Technical Report: “Exploration Driven by an Optimistic Bellman Equation”

Tosatto Samuele[1], Carlo D’eramo[2], Joni Pajarinen[1], Marcello Restelli[2], and Jan Peters[1]

[1]Intelligent Autonomous Systems, Technische Univestität Darmstadt,
given_name@robot-learning.de
[2]AIRLab, Politecnico di Milano, *given_name.family_name@polimi.it*

December 14, 2018

1 Introduction

This supplement provides details on all the proofs presented in the paper "Exploration Driven by an Optimistic Bellman Equation" and shows experimental details that have been left out of the main paper due to space constraints. We refer to equations and definitions that are presented in the main paper with their assigned numbers.

First, we introduce additional notation for the classic Bellman operator and our optimistic version. We show how to derive the optimistic Bellman equation from an entropic regularized version of the bellman equation defined on a Q-value ensemble. Subsequently, we show how to interpret the optimistic Bellman equation under an intrinsic motivation perspective; we will show how the exploration bonus is related to the central moments of the approximations made. We then analyze two algorithms: Optimistic Value Iteration (OVI), which is presented mainly for theoretical reasons, introducing the optimistic Bellman operator and its properties, and optimistic Q-learning (OQL). For both algorithms we provide convergence proofs. We also show that the exploration bonus decreases according to the learning rate and state visits. Finally, we report the hyper-parameters used in our experimental setting.

1.1 Notation

This section presents the mathematical notation used in the proofs.

Definition 1 (Bellman operator)**.** *We define the Bellman operator* $\mathcal{T} : (\mathcal{S} \times \mathcal{A} \to \mathbb{R}) \to (\mathcal{S} \times \mathcal{A} \to \mathbb{R})$ *as:*

$$(\mathcal{T}Q)(s,a) = \overline{R}(s,a) + \gamma \int P(s'|s,a) \max_{a'} Q(s',a') \,\mathrm{d}\, s' \qquad \forall s, a \in \mathcal{S} \times \mathcal{A}$$

for each $Q : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$*.*

Definition 2 (Optimistic Bellman operator)**.** *We define the optimistic Bellman operator* $\mathring{\mathcal{T}}_\eta^M : (\mathcal{S} \times \mathcal{A} \to \mathbb{R})^M \to (\mathcal{S} \times \mathcal{A} \to \mathbb{R})^M$ *as:*

$$(\mathring{\mathcal{T}}_\eta Q)_i(s,a) = \overline{R}(s,a) + \eta^{-1} \log \sum_{m=1}^{M} \frac{e^{\eta\gamma \int P(s'|s,a) \max_{a'} Q_m(s',a') \,\mathrm{d}\, s'}}{M} \qquad \forall s, a, i \in \mathcal{S} \times \mathcal{A} \times \{1, \dots, M\}$$

for each $Q : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$*.*

Definition 3 (Optimal Q)**.** *We define the optimal* Q^* *as the solution:*

$$\mathcal{T}Q^* = Q^*.$$

We know from dynamic programming that Q^* *exists and it is unique.*

Definition 4 (Optimistic Value Iteration (OVI))**.** *Let* $Q = \{Q_i\}_{i=1}^M$ *be an arbitrary set of Q-functions, let* $\mathring{\mathcal{T}}_\eta^N Q$ *denote the application of* $\mathring{\mathcal{T}}_\eta$ N *times on* Q*, let OVI be the procedure that computes* $\tilde{Q} = \mathring{\mathcal{T}}_\eta^N Q$*.*

1.2 Derivation of the Optimistic Bellman Equation

In this section, we derive the optimistic Bellman equation (OBE) from an entropic-regularized version of the bellman equation. The entropic-regularization is defined on a set of Q-value function.

