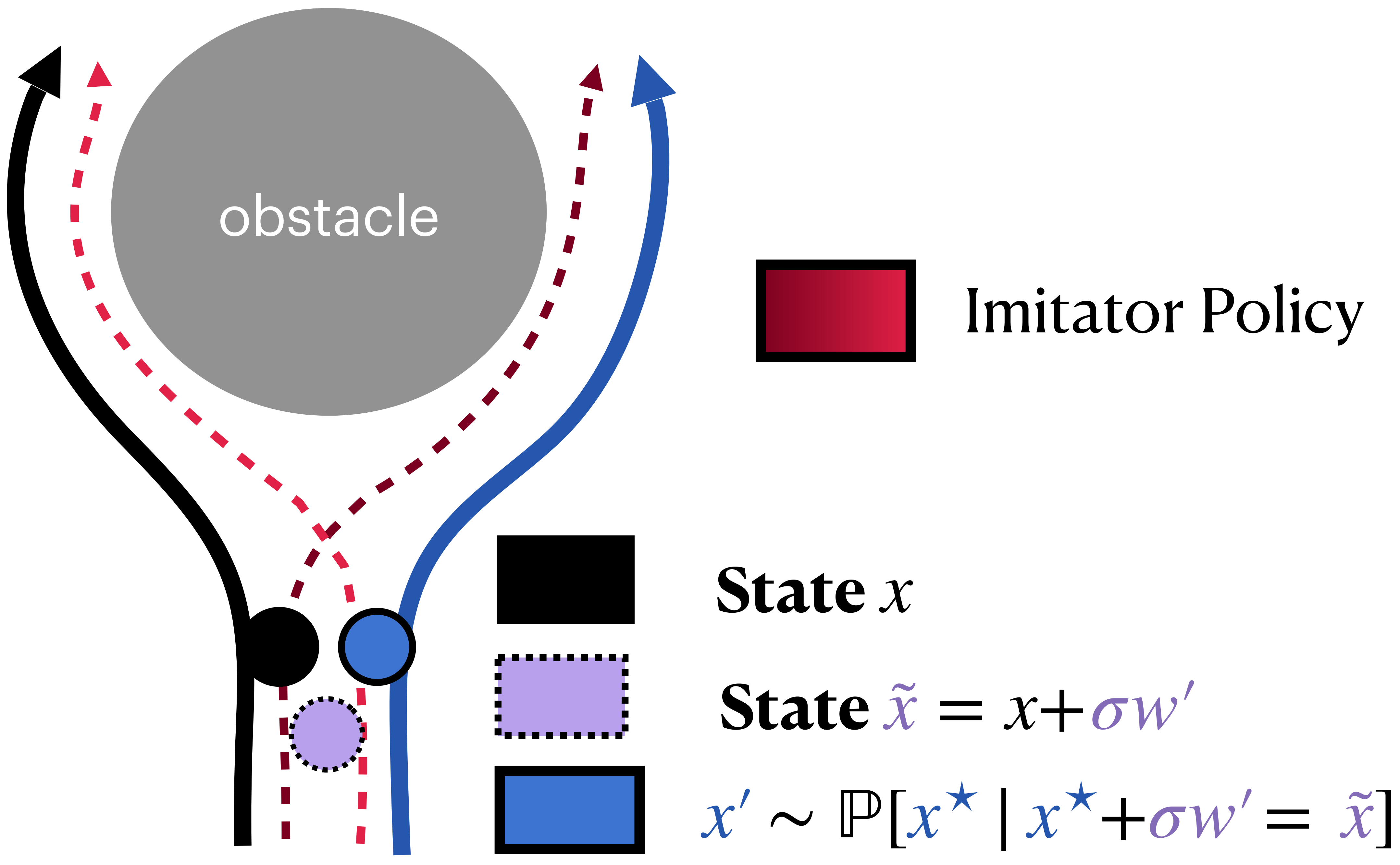
Lemma: Let $x \sim \mathrm{Law}(x^\star)$, and let $x' \sim \mathrm{K}^{\mathrm{rep}}(x)$. Then, (x, x') are

(1) **identically distributed** (and exchangeable)

(2) $\mathbb{P}[\|x - x'\| > 2\sigma\tau] \leq 2\mathbb{P}[\|w\| > \tau]$

This argument requires modeling **distributions,** not simply '**means**'!

Replica Noising



Theorem: tuning $\sigma = \epsilon^{1/2}$, and with some caveats

$$\mathrm{D}_{\text{test},\epsilon}(\hat{\pi} \,\|\, \pi^\star) \leq O(H) \cdot \mathrm{D}_{\text{train},\epsilon^2}(\hat{\pi} \,\|\, \pi^\star)$$

1. **TVC** enforced, not **assumed!**
2. **Degradation** in rates due to noising parameter tradeoff
3. **Noising** introduces the possibility of 'mode swapping'...

*... which means we imitation **joint distributions,** not per-trajectory ones.*

What did we do?

Theorem: tuning $\sigma = \epsilon^{1/2}$, and with some caveats

$$\mathrm{D}_{\text{train},\epsilon}(\hat{\pi} \,\|\, \pi^{\star}) \leq O(H) \cdot \mathrm{D}_{\text{train},\epsilon^2}(\hat{\pi} \,\|\, \pi^{\star})$$

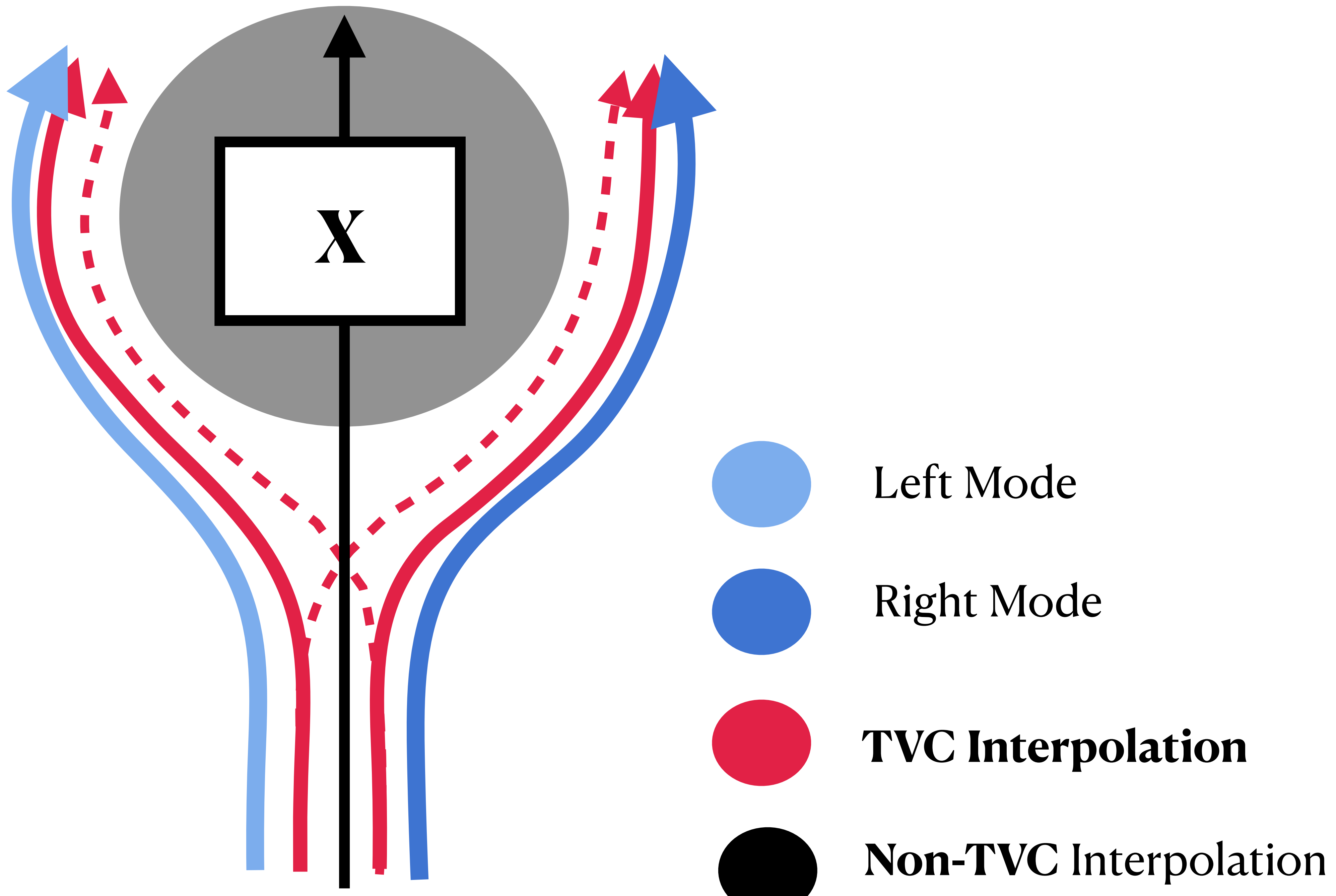
Clever **smoothing** with noise induces **TVC**

