Trade-Offs in Robot Skill Learning

Jan Peters

Technische Universität Darmstadt

Max Planck Institute for Intelligent Systems

TECHNISCHE UNIVERSITÄT DARMSTADT

Motivation

Can we create such robots?

Motivation

Uncertainty in tasks
and environment

Adapt to humans

Programming complexity
beyond human imagination

How can we fulfill Hollywood's vision of future robots?

- Smart Humans? Hand-coding of behaviors has allowed us to go *very far*!
- Maybe we should allow the robot to learn new tricks, adapt to situations, refine skills?
- Why have "Off-the-shelf" machine learning approaches not delivered the need intelligence?

➡ We trade-off between autonomy, human insight and behavior quality to get close!

Trade-Offs in Robot Skill Learning

Behavior Quality

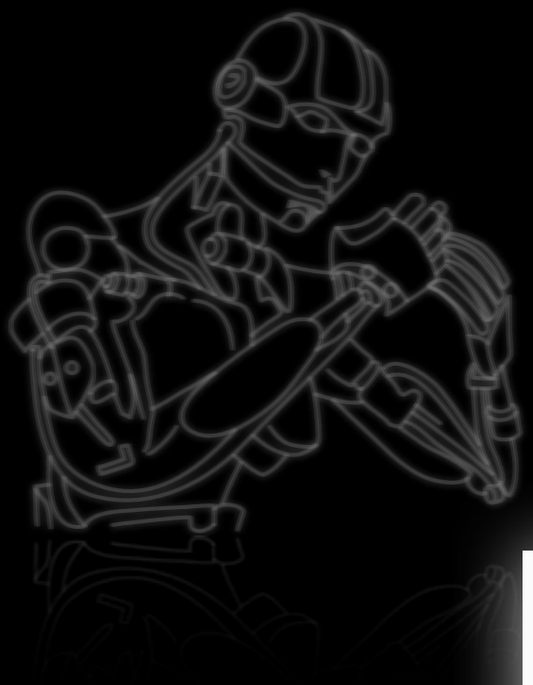
Efficiency of Generation

Trade-Offs in Robot Learning

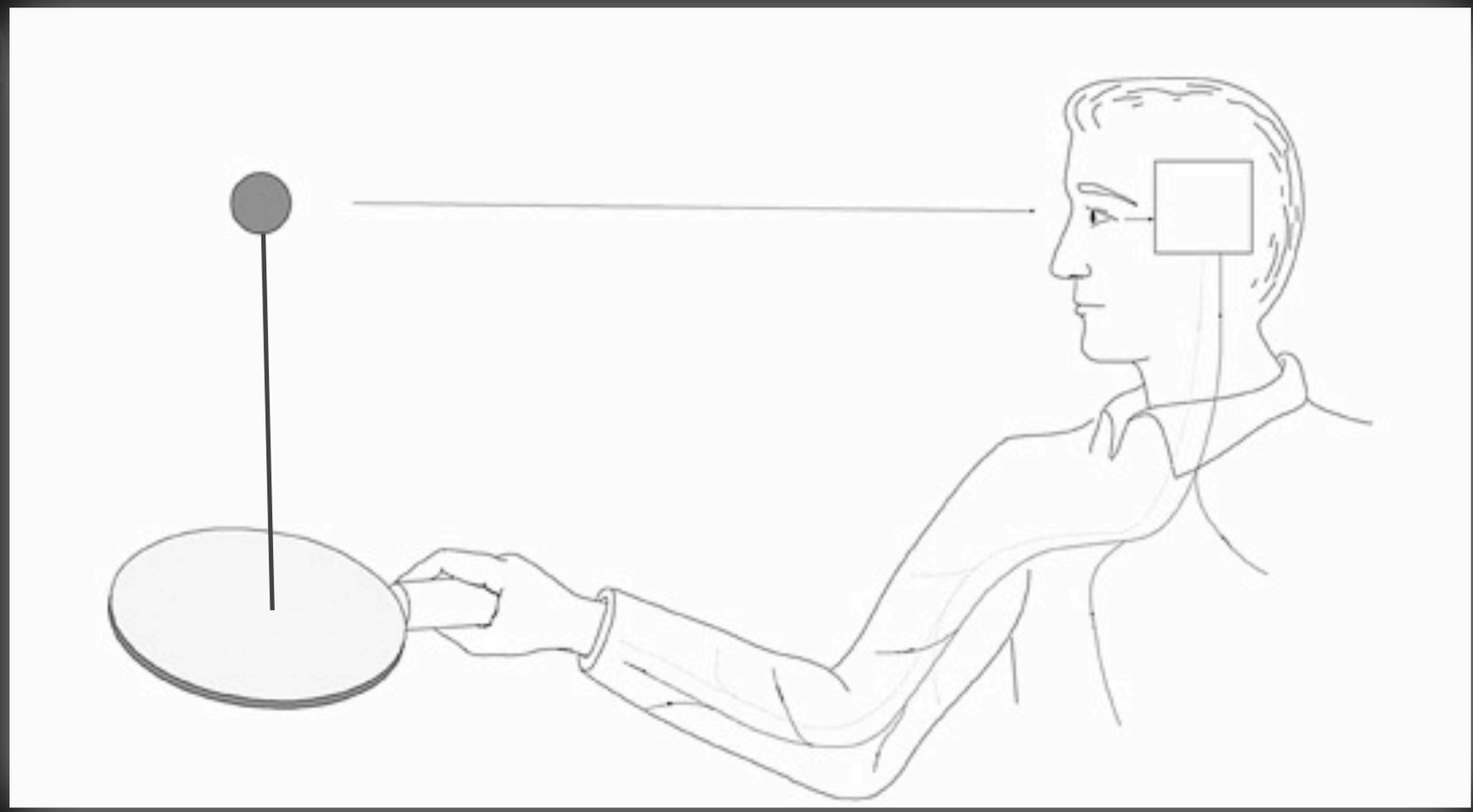
Autonomy & Generality

Outline

1. Introduction: Robot Skill Learning in Four Slides
2. Trade-Off I: Generality vs Physical Knowledge
3. Trade-Off II: Information Loss vs Self-Improvement
4. Trade-Off III: Large Repertoire vs Parsimonious Behavior Representation
5. Conclusion

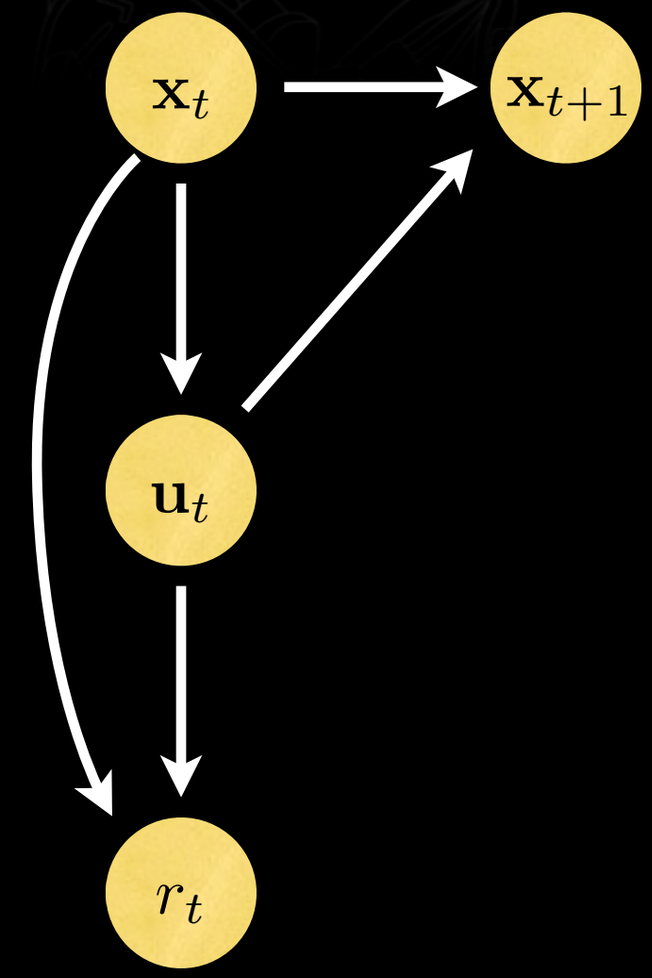


Example



Internal and external state $\mathbf{x}_t$, action $\mathbf{u}_t$.

Modeling Assumptions



Policy: Generates action $\mathbf{u}_t$ in state $\mathbf{x}_t$.

Should we use a deterministic policy $\mathbf{u}_t = \pi(\mathbf{x}_t)$?

NO! *Stochasticity* is important:

- needed for exploration
- breaks "curse of dimensionality"
- optimal solution can be stochastic

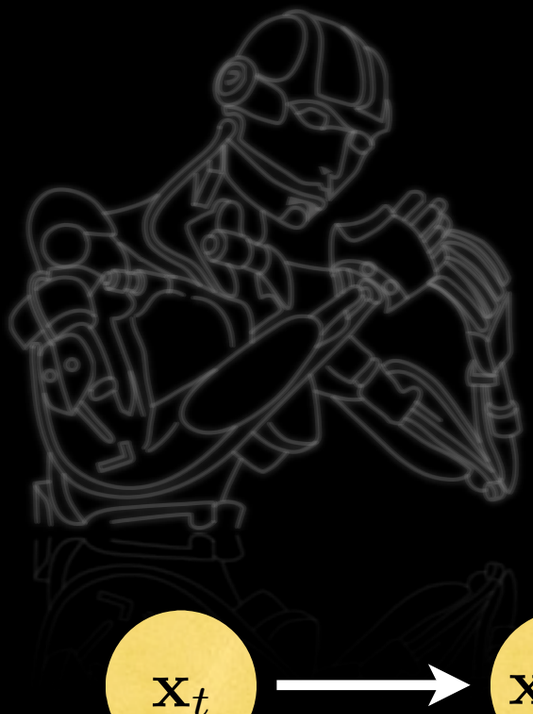
Robot learning implies "policy optimization"!

