Universität Hamburg
DER FORSCHUNG | DER LEHRE | DER BILDUNG

Model-Based Reinforcement Learning in Robotics

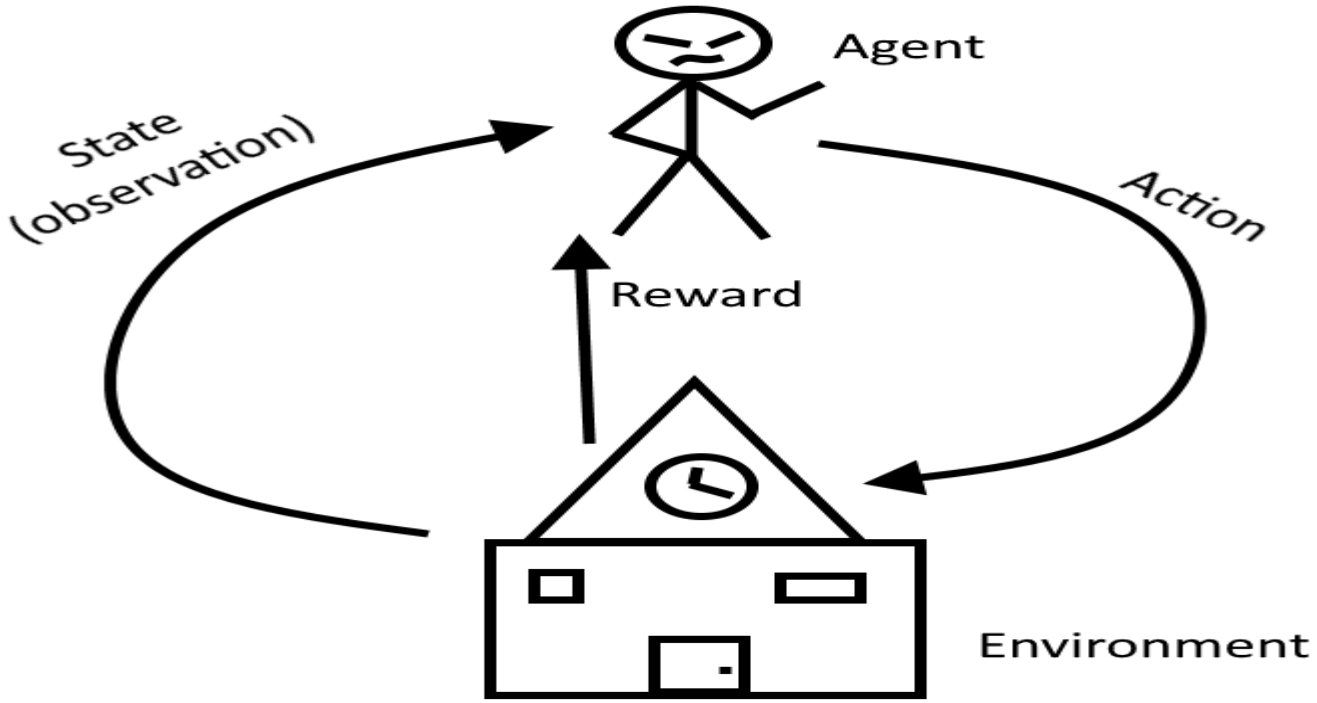
SOLVING THE CART-POLE TASK

Outline

1. Introduction
2. Background Knowledge
3. Methods
4. Results
5. Discussion
6. Conclusion & What‘s left

What is RL?

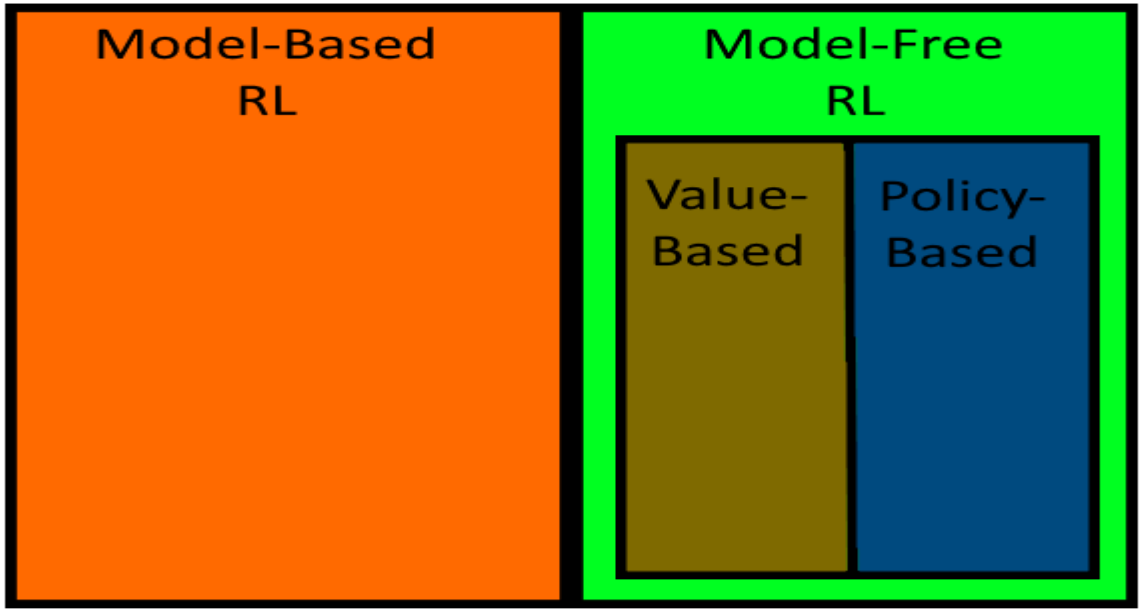
- RL is a subfield of machine learning
  - *Agent* performs optimally in its *environment;* i.e. *maximizes reward*



RL Framework. Adapted from [1]