↓

TVC converts 'metric error' into '*discrete-token-error*'

↓

Imitation with '*discrete-token-error*' is easier

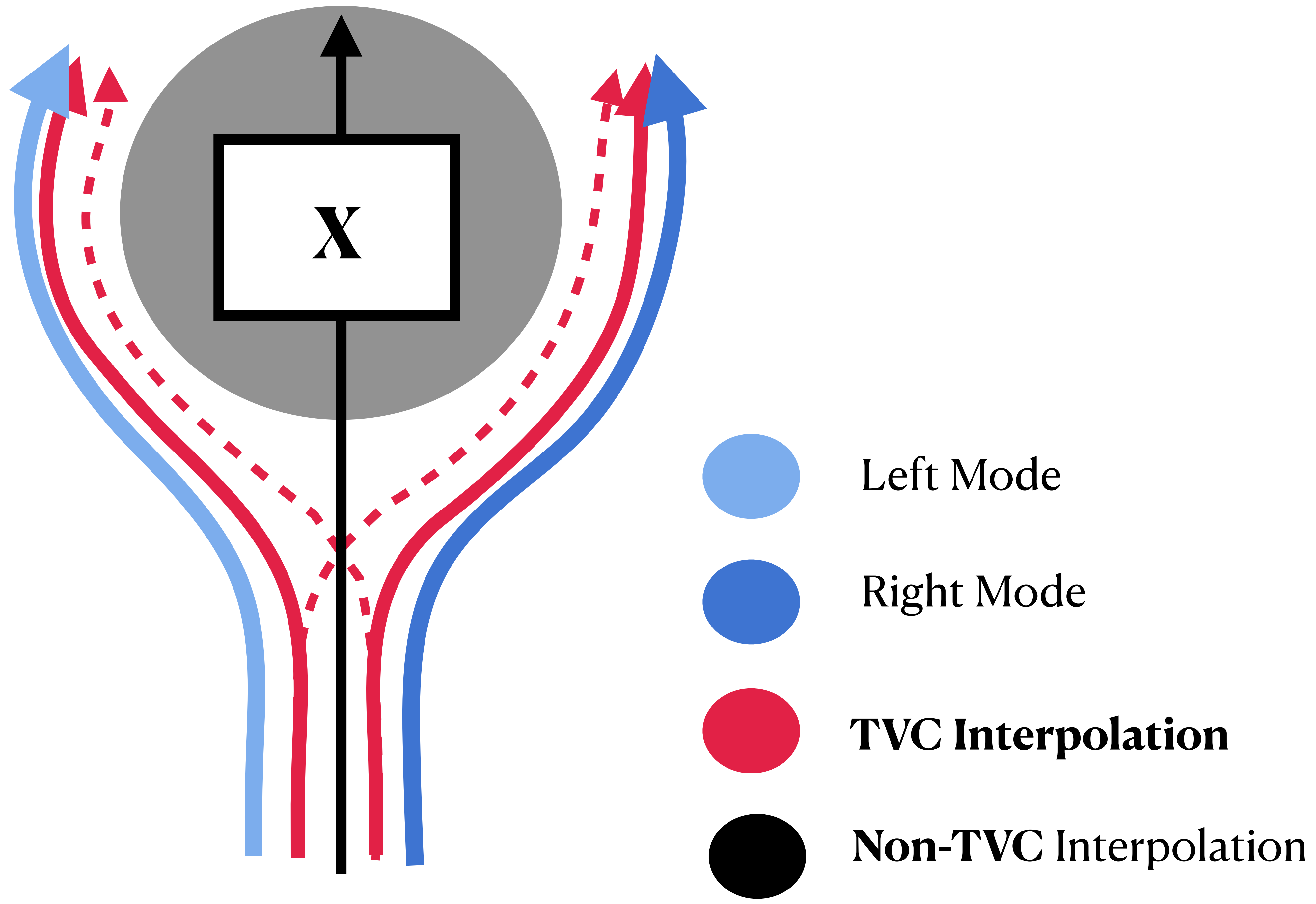


(+ use **control theory to induce**

What did we do?

Theorem 1 *(super informal)*: We can imitate **without exponentially compounding error** in **contractive systems.**

We algorithmically enforced the TVC inductive bias.



(+ use **control theory to induce contractivity)**

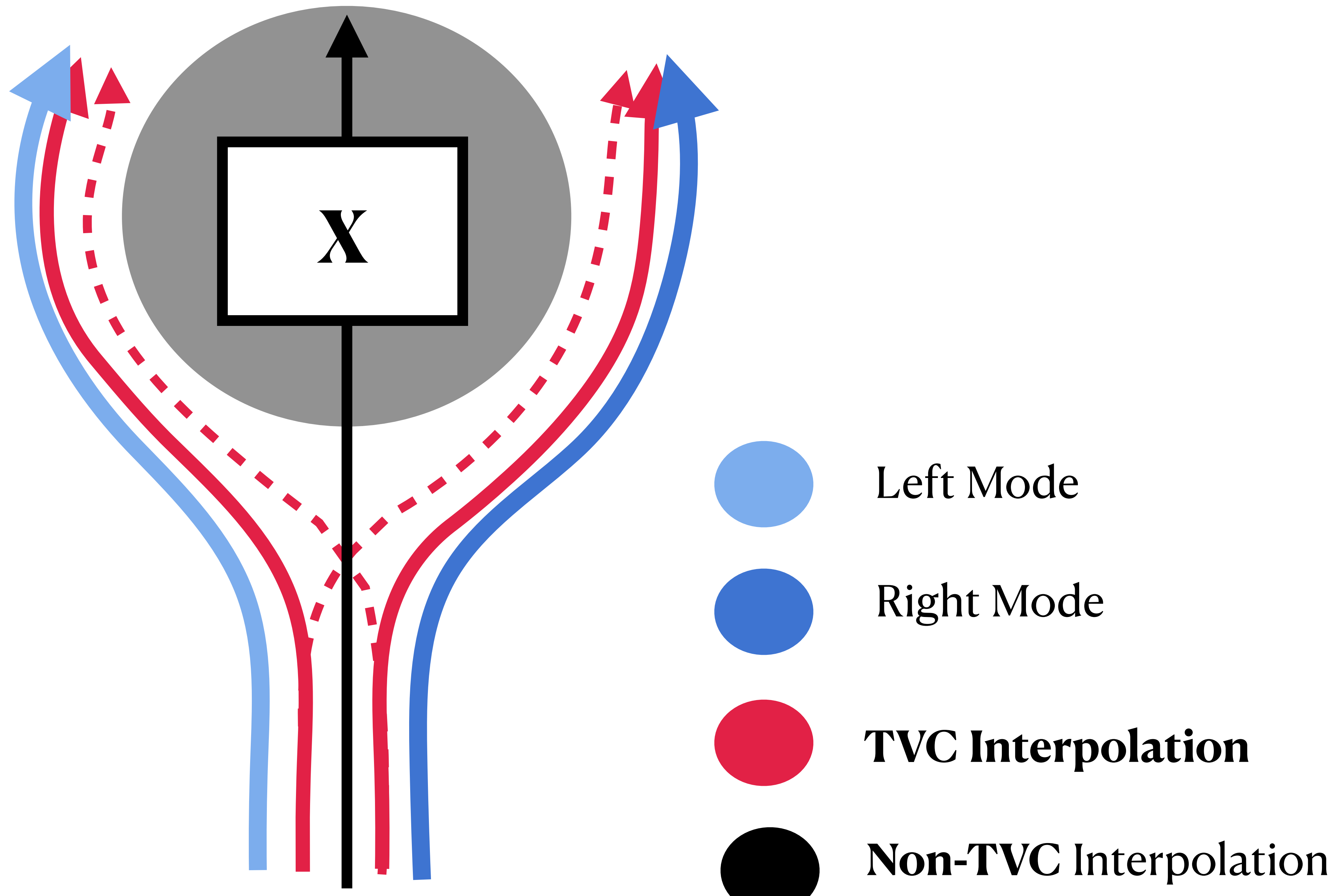
What did we do?

