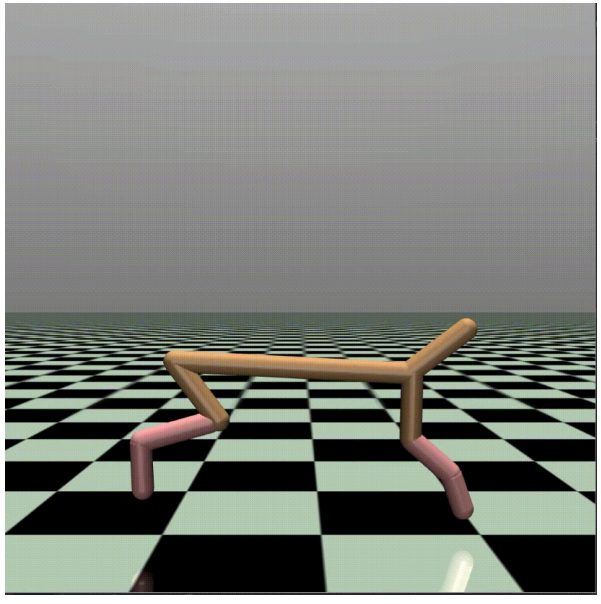
Reinforcement Learning

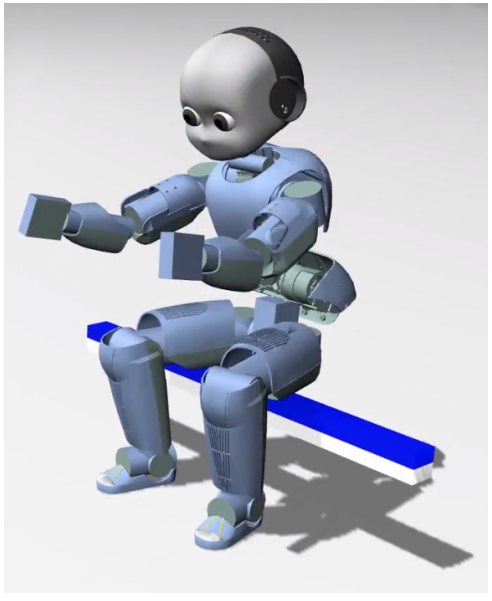
8. Deep Deterministic Policy Gradient

Olivier Sigaud

Sorbonne Université
http://people.isir.upmc.fr/sigaud

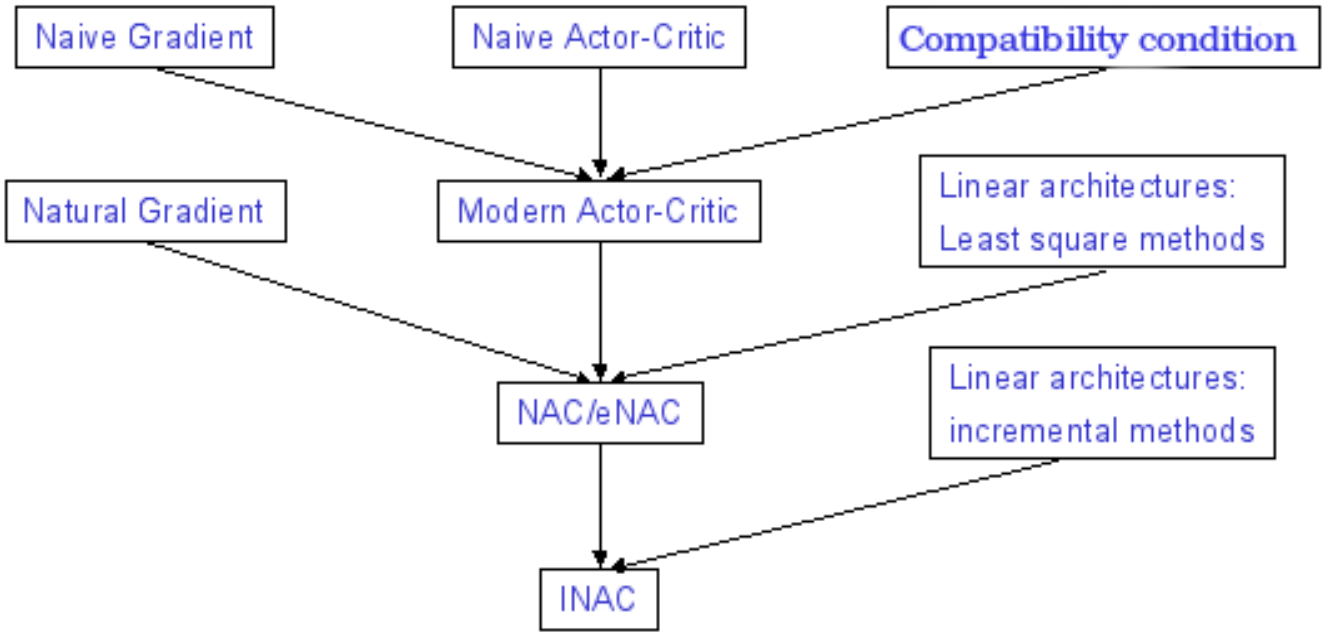
Reinforcement learning over continuous actions