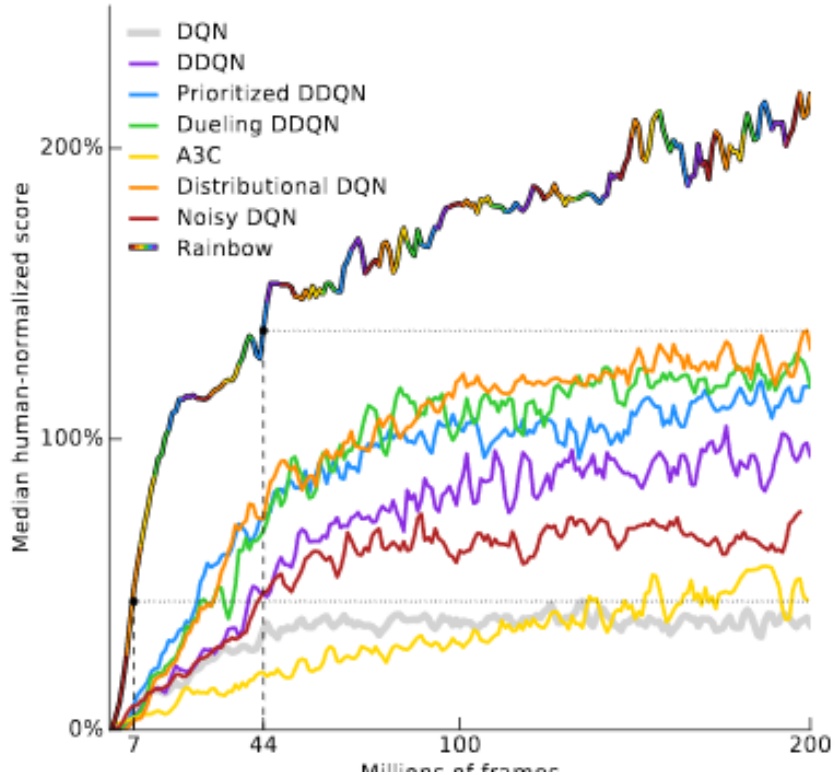
Subfields of RL

There are 3 RL categories



Subfields of RL. By author

Model-Free Example



Rainbow Algorithm [2]

$$1\,f = \frac{1}{60}s$$

$$200.000.000\,f = \ldots = 925{,}925\,h = 38{,}58\,d$$

Not feasible in robotics!

Model-Based vs. Model-Free

	Model-Based	Model-Free
+	Sample Efficient	High Performance
-	Low Performance	Very inefficient

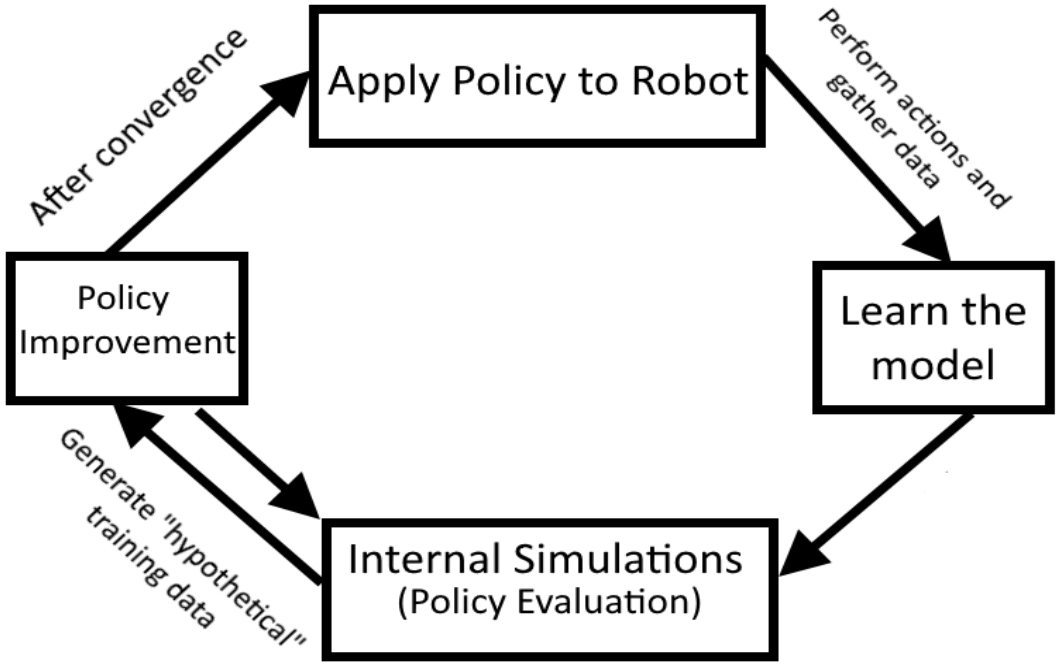
MB vs. MF Comparison [3][4]

Huge problem!

Outline

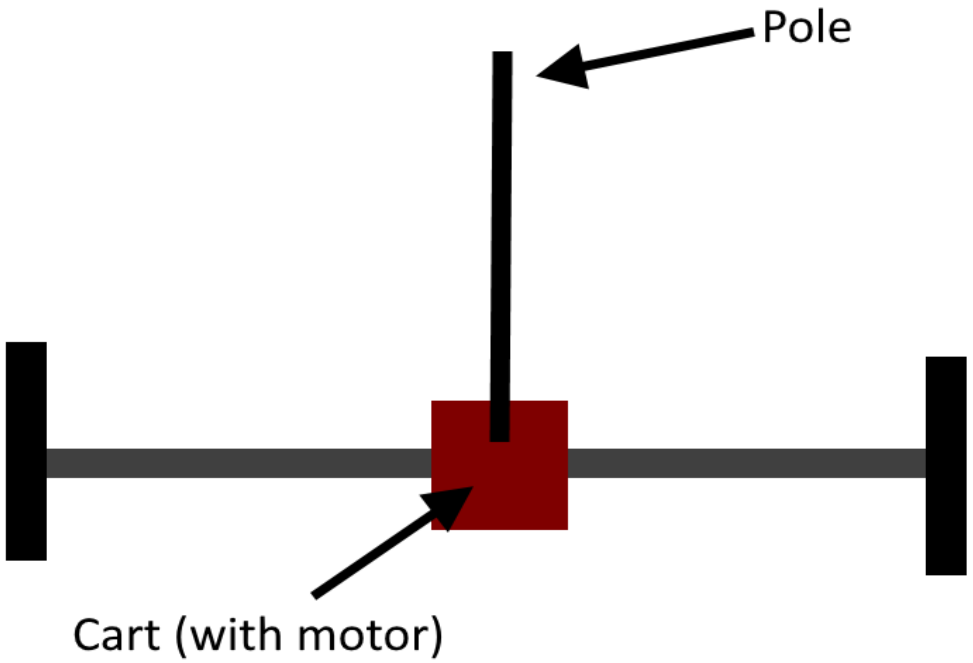
1. Introduction
2. Background Knowledge
3. Methods
4. Results
5. Discussion
6. Conclusion & What‘s left

How does Model-Based RL work?



Model-Based RL Framework. Adapted from [4]

What is the Cart-Pole Problem?