Theorem 1 *(super informal)*: We can imitate **without exponentially compounding error** in **contractive systems.**

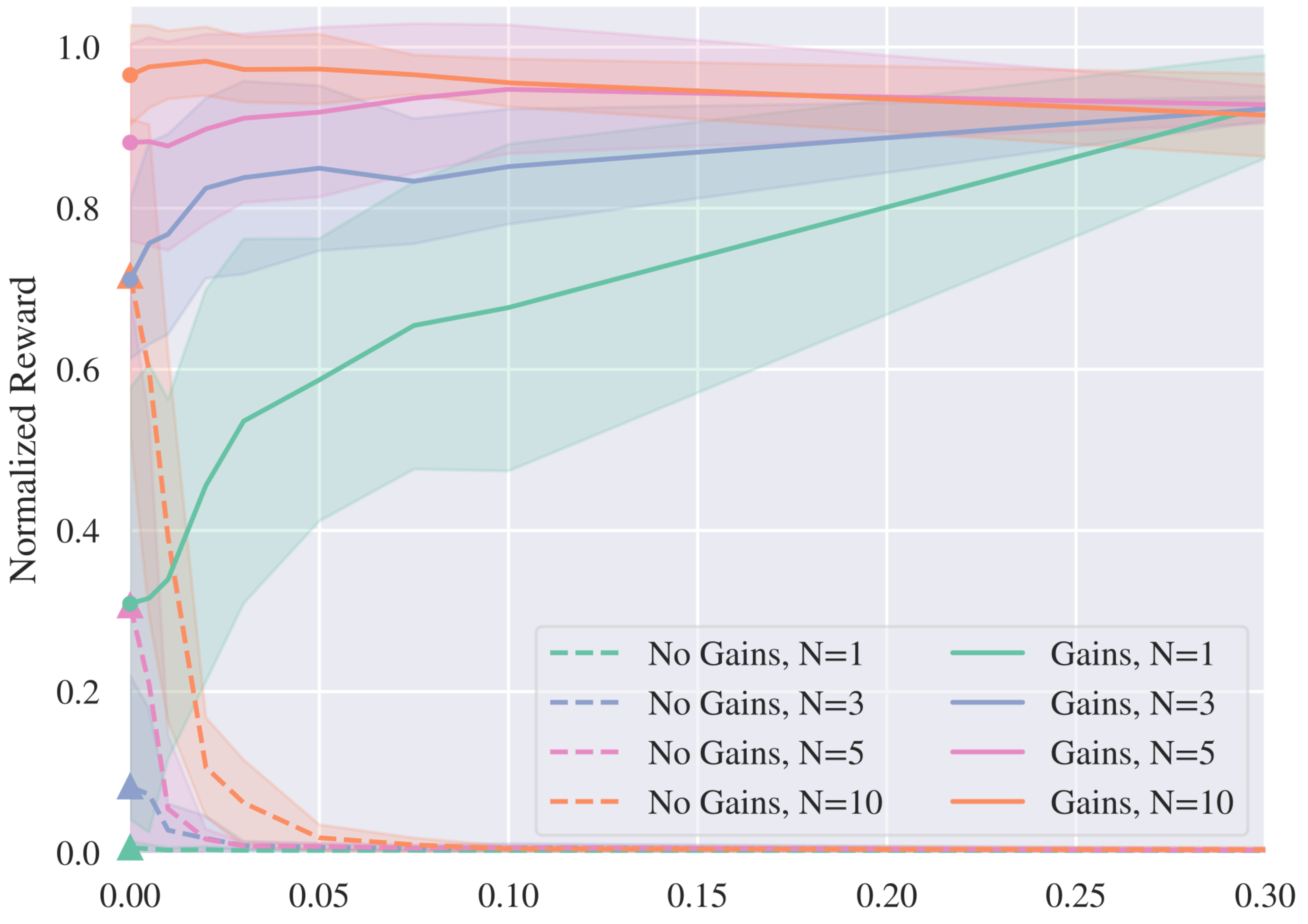
Open Question: What are the intrinsic inductive biases of diffusion models?

$$x \mapsto u \sim P(x)$$

Forthcoming work: Validates that diffusion models are not just 'more expressive', but have different inductive biases OOD.



Simulation Study.



added noise σ

2d Quadcopter

low-level control helps!

data noising helps!

data noising hurts without stabilization

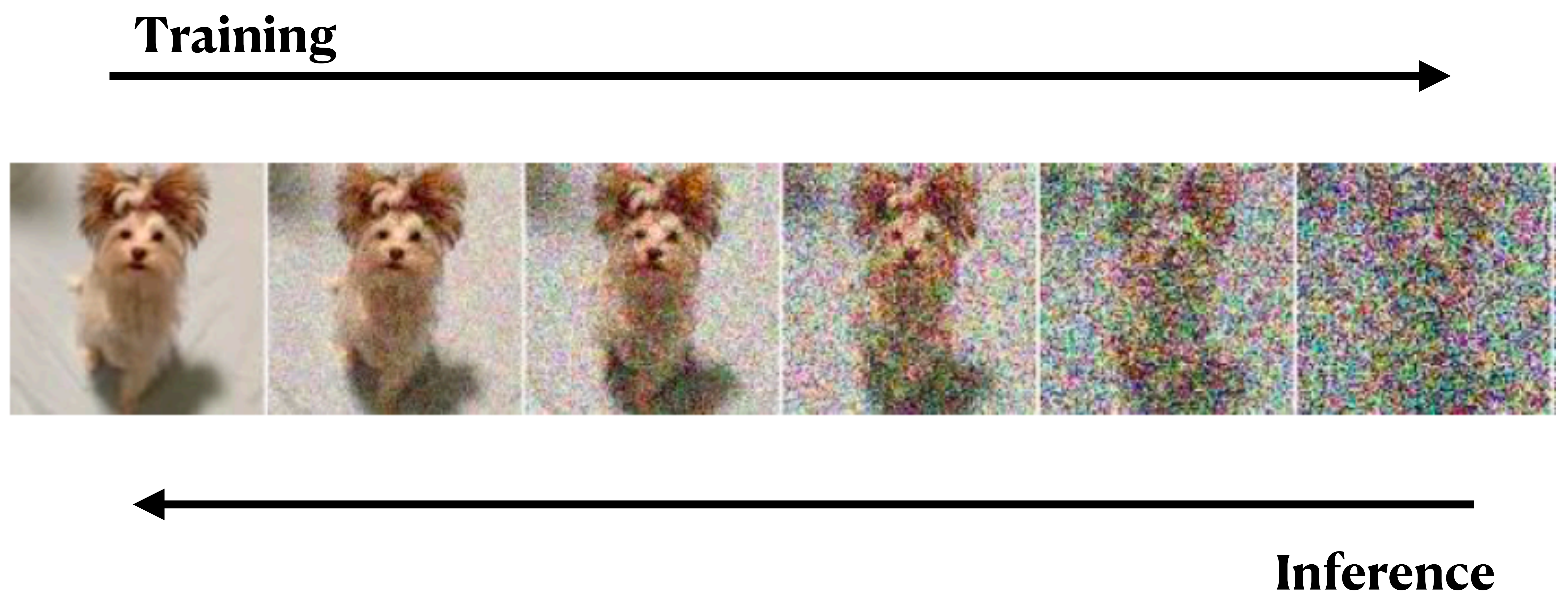
Applications?

