Learning to Adapt in Dynamic, Real-World Environments Through Meta-Reinforcement Learning

Anusha Nagabandi*, Ignasi Clavera*
Simin Liu, Ron S. Fearing , Pieter Abbeel, Sergey Levine, Chelsea Finn

Problem

Our Deep-RL agents mostly operate in the regime where they are very good at succeeding in specific settings, but fail in the face of any changes or new settings at run time.

- Need **adaptation**

Large function approximators cannot naively be updated online, using small amount of data

- Need **meta-learning** to allow for fast adaptation

We want to adapt to dynamics changes, and also, collecting training data in expensive

- Use a **sample-efficient**, model-based reinforcement learning algorithm

Background: Meta-Learning

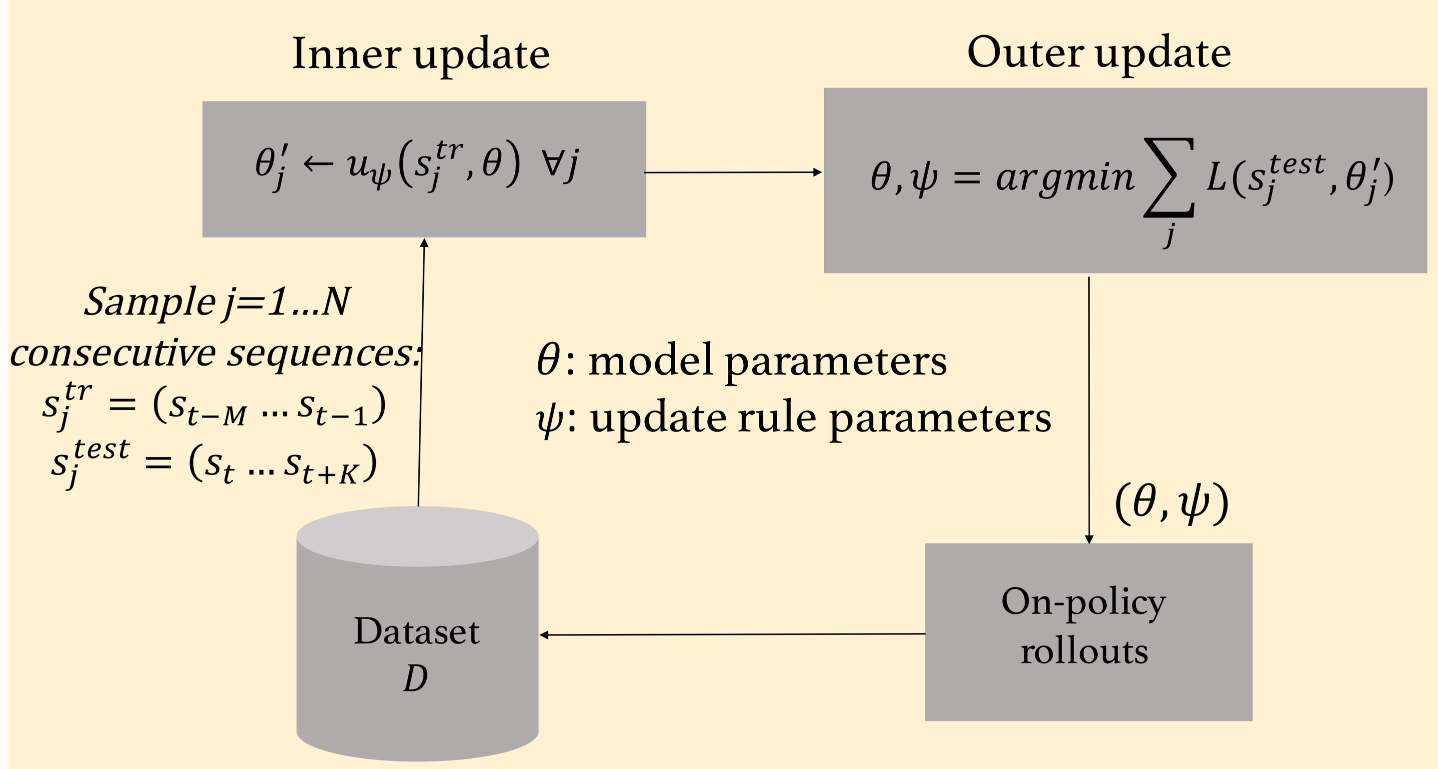
Goal: find optimal parameters (θ, ψ) such that updated model parameters θ' produced by the update rule u_ψ optimize our objective, across tasks T:

$$\min_{\theta,\psi} \sum_{T} L(D_T^{test}, \theta') \quad s.t. \quad \theta' = u_\psi(D_T^{tr}, \theta)$$

