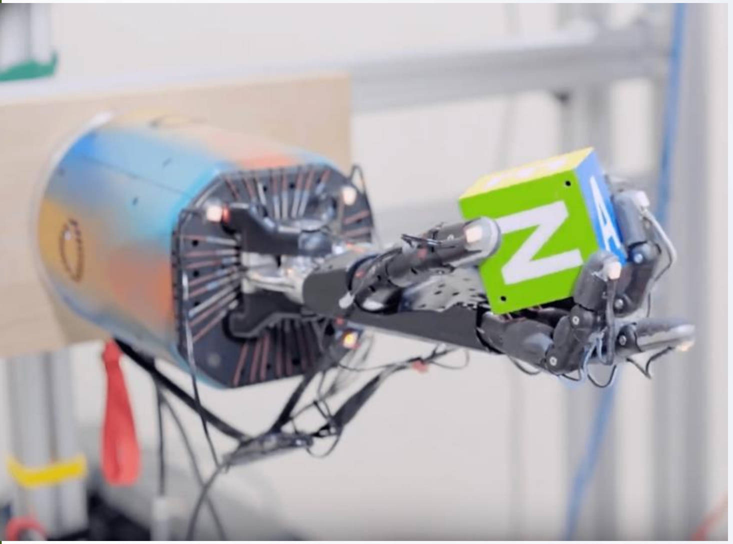


Hybrid, Real-Time Model-Based Reinforcement Policy Learning

Zakariae El Asri

ISIR INSTITUT DES SYSTÈMES INTELLIGENTS ET DE ROBOTIQUE

MLIA

cnrs

SORBONNE UNIVERSITÉ

[RLC 2024] **Physics-Informed Model and Hybrid Planning for Efficient Dyna-Style Reinforcement Learning.**

Zakariae El Asri **Olivier Sigaud** **Nicolas Thome**
Sorbonne Université, CNRS, ISIR, Paris, France

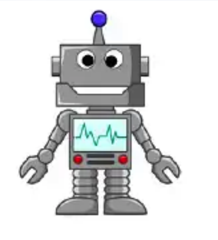
[IROS 2025] **RT-HCP: Dealing with Inference Delays and Sample Efficiency to Learn Directly on Robotic Platforms.**

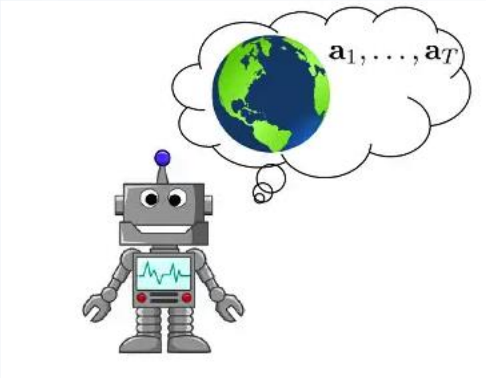
Zakariae El Asri **Ibrahim Laiche** **Clément Rambour** **Olivier Sigaud** **Nicolas Thome**
Sorbonne Université, CNRS, ISIR, Paris, France

Learning controllers with reinforcement learning (RL)

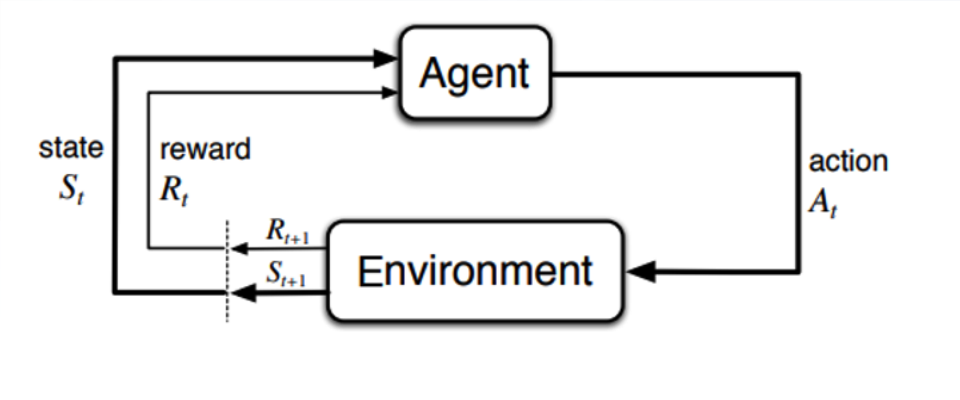
Markov Decision Process (MDP) : $(\boldsymbol{\mathcal{S}}, \boldsymbol{\mathcal{A}}, \boldsymbol{\mathcal{T}}, \boldsymbol{\mathcal{R}})$

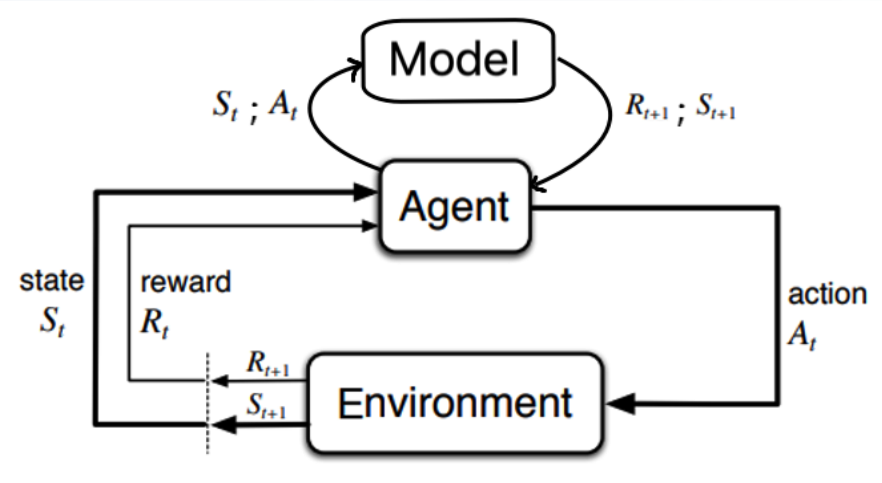
Objective in RL: maximize $\sum_{t=t_0}^{\infty} \gamma^{t-t_0} . r_t$





Model-Free RL Vs Model-Based RL





Model-based vs model-free RL

Model-Free RL

- (+) Asymptotic performance
- (−) Sample efficiency
- (+) Time efficiency

+

Model-Based RL

- (+) Asymptotic performance
- (+) Sample efficiency
- (−) Time efficiency

=

GOAL

- Asymptotic performance
- Sample efficiency
- Time efficiency

- (+) Asymptotic performance
- (+) Sample efficiency
- (+) Time efficiency

Critical in real robotics applications

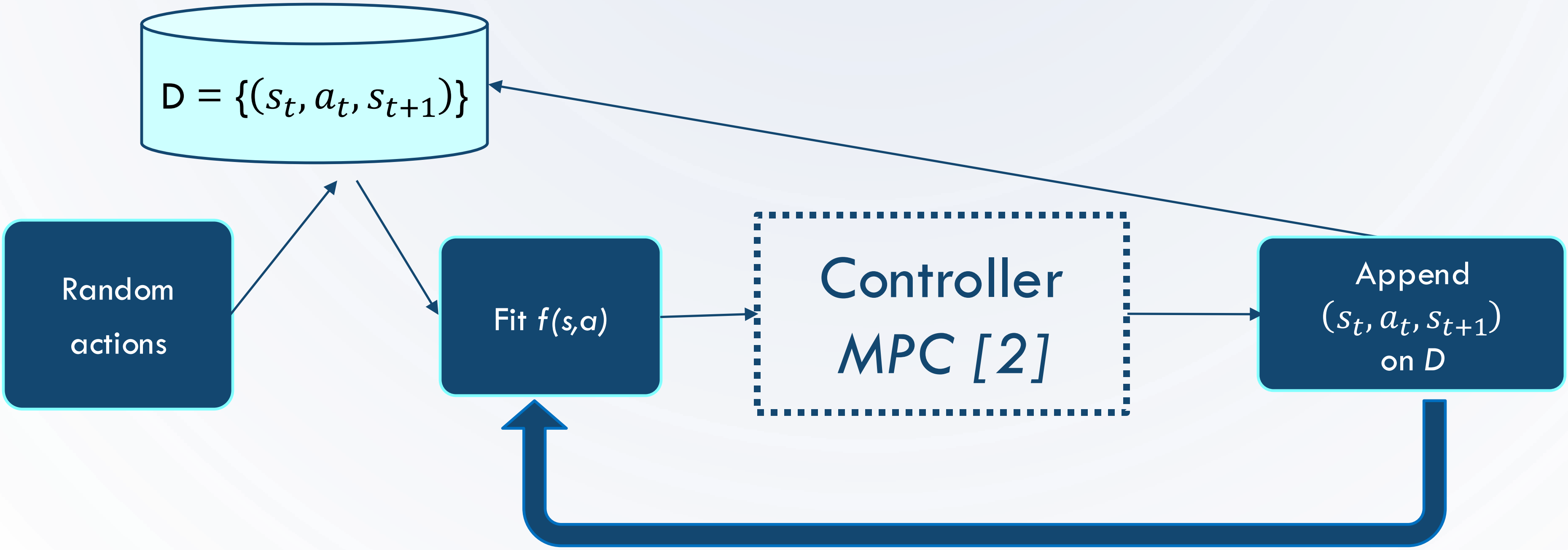
- Interactions costly, complex, harmful
- Need high frequency control