Hence, we use a stochastic policy: $\mathbf{u}_t \sim \pi(\mathbf{u}_t|\mathbf{x}_t)$

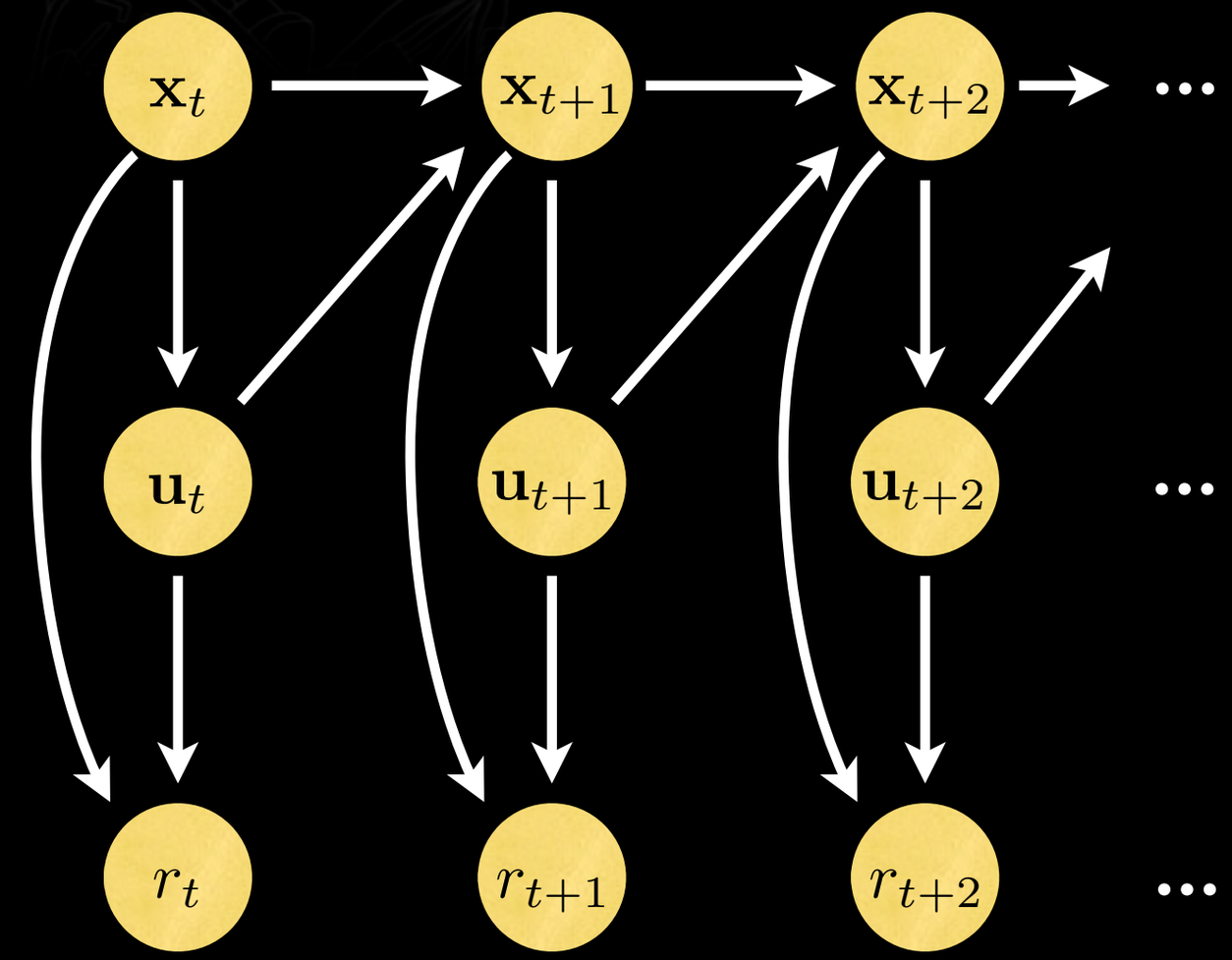
Teacher: Evaluates the performance and rates it with r_t.

Environment: An action $\mathbf{u}_t$ causes the system to change state from $\mathbf{x}_t$ to $\mathbf{x}_{t+1}$.

Model in the real world: $\mathbf{x}_{t+1} \sim p(\mathbf{x}_{t+1}|\mathbf{x}_t, \mathbf{u}_t)$



Let the loop roll out!



Trajectories

$$\boldsymbol{\tau} = [\mathbf{x}_0, \mathbf{u}_0, \mathbf{x}_1, \mathbf{u}_1 \ldots, \mathbf{x}_{T-1}, \mathbf{u}_{T-1}, \mathbf{x}_T]$$

Path distributions

$$p(\tau) = p(\mathbf{x}_0) \prod_{t=0}^{T-1} p(\mathbf{x}_{t+1}|\mathbf{x}_t, \mathbf{u}_t)\pi(\mathbf{u}_t|\mathbf{x}_t)$$

Path rewards:

$$r(\boldsymbol{\tau}) = \sum_{t=0}^{T} \alpha_t r(\mathbf{x}_t, \mathbf{u}_t)$$

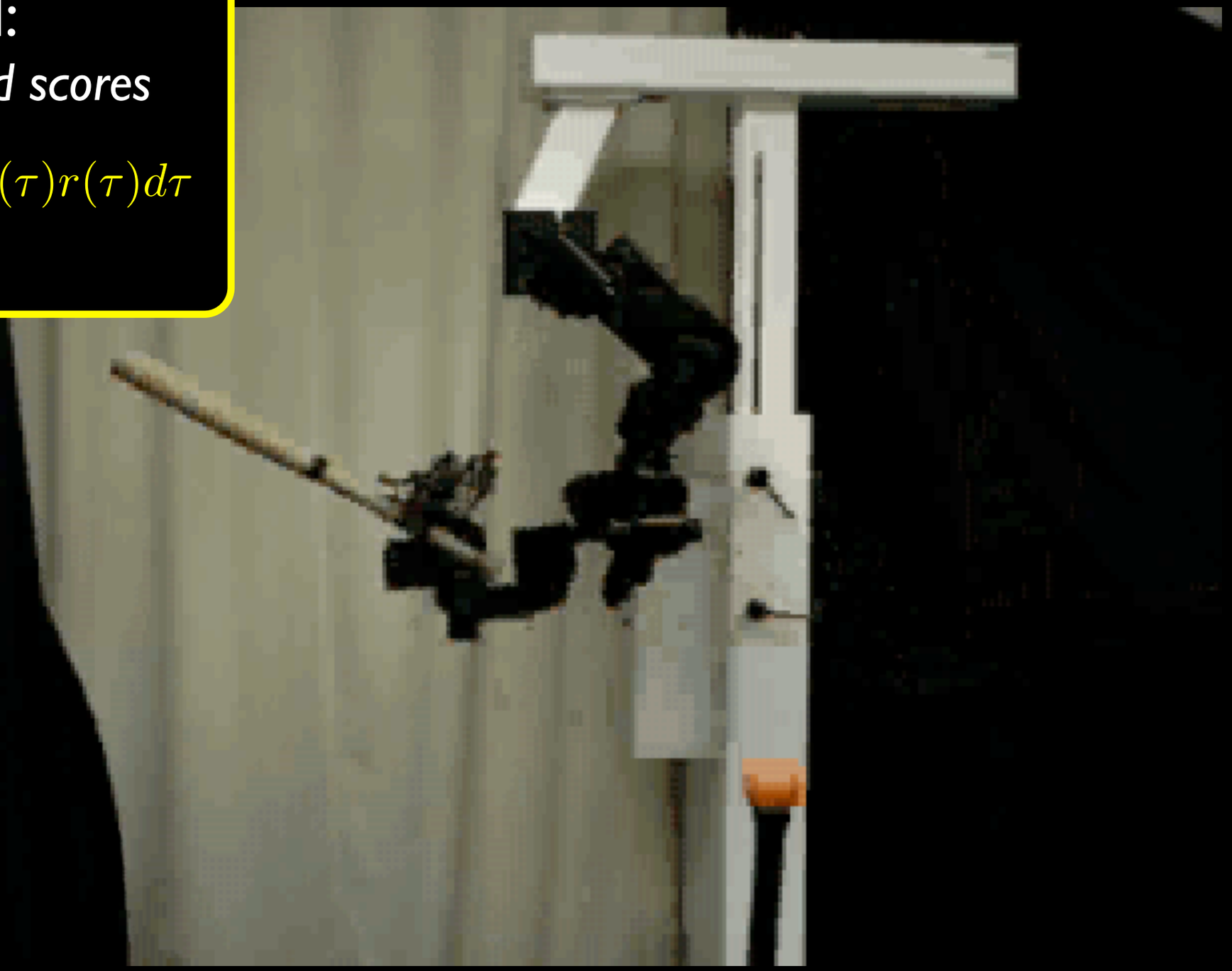
What is robot learning?

In our model:

Optimize the *expected scores*

$$J(\theta) = E_\tau\{r(\tau)\} = \int_{\mathbb{T}} p_\theta(\tau) r(\tau) d\tau$$

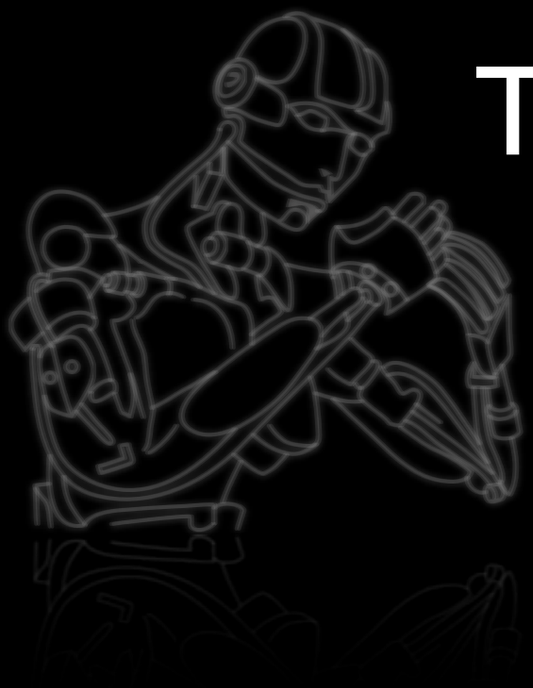
of the teacher.



Peters & Schaal (2003). Reinforcement Learning for Humanoid Robotics, HUMANOIDS

Outline

1. Introduction: Robot Skill Learning in Four Slides
2. Trade-Off I: Generality vs Physical Knowledge
3. Trade-Off II: Information Loss vs Self-Improvement
4. Trade-Off III: Large Repertoire vs Parsimonious Behavior Representation
5. Conclusion



Trade-Off 1: Physical Knowledge vs Generality

Use all physics... ⟷ Learn everything...

Trade-Off 1: Physical Knowledge vs Generality

Use all physics... ⟷ Learn everything...

- Physical knowledge is extremely powerful!
- Physics yields general solutions!
- Physics can explain itself!
- Handcrafting with Physics Knowledge can be easy

Trade-Off 1: Physical Knowledge vs Generality

Use all physics... ⟷ Learn everything...

- Physical knowledge is extremely powerful, but ...
- ...classical robotics as well as classical AI have failed at encoding the whole world with unlimited precision.
- ...all physical models are wrong (but some are useful)
- learning allows for higher accuracy, robustness and autonomy!

