EPFL

NX-414: Brain-like computation and intelligence

Alexander Mathis
alexander.mathis@epfl.ch

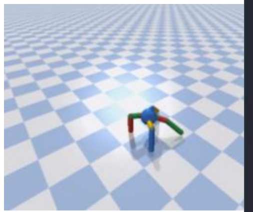
Lecture 13, May 21st

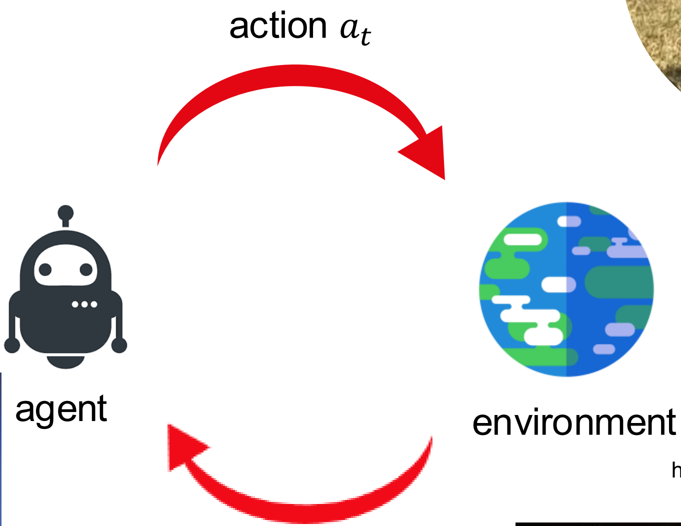
EPFL

Reinforcement learning

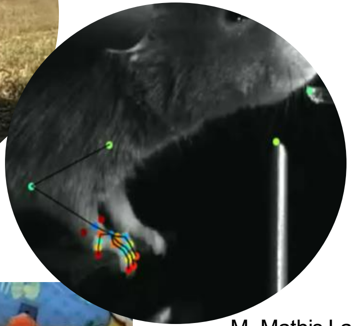
Chess: 10^{120}

Go: 3^{361}





state s_t
reward r_t



Nath*, Mathis* et al.
Nature Protocols 2019

M. Mathis Lab

https://www.youtube.com/watch?v=8vNxjwt2AqY

https://www.symphonikerhamburg.de/konzerte/martha-argerich-70

https://www.forbes.com/

enormous gap

What is missing?

- Exploration (minimal coverage in last lecture)
- *Baked in reward functions (which we don’t know & discuss)*
- Internal models
- Inductive biases (innate architecture)
- Curriculum learning
- Deliberate practice
- Using language
- ….

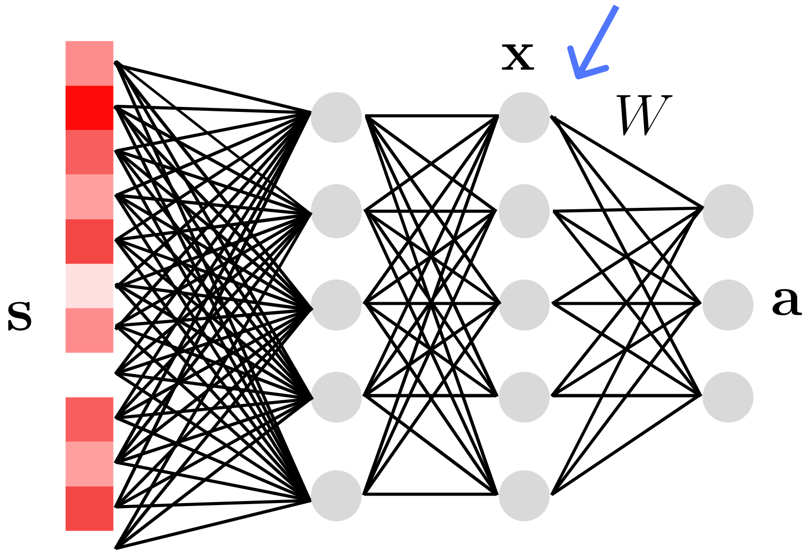
More efficient exploration

Exploration & play

https://www.youtube.com/watch?v=8vNxjwt2AqY

Reminder: Latent time-correlated exploration

LATTICE - **LAT**ent **TI**me-**C**orrelated **E**xploration



Perturbation matrices

$N_{\mathbf{a}}$ $\boxed{P_{\mathbf{a}}}$ $N_{\mathbf{x}}$ $\quad (P_{\mathbf{a}})_{i,j} \sim \mathcal{N}\left(0, (S_{\mathbf{a}})_{i,j}\right)$

$N_{\mathbf{x}}$ $\boxed{P_{\mathbf{x}}}$ $\quad (P_{\mathbf{x}})_{i,j} \sim \mathcal{N}\left(0, (S_{\mathbf{x}})_{i,j}\right)$

LATTICE $\mathbf{a} = (W + P_{\mathbf{a}} + W P_{\mathbf{x}})\mathbf{x}$

State-Dependent Exploration

gSDE $\mathbf{a} = (W + P_{\mathbf{a}})\mathbf{x}$

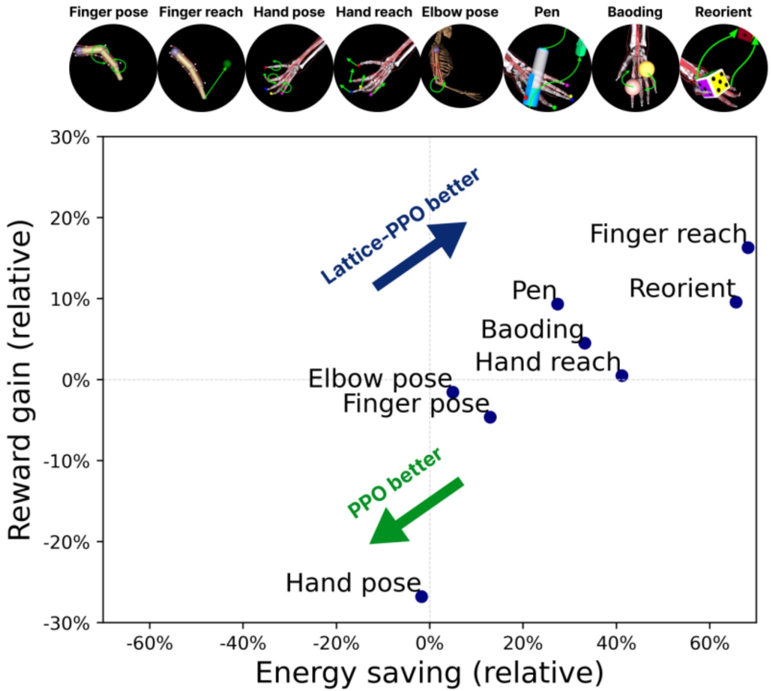
Default $\mathbf{a} = W\mathbf{x} + \epsilon$

Time

Time + Action

Chiappa, Marin Vargas, Huang, Mathis, NeurIPS /arxiv 2023

Lattice learns more energy efficient solutions



Chiappa, Marin Vargas, Huang, Mathis, NeurIPS /arxiv 2023

Another recent example for better exploration

Published as a conference paper at ICLR 2023

DEP-RL: Embodied Exploration for Reinforcement Learning in Overactuated and Musculoskeletal Systems

Pierre Schumacher[1,2] Daniel F.B. Haeufle[2,3] Dieter Büchler[1] Syn Schmitt[3] Georg Martius[1]

[1]Max Planck Institute for Intelligent Systems, Tübingen, Germany
[2]Hertie-Institute for Clinical Brain Research, Tübingen, Germany
[3]Institute for Modelling and Simulation of Biomechanical Systems, Stuttgart, Germany

$a \in \mathbb{R}^{2\dots600}$ $a \in \mathbb{R}^{6\dots600}$ $a \in \mathbb{R}^{50}$ $a \in \mathbb{R}^{52}$ $a \in \mathbb{R}^{120}$ $a \in \mathbb{R}^{18}$ $a \in \mathbb{R}^{18}$

Internal models

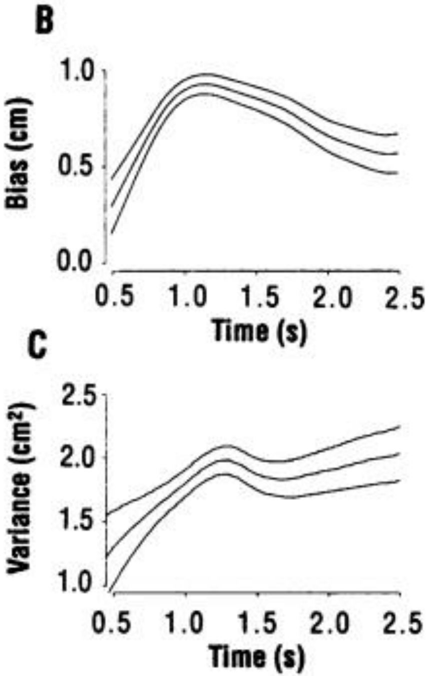
EPFL

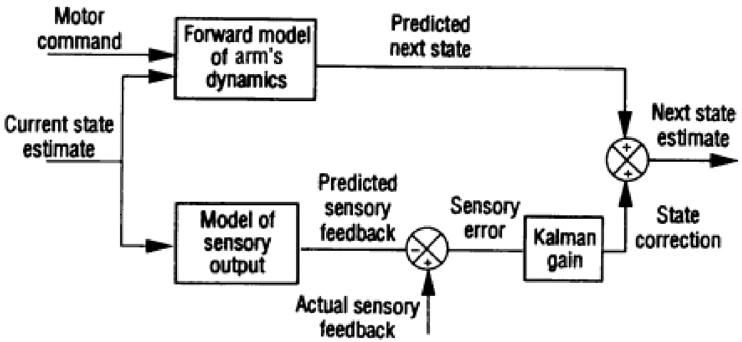
An Internal Model for Sensorimotor Integration

Daniel M. Wolpert,* Zoubin Ghahramani, Michael I. Jordan

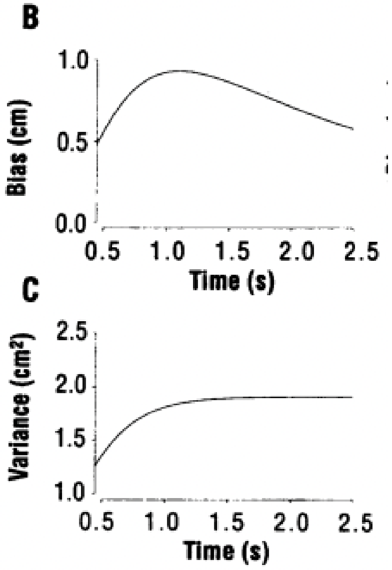
On the basis of computational studies it has been proposed that the central nervous system internally simulates the dynamic behavior of the motor system in planning, control, and learning; the existence and use of such an internal model is still under debate. A sensorimotor integration task was investigated in which participants estimated the location of one of their hands at the end of movements made in the dark and under externally imposed forces. The temporal propagation of errors in this task was analyzed within the theoretical framework of optimal state estimation. These results provide direct support for the existence of an internal model.

Data





Prediction

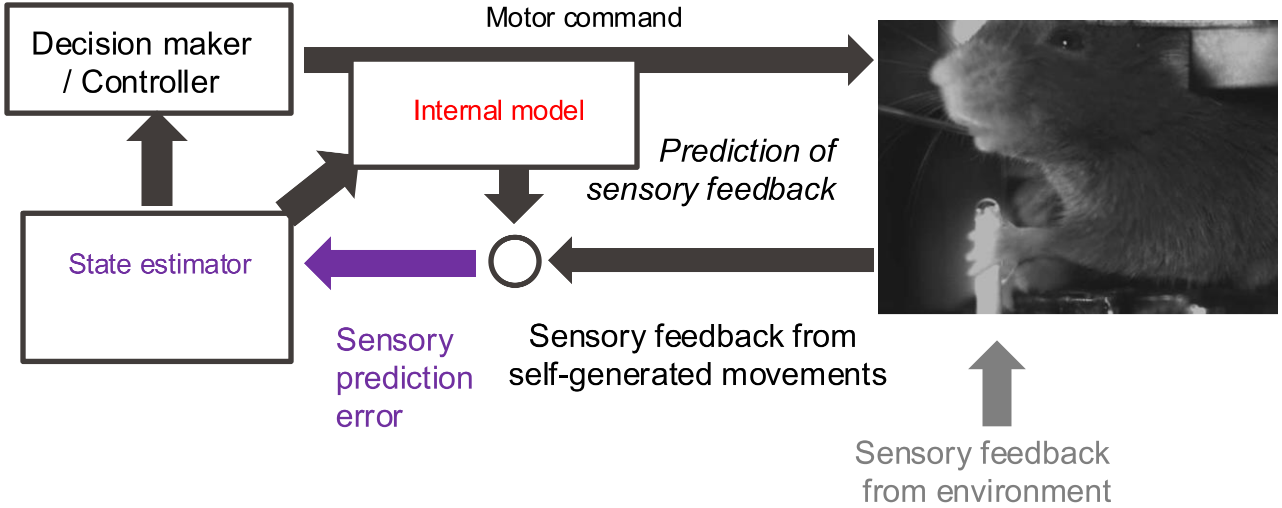


Wolpert, Ghahramani, Jordan, Science 199

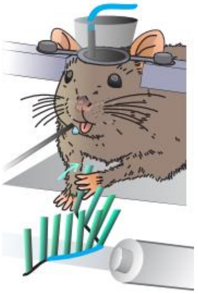
Somatosensory cortex updates the internal model

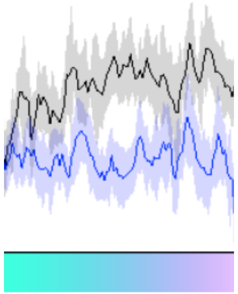
1 — Did inactivation block **acquisition?**

2 — Or block **expression** of an adapted motor command?



$$m_{k+1} = \underbrace{\widehat{m}_k}_{\text{“= 0” (memory hyp.)}} + \underbrace{K_k \, (s - \hat{s})}_{\text{“= 0” (updating hyp.)}}$$

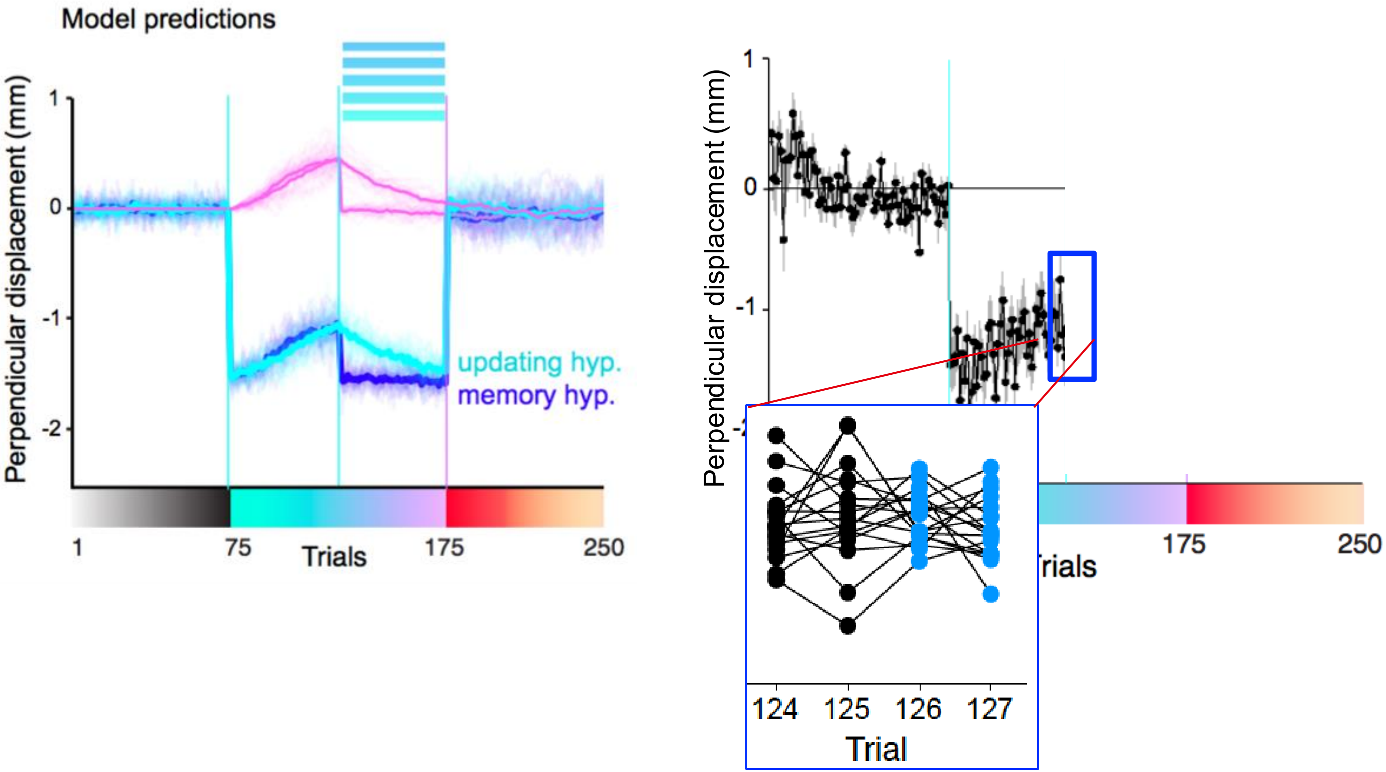




Slide, courtesy M.W. Mathis

M.W. Mathis, A. Mathis, N. Uchida, Neuron 2017

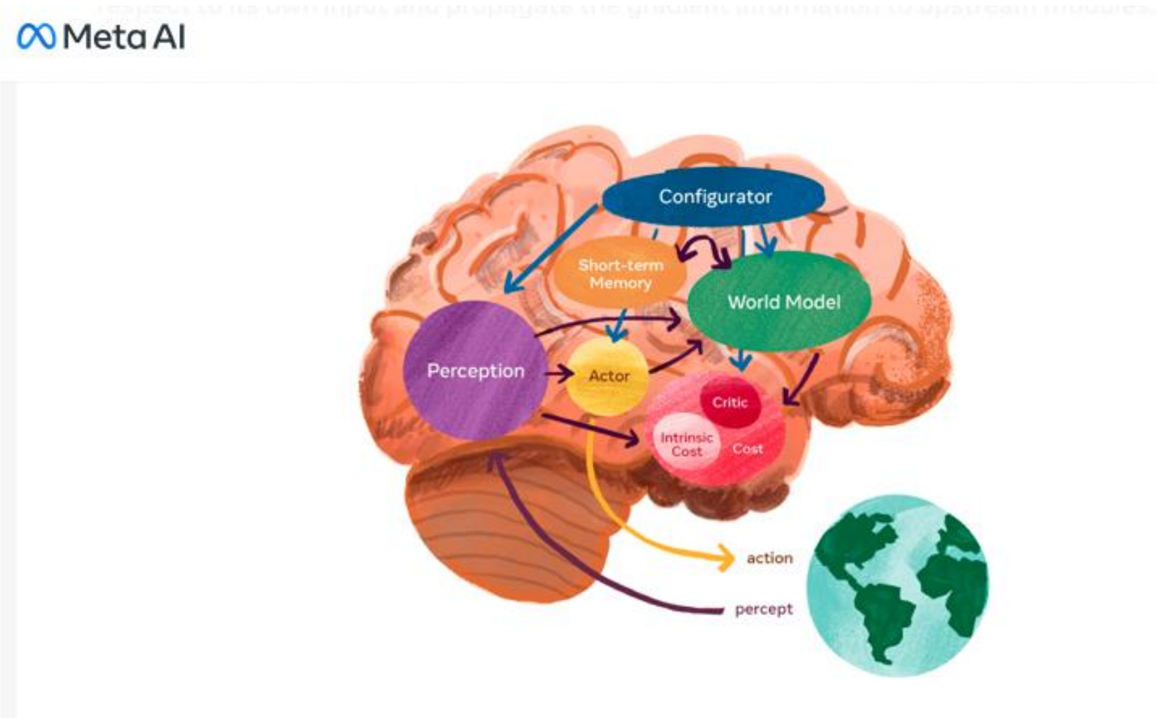
S1 inactivation after adaptation:
S1 does not exclusively house the model of the perturbation



Slide, courtesy M.W. Mathis

M.W. Mathis, A. Mathis, N. Uchida, Neuron 2017

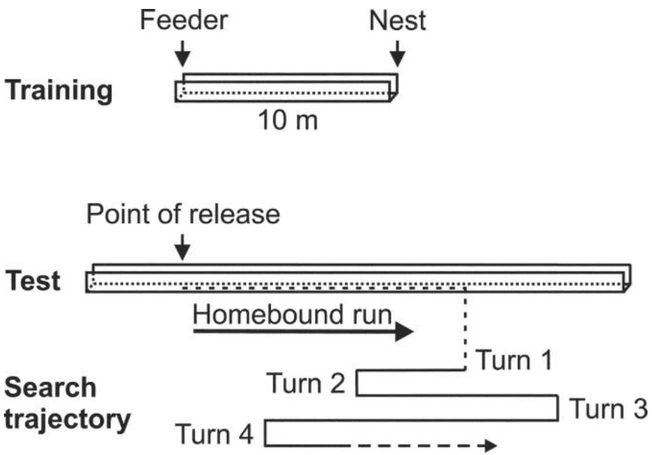
What is missing in AI?