ChatGPT

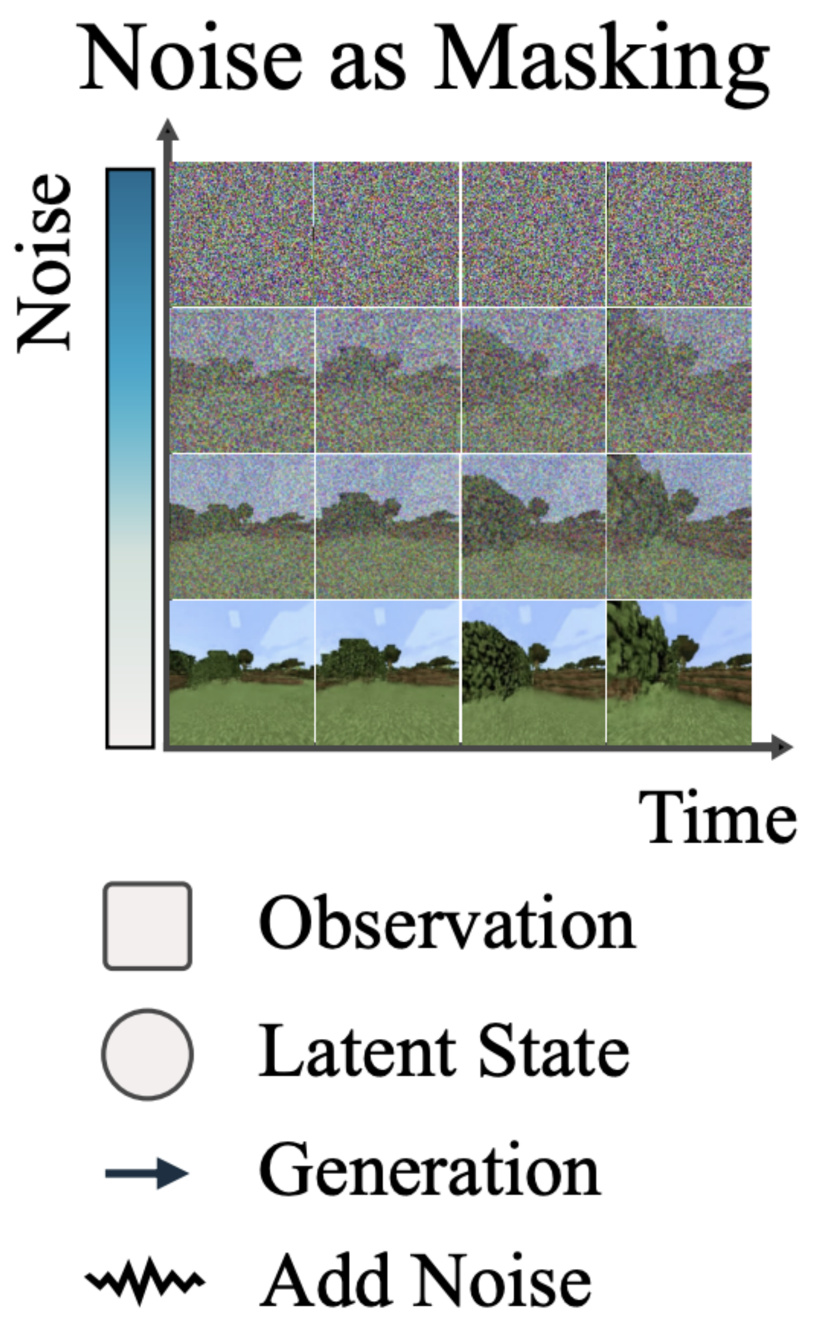
discrete-token
sequence model.

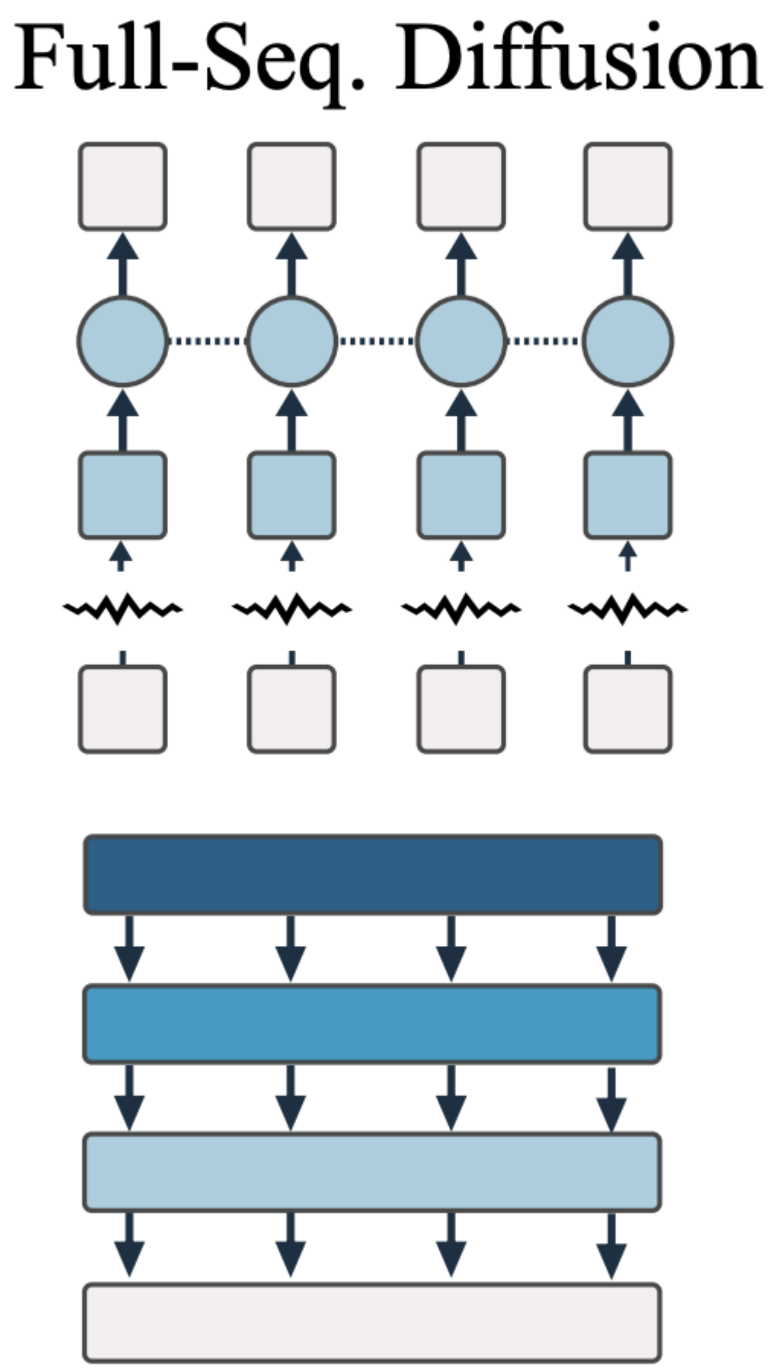
continuous-token
sequence model.