Theorem 1. *Consider the problem:*

$$Q_i(s,a) = \max_{b(s,a) \in \mathcal{P}^M} f\big(s, a; b(s,a)\big) - \tfrac{1}{\eta} D_{\mathrm{KL}}\big(p(s,a) \big\| u\big)$$

$$s.t. \textstyle\sum_{m=1}^{M} b_m(s,a) = 1$$

$$\forall s, a, i \in \mathcal{S} \times \mathcal{A} \times \{1, \dots, M\}$$

where $f(a,s;p) = R(s,a) + \gamma \sum_m b_m(s,a) V'_m(s,a)$, $V'_m(s,a) = \sum_{s'} P(s'|s,a) \max_{a'} Q_m(s',a')$,$u_m = 1/M$, $D_{KL}(b(s,a) \| u)$ *is the Kullback-Leibler divergence between the belief* $b(s,a)$ *and the uniform distribution* u*. From Problem 1, we can derive the optimistic Bellman equation (OBE)*

$$Q_i(s,a) = \overline{R}(s,a) + \frac{1}{\eta} \log \frac{\sum_m e^{\eta\left(\gamma \sum_{s'} P(s'|s,a) \max_{a'} Q_m(s',a')\right)}}{M} \quad \forall i \in \{1, \dots, M\}.$$

Proof. Let L_i be the Lagrangian of the i^{th} problem:

$$L_i(s,a) = \overline{R}(s,a)+\gamma \sum_m \Big(\sum_{s'} P(s'|s,a) \max_{a'} Q_m(s',a') \Big) b_m(s,a) - \frac{1}{\eta} \sum_m b_m(s,a) \log \frac{b_m(s,a)}{M^{-1}} + \lambda (\sum_m b_m(s,a) - 1) \tag{1}$$

To find the maximum of the Lagrangian, set the partial derivatives to zero. First, w.r.t. p_m:

$$\begin{aligned}
\partial_{b_m(s,a)} L_i(s,a) &= \gamma \sum_{s'} P(s'|s,a) \max_{a'} Q_m(s',a') - \frac{1}{\eta} \log \frac{b_m(s,a)}{M^{-1}} - \frac{M^{-1}}{\eta} + \lambda = 0 \\
\implies \frac{1}{\eta} \log \frac{b_m(s,a)}{M^{-1}} &= \gamma \sum_{s'} P(s'|s,a) \max_{a'} Q_m(s',a') - \frac{M^{-1}}{\eta} + \lambda \\
\implies b_m(s,a) &= M^{-1} e^{\eta \left(\gamma \sum_{s'} P(s'|s,a) \max_{a'} Q_m(s',a') - \frac{M^{-1}}{\eta} + \lambda \right)}. \qquad (2)
\end{aligned}$$

Next, set partial derivative w.r.t. λ to zero:

$$\begin{aligned}
\partial_\lambda L_i(s,a) &= \sum_m b_m(s,a) - 1 = 0 \\
\implies 1 &= \sum_m M^{-1} e^{\eta \left(\gamma \sum_{s'} P(s'|s,a) \max_{a'} Q_m(s',a') - \frac{M^{-1}}{\eta} + \lambda \right)} \\
\implies e^{-\lambda \eta} &= \sum_m M^{-1} e^{\eta \left(\gamma \sum_{s'} P(s'|s,a) \max_{a'} Q_m(s',a') - \frac{M^{-1}}{\eta} \right)} \\
\implies \lambda &= -\frac{1}{\eta} \log \sum_m M^{-1} e^{\eta \left(\gamma \sum_{s'} P(s'|s,a) \max_{a'} Q_m(s',a') - \frac{M^{-1}}{\eta} \right)}. \qquad (3)
\end{aligned}$$

Let's substitute (3) into (2):

$$\begin{aligned}
b_m(s,a) &= \frac{M^{-1} e^{\eta \left(\gamma \sum_{s'} P(s'|s,a) \max_{a'} Q_m(s',a') - \frac{M^{-1}}{\eta} \right)}}{\sum_{m'} M^{-1} e^{\eta \left(\gamma \sum_{s'} P(s'|s,a) \max_{a'} Q_{m'}(s',a') - \frac{M^{-1}}{\eta} \right)}} \\
&= \frac{e^{\eta \left(\gamma \sum_{s'} P(s'|s,a) \max_{a'} Q_m(s',a') \right)}}{\sum_{m'} e^{\eta \left(\gamma \sum_{s'} P(s'|s,a) \max_{a'} Q_{m'}(s',a') \right)}}.
\end{aligned}$$