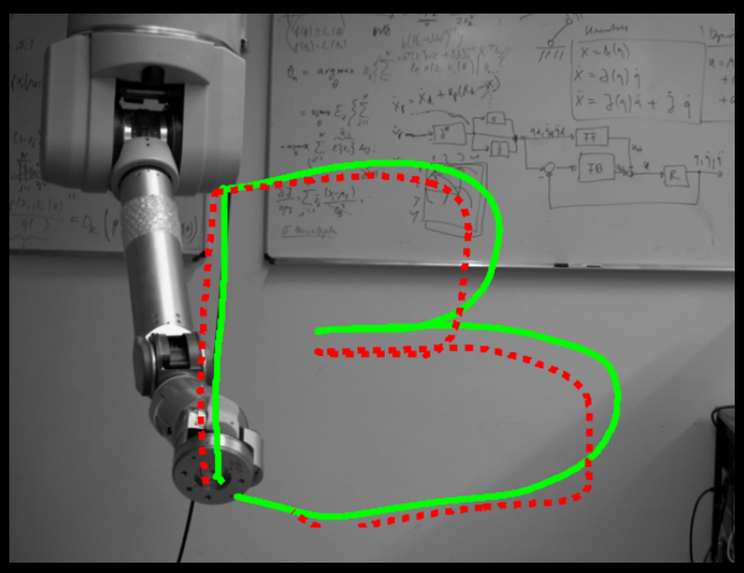
Trade-Off 1: Physical Knowledge vs Generality

Optimal Trade-Off

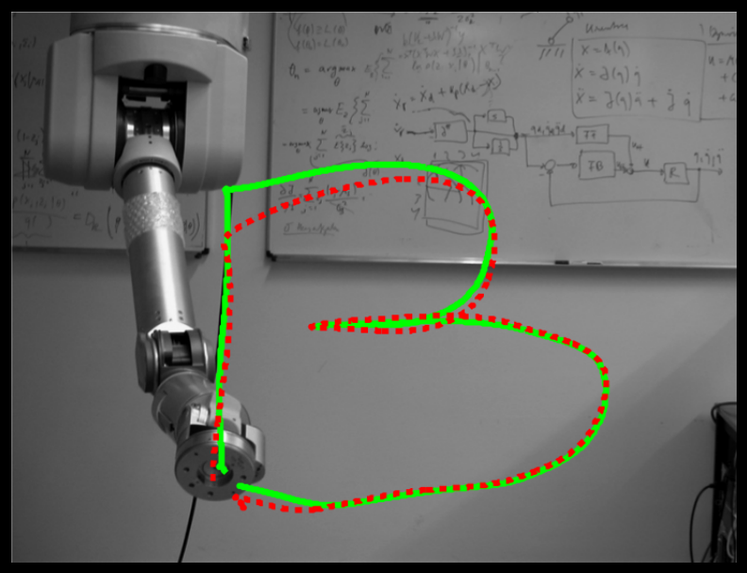
Use all physics... ← → Learn everything...

Learning beats physical knowledge if we extract crucial features...

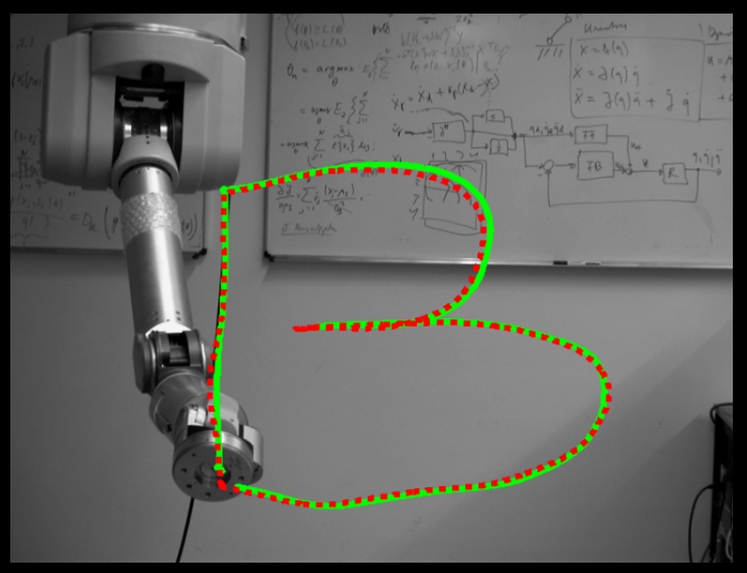
Physical Model



Offline Trained Model



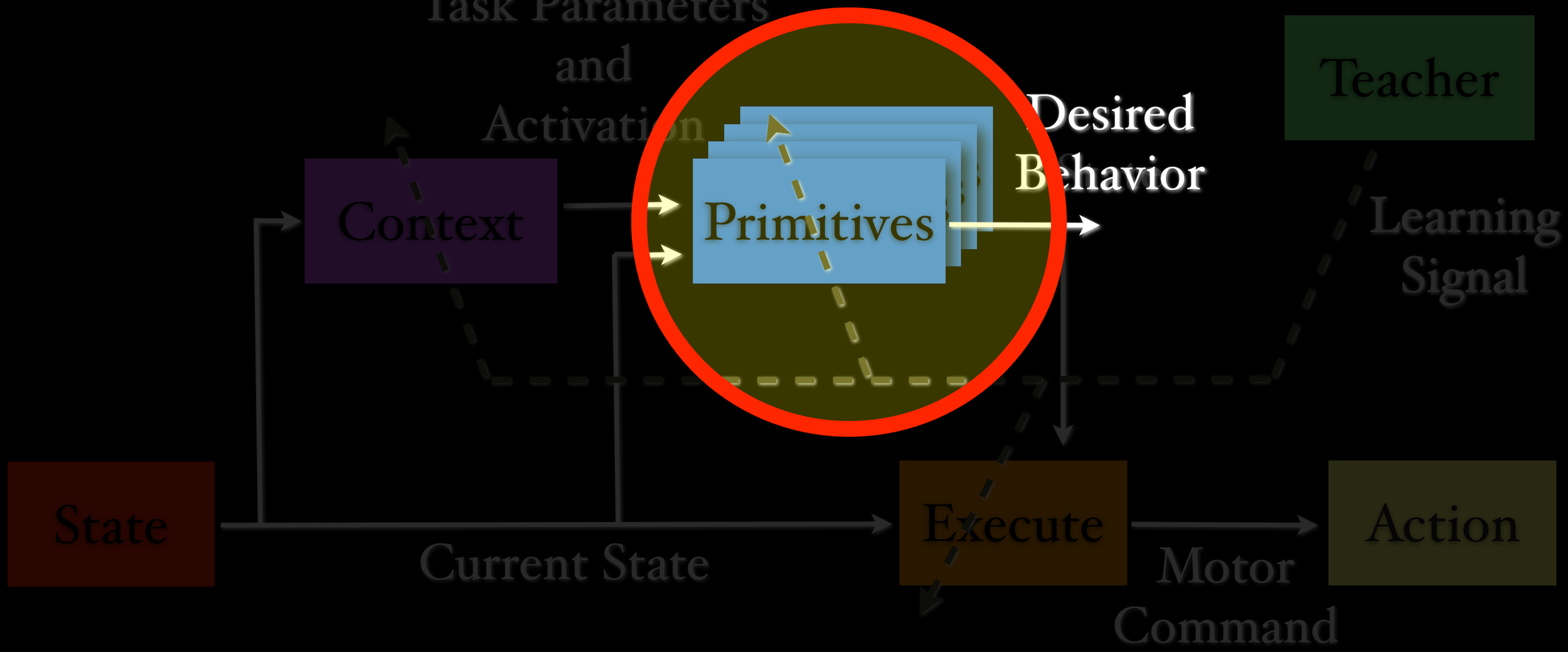
Online Trained



Lesson: We need a physics-based blue print!

Nguyen-Tuong & Peters, ARJ 2010

A Physics-inspired Blue Print



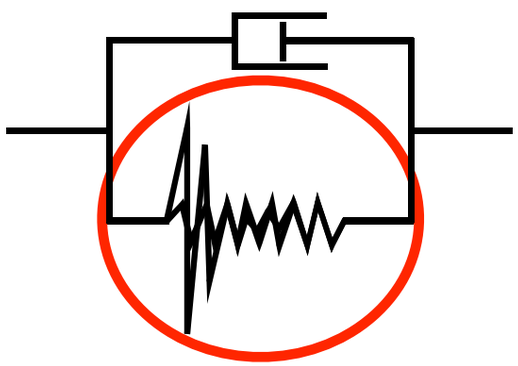
A Physics-inspired Blue Print

Use all physics... ⟵ Optimal Trade-Off ⟶ Learn everything...

Task/Hyperparameter

Trajectory Plan
Dynamics

$$\begin{cases} \dot{z} = \alpha_z\left(\beta_z(g-y)-z\right) \\ \dot{y} = \alpha_y\left(f(x,v)+z\right) \end{cases}$$



where

Canonical
Dynamics

$$\begin{cases} \dot{v} = \alpha_v\left(\beta_v(g-x)-v\right) \\ \dot{x} = \alpha_x v \end{cases}$$

Linear in learnable
Policy Parameters

Local Linear
Model Approx.

$$\begin{cases} f(x,v) = \dfrac{\sum_{i=1}^{k} w_i b_i v}{\sum_{i=1}^{k} w_i} \\ w_i = \exp\left(-\frac{1}{2} d_i\left(\bar{x}-c_i\right)^2\right) \text{ and } \bar{x} = \dfrac{x-x_0}{g-x_0} \end{cases}$$

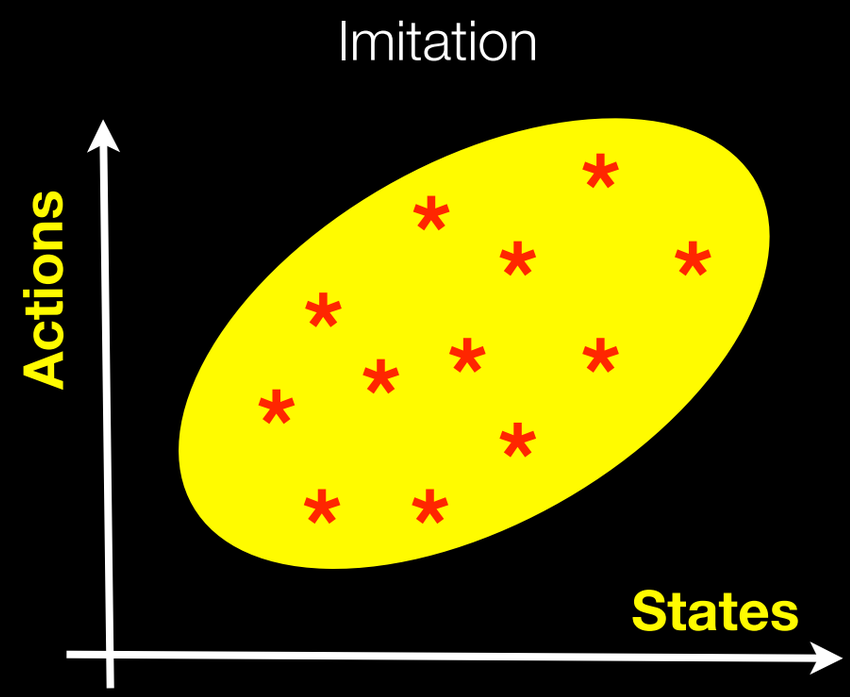
Imitation Learning

Given a physics-based policy, can we reproduce a path distribution?