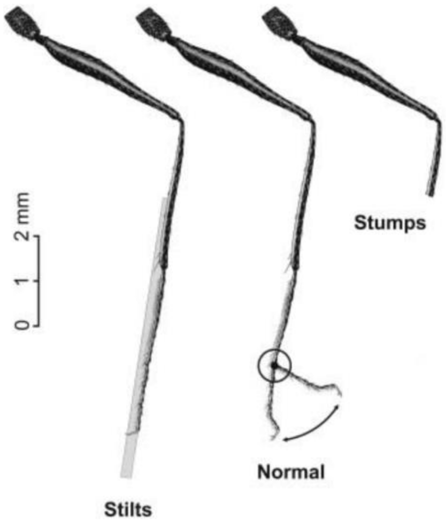


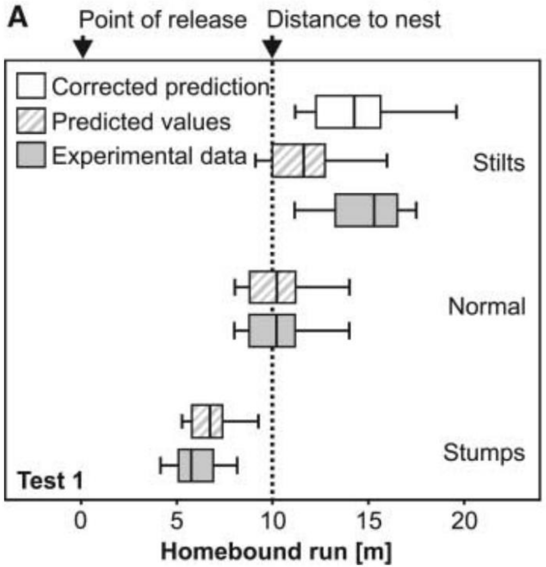
https://ai.facebook.com/blog/yann-lecun-advances-in-ai-research/

Another motor adaptation example

How do ants estimate the distance?

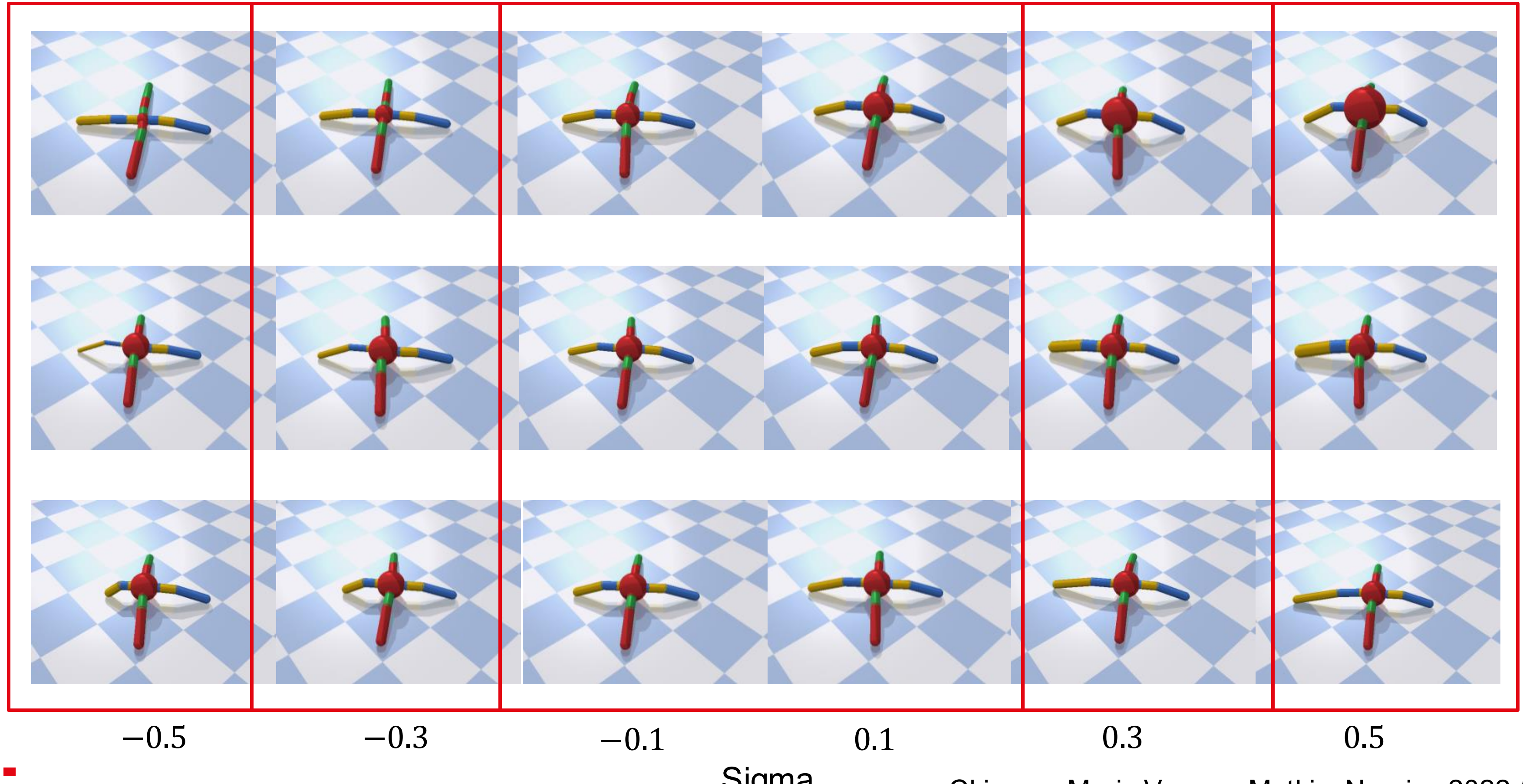






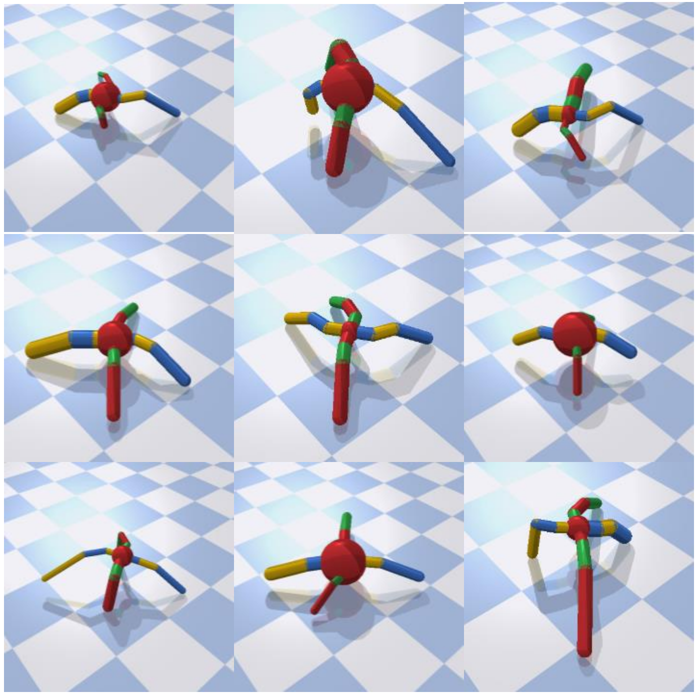
Wittlinger, Wehner & Wolf, Science 2006

How can we control them all?



Chiappa, Marin Vargas, Mathis, Neurips 2022 /arxiv

Morphology perturbation space



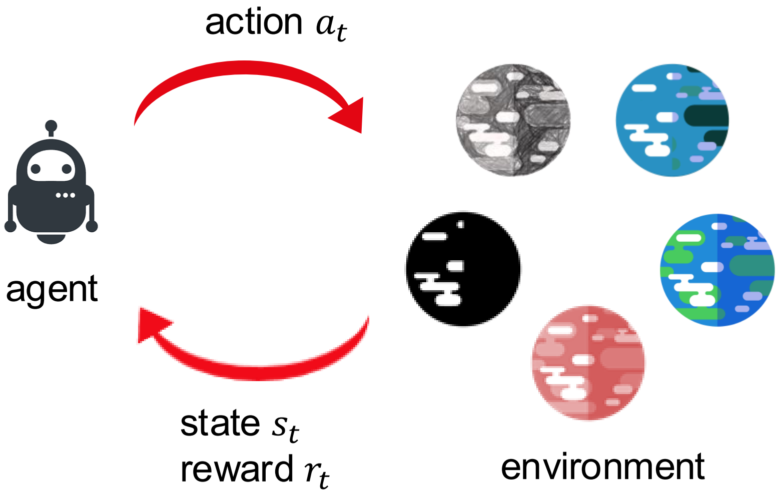
Perturbations:

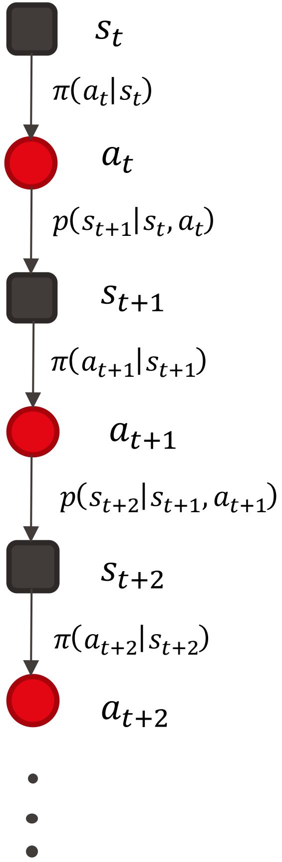
- Torso size
- Limb length
- Limb size

Every episode begins with a different perturbation of the base morphology

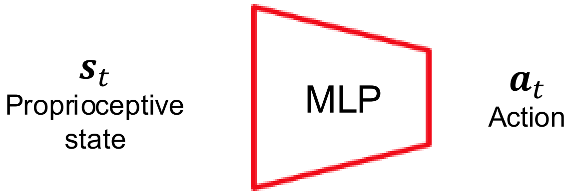
Chiappa, Marin Vargas, Mathis, Neurips 2022 /arxiv

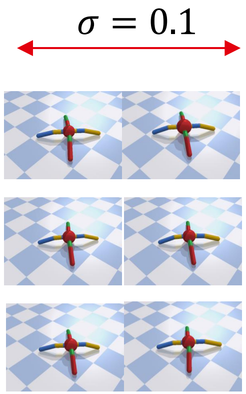
Contextual Markov Decision Process

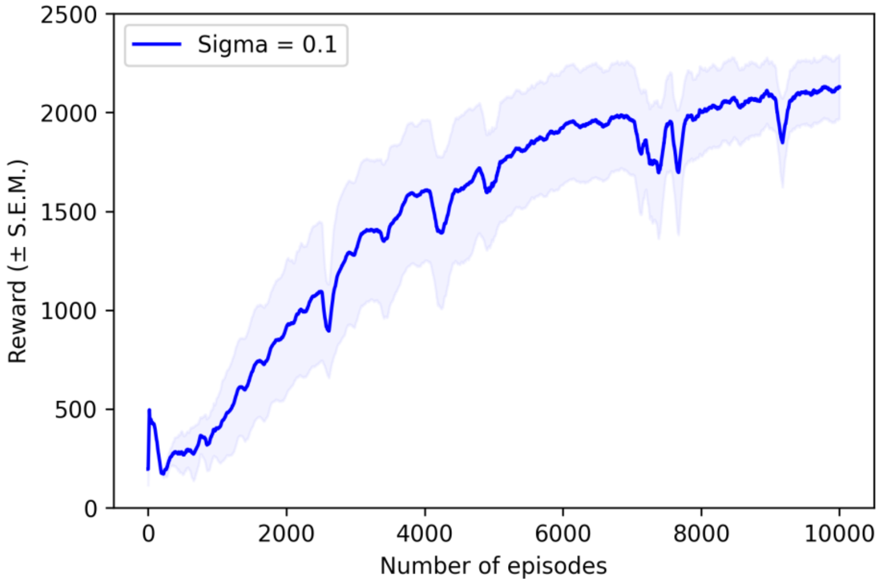




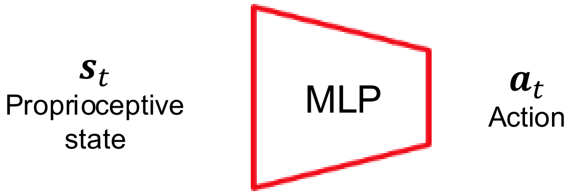
Baseline: multi-layer perceptron (MLP) policy with SAC



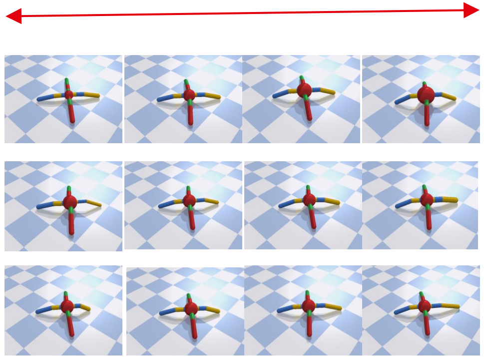


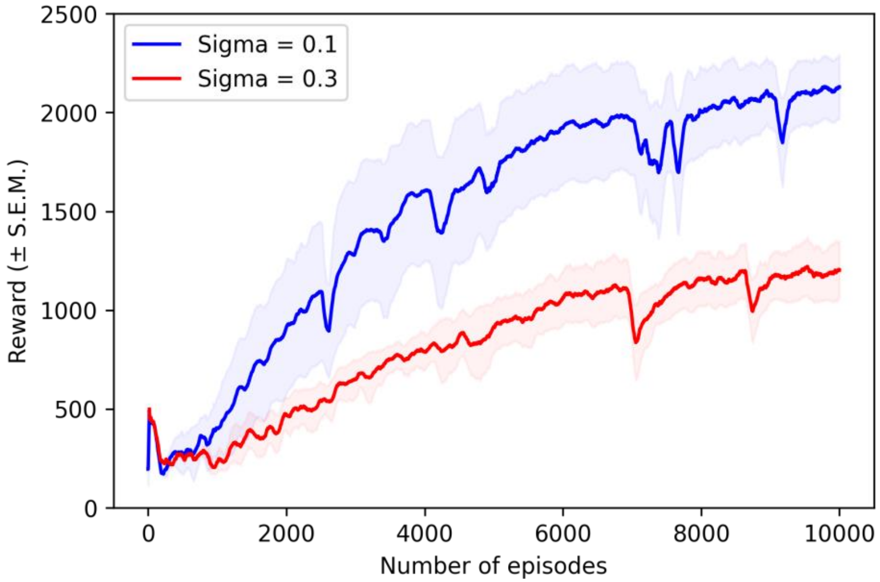


Baseline: MLP policy

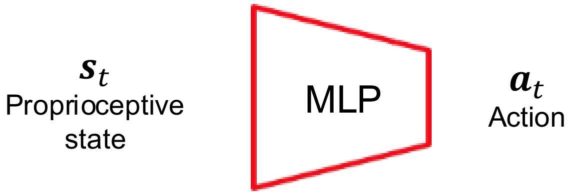


$\sigma = 0.3$

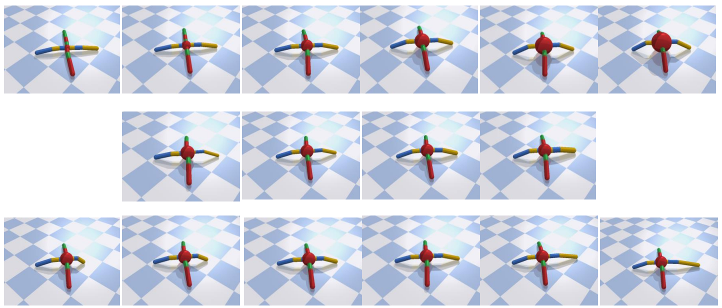


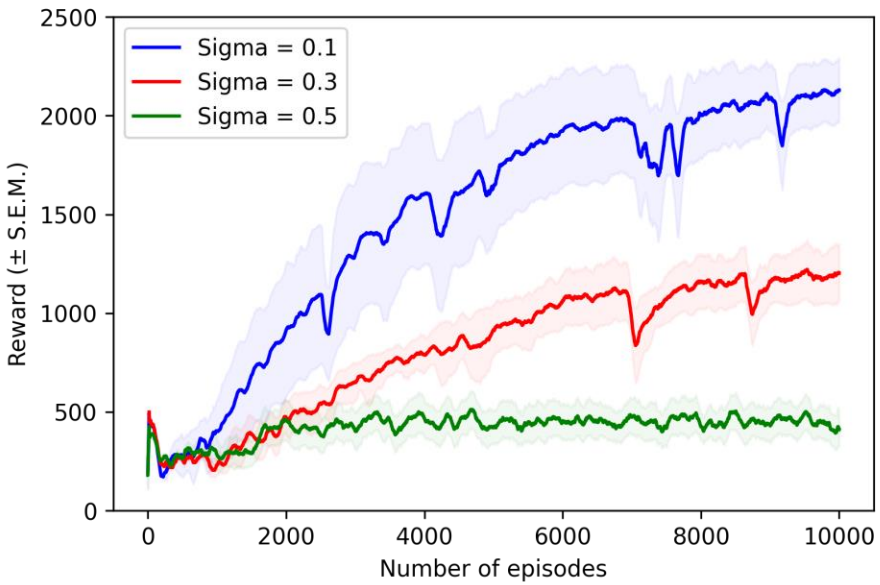


Baseline: MLP policy

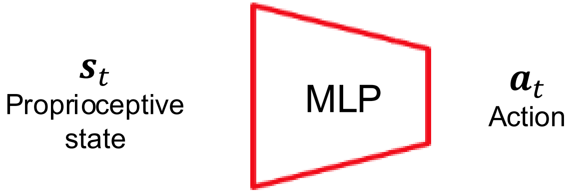


$\sigma = 0.5$

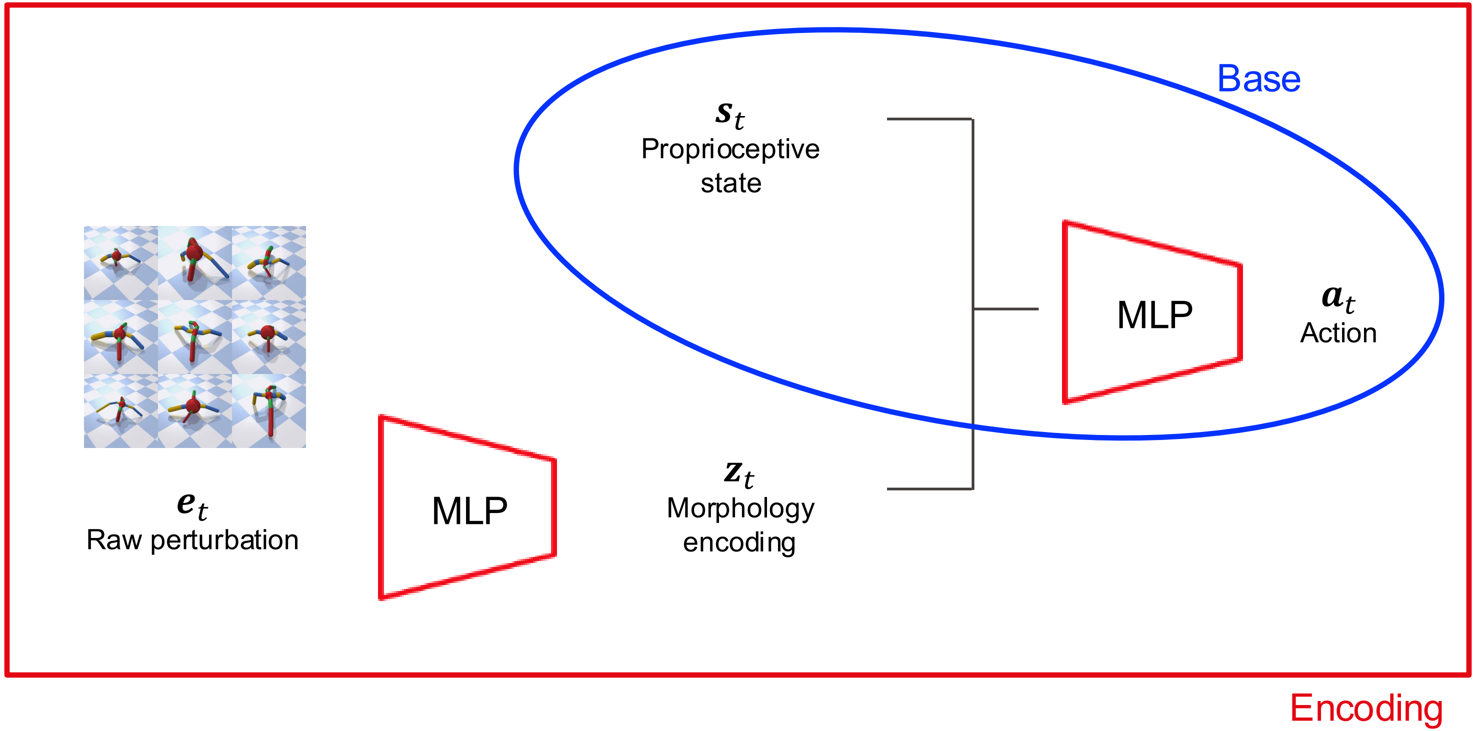




Baseline: MLP policy

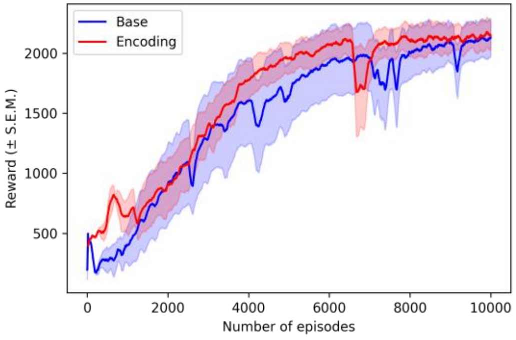


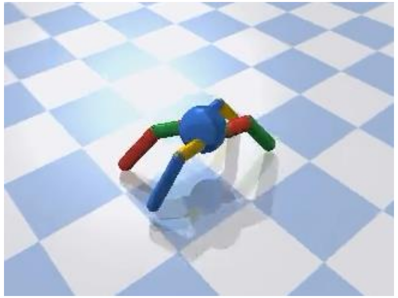
Morphology encoding policy (aka oracle)



