Asymmetric DQN for Partially Observable Reinforcement Learning

Andrea Baisero Brett Daley Christopher Amato
{baisero.a,daley.br,c.amato}@northeastern.edu
Khoury College of Computer Sciences, Northeastern University, Boston, USA

Problem Statement

- Many control problems are **partially observable** (PO), the agent acts without knowing the environment state s, and must rely on past observations, a.k.a., the history h.
- Training in simulation provides access to the simulation state.
- Actor-Critic methods exploit state information via **asymmetry**.

Q: Can value-based methods also use state information?
Can we develop deep algorithms that use state information?
A: Yes, through the theory of **asymmetric** value-based control.

Contributions

- Theory of **asymmetric** value-based PO control methods.
- Asymmetric value-based algorithms with focus on correctness: Asymmetric Policy Iteration, Asymmetric Action-Value Iteration, Asymmetric Q-Learning, **Asymmetric DQN**.
- Evaluation in environments with **significant** partial observability.

Motivation and Background

Partially Observable Control (PO Control)

- PO tasks require information gathering and memorization of past.
- Agent relies on good representation of history $\phi(h)$, **hard** to learn.
- Good $\phi(h)$ extracts key events and filters the rest, but ...
  ... identifying key events is like finding a needle in a haystack ...
  ... while learning to recognize needles and haystacks ...
  ... without supervision.

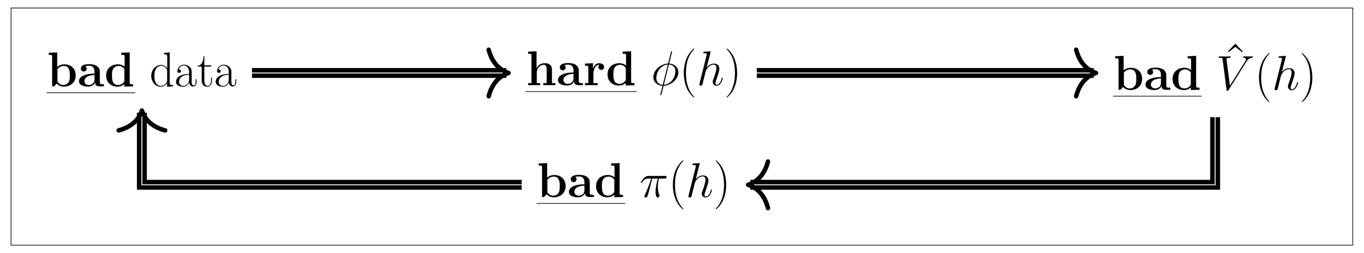

Figure: A vicious Actor-Critic cycle.

Offline Training and Online Execution (OTOE)

- Agent trains **offline** in simulation, executes **online**.
- Offline training algorithms can access environment state ...
  ... which can be used via **asymmetry** [1, 2], e.g.,
  Unbiased Asymmetric Actor-Critic [2] trains $\pi(h)$ using $\hat{V}(h, s)$.
- Representation of state $\phi(s)$ is **easier** to learn than $\phi(h)$ ...
  ... which helps learn a better critic $\hat{V}(h, s)$ ...
  ... bootstrap a better $\phi(h)$ ...
  ... leading to a better policy $\pi(h)$.

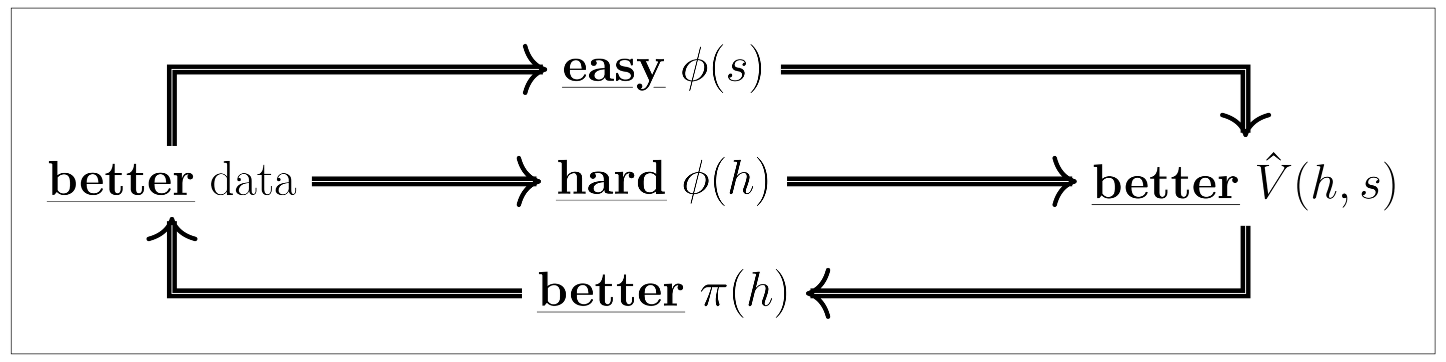

Figure: A better Asymmetric Actor-Critic cycle.