- match given path distribution $p(\tau)$ with a new one $p_\theta(\tau)$, i.e.,

$$D(p_{\boldsymbol{\theta}}(\boldsymbol{\tau})||p(\boldsymbol{\tau})) \rightarrow \min$$

- only adapt the policy parameters θ
- model-free, purely sample-based
- results in one-shot and expectation maximization algorithms



Acquisition by Imitation

Kober & Peters (2009). Learning Motor Primitives, ICRA

Trade-Offs in Robot Skill Learning

Behavior Quality

Online Learning for higher Accuracy

Physics-knowledge in the representation

Efficiency of Generation

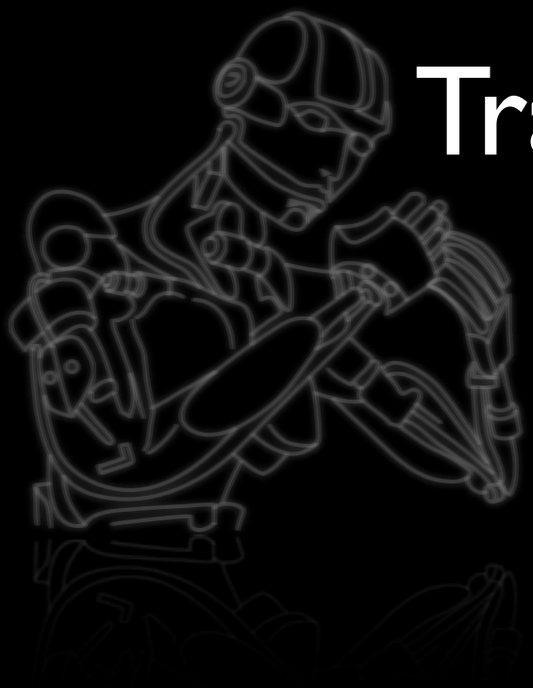
Trade-Offs in Robot Learning

Sufficiently general representations

Autonomy & Generality

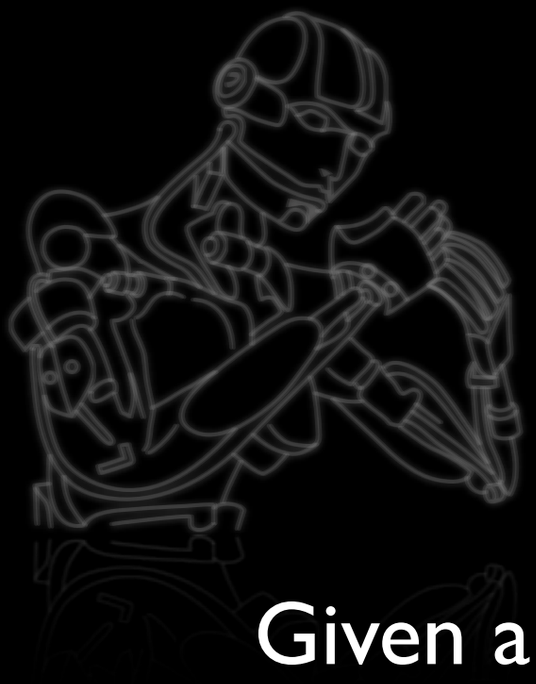
Outline

1. Introduction: Robot Skill Learning in Four Slides
2. Trade-Off I: Generality vs Physical Knowledge
3. Trade-Off II: Information Loss vs Self-Improvement
4. Trade-Off III: Large Repertoire vs Parsimonious Behavior Representation
5. Conclusion



Trade-Off II: Information Loss vs Self-Improvement

Experience ⟷ High Reward



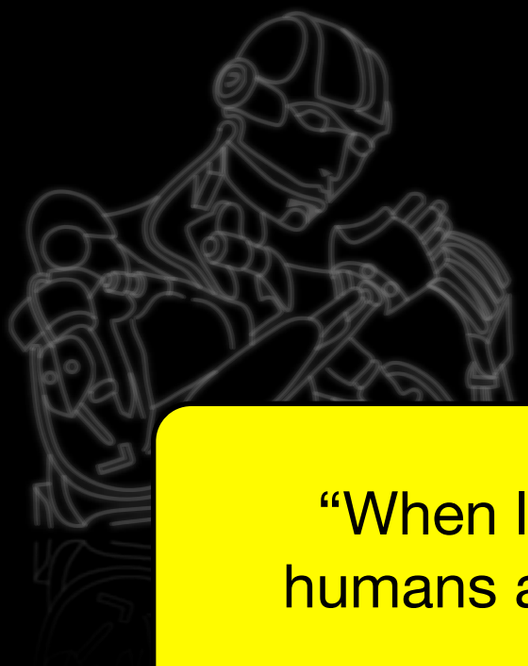
Reinforcement Learning

Given a path distribution, can we find the optimal policy?

- *Goal*: maximize the return of the paths $r(\tau)$ generated by path distribution $p_\theta(\tau)$
- Optimization function is the expected reward

$$J(\boldsymbol{\theta}) = \int_{\mathbb{T}} p_{\boldsymbol{\theta}}(\boldsymbol{\tau}) r(\boldsymbol{\tau}) d\boldsymbol{\tau}$$

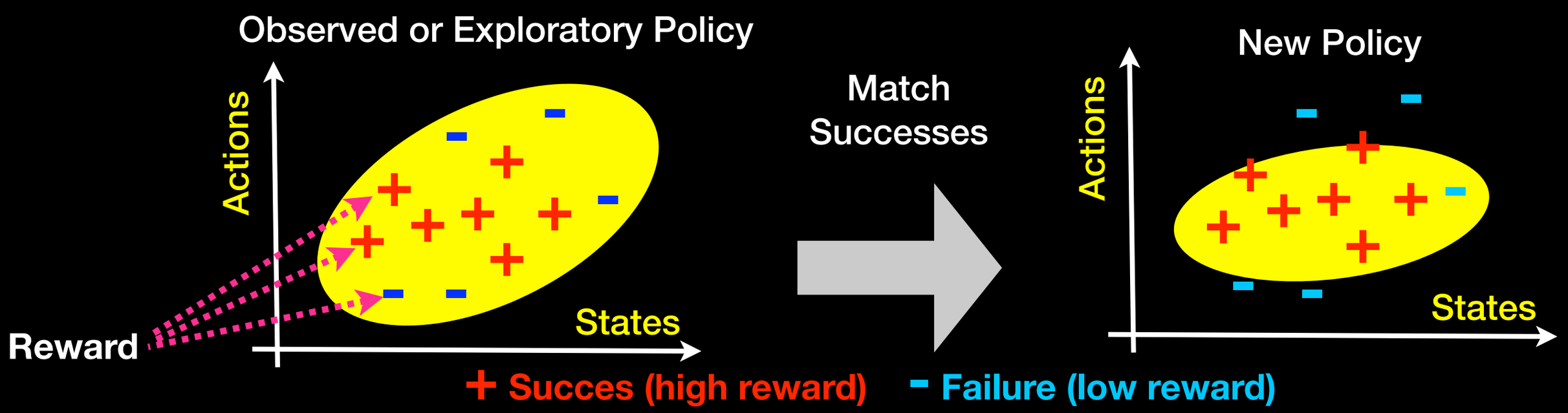
- This optimal control problem has no data in it!
- This part usually results into a greedy, softmax updates or a `vanilla' policy gradient algorithm...



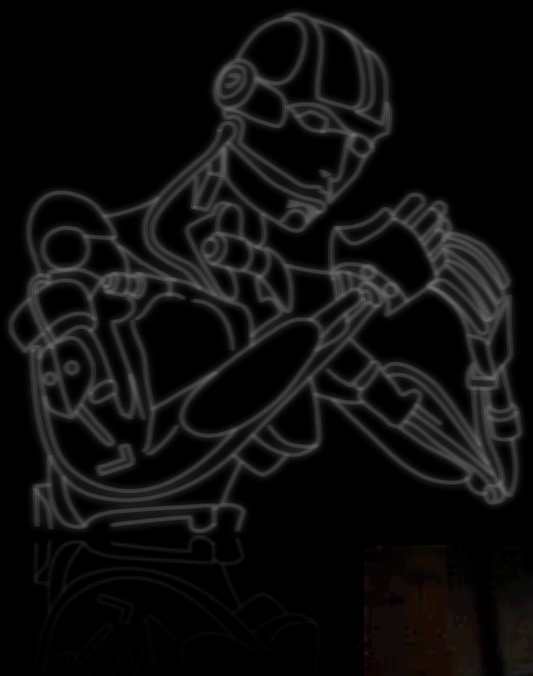
Success Matching

"When learning from a set of their own trials in iterated decision problems, humans attempt to match not the best taken action but the reward-weighted frequency of their actions and outcomes" (Arrow, 1958).

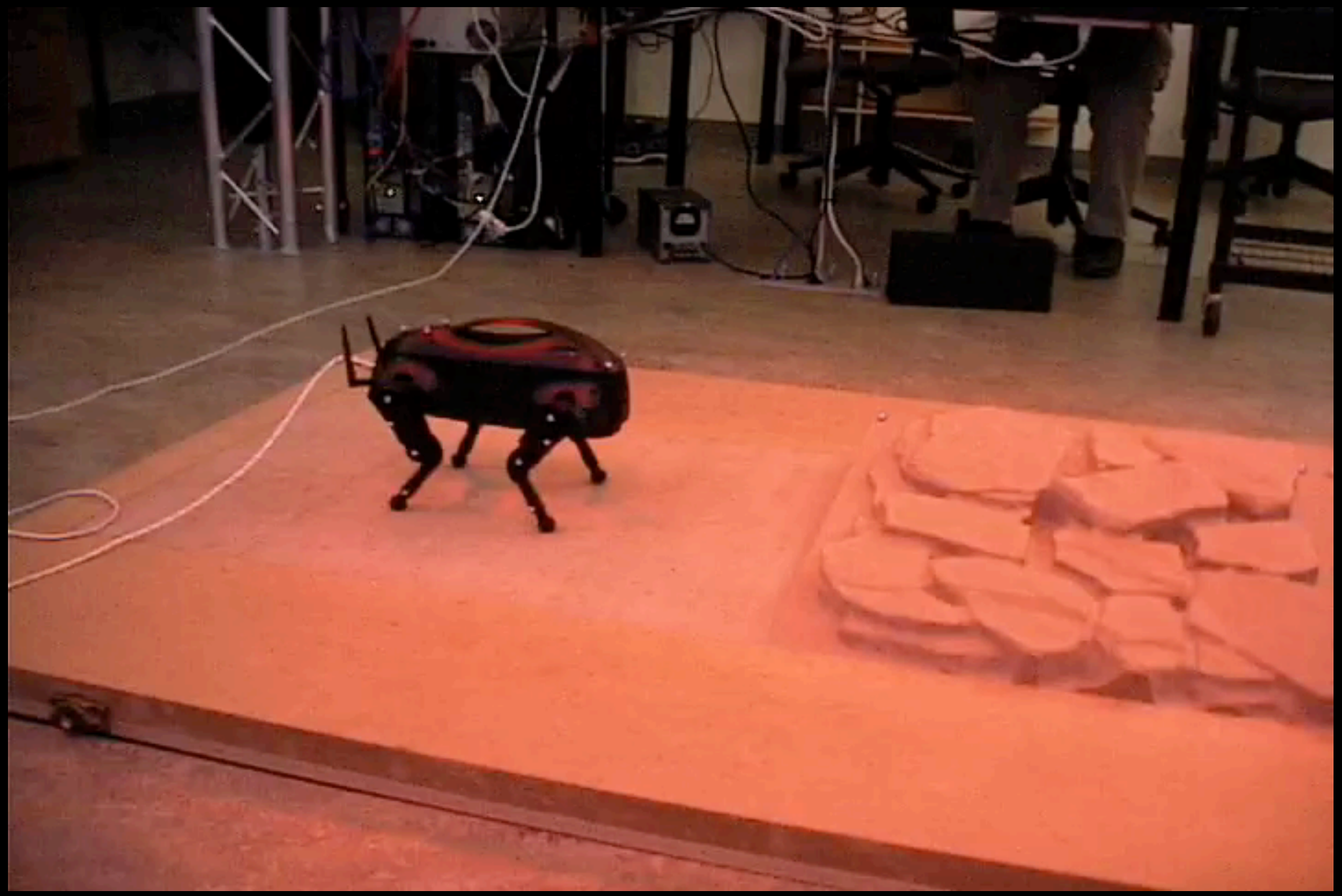
Can we create better policies by matching the reward-weighted previous policy ?