Recap: Diffusion



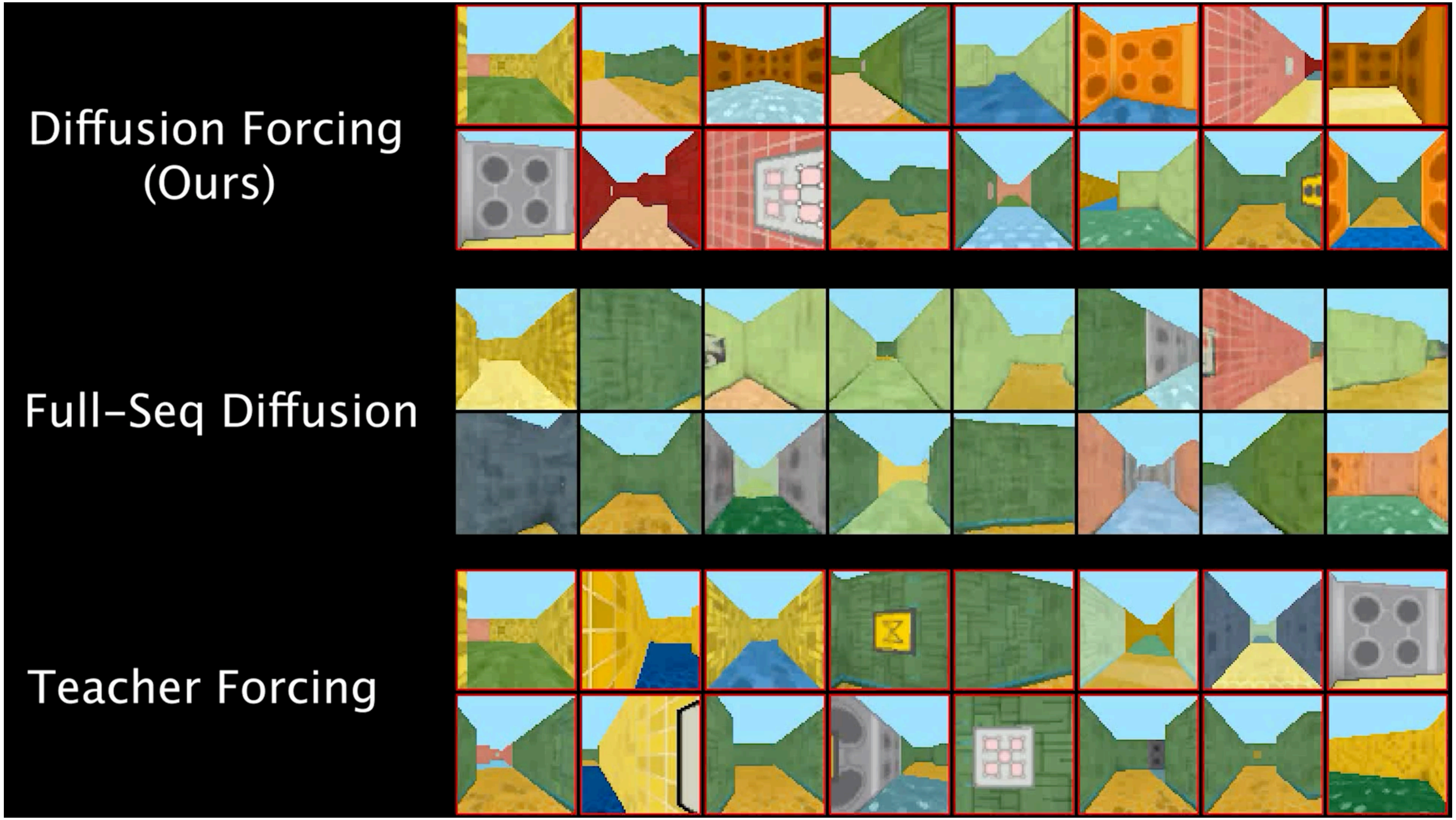
Diffusion for Sequences



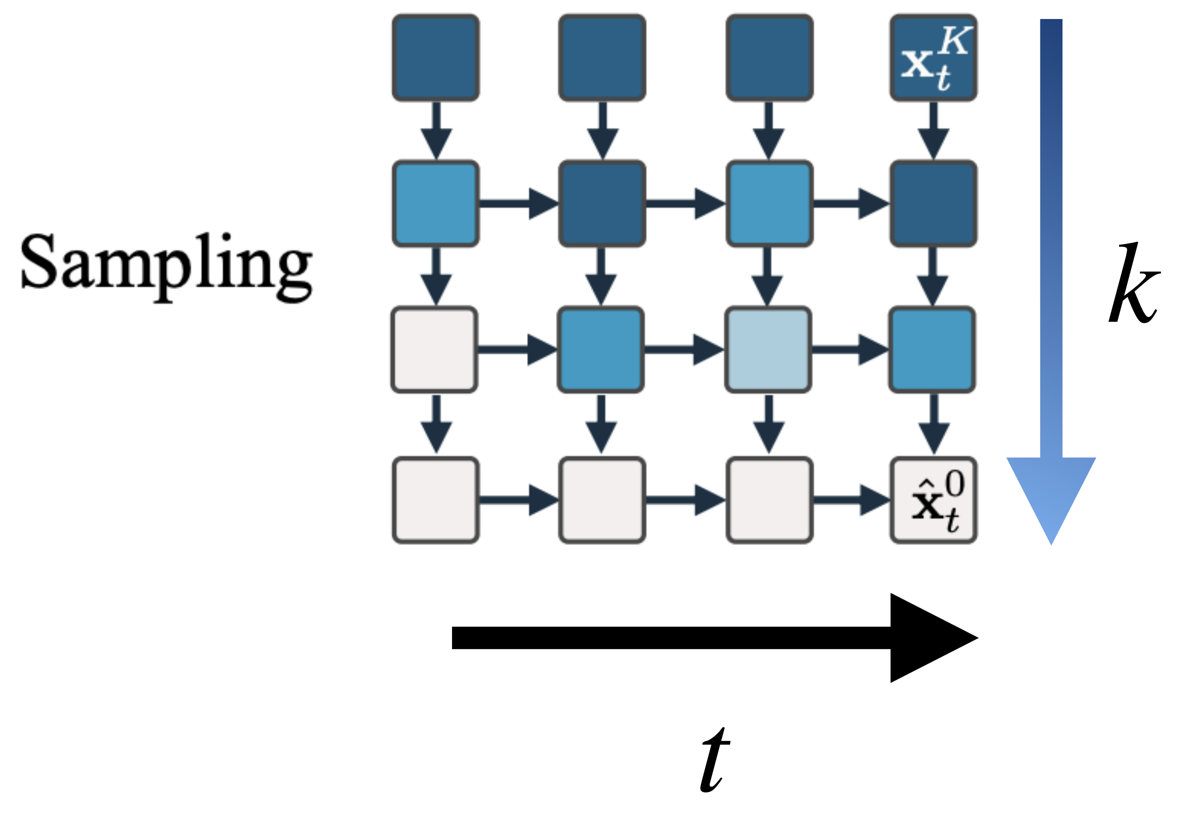


*(Boyuan Chen ... **S** ... et al. '24)*

*we **tell** the model the noise level


Diffusion Forcing (Ours)
Full-Seq Diffusion
Teacher Forcing

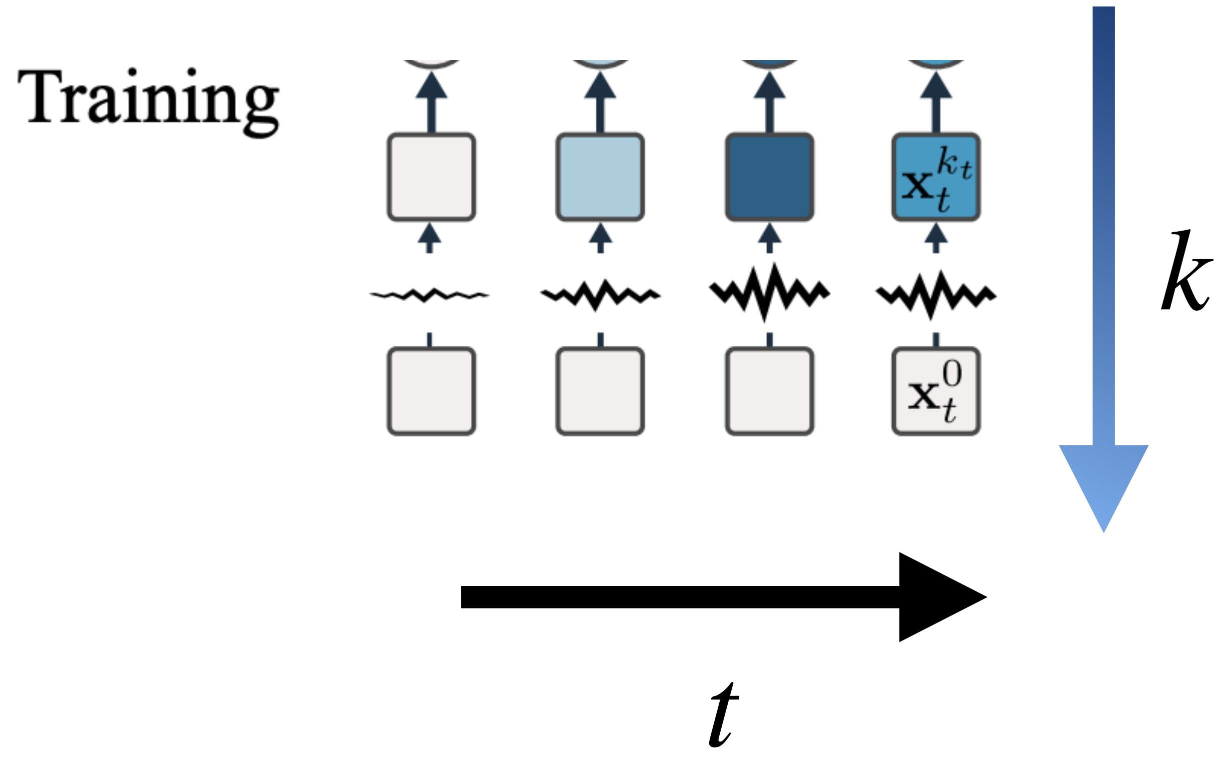
Example: Explicit Algorithmic Modification enabled by Generative Model