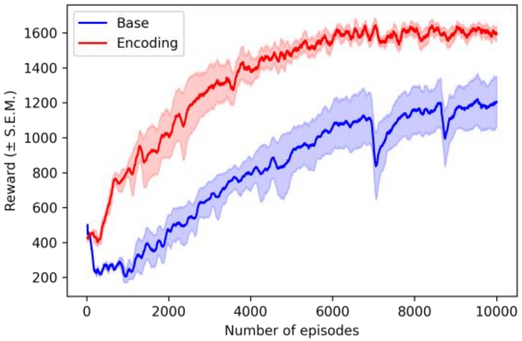
Morphology encoding policy

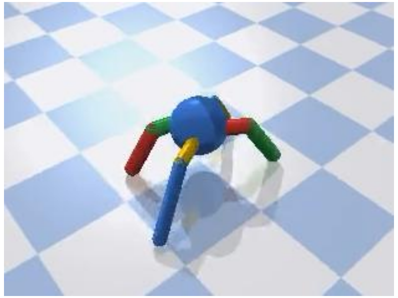
$\sigma = 0.1$



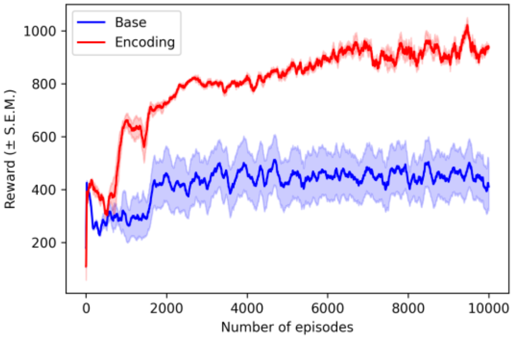


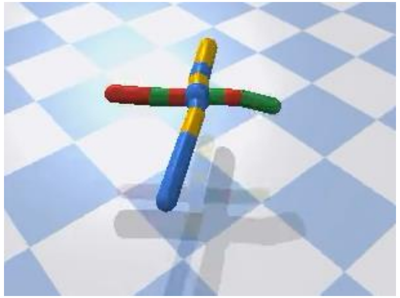
$\sigma = 0.3$





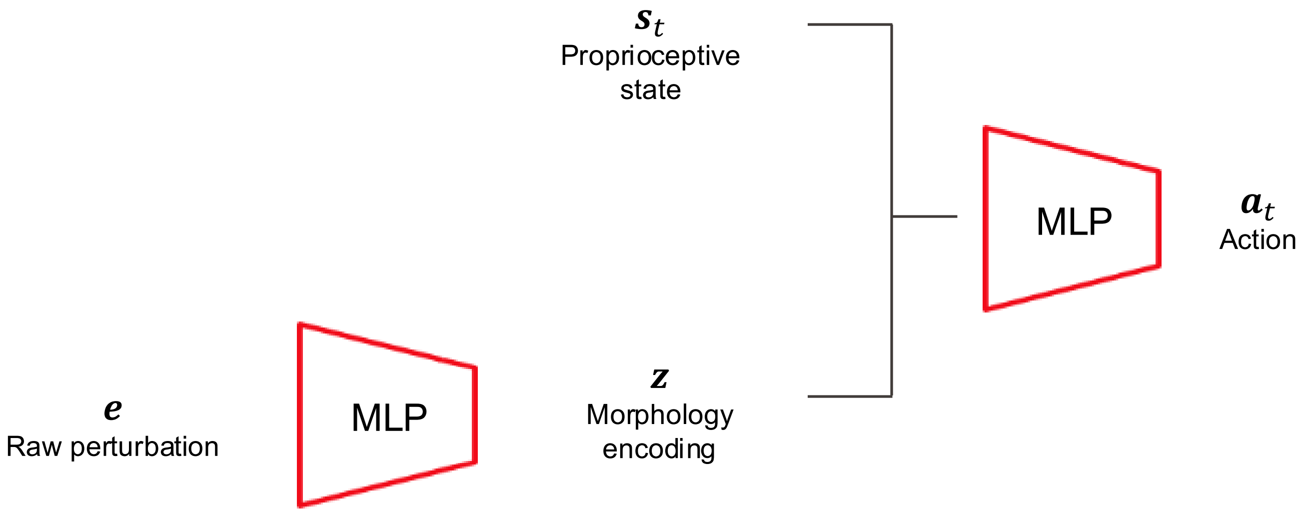
$\sigma = 0.5$





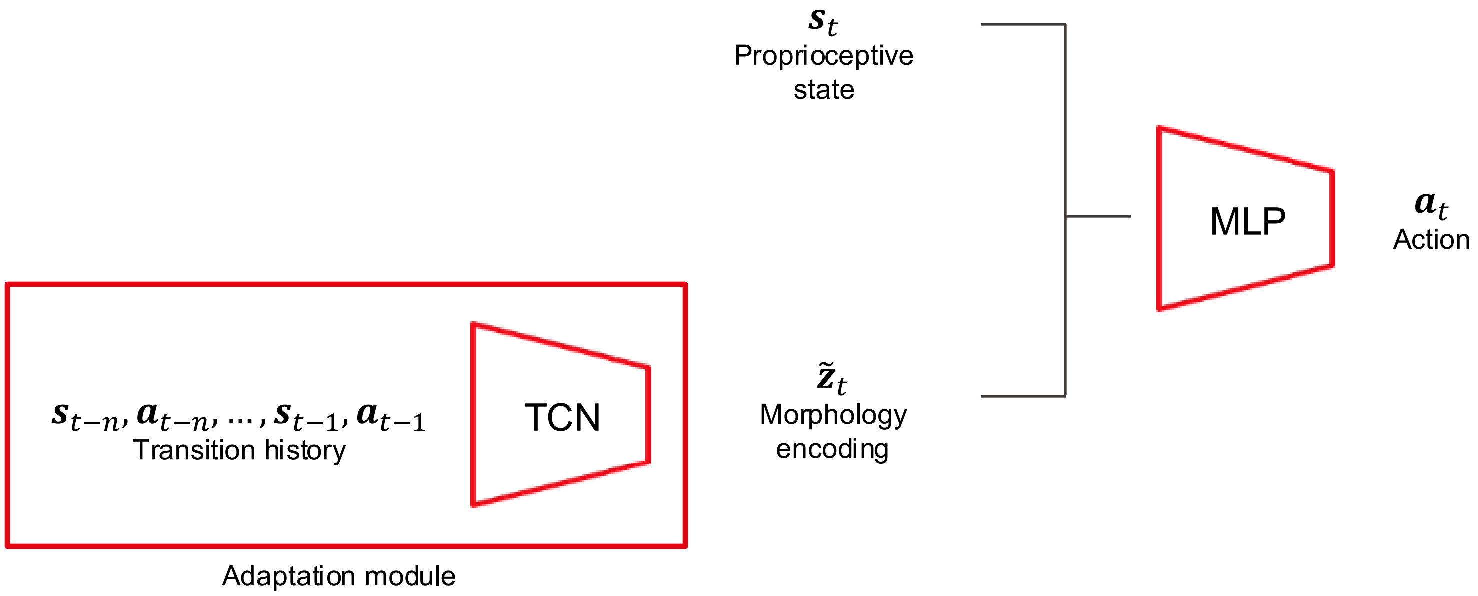
Chiappa, Marin Vargas, Mathis, Neurips 2022 /arxiv

Morphology encoding from experience



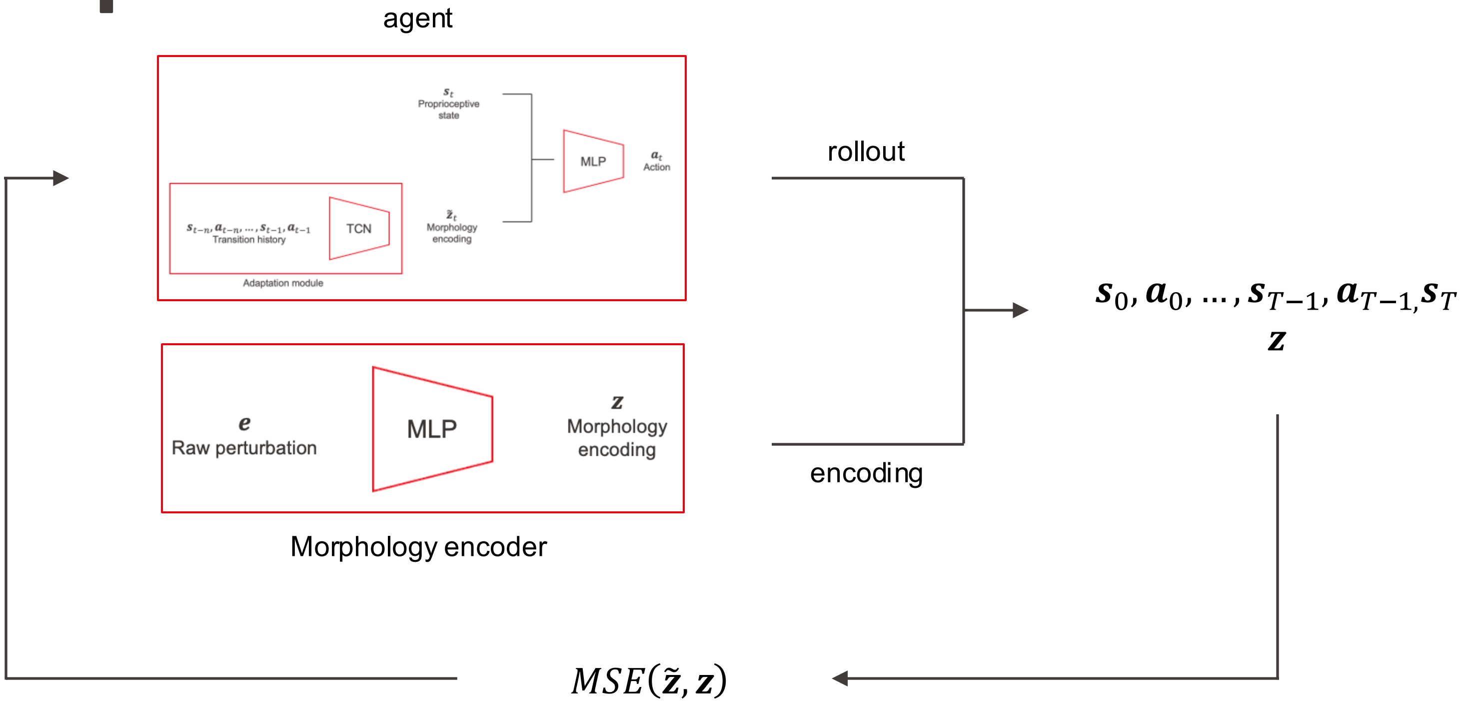
- Kumar, A., Fu, Z., Pathak, D., & Malik, J, “RMA: Rapid Motor Adaptation for Legged Robots, 2021

Morphology encoding from experience



- Kumar, A., Fu, Z., Pathak, D., & Malik, J, "RMA: Rapid Motor Adaptation for Legged Robots, 2021

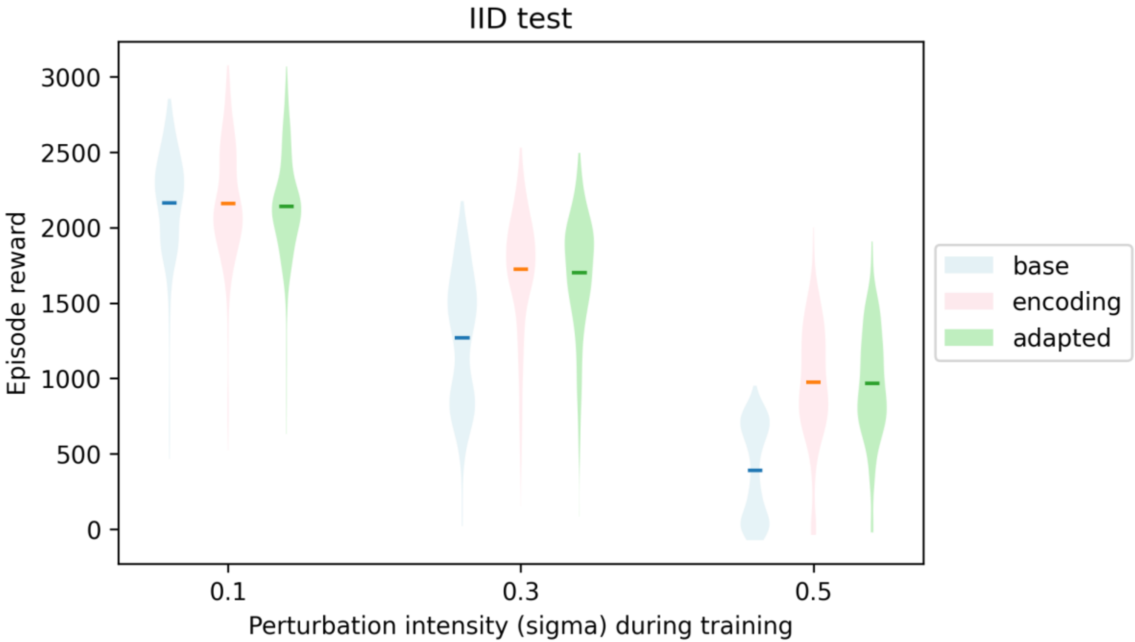
Morphology encoding from experience



- Kumar, A., Fu, Z., Pathak, D., & Malik, J, “RMA: Rapid Motor Adaptation for Legged Robots, 2021

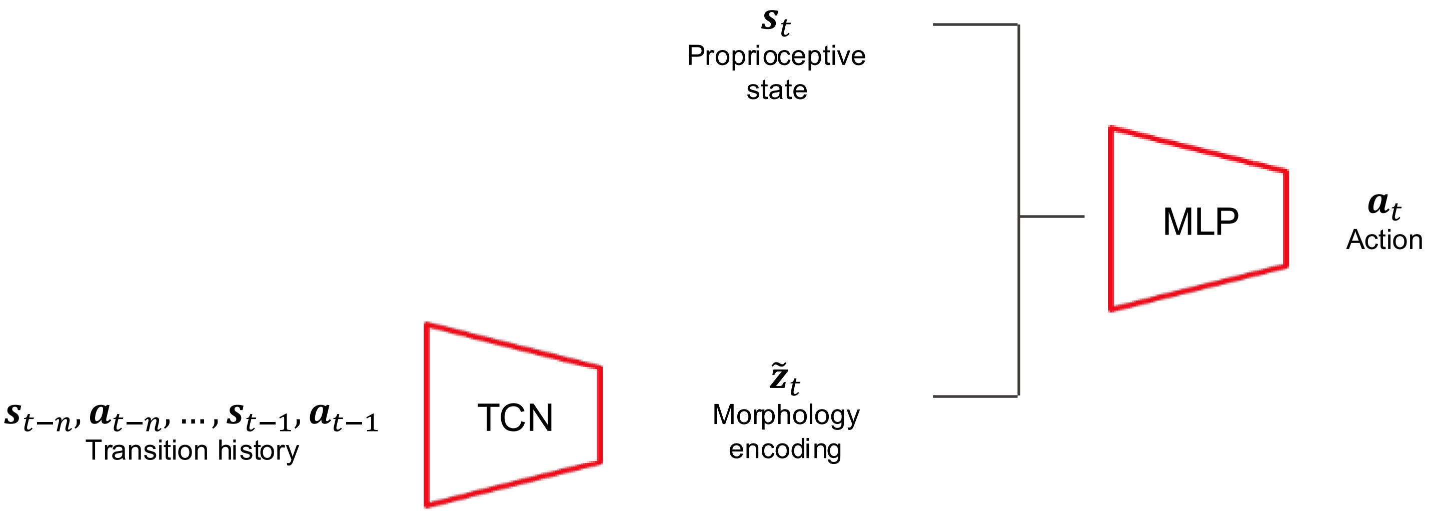
Learning with a perturbed body

100 perturbations per intensity level

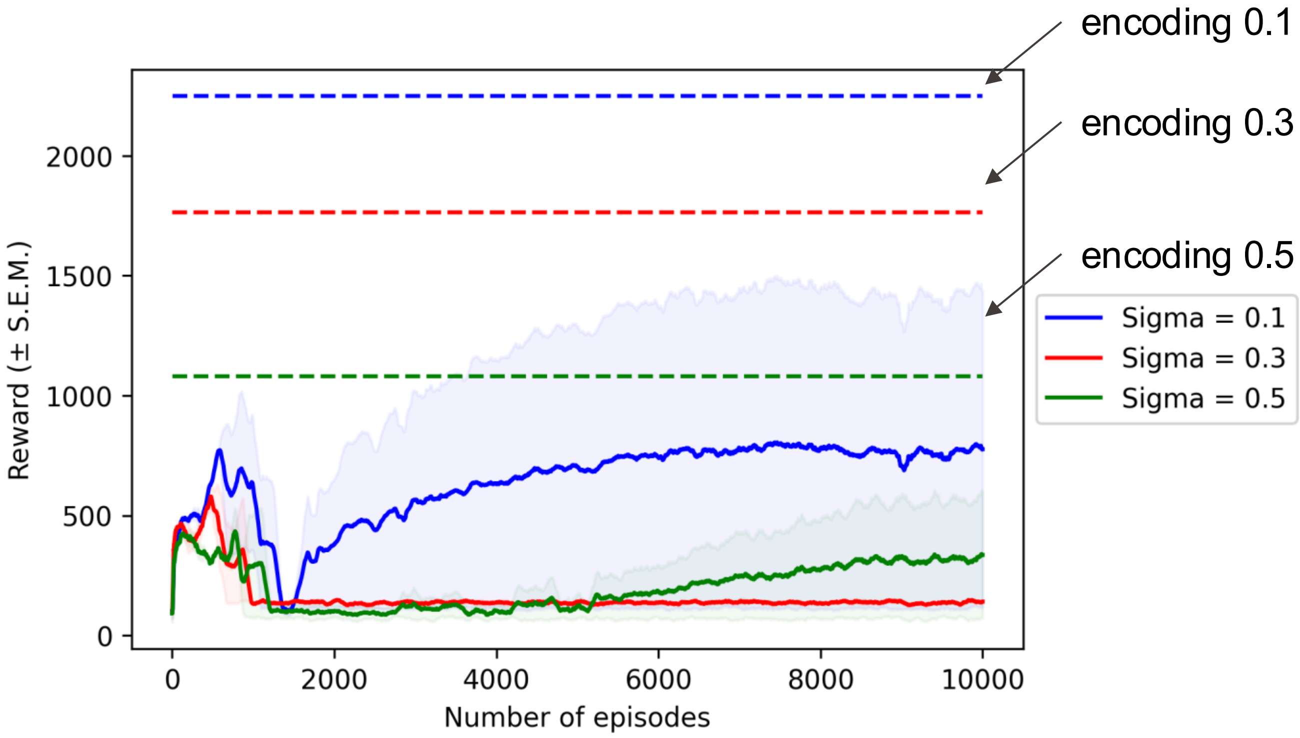


Chiappa, Marin Vargas, Mathis, Neurips 2022 /arxiv

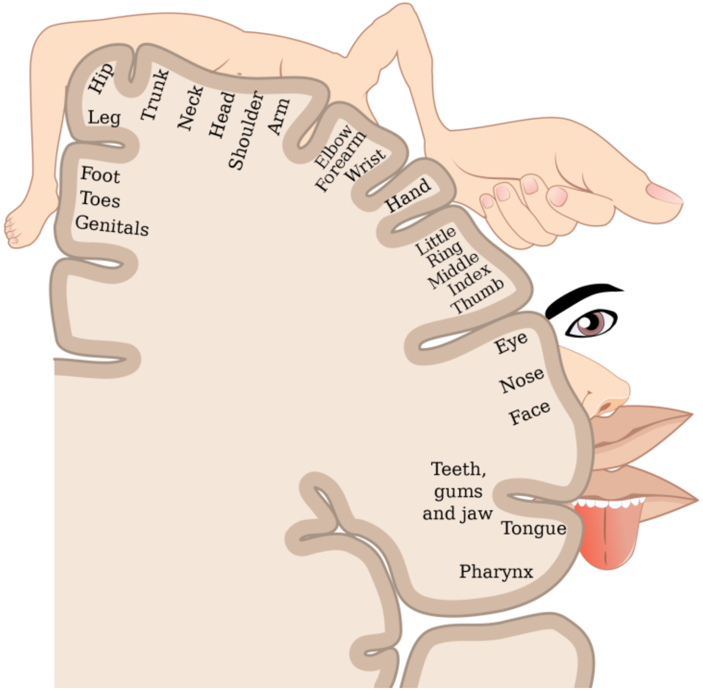
Is the 2-step training necessary?



