Notes on Hybrid RL

Yuda Song

1 Setup

We have seen so far that in both online and offline RL, no algorithm is computationally tractable in the general function approximation setting. The reason is that, to achieve optimism/pessimism, the algorithm requires to search over the whole version space to find the most optimistic/pessimistic function or model.

However, if we think about real-world application, there is no reason to stop us from doing both – for example, in robotics, we nowadays have abundant offline demonstration data, and we often have access to online interaction as well. This gives the idea of Hybrid RL, which allows the learner to have both offline data and online interaction. As we will see, this framework indeed breaks the computational barrier of online or offline RL in the general function approximation setting.

Notation. We consider finite horizon Markov Decision Process $M = \{\mathcal{S}, \mathcal{A}, H, R, P, d_0\}$. We define a policy $\pi := \{\pi_0, \ldots, \pi_{H-1}\}$ where $\pi_h : \mathcal{S} \mapsto \Delta(\mathcal{A})$ and let d_h^π denotes the state-action occupancy induced by π at step h. Let $V_h^\pi(s) = \mathbb{E}[\sum_{\tau=h}^{H-1} r_\tau | \pi, s_h = s]$ and $Q_h^\pi(s,a) = \mathbb{E}[\sum_{\tau=h}^{H-1} r_\tau | \pi, s_h = s, a_h = a]$ be value functions and let $Q^\star$ and $V^\star$ denote the optimal value functions. We define the Bellman operator $\mathcal{T}$ such that for any $f : \mathcal{S} \times \mathcal{A} \mapsto \mathbb{R}$, $\mathcal{T} f(s,a) = \mathbb{E}[R(s,a)] + \mathbb{E}_{s' \sim P(s,a)} \max_{a'} f(s', a')$.

We assume that for each h we have an offline dataset $\mathcal{D}_h$ of m samples (s, a, r, s') drawn iid via $(s,a) \sim \nu_h, r \sim R(s,a), s' \sim P(s,a)$. For function approximation, we are given a function class $\mathcal{F} = \mathcal{F}_0 \times \cdots \times \mathcal{F}_{H-1}$ with $\mathcal{F}_h \subset \mathcal{S} \times \mathcal{A} \mapsto [0, V_{\max}]$. Let π^f to be the greedy policy w.r.t. f.

2 Hybrid Q Iteration

Now let us consider perhaps the most natural way to combine offline and online data: I use both offline and online data to fit a value function, and then I act greedily w.r.t. this value function, collect more online data, use both offline and online data to learn a new value function and repeat. We can see that this procedure is very simple - no complicated schemes of optimism or pessimism are needed, and as we will see, this simple procedure indeed has provable guarantees.

We outlined the algorithm in Algorithm 1. Specifically, to combine offline and online data, Algorithm 1 uses a half and half mixture. For the value function learning, it performs the finite horizon Fitted-Q-Iteration (FQI) (Munos and Szepesvári, 2008), treating the data mixture as an offline dataset. Note that the major computation requirement of Algorithm 1 is the least squares regression in FQI, and thus the algorithm is oracle-efficient.

3 Proof Sketch

We start with the stardard model-free function approximation assumption on the realizable and Bellman-complete value function class.

Assumption 3.1 (Realizability and Bellman completeness)**.** *For any h, we have $Q_h^\star \in \mathcal{F}_h$. Additionally, for any $f_{h+1} \in \mathcal{F}_{h+1}$, we have $\mathcal{T} f_{h+1} \in \mathcal{F}_h$.*

Algorithm 1 Hybrid Q-Iteration (Hy-Q)

require Value class: $\mathcal{D}$, #iterations: T, offline dataset $\mathcal{D}_h^\nu$ of size $m_{\text{off}} = T$ for $h \in [H-1]$.

Initialize $f_h^1(s,a) = 0$.

for $t = 1, \dots, T$ **do**

Let π^t be the greedy policy w.r.t. f^t i.e., $\pi_h^t(s) = \arg\max_a f_h^t(s,a)$.

For each h, collect $m_{\text{on}} = 1$ online tuples $\mathcal{D}_h^t \sim d_h^{\pi^t}$.