Asymmetric Value-Based PORL

Introducing Asymmetry into Value-Based PORL

- Actor-Critic methods implement asymmetry via $\pi(h)$ and $\hat{V}(h, s)$.
- Value-Based methods employ a single model $\hat{Q}(h, a)$.
  $\Rightarrow$ We implement asymmetry via $\hat{U}(h, s, a)$.

Asymmetric Policy Iteration (API)

From arbitrary U_0, Q_0, and π_0, generate the sequence U_k, Q_k, and π_k,

$$U_{k+1} \leftarrow \lim_{n\to\infty} B^n_{\pi_k} U_k \,, \qquad \text{(U-evaluation)} \qquad (1)$$
$$Q_{k+1} \leftarrow E U_{k+1} \,, \qquad \text{(Q-evaluation)} \qquad (2)$$
$$\pi_{k+1} \leftarrow g(Q_{k+1}) \,. \qquad \text{(improvement)} \qquad (3)$$

API is the simplest asymmetric method, analogous to Policy Iteration [3].

Why API?

- Demonstrates state can be used in value-based solution method.
- Serves as a basis for the other algorithms.

Limitations

- Requires POMDP model and expectations.
- Requires multiple iterations to approximate limit.
- Applicable to finite POMDPs only.
- Requires more computation than PI to achieve the same result.

Asymmetric Action-Value Iteration (AAVI)

From arbitrary U_0 and Q_0, generate the sequence U_k and Q_k,

$$U_{k+1} \leftarrow B_{g(Q_k)} U_k \,, \qquad \text{(U-evaluation + improvement)} \qquad (4)$$
$$Q_{k+1} \leftarrow E U_{k+1} \,. \qquad \text{(Q-evaluation)} \qquad (5)$$

AAVI is an eager variant of API, analogous to Value Iteration [3]).

Improvements

- Removes limit in U-evaluation step.
- Removes need for explicit policy representation.

Asymmetric Q-Learning (AQL)

We introduce incremental stochastic updates based on sample transitions. From arbitrary $Q_0 = EU_0$, generate the sequence U_k and Q_k,

$$U_{k+1} \leftarrow (1 - \alpha_k) U_k + \alpha_k (B_{g(Q_k)} U_k + w_k) \,, \qquad (6)$$
$$Q_{k+1} \leftarrow (1 - \alpha_k) Q_k + \alpha_k (E B_{g(Q_k)} U_k + v_k) \,. \qquad (7)$$

AQL uses sample experience, analogous to Q-learning [3].

Improvements

- Removes need for POMDP model and expectations.

Asymmetric DQN (ADQN)

We introduce value function approximation and reformulate the tabular update rules as squared-error losses on deep parametric models,

$$\mathcal{L}_{\hat{U}} = \frac{1}{2} \left(r + \gamma \hat{U}(hao, s', \operatorname*{argmax}_{a'} \hat{Q}(hao, a')) - \hat{U}(h, s, a) \right)^2 \,, \qquad (8)$$
$$\mathcal{L}_{\hat{Q}} = \frac{1}{2} \left(r + \gamma \hat{U}(hao, s', \operatorname*{argmax}_{a'} \hat{Q}(hao, a')) - \hat{Q}(h, a) \right)^2 \,. \qquad (9)$$

ADQN is a deep learning algorithm, analogous to DQN [4].

Improvements

- Applicable to POMDPs with high-dimensional states/observations.