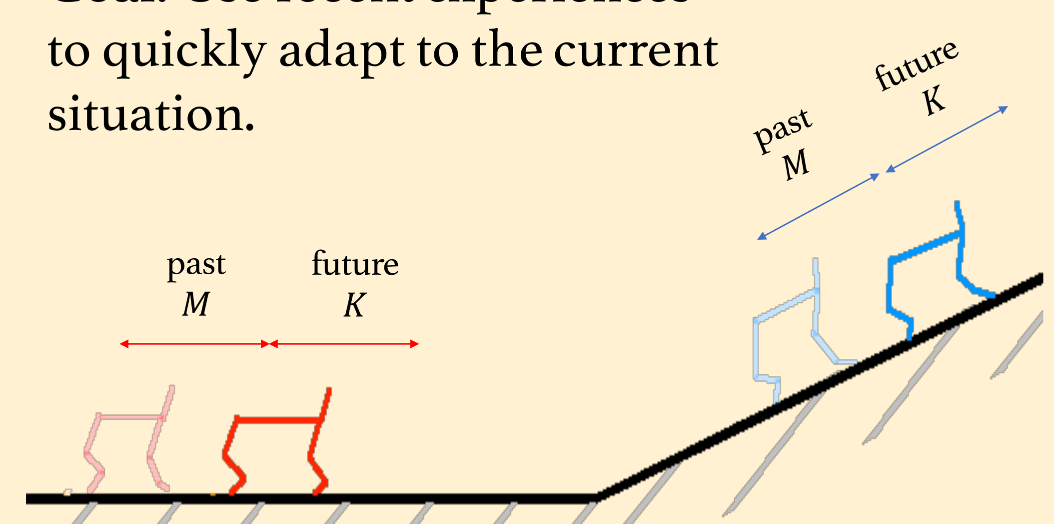
Update rule u uses recent data D_T^{tr} from a given task to adapt for other data D_T^{test} from the same task.

- **Gradient-based meta-learning:** $u_\psi = \theta - \alpha \nabla_\theta L(D_T^{tr}, \theta)$
- **Recurrence-based meta-learning:** u_ψ: RNN that takes in D_T^{tr} sequentially