Set $f_H^{t+1}(s,a) = 0$.

for $h = H-1, \dots, 0$ **do**

Estimate f_h^{t+1} using least squares regression on the aggregated data $\mathcal{D}_h^t = \mathcal{D}_h^\nu + \sum_{\tau=1}^t \mathcal{D}_h^\tau$:

$$f_h^{t+1} \leftarrow \underset{f \in \mathcal{F}_h}{\arg\min} \left\{ \widehat{\mathbb{E}}_{\mathcal{D}_h^t} (f(s,a) - r - \max_{a'} f_{h+1}^{t+1}(s',a'))^2 \right\}$$

With this assumption, we have the usual guarantee that our learned value function has small error on both the offline data and the historical online data:

Lemma 3.1 (Bellman error bound for FQI)**.** *Let $\delta \in (0,1)$, with probability at least $1-\delta$, for any $h \in [H-1]$ and $t \in [T]$,*

$$\left\| f_h^{t+1} - \mathcal{T} f_{h+1}^{t+1} \right\|_{2,\nu_h}^2 \leq O\left(\frac{V_{\max}^2 \log(2HT|\mathcal{F}|/\delta)}{t} \right),$$

and

$$\sum_{\tau=1}^t \left\| f_h^{t+1} - \mathcal{T} f_{h+1}^{t+1} \right\|_{2,d_h^{\pi^\tau}}^2 \leq O\big(V_{\max}^2 \log(2HT|\mathcal{F}|/\delta) \big).$$

This is just by standard concentration arguments.

Hybrid RL decomposition. With this in mind, the following is the core idea of hybrid RL, which state that, given any comparator policy π^e as long as the learned value function has small Bellman error on both π^e's visitation distribution, and the greedy policy w.r.t. the learned value function, then the greedy policy can compete with π^e.

Lemma 3.2. *Given any comparator policy π^e, for any $f \in \mathcal{F}$ and corresponding greedy policy π^f, we have*

$$\mathbb{E}_{s_0 \sim d_0}\left[V_0^{\pi^e}(s_0) - V_0^{\pi^f}(s_0) \right] \leq \sum_{h=0}^{H-1} \underbrace{\mathbb{E}_{s_h,a_h \sim d_h^{\pi^e}} [\mathcal{T} f_{h+1}(s_h,a_h) - f_h(s_h,a_h)]}_{\textit{offline error}} + \underbrace{\mathbb{E}_{s_h,a_h \sim d_h^{\pi^f}} [f_h(s_h,a_h) - \mathcal{T} f_{h+1}(s_h,a_h)]}_{\textit{online error}}.$$

To see why this is true, we can consider the following decomposition:

$$\mathbb{E}_{s_0 \sim d_0}\left[V_0^{\pi^e}(s_0) - V_0^{\pi^f}(s_0) \right] = \mathbb{E}_{s_0 \sim d_0}\left[V_0^{\pi^e}(s_0) - \max_a f_0(s_0,a) + \max_a f_0(s_0,a) - V_0^{\pi^f}(s_0) \right].$$

The second difference should be famaliar to some of the readers since it is just a variant of the performance difference lemma:

$$\begin{aligned} \mathbb{E}_{s \sim d_0}[\max_a f_0(s,a) - V^{\pi^f}(s)] &= \mathbb{E}_{s \sim d_0}[\mathbb{E}_{a \sim \pi_0^f(s)} f_0(s,a) - V_0^{\pi^f}(s)] \\ &= \mathbb{E}_{s \sim d_0}[\mathbb{E}_{a \sim \pi_0^f(s)} f_0(s,a) - \mathcal{T} f_1(s,a)] + \mathbb{E}_{s \sim d_0}[\mathbb{E}_{a \sim \pi_0^f(s)} \mathcal{T} f_1(s,a) - V_0^{\pi^f}(s)] \\ &= \mathbb{E}_{s,a \sim d_0^{\pi^f}} [f_0(s,a) - \mathcal{T} f_1(s,a)] + \\ &\quad \mathbb{E}_{s \sim d_0}[\mathbb{E}_{a \sim \pi_0^f(s)} [R(s,a) + \gamma \mathbb{E}_{s' \sim \mathcal{P}(s,a)} \max_{a'} f_1(s',a') - R(s,a) + \mathbb{E}_{s' \sim \mathcal{P}(s,a)} V_1^{\pi^f}(s')]] \\ &= \mathbb{E}_{s,a \sim d_0^{\pi^f}} [f_0(s,a) - \mathcal{T} f_1(s,a)] + \mathbb{E}_{s \sim d_1^{\pi^f}} [\max_a f_1(s,a) - V_1^{\pi^f}(s)] \end{aligned}$$

and we can complete the second part by induction. The proof for the offline error is similar, and we leave it as an exercise for the readers.