Training the CNN encoder end-to-end



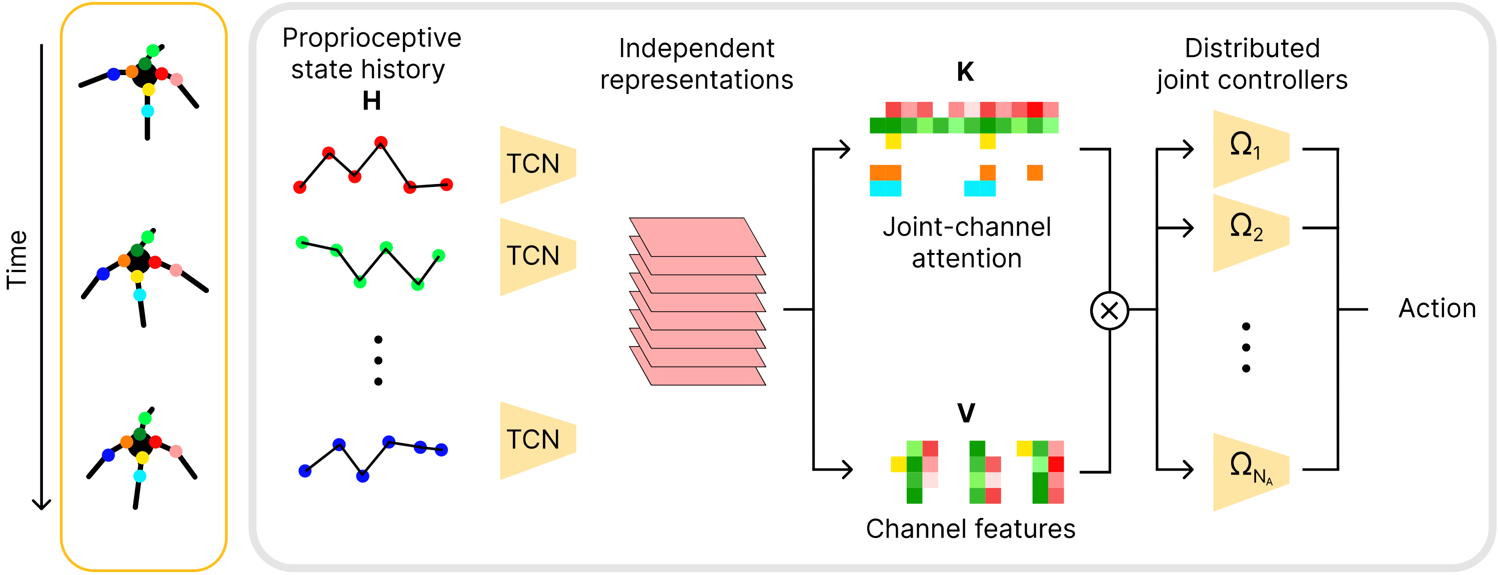
Distributed sensing and control



- Independent low-level processing
- High-level proprioceptive input integration
- Distributed control

DMAP's brain inspired architecture

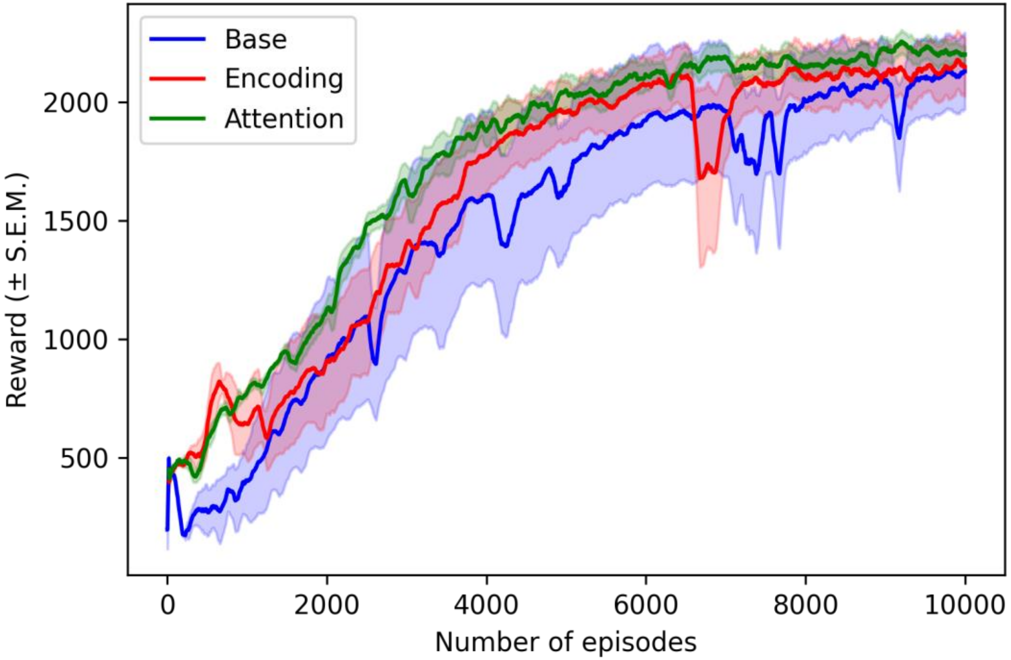
DMAP - **D**istributed **M**orphological **A**ttention **P**olicy



Chiappa, Marin Vargas, Mathis, Neurips 2022 /arxiv

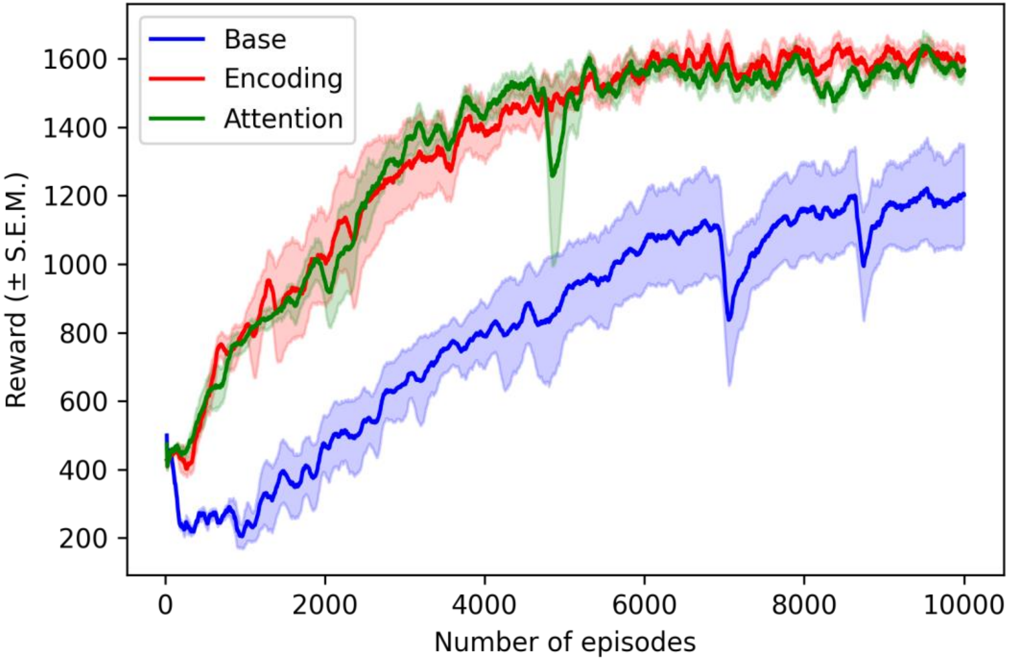
DMAP performance comparison

Sigma = 0.1



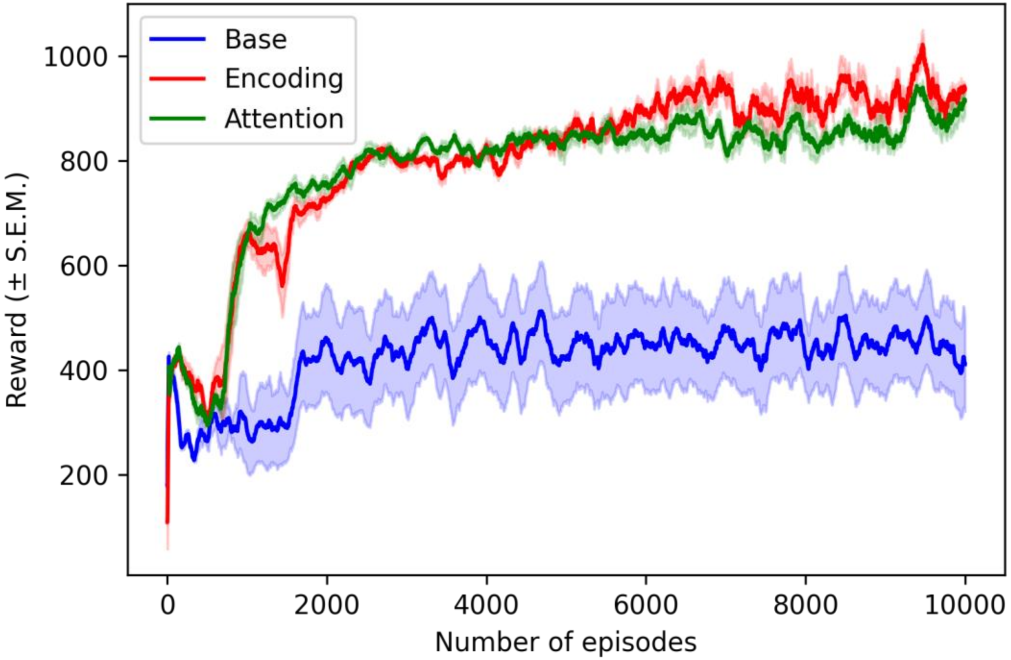
DMAP performance comparison

Sigma = 0.3

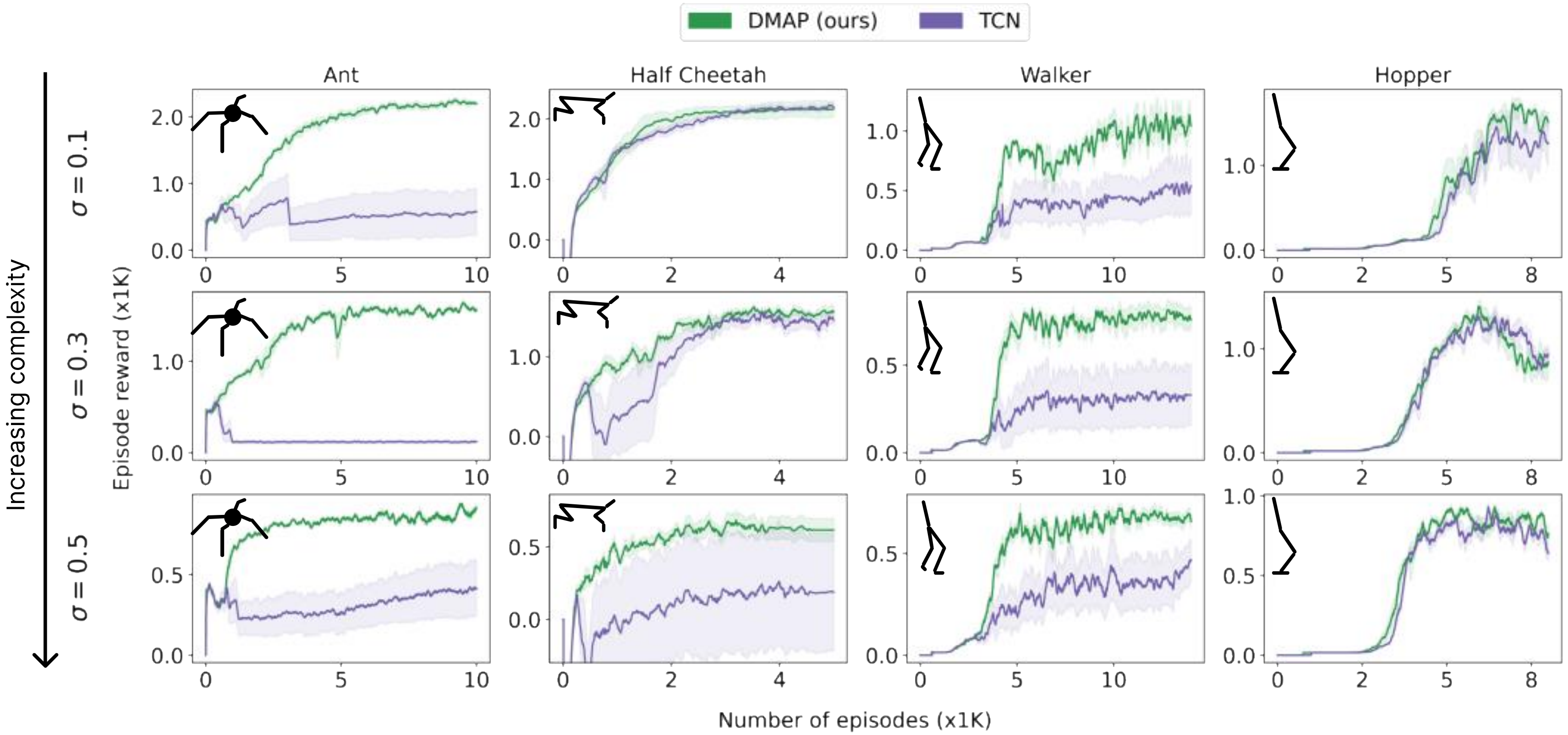


DMAP performance comparison

Sigma = 0.5

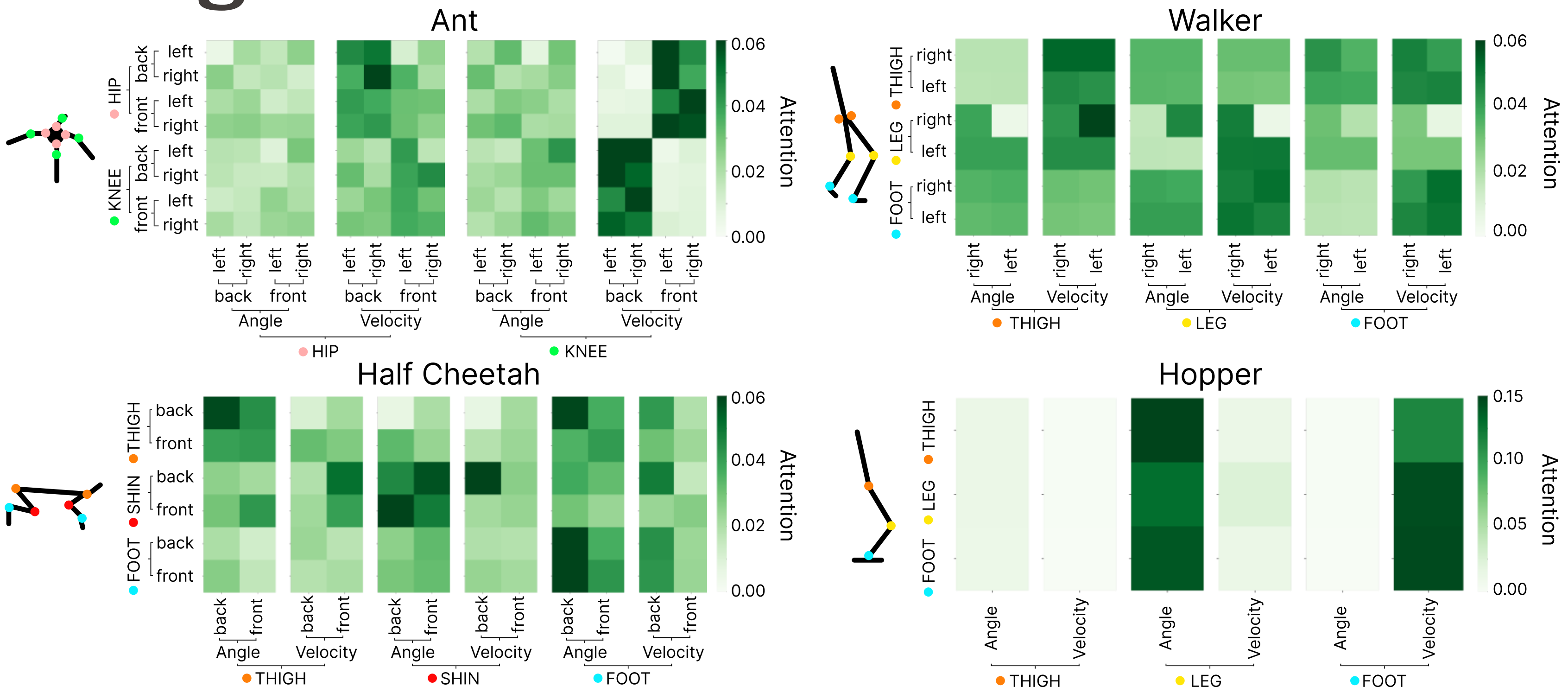


Results also hold across different morphologies



Chiappa, Marin Vargas, Mathis, Neurips 2022 /arxiv

Analysis of the learned attention weights



Inductive biases

- We discussed a brain-inspired inductive bias (DMAP) that implicitly can deal with changing bodies (better than other policies)
- Check out Tony Zador's Perspective in Nat. Comm 2019

A critique of pure learning and what artificial neural networks can learn from animal brains

Anthony M. Zador[1]

Artificial neural networks (ANNs) have undergone a revolution, catalyzed by better supervised learning algorithms. However, in stark contrast to young animals (including humans), training such networks requires enormous numbers of labeled examples, leading to the belief that animals must rely instead mainly on unsupervised learning. Here we argue that most animal behavior is not the result of clever learning algorithms—supervised or unsupervised—but is encoded in the genome. Specifically, animals are born with highly structured brain connectivity, which enables them to learn very rapidly. Because the wiring diagram is far too complex to be specified explicitly in the genome, it must be compressed through a "genomic bottleneck". The genomic bottleneck suggests a path toward ANNs capable of rapid learning.

