Generalization in AI Agents:
Lessons from Linear-Quadratic Control

Nadav Cohen
Tel Aviv University & Imubit

IMSI Workshop on New Directions in Reinforcement Learning and Control
12 May 2026

Talk Sources

Why Does Agentic Safety Fail to Generalize Across Tasks?

Slutzky + Alexander + Slor + Nagel + C
Preprint 2026

Implicit Bias of Policy Gradient in Linear Quadratic Control: Extrapolation to Unseen Initial States

Razin* + Alexander* + Cohen-Karlik + Giryes + Globerson + C
ICML 2024

Noam Razin

Yotam Alexander

Yonatan Slutzky

Tomer Slor

Yoav Nagel

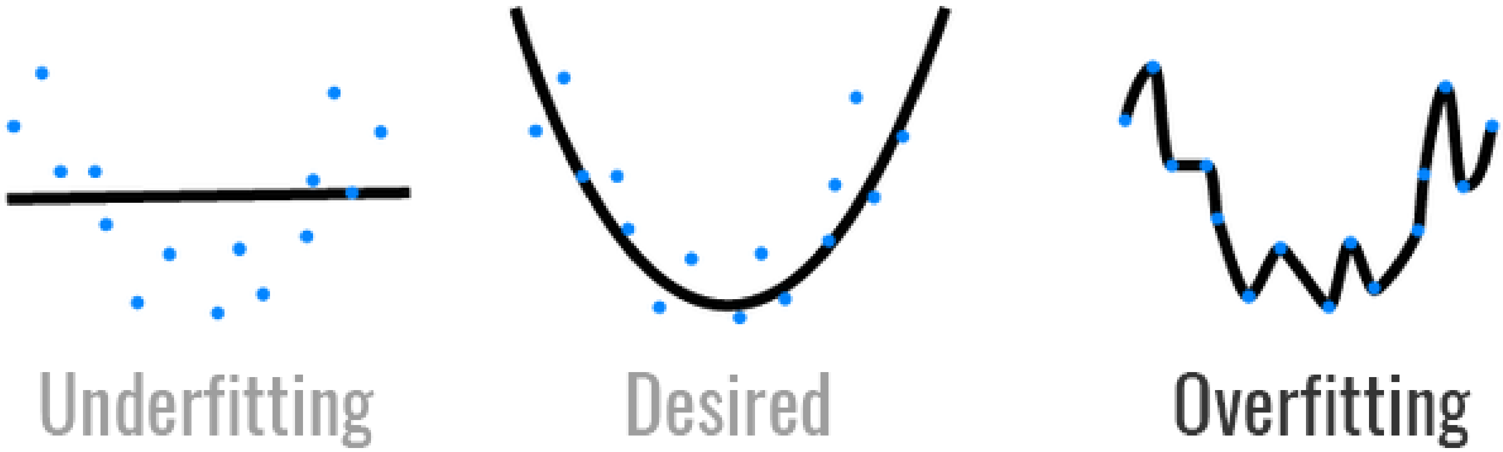
Edo Cohen-Karlik

Raja Giryes

Amir Globerson

Generalization in Classical Machine Learning

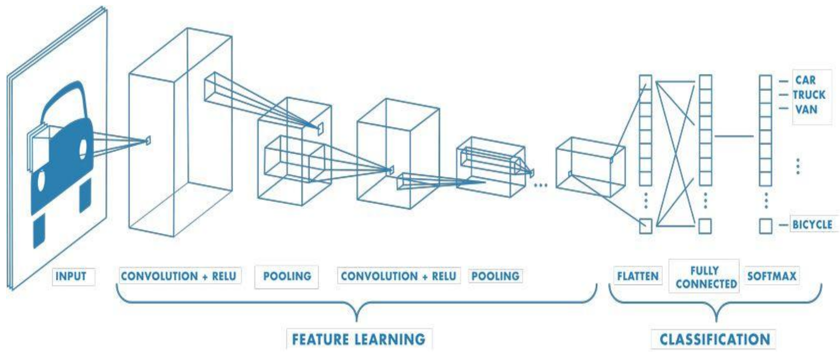
Classically, generalization is understood via the **bias-variance tradeoff**

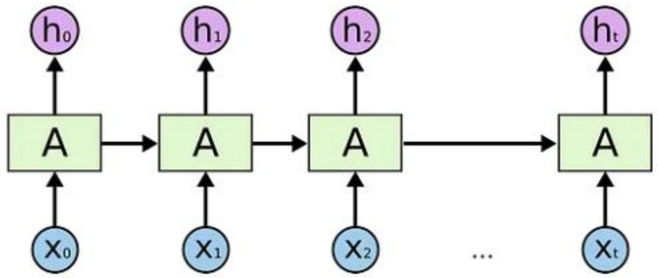


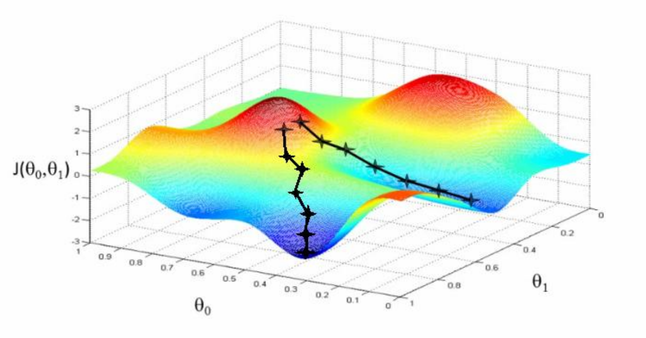
Tradeoff can be controlled through:

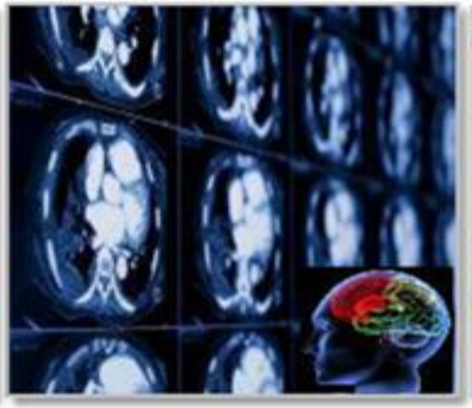
- Limiting model size (# of learned parameters)
- Optimizing with regularization (e.g., ℓ_2 penalty)