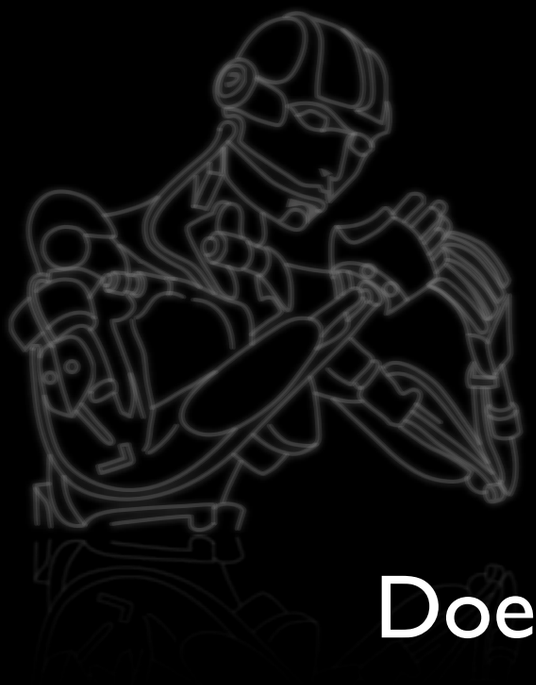
Many related frameworks, e.g., (Dayan&Hinton 1992;Andrews,'03;Attias,'04;Bagnell,'03;Toussaint,'06;...).



Illustrative Example
Foothold Selection



Match successful footholds!



What about the new cost function?

Does the new goal function help?

- We have a lower bound

$$\begin{aligned}\log J(\boldsymbol{\theta}) &= \log \int \frac{p(\boldsymbol{\tau})}{p(\boldsymbol{\tau})} p_{\boldsymbol{\theta}}(\boldsymbol{\tau}) r(\boldsymbol{\tau}) d\boldsymbol{\tau}, \\ &\geq \int p(\boldsymbol{\tau}) r(\boldsymbol{\tau}) \log \frac{p_{\boldsymbol{\theta}}(\boldsymbol{\tau})}{p(\boldsymbol{\tau})} d\boldsymbol{\tau} \propto -D(p_{\boldsymbol{\theta}}(\boldsymbol{\tau}) || p(\boldsymbol{\tau}) r(\boldsymbol{\tau})),\end{aligned}$$

- Having taken a low reward trajectory many times resembles having taken a high reward trajectory once!

➡ A safe way of trading experience against reward!

Experience ⟷ High Reward

Safe but unclear Trade-Off

Reinforcement Learning by Reward-Weighted Imitation

Matching successful actions corresponds to minimizing the Kullback-Leibler 'distance'

$$D(p_{\boldsymbol{\theta}}(\boldsymbol{\tau})||r(\boldsymbol{\tau})p(\boldsymbol{\tau})) \rightarrow \min$$

For a Gaussian policy $\pi(\mathbf{u}|\mathbf{x}) = \mathcal{N}(\mathbf{u}|\boldsymbol{\phi}(\mathbf{x})^T\boldsymbol{\theta}, \sigma^2\mathbf{I})$, we get the update rule

$$\theta_{k+1} = (\boldsymbol{\Phi}^T\mathbf{R}\boldsymbol{\Phi})^{-1}\boldsymbol{\Phi}^T\mathbf{R}\mathbf{U}$$

New Policy Parameters — Features — Rewards — Actions

➡Reduces Reinforcement Learning onto Reward Weighted Regression!

Peters & Schaal (2007). Policy Learning for Motor Skills, International Conference on Machine Learning (ICML)
Kober & Peters (2009). Policy Search for Motor Primitives in Robotics, Advances in Neural Information Processing Systems (NIPS)

Self-Improvement by Reinforcement Learning

Kober & Peters (2009). Policy Search for Motor Primitives in Robotics, NIPS

More focussed trade-off?

EM-like Methods: Safe but unclear Trade-Off

Experience ⟷ High Reward

REPS: Adjustable Trade-Off

Relative Entropy Policy Search (REPS)

$$\max_{\pi,\mu^\pi} J(\pi) = \sum_{s,a} \mu^\pi(s)\pi(a|s)\mathcal{R}_{sa}$$ Maximize reward

$$1 = \sum_{s,a} \mu^\pi(s)\pi(a|s)$$ Probability distribution

$$\mu^\pi(s') = \sum_{s,a} \mathcal{P}^a_{ss'}\mu^\pi(s)\pi(a|s)$$ Follow system dynamics

$$\epsilon \geq \sum_{s,a} \mu^\pi(s)\pi(a|s)\log\frac{\mu^\pi(s)\pi(a|s)}{q(s,a)}$$ Close to training data (no wild exploration)

Peters, Muelling, Altun (2010). Relative Entropy Policy Search, AAAI

Trade-Offs in Robot Skill Learning

Behavior Quality

Highly rewarded behaviors!

Safe Learning Approaches by Retaining Experience!

Initial Demonstrations

Efficiency of Generation

Slow Loss of (Bad) Experience

Trade-Offs in Robot Learning

Autonomy & Generality

Outline

1. Introduction: Robot Skill Learning in Four Slides
2. Trade-Off I: Generality vs Physical Knowledge
3. Trade-Off II: Information Loss vs Self-Improvement
4. Trade-Off III: Large Repertoire vs Parsimonious Behavior Representation
5. Conclusion

Trade-Off III: Large Repertoire vs Parsimonious Behavior Representation

Versatility ⟷ Parsimony

- Versatility: Complex behaviors rely on a combination of as many strategies as possible.
- Parsimony: Multiple, distinct solutions are required for *efficient* learning.

Versatility with Many Primitives

Relative Entropy Policy Search (REPS)

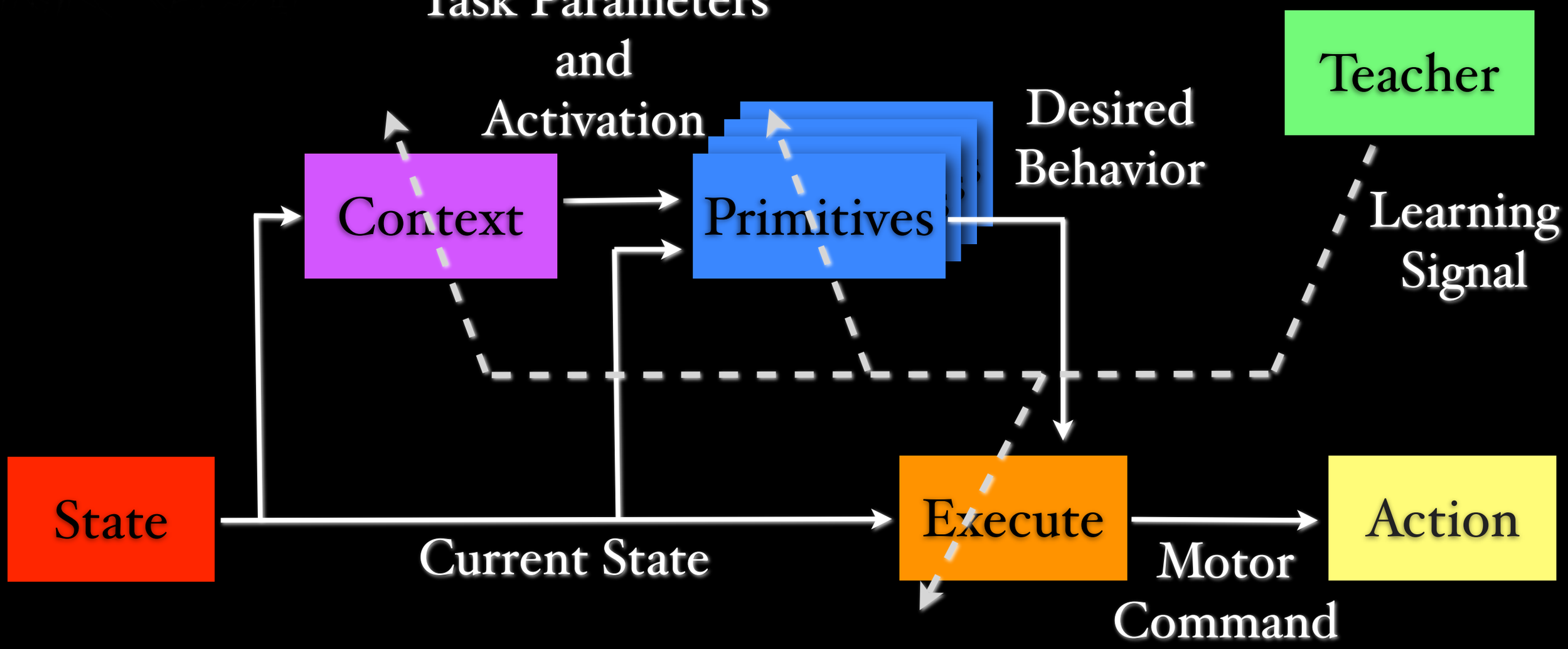
$$\max_{\pi,\mu^\pi} J(\pi) = \sum_{s,a} \mu^\pi(s)\pi(a|s)\mathcal{R}_{sa}$$ Maximize reward

$$1 = \sum_{s,a} \mu^\pi(s)\pi(a|s)$$ Probability distribution

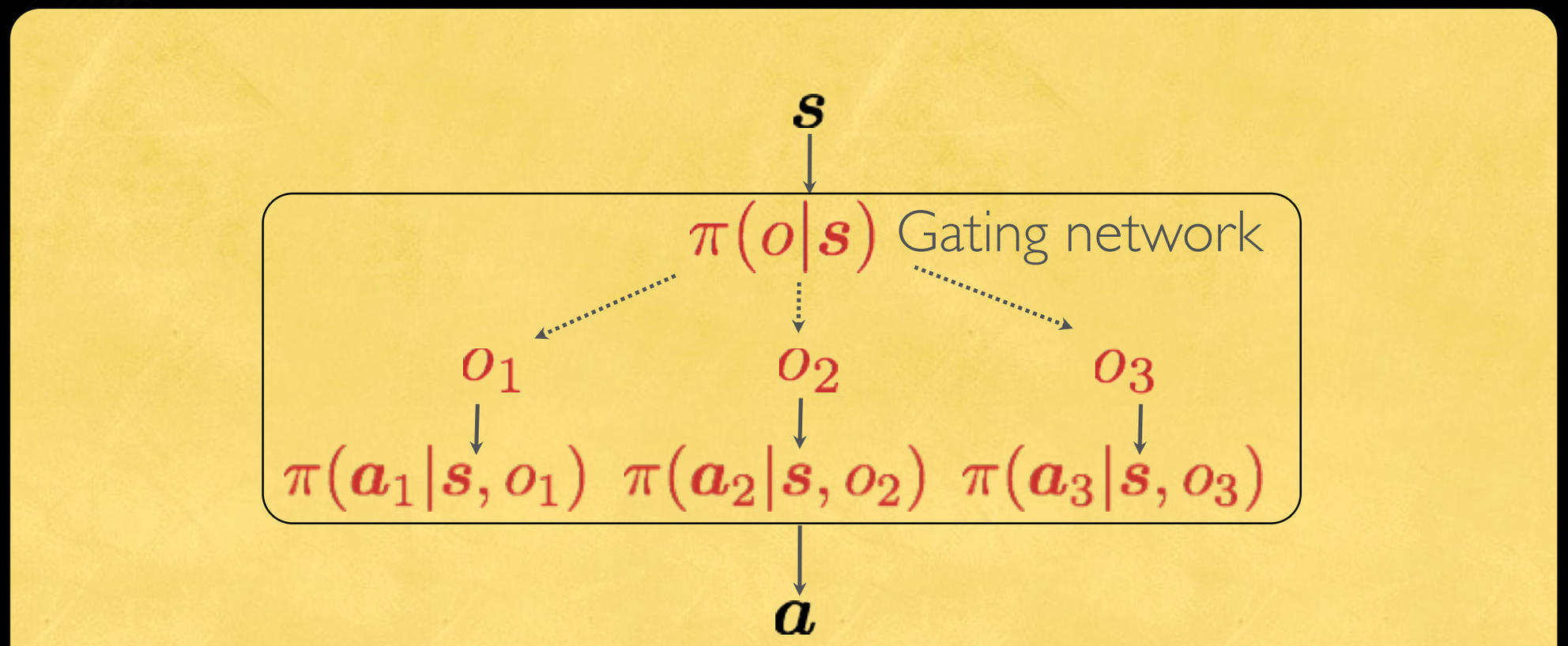
$$\mu^\pi(s') = \sum_{s,a} \mathcal{P}^a_{ss'}\mu^\pi(s)\pi(a|s)$$ Follow system dynamics

$$\epsilon \geq \sum_{s,a} \mu^\pi(s)\pi(a|s)\log\frac{\mu^\pi(s)\pi(a|s)}{q(s,a)}$$ Close to training data (no wild exploration)