Now that we have solved for the Lagrangian multipliers, substitute $b_m(s,a)$ into Equation (1):

$$L_i(s,a) = \overline{R}(s,a) + \gamma \sum_m \Big(\sum_{s'} P(s'|s,a) \max_{a'} Q_m(s',a') \Big) b_m(s,a)(s,a) - \frac{1}{\eta} \sum_m p_m(s,a) \log \frac{p_m(s,a)}{u_m} \tag{4}$$

to get

$$\begin{aligned}
L_i(s,a) &= \overline{R}(s,a) + \gamma \sum_m \Big(\sum_{s'} P(s'|s,a) \max_{a'} Q_m(s',a') \Big) b_m(s,a) \\
&\quad -\frac{1}{\eta} \sum_m b_m(s,a) \log \frac{e^{\eta\left(\gamma \sum_{s'} P(s'|s,a) \max_{a'} Q_m(s',a')\right)}}{\sum_{m'} e^{\eta\left(\gamma \sum_{s'} P(s'|s,a) \max_{a'} Q_{m'}(s',a')\right)}} - \frac{\log M}{\eta} \\
&= \overline{R}(s,a) + \gamma \sum_m \Big(\sum_{s'} P(s'|s,a) \max_{a'} Q_m(s',a') \Big) p_m(s,a) \\
&\quad -\frac{1}{\eta} \sum_m b_m(s,a) \left(\log e^{\eta\left(\gamma \sum_{s'} P(s'|s,a) \max_{a'} Q_m(s',a')\right)} - \log \sum_{m'} e^{\eta\left(\gamma \sum_{s'} P(s'|s,a) \max_{a'} Q_{m'}(s',a')\right)} \right) \\
&\quad -\frac{\log M}{\eta} \\
&= \overline{R}(s,a) + \frac{1}{\eta} \sum_m b_m(s,a) \log \frac{\sum_{m'} e^{\eta\left(\gamma \sum_{s'} P(s'|s,a) \max_{a'} Q_{m'}(s',a')\right)}}{M} \\
&= \overline{R}(s,a) + \frac{1}{\eta} \log \frac{\sum_{m} e^{\eta\left(\gamma \sum_{s'} P(s'|s,a) \max_{a'} Q_{m}(s',a')\right)}}{M}
\end{aligned} \tag{5}$$

Therefore (5) implies:

$$Q_i(s,a) = \overline{R}(s,a) + \frac{1}{\eta} \log \frac{\sum_{m} e^{\eta\left(\gamma \sum_{s'} P(s'|s,a) \max_{a'} Q_{m}(s',a')\right)}}{M} \qquad \forall i \in \{1, \dots, M\}.$$

□

1.3 Exploration Bonus

We can show that OBE can be seen as an intrinsic motivation technique. More in detail, is it possible to see, that the optimistic over-estimation of the Q-ensembles is equivalent to an unbiased estimation of the ensemble plus a positive term which is connected with the uncertainty between values of the ensemble. We consider the unbiased estimate of the value of the next state

$$\overline{V}'(s,a) = \sum_{m=1}^{M} \frac{V'_m(s,a)}{M}. \tag{6}$$

The exploration bonus can then be expressed as

$$\begin{aligned}
U(s,a) &= \frac{1}{\eta} \log \Big(\sum_{m=1}^{M} \frac{e^{\eta\gamma V'_m(s,a)}}{M} \Big) - \overline{V}'(s,a) \\
&= \frac{1}{\eta} \left(\log \Big(\sum_{m=1}^{M} \frac{e^{\eta\gamma V'_m(s,a)}}{M} \Big) - \eta \overline{V}'(s,a) \right) \\
&= \frac{1}{\eta} \log \Big(\sum_{m=1}^{M} \frac{e^{\eta\gamma V'_m(s,a)}}{M} e^{-\eta \overline{V}'(s,a)} \Big) \\
&= \frac{1}{\eta} \log \sum_{m=1}^{M} \frac{e^{\eta\gamma (V'_m(s,a) - \overline{V}'(s,a))}}{M}.
\end{aligned} \tag{7}$$