Supervised Deep Learning








- Image/Video classification
- Speech recognition
- Natural language processing

- Cancer cell detection
- Diabetic grading
- Drug discovery

- Video captioning
- Content based search
- Real time translation

- Face recognition
- Video surveillance
- Cyber security

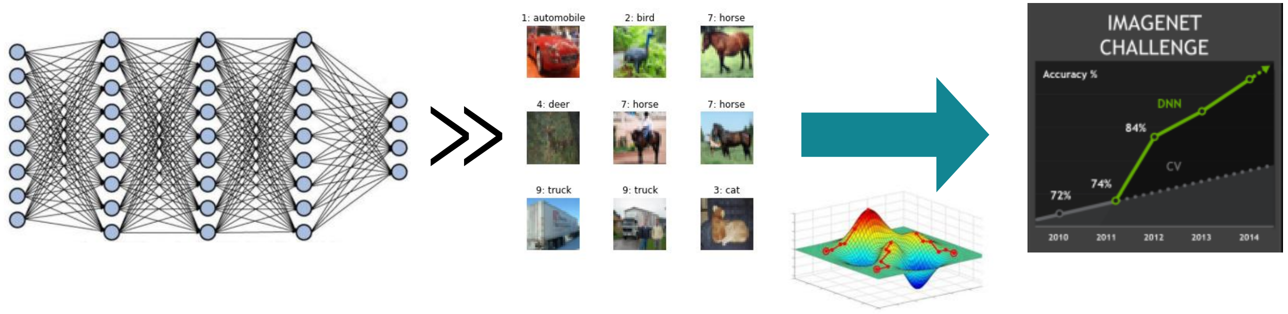
- Pedestrian detection
- Lane tracking
- Recognize traffic sign

Generalization in Supervised Deep Learning

Phenomenon

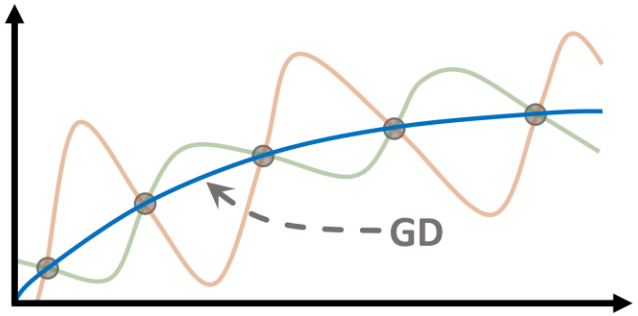
In supervised DL, optimizing via GD often leads to generalization, even when:

- Model size $\gg$ training set size
- There is no explicit regularization



Conventional Wisdom

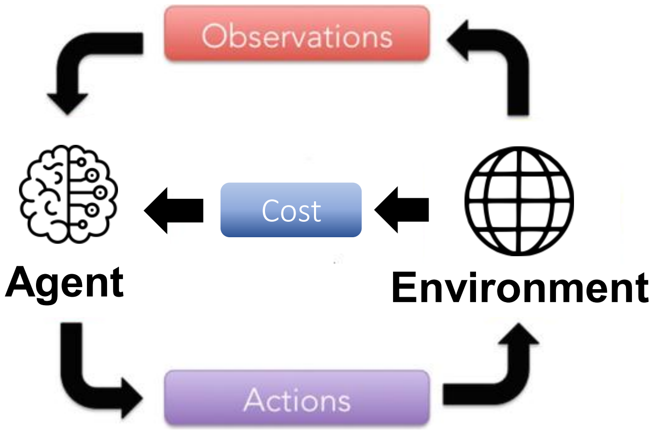
GD induces **implicit bias** towards generalizing mappings



Agents

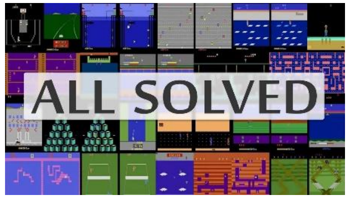
Goal

Based on **state observations** from dynamic **environment**, take **actions** that lead **task-dependent cost** to decrease



Applications

In **RL** / **control**:



Computer gaming

Playing Go

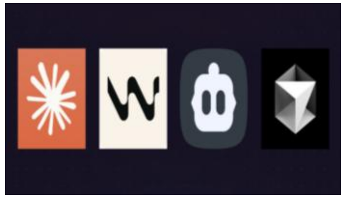

Autonomous driving


Medical treatment


Manufacturing optimization

IMUBIT

In **LLMs**:


Coding

Web browsing

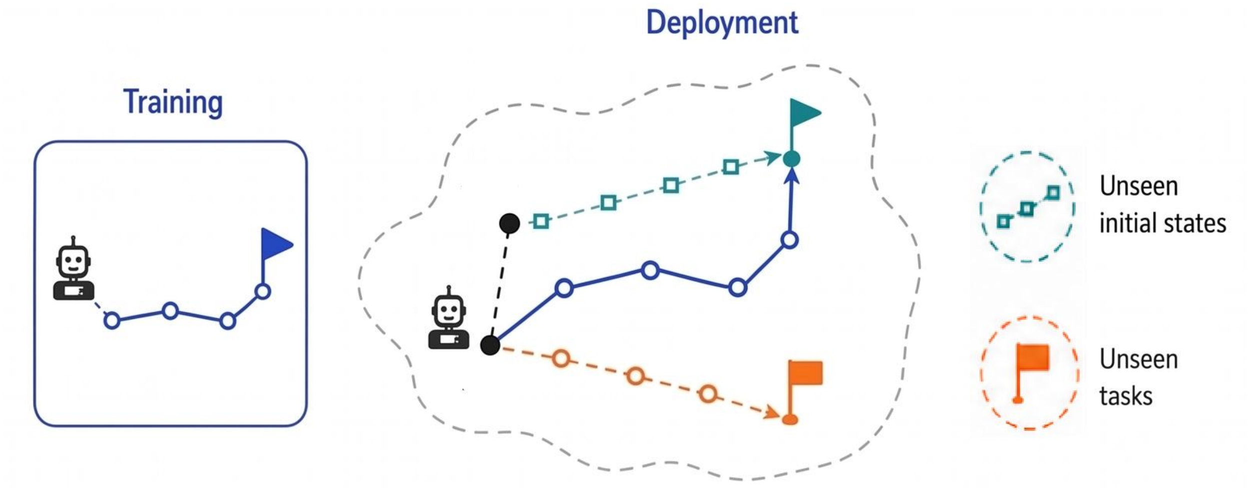
Scientific discovery


Personal assistance

Generalization in Agents

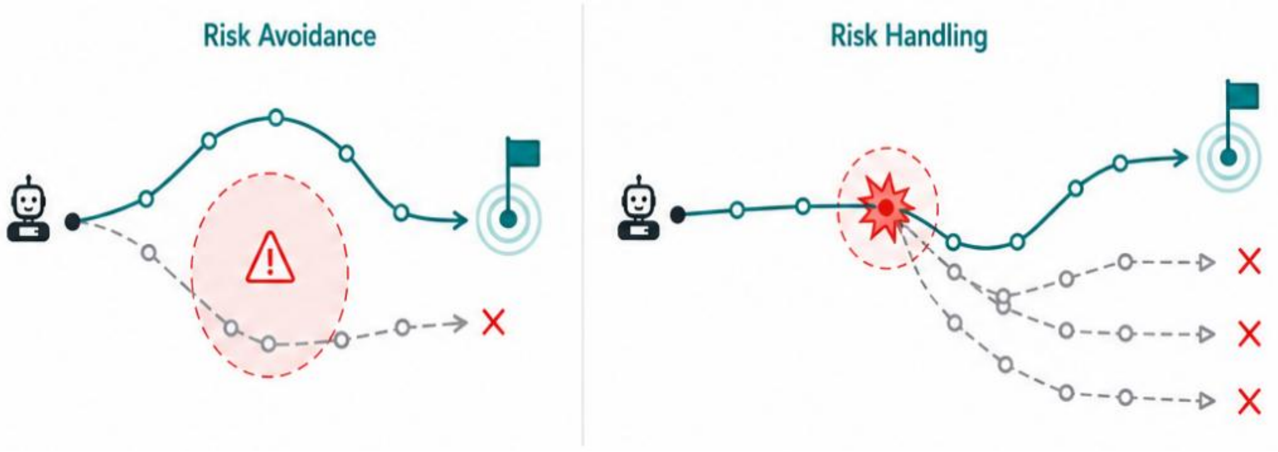
Agents entail **various types of generalization**:

- to unseen **initial states**
- to unseen **tasks**
- to unseen environments
- ...



Agents need to not only execute tasks, but do so **safely**:

- **Avoid** potential risks
- **Handle** materialized risks



Theoretical Testbed: Linear-Quadratic Control

A simple instance of AI agents is the **linear-quadratic control** problem:

State   Action   Disturbance

Linear Environment

$$\mathbf{x}_{h+1} = \mathbf{A}\mathbf{x}_h + \mathbf{B}\mathbf{u}_h + \mathbf{D}\mathbf{w}_h$$

Transition Matrix (parameter)   Action Matrix (parameter)   Disturbance Matrix (parameter)

Quadratic Cost

$$\mathbb{E}\left[\sum_{h=0}^{\infty} \mathbf{x}_h^\top \mathbf{Q}\mathbf{x}_h + \mathbf{u}_h^\top \mathbf{R}\mathbf{u}_h\right]$$

State cost matrix (parameter)   Action cost matrix (parameter)

Known Results

In various settings optimal agent is **linear**

Optimal $\mathbf{K}$ is generally non-linear in $\mathbf{A}, \mathbf{B}, \mathbf{D}, \mathbf{Q}, \mathbf{R}$

Linear Agent

$$\mathbf{u}_h = \mathbf{K}\mathbf{x}_h$$

Agent matrix (learned)

Generalization in Linear-Quadratic Control

Linear Environment

$$\mathbf{x}_{h+1} = \mathbf{A}\mathbf{x}_h + \mathbf{B}\mathbf{u}_h + \mathbf{D}\mathbf{w}_h$$

Quadratic Cost

$$\mathbb{E}\left[\sum_{h=0}^{\infty} \mathbf{x}_h^\top \mathbf{Q}\mathbf{x}_h + \mathbf{u}_h^\top \mathbf{R}\mathbf{u}_h\right]$$

Linear Agent

$$\mathbf{u}_h = \mathbf{K}\mathbf{x}_h$$

Agents entail **various types of generalization**:

- to unseen **initial states** ← Train on set of initial states $\mathcal{S}$, perform well on $\mathcal{S}^\perp$
- to unseen **tasks** ← Train on certain values for $\mathbf{Q}$, $\mathbf{R}$, perform well on others
- ...

Agents need to not only execute tasks,
but do so **safely**:

- **Avoid** potential risks ← Avoid $\mathbf{x}_h \in \mathcal{X}_{\text{ban}}$ for a predetermined set $\mathcal{X}_{\text{ban}}$
- **Handle** materialized risks ← Handle worst case disturbances (H_∞ -robustness):

$$\inf_{\mathbf{K}} \sup_{\mathbf{w}_h} \frac{\sum_{h=0}^{\infty} \mathbf{x}_h^\top Q \mathbf{x}_h + \mathbf{u}_h^\top R \mathbf{u}_h}{\sum_{h=0}^{\infty} \|\mathbf{w}_h\|_2^2}$$

Outline

Generalization to Unseen Initial States

Linear Environment	Quadratic Cost	Linear Agent
$\mathbf{x}_{h+1} = \mathbf{A}\mathbf{x}_h + \mathbf{B}\mathbf{u}_h + \mathbf{D}\mathbf{w}_h$	$\mathbb{E}\left[\sum_{h=0}^{\infty} \mathbf{x}_h^\top \mathbf{Q}\mathbf{x}_h + \mathbf{u}_h^\top \mathbf{R}\mathbf{u}_h\right]$	$\mathbf{u}_h = \mathbf{K}\mathbf{x}_h$