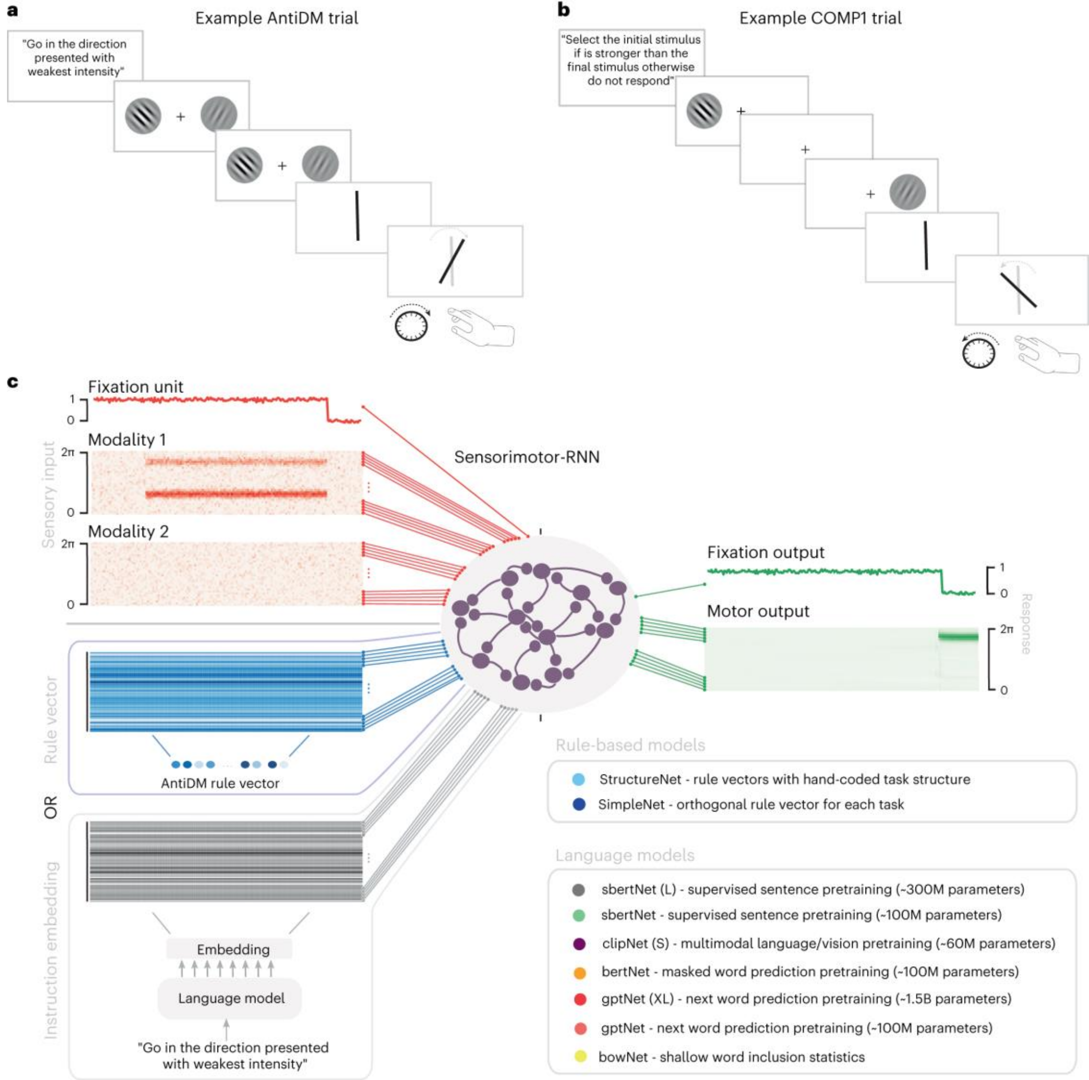
Using Language

Natural language instructions induce compositional generalization in networks of neurons

Reidar Riveland & Alexandre Pouget

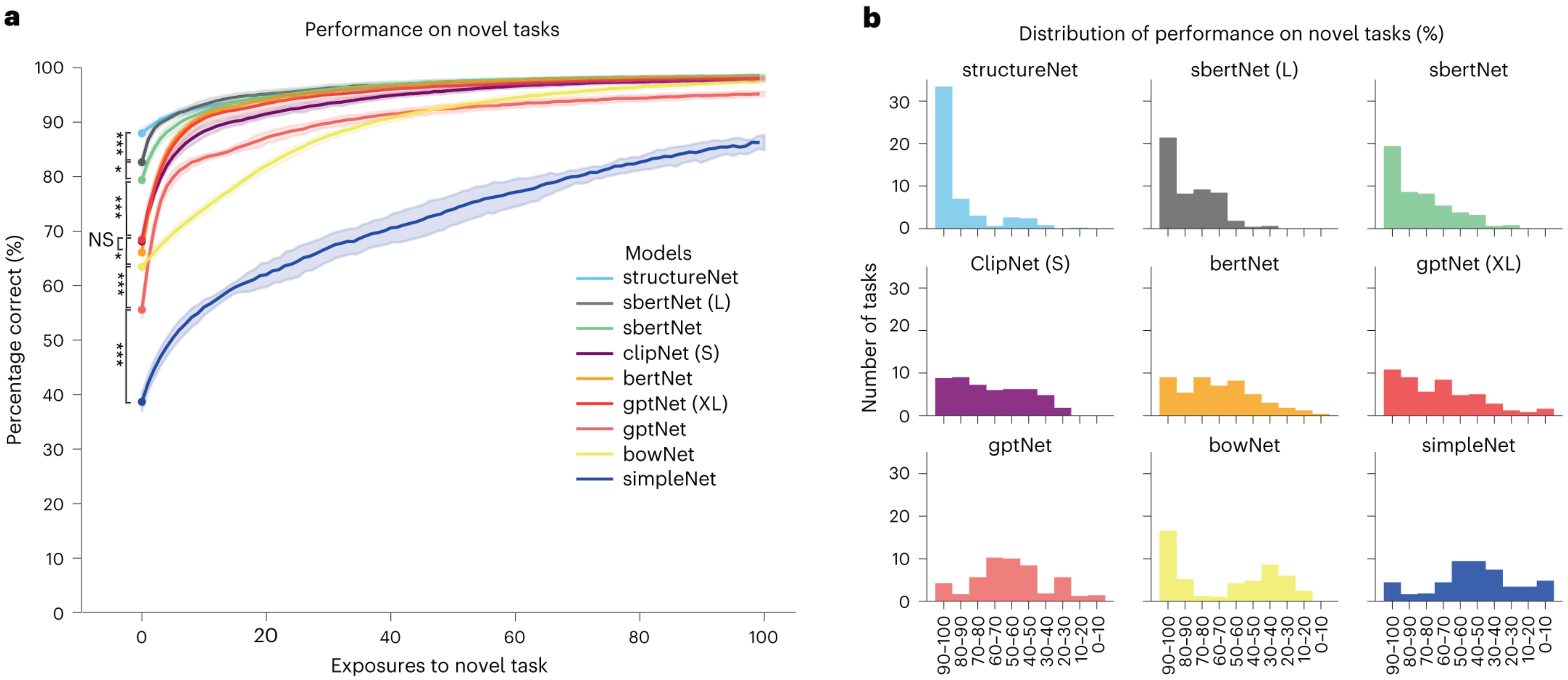
Nature Neuroscience (2024) | Cite this article

Recent example of a illustrating the power of natural language instructions
For (mostly) "cognitive tasks.



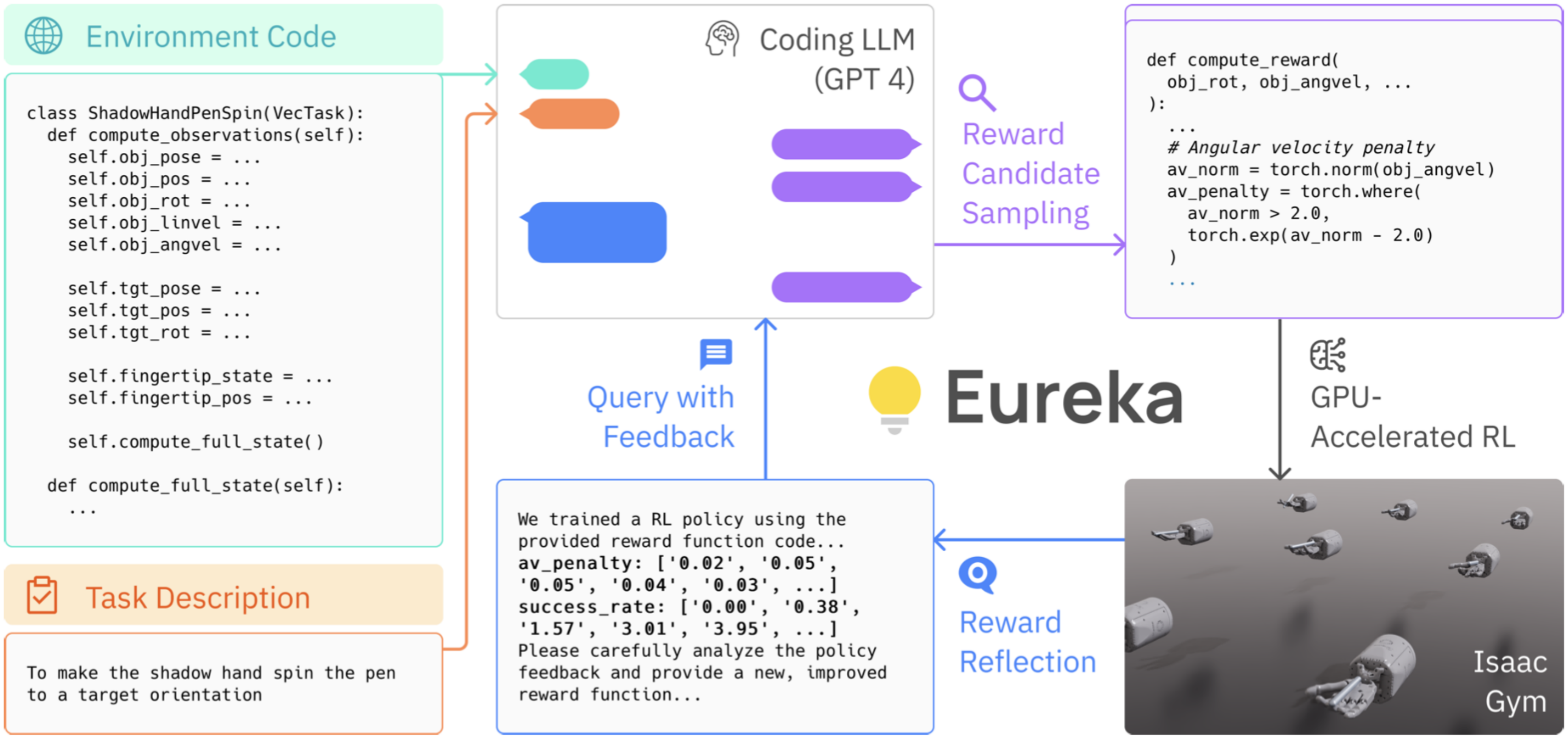
Caption: **a**,**b**, Illustrations of example trials as they might appear in a laboratory setting. The trial is instructed, then stimuli are presented with different angles and strengths of contrast. The agent must then respond with the proper angle during the response period. **a**, An example AntiDM trial where the agent must respond to the angle presented with the least intensity. **b**, An example COMP1 trial where the agent must respond to the first angle if it is presented with higher intensity than the second angle otherwise repress response. **c**, Diagram of model inputs and outputs. Sensory inputs (fixation unit, modality 1, modality 2) are shown in red and model outputs (fixation output, motor output) are shown in green. Models also receive a rule vector (blue) or the embedding that results from passing task instructions through a pretrained language model (gray). A list of models tested is provided in the inset.

Natural language instructions induce compositional generalization in networks of neurons



Riveland & Pouget, Nature Neuro 2024

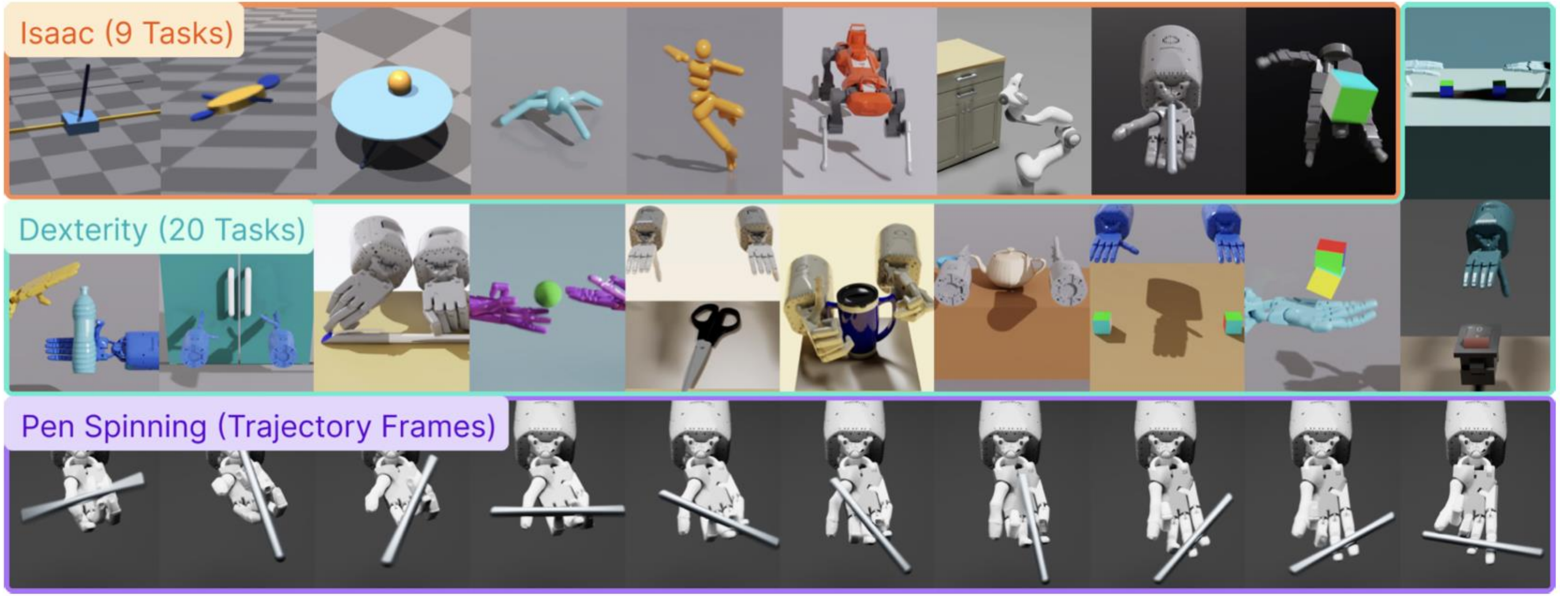
EUREKA: HUMAN-LEVEL REWARD DESIGN VIA CODING LARGE LANGUAGE MODELS



- https://eureka-research.github.io

Ma et al. ICLR 2024

EUREKA: HUMAN-LEVEL REWARD DESIGN VIA CODING LARGE LANGUAGE MODELS



- https://eureka-research.github.io

Ma et al. ICLR 2024

EUREKA: HUMAN-LEVEL REWARD DESIGN VIA CODING LARGE LANGUAGE MODELS

Algorithm 1 EUREKA

Require: Task description l, environment code M, coding LLM $\texttt{LLM}$, fitness function F, initial prompt $\texttt{prompt}$
Hyperparameters: search iteration N, iteration batch size K
for N iterations **do**
  `// Sample` K `reward code from LLM`
  $\texttt{R}_1, ..., \texttt{R}_k \sim \texttt{LLM}(l, M, \texttt{prompt})$
  `// Evaluate reward candidates`
  $s_1 = F(\texttt{R}_1), ..., s_K = F(\texttt{R}_K)$
  `// Reward reflection`
  $\texttt{prompt} := \texttt{prompt} : \texttt{Reflection}(R^n_{best}, s^n_{best})$,
  where $best = \arg\max_k s_1, ..., s_K$
  `// Update Eureka reward`
  $R_{\text{Eureka}}, s_{\text{Eureka}} = (R^n_{\text{best}}, s^n_{\text{best}}), \quad \text{if } s^n_{\text{best}} > s_{\text{Eureka}}$
Output: R_{Eureka}

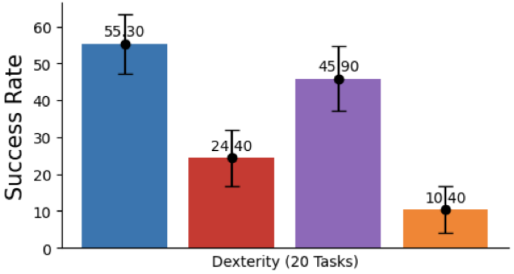
```
  def compute_reward(object_rot, goal_rot, object_angvel, object_pos, fingertip_pos):
      # Rotation reward
      rot_diff = torch.abs(torch.sum(object_rot * goal_rot, dim=1) - 1) / 2
-     rotation_reward_temp = 20.0
+     rotation_reward_temp = 30.0                                        Changing hyperparameter
      rotation_reward = torch.exp(-rotation_reward_temp * rot_diff)

      # Distance reward
+     min_distance_temp = 10.0
      min_distance = torch.min(torch.norm(fingertip_pos - object_pos[:, None], dim=2), dim=1).values
-     distance_reward = min_distance
+     uncapped_distance_reward = torch.exp(-min_distance_temp * min_distance)
+     distance_reward = torch.clamp(uncapped_distance_reward, 0.0, 1.0)   Changing functional form

-     total_reward = rotation_reward + distance_reward
+     # Angular velocity penalty                                          Adding new component
+     angvel_norm = torch.norm(object_angvel, dim=1)
+     angvel_threshold = 0.5
+     angvel_penalty_temp = 5.0
+     angular_velocity_penalty = torch.where(angvel_norm > angvel_threshold,
+         torch.exp(-angvel_penalty_temp * (angvel_norm - angvel_threshold)), torch.zeros_like(angvel_norm))
+
+     total_reward = 0.5 * rotation_reward + 0.3 * distance_reward - 0.2 * angular_velocity_penalty

      reward_components = {
          "rotation_reward": rotation_reward,
          "distance_reward": distance_reward,
+         "angular_velocity_penalty": angular_velocity_penalty,
      }

      return total_reward, reward_components
```

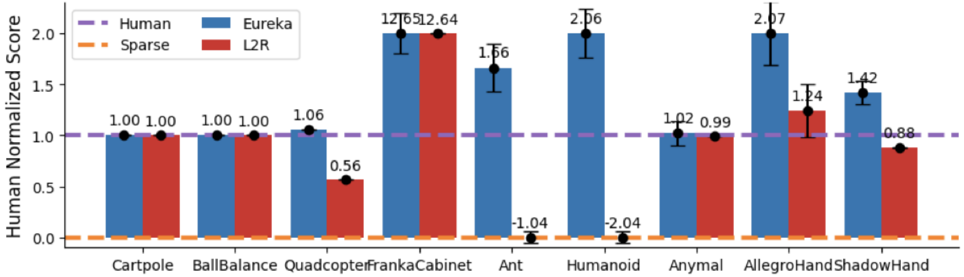
Figure 3: EUREKA can zero-shot generate executable rewards and then flexibly improve them with many distinct types of free-form modification, such as (1) changing the hyperparameter of existing reward components, (2) changing the functional form of existing reward components, and (3) introducing new reward components.