Note that the average over the exponential function is the sample moment generating function. Thus, Equation (7) can be rephrased as

$$U(s,a) = \lim_{N \to +\infty} \frac{1}{\eta} \log \left[1 + \sum_{n=2}^{N} \frac{(\eta\gamma)^n}{n!} \mathcal{M}_n(s,a) \right].$$

where

$$\begin{aligned}\mathcal{M}_n(s,a) &= M^{-1}\sum_{m=1}^{M}\left[\left(V'_m(s,a)-\overline{V}(s,a)\right)^n\right] \\ &= \eta\gamma\mathcal{M}_2(s,a)+O(\eta^2) \qquad (8) \\ & \qquad (9)\end{aligned}$$

denotes the n^{th}sample central moment. Note that the further simplification come from the expansion w.r.t. η.

1.4 Convergence of Value Iteration with the Optimistic Bellman Equation

In this Section, we show that value iteration using the optimistic Bellman equation (OBE), which we call optimistic value iteration (OVI), converges. First, we show that the fixed point of value iteration with OBE is identical to the fixed point of the classic Bellman equation (BE). Next, we show the max-norm contractivity of the optimistic Bellman operator. Finally, we use the fixed point and max-norm contractivity results to show that value iteration with OBE converges.

Lemma 1 (Fixed point of OBE). *If $Q_i = Q^*$ then $(\mathring{\mathcal{T}}_\eta Q)_i = Q^* \quad \forall i \in \{1,\dots M\}$ where Q^* is the unique fixed point of the classic BE .*

Proof.

$$\begin{aligned}(\tilde{T}^*Q^*)(s,a) &= \overline{R}(s,a)+\frac{1}{\eta}\log\frac{1}{M}\sum_{i=1}^{M}e^{\eta\gamma\int P(s'|s,a)\max_{a'}Q^*(s',a')\,\mathrm{d}\,s'} \\ &= \overline{R}(s,a)+\frac{1}{\eta}\log e^{\eta\gamma\int P(s'|s,a)\max_{a'}Q^*(s',a')\,\mathrm{d}\,s'} \\ &= \overline{R}(s,a)+\gamma\int P(s'|s,a)\max_{a'}Q^*(s',a')\,\mathrm{d}\,s' \\ &= (T^*Q^*)(s,a) \\ &= Q^*(s,a)\end{aligned}$$

□

Lemma 2 (Max-Norm contractivity of the optimistic Bellman operator). *Given $\{Q_{1,k}\}_{k=1}^{M}$, $\{Q_{2,k}\}_{k=1}^{M}$, and $\delta > 0$ such that*

$$\|Q_{1,k}-Q_{2,k}\|_\infty \le \delta \qquad \forall k \in \{1,\dots,M\}$$

implies that:

$$\|(\mathring{\mathcal{T}}_\eta Q_1)_k-(\mathring{\mathcal{T}}_\eta Q_2)_k\|_\infty \le \gamma\delta \qquad \forall k \in \{1,\dots,M\}.$$

Proof.

$$\begin{aligned}(\mathring{\mathcal{T}}_\eta Q_1)_k(s,a)-(\mathring{\mathcal{T}}_\eta Q_2)_k(s,a) &= \overline{R}(s,a)+\frac{1}{\eta}\log\frac{1}{M}\sum_{i=1}^{M}e^{\eta\gamma\int P(s'|s,a)\max_{a'}Q_{1,i}(s',a')\,\mathrm{d}\,s'} \\ &\quad -\overline{R}(s,a)-\frac{1}{\eta}\log\frac{1}{M}\sum_{i=1}^{M}e^{\eta\gamma\int P(s'|s,a)\max_{a'}Q_{2,i}(s',a')\,\mathrm{d}\,s'} \\ &= \frac{1}{\eta}\log\frac{1}{M}\sum_{i=1}^{M}e^{\eta\gamma\int P(s'|s,a)(\max_{a'}Q_{1,i}(s',a')-\max_{a'}Q_{2,i}(s',a'))\,\mathrm{d}\,s'} \\ &\le \frac{1}{\eta}\log\frac{1}{M}\sum_{i=1}^{M}e^{\eta\gamma\int P(s'|s,a)\delta\,\mathrm{d}\,s'} \\ &= \gamma\delta\end{aligned}$$