Controlling Offline Error. To control the offline error, like in the offline RL literature, we need to make an assumption on the coverage of the offline data. To see why this makes sense, consider running Algorithm 1 with an offline data with no information provided, and since Algorithm 1 does not perform any exploration, we should not expect the returned policy to be good. Specifically, we use the following notion of coverage:

Definition 3.1 (Bellman error transfer coefficient)**.** *For any policy π, define the transfer coefficient as*

$$C_\pi := \max\left\{0,\ \max_{f\in\mathcal{F}} \frac{\sum_{h=0}^{H-1} \mathbb{E}_{s,a\sim d_h^\pi}\left[\mathcal{T} f_{h+1}(s,a) - f_h(s,a)\right]}{\sqrt{\sum_{h=0}^{H-1} \mathbb{E}_{s,a\sim\nu_h}\left(\mathcal{T} f_{h+1}(s,a) - f_h(s,a)\right)^2}}\right\}.$$

The definition cares about the ratio of the expected worst-case (in the context of the function class) Bellman error under the policy π to the expected Bellman error under the offline data. Note that this notion of coverage in terms of expected Bellman error is very general in the sense that it is smaller than the coverage definition used previously. It is easy to see that $C^\pi \le \sup_{s,a,h} \frac{d_h^\pi(s,a)}{\nu_h(s,a)}$, the density ratio coverage used in tabular MDPs. And one can also prove that C^π is smaller than the relative condition number used in linear MDPs.

Now with the transfer coefficient, we can immediately bound the offline error: for each h, we have with probability at least $1-\delta$,

$$\sum_{t=1}^{T} \mathbb{E}_{s,a\sim d_h^{\pi^e}}\left[\mathcal{T} f_{h+1}^t(s,a) - f_h^t(s,a)\right] \le \sum_{t=1}^{T} C_{\pi^e} \sqrt{\mathbb{E}_{s,a\sim\nu_h}\left(\mathcal{T} f_{h+1}^t(s,a) - f_h^t(s,a)\right)^2} \le \tilde{O}(\sqrt{TV_{\max}^2 \log(|\mathcal{F}|/\delta)}).$$

Controlling Online Error. The online error is the Bellman error of the current value function under the greedy policy w.r.t. the function. This term suggests that there is an implicit exploration in the procedure: if the current value function is accurate on its own, then we are done; otherwise, we explore. To bound this term, we can use any existing complexity measure in the online RL literature, that measures "how many times of distribution shift one can expect in a structured MDPs", for example, Bellman rank (Jiang et al., 2017), bilinear rank (Du et al., 2021), Bellman eluder dimension (Jin et al., 2021), or coverage (Xie et al., 2023). In this note, we use the bilinear rank as an example.

Definition 3.2 (Bilinear model (Du et al., 2021))**.** *We say that the MDP together with the function class $\mathcal{F}$ is a bilinear model of rank d if for any $h \in [H-1]$, there exist two (unknown) mappings $X_h, W_h : \mathcal{F} \mapsto \mathbb{R}^d$ with $\max_f \|X_h(f)\|_2 \le B_X$ and $\max_f \|W_h(f)\|_2 \le B_W$ such that:*

$$\forall f, g \in \mathcal{F}:\ \left|\mathbb{E}_{s,a\sim d_h^{\pi^f}}\left[g_h(s,a) - \mathcal{T} g_{h+1}(s,a)\right]\right| = |\langle X_h(f), W_h(g)\rangle|.$$

The intuition of the bilinear model is that, consider the Bellman error matrix $\mathcal{E} \in \mathbb{R}^{|\mathcal{F}|\times|\mathcal{F}|}$, where $\mathcal{E}_{f,g}$ denotes the Bellman error of g under the π^f, where $f, g \in \mathcal{F}$, then this matrix has rank at most d. Thus we should only expect $O(d)$ times of distribution shift – the Bellman error of any function under any policy can be well approximated by a linear combination of d other policies. Thus we can bound the online error as

$$\sum_{t=1}^{T} \mathbb{E}_{s,a\sim d_h^{\pi^f}}\left[f_h^t(s,a) - \mathcal{T} f_{h+1}^t(s,a)\right] \le \sum_{t=1}^{T}\left|\mathbb{E}_{s,a\sim d_h^{\pi^f}}\left[f_h^t(s,a) - \mathcal{T} f_{h+1}^t(s,a)\right]\right| = \sum_{t=1}^{T}\left|\left\langle X_h(f^t), W_h(f^t)\right\rangle\right|.$$