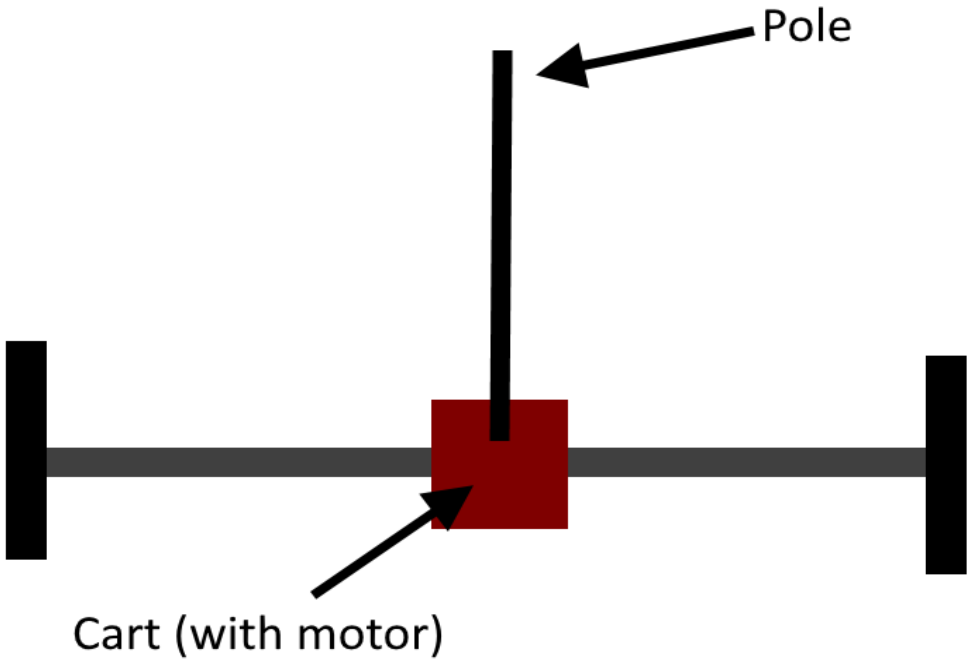


Cart-Pole Setting. By author; as described in [5]

Goal: Balance the pole as long as you can!

Actions: Move left or right

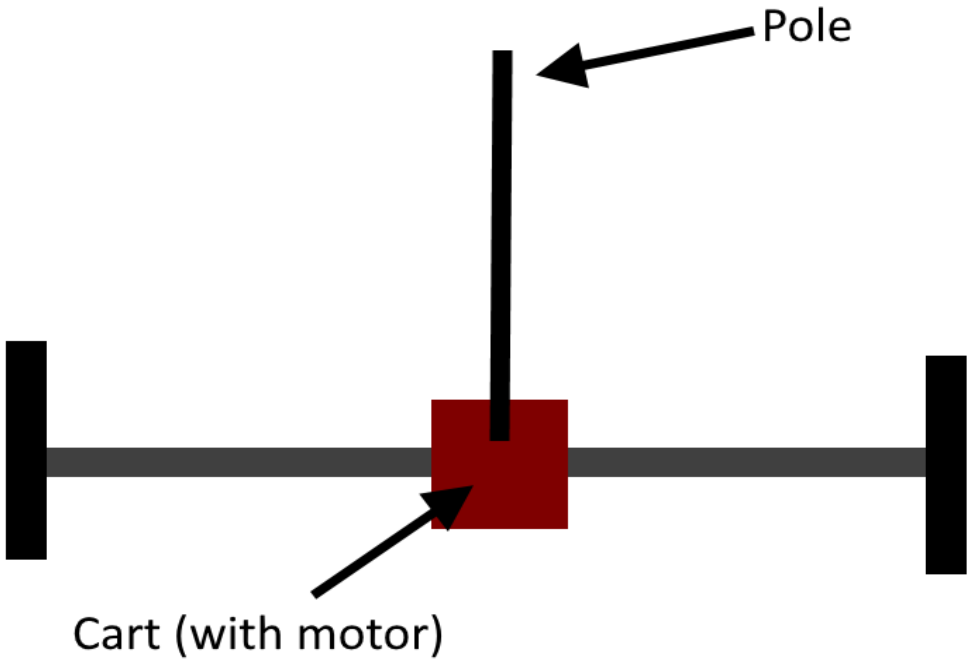
But why specifically the CP problem?



Cart-Pole Setting. By author; as described in [5]

- CP is (cheap and) easy to solve
  - If your algorithm can‘t handle CP, it won‘t handle anything

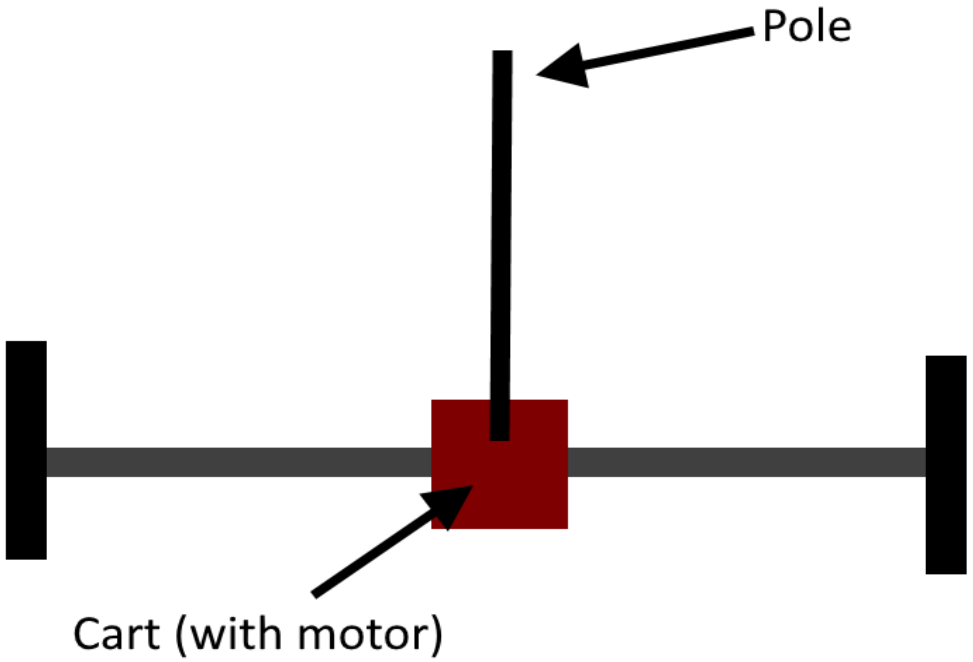
But why specifically the CP problem?



Cart-Pole Setting. By author; as described in [5]

- Quick way to see if you are on the right track
- Don‘t want to employ a „bad“ learning algorithm onto an expensive robot

But why specifically the CP problem?