□

Theorem 2 (Convergence of OBE). *If*

$$|Q_i(s,a)-Q^*(s,a)| \le \epsilon \qquad s,a,i \in \mathcal{S}\times\mathcal{A}\times\{1,\dots,M\} \tag{10}$$

then

$$|(\mathring{\mathcal{T}}_\eta Q)_i(s,a)-Q^*(s,a)| \le \gamma\epsilon \qquad s,a,i \in \mathcal{S}\times\mathcal{A}\times\{1,\dots,M\}. \tag{11}$$

Note that this implies that given an initial set of $Q = \{Q_i\}_{i=1}^{M}$, $\lim_{N\to\infty} \mathring{\mathcal{T}}_\eta^N Q = Q^*$, therefore implies the convergence of OVI.

Proof.

$$\begin{aligned} \left|(\mathring{\mathcal{T}}_\eta Q)_i(s,a) - Q^*(s,a)\right| &= \left|(\mathring{\mathcal{T}}_\eta * Q)_i(s,a) - \mathring{\mathcal{T}}_\eta Q^*(s,a)\right| \\ &\leq \gamma\epsilon \end{aligned}$$

□

We note that OVI converges with the same rate as VI.

1.5 Optimistic Q-Learning

This section provides a more detailed proof of optimistic Q-learning (OQL) converging to $Q(s,a)^*$. We first show that the exploration bonus vanishes in OQL and then give the main proof of convergence.

Definition 5 (Optimistic Q-learning - *theoretical version*). [1]

$$\begin{cases} Q_{j,t+1}(s,a) = (1-\alpha_t)Q_{i,t}(s,a) & \\ \qquad\qquad\qquad +\alpha_t\Big(r_t + \gamma\frac{1}{M}\sum_{j=1}^{M}\max_{a'} Q_{j,t}(s_{t+1},a')\Big) & \text{if} s = s_t \wedge a = a_t \\ Q_{i,t+1}(s,a) = Q_{i,t}(s,a) & \text{otherwise} \end{cases}$$

Theorem 3 (Vanishing bonus for optimistic Q-learning). *Let's consider optimistic Q-Learning (OQL) described in Definition 5. Let's suppose to have a set of Q_i. Each entry in the table at time $t = 0$ has central moment $\mathcal{M}_{0,n} \in \mathbb{R}$. If we consider a specific entry (s,a), and a sequence of learning rate $\{\alpha_t\}_{t=0}^{T}$ where $\alpha_t \in [0,1]$, then*

$$\mathcal{M}_{T+1,n}(s,a) = \prod_{t=0}^{T}(1-\alpha_t)^n \mathcal{M}_{0,n}(s,a) \tag{12}$$

where $\mathcal{M}_{T+1,n}(s,a)$ is the n^{th} central moment of the entry (s,a) at time $T+1$.

Proof. Please, note that we always update an entry (s,a) with the same value for all the M tables. We can refer to the sequence of updates to a single state-action pair (s,a) as $\{y_t\}_{t=0}^{T}$, where each value belongs to $\mathbb{R}$. Now, let's consider the process of updating of the entry s,a

$$Q_{t+1,i}(s,a) = (1-\alpha_t)Q_{t,i}(s,a) + \alpha_t y_t \qquad \forall i \in \{1,\ldots M\}. \tag{13}$$

We can write the central moments at time $t+1$ as

$$\begin{aligned} \mathcal{M}_{t+1,n}(s,a) &= M^{-1}\sum_m \Big(Q_{t+1,m}(s,a) - M^{-1}\sum_i Q_{t+1,i}(s,a)\Big)^n \\ &= M^{-1}\sum_m \Big((1-\alpha_t)Q_{t,m}(s,a) + \alpha_t - M^{-1}\sum_i \big((1-\alpha_t y_t)Q_{t,i}(s,a) + \alpha_t\big)\Big)^n \\ &= M^{-1}\sum_m \Big((1-\alpha_t)Q_{t,m}(s,a) - M^{-1}\sum_i (1-\alpha_t y_t)Q_{t,i}(s,a)\Big)^n \\ &= (1-\alpha_t)^n M^{-1}\sum_m \Big(Q_{t,m}(s,a) - M^{-1}\sum_i Q_{t,i}(s,a)\Big)^n \\ &= (1-\alpha_t)^n \mathcal{M}_{t,n}(s,a) \end{aligned}$$

