Uncertainty Based Exploration in Reinforcement Learning: Analyzing the Robustness of Bayesian Deep Q-Networks

Sagi Schwartz[1], Supervised by Pascal van der Vaart[1] and Neil Yorke−Smith[1]

[1]EEMCS, Delft University of Technology

Background

Reinforcement learning suffers from poor sample efficiency which makes naïve approaches extremely computationally expensive. One of the remedies is to use a Bayesian approach coupled with Thompson sampling to achieve better performance – this is implemented in an algorithm called **Bayesian Deep Q-Networks (BDQN)** [1].

BDQN replaces the last layer of double DQN with a Bayesian linear regression (BLR) layer, which allows efficient posterior updates and uncertainty aware exploration. The posterior distribution of the weights is estimated as:

$$\overline{w}_a := \frac{1}{\sigma_\epsilon^2}\mathrm{Cov}_a \Phi_a^\theta \mathbf{y}_a$$

$$\mathrm{Cov}_a := \left(\frac{1}{\sigma_\epsilon^2}\Phi_a^\theta {\Phi_a^\theta}^\top + \frac{1}{\sigma^2} I\right)^{-1}$$

We also define the uncertainty as the variance of the posterior distribution of the Q-values:

$$Q(x,a) \mid \mathcal{D}_a \sim \mathcal{N}\left(\frac{1}{\sigma_\epsilon^2}\phi(x)^\top \mathrm{Cov}_a \Phi_a \mathbf{y}_a,\ \phi(x)^\top \mathrm{Cov}_a \phi(x)\right)$$

Efficient exploration methods such as BDQN are often only tested in complex environments such as Atari games and lack hyperparameter sensitivity and performance analysis in simpler environments.

Such environments include **Deep Sea** (varying difficulties), **MNIST Bandit** (contextual bandit) and **Cart Pole** (low exploration), suggested in Osband et al. [2]. Additionally, Adkins et al. [3] developed a framework to assess RL algorithms **hyperparameter sensitivity**:

$$\Phi(\omega) \doteq \frac{1}{|\mathcal{E}|}\sum_{e\in\mathcal{E}} \max_{h\in H^\omega} \Gamma(\omega,e,h) - \max_{h\in H^\omega} \frac{1}{|\mathcal{E}|}\sum_{e\in\mathcal{E}} \Gamma(\omega,e,h)$$

Research Questions

1. **Performance**: How does BDQN's exploration method perform in solving reinforcement learning environments, compared to the ϵ-greedy exploration strategy?
2. **Sensitivity**: To which BLR hyperparameters is BDQN sensitive? What is the effect of each BLR hyperparameter?
3. **Transferability**: Does BDQN performance transfer well across tasks? Do optimal hyperparameter values remain stable or vary?

Methodology

Environments: Deep Sea (10×10, 20×20) - tests exploration in sparse-reward settings. Cart Pole - evaluates performance in low-exploration tasks. MNIST Bandit - assesses exploitation in a contextual bandit.

Baseline: DQN with ϵ-greedy (shared architecture/hyperparameters except exploration).

Hyperparameters:

BDQN: 6 exploration-specific (prior and noise σ, batch size, posterior update frequency, weight sampling frequency and forgetting factor). DQN: 3 ϵ-greedy hyperparameters (start/end ϵ, exploration fraction).

Primary metrics: Average cumulative reward (score) for performance, Q-value variance (uncertainty) and exploration steps (where expected action is different from sampled action) for further analysis.

Method: Hyperparameter sweeps to identify influence of hyperparameters on performance and exploration. Applying framework by Adkins et al. [3] to compute hyperparameter sensitivity.