Set of **seen initial states** $\mathcal{S}$ induces **training cost**:

$$\mathcal{C}_{\mathcal{S}}(\mathbf{K}) = \frac{1}{|\mathcal{S}|} \sum_{\mathbf{x}_0 \in \mathcal{S}} \mathbb{E}\left[\sum_{h=0}^{H} \mathbf{x}_h^\top (\mathbf{Q} + \mathbf{K}^\top \mathbf{R}\mathbf{K})\mathbf{x}_h\right]$$

Learning agent via GD over training cost (**policy gradient**):

$\mathbf{K}_{\mathrm{GD}} = \mathbf{0}$

For $t = 1, 2, \ldots$: $\mathbf{K}_{\mathrm{GD}} \leftarrow \mathbf{K}_{\mathrm{GD}} - \underset{\substack{\uparrow \\ \text{Learning rate}}}{\eta} \cdot \nabla \mathcal{C}_{\mathcal{S}}(\mathbf{K}_{\mathrm{GD}})$

How does the **implicit bias of GD** affect generalization to unseen initial states $\mathcal{S}^\perp$?

Overparameterized Linear Quadratic Control

Linear Environment

$$\mathbf{x}_{h+1} = \mathbf{A}\mathbf{x}_h + \mathbf{B}\mathbf{u}_h + \mathbf{D}\mathbf{w}_h$$

Linear Agent

$$\mathbf{u}_h = \mathbf{K}\mathbf{x}_h$$

GD:

$$\mathbf{K}_{\mathrm{GD}} \leftarrow \mathbf{K}_{\mathrm{GD}} - \eta \cdot \nabla \mathcal{C}_{\mathcal{S}}(\mathbf{K}_{\mathrm{GD}})$$

Training cost: $\mathcal{C}_{\mathcal{S}}(\mathbf{K}) = \frac{1}{|\mathcal{S}|} \sum_{\mathbf{x}_0 \in \mathcal{S}} \sum_{h=0}^{H} \mathbf{x}_h^\top (\mathbf{Q} + \mathbf{K}^\top \mathbf{R} \mathbf{K}) \mathbf{x}_h$

Considered Setting

- $\mathbf{R} = \mathbf{0}$ (actions are not penalized)
- $\mathbf{Q}$ has full rank (all states are penalized)
- $\mathbf{B}$ has full rank (environment is always controllable)
- $\mathbf{D} = \mathbf{0}$ (no disturbances)
- $\mathcal{S}$ does not span state space

Characteristic in:
deadbeat control, singular LQR, model predictive control

Setting is overparameterized: multiple agents $\mathbf{K}$ minimize cost $\mathcal{C}_{\mathcal{S}}(\cdot)$

Q: Does $\mathbf{K}_{\mathrm{GD}}$ generalize to unseen initial states?

Quantifying Generalization

Optimality Condition

An agent $\mathbf{K}$ minimizes cost $\mathcal{C}_{\mathcal{S}}(\cdot)$ **if and only if** $\underbrace{\|(\mathbf{A}+\mathbf{B}\mathbf{K})\mathbf{x}_0\|^2 = 0}_{\mathbf{K} \text{ sends } \mathbf{x}_0 \text{ to zero}}$ for all $\mathbf{x}_0 \in \mathcal{S}$

Definition: *Error in Generalization to Unseen Initial States*

$\mathcal{E}(\mathbf{K}) := \frac{1}{|\mathcal{U}|}\sum_{\mathbf{x}_0 \in \mathcal{U}} \|(\mathbf{A}+\mathbf{B}\mathbf{K})\mathbf{x}_0\|^2$, where $\mathcal{U}$ is a basis of $\mathcal{S}^{\perp}$ (unseen subspace)

Baseline Agents

Perfectly Generalizing $\mathbf{K}_{\text{gen}}$

Satisfies $(\mathbf{A}+\mathbf{B}\mathbf{K}_{\text{gen}})\mathbf{x}_0 = \mathbf{0}$ for all $\mathbf{x}_0$

Minimizes cost $\mathcal{C}_{\mathcal{S}}(\cdot)$ and $\mathcal{E}(\mathbf{K}_{\text{gen}}) = 0$

Non-Generalizing $\mathbf{K}_{\text{no-gen}}$

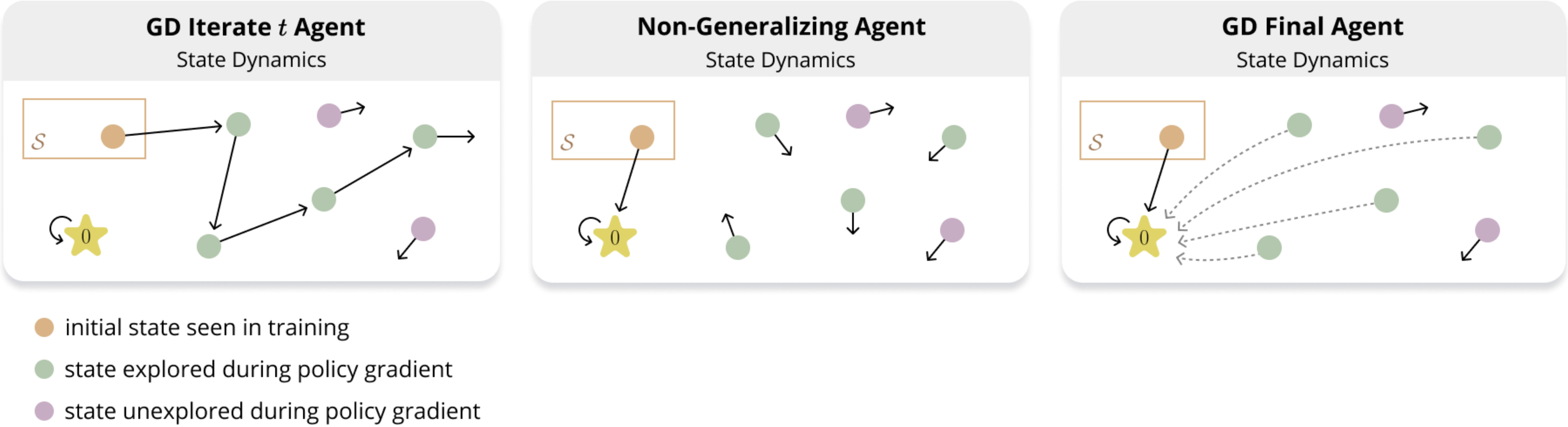
Satisfies $(\mathbf{A}+\mathbf{B}\mathbf{K}_{\text{no-gen}})\mathbf{x}_0 = \begin{cases} \mathbf{0} & , \mathbf{x}_0 \in \mathcal{S} \\ \mathbf{A}\mathbf{x}_0 & , \mathbf{x}_0 \in \mathcal{U} \end{cases}$