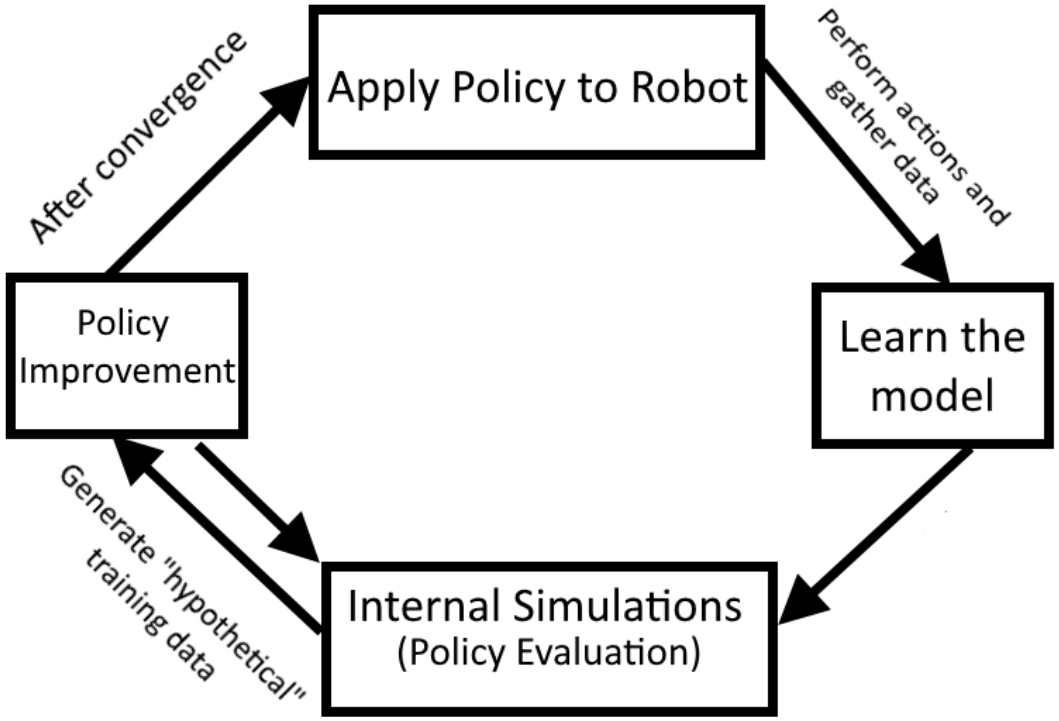
Cart-Pole Setting. By author; as described in [5]

- A benchmark to compare algorithms! (Atari for games; MuJoCo for physics simulations; and for robotics?)

Outline

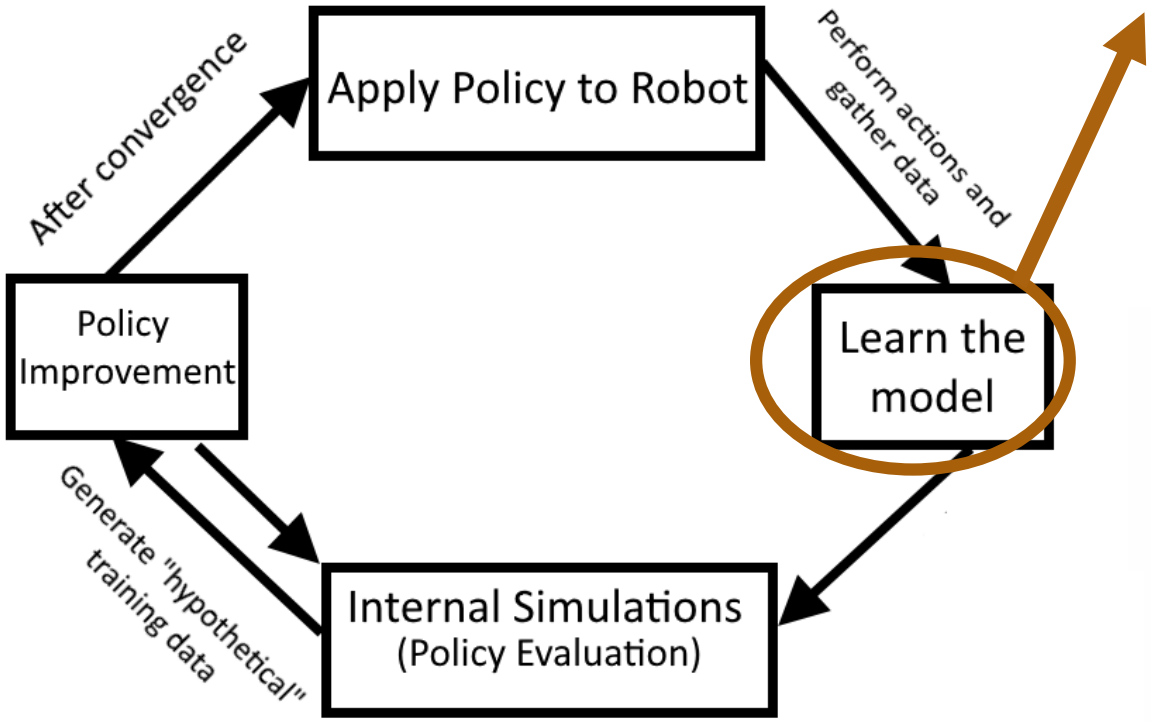
1. Introduction
2. Background Knowledge
3. Methods
4. Results
5. Discussion
6. Conclusion & What‘s left

PILCO [5]



Model-Based RL Framework. Adapted from [4]

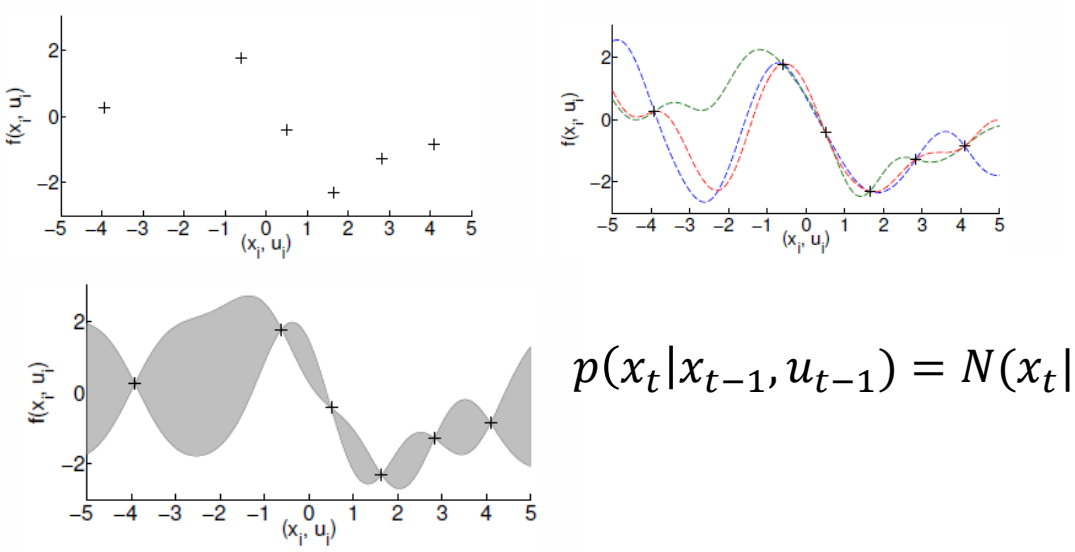
PILCO [5]