Inference

Predict token $x_t^0 \mid x_{t-1}^{k_0}, x_{t-2}^{k_0}, \ldots$

$$\approx x_t^0 \mid x_{t-1}^0 + \sigma w'_{t-1}, x_{t-2}^0 + \sigma w'_{t-2}, \ldots$$



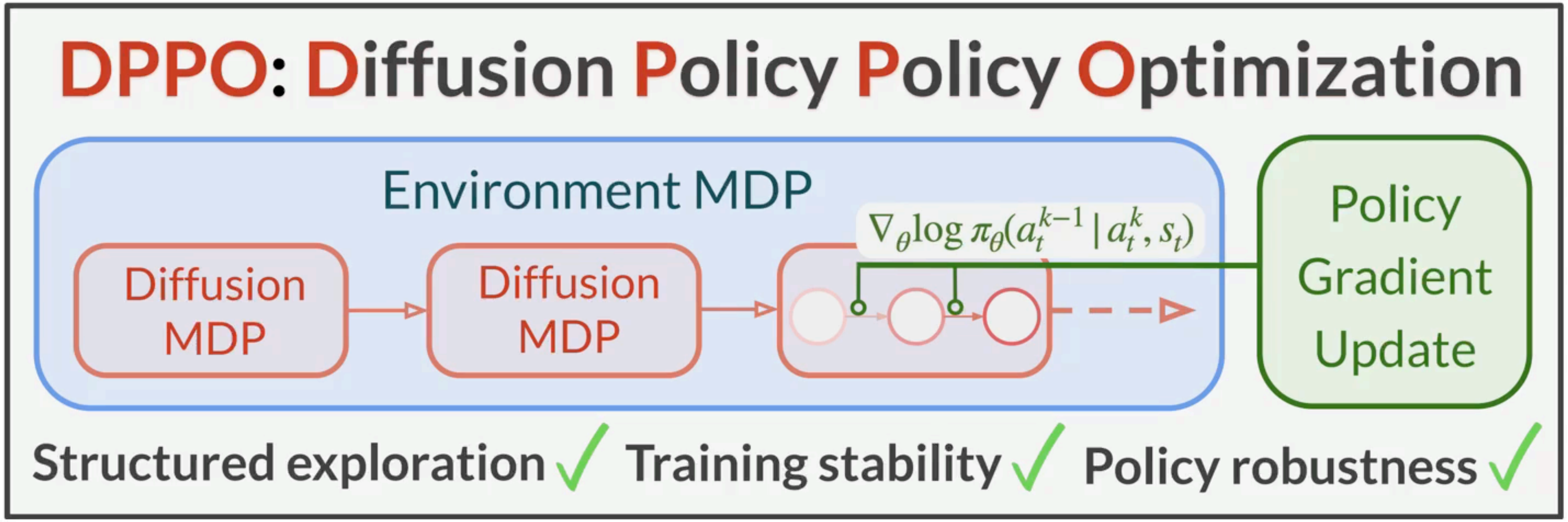
Training

Predict token $x_t^0 \mid x_{t-1}^{k_0}, x_{t-2}^{k_0}, \ldots$

$$= x_t^0 \mid x_{t-1}^0 + \sigma w_{t-1}, x_{t-2}^0 + \sigma w_{t-2}, \ldots$$

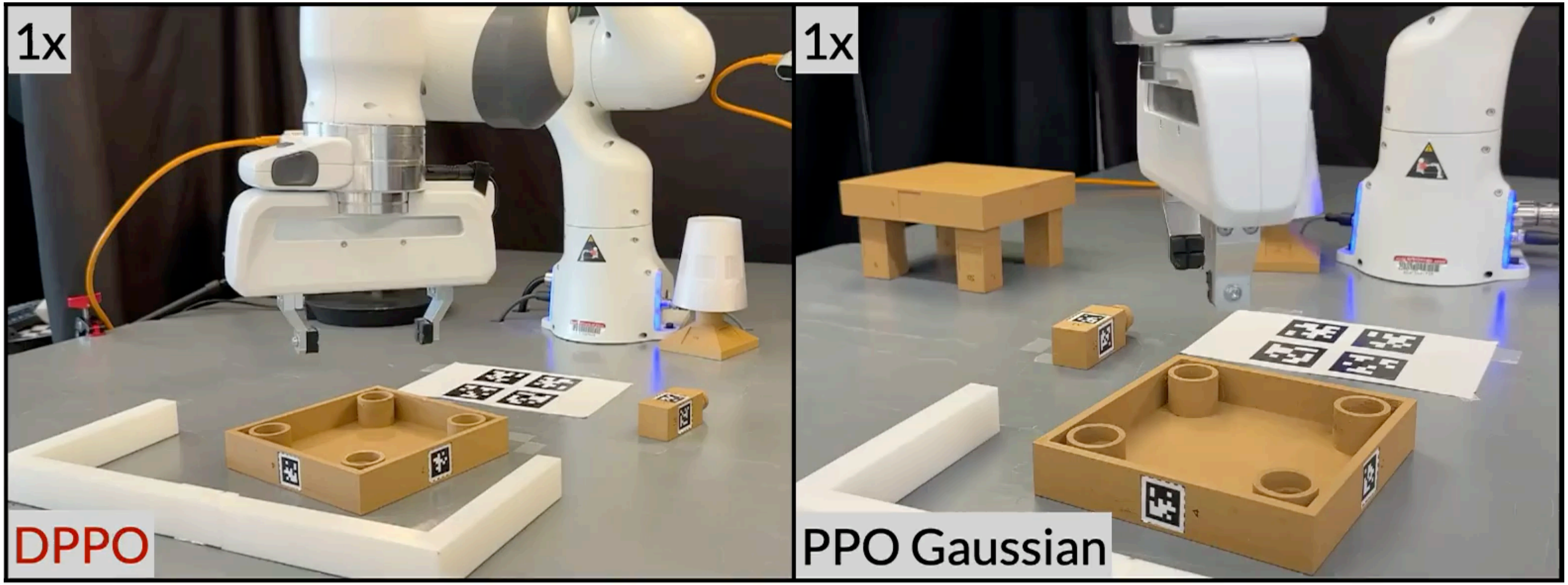
Replica Noising!

Replica Noising ‘Mathematical Foundation’ for why this works....