Train time: Learning to Adapt

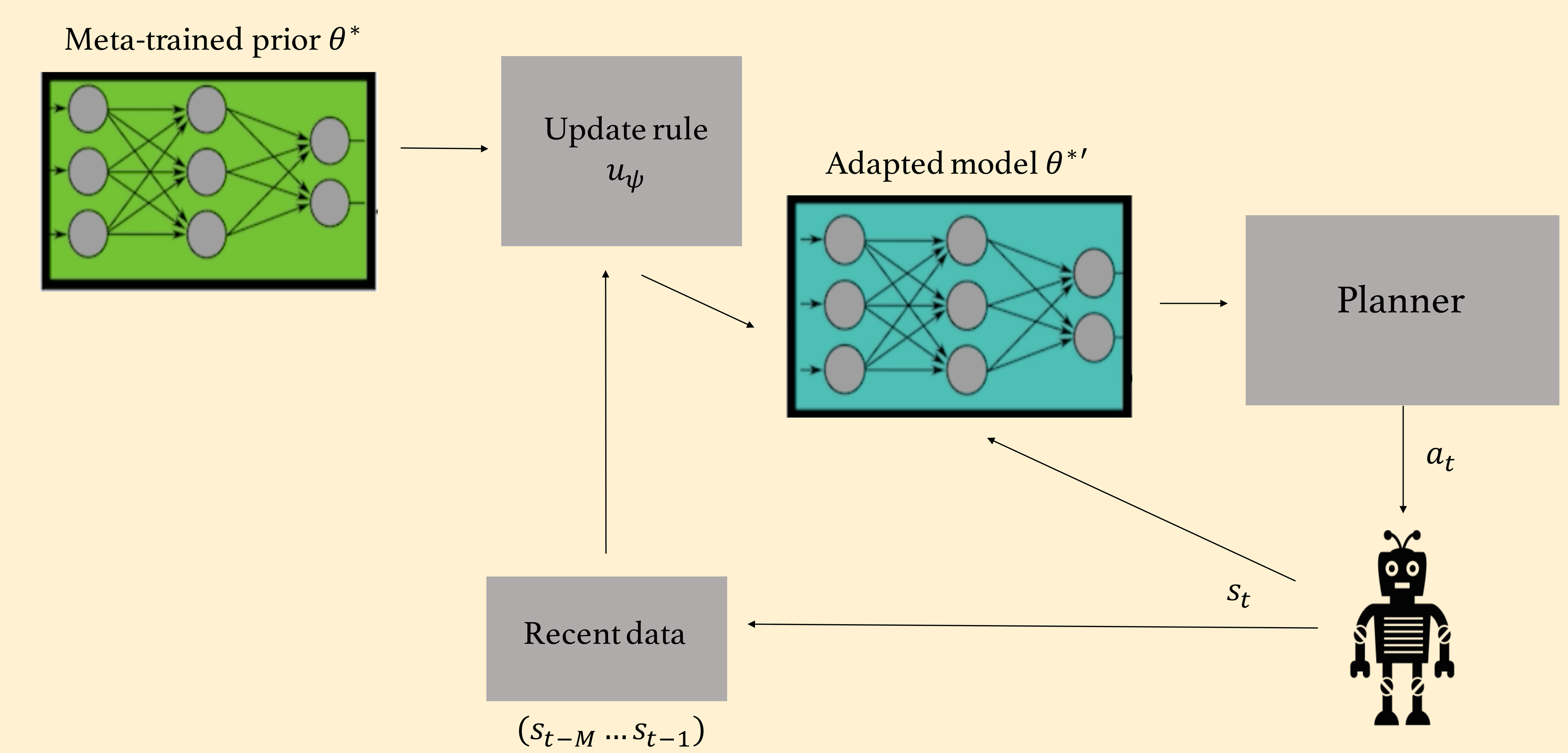


Goal: Use recent experiences to quickly adapt to the current situation.

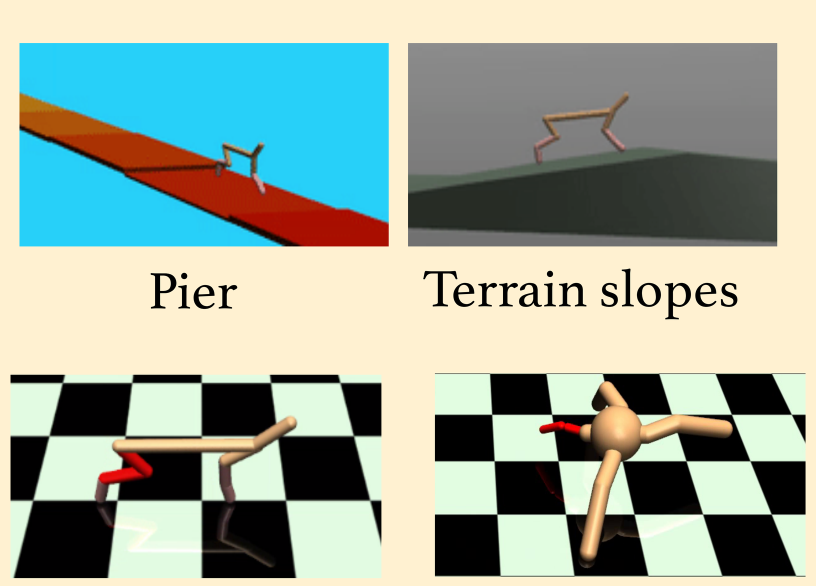


We construct "tasks" by using windows of time:
Recent experience can be informative of current/near-future settings
(i.e., dynamics, environmental perturbations, goals, etc.)