- https://eureka-research.github.io

Ma et al. ICLR 2024

EUREKA: HUMAN-LEVEL REWARD DESIGN VIA CODING LARGE LANGUAGE MODELS





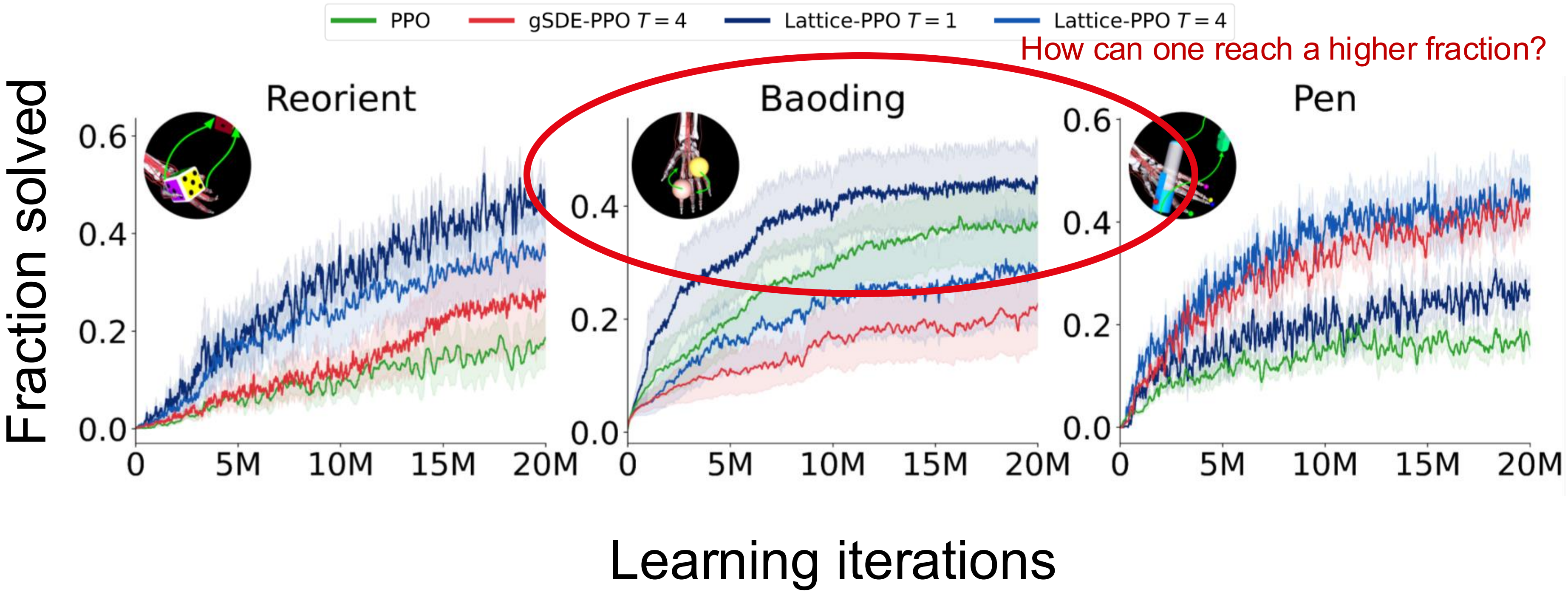
Human: expert written reward functions

- https://eureka-research.github.io

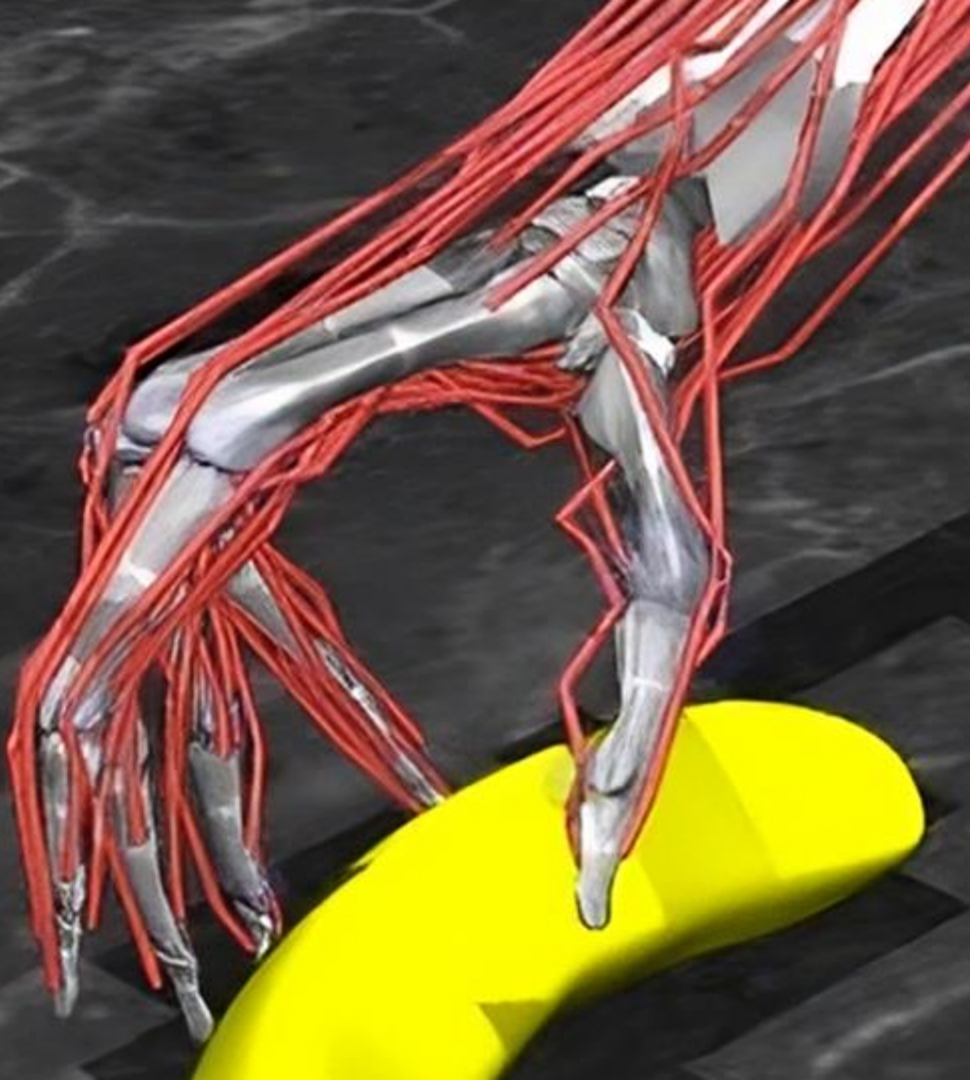
Ma et al. ICLR 2024

Check out some videos: https://eureka-research.github.io

Reminder: Object manipulation learning curves



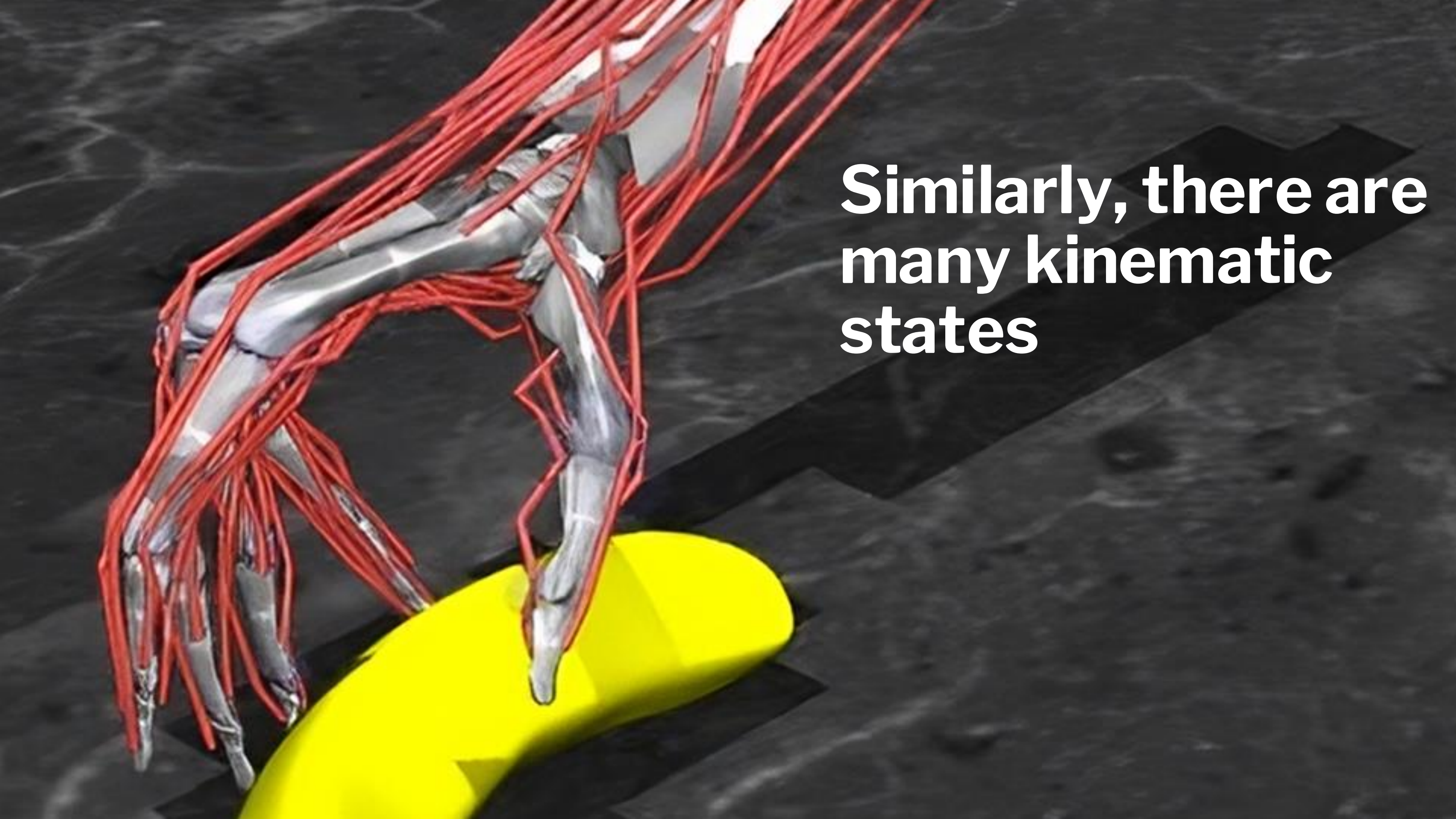
Chiappa, Marin Vargas, Huang, Mathis, NeurIPS 2023



How many muscle states are there?

q^600

(for 600 muscles assuming q states per muscle)

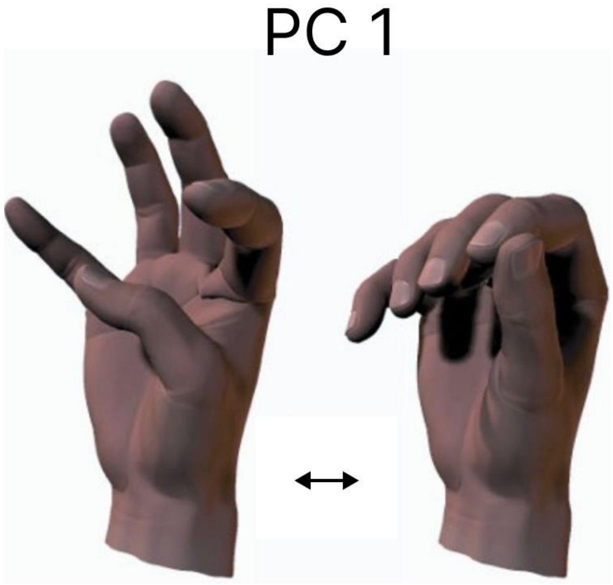


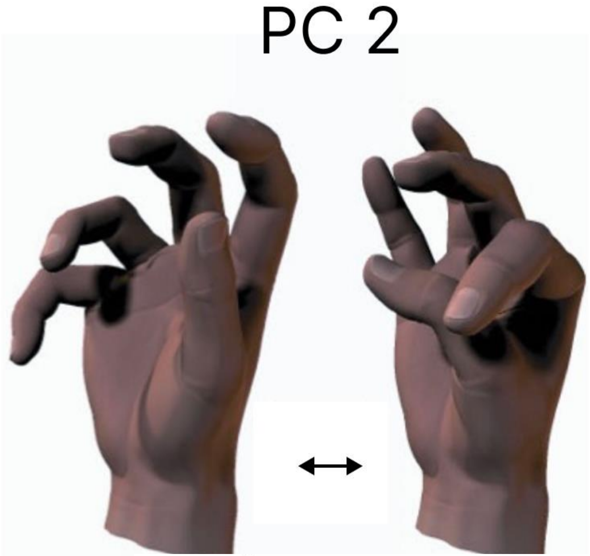
Similarly, there are many kinematic states

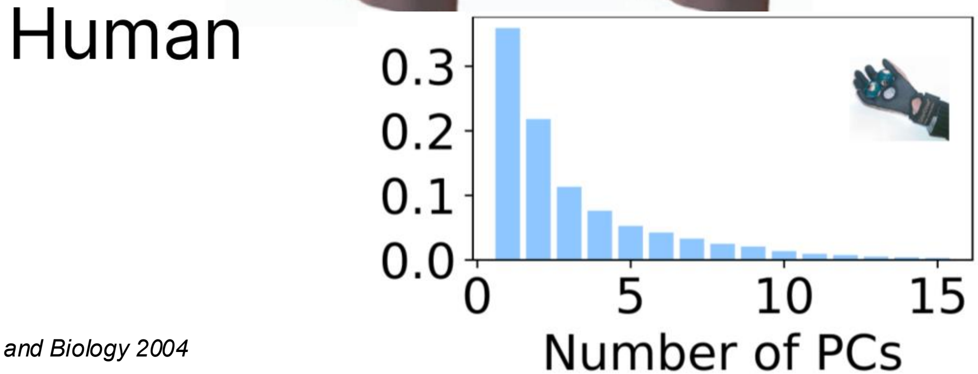
Boading balls: an example skill

Source: reddit

How do humans control the hand?







Classic result: Bernstein, Bizzi, D'Avella, …

- *Todorov, & Ghahramani,*
  Annual International Conference of the IEEE Engineering in Medicine and Biology 2004

What about other object-manipulation tasks?

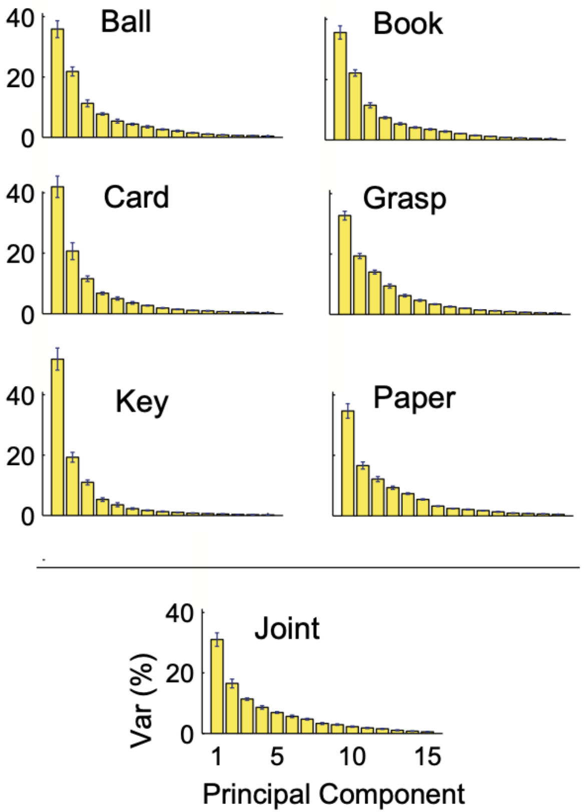
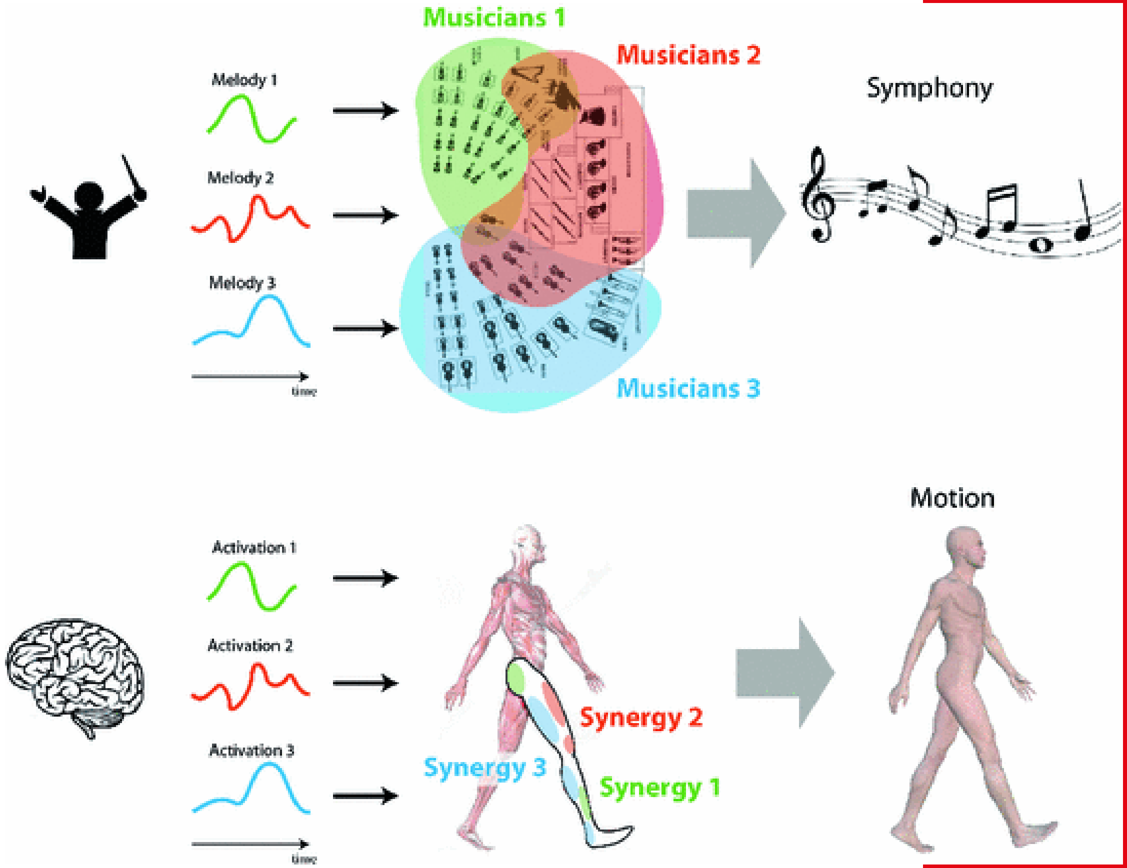


Figure 2. Variance accounted for by the first 15 principal components in each task. Results are averaged over subjects.

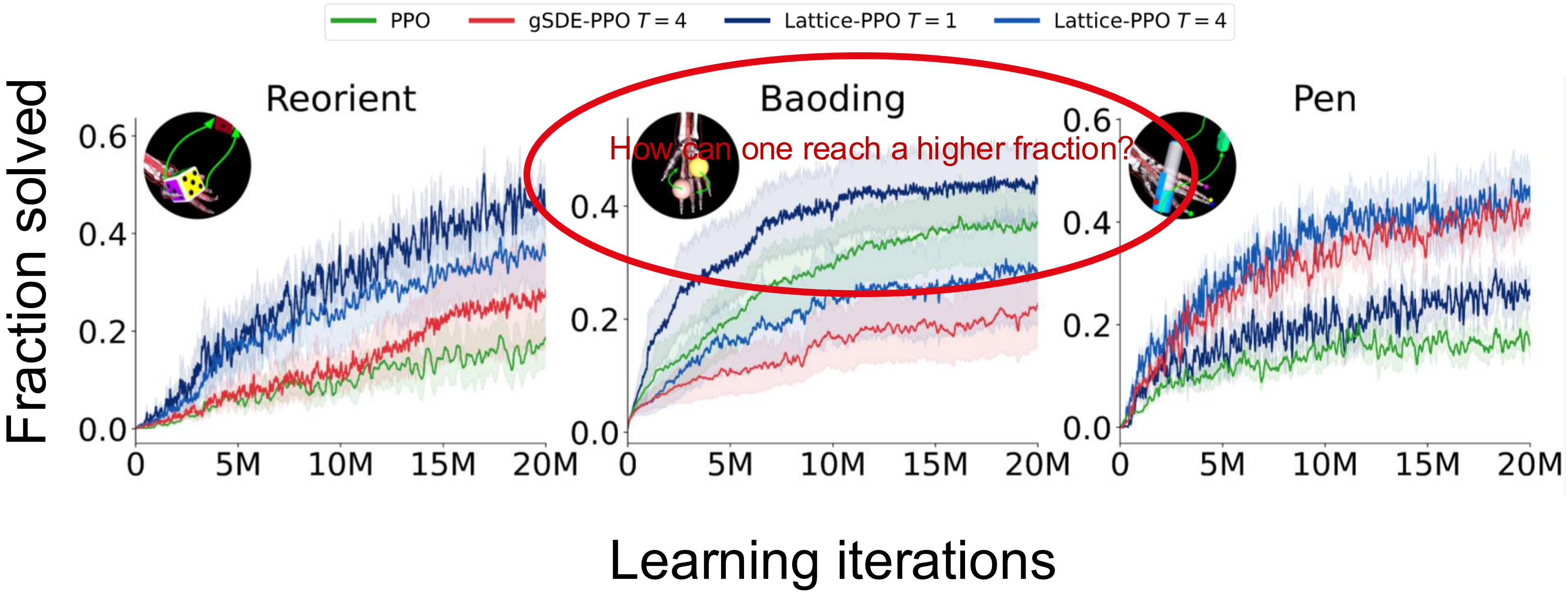
- *Todorov, & Ghahramani,*
  Annual International Conference of the IEEE Engineering in Medicine and Biology 2004

Reminder: Muscle synergies as principle for motor control



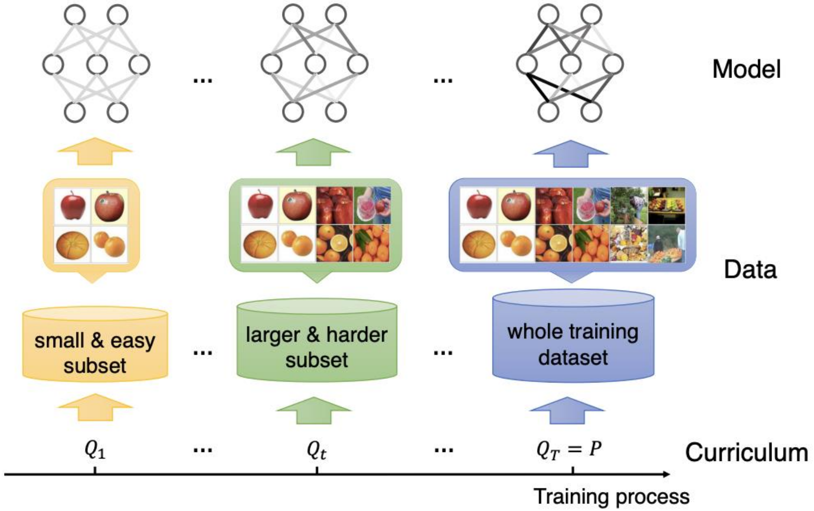
Torricelli et al., Emerging Therapies in Neurorehabilitation 2015