and therefore, unfolding the recursion,

$$\mathcal{M}_{T+1,n}(s,a) = \prod_{t=0}^{T}(1-\alpha_t)^n \mathcal{M}_{0,n}(s,a)$$

□

Next, we prove the convergence of OQL. Our proof is based on the work of [2], which relies on the following theorem [1]:

[1] We use α_t as a shortcut for $\alpha_t(s,a)$.

Theorem 4. *The random process $\{\Delta_t\}$ taking values in $\mathbb{R}$ and defined as:*

$$\Delta_{t+1}(x) = (1 - \alpha_t(x))\Delta_t(x) + \alpha_t F_t(x) \tag{14}$$

converges to zero w.p. 1 under the following assumptions:

- $0 \leq \alpha_t \leq 1$, $\sum_t \alpha_t(x) = \infty$ *and* $\sum_t \alpha_t^2(x) < \infty$
- $\| \mathbb{E}[F_t(x)|\mathcal{F}_t]\|_W \leq \gamma\|\Delta_t\|_W$ *with* $0 \leq \gamma < 1$
- $\mathrm{Var}[F_t(x)|\mathcal{F}_t] \leq C(1 + \|\Delta_t\|_W^2)$ *for* $C > 0$.

We are now ready to prove the convergence of optimistic Q-learning.

Theorem 5 (Convergence of Optimistic Q-learning)**.** *Let us consider the algorithm provided in Definition 5. Suppose that the rewards are bounded, and consider a learning rate $\alpha_t(s, a)$ which satisfies $0 \leq \alpha_t \leq 1$, $\sum_t \alpha_t(x) = \infty$ and $\sum_t \alpha_t^2(x) < \infty$. Suppose that each state action pair is visited infinitely many times, then, $\lim_{t\to\infty} Q_{i,t}(s, a) \to Q^*(s, a) \forall i \in \{1, \dots, M\}$ with probability 1. Then the algorithm converges.*

Proof. Let's consider the following stochastic process:

$$Q_{i,t+1}(s, a) = (1 - \alpha_t)Q_{i,t}(s, a) + \alpha_t\Big(r_t + \frac{1}{\eta}\log M^{-1}\sum_{j=1}^{M} e^{\gamma \max_{a'} Qj,t(s_{t+1},a')}\Big)$$

with $\alpha_t = \alpha_t(s, a)$ (for brevity), and coherent with the assumpions. Let's rename $\Delta_{i,t}(s, a) = Q_{i,t}(s, a) - Q^*(s, a)$ where $Q^*(s, a)$ is the fixed point of $\mathring{\mathcal{T}}_\eta$. We can now write:

$$\Delta_{i,t+1}(s, a) = (1 - \alpha_t)\Delta_{i,t}(s, a) + \alpha_t\Big(r_t + \frac{1}{\eta}\log M^{-1}\sum_{j=1}^{M} e^{\gamma \max_{a'} Qj,t(s_{t+1},a')} - Q^*(s, a)\Big).$$

We can rename $F_t(s, a) = r_t + \frac{1}{\eta}\log M^{-1}\sum_{j=1}^{M} e^{\gamma \max_{a'} Qj,t(s_{t+1},a')} - Q^*(s, a)$ and observe that $\mathbb{E}[F_t(s, a)] = \mathring{\mathcal{T}}_\eta Q_t(s, a) - Q^*(s, a)$, and subsequently, thanks to the max-norm contraction of the optimistic Bellman operator,

$$\begin{aligned} \| \mathbb{E}[F_t(s, a)]\|_\infty &\leq \gamma\|Q_{i,t} - Q^*\|_\infty \\ &= \gamma\|\Delta_{i,t+1}(s, a)\|_\infty \qquad \forall i \in \{1, \dots, M\}. \end{aligned}$$

