SAFE MODEL-BASED REINFORCEMENT LEARNING

Gwendal PRISER, Goran FREHSE, David FILLIAT

Unité d'Informatique et d'Ingénierie des Systèmes, ENSTA Paris, Institut Polytechnique de Paris

Objectives

Ensuring the **safety** of the system during policy learning for dealing with **real-world systems**.

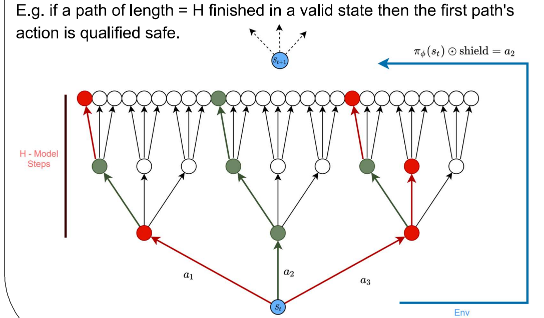
The algorithm must be **data-efficient** and **provably safe**.

Safety

A shield [2] is added to the agent so that it can only choose from actions of a safe action set.

A lookahead tree of short-term horizon H completes the shield and ensures the safety of action.

E.g. if a path of length = H finished in a valid state then the first path's action is qualified safe.

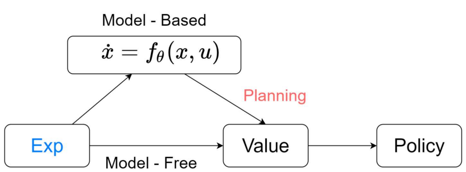


RL Algorithms

Reinforcement Learning algorithms fall into one of two classes :

Model-based algorithms build the dynamic equation of the system and use it to control the system.

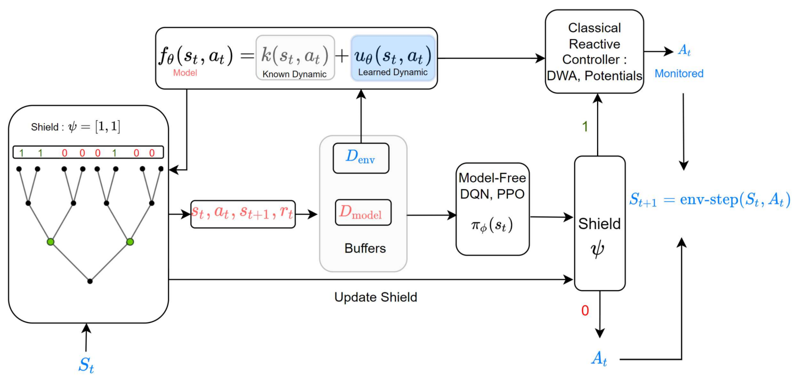
Model-free algorithms learn a direct mapping from states to actions.



Algorithm

A **model of the system** is used to **qualify the safety** of action as well as to increase the number of transitions in the dataset.

The state of the system is propagated through a lookahead tree. Then, the shield can be applied when executing the policy on the real-world system.



MBPO [1] is extented with actions taken **from a safe action set** :

Algorithm 2 Safe Augmented Deep-Q-Learning (SADQN)

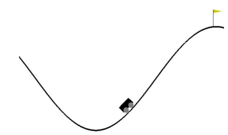
Initialize policy π_ϕ, predictive model f_θ, environment dataset $\mathcal{D}_{\text{env}}$, model dataset $\mathcal{D}_{\text{model}}$, shield Ψ
for N epochs **do**
  Train model f_θ on $\mathcal{D}_{\text{env}}$ via maximum likelihood
  for E steps **do**
    Take safe action in environment according to $\Psi(\pi_\phi)$, arrive in s_t; add to $\mathcal{D}_{\text{env}}$
    for M model rollouts **do**
      Perform H-step random model-rollout starting from s_t using policy π_ϕ; add to $\mathcal{D}_{\text{model}}$
      Update shield Ψ
    for G gradient updates **do**
      Update policy parameters on model data: $\phi \leftarrow \phi - \lambda_\pi \hat{\nabla}_\phi J_\pi(\phi, \mathcal{D}_{\text{model}})$

Experiments

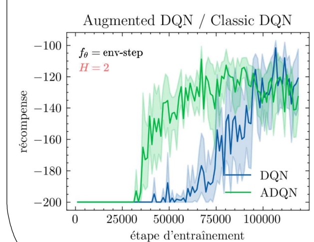
The algorithm has been applied to discrete classical problems with better performances (mean rewards for 100 episodes) compared to a classical off-policy algorithm (DQN).

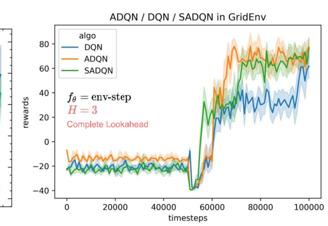
The addition of the shield in Grid Env did not reduce the performances.

Classic Control [4]



Grid Env





Perspectives

Use a **set-based method** [3] to qualify the safety of actions and extend the prediction horizon.

Apply the algorithm to **continuous action space** with the shield.

Acknowledgements This work was fully supported by ANR via ANR-21-FAI1-0005

References

[1] Janner, Michael, Justin Fu, Marvin Zhang, and Sergey Levine. "When to trust your model: Model-based policy optimization." Advances in Neural Information Processing Systems 32 (2019)

[2] Krasowski, Hanna; Wang, Xiao; Althoff, Matthias: Safe Reinforcement Learning for Autonomous Lane Changing Using Set-Based Prediction. 2020 IEEE International Conference on Intelligent Transportation

[3] Jaulin, Luc, Michel Kieffer, Olivier Didrit, and Eric Walter. "Interval analysis." In Applied interval analysis, pp. 11- 43. Springer, London, 2001.

[4] Brockman, Greg, Vicki Cheung, Ludwig Pettersson, Jonas Schneider, John Schulman, Jie Tang, Wojciech Zaremba. "Openai gym." arXiv preprint arXiv:1606.01540 (2016).