Model-Based RL Framework. Adapted from [4]

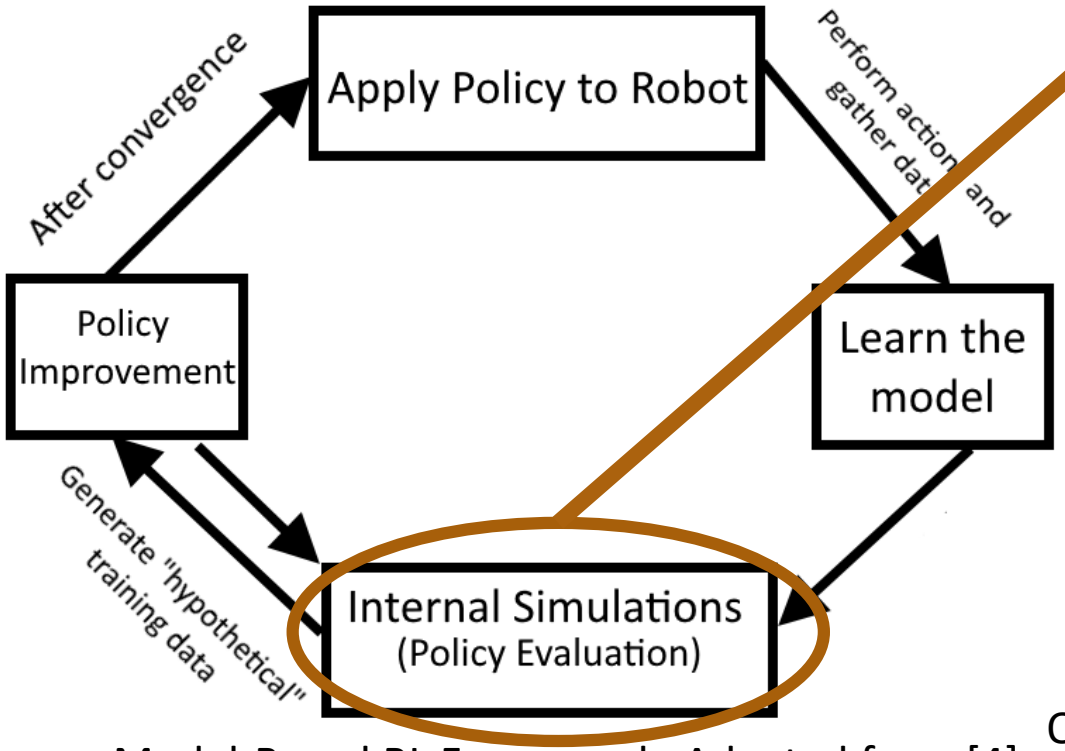
Gaussian Process

Basically:
A probabilistic function approximator



$$p(x_t|x_{t-1}, u_{t-1}) = N(x_t|\mu_t, \Sigma_t)$$

PILCO [5]



Model-Based RL Framework. Adapted from [4]

Policy Evaluation

Calculate the cost (i.e. evaluate the performance of the policy)

$$J^{\pi}(\theta) = \sum_{t=0}^{T} \mathbb{E}_{x_t}[c(x_t)]$$

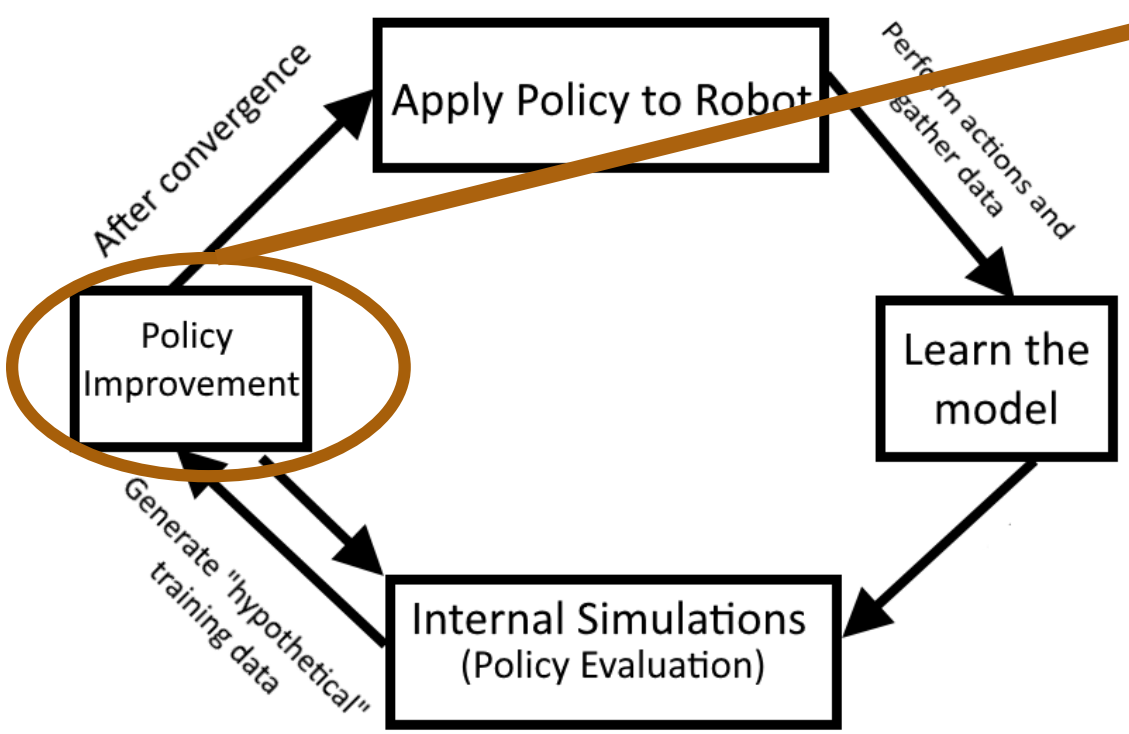
Objective

Policy

Parameters of the policy

Cost function

PILCO [5]



