Model-based Meta Reinforcement Learning using Graph Structured Surrogate Models and Amortized Policy Search

Qi Wang & Herke van Hoof

Amsterdam Machine Learning Lab, UvA

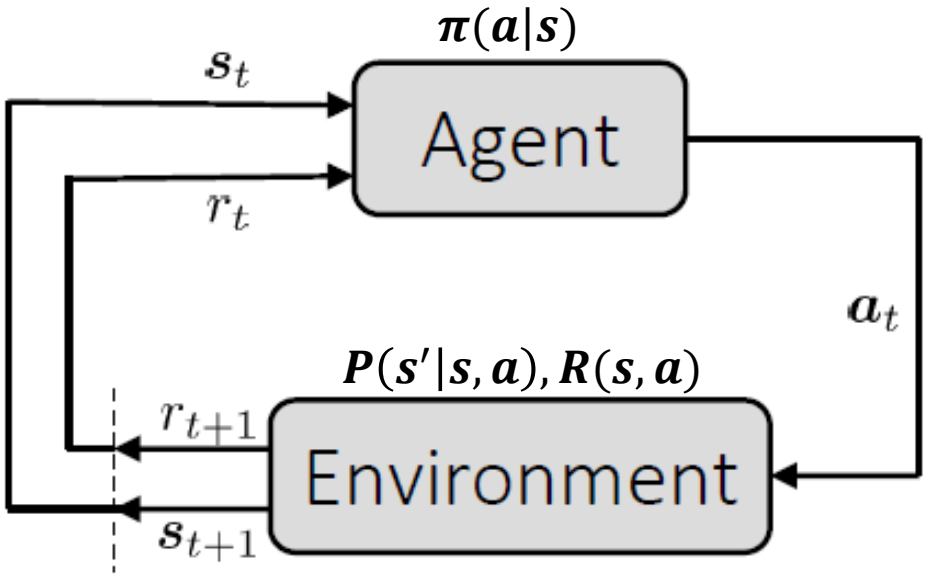
UNIVERSITY OF AMSTERDAM

AMLAB Amsterdam Machine Learning Lab

Research Motivations

Sequential Decision-Making in MDP =.=

- ***Learn policies from Feedbacks*** → **RL** to solve **MDPs**
- *Crucial elements in* ***MDPs*** $\boldsymbol{M =< S, A, P, R, \gamma >}$**:**
  - ✓State Space $\mathbf{s} \in \boldsymbol{S}$
  - ✓Action Space $\mathbf{a} \in \boldsymbol{A}$
  - ✓Policy Function $\boldsymbol{\pi(a|s)}$
  - ✓Transition Dynamics $\boldsymbol{P(s'|s,a)}$
  - ✓Reward Function $\boldsymbol{R(s,a)}$
  - ✓Step Reward Discount $\boldsymbol{\gamma}$

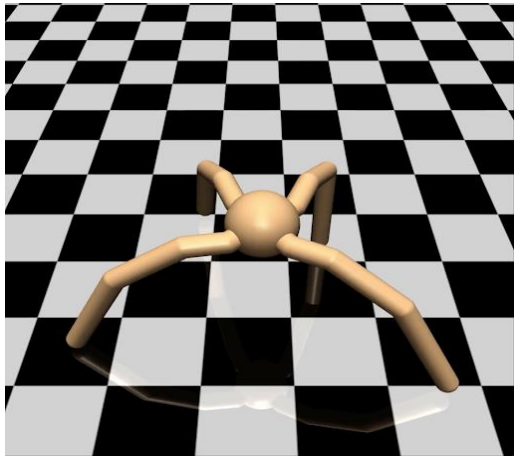


Learning Objective in MDPs:

$$\max_{\pi} \; J(\pi) = \mathbb{E}_{\mu_0,P,\pi}[\sum_{t=0}^{H-1} \gamma^t r_t]$$

RL in Real-Life is Really Challenging! ☹

- ***Data efficiency*** makes ***Model-free RL*** Algorithms prohibitively expensive in practice~
- ***Effective Skill Transfer*** is a long-standing bottleneck in in RL algorithms' deployment, *e.g.* ***Sim2Real (Dynamics mismatch)***~