Then, noticing $\mathrm{Var}[F_t(s, a)] = \mathrm{Var}[F_t(s, a) - Q^*] = \mathrm{Var}[r_t + \frac{1}{\eta}\log M^{-1}\sum_{j=1}^{M} e^{\gamma \max_{a'} Qj,t(s_{t+1},a')}]$ which, considering that the rewards are assumed to be bounded, leads to:

$$\exists C : \mathrm{Var}[F_t(s, a)] \leq C(1 + \|\Delta_{i,t}\|) \qquad \forall i \in \{1, \dots, M\}. \tag{15}$$

Using Theorem 4 we can therefore say that $Q_t(s, a)$ converges to $Q^*(s, a)$ w.p. 1. □

1.6 Details on the Experiments

In this section, we provide details on the exact form of bootstrapped Q-Learning and optimistic Q-learning update rules. Furthermore, we describe how hyper-parameters were chosen in the experiments and what kind of neural network structure was used in the experiments with neural network based function approximation.

We define Bootstrapped Q-Learning (BQL) with the following update rule

Definition 6 (Bootstrapped Q-learning)**.** [2]

$$\begin{cases} Q_{i,t+1}(s, a) = (1 - \alpha_t)Q_{i,t}(s, a) & \\ \qquad\qquad +\alpha_t\Big(r_t + \frac{1}{\eta}\log M^{-1}\sum_{j=1}^{M} e^{\gamma \max_{a'} Qj,t(s_{t+1},a')}\Big) & \text{if} s = s_t \wedge a = a_t \\ Q_{i,t+1}(s, a) = Q_{i,t}(s, a) & \text{otherwise} \end{cases}$$

and OQL with

[2] We use α_t as a shortcut for $\alpha_t(s, a)$.

Definition 7 (Optimistic Q-learning - *empirical analysis version*)**.**

$$\begin{cases} Q_{j,t+1}(s,a) = (1-\alpha_t)Q_{i,t}(s,a) & \\ \qquad +\alpha_t\Big(r_t + \frac{1}{\eta}\log\frac{1}{M-1}\sum_{j=2}^{M} e^{\gamma \max_{a'} Q_{j,t}(s_{t+1},a')+Q_1(s_{t+1},a')} & \\ \qquad -\gamma\frac{1}{M-1}\sum_{j=2}^{M}\max_{a'} Qj,t(s_{t+1},a')\Big) & \text{if} s = s_t \wedge a = a_t \wedge j = 1 \\ Q_{j,t+1}(s,a) = (1-\alpha_t)Q_{i,t}(s,a) & \\ \qquad +\alpha_t\Big(r_t + \gamma\frac{1}{M-1}\sum_{j=2}^{M}\max_{a'} Q_{j,t}(s_{t+1},a')\Big) & \text{if} s = s_t \wedge a = a_t \wedge j \geq 2 \\ Q_{i,t+1}(s,a) = Q_{i,t}(s,a) & \text{otherwise} \end{cases}$$

using the "explicit exploration" formulation, in order to enable the evaluation with unbiased Q values. The settings provided by Table 1 are fixed for all the different environments.

Parameter	**QL**	**OIQL**	**BQL**	**OQL**
Number of approximators	1	1	10	10
Initialization	$\mathcal{N}(0,2)$	$R_{\max}$	$\mathcal{N}(0,2)$	$\mathcal{N}(0,2)$
Learning rate	0.15	0.15	0.15	0.15
ϵ-greedy	0.01	0.01	0.01	0.01

Table 1: Setting used for tabular algorithms.

1.6.1 DQN and ODQN

Acrobot configuration. For Acrobot, we used a single-layer neural network as base component of the ensemble. The input of the neural newtork is 4-dimensional (and corresponds to the dimension of the state space), and the output has 2 dimensions, corresponding to the two possible actions. Table 4 shows the fixed hyper-parameters for Acrobot. Additionally, we performed a grid search over the number of neurons in the hidden layer, and for the "bootstrapped mask" (see Table 2). We measured both the mean return averaged over all episodes denoted by "avg" (which should give an idea about how fast an algorithm can improve performance w.r.t. the number of samples), and also the mean of the final performance. For both ODQN and BDQN we selected the hyper-parameters corresponding to the best average performance of BDQN yielding yielding 100 neurons and a bootstrap mask of 0.5 for the final performance evaluation shown in the main paper. For ODQN we use $\chi = 0.25$ and $\iota_{\max} = 1$.