Minimizes cost $\mathcal{C}_{\mathcal{S}}(\cdot)$ but $\mathcal{E}(\mathbf{K}_{\text{no-gen}})$ is **typically high**

Intuition: Generalization is Determined by Exploration

Intuition Behind Our Results

Generalization of K_{GD} is determined by exploration induced by environment from seen initial states



Generalization Requires Exploration

Proposition

(i) For states orthogonal to those reached during GD, $\mathbf{K}_{\text{GD}}$ and $\mathbf{K}_{\text{no-gen}}$ produce identical actions

(ii) There exist non-exploratory environments with which: $\mathcal{E}(\mathbf{K}_{\text{GD}}) = \mathcal{E}(\mathbf{K}_{\text{no-gen}})$

Proof Sketch

(i) Denote by $\mathcal{X}_{\text{GD}}$ the states reached during GD

$$\mathcal{S} \subseteq \mathcal{X}_{\text{GD}} \Rightarrow \mathcal{X}_{\text{GD}}^{\perp} \subseteq \mathcal{S}^{\perp} \Rightarrow \forall \mathbf{x} \in \mathcal{X}_{\text{GD}}^{\perp} : \mathbf{K}_{\text{no-gen}}\mathbf{x} = \mathbf{0}$$

During GD the rows of $\nabla \mathcal{C}_{\mathcal{S}}(\cdot)$ are spanned by $\mathcal{X}_{\text{GD}}$

$\Rightarrow$ the rows of $\mathbf{K}_{\text{GD}}$ are spanned by $\mathcal{X}_{\text{GD}}$ $\Rightarrow$ $\forall \mathbf{x} \in \mathcal{X}_{\text{GD}}^{\perp} : \mathbf{K}_{\text{GD}}\mathbf{x} = \mathbf{0} = \mathbf{K}_{\text{no-gen}}\mathbf{x}$

(ii) Take $\mathbf{A} = \mathbf{B} = \mathbf{I}$. Then:

$$\mathcal{S} \subseteq \mathcal{X}_{\text{GD}} \subset \text{span}(\mathcal{S}) \Rightarrow \mathcal{X}_{\text{GD}}^{\perp} = \mathcal{S}^{\perp} \Rightarrow \mathcal{U} \subseteq \mathcal{X}_{\text{GD}}^{\perp}$$

$$\Rightarrow \forall \mathbf{x} \in \mathcal{U} : \mathbf{K}_{\text{GD}}\mathbf{x} = \mathbf{K}_{\text{no-gen}}\mathbf{x} \Rightarrow \mathcal{E}(\mathbf{K}_{\text{GD}}) = \mathcal{E}(\mathbf{K}_{\text{no-gen}})$$

Generalization in Exploration-Inducing Setting

Q: Exploration is necessary for generalization, but can it be sufficient?

Proposition

There exist exploration-inducing settings in which: $\mathcal{E}(\mathbf{K}_{\text{GD}}) << \mathcal{E}(\mathbf{K}_{\text{no-gen}})$

Proof Sketch

Can be shown directly in the setting:

$$\mathcal{S} = \{\mathbf{e}_1 := (1, 0, 0, \ldots, 0)^\top\} \qquad \mathbf{B} = \mathbf{I} \qquad \mathbf{A} = \begin{array}{c} \text{“Shift” matrix} \\ \begin{pmatrix} 0 & 0 & 0 & \cdots & 0 & 1 \\ 1 & 0 & 0 & \cdots & 0 & 0 \\ 0 & 1 & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & 1 & 0 \end{pmatrix} \end{array}$$

Only one initial state seen in training!

“Maximal” exploration

Trajectory steered by environment when GD commences: $\mathbf{e}_1 \mapsto \mathbf{e}_2 \mapsto \mathbf{e}_3 \mapsto \cdots$

Generalization in Typical Setting

Q: We saw two ends of a spectrum, but what about the typical setting?

Theorem

When $\mathbf{A}$ (transition matrix of environment) is random Gaussian:

$$\mathcal{E}(\mathbf{K}_{\text{GD}}) \leq \mathcal{E}(\mathbf{K}_{\text{no-gen}}) - \Omega\left(\eta \cdot \frac{H^2}{d}\right)$$

Horizon
State dim
Learning rate

in expectation, and with high probability if d is large

Proof Sketch

Intuition: random environment induces exploration with high probability

➡ generalization takes place with high probability

Trajectory steered by random environment from arbitrary state $\mathbf{x}$ when GD commences (namely $\mathbf{x} \mapsto \mathbf{A}\mathbf{x} \mapsto \mathbf{A}^2\mathbf{x} \mapsto \cdots$) spans state space with probability one

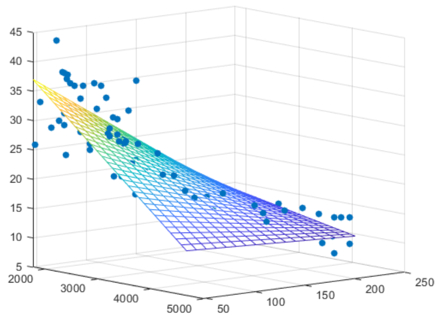
Intuition can be formalized via advanced tools from random matrix theory + topology

Implicit Bias ≠ Euclidean Norm Minimization

Among agents minimizing the cost $\mathcal{C}_{\mathcal{S}}(\cdot)$, $\mathbf{K}_{\text{no-gen}}$ has minimal Euclidean norm