Let $\Sigma_h^t := \sum_{\tau=1}^t X_h(f^\tau) X_h(f^\tau)^\top + \lambda\mathbb{I}$, we get

$$\sum_{t=1}^{T}\left|\left\langle X_h(f^t), W_h(f^t)\right\rangle\right| \le \sum_{t=1}^{T} \|X_h(f^t)\|_{\Sigma_{t-1;h}^{-1}} \sqrt{\sum_{\tau=1}^{t-1} \mathbb{E}_{s,a\sim d_h^\tau}\left[\left(f_h^t(s,a) - \mathcal{T} f_{h+1}^t(s,a)\right)^2\right] + \lambda B_W^2}.$$

Using standard elliptical potential argument (Lemma 3.4), the first term $\sum_{t=1}^{T}\|X_h(f^t)\|_{\Sigma_{t-1;h}^{-1}} \le O(\sqrt{dT})$, and the second term is just the historical Bellman error, and together we have the online error is bounded by $\widetilde{O}(\sqrt{TdV_{\max}^2\log(|\mathcal{F}|/\delta)})$.

Thus combining everything, we have the following theorem:

Theorem 3.1 (Cumulative suboptimality)**.** *With probability at least $1-\delta$, Algorithm 1 obtains the following bound on cumulative subpotimality w.r.t. any comparator policy π^e,*

$$\sum_{t=1}^{T} V^{\pi^e} - V^{\pi^t} = \widetilde{O}\Big(\Big(\max\{C_{\pi^e},1\}+\sqrt{d}\Big)\cdot\sqrt{V_{\max}^2 H^2 T\cdot\log(|\mathcal{F}|/\delta)}\Big).$$

Now we can compare with the online RL results: for example in bilinear models, the best known regret bound is $\tilde{O}(\sqrt{dV_{\max}^2H^2T\cdot\log(|\mathcal{F}|/\delta)})$, and we can see that the hybrid RL algorithm only needs to pay for the additional coverage term C_{π^e}. In return, we get a computationally efficient algorithm without any deliberate designs for optimism or pessimism.

From the statistical perspective, we see that in the worst case, hybrid RL does not seem to have any advantage over online RL. This point is rigorously shown in Xie et al. (2021), with a lower bound in the tabular setting that matches the lower bound for either online or offline RL. More recently, Li et al. (2024) and Tan et al. (2024) gives more refined analysis using a more instance-dependent style coverage measure.

3.1 Example: Linear Bellman Completeness

Now we consider a canonical example that is still considered computationally hard without future assumption on the dynamics or action space: linear Bellman completeness (Wu et al., 2024; Golowich and Moitra, 2024).

Definition 3.3. *Consider linear function approximation, where $\mathcal{F}_h = \big\{f : f(s,a)=\theta^\top\phi_h(s,a), \|\theta\|_2\le B\big\}$, and $\phi_h\in\mathbb{R}^d$. We say the MDP is linear Bellman complete if for all h, there exists a mapping $\mathcal{T}_h:\mathbb{R}^d\to\mathbb{R}^d$ such that, for all θ with $\|\theta\|\le B$ and s,a, we have*

$$\langle\phi_h(s,a),\mathcal{T}\theta\rangle = \mathbb{E}_{s'\sim P_h(s,a)}[\max_{a'}\langle\phi_{h+1}(s',a'),\theta\rangle].$$

If we plug into the Bellman optimality condition we can see that $Q^\star\in\mathcal{F}$ and the completeness condition holds by definition.

To see why the bilinear rank assumption holds, we can see by definition, we can take W_h to be $\theta_h - \mathcal{T}\theta_{h+1}$ and X_h be the expected feature map. Thus the bilinear rank is at most d.