A Physics-inspired Blue Print



Hierarchies with Primitives



Daniel, Neumann & Peters (2012). Hierarchical Relative Entropy Policy Search, AI-Stats

Versatility with Many Primitives

"Naive" Extension of REPS

$$\max_{\pi,\mu^\pi} J(\pi) = \sum_{\boldsymbol{s},\boldsymbol{a},o} \mu^\pi(\boldsymbol{s})\pi(\boldsymbol{a},o|\boldsymbol{s})\mathcal{R}_{\boldsymbol{sa}}$$ Maximize reward

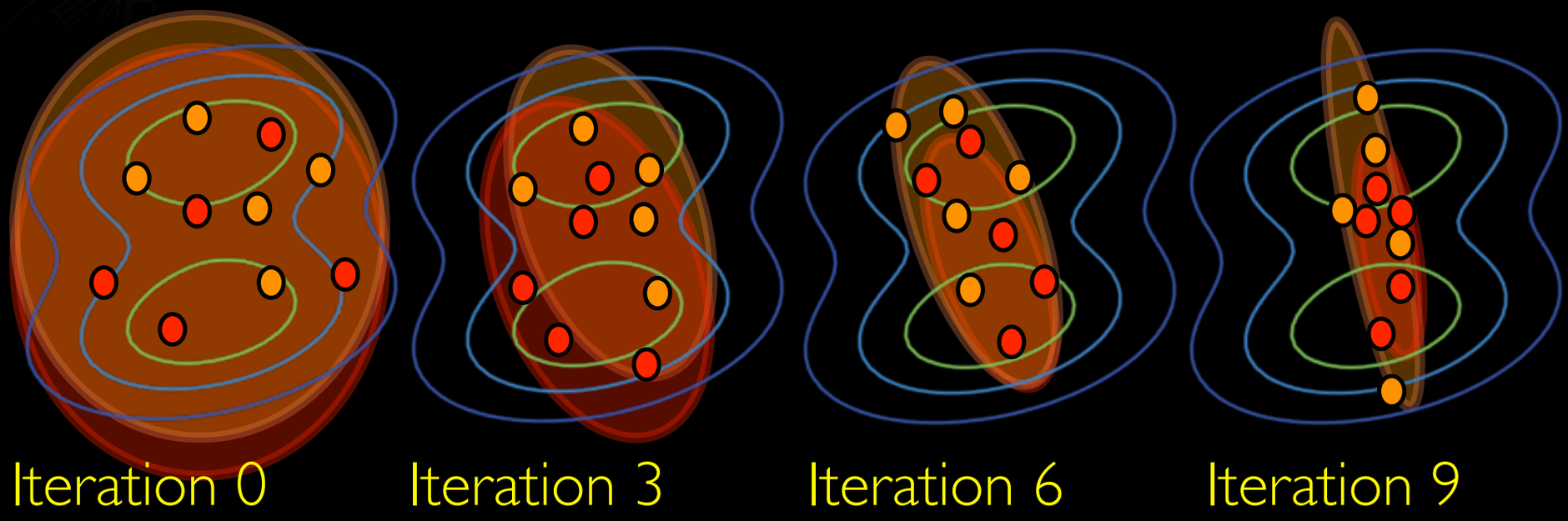
$$1 = \sum_{\boldsymbol{s},\boldsymbol{a},o} \mu^\pi(\boldsymbol{s})\pi(\boldsymbol{a},o|\boldsymbol{s})$$ Probability distribution

$$\mu^\pi(\boldsymbol{s}') = \sum_{\boldsymbol{s},\boldsymbol{a},o} \mathcal{P}^{\boldsymbol{a}}_{\boldsymbol{ss}'}\mu^\pi(\boldsymbol{s})\pi(\boldsymbol{a}|o,\boldsymbol{s})\pi(o|\boldsymbol{s})$$ Follow system dynamics

$$\epsilon \geq \sum_{\boldsymbol{s},\boldsymbol{a},o} \mu^\pi(\boldsymbol{s})\pi(\boldsymbol{a},o|\boldsymbol{s}) \log \frac{\mu^\pi(\boldsymbol{s})\pi(\boldsymbol{a},o|\boldsymbol{s})}{q(\boldsymbol{s},\boldsymbol{a},o)}$$ Close to training data (no wild exploration)

Daniel, Neumann & Peters (2012). Hierarchical Relative Entropy Policy Search, AI-Stats

Need for Separation



If all primitives are equally responsible, we can represent versatile behavior but it will never be parsimonious.

Daniel, Neumann & Peters (2012). Hierarchical Relative Entropy Policy Search, AI-Stats

Versatility with Many Primitives

Hierachical Relative Entropy Policy Search (HiREPS)

$$\max_{\pi,\mu^\pi} J(\pi) = \sum_{\boldsymbol{s},\boldsymbol{a},o} \mu^\pi(\boldsymbol{s})\pi(\boldsymbol{a},o|\boldsymbol{s})\mathcal{R}_{\boldsymbol{sa}}$$ Maximize reward

$$1 = \sum_{\boldsymbol{s},\boldsymbol{a},o} \mu^\pi(\boldsymbol{s})\pi(\boldsymbol{a},o|\boldsymbol{s})$$ Probability distribution

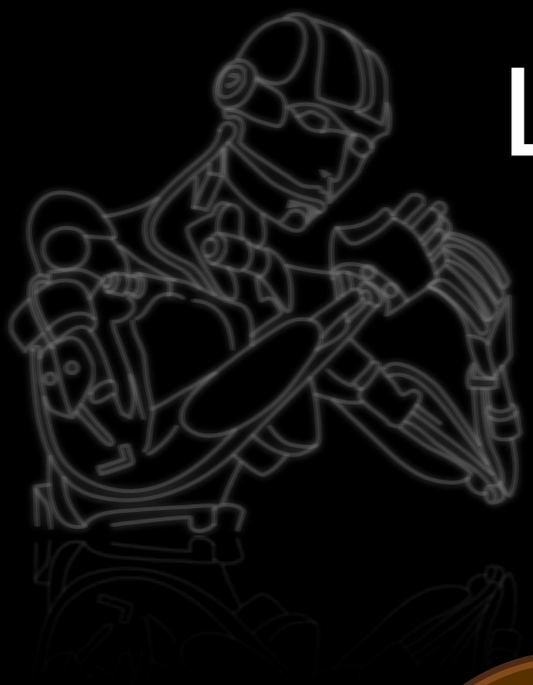
$$\mu^\pi(\boldsymbol{s}') = \sum_{\boldsymbol{s},\boldsymbol{a},o} \mathcal{P}^{\boldsymbol{a}}_{\boldsymbol{ss}'}\mu^\pi(\boldsymbol{s})\pi(\boldsymbol{a}|o,\boldsymbol{s})\pi(o|\boldsymbol{s})$$ Follow system dynamics

$$\epsilon \geq \sum_{\boldsymbol{s},\boldsymbol{a},o} \mu^\pi(\boldsymbol{s})\pi(\boldsymbol{a},o|\boldsymbol{s}) \log \frac{\mu^\pi(\boldsymbol{s})\pi(\boldsymbol{a},o|\boldsymbol{s})}{q(\boldsymbol{s},\boldsymbol{a},o)}$$ Close to training data (no wild exploration)

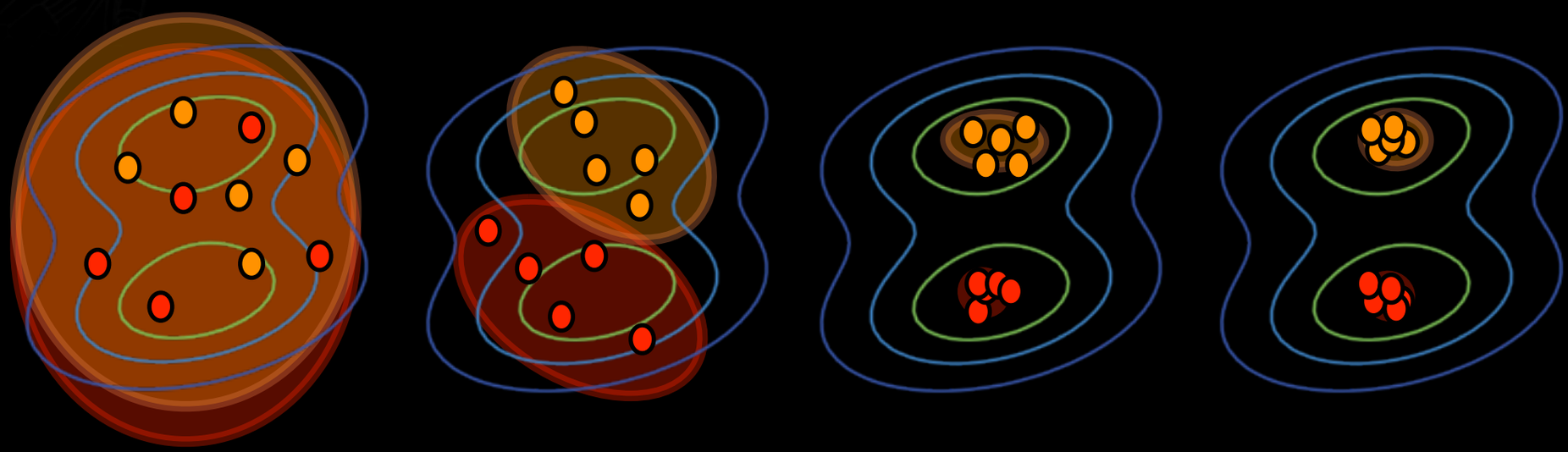
$$\kappa \geq \mathbb{E}_{\boldsymbol{s},\boldsymbol{a}}\left[\sum_{o} -p(o|\boldsymbol{s},\boldsymbol{a}) \log p(o|\boldsymbol{s},\boldsymbol{a})\right]$$ Force the primitives to limited responsibility

High entropy indicates overlap!