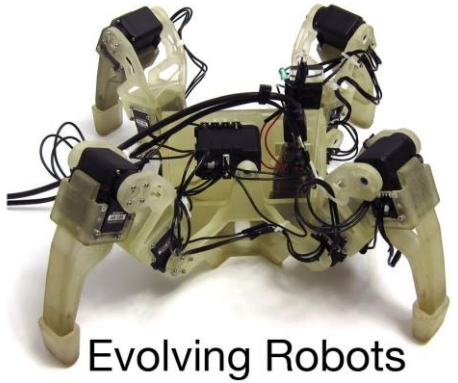


$p_{sim}(s', r|[s, a]) \neq p_{real}(s', r|[s, a])$

D_{sim}

Expensive Calibration

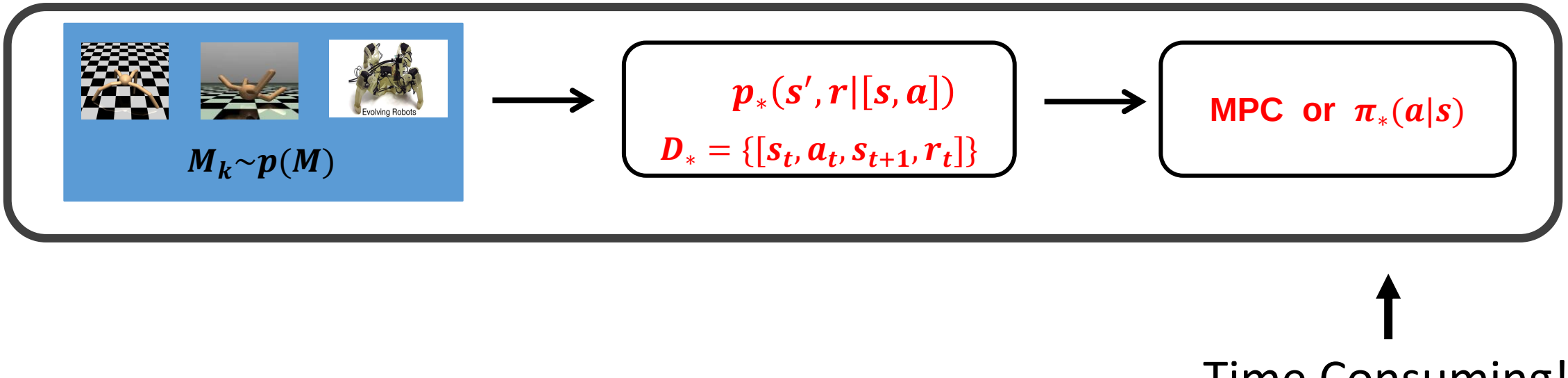
$\pi_{sim}(a|s)$ → $\pi_{real}(a|s)$



$D_{real} = \{[s_t, a_t, s_{t+1}, r_t]\}$

Model-based Meta RL Might Make Sense! ☺

- RL in *a collection of tasks* $\boldsymbol{M_k} =< \boldsymbol{S}, \boldsymbol{A}, \boldsymbol{P_k}, \boldsymbol{R_k}, \boldsymbol{\gamma} >,\ \boldsymbol{M_k} \sim \boldsymbol{p}(\boldsymbol{M})$:
  - ✓ Learn to adapt to a New Environment $\boldsymbol{M_*}$ within a few shots of interactions.
  - ✓ Execute time-consuming MPC or retrain policies in an approximate MDP $\boldsymbol{p_*}(\boldsymbol{s'}, \boldsymbol{r} | [\boldsymbol{s}, \boldsymbol{a}])$.



Proposed Method & Contributions

A New Optimization Framework of MBMRL

- Meta Dynamics Model (GSSM):
  - ✓ System identification with L.V.s
    → Task representation
  - ✓ Boost NPs with Message Passing
    → Minimize model discrepancy

- Amortized Policy Search (APS):
  - ✓ Amortize Task-specific policies
    → Policy fast adaptations (no gradient updates)