<u>Our motivation:</u>
proper combination MB/MF in RL

Model-based RL, e.g., PETS (2018) [1]

Model-based RL methos rely on:

- Learning a world (dynamical) model $f(s_t, a_t) = s_{t+1}\ or\ p(s_{t+1} | s_t, a_t)$
- Control through $f(s_t, a_t)$ to choose actions

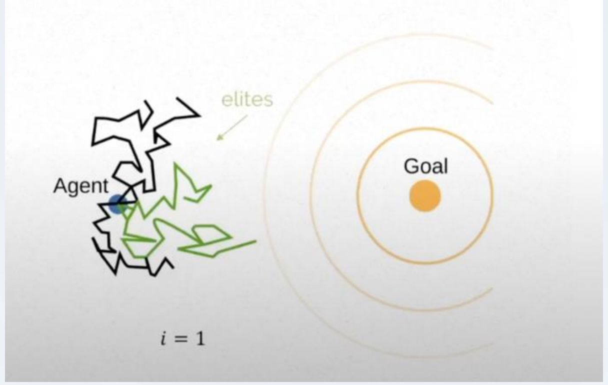


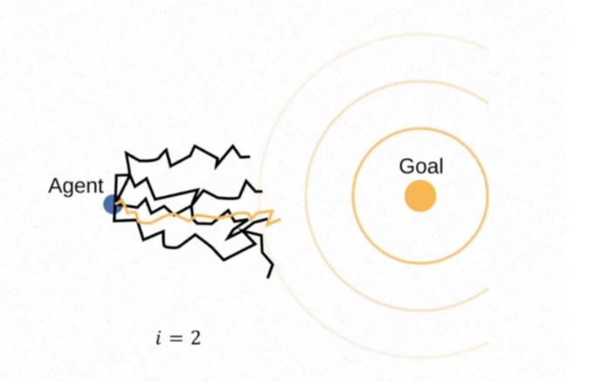
[1] K. Chua, R. Calandra, R. McAllister, and S. Levine. Deep reinforcement learning in a handful of trials using probabilistic dynamics models. NeurIPS 2018.

CEM for sequential decision making [3]

Cross-Entropy Method

$$\mathbf{A}_t^{(H)} = \underset{\mathbf{A}}{\operatorname{argmax}} \sum_{t'=t}^{t+H-1} r(\mathbf{s}_{t'}, \mathbf{a}_{t'})$$





<u>Main CEM hyper-parameters:</u>

- Number of iterations I
- Population size P
- Horizon H

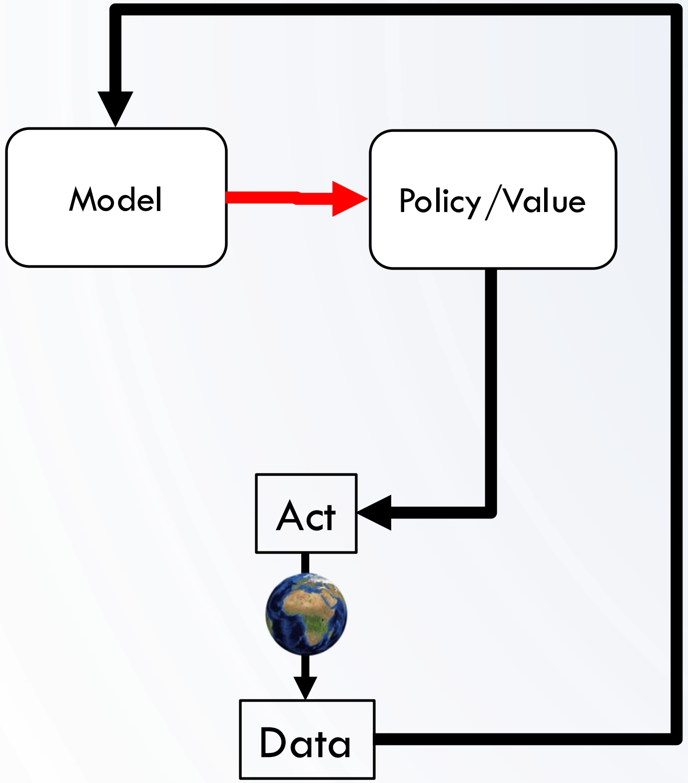
Model-based methods need large H, P *and* I

$\text{Inference time} = \mathrm{f}(\mathrm{H}, \mathrm{P}, \mathrm{I}) \uparrow\uparrow$

[3] R.Y. Rubinstein, D.P. Kroese. The Cross-Entropy Method: A Unified Approach to Combinatorial Optimization, Monte-Carlo Simulation, and Machine Learning, Springer-Verlag, New York, 2004.

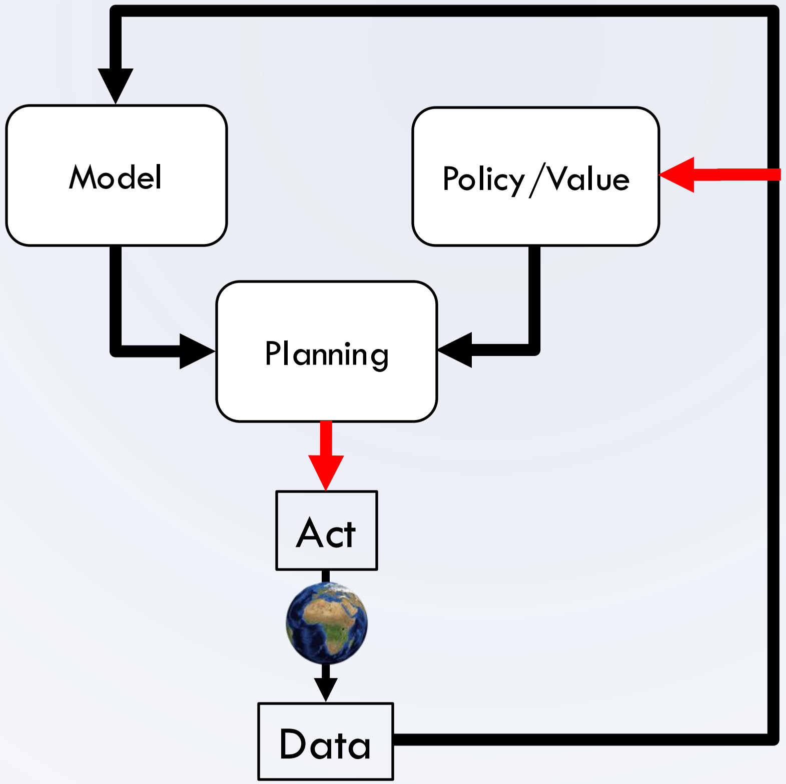
Hybrid MB/MF methods

Dyna style RL (LOOP [4])



- Asymptotic performance
- Sample efficiency
- Time efficiency

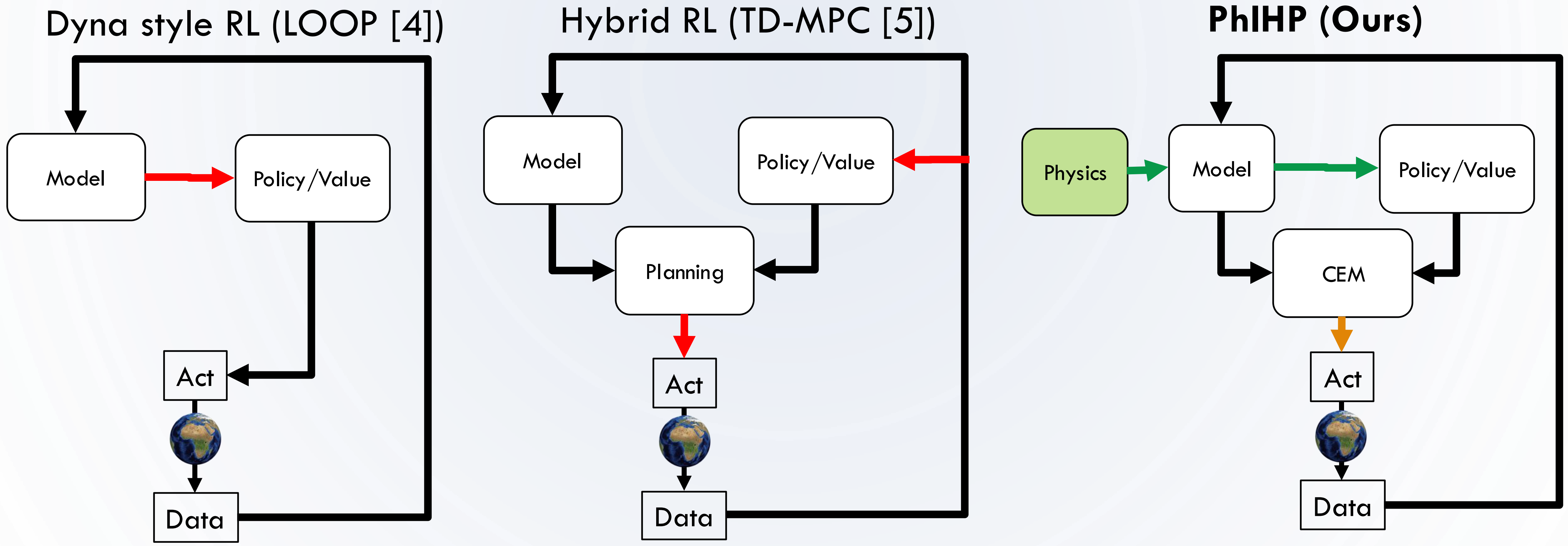
Hybrid RL (TD-MPC [5])



- Asymptotic performance
- Sample efficiency
- Time efficiency

[4] Harshit Sikchi, Wenxuan Zhou, and David Held. Learning off-policy with online planning. CoRL 2022.
[5] Nicklas Hansen, Xiaolong Wang, and Hao Su. Temporal difference learning for model predictive control. ICML, 2022.