Daniel, Neumann & Peters (2012). Hierarchical Relative Entropy Policy Search, AI-Stats



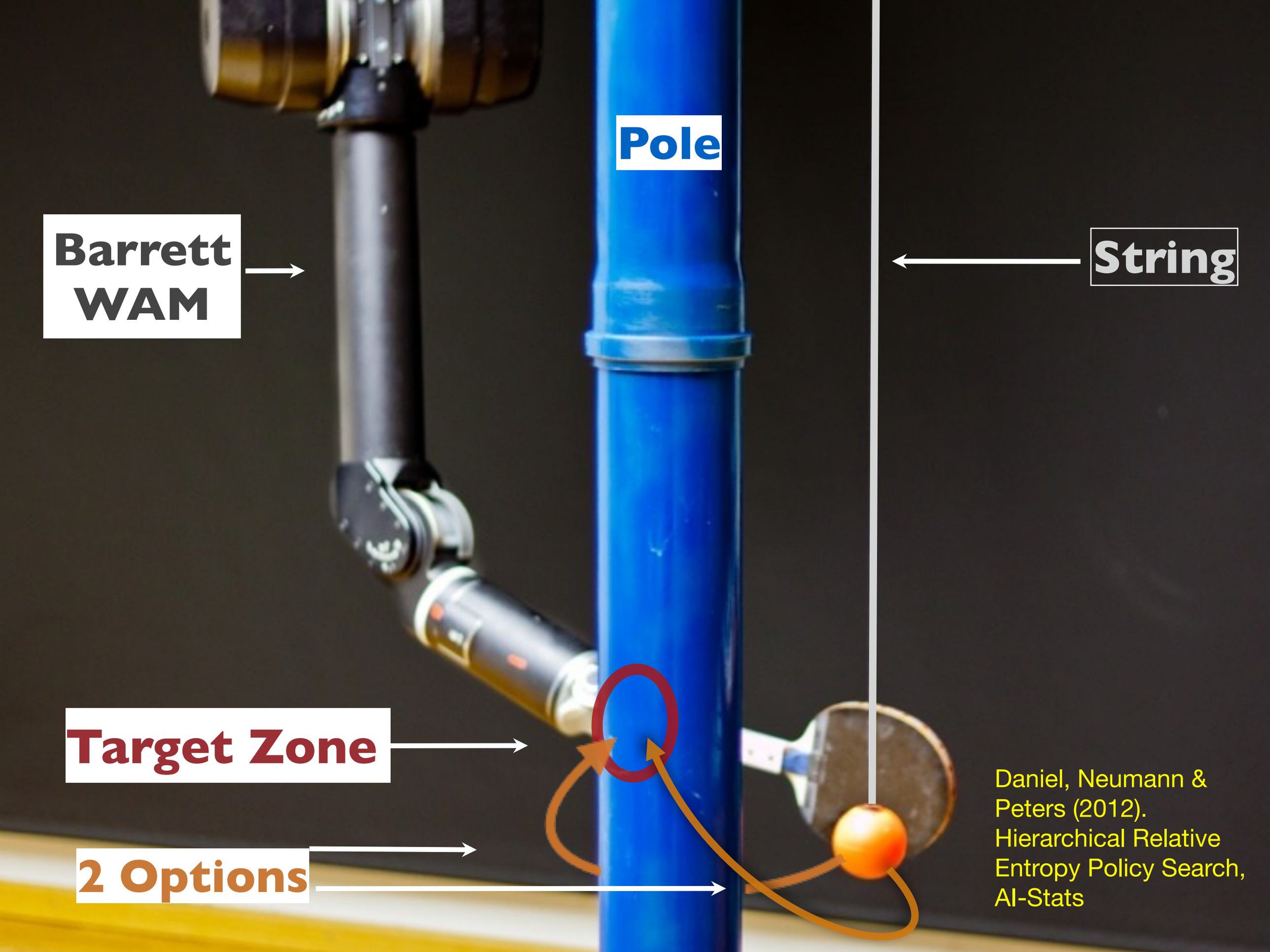
Localized behavior can be learned efficiently!



Iteration 0 Iteration 3 Iteration 6 Iteration 9

We can reduce to the number of needed primitives!

Daniel, Neumann & Peters (2012). Hierarchical Relative Entropy Policy Search, AI-Stats


Pole
Barrett WAM
String
Target Zone
2 Options
Daniel, Neumann & Peters (2012). Hierarchical Relative Entropy Policy Search, AI-Stats

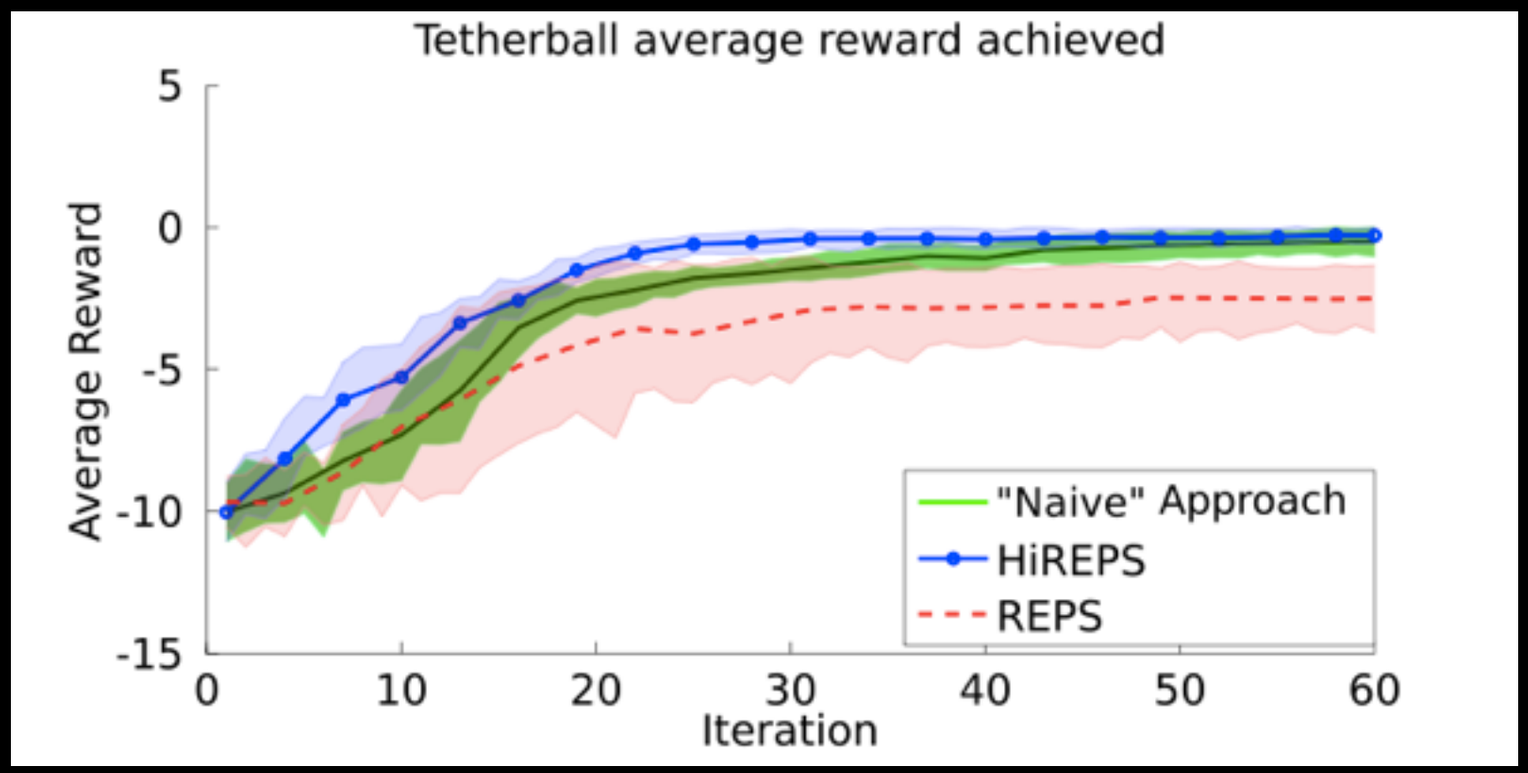
Versatility ⟷ Parsimony

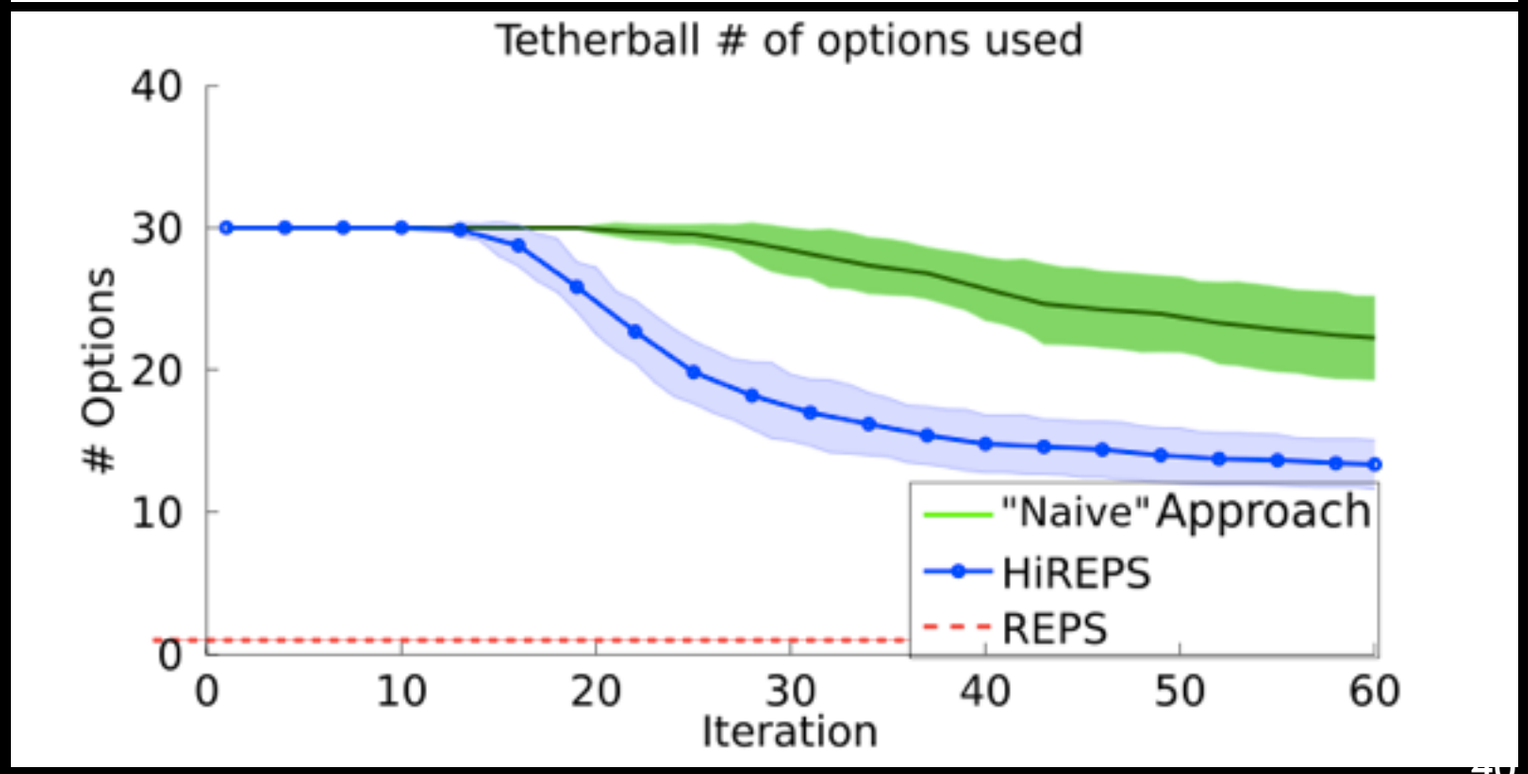
Adjustable Trade-Off

Good performance

Fast reduction in the number of primitives

Daniel, Neumann & Peters (2012). Hierarchical Relative Entropy Policy Search, AI-Stats