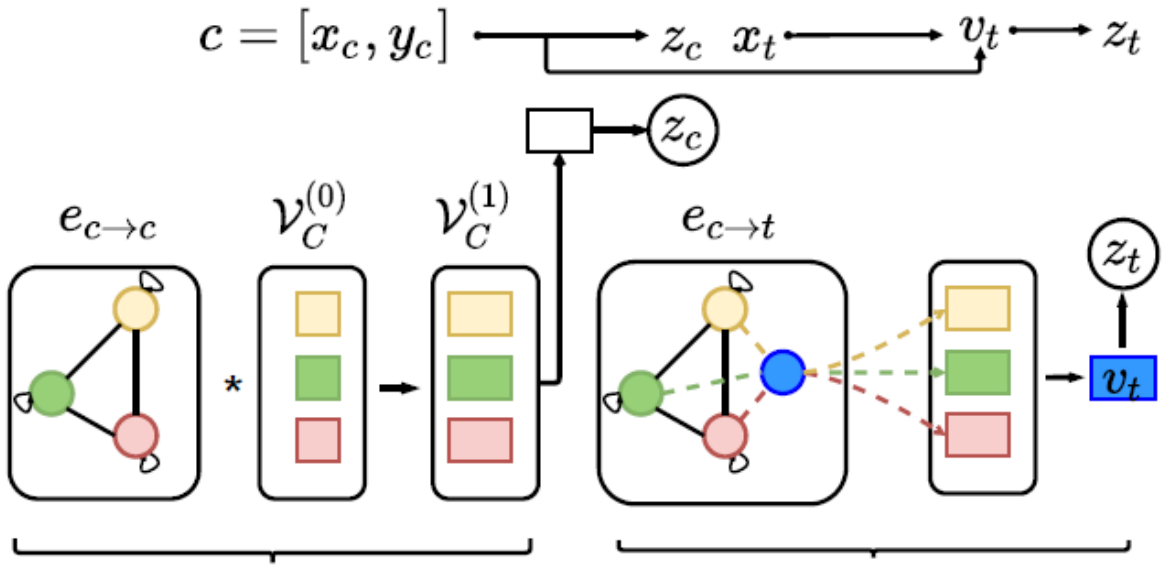
$$\max_{\theta} \mathbb{E}_{\mathcal{M}\sim p(M)} \mathbb{E}_{([s,a],s')\sim\mathcal{M}} \ln\left[p_{\theta_{\mathcal{M}}}(s'|[s,a])\right] \quad \text{s.t. } p_{\theta_{\mathcal{M}}} = u(\theta, \mathcal{D}^{\text{tr}}_{\mathcal{M}})$$

$$\max_{\varphi} \mathbb{E}_{\mathcal{M}\sim p(M)} \mathbb{E}_{\substack{s'\sim p_{\theta_{\mathcal{M}}}(s'|[s,a]) \\ a\sim\pi_{\varphi_{\mathcal{M}}}}} \left[\sum_{t=0}^{H-1} \gamma^t r_{\mathcal{M}(s_t,a_t)}\right] \quad \text{s.t. } \pi_{\varphi_{\mathcal{M}}} = v(\varphi, \mathcal{E}^{\text{tr}}_{\mathcal{M}})$$

Our Objective

- ☐ Graph Structured Surrogate Model
  $p_\theta(s_{t+1}|[s_t, a_t], z_t)$
- ☐ Amortized Policy Search
  $\pi_\varphi(a|s, z_C)$

Training Framework: *Unify two independent phases* together using L.V.s.
This is from a variant of *Thompson/Posterior Sampling* perspective.

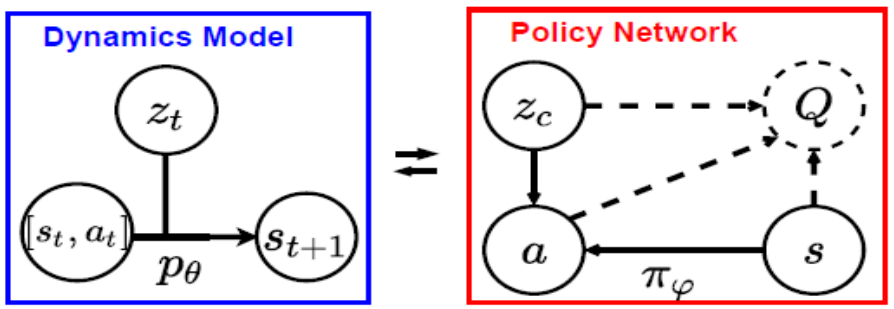


$$x = [s, a], y = s' \text{ or } y = \Delta s$$

Meta Learning Dynamics Models

$$\mathbb{E}_{p(D)}\left[\ln p(y_t|x_t, x_c, y_c)\right] \geq \mathbb{E}_{p(D)}\left[\mathbb{E}_{q_{\phi_1}}\left[\ln p_\theta(y_t|x_t, z_t)\right] - D_{KL}\left[q_{\phi_1}(z_t|x_t, x_c, y_c) \parallel q_{\phi_2}(z_c|x_c, y_c)\right]\right] \quad (6)$$

Meta Model-based Policy Search



$$\nabla_\varphi \mathcal{J}(\varphi) = \mathbb{E}_{\substack{\hat{\mathcal{M}}\sim p(\hat{\mathcal{M}};\theta,\phi) \\ \tau\sim p(\tau|\hat{\mathcal{M}};\varphi,\phi)}}\left[\sum_{t=0}^{T-1} \nabla_\varphi \ln \pi(a_t|[s_t, z_c]) \cdot A_t([s_t, z_c], a_t)\right] \quad (11)$$