Physics-Informed Model and Hybrid Planning (PhIHP)



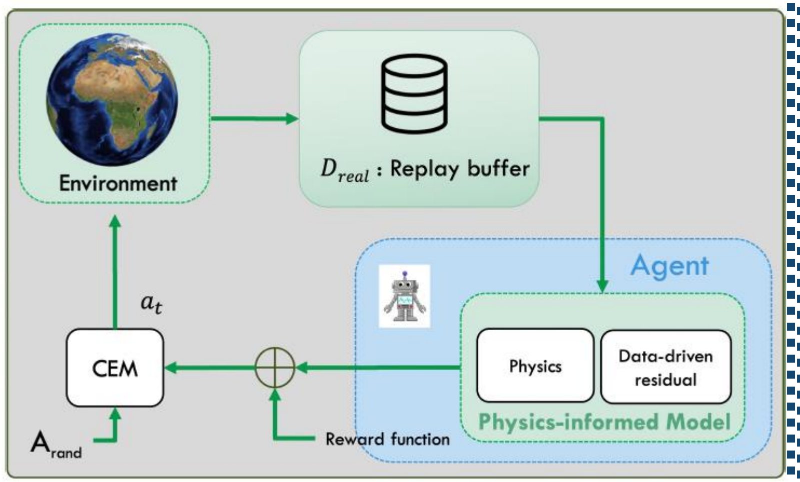
Dyna style RL (LOOP [4])
- − Asymptotic performance
- + Sample efficiency
- + Time efficiency

Hybrid RL (TD-MPC [5])
- + Asymptotic performance
- − Sample efficiency
- − Time efficiency

PhIHP (Ours)
- + Asymptotic performance
- + Sample efficiency
- + Time efficiency

[4] Harshit Sikchi, Wenxuan Zhou, and David Held. Learning off-policy with online planning. CoRL 2022.
[5] Nicklas Hansen, Xiaolong Wang, and Hao Su. Temporal difference learning for model predictive control. ICML, 2022.

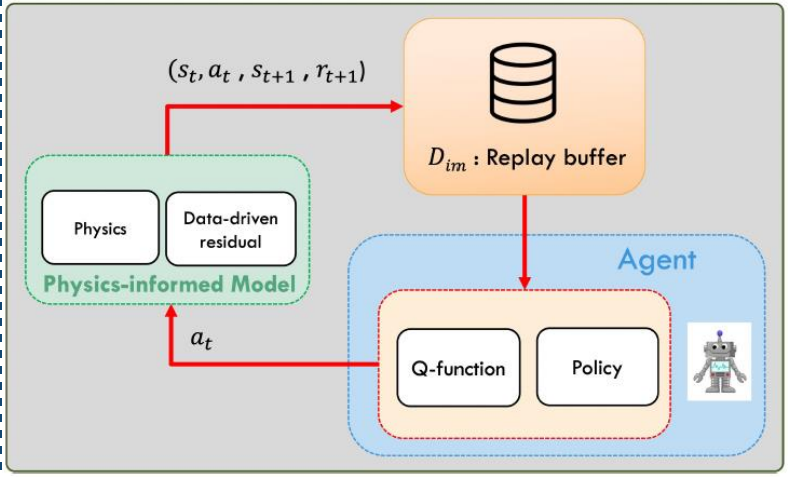
PhIHP pipeline



(a) Learn a physics-informed model

- Sample efficiency
- Reduced bias

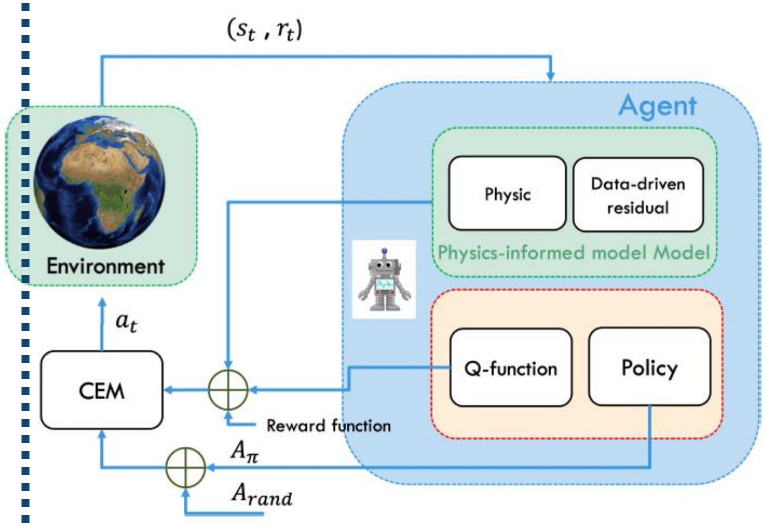
Hybrid model



(b) Learn an actor/critic offline

- Sample efficiency
- Time efficiency

Learning through imagination



(c) Behaviour at inference time

- Sample efficiency
- Time efficiency
- Asymptotic performance

Hybrid TD3/MPC Control

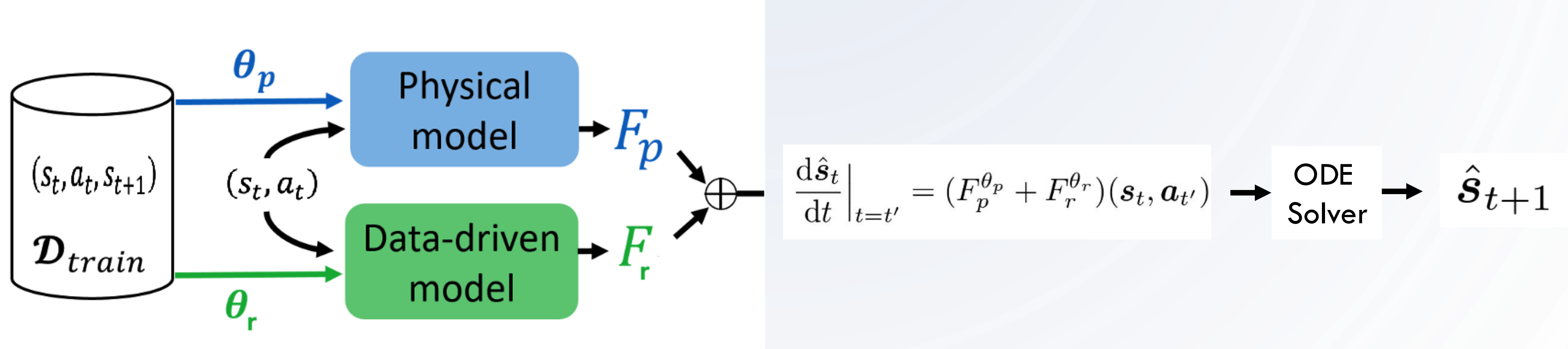
PhIHP: learning a hybrid model

Physics:

An approximate model described as ODE [6]

Learning residual + physical parameters

Training Strategy:



$$\text{Loss} = \sum_i \|f(s_i, a_i) - s_i'\|^2 + \lambda \|F_a\| \quad \text{s.a} \quad f(s_i, a_i) = (F_a + F_p)(s_i, a_i)$$

[6] Yuan Yin, Vincent Le Guen, Jeremie Dona, Emmanuel de Bézenac, Ibrahim Ayed, Nicolas Thome, and Patrick Gallinari. "Augmenting physical models with deep networks for complex dynamics forecasting." ICLR, 2021.