Reminder: Object manipulation learning curves

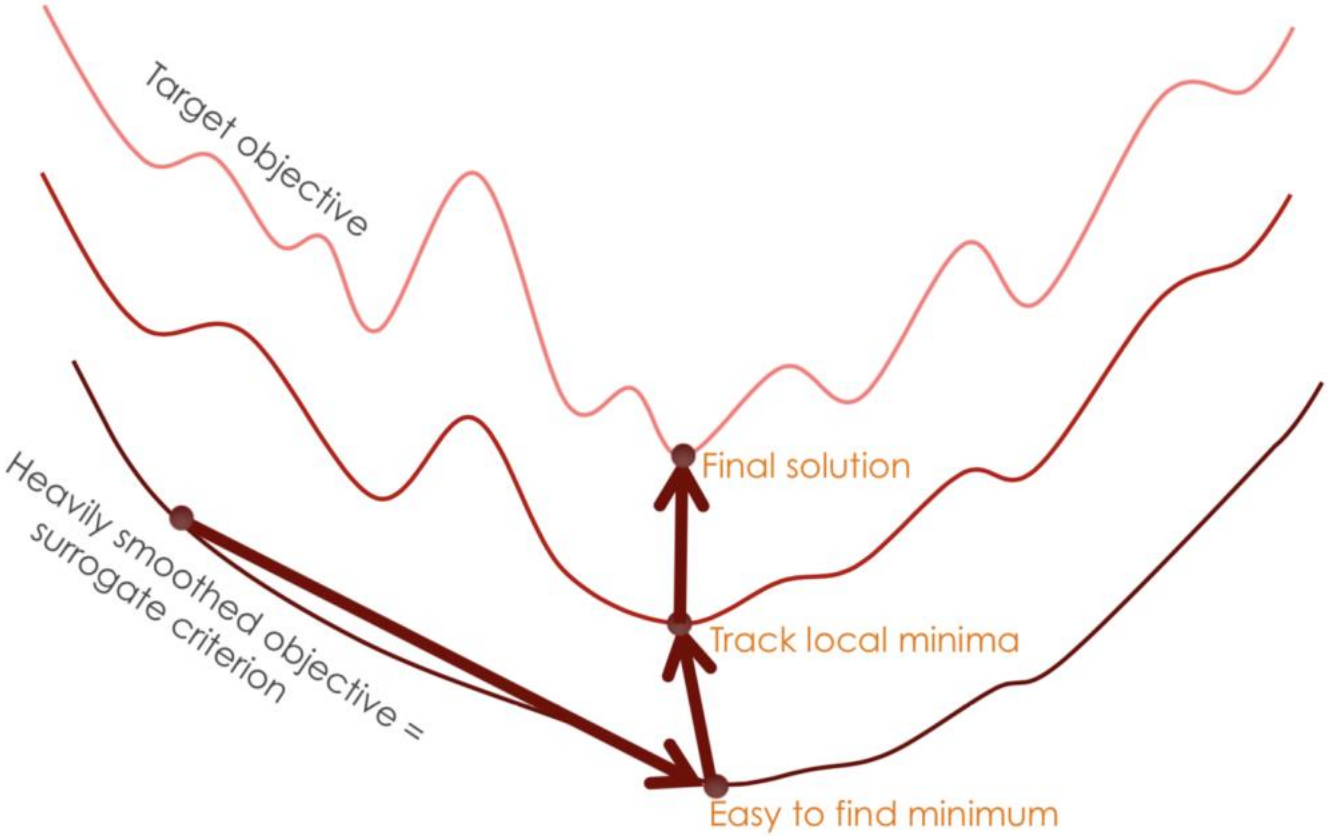


Chiappa, Marin Vargas, Huang, Mathis, NeurIPS 2023

Curriculum learning



Wang, Chen, Zhu IEEE TPAMI, 2021

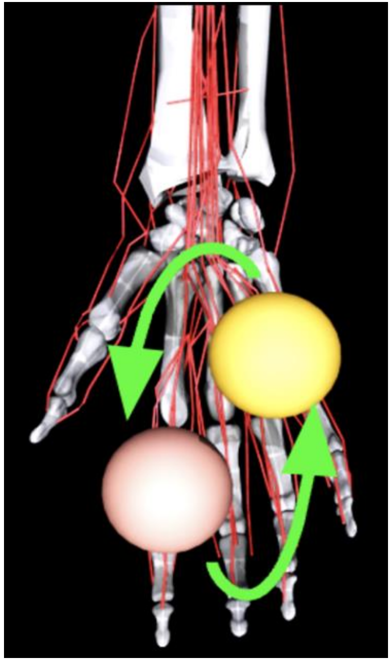


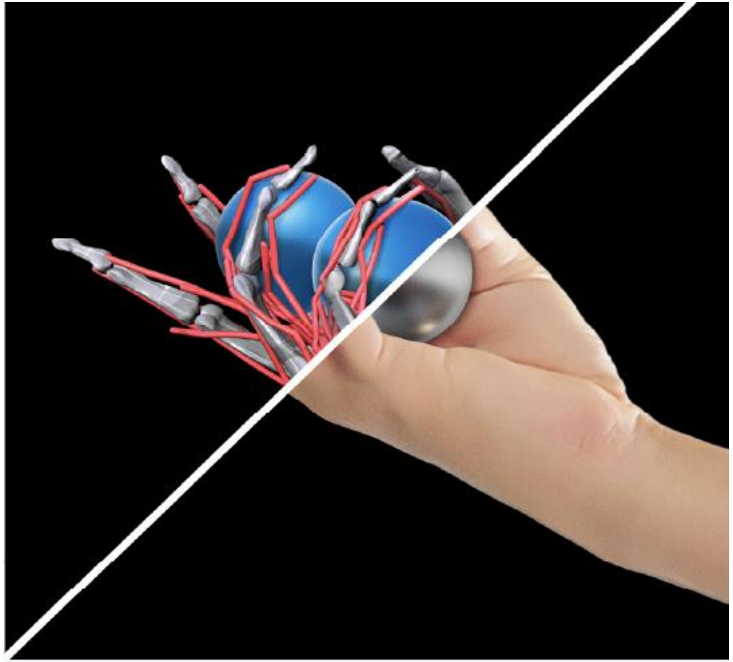
Wang, Chen, Zhu IEEE TPAMI, 2021

MyoChallenge: Baoding Balls

Inaugural NeurIPS Challenge 2022

q^39





SOTA RL:

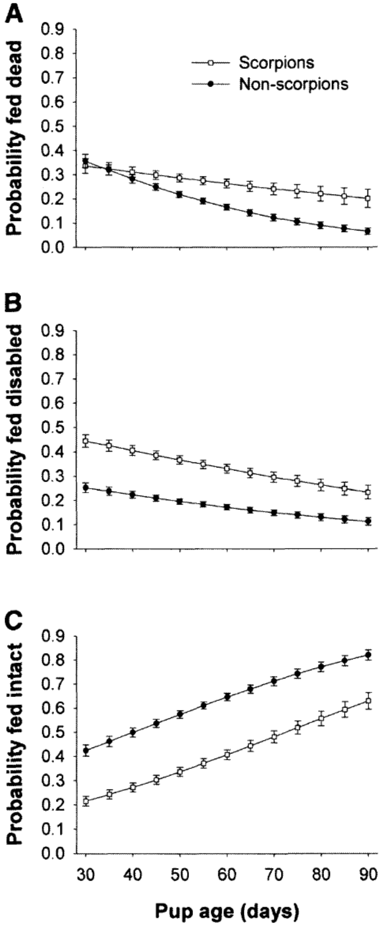
Phase 1	Phase 2
41%	0%

Caggiano et al. *Proceedings of the NeurIPS 2022 Competitions Track*, PMLR 220:233-250

Curriculum learning in biology

Teaching in Wild Meerkats

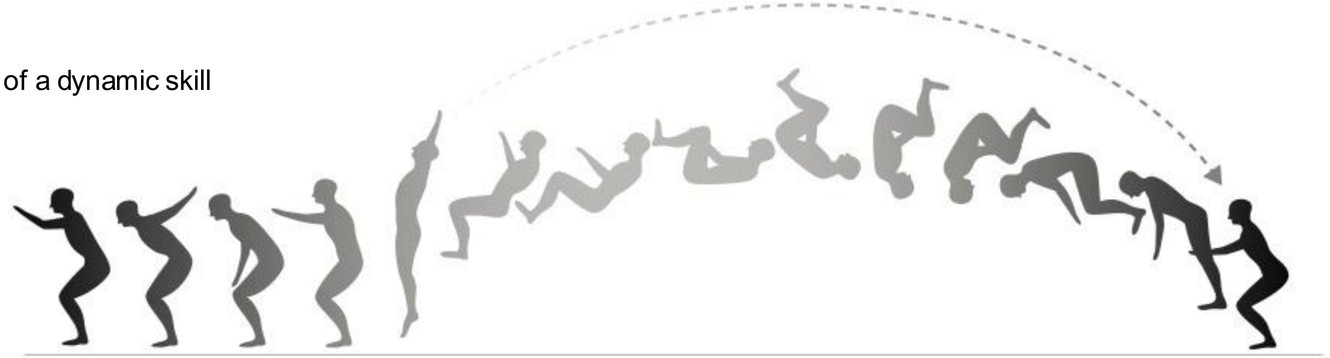
Alex Thornton* and Katherine McAuliffe



Science, 2006

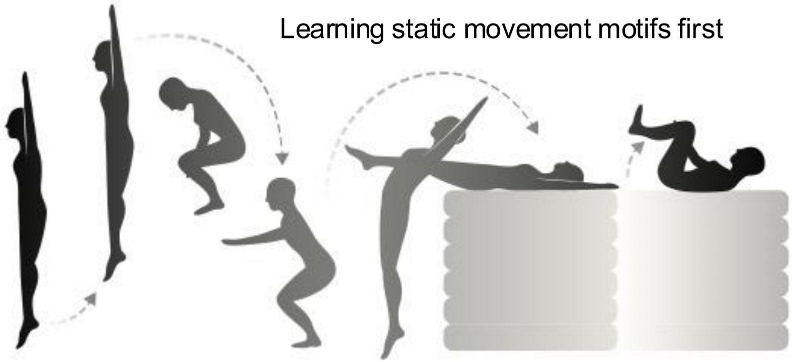
Inspiration from coaching: part-to-whole practice

States of a dynamic skill



Learning static movement motifs first

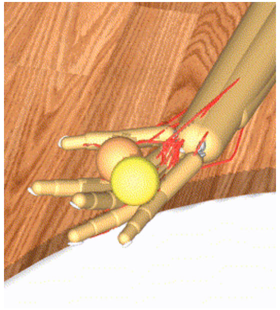
Recommended strategy:

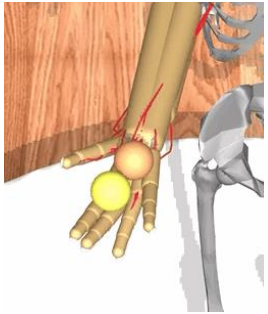


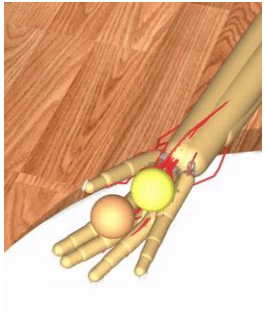
Chiappa, A. S., Tano*, P., Patel*, N., Ingster, A., Pouget, A., & Mathis, A. Neuron 2024*

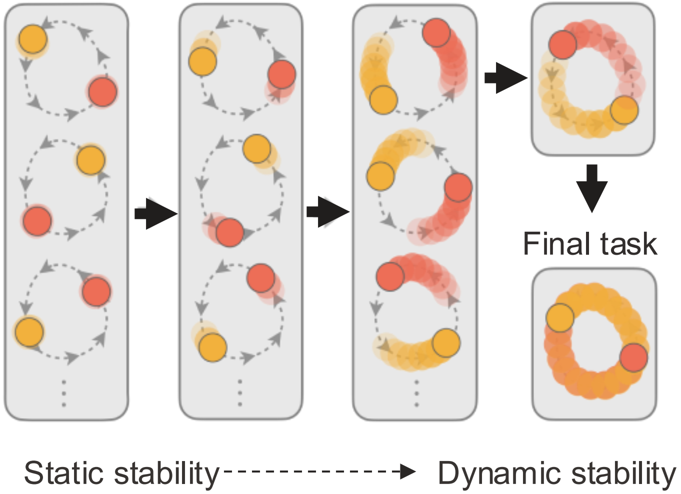
Curriculum learning

- Static to Dynamic Stability (SDS)
  - SDS creates stability at desired states *before* learning a policy that reaches them
  - A curriculum gradually transforms static stability into dynamic movement motifs





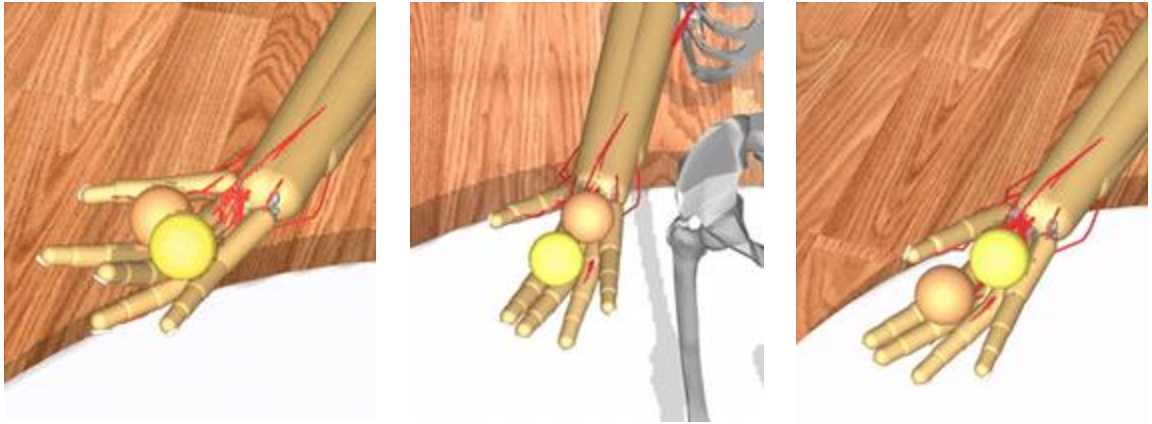




- *Chiappa*, A. S., Tano*, P., Patel*, N., Ingster, A., Pouget, A., & Mathis, A. Neuron 2024*
  Caggiano et al. Proceedings of the NeurIPS 2022 Competitions Track, PMLR 220:233-250

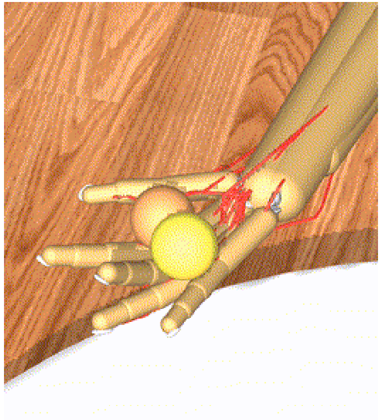
Curriculum learning

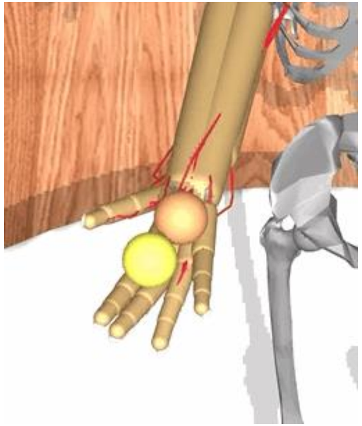
- Going *gradually* from static to dynamic stability allows to learn a complex motor skill (100% performance) for Phase I.

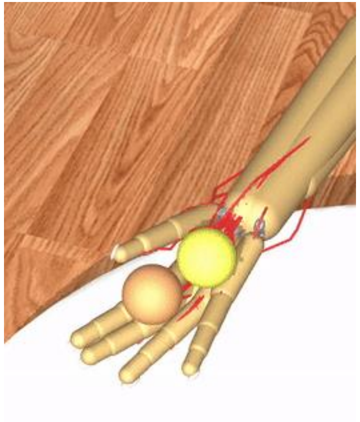


- Learning without a curriculum fails (41% performance)
- Going *directly* from static stability to the final task fails (42% performance)
- Standard speed curriculum fails (45% performance)

EPFL







Curriculum	Phase 1	Phase 2
None	41%	0%
Location only	42%	4%
Speed only	45%	0%
SDS (ours)	100%	55%

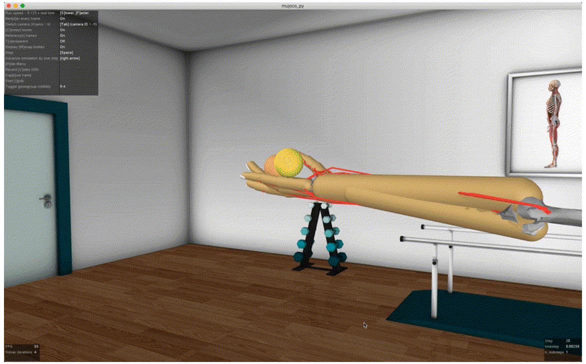
Team	Performance
SDS (ours)	55%
AI4Muscles	41%
IARAI-JKU	15%
pkumarl	14%

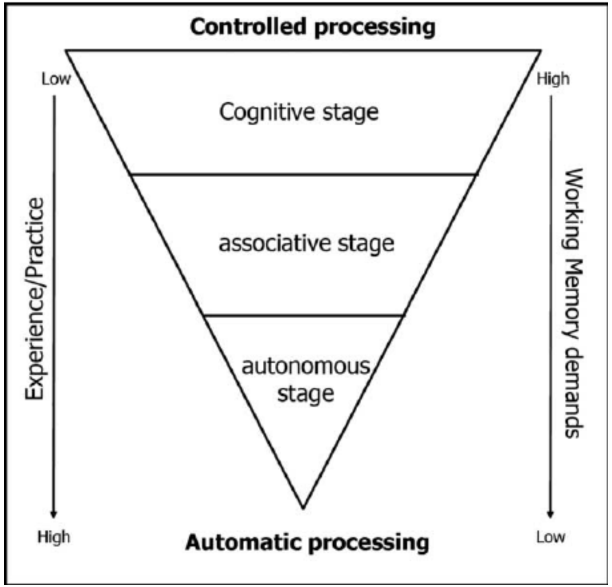
Chiappa, A. S., Tano*, P., Patel*, N., Ingster, A., Pouget, A., & Mathis, A. Neuron 2024*

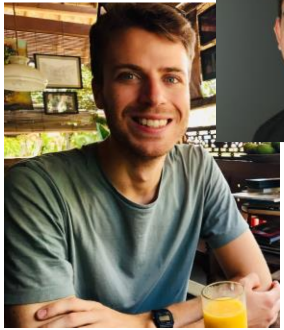
Caggiano et al. Proceedings of the NeurIPS 2022 Competitions Track, PMLR 220:233-250

Fitts Posner's 1967 model of skill learning

Policies can get trapped in local minima





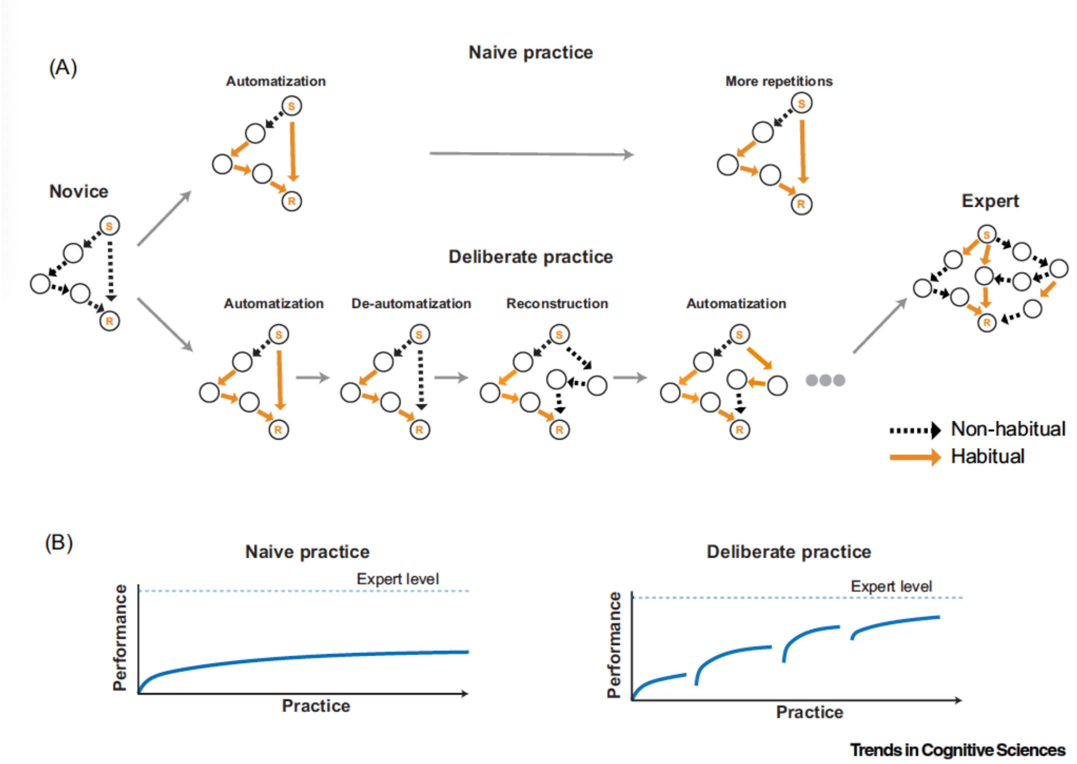


Alberto Chiappa
Nisheet Patel
Pablo Tano

PPO, …

Furley & Memmert, International Review of Sport and Exercise Psychology 2010

Deliberate practice

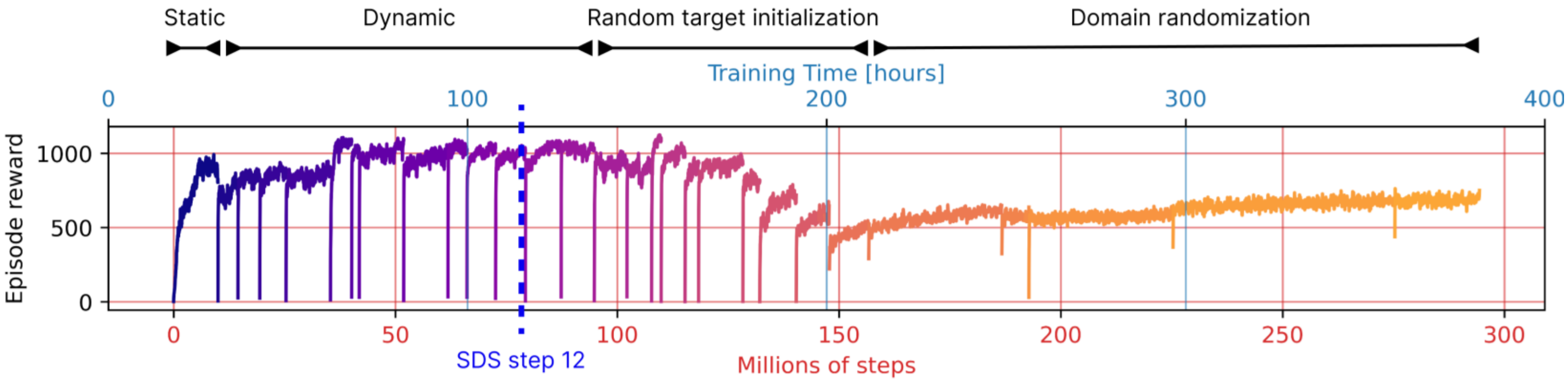


Du, Krakauer, Haith Trends in Cog Sci 2022

Learning curve for our policy

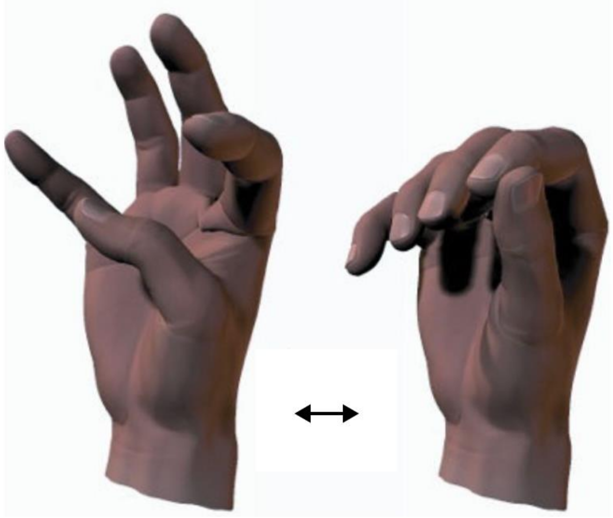
Sport science terminology: Part to whole practice Deliberate practice

ML terminology:

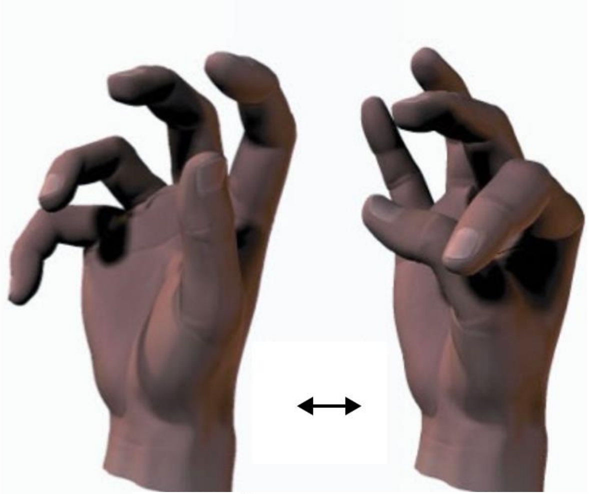


How do humans achieve this task?