Model-Based RL Framework. Adapted from [4]

Policy Improvement

How does $J^{\pi}(\theta)$ behave when θ is changed

1. Calculate gradient $\frac{dJ^{\pi}(\theta)}{d\theta}$

2. Update θ to minimize $J^{\pi}(\theta)$

(gradient-based optimization)

M-T PS [6]

- Similar to PILCO but capable to generalize across different tasks
  - Policy is a function of the state x_t, its parameters θ *and the task* η
  - Cost function incorporates the task η
- In other words: multi-task extension of PILCO

- Constraints:
  - Dynamics must be stationary
  - Environment must be the same across tasks

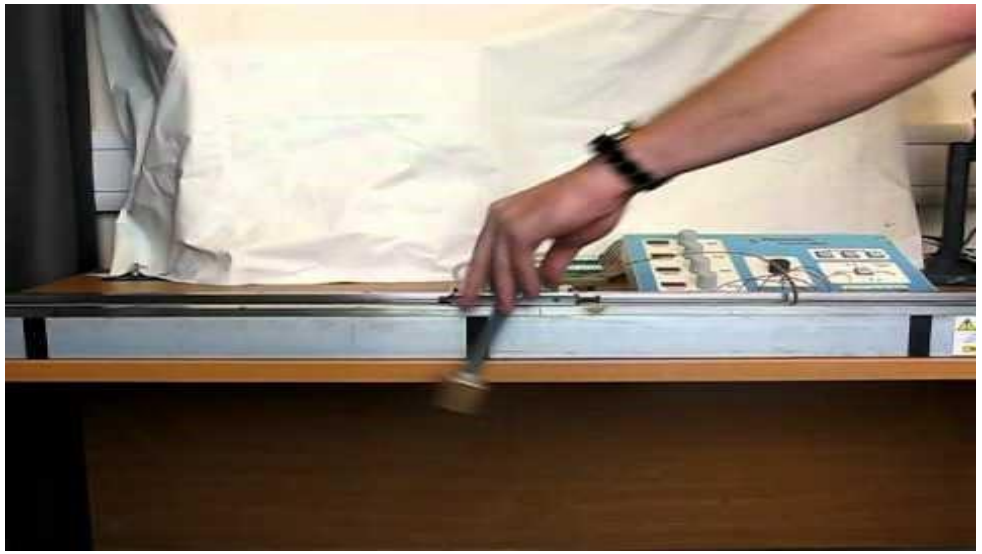
Outline

1. Introduction
2. Background Knowledge
3. Methods
4. Results
5. Discussion
6. Conclusion & What‘s left

PILCO [5]

Real time training [10]

Result Comparison



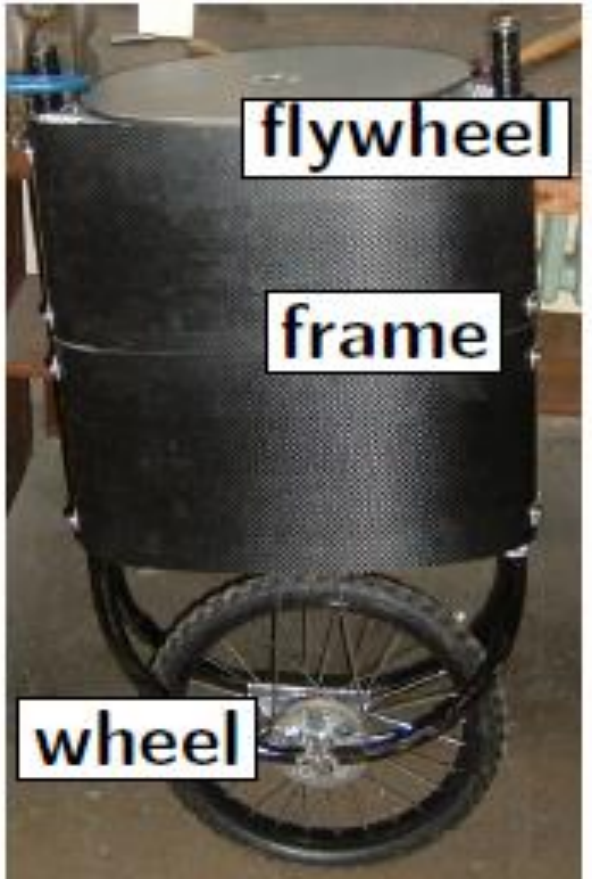
Real time training [10]

- PILCO: ≤ 10 trials, 17.5 seconds
- MT-PS: 20 trials, around 70 seconds
- AGP-iLQR [7]: 4-7 Episodes, 20-42 seconds

- Best result for Model-Free RL methods:
  19 episodes [8]

What else can MB-RL do?

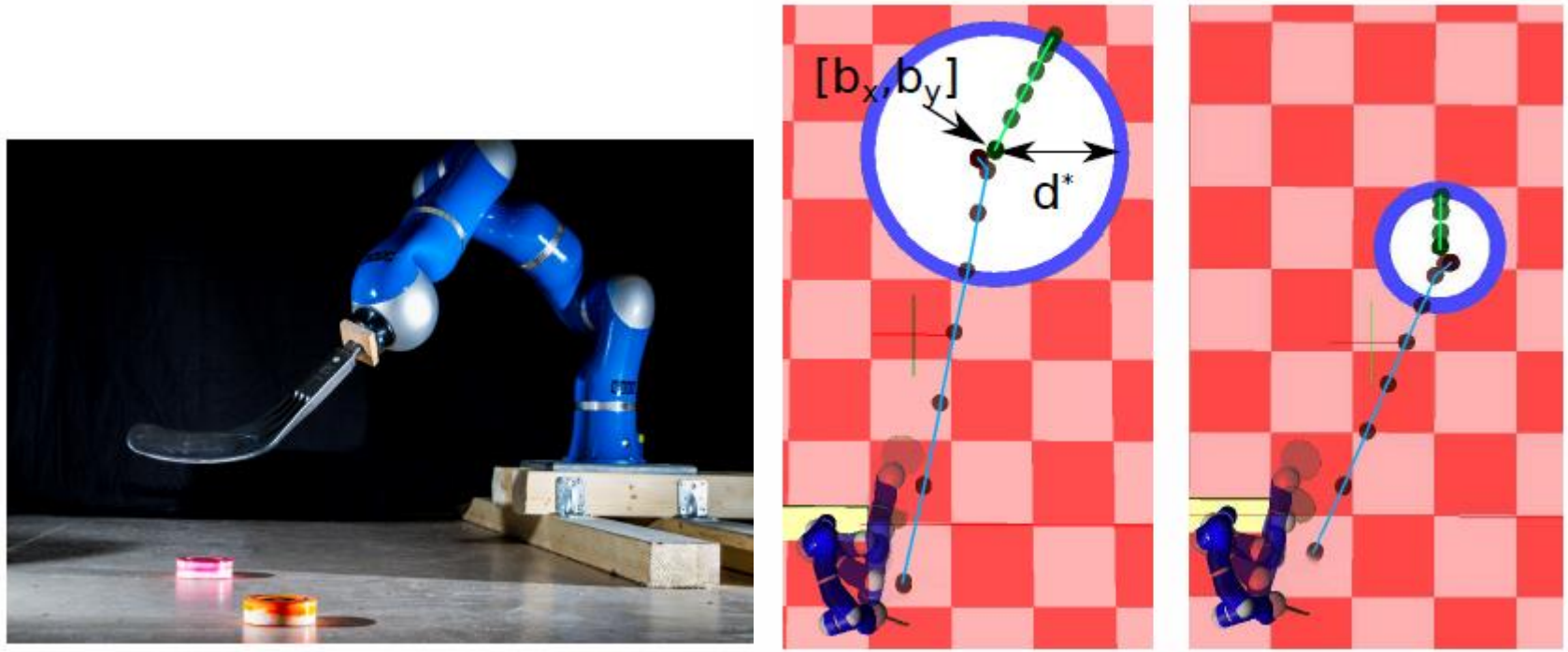
- Ride a unicycle in around 20 seconds [5]