Finally, we can use standard covering number argument to show that a ℓ_∞ ε-net of $\mathcal{F}$ has size at most $O\Big(\big(\frac{B}{\varepsilon}\big)^d\Big)$, and thus we can get the following theorem:

Lemma 3.3. *Let $\delta\in(0,1)$, suppose the MDP is linear Bellman complete, $C_{\pi^*}<\infty$, and consider $\mathcal{F}_h$ defined above. Then, with probability $1-\delta$, Algorithm 1 finds an ε-suboptimal policy with total sample complexity (offline + online):*

$$n = \widetilde{O}\bigg(\frac{B^2C_{\pi^*}^2H^4d^2\log(B/\varepsilon\delta)}{\varepsilon^2}\bigg).$$

3.2 Technical Lemma

Lemma 3.4. *Let $X_h(f^1),\dots,X_h(f^T)\in\mathbb{R}^d$ be a sequence of vectors with $\|X_h(f^t)\|\le B_X<\infty$ for all $t\le T$. Then,*

$$\sum_{t=1}^{T}\|X_h(f^t)\|_{\Sigma_{t-1;h}^{-1}} \le \sqrt{2dT\log\bigg(1+\frac{TB_X^2}{\lambda d}\bigg)},$$

where the matrix $\Sigma_{t;h} := \sum_{\tau=1}^{t} X_h(f^\tau)X_h(f^\tau)^\top + \lambda\mathbb{I}$ for $t\in[T]$ and $\lambda\ge B_X^2$.

Proof. Since $\lambda \geq B_X^2$, we have that

$$\|X_h(f^t)\|^2_{\Sigma^{-1}_{t-1;h}} \leq \frac{1}{\lambda}\|X_h(f^t)\|^2 \leq 1.$$

Thus, using elliptical potential lemma (Lattimore and Szepesvári, 2020, Lemma 19.4), we get that

$$\sum_{t=1}^{T} \|X_h(f^t)\|^2_{\Sigma^{-1}_{t-1;h}} \leq 2d \log\left(1 + \frac{TB_X^2}{\lambda d}\right).$$

The desired bound follows from Jensen's inequality which implies that

$$\sum_{t=1}^{T} \|X_h(f^t)\|_{\Sigma^{-1}_{t-1;h}} \leq \sqrt{T \cdot \sum_{t=1}^{T} \|X_h(f^t)\|^2_{\Sigma^{-1}_{t-1;h}}} \leq \sqrt{2Td \log\left(1 + \frac{TB_X^2}{\lambda d}\right)}.$$

□

References

Simon S Du, Sham M Kakade, Jason D Lee, Shachar Lovett, Gaurav Mahajan, Wen Sun, and Ruosong Wang. Bilinear classes: A structural framework for provable generalization in RL. *International Conference on Machine Learning*, 2021.

Noah Golowich and Ankur Moitra. Linear bellman completeness suffices for efficient online reinforcement learning with few actions. *arXiv preprint arXiv:2406.11640*, 2024.

Nan Jiang, Akshay Krishnamurthy, Alekh Agarwal, John Langford, and Robert E Schapire. Contextual decision processes with low Bellman rank are PAC-learnable. In *International Conference on Machine Learning*, pages 1704–1713, 2017.

Chi Jin, Qinghua Liu, and Sobhan Miryoosefi. Bellman eluder dimension: New rich classes of RL problems, and sample-efficient algorithms. *Neural Information Processing Systems*, 2021.

Tor Lattimore and Csaba Szepesvári. *Bandit algorithms*. Cambridge University Press, 2020.

Gen Li, Wenhao Zhan, Jason D Lee, Yuejie Chi, and Yuxin Chen. Reward-agnostic fine-tuning: Provable statistical benefits of hybrid reinforcement learning. *Advances in Neural Information Processing Systems*, 36, 2024.

Rémi Munos and Csaba Szepesvári. Finite-time bounds for fitted value iteration. *Journal of Machine Learning Research*, 9(5), 2008.

Kevin Tan, Wei Fan, and Yuting Wei. Hybrid reinforcement learning breaks sample size barriers in linear mdps. *arXiv preprint arXiv:2408.04526*, 2024.

Runzhe Wu, Ayush Sekhari, Akshay Krishnamurthy, and Wen Sun. Computationally efficient rl under linear bellman completeness for deterministic dynamics. *arXiv preprint arXiv:2406.11810*, 2024.

Tengyang Xie, Nan Jiang, Huan Wang, Caiming Xiong, and Yu Bai. Policy finetuning: Bridging sample-efficient offline and online reinforcement learning. *Advances in neural information processing systems*, 34: 27395–27407, 2021.

Tengyang Xie, Dylan J Foster, Yu Bai, Nan Jiang, and Sham M. Kakade. The role of coverage in online reinforcement learning. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=LQIjzPdDt3q.