Evaluation

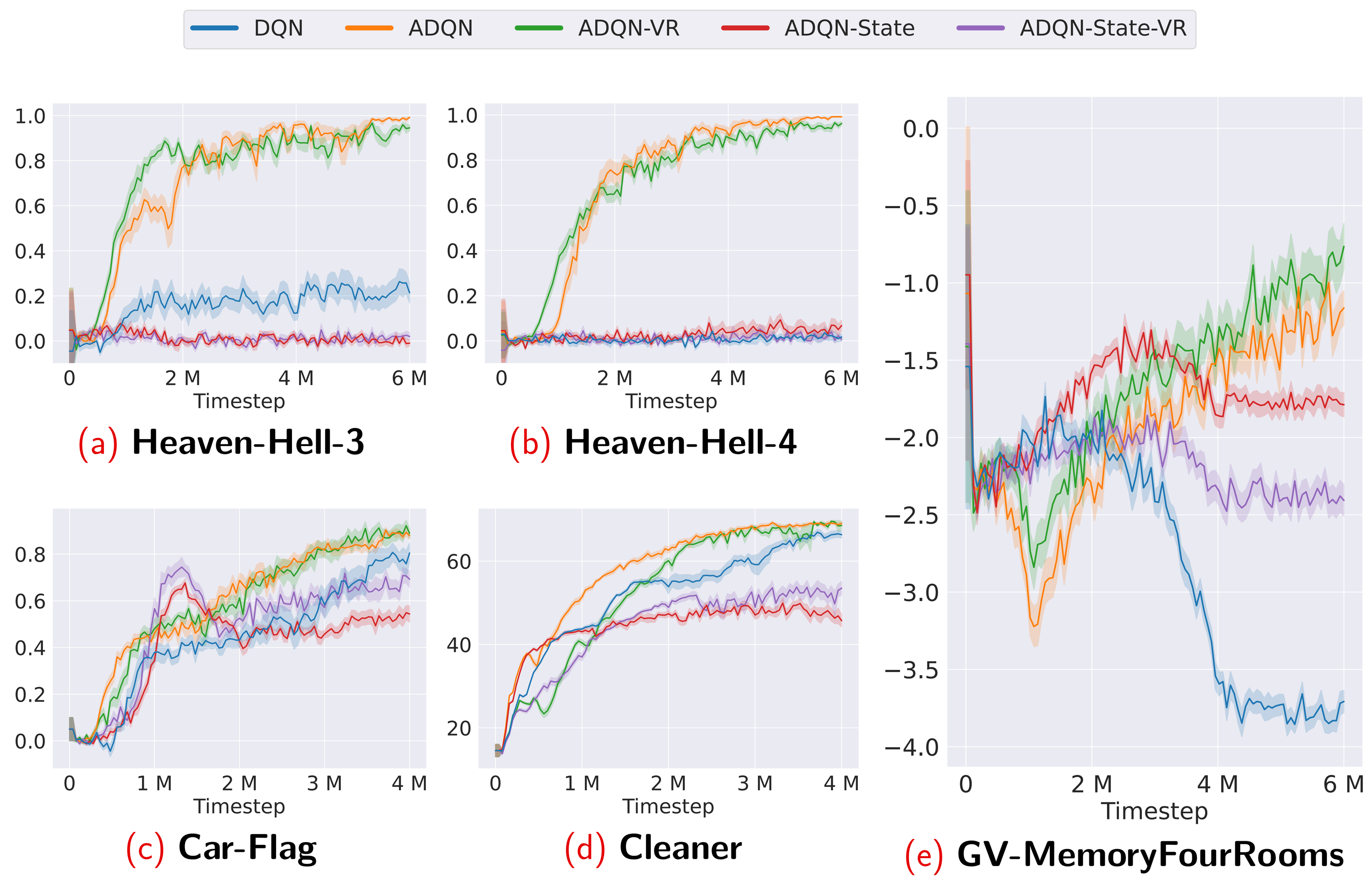

Figure: Episodic returns averaged over last 100 completed episodes, statistics computed over 5 independent runs. Shaded areas represent one standard error around the mean.

References

[1] L. Pinto, M. Andrychowicz, P. Welinder, W. Zaremba, and P. Abbeel, "Asymmetric actor critic for image-based robot learning," in *Proceedings of Robotics: Science and Systems*, 2018.

[2] A. Baisero and C. Amato, "Unbiased asymmetric reinforcement learning under partial observability," in *Proceedings of the Conference on Autonomous Agents and Multiagent Systems*, 2022.

[3] R. S. Sutton and A. G. Barto, *Reinforcement learning: An introduction.* MIT press, 2018.

[4] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves *et al.*, "Human-level control through deep reinforcement learning," *Nature*, vol. 518, no. 7540, p. 529, 2015.