Robotic unicycle [5]

What else can MB-RL do?

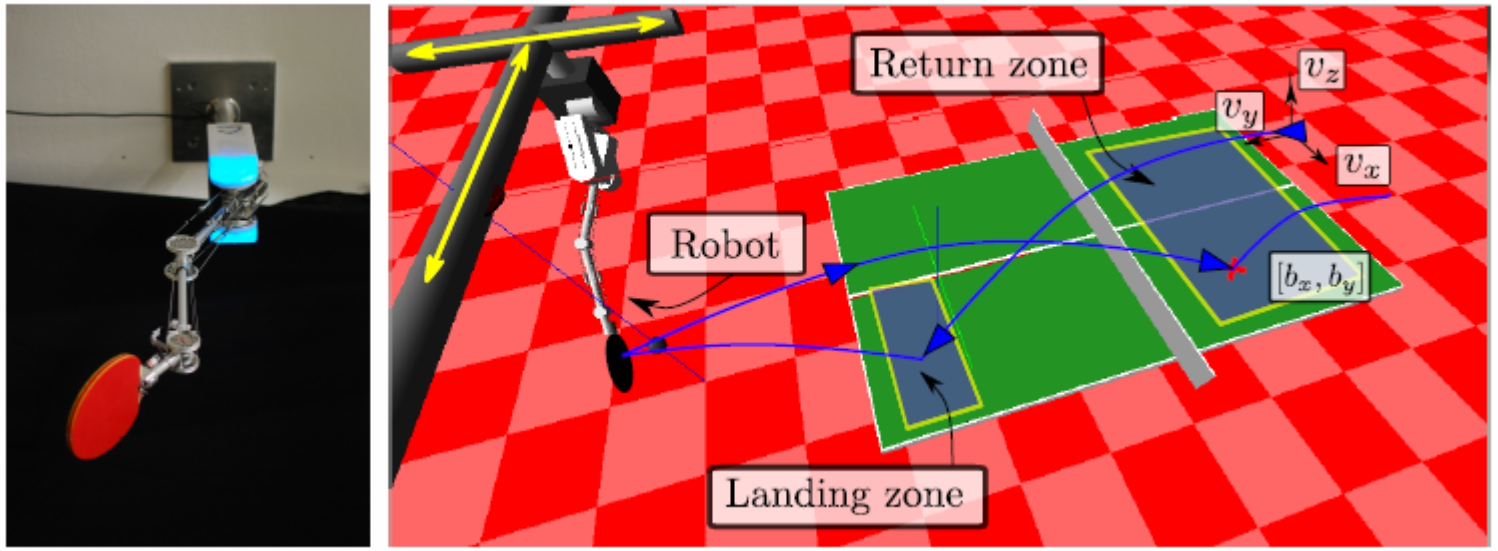
- Learn hockey



KUKA; Hockey Task [11]

What else can MB-RL do?

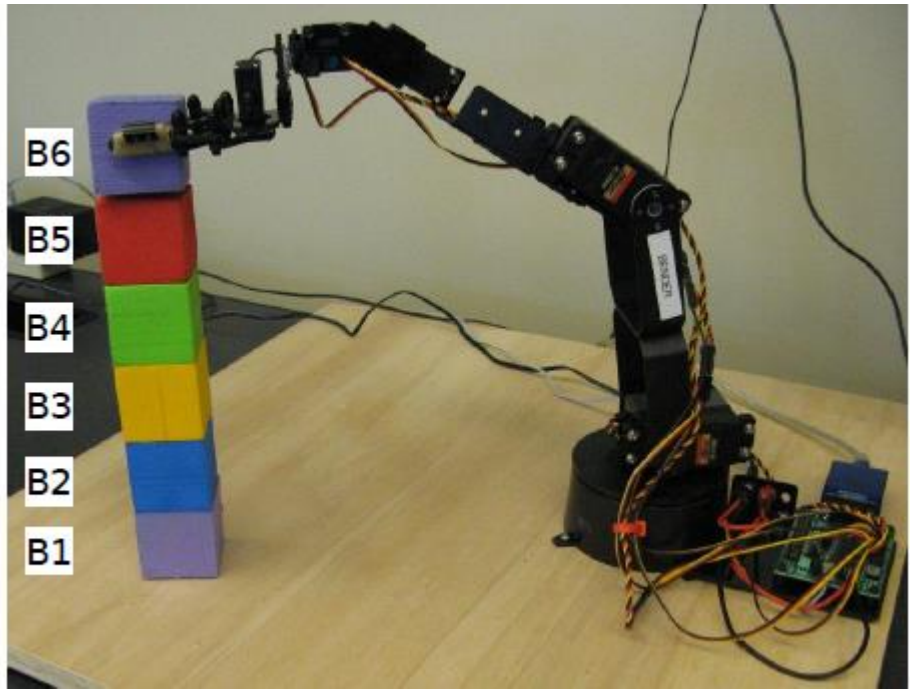
- Play tabletennis



Tabletennis arm and task [11]

What else can MB-RL do?

- Manipulation



Manipulation task [6]; Robot by Lynxmotion [12]

Outline

1. Introduction
2. Background Knowledge
3. Methods
4. Results
5. Discussion
6. Conclusion & What‘s left

So what‘s the catch?