- In RL, you need a max over actions
- If the action space is continuous, this is a difficult optimization problem
- Policy gradient methods and actor-critic methods mitigate the problem by looking for a local optimum (Pontryagine methods vs Bellman methods)
- In this class, we focus on Actor-Critic methods

Quick history of previous attempts (J. Peters' and Sutton's groups)



- Those methods proved inefficient for robot RL
- Keys issues: value function estimation based on linear regression is too inaccurate, tuning the stepsize is critical

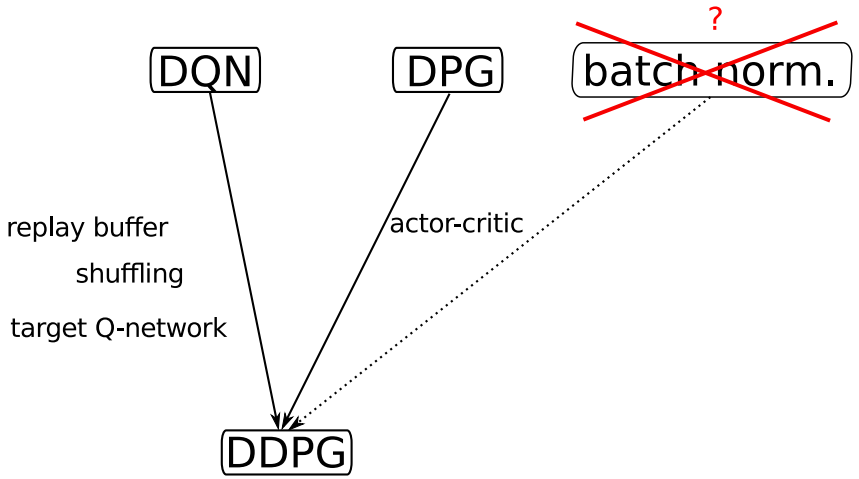
Sutton, R. S., McAllester, D., Singh, S., & Mansour, Y. (2000) Policy gradient methods for reinforcement learning with function approximation. In NIPS 12 (pp. 1057–1063).: MIT Press.

Deep Deterministic Policy Gradient

- Continuous control with deep reinforcement learning
- Works well on "more than 20" (27-32) domains coded with MuJoCo (Todorov) / TORCS
- End-to-end policies (from pixels to control) or from state variables

Lillicrap, T. P., Hunt, J. J., Pritzel, A., Heess, N., Erez, T., Tassa, Y., Silver, D., and Wierstra, D. (2015) Continuous control with deep reinforcement learning. *arXiv preprint arXiv:1509.02971 7/9/15*