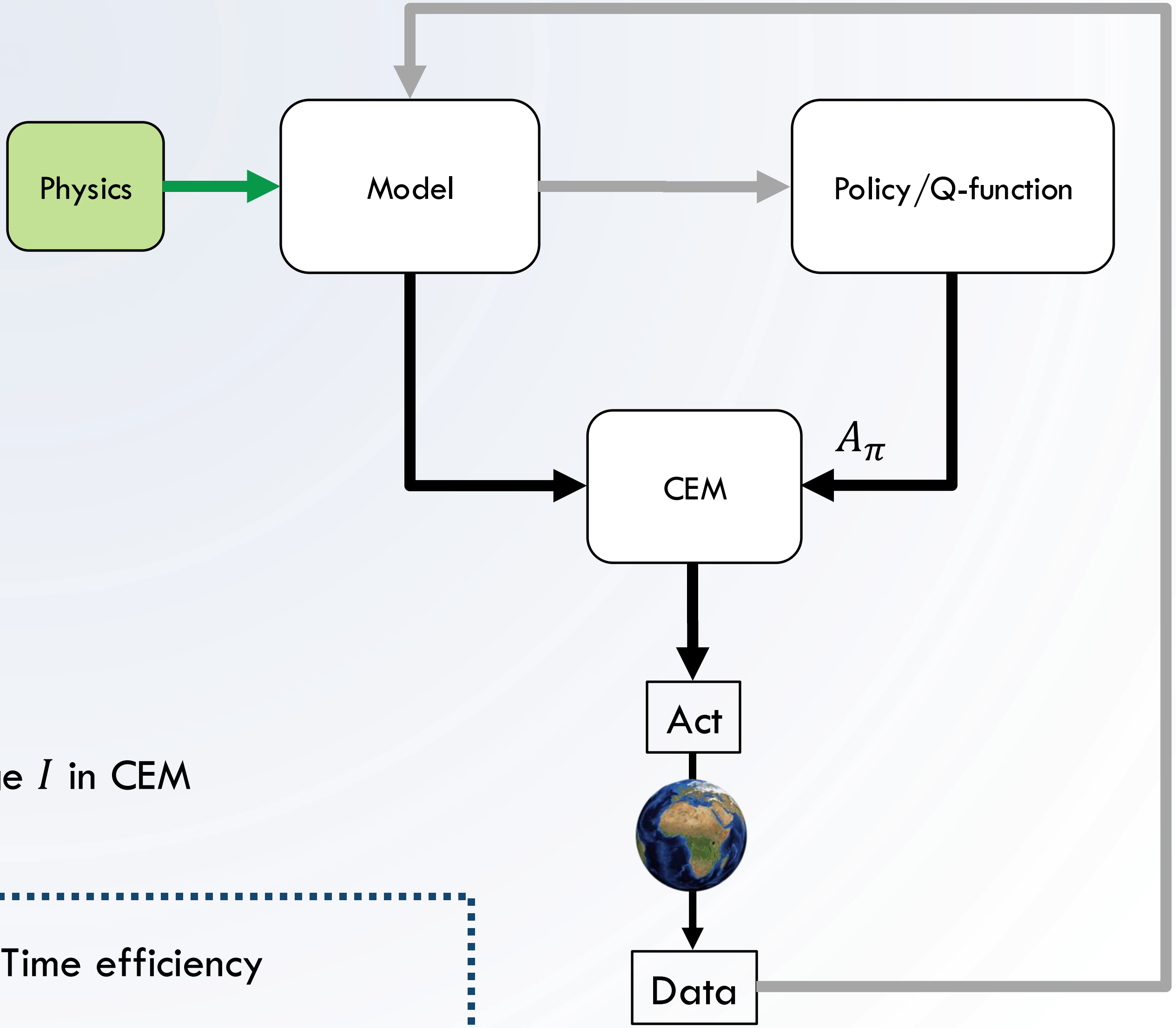
PhIHP: hybrid controller

- Combine A_π (policy) and A_{random}
- Trajectory score

$$A^* = \arg\max_{A \in \mathcal{A}^H} \Big(\sum_{t=t_0}^{H} \gamma^{t-t_0} R(s_t, a_t) + \alpha \cdot \gamma^{H-t_0} Q(s_H) \Big)$$

long-term reward

local solution



Good physics-informed model $\Rightarrow$ good policy

Good policy: Informative candidates A_π,

$\Rightarrow$ reduce population size large P and iterations large I in CEM

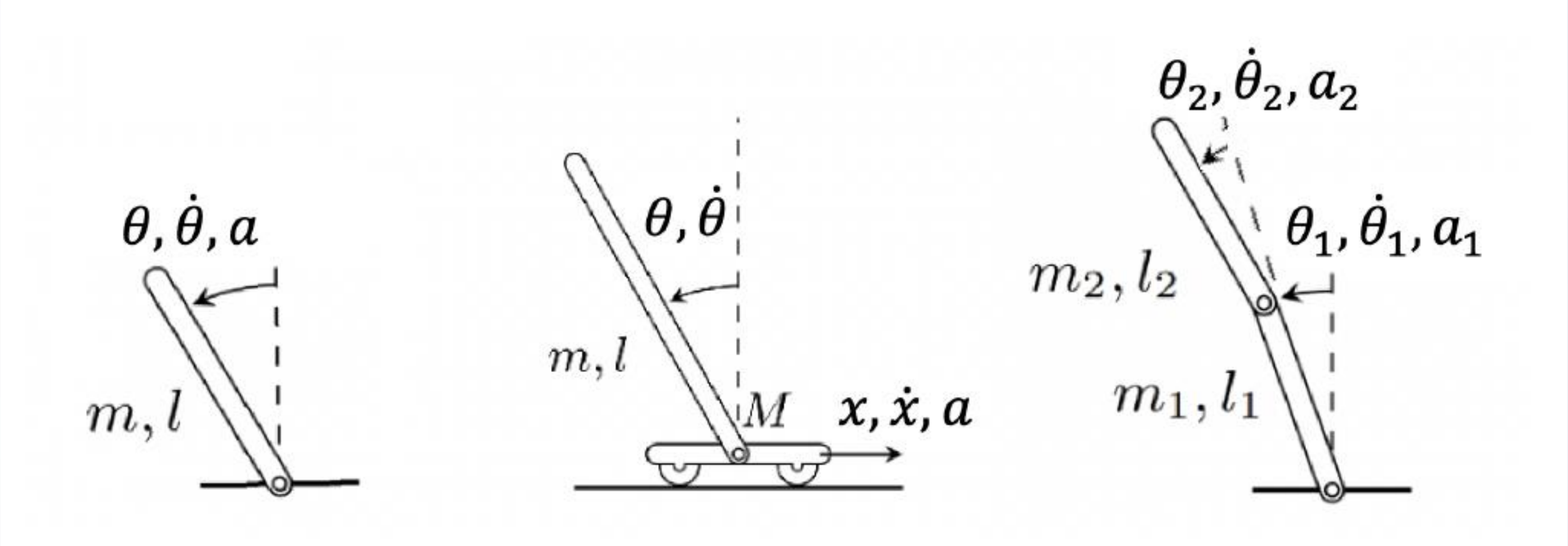
Good Q-function:

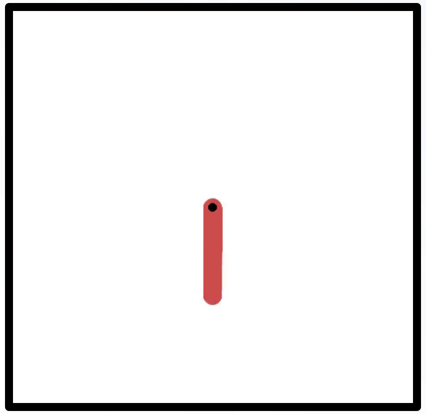
$\Rightarrow$ reduce the planning horizon H

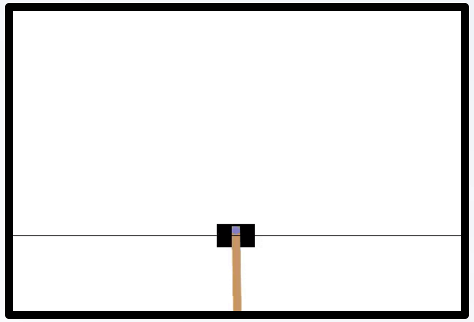
- Time efficiency
- Asymptotic performance

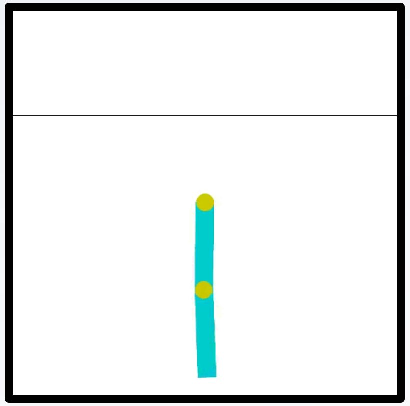
PhIHP: Experiments

Pendulum | **CartPole** | **Acrobot**



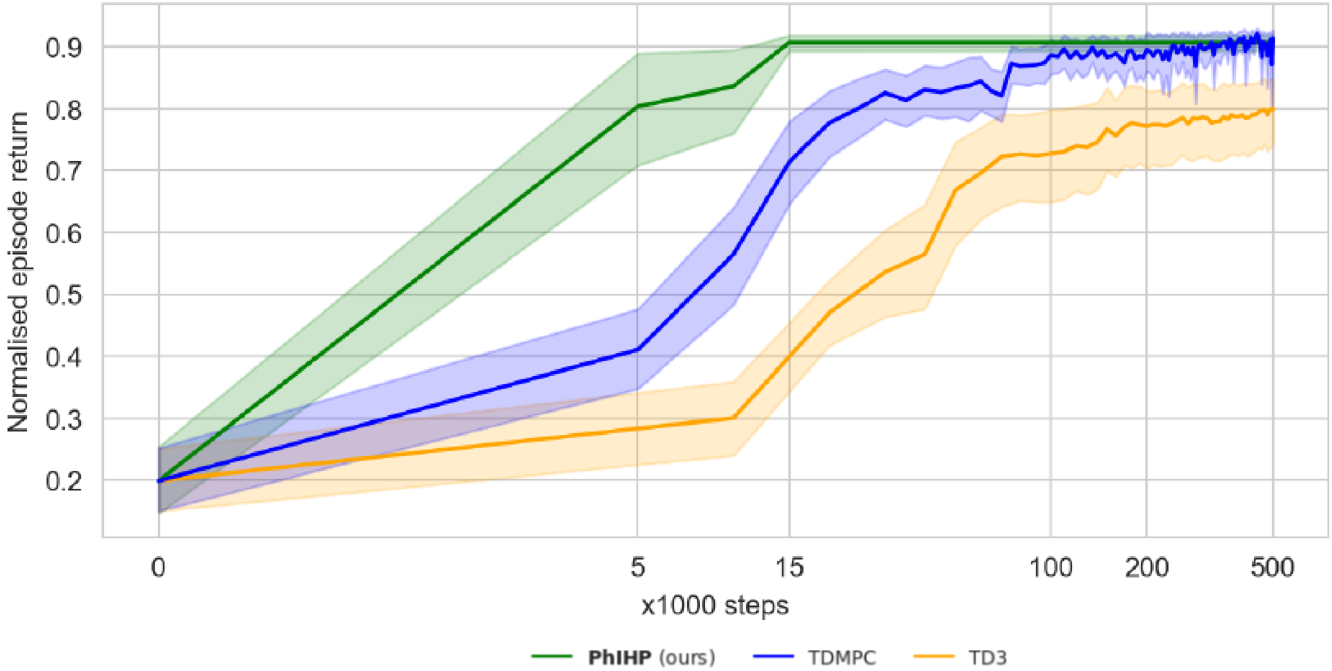






Approximate physics:
no friction

PhIHP: Results



(a) Learning curves, the x-axis uses a symlog scale.