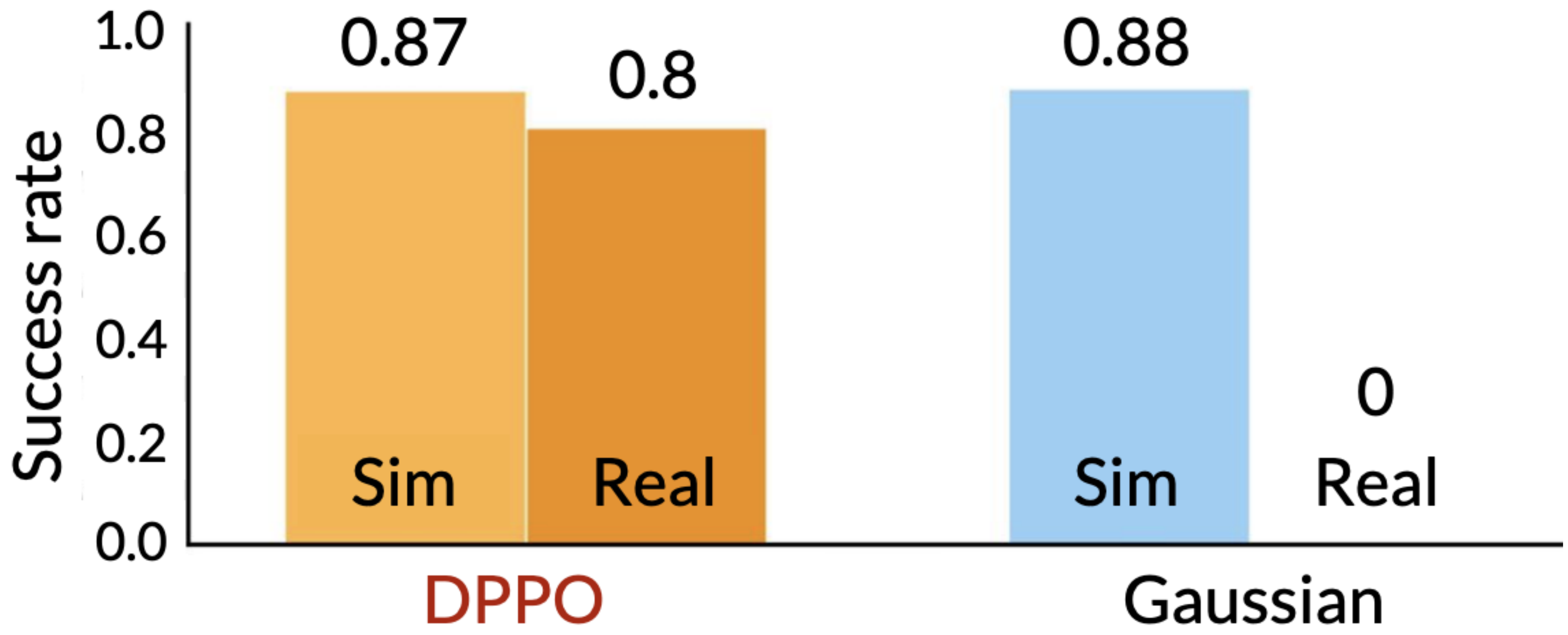
*(Allen Ren ... **S** et al. '24)*

conditional sampling $\pi : x \mapsto f(x) + \textbf{noise}$



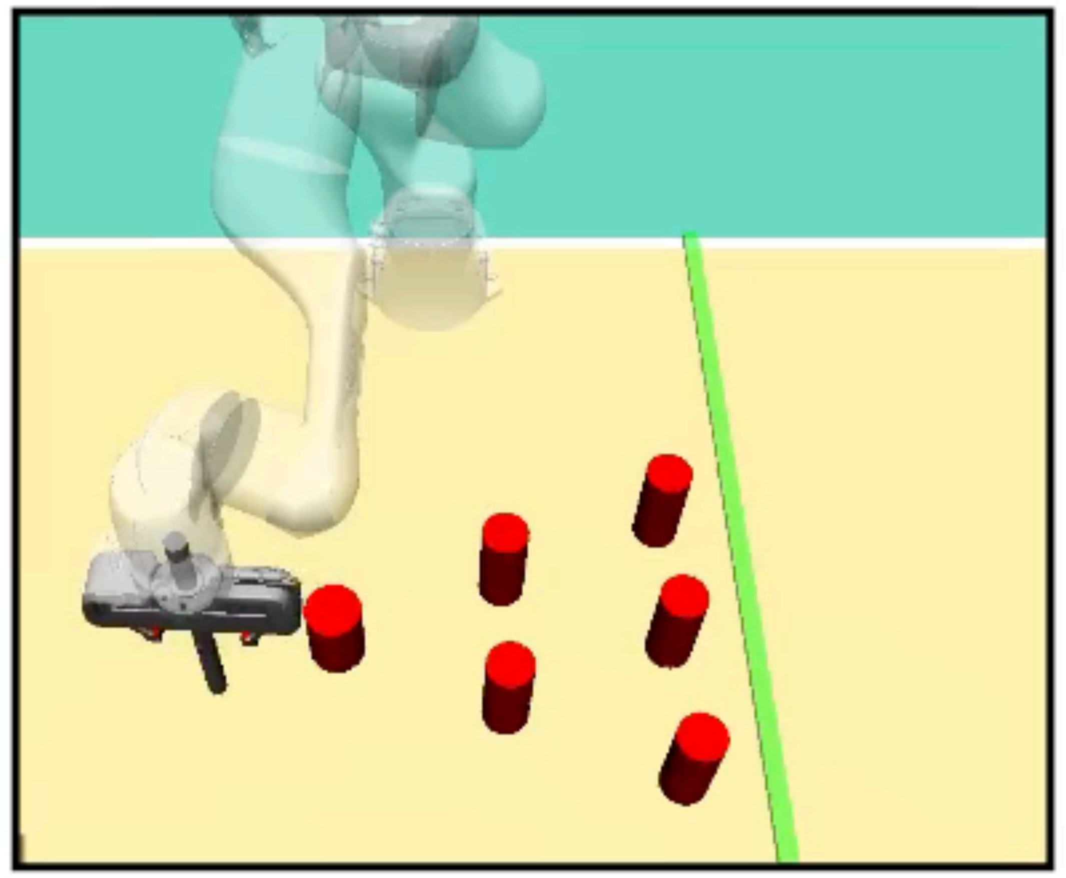
Real Hardware!

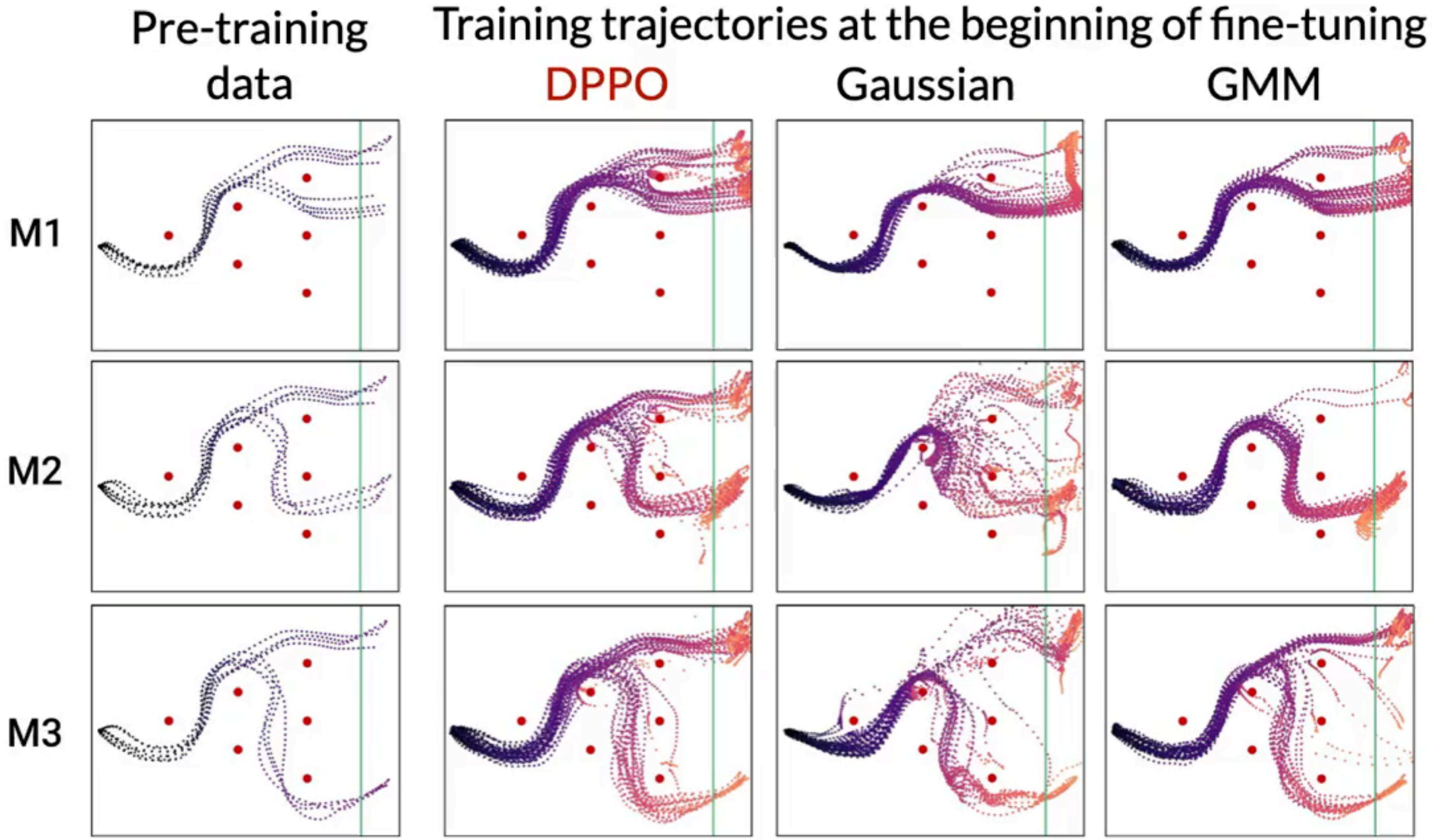
Simply that Diffusion Policies can 'represent' better performing policies?