DDPG: ancestors

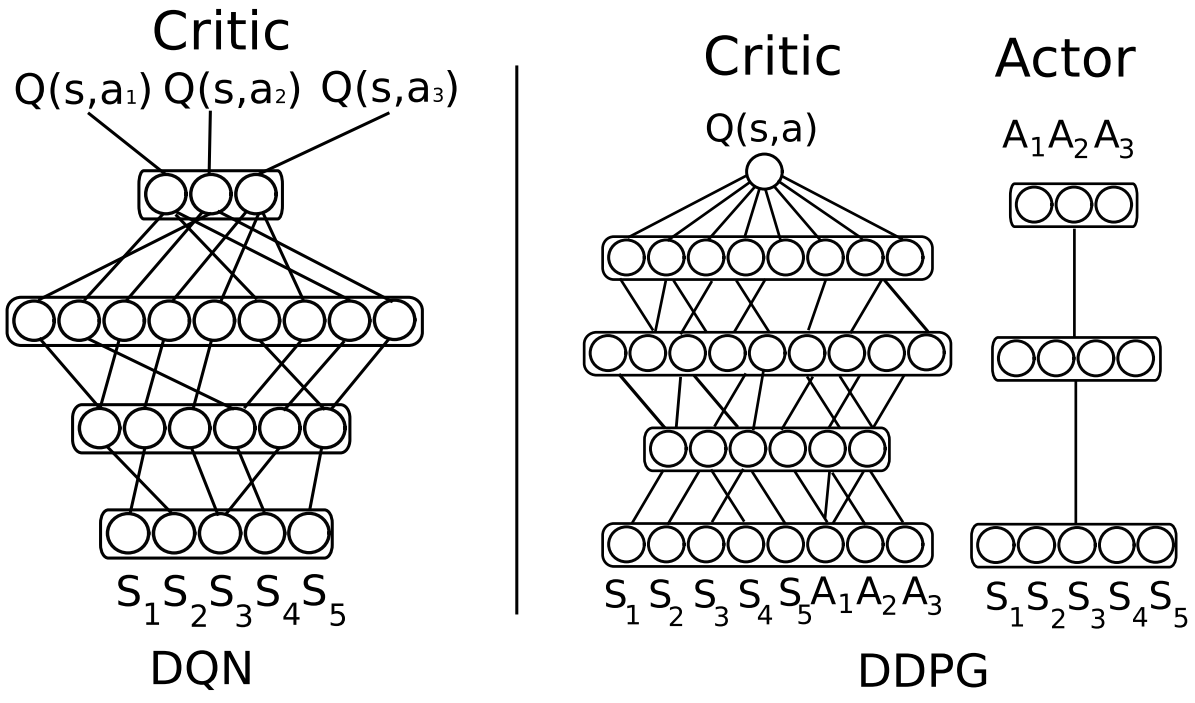


- Most of the actor-critic theory for continuous problem is for stochastic policies (policy gradient theorem, compatible features, etc.)
- DPG: an efficient gradient computation for deterministic policies, with proof of convergence
- Batch norm: inconclusive studies about importance

Silver, D., Lever, G., Heess, N., Degris, T., Wierstra, D., & Riedmiller, M. (2014) Deterministic policy gradient algorithms. In *ICML*

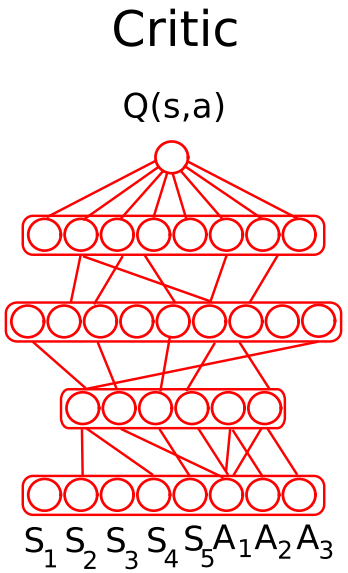
Ioffe, S. & Szegedy, C. (2015) Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv:1502.03167*

General architecture



- Actor $\pi_\mu(a_t|s_t)$, critic $Q(s_t, a_t|\theta)$
- All updates based on SGD
- Adaptive gradient descent techniques tune the step size (RProp, RMSProp, Adagrad, Adam...)