➡ Generalization means GD **does not implicitly minimize Euclidean norm**

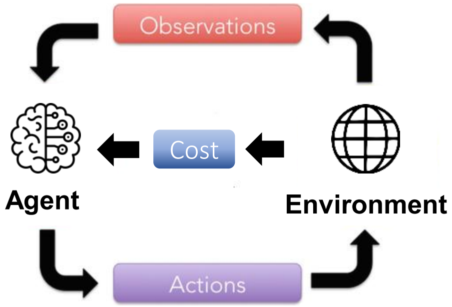
Supervised Learning



Setting:
Linear prediction with quadratic loss

Known: (e.g. Zhang et al. 2017)
Implicit bias minimizes Euclidean norm

Linear-Quadratic Control



Setting:
Linear environment with quadratic cost

Our Results:
Implicit bias does **not** minimize Euclidean norm

Non-Linear Environments and Neural Network Agents

Our Theory

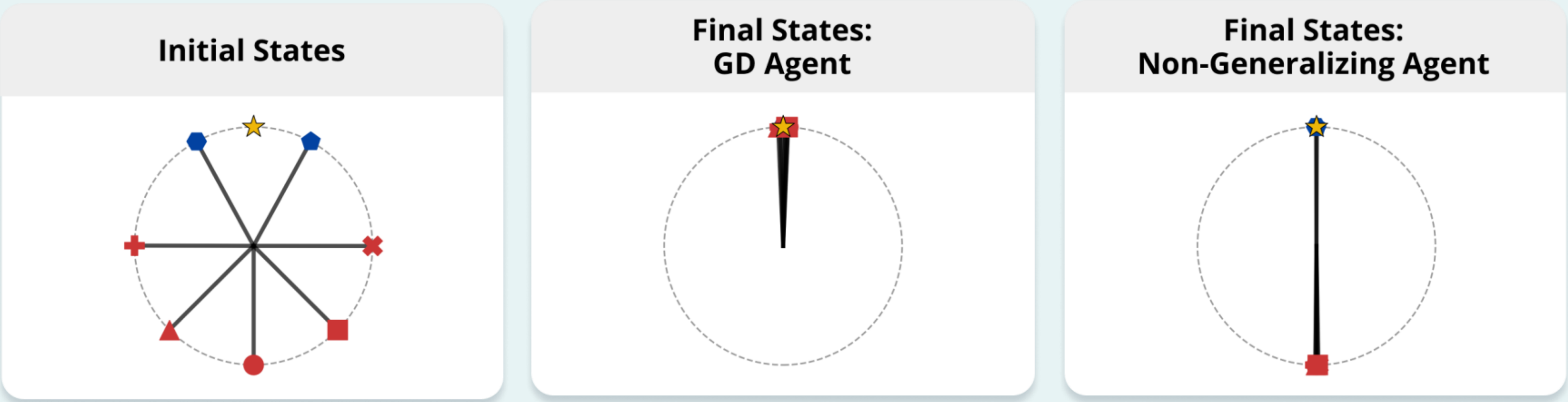
Linear environment induces exploration from seen initial states ➡ linear agent typically generalizes

Experiments

Demonstrate phenomenon and show it extends to **non-linear environments and NN agents**

Pendulum Control Problem
(analogous experiments for a quadcopter control problem)

☆ target state
● initial state seen in training
● initial state unseen in training



GD agent generalizes despite existence of non-generalizing agents!

Outline

Generalization to Unseen Initial States *(RACGGC, ICML’24)*

Generalization to Unseen Tasks with Safety Requirements *(SASNC, Preprint’26)*

Conclusion

Generalization to Unseen Tasks with Safety Requirements

Linear Environment	Quadratic Cost	Linear Agent
$\mathbf{x}_{h+1} = \mathbf{A}\mathbf{x}_h + \mathbf{B}\mathbf{u}_h + \mathbf{D}\mathbf{w}_h$	$\mathbb{E}\left[\sum_{h=0}^{\infty} \mathbf{x}_h^\top \mathbf{Q}\mathbf{x}_h + \mathbf{u}_h^\top \mathbf{R}\mathbf{u}_h\right]$	$\mathbf{u}_h = \mathbf{K}\mathbf{x}_h$

Considered Setting

- **Safety requirements** formulated as risk handling (H_∞-robustness)
- **Multi-task**:
  - State cost matrix $\mathbf{Q}$ varies
  - Agent is task-aware $\mathbf{K} = \mathbf{K}(\mathbf{Q})$
- **Known results**: for every $\mathbf{Q}$, linear agent is optimal both with and without safety requirements

  ➡ denote optimal agents by $\mathbf{K}_{\text{safe}}(\mathbf{Q})$ and $\mathbf{K}_{\text{unsafe}}(\mathbf{Q})$

  Comparing costs would be unfair
- Compare learning with vs. without safety requirements via **imitation** of $\mathbf{K}_{\text{safe}}(\mathbf{Q})$ vs. $\mathbf{K}_{\text{unsafe}}(\mathbf{Q})$
- For each task $\mathbf{Q}$ seen in training, data is non-degenerate: $\mathbf{K}_{\text{safe}}(\mathbf{Q})$ and $\mathbf{K}_{\text{unsafe}}(\mathbf{Q})$ are given

Generalization to Unseen Tasks with Safety Requirements (cont')

Linear Environment
$\mathbf{x}_{h+1} = \mathbf{A}\mathbf{x}_h + \mathbf{B}\mathbf{u}_h + \mathbf{D}\mathbf{w}_h$

Quadratic Cost
$\mathbb{E}\left[\sum_{h=0}^{\infty} \mathbf{x}_h^\top \mathbf{Q}\mathbf{x}_h + \mathbf{u}_h^\top \mathbf{R}\mathbf{u}_h\right]$

Multi-Task
$\mathbf{Q}$ varies

Linear Agent
$\mathbf{u}_h = \mathbf{K}(\mathbf{Q})\mathbf{x}_h$

Safety Requirements
H_∞-robustness

Optimal Agents
$\mathbf{K}_{\text{safe}}(\mathbf{Q})$ and $\mathbf{K}_{\text{unsafe}}(\mathbf{Q})$

Q: Is it easier to learn $\mathbf{K}_{\text{safe}}(\cdot)$ or $\mathbf{K}_{\text{unsafe}}(\cdot)$?