Experiments and Results

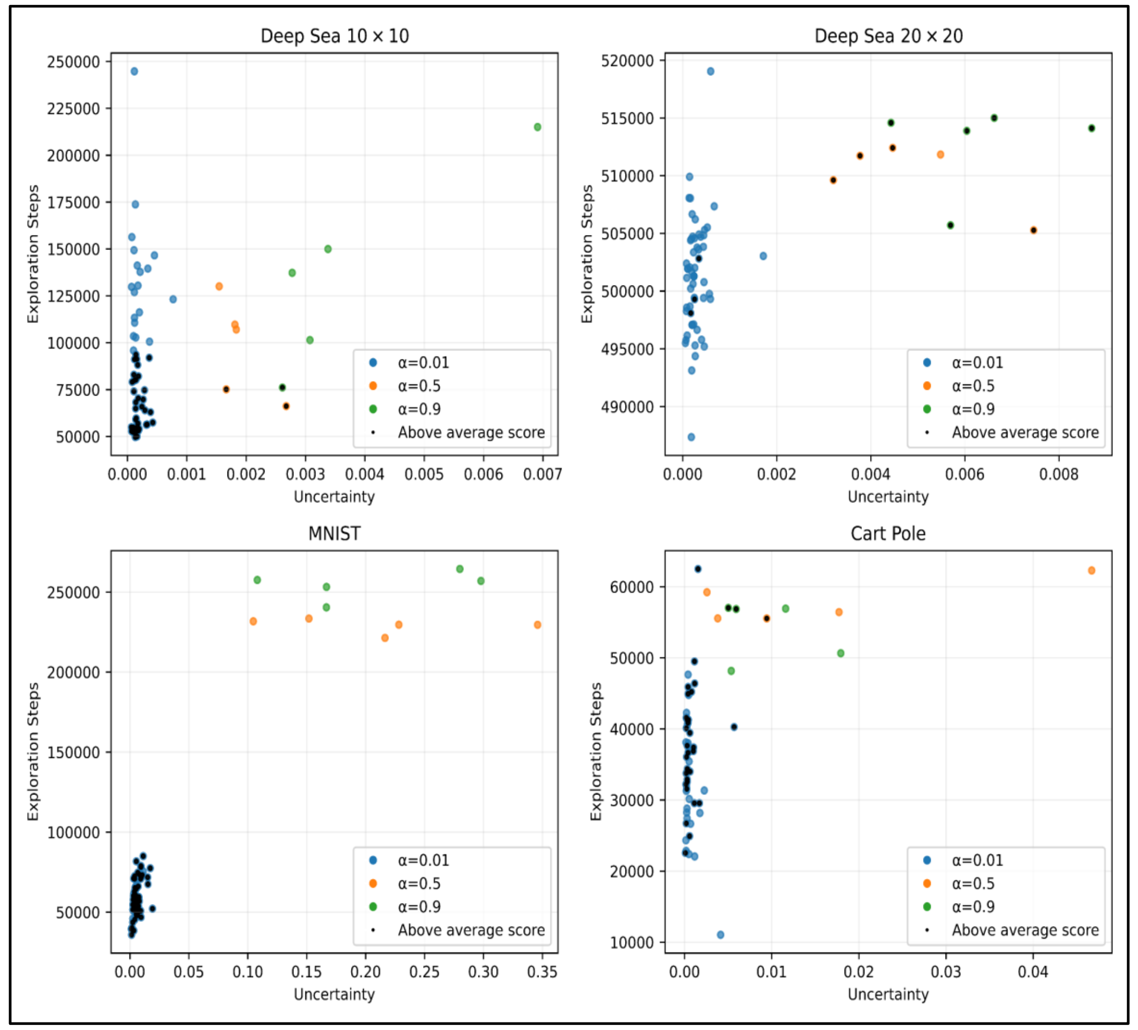

Figure 1: Exploration as a function of uncertainty of all runs in tested environment. Different values of alpha are highlighted, along with runs with an above average score which are filled with a dark dot. Clearly, smaller alpha is correlated with less exploration (benefits simple environments) and larger alphas are correlated with more exploration (benefits exploration heavy Deep Sea 20x20)

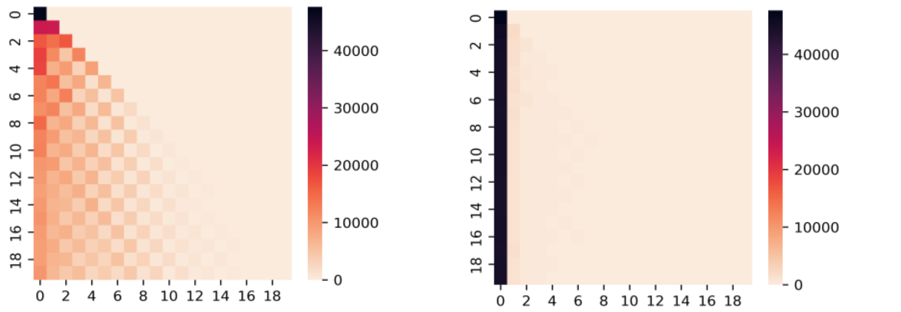

Figure 2: BDQN (left) and DQN (right) state coverage in Deep Sea of size 20x20. BDQN manages to reach the reward, while DQN does not.