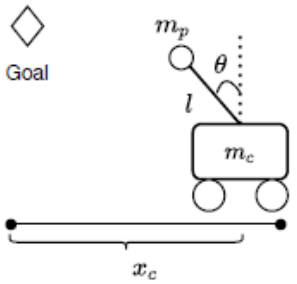
← Dyna-style

Training Paradigm in GSSM in the Background of a Collection of MDPs

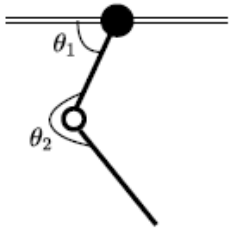
Results & Analysis

Generate a Collection of MDPs

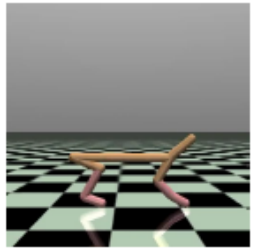
- **Varying Dynamics via Adapting Hyper-Parameters**
  - ✓ Cart-Pole Swing-up/ Acrobot/ Mujoco-Tasks
- **Baselines**
  - ✓ M-DPILCO → BNN-based Meta-RL
  - ✓ MLSM-v0 → Latent Variable-based Meta-RL
  - ✓ MLSM-v1 → Latent Variable-based Meta-RL
  - ✓ L2A → MAML-based Meta-RL
  - ✓ DR-PPO → Train PPO across MDPs by Domain Randomization
  - ✓ PE-PPO → Train PPO with Probabilistic Embedding Contexts $\pi(a|s,z)$



(a) Cart-Pole



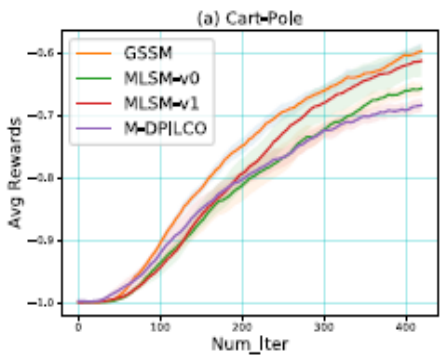
(b) Acrobot

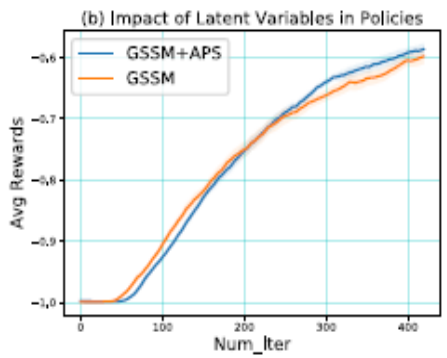

(c) Half-Cheetah

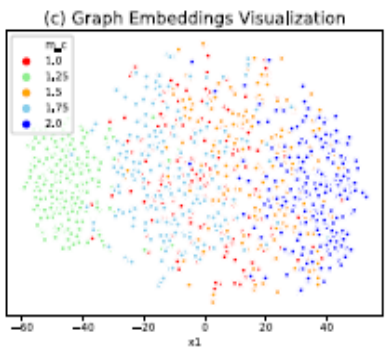

(d) Slim-Humanoid

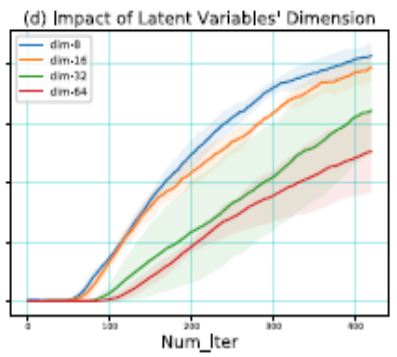
Some Results and Analysis

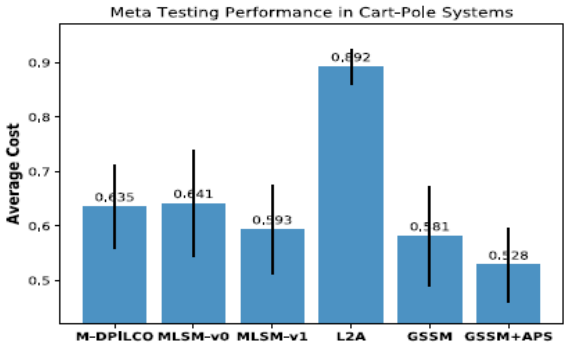
- **Cart-Pole Meta-Training (Left)/Testing Results (Right)**