Figure 3: Comparison of PhIHP *vs* baselines aggregated on 6 control tasks (10 runs). a) PhIHP shows excellent sample efficiency and better asymptotic performance.

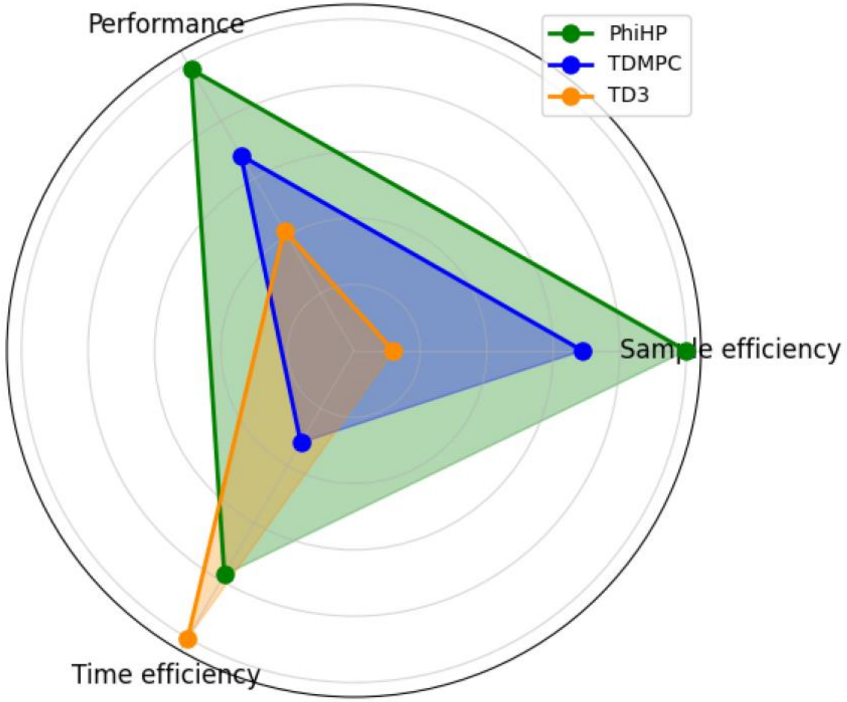


Figure 1: PhIHP includes a Physics-Informed model and hybrid planning for efficient policy learning in RL. PhIHP improves the compromise over state-of-the-art methods, model-free TD3 and hybrid TD-MPC, between sample efficiency, time efficiency, and performance. Results averaged over 6 tasks (Towers et al., 2023).

Real-Time Hybrid control with Physics (RT-HCP)

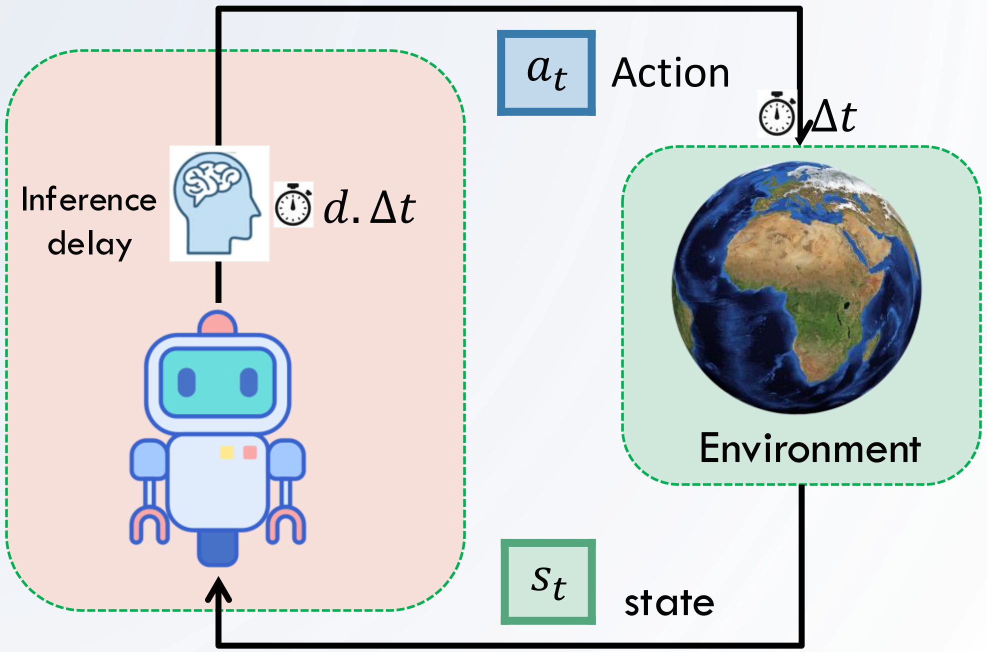
Motivation:

Extending PhiP to directly learn on robotic platforms

Main challenge:

inference delay in embedded devices: d steps

b) Inference delay



Env state	s_0	...	s_d	...	s_{2d}	...	$s_{d.t}$
Env action	a_0		a_1		a_2		a_t

Methodology: how many actions to send?

Δt given, H^p : Planning Horizon setup for desired performances

- **Compute inference time T_i:**

Measure the inference time on the robot.

- **Calculate the relative delay in timesteps:**

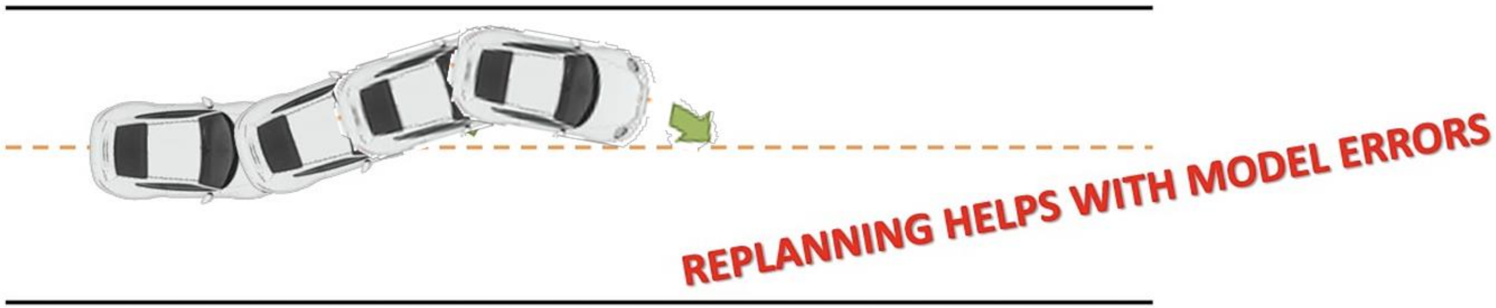
$d = \frac{T_i}{\Delta t}$ where Δt is the timestep duration.

- **Define an execution horizon $\boldsymbol{H^e}$; $d \leq H^e \leq H^p$:**

Set $\boldsymbol{H^e_{min} = int\left(\frac{T_i}{\Delta t}\right) + 1}$.