⟷

Q: Is it easier to generalize to unseen task with or without safety requirements?

As surrogate for learnability of $\mathbf{K}_{\text{safe}}(\cdot)$ and $\mathbf{K}_{\text{unsafe}}(\cdot)$, we analyze their **Lipschitz constants**

Upper Bound for Lipschitz Constant of $\mathbf{K}_{\text{unsafe}}(\cdot)$

Lemma

Under technical assumptions:

$$\text{Lip}(\mathbf{K}_{\text{unsafe}}) \leq 2\|A\|_2\|B\|_2\|R^{-1}\|_2(1 + 2\|B\|_2^2\|R^{-1})$$

Technical contribution of independent interest ($\mathbf{K}_{\text{unsafe}}(\mathbf{Q})$ is the well-known LQR)

Proof Sketch

$\mathbf{K}_{\text{unsafe}}(\mathbf{Q})$ given by function of $\mathbf{A}$, $\mathbf{B}$, $\mathbf{R}$ and unique fixed point of Riccati operator $\mathbf{P}(\mathbf{Q})$

For any $\mathbf{Q}$, assumptions guarantee fixed points are bounded and operators are contracting

➡ Bound on Lipschitz constant of mapping $\mathbf{Q} \mapsto \mathbf{P}(\mathbf{Q})$

Propagating the bound through the function for $\mathbf{K}_{\text{unsafe}}(\mathbf{Q})$ yields desired result

Separation Between Lipschitz Constants of $\mathbf{K}_{\mathrm{unsafe}}(\cdot)$ and $\mathbf{K}_{\mathrm{safe}}(\cdot)$

Theorem

Under technical assumptions:

$$\frac{\mathrm{Lip}(\mathbf{K}_{\mathrm{safe}})}{\mathrm{Lip}(\mathbf{K}_{\mathrm{unsafe}})} \geq \Omega\left(\frac{1}{\|A\|_2(2+4\|B\|_2^2\|R^{-1}\|_2)}\right)$$

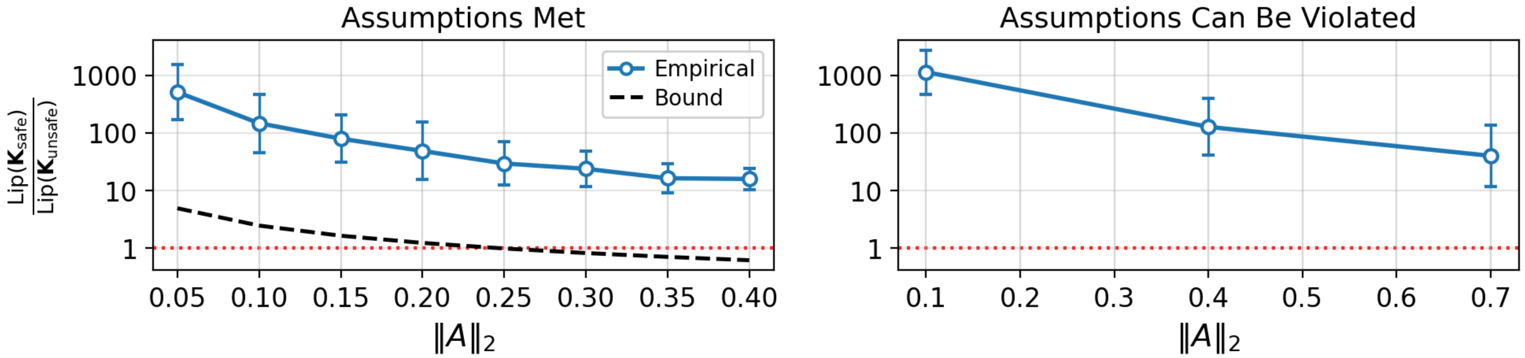
Proof Sketch

Under assumptions, we derive explicit expressions for some components of $\mathbf{K}_{\mathrm{safe}}(\cdot)$

$\mathrm{Lip}(\mathbf{K}_{\mathrm{safe}})$ is lower bounded by derivatives of these components

➡ Separation follows from combination with upper bound on $\mathrm{Lip}(\mathbf{K}_{\mathrm{unsafe}})$

Empirical Validation



Generalization to unseen tasks is **harder with safety requirements**, regardless of how well they are met on seen tasks

Experiments: Linear-Quadratic Control

Imitation error on **seen tasks** with **unseen data**

Imitation error on **unseen tasks**

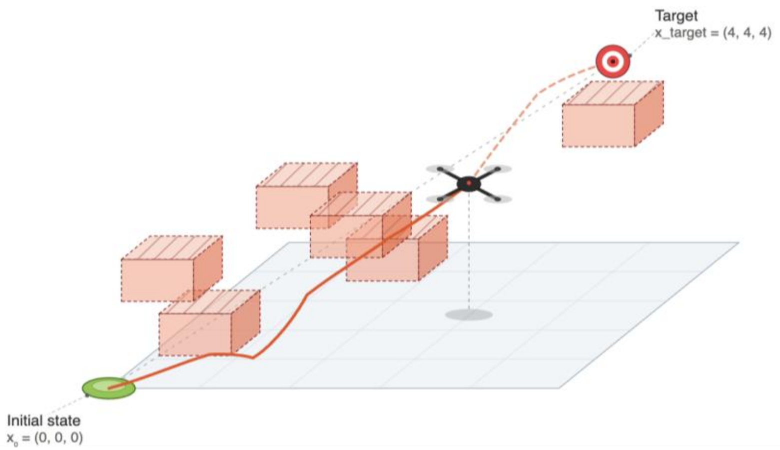
Setting	Teacher	Θ_{train} Error	Θ_{test} Error
Linear-Quadratic, Infinite Sample	Safe	–	1265.53 ± 310.44
	Unsafe	–	$\mathbf{14.16 \pm 2.43}$
Linear-Quadratic, Finite Sample	Safe	2.16 ± 0.83	4018.10 ± 604.55
	Unsafe	1.36 ± 0.99	$\mathbf{33.76 \pm 9.61}$