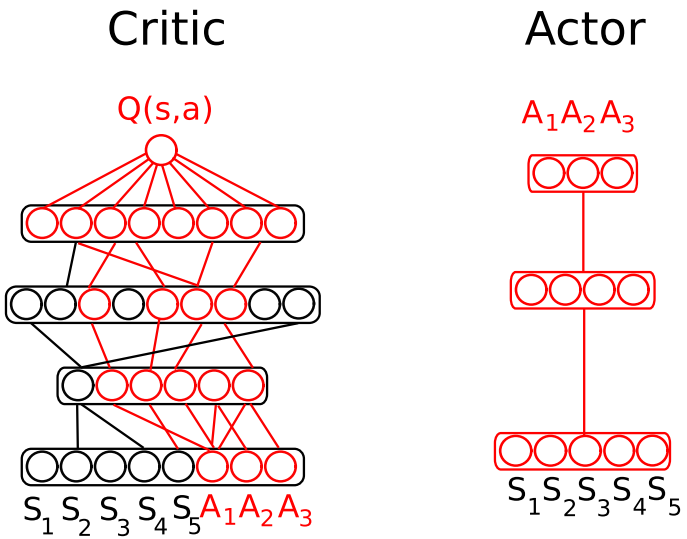
Training the critic



- Same idea as in DQN, but for actor-critic rather than Q-LEARNING
- Minimize the RPE: $\delta_t = r_t + \gamma Q(s_{t+1}, \pi(s_{t+1})|\theta) - Q(s_t, a_t|\theta)$
- Given a minibatch of N samples $\{s_i, a_i, r_i, s_{i+1}\}$ and a target network Q', compute $y_i = r_i + \gamma Q'(s_{i+1}, \pi(s_{i+1})|\theta')$
- And update θ by minimizing the loss function

$$L = 1/N \sum_i (y_i - Q(s_i, a_i|\theta))^2 \quad (1)$$

Training the actor



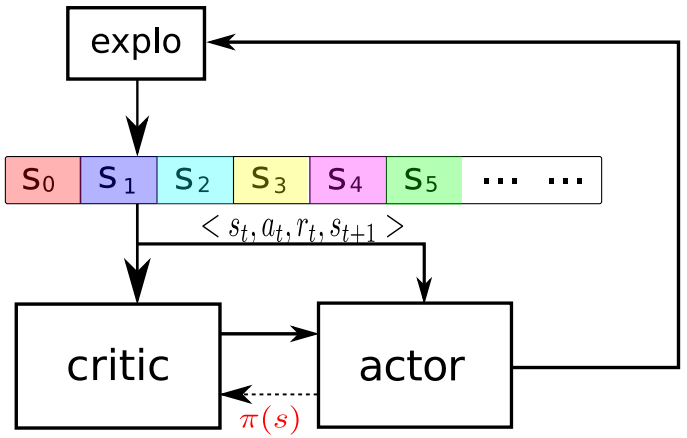
- Deterministic policy gradient theorem: the true policy gradient is

$$\nabla_\mu \pi(s,a) = \mathbb{E}_{\rho(s)}[\nabla_a Q(s,a|\theta)\nabla_\mu \pi(s|\mu)] \quad (2)$$

- $\nabla_a Q(s,a|\theta)$ is used as error signal to update the actor weights.
- Comes from NFQCA
- $\nabla_a Q(s,a|\theta)$ is a gradient over actions
- $y = f(w.x + b)$ (symmetric roles of weights and inputs)
- Gradient over actions $\sim$ gradient over weights

Hafner, R. & Riedmiller, M. (2011) Reinforcement learning in feedback control. *Machine learning*, 84(1-2), 137–169.