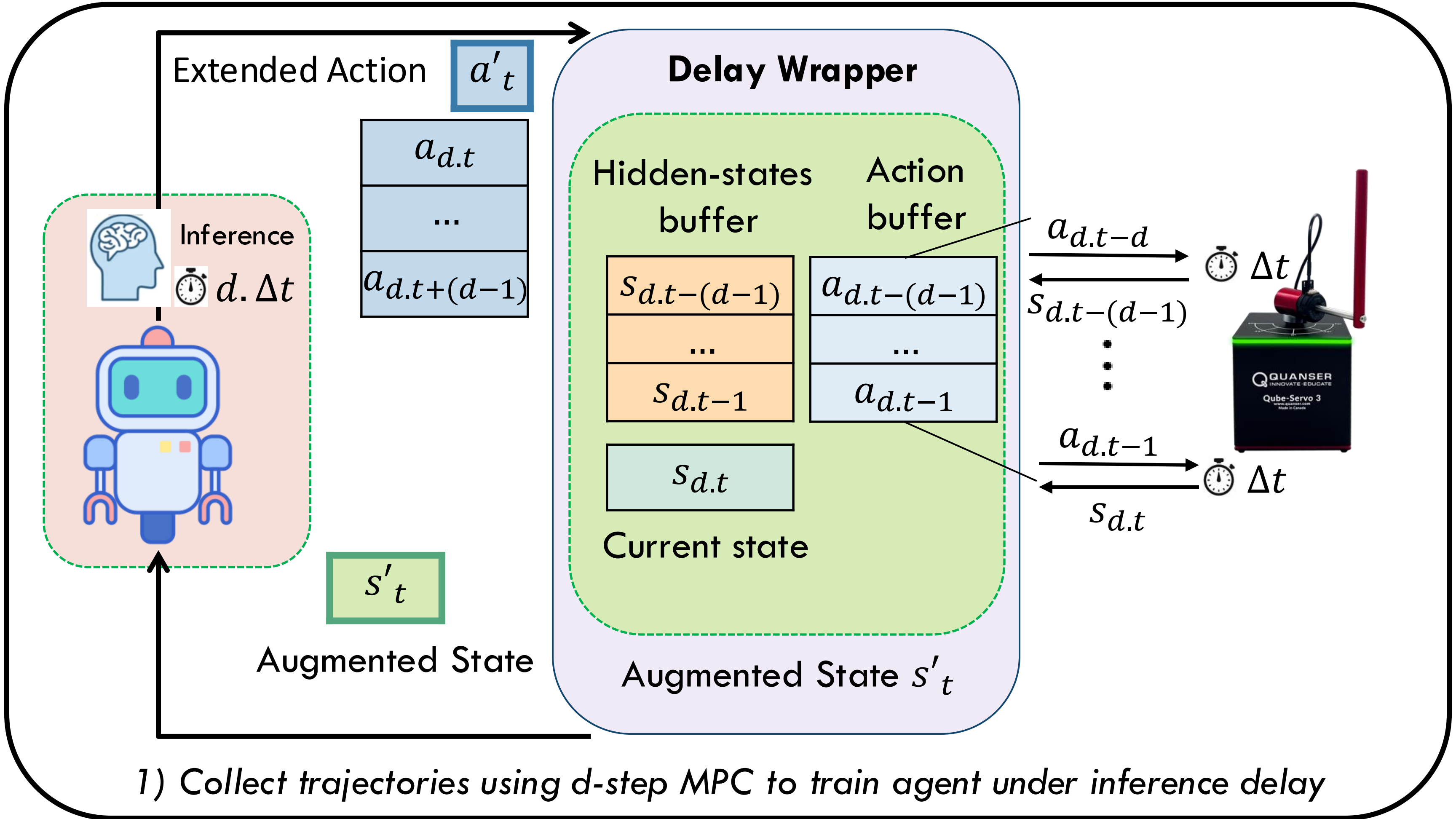
- **Apply n-step MPC:**

Select the first H^e actions from the optimized plan.