Demonstrations

K.Mülling, J.Kober, O. Krömer, J.Peters (accepted). *Learning to Select and Generalize Striking Movements in Robot Table Tennis*, International Journal of Robotics Research

Select & Generalize

From Imitation Learning

we obtain 25 Movement Primitives

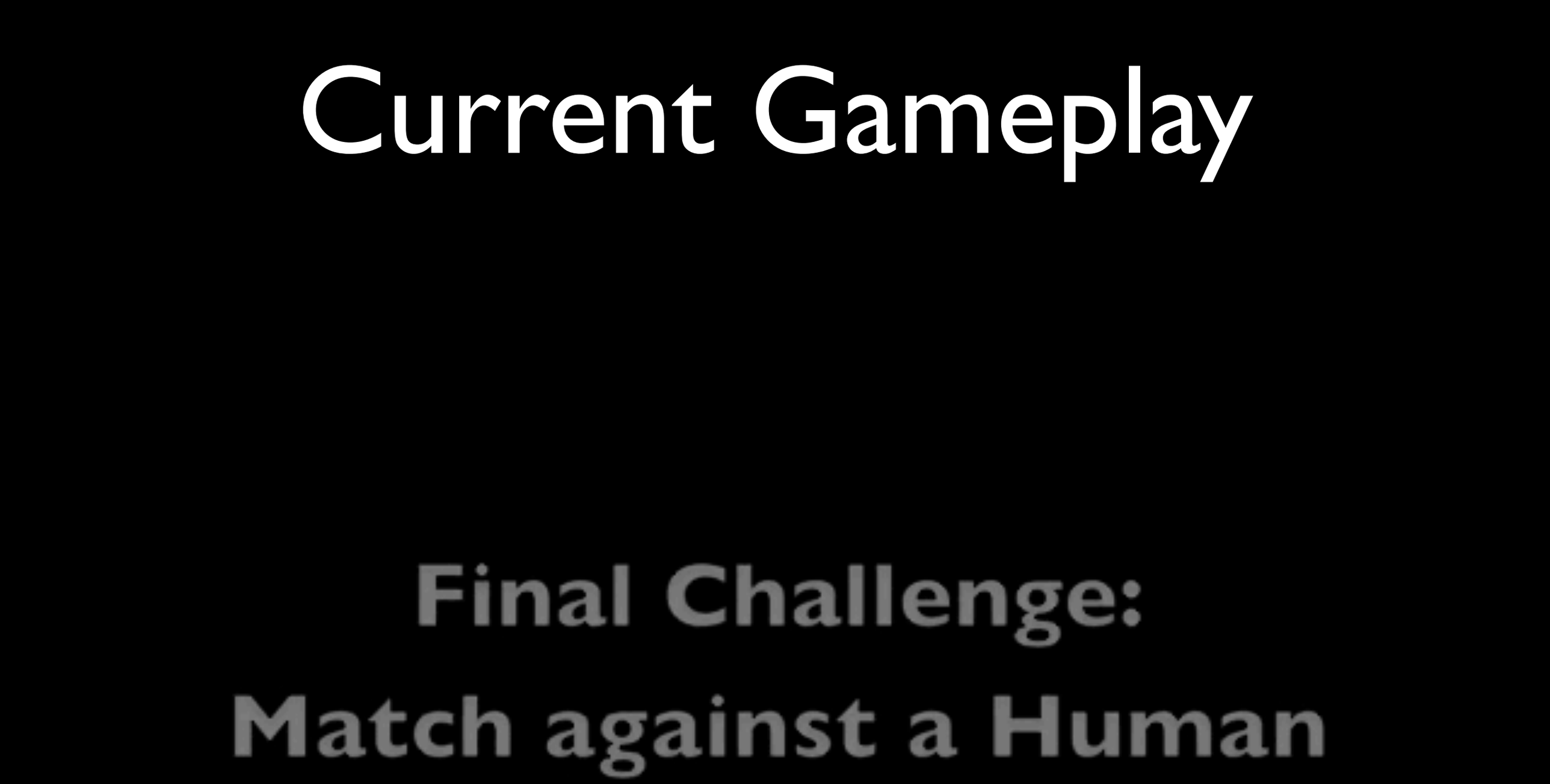
K.Mülling, J.Kober, O. Krömer, J.Peters (accepted). *Learning to Select and Generalize Striking Movements in Robot Table Tennis*, International Journal of Robotics Research

Self-Improvement

Training a Hitting Region with an Initial Success Rate of 0%

K.Mülling, J.Kober, O. Krömer, J.Peters (accepted). *Learning to Select and Generalize Striking Movements in Robot Table Tennis*, International Journal of Robotics Research

Current Gameplay



K.Mülling, J.Kober, O. Krömer, J.Peters (accepted). *Learning to Select and Generalize Striking Movements in Robot Table Tennis*, International Journal of Robotics Research

Trade-Offs in Robot Skill Learning

Behavior Quality

Efficiency of Generation

Versatility

Parsimonity

Trade-Offs in Robot Learning

Autonomy & Generality

Outline

1. Introduction: Robot Skill Learning in Four Slides
2. Trade-Off I: Generality vs Physical Knowledge
3. Trade-Off II: Information Loss vs Self-Improvement
4. Trade-Off III: Large Repertoire vs Parsimonious Behavior Representation
5. Conclusion

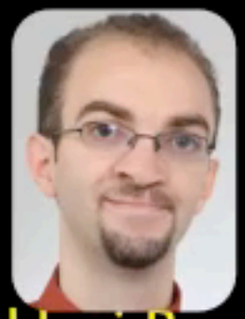
Conclusion

- Motor skill learning is a promising way to avoid programming all possible scenarios and continuously adapt to the environment.
- To make it work, we always have to trade-off at least:
  - Physical Knowledge vs Learning
  - Experienced vs High Rewards
  - Versatility vs Parsimony
- There are many more trade-offs in practice.

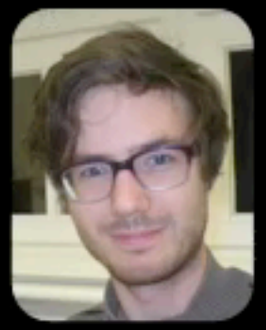
Thanks for your Attention!

Zhikun Wang

Abdeslam Boularias

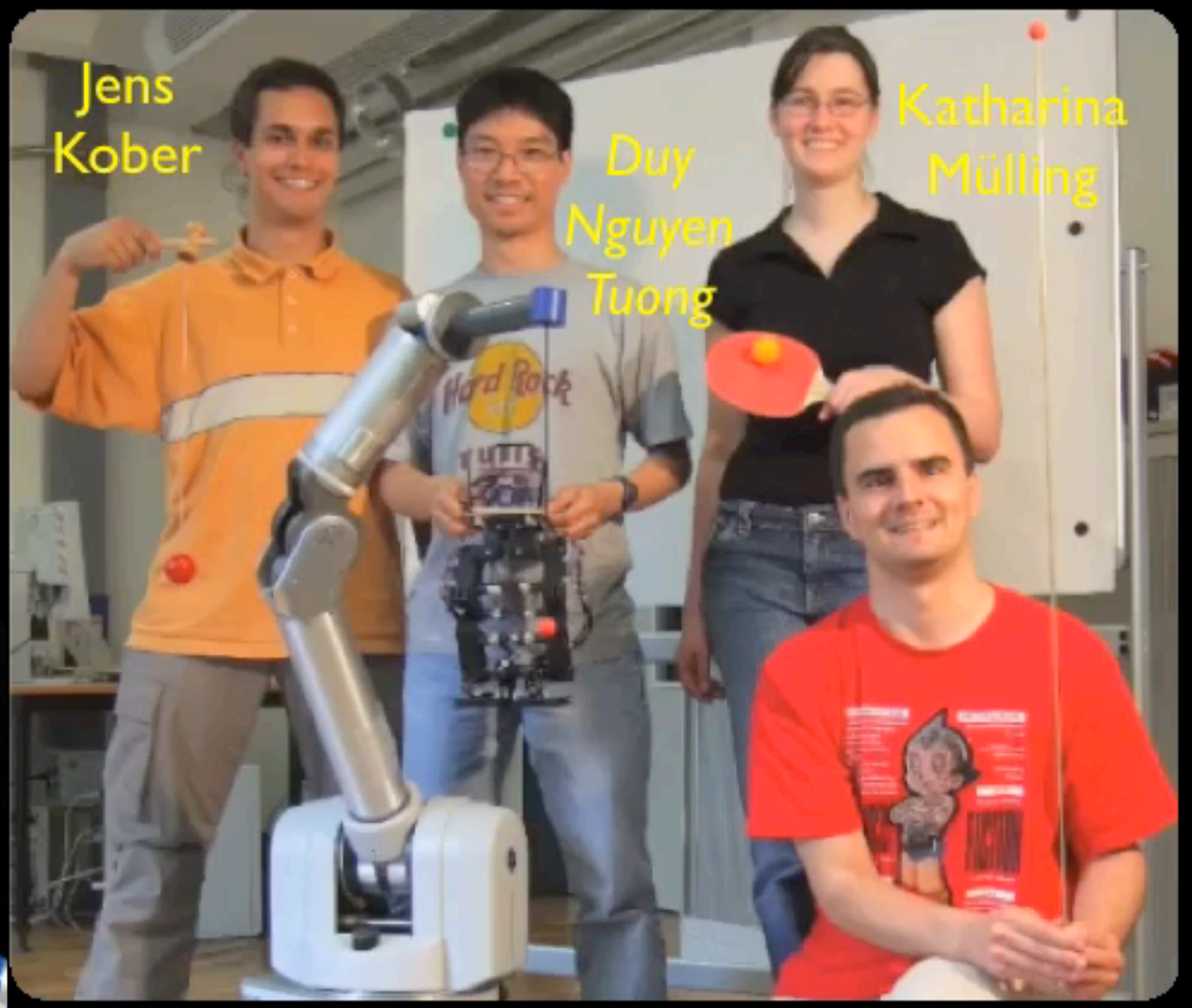


Heni Ben Amor



Gerhard Neumann

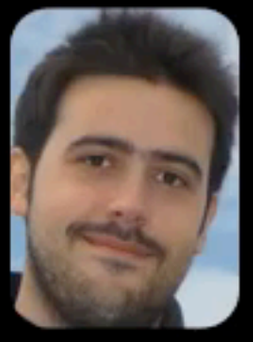
Oliver Kroemer



Jens Kober

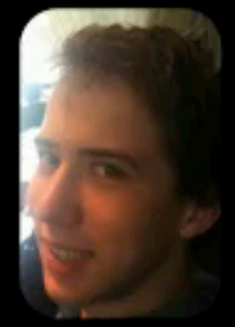
Duy Nguyen Tuong

Katharina Mülling



Roberto Calandra

Marc Deisenroth



Herke van Hoof

Christian Daniel

Alexandros Paraschos