Model-Based vs. Model-Free

	Model-Based	Model-Free
+	Sample Efficient	High Performance
-	Low Performance	Very inefficient

MB vs. MF Comparison [3][4]

Huge problem!

So what‘s the catch?

	Model-Based	Model-Free
+	Sample Efficient	High Performance
-	Low Performance	Very inefficient

MB vs. MF Comparison [3][4]

Also a problem!

Huge problem!

Discussion

- Ideally: Combine MB & MF RL
  - Learn environment‘s dynamics (MB)
    Agent knows nothing
  - Use model to generate data to initialize model-free learner (i.e. a policy, e.g. a neural network)
    Agent is compitent
  - Continue MF learning (policy *fine tuning*)
    Agent becomes an expert

Combination has been tested

Neural Network Dynamics for Model-Based Deep Reinforcement Learning with Model-Free Fine-Tuning

Anusha Nagabandi, Gregory Kahn, Ronald S. Fearing, Sergey Levine
University of California, Berkeley

[3]

- Used MuJoCo [13]

Biological Background

- Model-Based methods use State-Prediction-Errors (SPE) to learn the model
- Model-Free methods use Reward-Prediction-Errors (RPE) to learn the model

- Evidence suggests that the human brain uses SPE and RPE [9]
  - Hinting that the brain is both a model-free and model-based learner

Maybe interesting for those in BAI

Why should we care?

- Need intelligent robots!
- Two ways to train robots:
  - From actual simulation (different problem – transfer learning)
  - Using the real world

Why should we care?

- Simulations might be hard to construct and harder to transfer from

- Real interactions have a better effect (but come at a cost)
- Need to find a way to decrease interactions while maintaining learning

Outline

1. Introduction
2. Background Knowledge
3. Methods
4. Results
5. Discussion
6. Conclusion & What‘s left

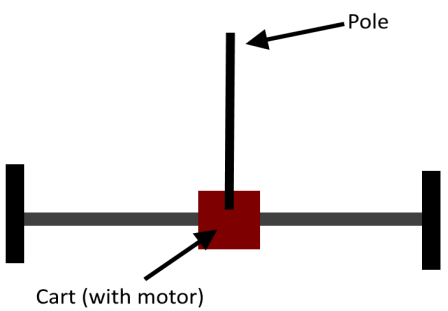
What‘s left for me

- Finish my paper
- Work through all the math (and understand it)
- Try to find more results
  - Sadly: (Published) Results are sparse

Conclusion & Main Take-Away

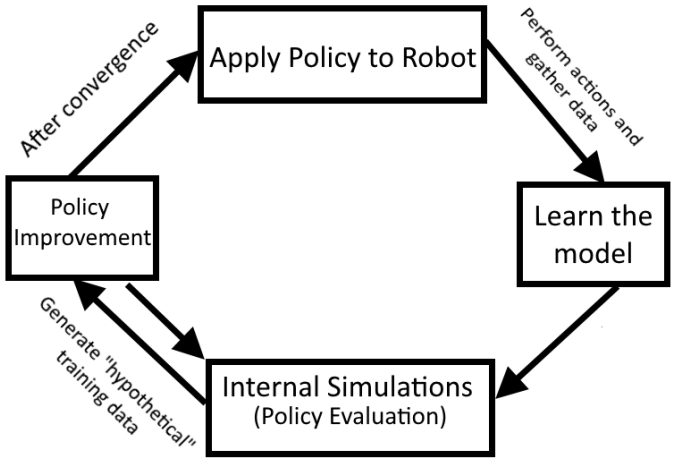
	Model-Based	Model-Free
+	Sample Efficient	High Performance
-	Low Performance	Very inefficient

MB vs. MF Comparison [3][4]



Cart-Pole Setting. By author; as described in [5]

- Quick way to see if you are on the right track
- If your algorithm can‘t handle CP, it won‘t handle anything



Model-Based RL Framework. Adapted from [4]

Ideally: Combine MB & MF RL

References

[1] Sutton, Richard S., Andrew G. Barto, and Francis Bach. *Reinforcement learning: An introduction*. MIT press, 1998
[2] Hessel, M., Modayil, J., Van Hasselt, H., Schaul, T., Ostrovski, G., Dabney, W., ... & Silver, D. (2017). Rainbow: Combining improvements in deep reinforcement learning. *arXiv preprint arXiv:1710.02298*.
[3] Nagabandi, A., Kahn, G., Fearing, R. S., & Levine, S. (2018, May). Neural network dynamics for model-based deep reinforcement learning with model-free fine-tuning. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*(pp. 7559-7566). IEEE
[4] Deisenroth, M. P., Neumann, G., & Peters, J. (2013). A survey on policy search for robotics. *Foundations and Trends® in Robotics*, *2*(1–2), 1-142.
[5] Deisenroth, M., & Rasmussen, C. E. (2011). PILCO: A model-based and data-efficient approach to policy search. In *Proceedings of the 28th International Conference on machine learning (ICML-11)* (pp. 465-472)
[6] Deisenroth, M. P., Englert, P., Peters, J., & Fox, D. (2013). Multi-task policy search. *arXiv preprint arXiv:1307.0813*.
[7] Boedecker, J., Springenberg, J. T., Wülfing, J., & Riedmiller, M. (2014, December). Approximate real-time optimal control based on sparse gaussian process models. In *Adaptive Dynamic Programming and Reinforcement Learning (ADPRL), 2014 IEEE Symposium on* (pp. 1-8). IEEE.
[8] Nagendra, S., Podila, N., Ugarakhod, R., & George, K. (2017, September). Comparison of reinforcement learning algorithms applied to the cart-pole problem. In *Advances in Computing, Communications and Informatics (ICACCI), 2017 International Conference on* (pp. 26-32). IEEE.
[9] Gläscher, J., Daw, N., Dayan, P., & O'Doherty, J. P. (2010). States versus rewards: dissociable neural prediction error signals underlying model-based and model-free reinforcement learning. *Neuron*, *66*(4), 585-595.
[10] “Cart-Pole Swing-up” https://www.youtube.com/watch?v=XiigTGKZfks – Accessed 01.11.2018, 15:41
[11] Kupcsik, A., Deisenroth, M. P., Peters, J., Loh, A. P., Vadakkepat, P., & Neumann, G. (2017). Model-based contextual policy search for data-efficient generalization of robot skills. *Artificial Intelligence*, *247*, 415-439.
[12] Lynxmotion Robot: http://www.lynxmotion.com/ Accessed 02.11.2018, 15:12
[13] Todorov, E., Erez, T., & Tassa, Y. (2012, October). Mujoco: A physics engine for model-based control. In *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on* (pp. 5026-5033). IEEE.

Thanks for listening! Any Questions?