Off-policiness

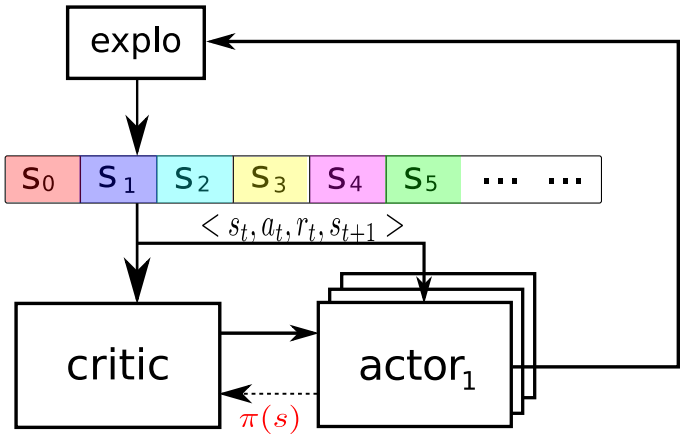


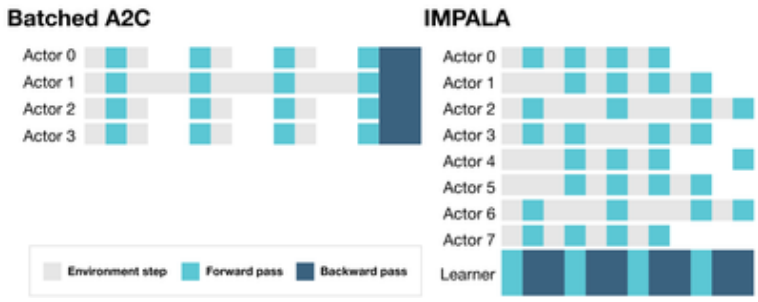
- The actor update rule is

$$\nabla_{\mathbf{w}}\pi(s_i) \approx 1/N \sum_i \nabla_a Q(s, a|\theta)|_{s=s_i, a=\pi(s_i)} \nabla_{\mathbf{w}}\pi(s)|_{s=s_i}$$

- The action from the actor is used:
  - To compute the target value $y_i = r_i + \gamma Q'(s_{i+1}, \pi(s_{i+1})|\theta')$
  - To update the actor
- As we have seen, actor-critic is off-policy, but convergence is fragile

Parallel updates



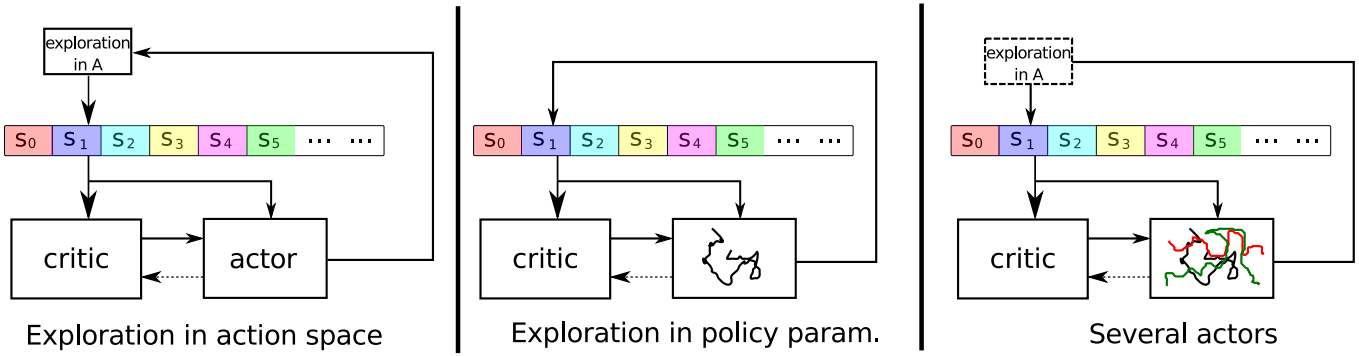


- Updating the critic and the actor can be done in parallel
- One may use several actors, several critics...
- Other state-of-the-art methods: Gorila, IMPALA: parallel implementations without replay buffers

Espeholt, L., Soyer, H., Munos, R., Simonyan, K., Mnih, V., Ward, T., Doron, Y., Firoiu, V., Harley, T., Dunning, I., et al. (2018) Impala: Scalable distributed deep-rl with importance weighted actor-learner architectures. *arXiv preprint arXiv:1802.01561*

Adamski, I., Adamski, R., Grel, T., Jedrych, A., Kaczmarek, K., & Michalewski, H. (2018) Distributed deep reinforcement learning: Learn how to play atari games in 21 minutes. *arXiv preprint arXiv:1801.02852*

Exploration (hot topic)



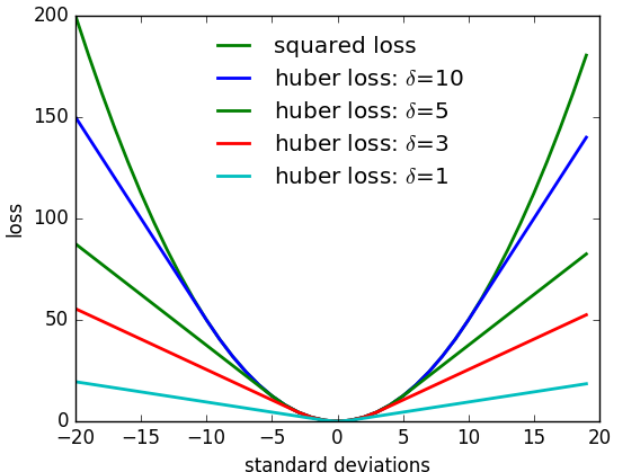
- Adding to the action an Ornstein-Uhlenbenk (correlated) noise process or Gaussian noise
- Action perturbation (versus param. perturbation, cf. e.g. Plappert or Fortunato, Noisy DQN)
- Several actors explore more