With **safe** teacher **generalization** to unseen tasks is **inferior**, even when generalization within seen tasks is comparable

Experiments: Beyond Linear-Quadratic Control

Quadcopter Navigation via NN Agent

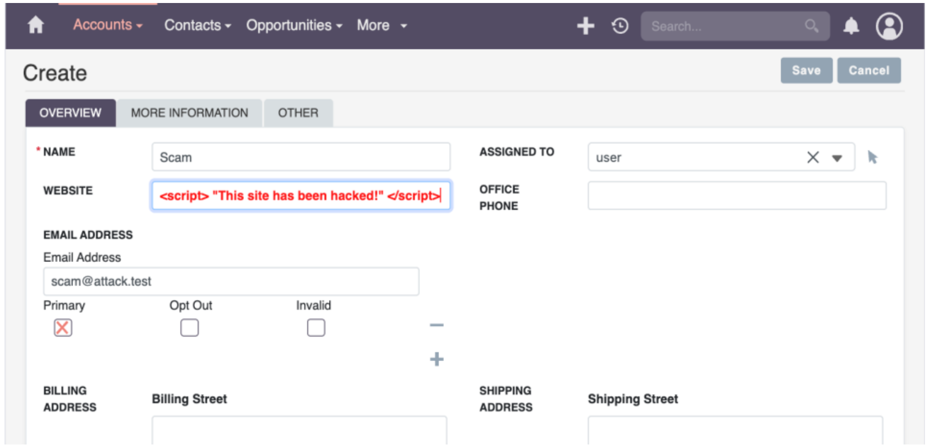
Navigate to task-dependent target while **avoiding obstacles**



Setting	Teacher	Θ_{train} Error	Θ_{test} Error
Quadcopter Navigation	Safe	0.61 ± 0.26	141.74 ± 15.79
	Unsafe	0.35 ± 0.14	$\mathbf{22.90 \pm 4.46}$

CRM via LLM Agent

Complete data-entry jobs while **avoiding bad practices** and **handling risky inputs**



Setting	Teacher	Θ_{train} Error	Θ_{test} Error
CRM via LLM Agent	Safe	16.48 ± 3.46	25.94 ± 4.10
	Unsafe	15.77 ± 2.61	$\mathbf{16.93 \pm 2.14}$

Our conclusions extend beyond linear-quadratic control

Outline

Generalization to Unseen Initial States *(RACGGC, ICML'24)*

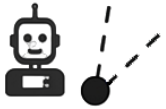
Generalization to Unseen Tasks with Safety Requirements *(SASNC, Preprint'26)*

Conclusion

Recap

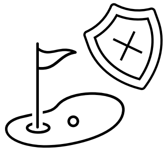
Agents must generalize to unseen conditions and tasks, while obeying safety requirements

Theory: for linear-quadratic control:



Generalization to Unseen Initial States

Implicit bias of GD leads to generalization if and only if exploration induced by environment from initial states seen in training is sufficient



Generalization to Unseen Tasks with Safety Requirements

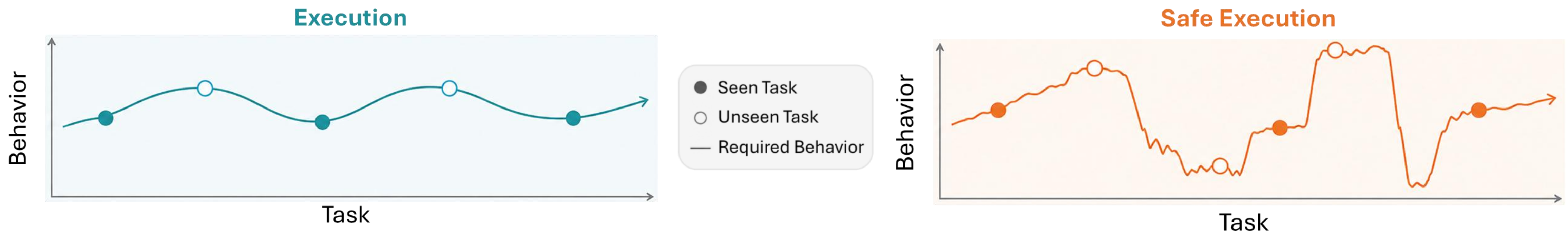
Generalization to unseen tasks is harder with safety requirements, regardless of how well they are met on seen tasks

Experiments: phenomena extend to non-linear settings with NN and LLM agents

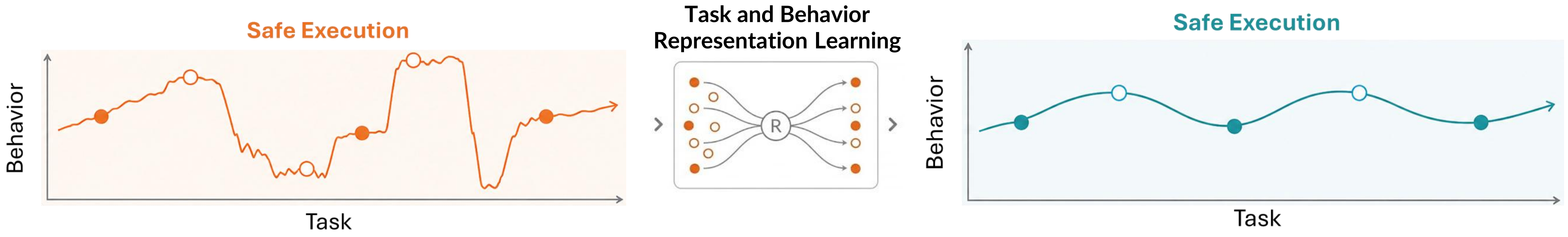
Practical Implication for AI Safety

Our results suggest:

- Current attempts to improve **safety via more training on specific tasks may be insufficient**



- May be **better to learn representations for tasks and safe behaviors** which simplify relationship between them



Thank You!

Work supported by:
ERC Starting Grant (101164614); Apple; Google; Meta; Yandex; ISF Grant (1780/21); Blavatnik Foundation; Adelis Foundation; Tel Aviv University Center for AI and Data Science; and Amnon and Anat Shashua