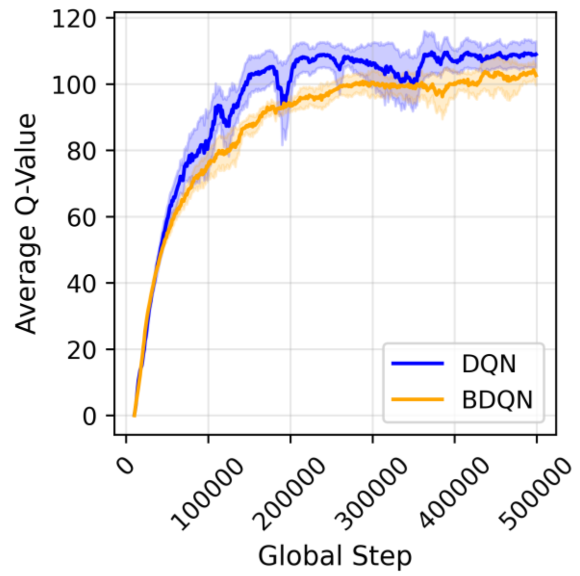

Figure 3: Average learned Q-values of sampled experiences from replay buffer of BDQN and DQN in Cart Pole with 95% CI, shows a slight difference between the algorithms. In the other environments that difference diminishes.

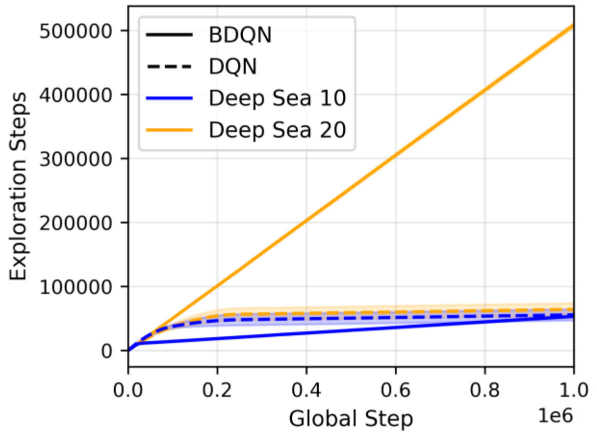

Figure 4: Comparison of BDQN and DQN exploration in Deep Sea environments with 95% CI. In a challenging environment, BDQN explores significantly more than DQN. In simpler environments however, BDQN might explore even less.

Limitations

The research is limited to three environments, which restricts the generalizability of the findings to more diverse reinforcement learning tasks. Hyperparameter tuning was more thorough for BDQN than for the ϵ-greedy baseline, potentially skewing performance comparisons. Additionally, the sensitivity score was based on a relatively new framework and conducted at a smaller scale than prior work, limiting some conclusions. Future research should expand BDQN evaluation to a broader set of environments and conduct more extensive hyperparameter studies.

Conclusion

We addressed three key questions:

Performance: BDQN outperforms ϵ-greedy DQN in exploration-heavy tasks (e.g., Deep Sea 20×20 but offers no consistent advantage in simpler environments.

Sensitivity: BDQN is highly sensitive to the forgetting factor (α), with other BLR hyperparameters having moderate effects on uncertainty and exploration. We analyzed how each hyperparameter influences exploration.

Transferability: BDQN's performance is task-dependent and requires careful tuning. Optimal hyperparameters vary significantly across tasks, making transferability non-trivial.

References

[1] Kamyar Azizzadenesheli and Animashree Anandkumar. Efficient Exploration throughBayesian Deep Q-Networks. arXiv:1802.04412 [cs].
[2] Djallan Osband et al. Behaviour Suite for Reinforcement Learning. arXiv:1908.03568 [cs].Feb. 2020. doi: 10.48550/arXiv.1908.03568
[3] Jacob Adkins, Michael Bowling, and Adam White. A Method for Evaluating Hyperpa-rameter Sensitivity in Reinforcement Learning. arXiv:2412.07165 [cs]. Feb. 2025. doi:10.48550/arXiv.2412.07165.