Plappert, M., Houthooft, R., Dhariwal, P., Sidor, S., Chen, R. Y., Chen, X., Asfour, T., Abbeel, P., & Andrychowicz, M. (2017) Parameter space noise for exploration. *arXiv preprint arXiv:1706.01905*

Fortunato, M., Azar, M. G., Piot, B., Menick, J., Osband, I., Graves, A., Mnih, V., Munos, R., Hassabis, D., Pietquin, O., et al. (2017) Noisy networks for exploration. *arXiv preprint arXiv:1706.10295*

Tuning hyper-parameters

tau	1e-4	1e-3	0.01	0.1	1.0	10.0	100.0
conv	2	33	34	43	36	37	33



- Influence of target critic update rate (τ)
  - If $\tau = 1$, no target critic from both sides (< 1, > 1)
  - In CMC, an optimum ~ 0.05 is found (non-standard DDPG code)
- Using Huber loss?
  - On some benchmark, the highest δ is best, thus no Huber loss
  - Unconclusive results, tuning is problem dependent
- Tuning hyper-parameters is difficult, start from the baselines

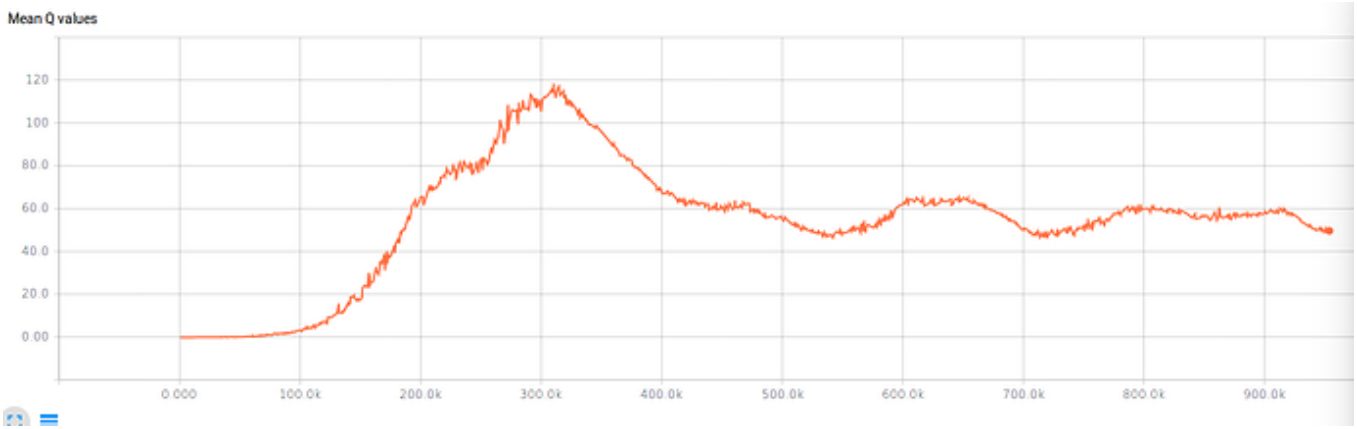
Dhariwal, P., Hesse, C., Klimov, O., Nichol, A., Plappert, M., Radford, A., Schulman, J., Sidor, S., & Wu, Y. (2017) OpenAI baselines. https://github.com/openai/baselines

Gradient inverter

- In bounded param. domains, the gradient may push beyond boundaries
- Invert the gradient when the parameter goes beyond the bound
- Better than gradient zeroing or gradient squashing (using *tanh* function)
- Efficient on CMC and Half-Cheetah

Hausknecht, M. & Stone, P. (2015) Deep reinforcement learning in parameterized action space. *arXiv preprint arXiv:1511.04143*