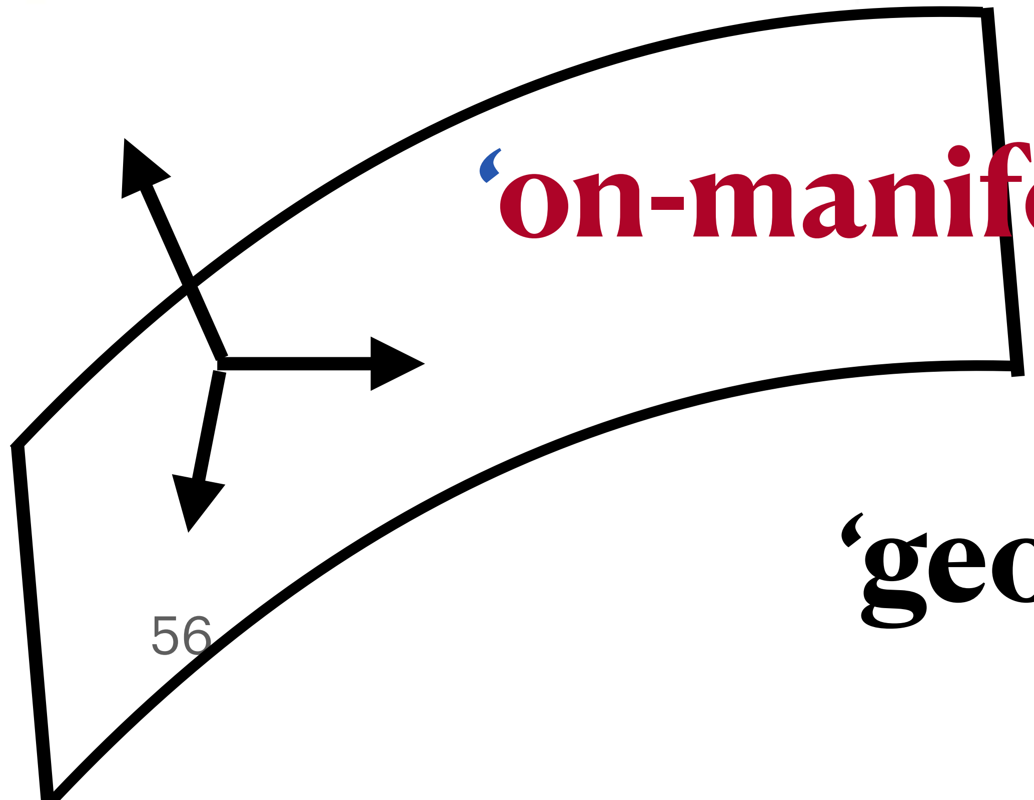


Example of richer models having better 'intrinsic' O.O.D. inductive bias

Avoid Environment from D3IL

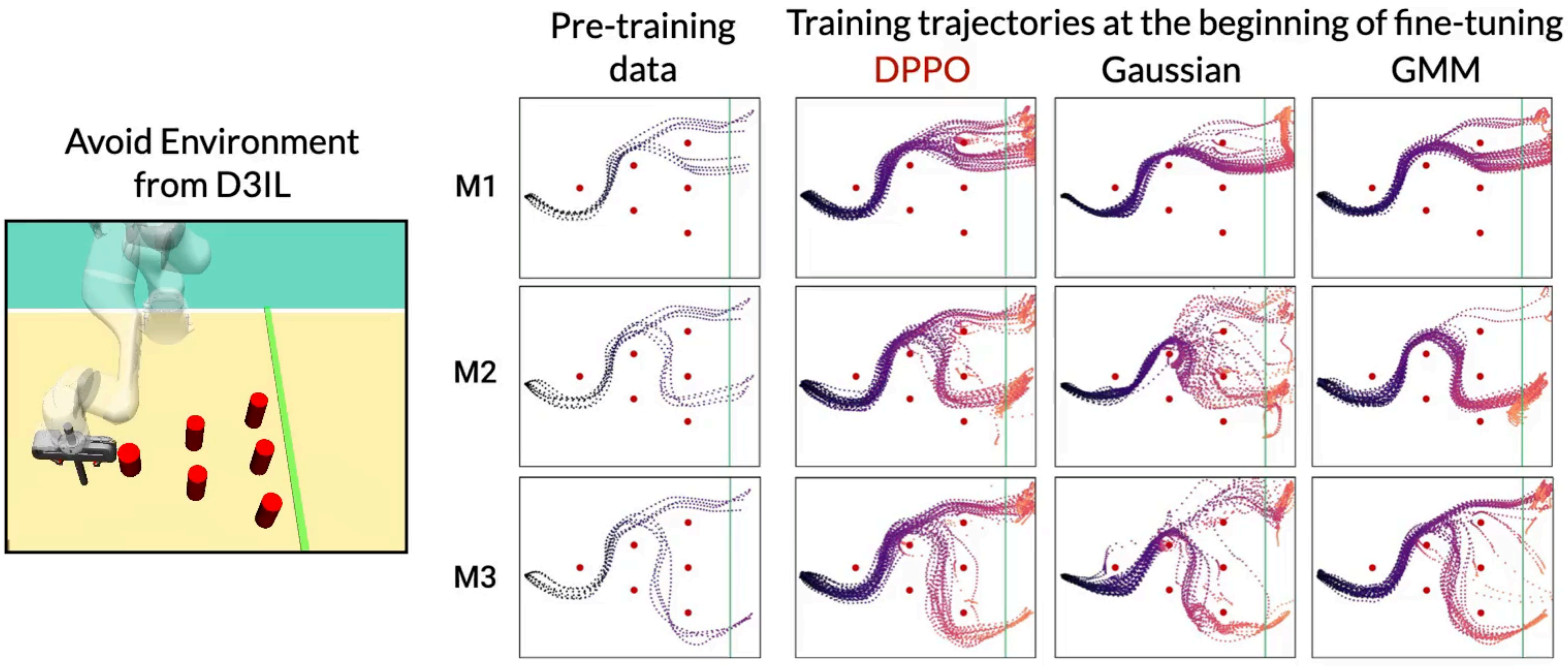






'on-manifold'

'geometry aware'



Open Question: Richer Models = More ‘Reasonable’ Exploration!

Pontification...

1. **Lot's of exciting questions in continuous-token prediction!**

   (robots, video, climate, AI4Science, conditional diffusion....)

2. **More expressive models** + **alg. choices** = **richer O.O.D. inductive biases!**

3. **How can we take full advantage of large/rich models for exploration?**

   (this should be true in LLMs!)

Provable Guarantees for Generative Behavior Cloning: Bridging Low-Level Stability and High-Level Behavior

Adam Block* Ali Jadbabaie Daniel Pfrommer† Max Simchowitz‡ Russ Tedrake

Massachusetts Institute of Technology

Diffusion Policy Policy Optimization

Allen Z. Ren[1], Justin Lidard[1], Lars L. Ankile[2], Anthony Simeonov[2],
Pulkit Agrawal[2], Anirudha Majumdar[1], Benjamin Burchfiel[3], Hongkai Dai[3], Max Simchowitz[2,4]

[1]Princeton University [2]Massachusetts Institute of Technology
[3]Toyota Research Institute [4]Carnegie Mellon University

Diffusion Forcing: Next-token Prediction Meets Full-Sequence Diffusion

Boyuan Chen
MIT CSAIL
boyuanc@mit.edu

Diego Martí Monsó*
Technical University of Munich
diego.marti@tum.de

Yilun Du
MIT CSAIL
yilundu@mit.edu

Max Simchowitz
MIT CSAIL
msimchow@mit.edu

Russ Tedrake
MIT CSAIL
russt@mit.edu

Vincent Sitzmann
MIT CSAIL
sitzmann@mit.edu

Butterfly Effects of SGD Noise: Error Amplification in Behavior Cloning and Autoregression

Adam Block[1*] Dylan J. Foster[2] Akshay Krishnamurthy[2] Max Simchowitz[1] Cyril Zhang[2]

[1]MIT [2]Microsoft Research NYC

Enjoy the weekend!