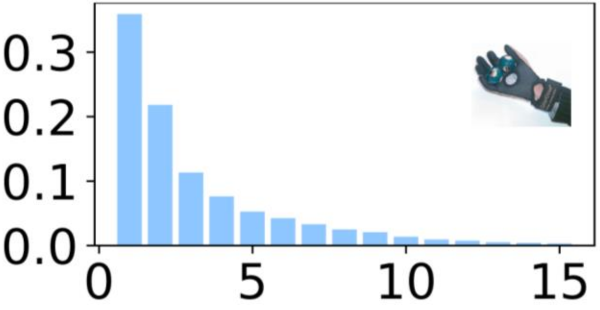
PC 1



PC 2



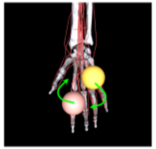
Human



Todorov, & Ghahramani,
Annual International Conference of the IEEE Engineering in Medicine and Biology 2004

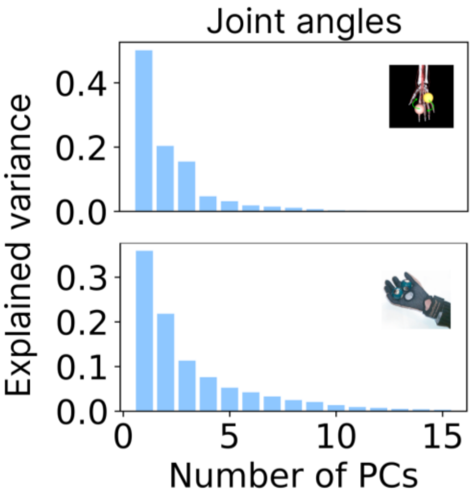
SDS also discovers a low-dimensional control space

SDS



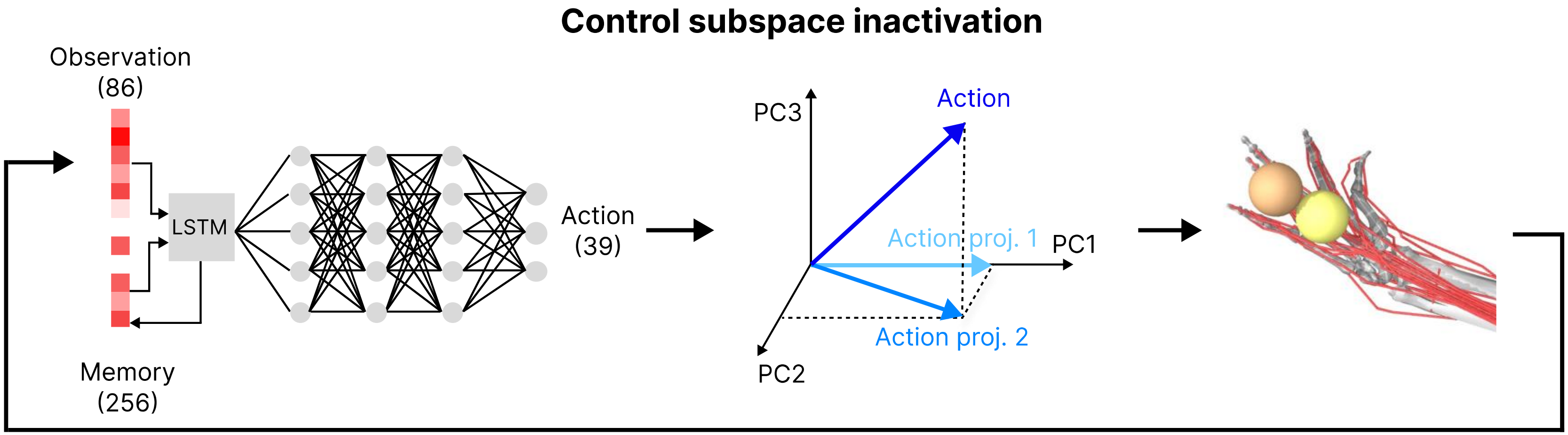
Human

		Position	Muscle act.
SDS	Baoding	4.5	12
	Control	8	7
Human	Baoding	5	
	Control	8.5	

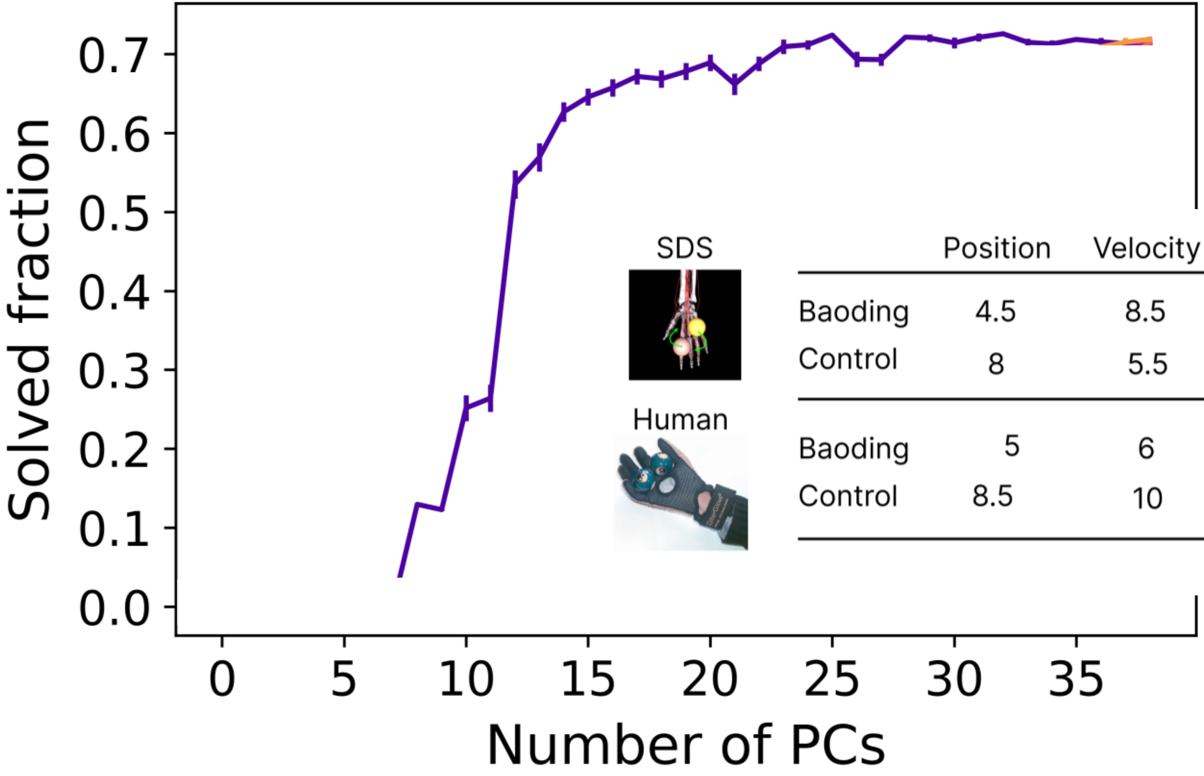


This notion of muscle/kinematic synergy is purely based on reconstruction error!

Physics engine allows causal experiments with “muscle synergies”



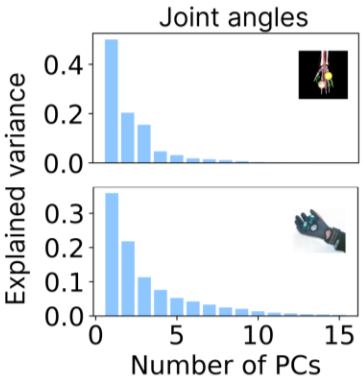
Signal reconstruction underestimates necessary DoF



SDS

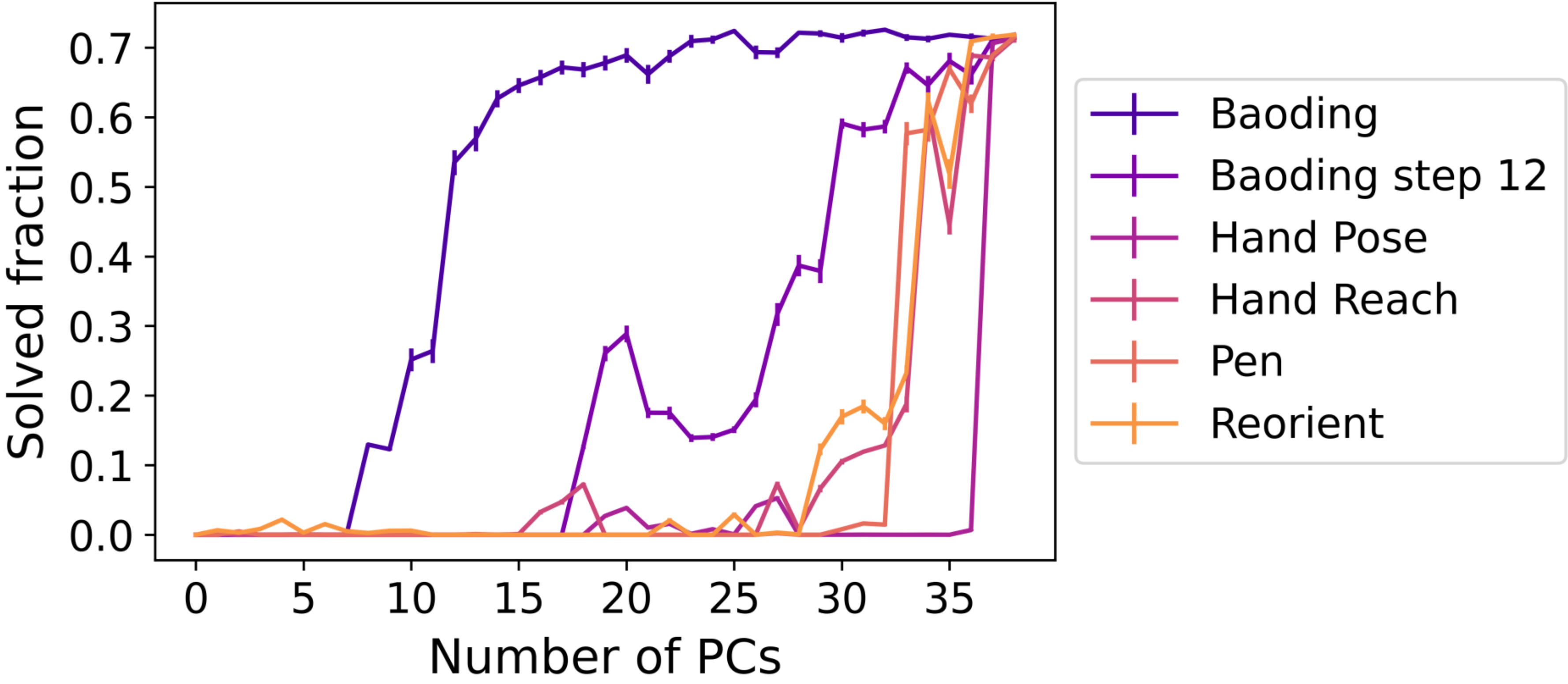
Human

		Position	Velocity	Muscle act.
SDS	Baoding	4.5	8.5	12
	Control	8	5.5	7
Human	Baoding	5	6	
	Control	8.5	10	



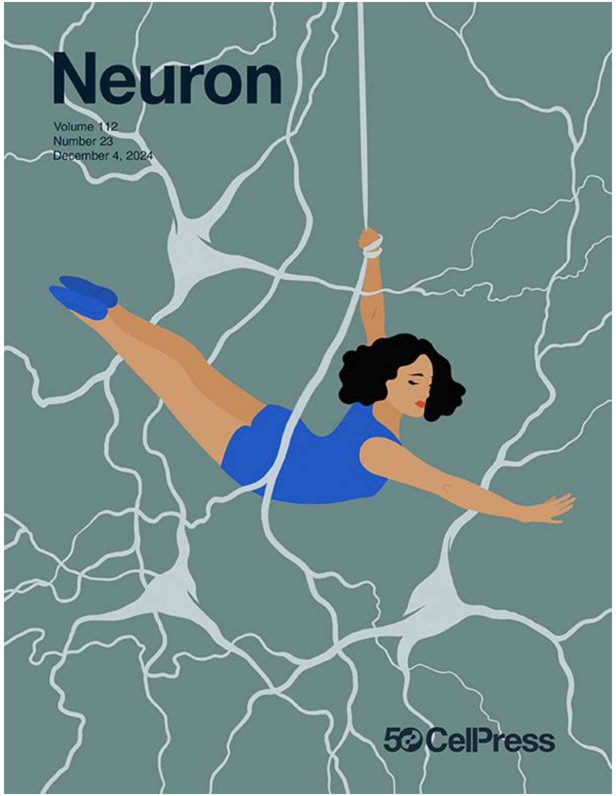

Late Lattice training performance
Chiappa et al. NeurIPS 2023
Run speed = 1.000 x real time [S]lower, [F]aster
Ren[d]er every frame On
Switch camera (#cams = 6) [Tab] (camera ID = -1)
[C]ontact forces On
Referenc[e] frames On
T[r]ansparent Off
Display [M]ocap bodies On
Stop [Space]
Advance simulation by one step [right arrow]
[H]ide Menu
Record [V]ideo (Off)
Cap[t]ure frame
Start [i]pdb
Toggle geomgroup visibility 0-4
FPS 267
Solver iterations 2
Reorient task
In MyoSuite/Mujoco
Step 530
timestep 0.00200
n_substeps 1

Control spaces are highly task-dependent & transfer poorly



Insights from analyzing SDS

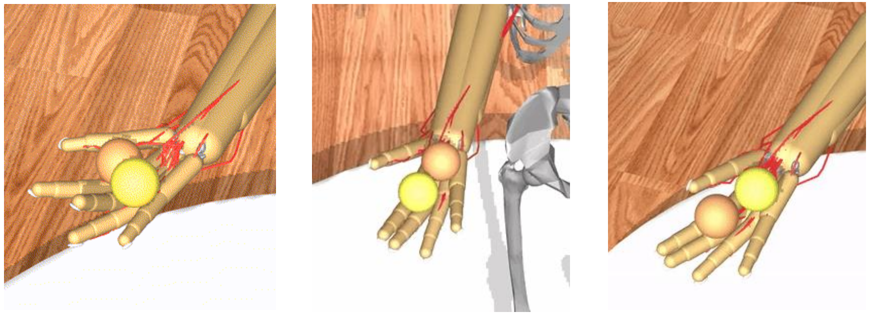
- Muscle synergies have been proposed as a key principle for motor control
- Yet, low-dimensional nature might be **underestimated with existing techniques!**
- For the hand -- learned muscle synergies are highly task-specific, and thus generalize poorly
- This suggests that low-dimensional control is an emergent property (of the task/biomechanics/distributed circuits) rather than the mechanism of control (not a simplifying strategy)
- Neural networks are ideal for taming complex biomechanics



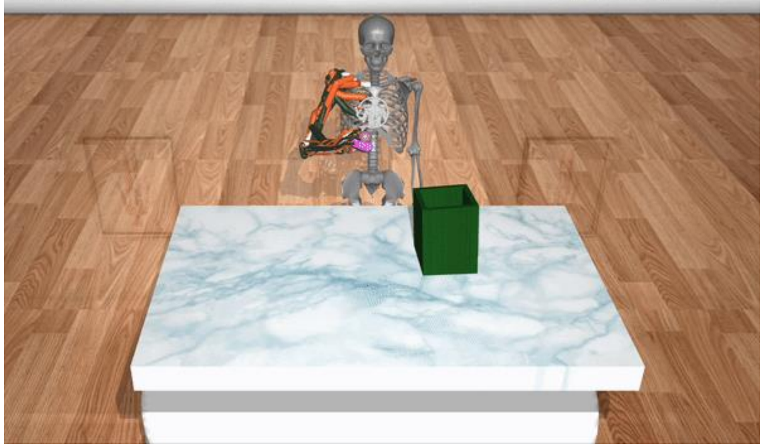
Curriculum learning

All the best solutions in the NeurIPS challenge are based on curriculum learning…

Winning solution of NeurIPS 2022 challenge



Winning solution of NeurIPS 2023 challenge



Marin Vargas, A., Chiappa, A. S., & Mathis, A

Chiappa, A. S., Tano*, P., Patel*, N., Ingster, A., Pouget, A., & Mathis, A. bioRxiv*

Caggiano et al. Proceedings of the NeurIPS 2022 Competitions Track, PMLR 220:233-250

Solution in part based on: Lattice and Curriculum Learning:
Chiappa, Marin Vargas, Huang, Mathis *NeurIPS. 2023*

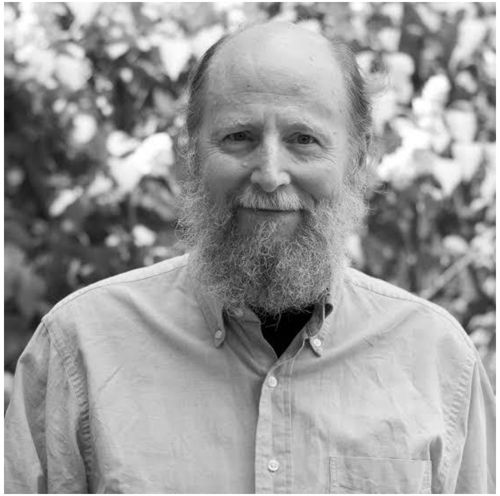
What is missing?

- Internal models
- Inductive biases (innate architecture)
- Better exploration
- Baked in reward functions (which we don’t know…)
- *Using language*
- *Curriculum learning (automatic curriculum discovery?)*
- *Deliberate practice*
- ….

While we do not know their contributions or even the necessity of either one of those claims, I will show preliminary evidence for each to give you an idea.

There is a lot of research to be done to close this gap & figure out what actually matters…

A counter point – the bitter lesson


Richard Sutton

"The bitter lesson is based on the historical observations that 1) AI researchers have often tried to build knowledge into their agents, 2) this always helps in the short term, and is personally satisfying to the researcher, but 3) in the long run it plateaus and even inhibits further progress, and 4) breakthrough progress eventually arrives by an opposing approach based on scaling computation by search and learning. *The eventual success is tinged with bitterness, and often incompletely digested, because it is success over a favored, human-centric approach."*

http://www.incompleteideas.net/IncIdeas/BitterLesson.html

Take-home messages

What might explain the gap between biological and artificial control?

- Internal models
- Inductive biases (innate architecture)
- Better exploration
- Baked in reward functions (which we don’t know…)
- *Using language*
- *Curriculum learning*
- *Deliberate practice*
- ….
- *But beware of the bitter lesson!*