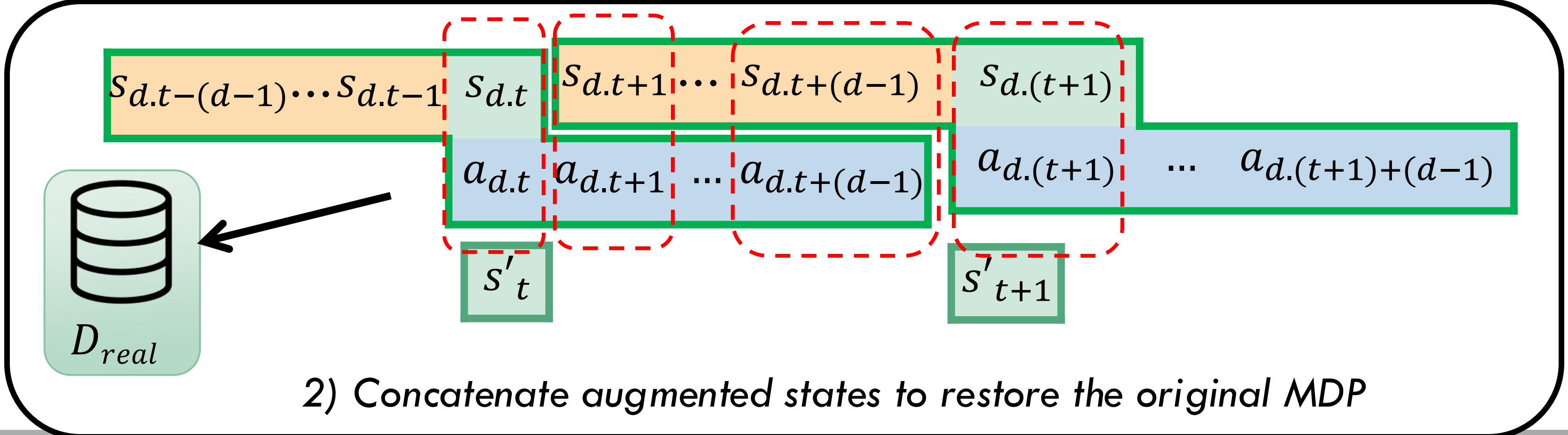


Method

D-step MPC

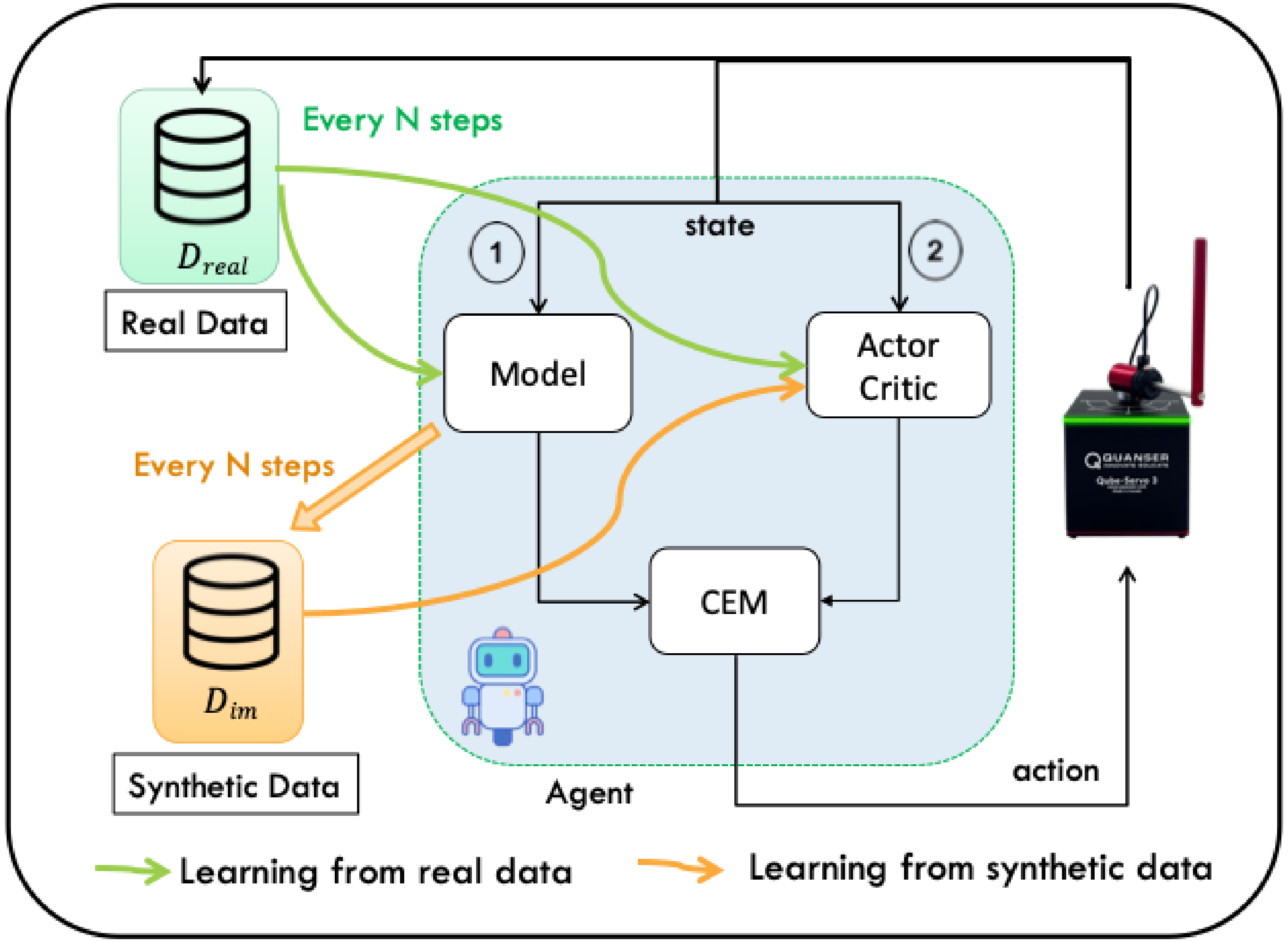


Delay-MDP

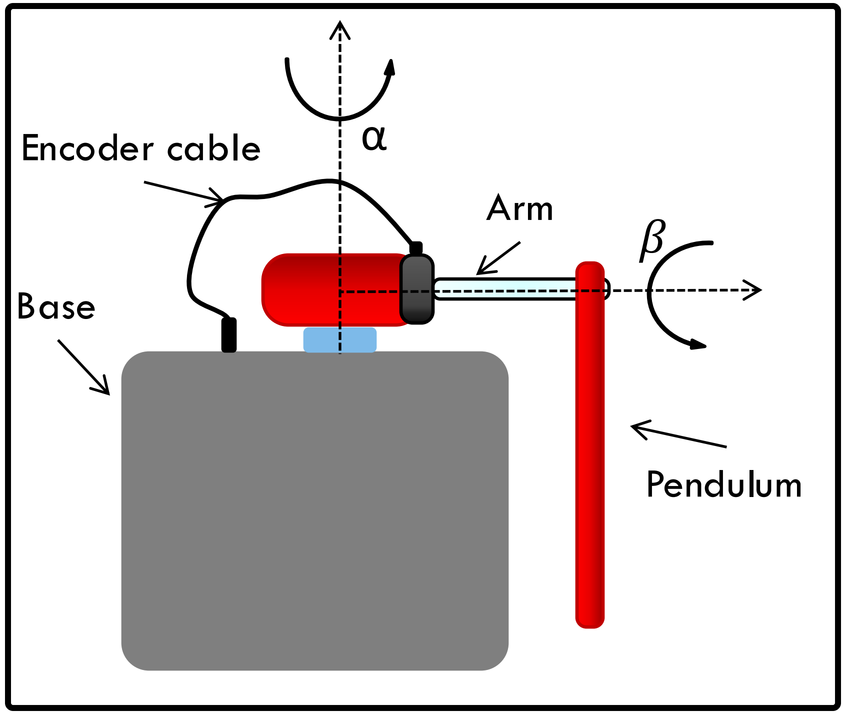


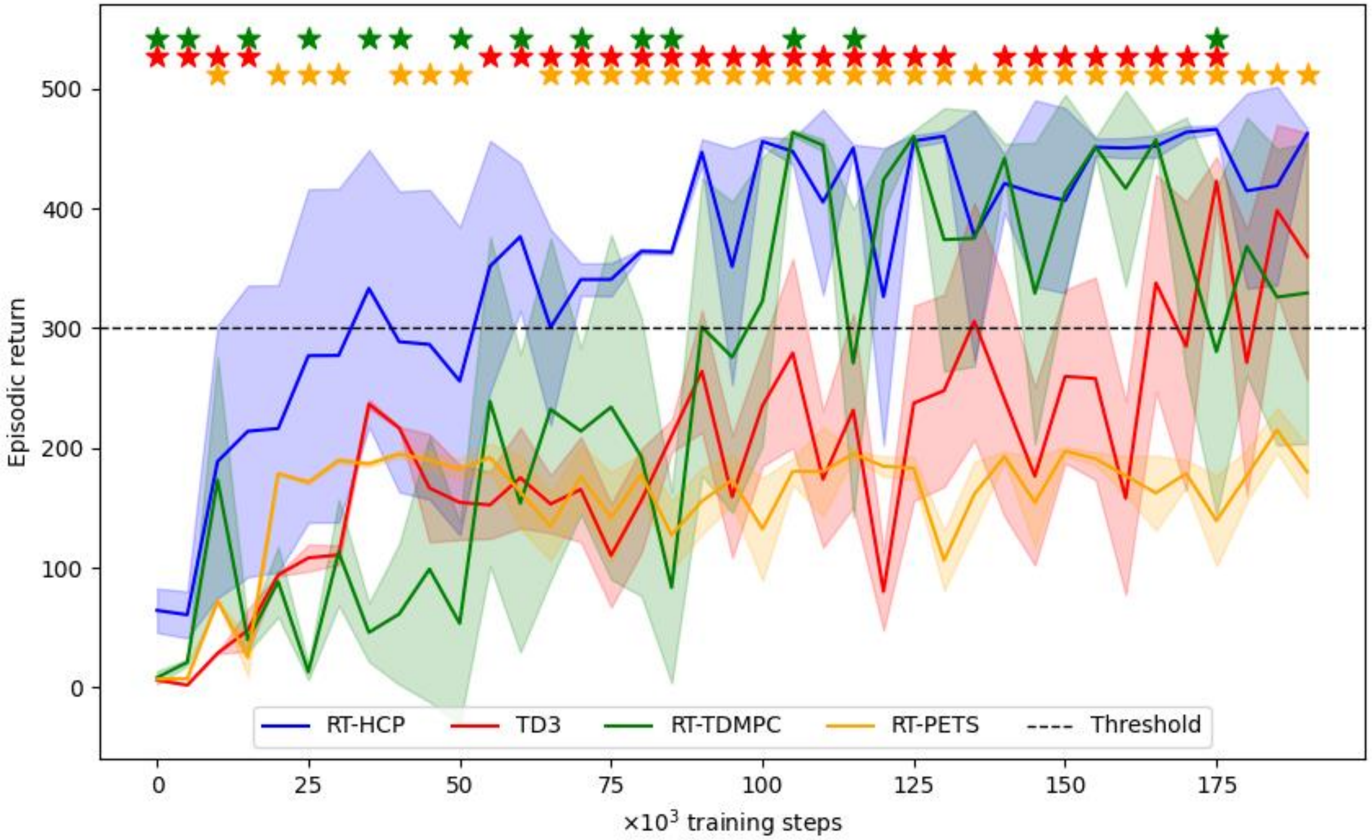
Real-Time Hybrid control with Physics (RT-HCP)

- Jointly learning environment model + controller on D_{real}.
- Periodically refine the policy through imagination on D_{real} + D_{im}



RT-HCP: Experiments & results





Real Furuta pendulum

- Approximate model: double pendulum
- Fine-tuning physical parameters
- Learning residual friction and cable effects

Robot frequency : Δt **= 20ms**
Agent frequency : T_i **= 60ms**

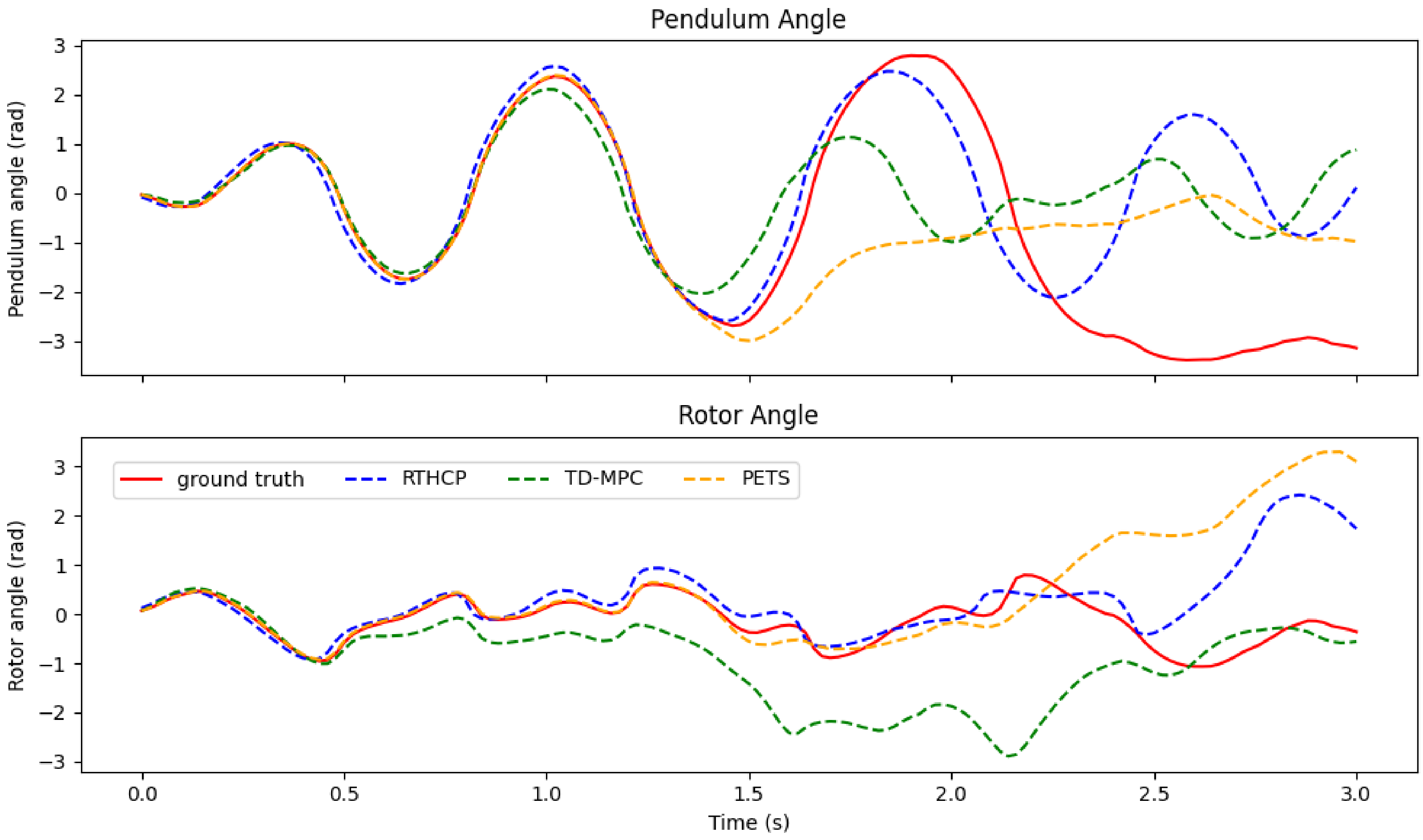
RT-HCP: Experiments & results

	RT-HCP	RT-TDMPC	TD3	RT-PETS
60k steps 23 minutes				
100k steps 35 minutes				
160k steps 58 minutes				

RT-HCP: Experiments & results

Model Prediction Accuracy:

- RT-HCP provides the most accurate trajectory predictions.
- TD-MPC exhibits the largest deviations over time.
- PETS fails to complete the swing-up task, despite its improved predictive accuracy.



Conclusion

PhIHP *[7]* improves the trade-off between sample efficiency, inference time, and asymptotic performance by combining physics-informed models and hybrid planning.

RT-HCP *[8]* extends this idea to real robotic systems, addressing inference delays.

Together, these methods bring us closer to deployable RL on physical robots, learning in real time, directly from interaction.