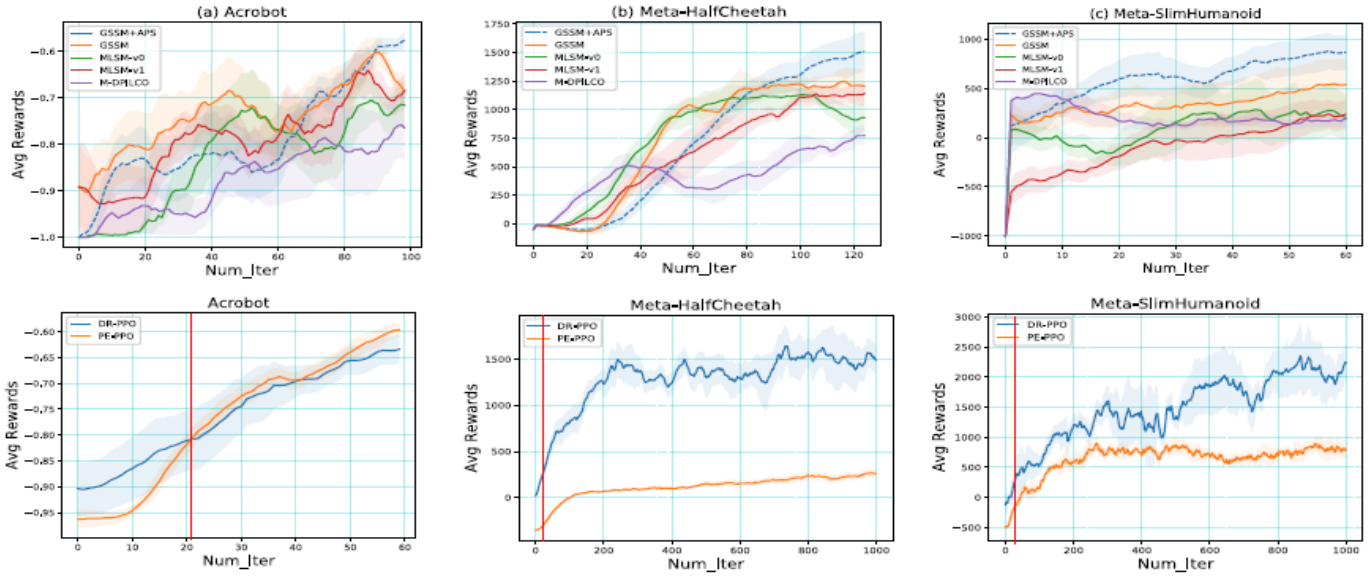
(1) GSSM/MLSM-v1 exhibit best performance in meta-training.

(2) Latent variables are meaningful by varying physics parameters.

(3) GSSM + Amortized Policy Search works best in meta-testing.

Some Results and Analysis

- **Acrobot/Mujoco Learning Curves in Meta-Training Processes**



(1) Our method (first column blue ones) is also effective in Acrobot/Mujoco environments.

(2) Most MBMRL baselines outperform model-free ones with same sample complexity.

Some Results and Analysis

- **Acrobot/Mujoco Performance in Meta-Testing Results**

Table 2. Average Rewards in Meta-testing Tasks using Learned Policy Networks. (For each testing task, 50 episodes are sampled and averaged in rewards. Figures in brackets are standard deviations across testing tasks, with bold ones the best.)

ENV	GSSM+APS	GSSM	M-DPILCO	MLSM-v0	MLSM-v1	L2A
ACROBOT	**-0.478(±0.049)**	-0.506(±0.068)	-0.645(±0.06)	-0.560(±0.064)	-0.524(±0.052)	-0.7775(±0.054)
H-CHEETAH	**1597.4(±200)**	1306.6(±140)	862.0(±280)	827.3(±190)	1226.8(±64)	-17.9(±130)
S-HUMANOID	**1641.8(±170)**	717.1(±130)	596.0(±340)	285.9(±360)	745.6(±150)	124.9(±570)

(1) With non-amortized policy search, GSSM and MLSM-v1 are comparable in cumulative returns.

(2) GSSM + Amortized Policy Search significantly surpasses others.

Limitations & Future Work

- Dyna-style training is sensitive to hyper-parameters, e.g. number of loops/optimization steps in each iteration and so forth.

- The model-based framework is data efficient but computationally intensive in meta training processes.

- Scalability in vision-based control has not been investigated yet.

Thanks for your Listening~