Test time: Model-Based Meta-RL

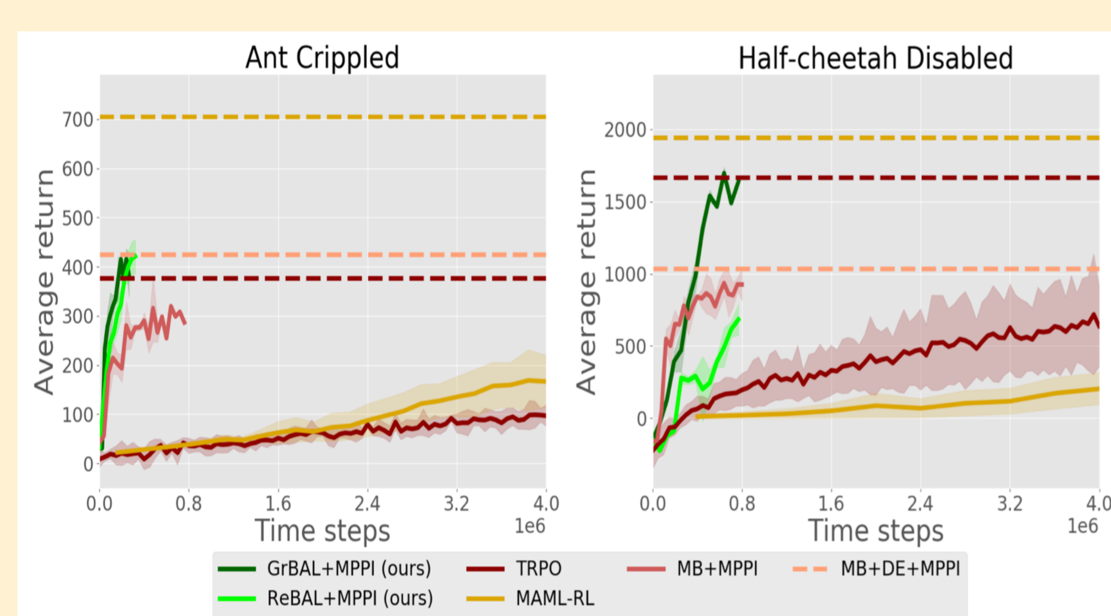


Simulation Results



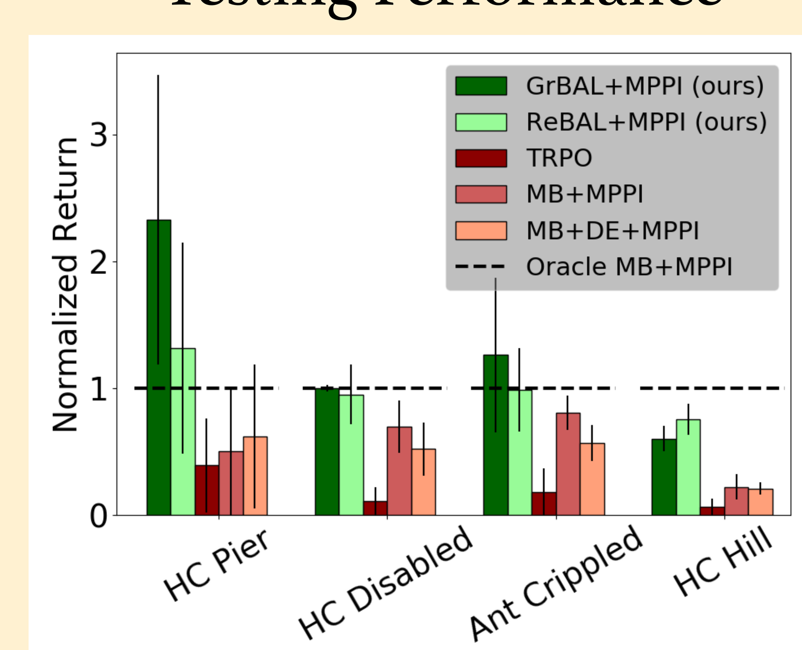
Pier | Terrain slopes | Disabled | Crippled

Meta-training Sample Efficiency



Sample efficiency: requires 1000x less meta-training data than the model-free methods, and achieves higher performance than the model-based methods.