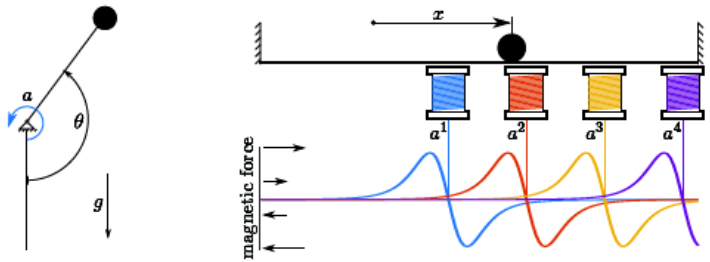
Over-estimation bias



- Clipping the target critic from the knowledge of R_{max} helps
- Several ways to act against an overestimation bias
- TD3: Have two critics, always consider the min, to prevent over-estimation
- Less problem knowledge than target critic clipping
- Gives a justification for target actor: slow update of policy is necessary

Fujimoto, S., van Hoof, H., & Meger, D. (2018) Addressing function approximation error in actor-critic methods. *arXiv preprint arXiv:1802.09477*

Replay buffer management



(a) Pendulum task (b) Magnetic manipulation task

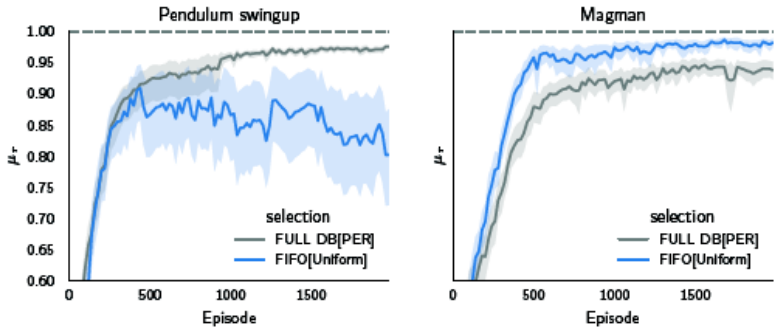
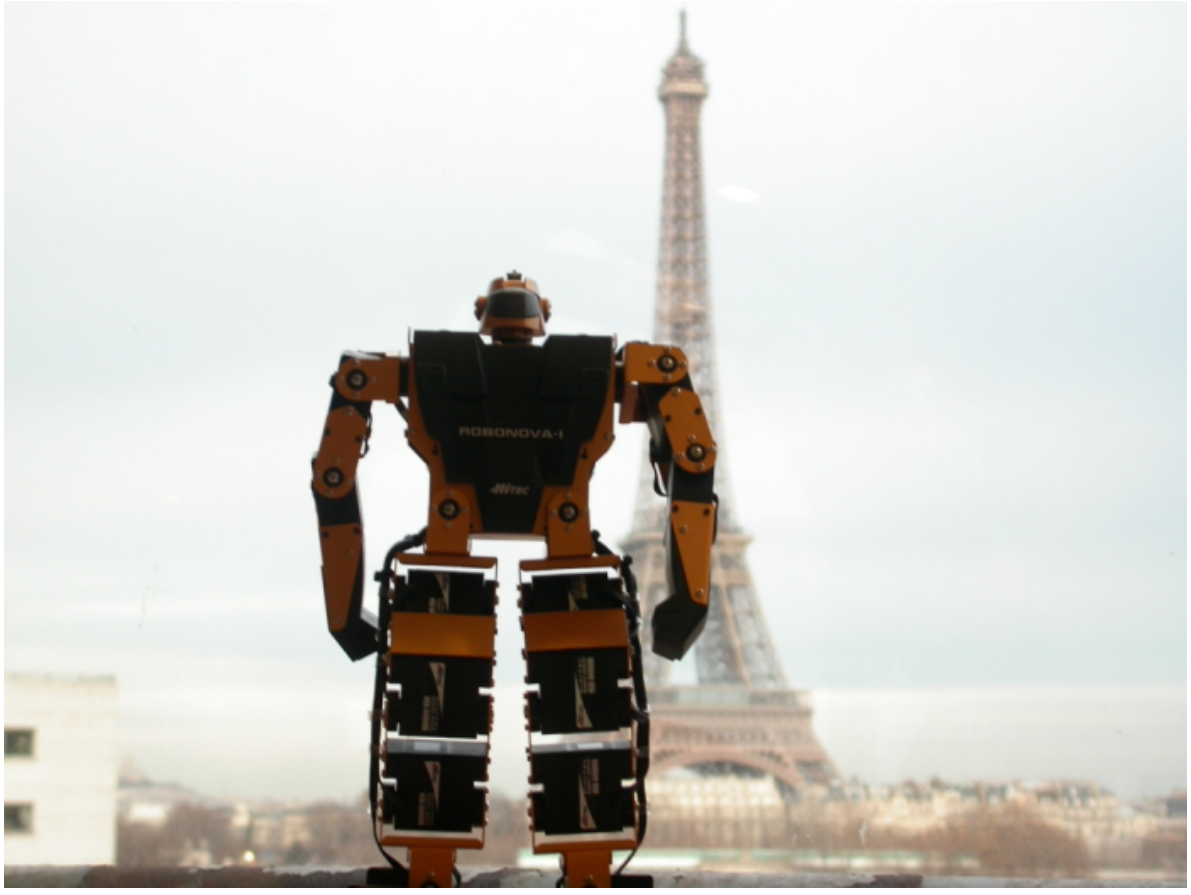