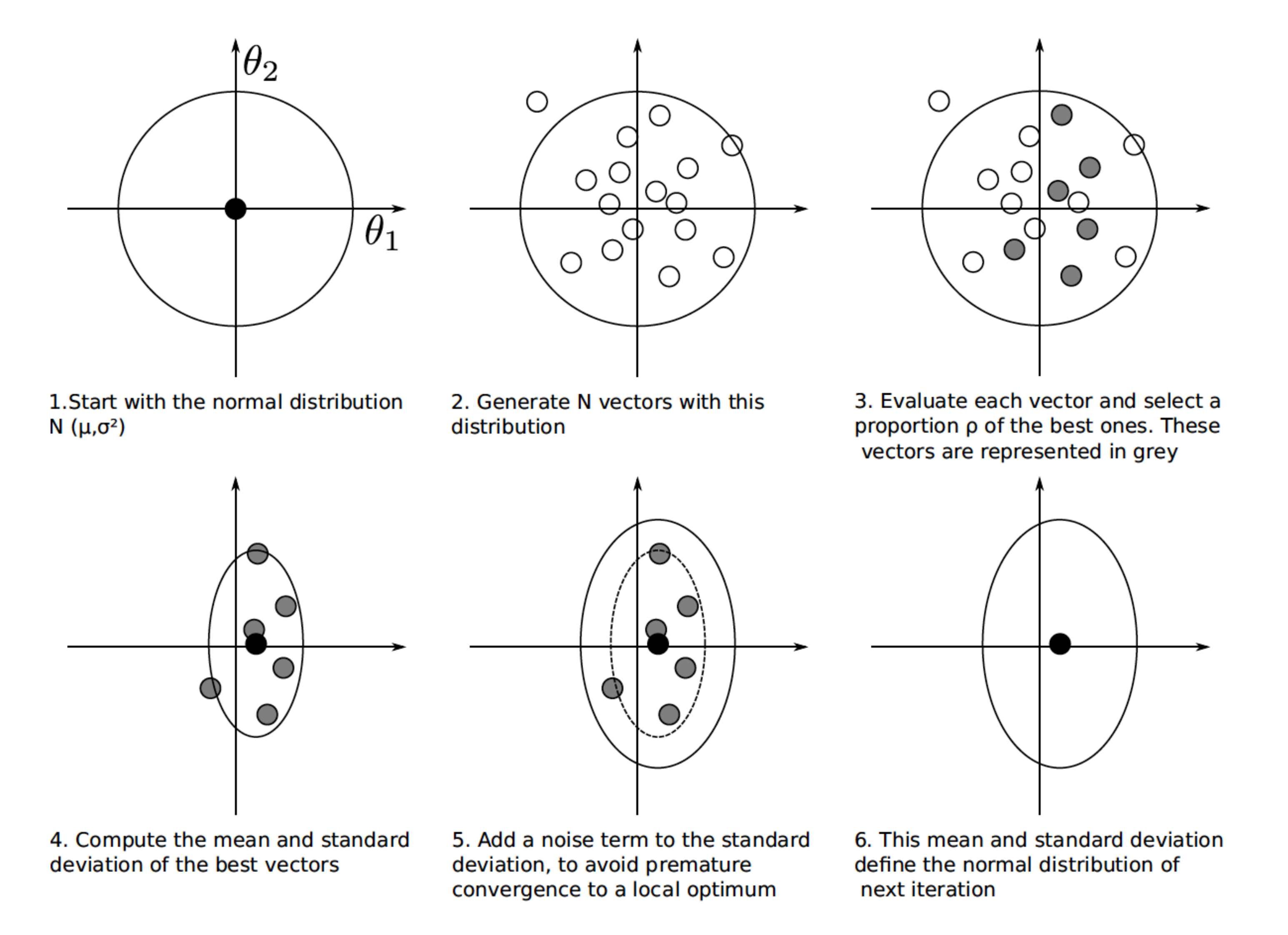
[7] Z. El Asri, O. Sigaud, N. Thome. **Physics-Informed Model and Hybrid Planning for Efficient Dyna-Style Reinforcement Learning. RLC 2024.**

[8] Z. El Asri, I. Laiche, C. Rambour, O. Sigaud, N. Thome. **RT-HCP: Dealing with Inference Delays and Sample Efficiency to Learn Directly on Robotic Platforms. IROS 2025.**

Thank you for your attention

Appendix

Reminder: Cross-Entropy Method (CEM)



Marin, and Sigaud, O. Towards fast and adaptive optimal control policies for robots: A direct policy search approach, Robotica 2012

The quality of the physics-informed model

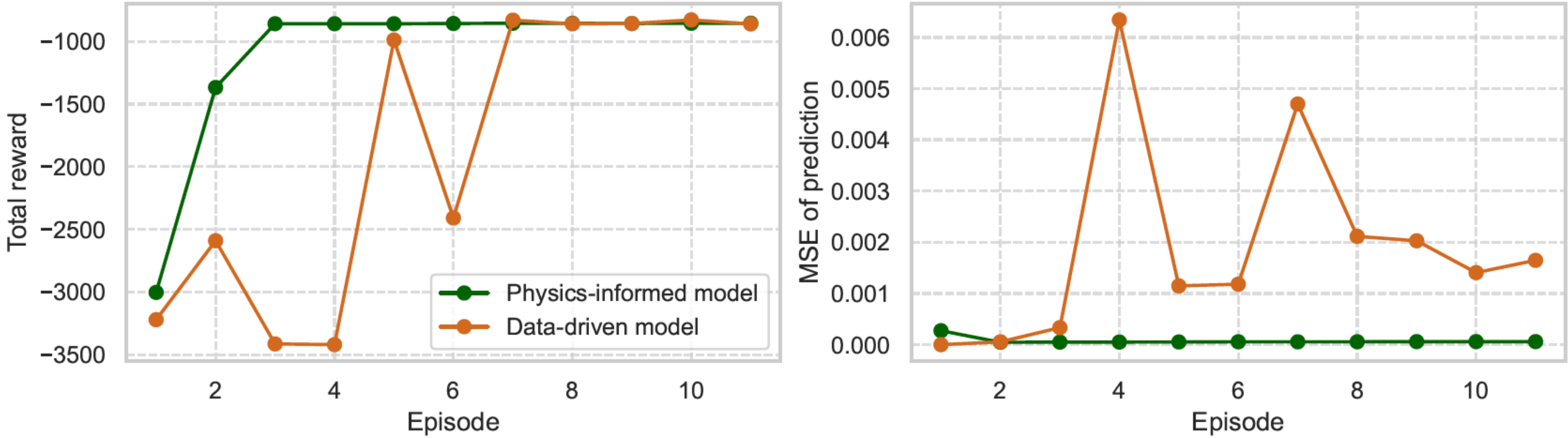


Figure 6: A data-driven model still poorly predicts the next states even when its asymptotic performance matches that of the physics-informed model. Figure obtained with 10 episodes of model training on Pendulum swingup.

Ablation study – Impact of learning through imagination & hybrid planning

- PhIHP
- PhIHP w/o "imagination"
- PhIHP w/o "imagination and physics"
- PhIHP w/o "policy"
- PhIHP w/o "policy and physics"

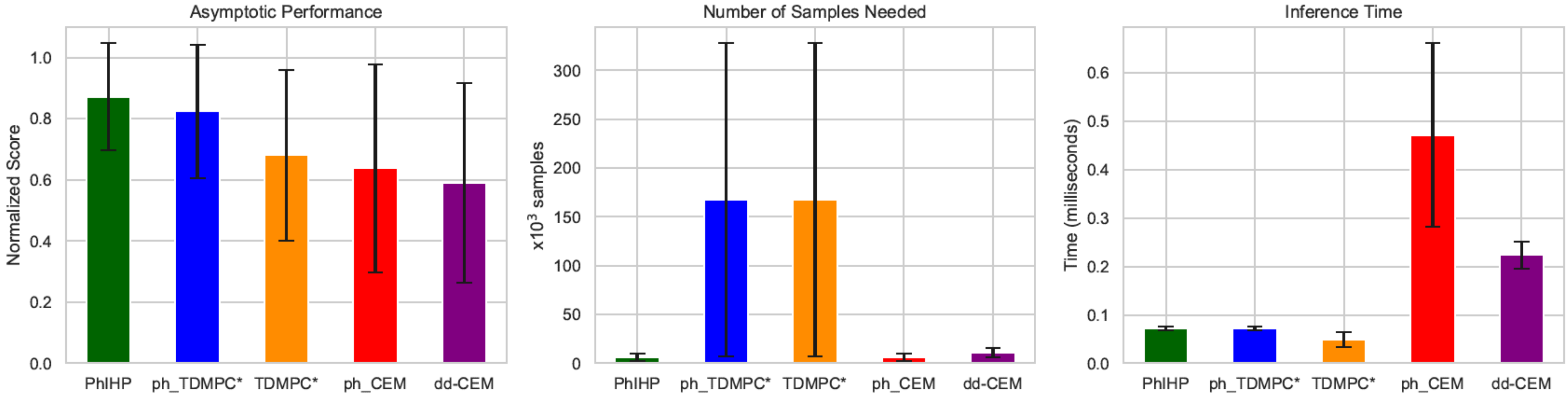


Figure 5: Comparison of PhIHP and its variants on the 3 main metrics. The figures illustrate the aggregated results of running all algorithms on 6 classic control tasks. Histograms and bars represent mean and std. over 10 runs.