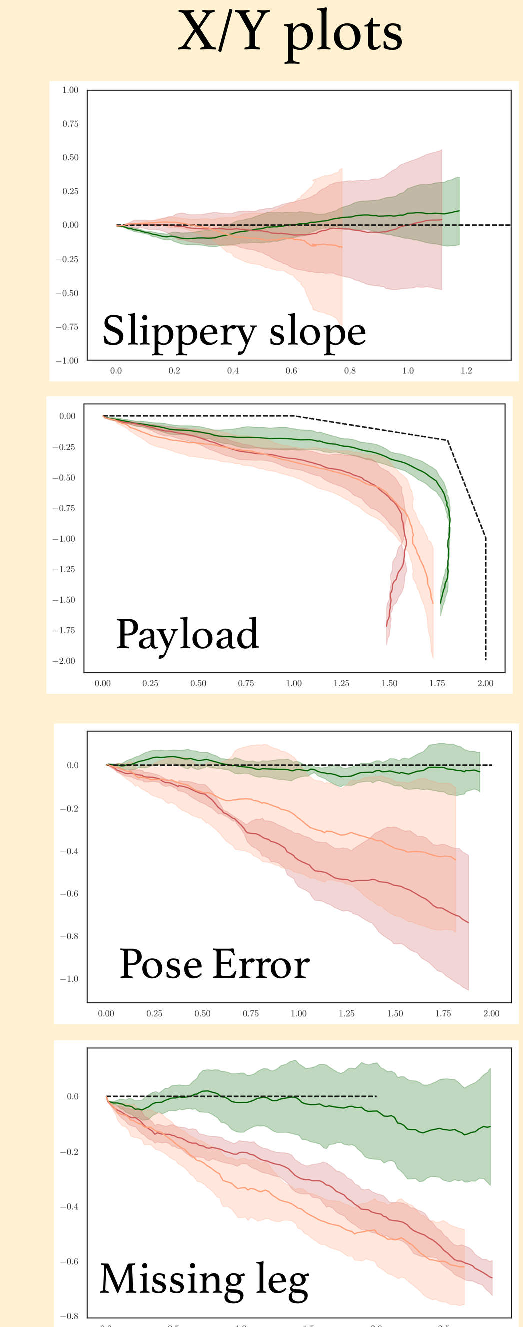
Testing Performance



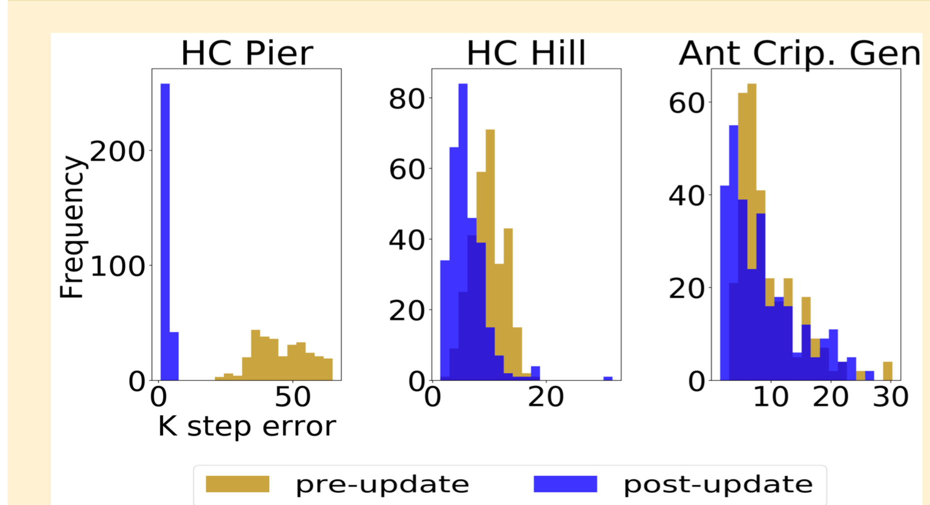
Comparison: Outperforms previous model-free (TRPO), model-based (MB), and adaptive (MB+DE) model-based methods.

Real-world Results

Tasks | X/Y plots

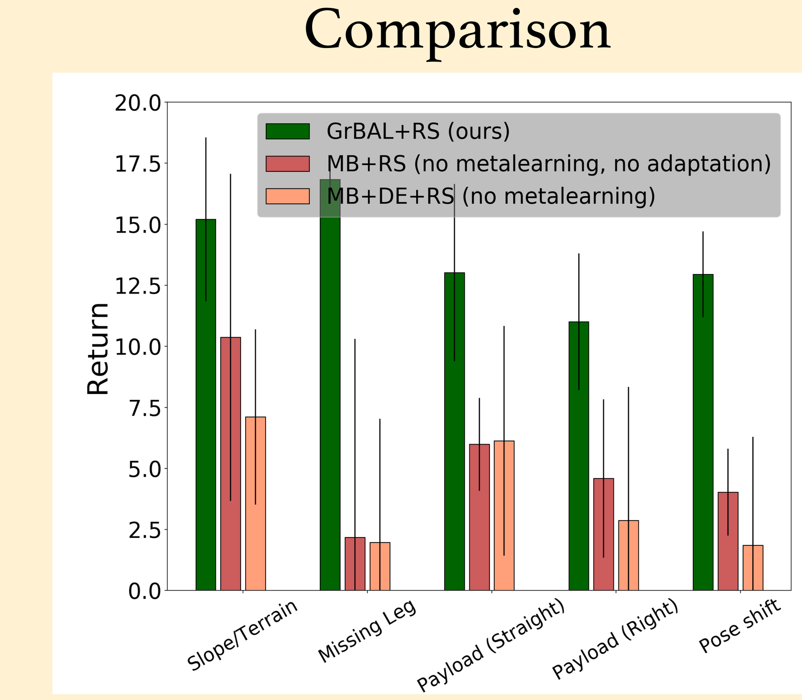


Model Adaptation



Model accuracy: Model prediction errors are reduced after model adaptation.

Comparison



Takeaways

- Use of meta-learning to enable **fast adaptation** (i.e. k-shot) of large function approximators
- **Sample-efficient** meta-training via our model-based formulation
- **Local fine-tuning** of a prior precludes need for a globally accurate model and allows for online adaptation to changes