Figure 3: Comparison of the state-of-the-art (FULL DB[PER]) and the default method (FIFO[Uniform]) for experience selection on our two benchmark problems.

- Different replay buffer management strategies are optimal in different problems

de Bruin, T., Kober, J., Tuyls, K., & Babuška, R. (2018) Experience selection in deep reinforcement learning for control. *Journal of Machine Learning Research*, 19(9):1–56

Any question?



Send mail to: Olivier.Sigaud@upmc.fr

Adamski, I., Adamski, R., Grel, T., Jedrych, A., Kaczmarek, K., & Michalewski, H. (2018).

Distributed deep reinforcement learning: Learn how to play atari games in 21 minutes.
arXiv preprint arXiv:1801.02852.

de Bruin, T., Kober, J., Tuyls, K., & Babuška, R. (2018).

Experience selection in deep reinforcement learning for control.
Journal of Machine Learning Research, 19(9):1–56.

Dhariwal, P., Hesse, C., Klimov, O., Nichol, A., Plappert, M., Radford, A., Schulman, J., Sidor, S., & Wu, Y. (2017).

OpenAI baselines.
https://github.com/openai/baselines.

Espeholt, L., Soyer, H., Munos, R., Simonyan, K., Mnih, V., Ward, T., Doron, Y., Firoiu, V., Harley, T., Dunning, I., et al. (2018).

Impala: Scalable distributed deep-rl with importance weighted actor-learner architectures.
arXiv preprint arXiv:1802.01561.

Fortunato, M., Azar, M. G., Piot, B., Menick, J., Osband, I., Graves, A., Mnih, V., Munos, R., Hassabis, D., Pietquin, O., et al. (2017).

Noisy networks for exploration.
arXiv preprint arXiv:1706.10295.

Fujimoto, S., van Hoof, H., & Meger, D. (2018).

Addressing function approximation error in actor-critic methods.
arXiv preprint arXiv:1802.09477.

Hafner, R. & Riedmiller, M. (2011).

Reinforcement learning in feedback control.
Machine learning, 84(1-2):137–169.

Hausknecht, M. & Stone, P. (2015).

Deep reinforcement learning in parameterized action space.
arXiv preprint arXiv:1511.04143.

Ioffe, S. & Szegedy, C. (2015).

Batch normalization: Accelerating deep network training by reducing internal covariate shift.
arXiv preprint arXiv:1502.03167.

Lillicrap, T. P., Hunt, J. J., Pritzel, A., Heess, N., Erez, T., Tassa, Y., Silver, D., & Wierstra, D. (2015).

Continuous control with deep reinforcement learning.
arXiv preprint arXiv:1509.02971.

Plappert, M., Houthooft, R., Dhariwal, P., Sidor, S., Chen, R. Y., Chen, X., Asfour, T., Abbeel, P., & Andrychowicz, M. (2017).

Parameter space noise for exploration.
arXiv preprint arXiv:1706.01905.

Silver, D., Lever, G., Heess, N., Degris, T., Wierstra, D., & Riedmiller, M. (2014).

Deterministic policy gradient algorithms.
Edité dans *Proceedings of the 30th International Conference in Machine Learning*.

Sutton, R. S., McAllester, D., Singh, S., & Mansour, Y. (2000).

Policy gradient methods for reinforcement learning with function approximation.
Edité dans *Advances in Neural Information Processing Systems 12*, pages 1057–1063. MIT Press.