Neurons	**Bootstrapped Mask**	**BDQN** avg	**BDQN** final	**ODQN** avg	**ODQN** final
100	0.5	-116.96^*	-84.63	-115.25	-86.04
100	1.0	-129.62	-86.91	-129.06	-95.60
150	0.5	-123.21	-85.04	-122.10	-80.38
150	1.0	-136.25	-89.50	-138.38	-83.62
200	0.5	-123.47	-87.89	-125.72	-84.41
200	1.0	-143.05	-87.70	-148.64	-81.91
300	0.5	-129.26	-82.60	-131.24	-81.12
300	1.0	-150.10	-83.98	-151.14	-86.30
400	0.5	-133.01	-81.43	-135.30	-83.82
400	1.0	-154.70	-87.58	-158.38	-86.55

Table 2: Tested hyper-parameters "Neurons" and "Bootstrapped Mask" for Acrobot with corresponding evaluations.

Taxi configuration. For Taxi, we decided to encode the state as a 2-dimensional grid, selecting only the position of the agent to 1 and the rest to zero, and additionally we provide a one-dimensional vector of length 3 providing the information about which flags where collected. We decided to use a shared convolutional layer with kernel of 2 and stride 1, in order to process the vector and reduce the dimension. Above the convolutional layer, we apply a different hidden layer for each component of the ensemble. The output of each component is 4-dimensional, corresponding to the four possible actions. Most of the parameters are chosen without any optimization (see Table 4), except for the number of neurons in the hidden layer, and for the "bootstrapped mask". We performed a grid search over these two parameters, measuring both the mean return averaged over all episodes and also the mean of the final performance, similarly to the Acrobot evaluation (please, see Table 3). For both ODQN and BDQN, we selected the hyper-parameters corresponding to the best average performance of BDQN yielding 200 neurons and a bootstrap mask of 0.5 for the final performance evaluation shown in the main paper.

Neurons	**Bootstrapped Mask**	**BDQN** avg	**BDQN** final	**ODQN** avg	**ODQN** final
200	0.5	9.51*	12.05	**10.05**	**14.25**
200	1.0	6.16	7.80	**8.63**	**12.00**
300	0.5	9.14	**10.95**	**9.63**	9.65
300	1.0	**9.36**	9.65	9.27	**10.55**
400	0.5	7.21	8.4	**10.30**	**13.50**
400	1.0	6.87	9.2	**9.42**	**11.25**

Table 3: Tested hyper-parameters "Neurons" and "Bootstrapped Mask" for Taxi with corresponding evaluations.

Parameter	**Acrobot**	**Taxi**
Number of approximators	10	10
Shared conv. layer	no	yes
Number of layers	1	1
Number of neurons	100	200
Activation function	relu	relu
Initialization	Glorot Uniform	Glorot Uniform
Loss	MSE	Huber Loss
Optimization	Adam	RMSProp
Learning rate	0.001	0.00075
Decay (only RMSProp)	none	0.95
Batch size	32	100
Max replay-memory size	5000	100000
Target update frequency	600	100
p-mask	0.5	0.5
ϵ-greedy (training)	0.0	0.05
ϵ-greedy (evaluation)	0.0	0.0
Evaluation frequency	3000	5000
Total training steps	250000	400000

Table 4: Common hyper-parameters for BDQN and ODQN.

References

[1] T. Jaakkola, M. I. Jordan, and S. P. Singh. Convergence of stochastic iterative dynamic programming algorithms. In *Advances in neural information processing systems*, pages 703–710, 1994.

[2] F. S. Melo. Convergence of q-learning: A simple proof. *Institute Of Systems and Robotics, Tech. Rep*, pages 1–4, 2001.