ESANN 2026 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium) and online event, 22-24 April 2026, i6doc.com publ., ISBN 9782875870964. Available from http://www.i6doc.com/en/.

Point-wise Q-value maximization for converging Q-learning in continuous state-spaces

Philipp Wissmann[1,2], Daniel Hein[1], Steffen Udluft[1], and Thomas Runkler[1,2] *

1- Siemens AG, Munich, Germany

2- TU Munich (TUM), Munich, Germany

Abstract. This paper introduces a novel Q-learning framework to address instabilities in offline reinforcement learning with continuous state spaces. We identify the recurring collapse of Q-value targets as core challenge and propose a stabilization technique that replaces the iteration-wise targets with their point-wise maximum across iterations. This approach enforces convergence and fully mitigates recursive errors. We show that a performance metric linking Q-values to policy performance is directly available. Our findings represent a first step toward stabilizing Q-learning in challenging settings and highlight the potential of model-based approaches.

1 Introduction & related work

Reinforcement learning (RL) plays a fundamental role in modern machine learning, providing a framework to derive policies that optimize long-term returns. Among the many approaches, Q-learning has emerged as one of the most foundational paradigms. Its intertwining between evaluating a policy and refining it has made Q-learning both conceptually simple and practically effective [1, 2].

A key strength of Q-learning lies in its iterative updating of state-action values, *Q-values*, which represents the expected future returns of actions taken from a specific state. Originally designed for table-based Markov Decision Processes (MDPs), its convergence guarantee under suitable conditions has expanded its utility to real-world applications such as robotics and industrial control.

However, despite its theoretical appeal, deploying Q-learning in industrial applications, introduces substantial challenges, *i.e.*, conducting learning offline due to safety concerns, high operational costs, or the risk of critical errors during exploratory training. Consequently, learning frequently relies on static, limited datasets collected under controlled conditions with unknown policies. Furthermore, accurate performance estimation of policies before deployment is crucial. A lot of modern industrial problems operate in continuous state-action spaces, necessitating the use of function approximators, such as neural networks (NNs), to enable Q-learning. These constraints lead to the interplay between bootstrapping, off-policy learning, and function approximation, also known as the deadly triad. This combination frequently is especially prone to an amplified overestimation bias, where errors in Q-value updates accumulate across iterations and inflate its estimates, leading to misinformed policies [3, 4].

*The project this report is based on was supported with funds from the German Federal Ministry of Research, Technology and Space under project number 16IS24087A. The sole responsibility for the report's contents lies with the authors.

ESANN 2026 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium) and online event, 22-24 April 2026, i6doc.com publ., ISBN 9782875870964. Available from http://www.i6doc.com/en/.

Prior studies have demonstrated that Q-learning's instability is not merely the result of suboptimal algorithmic design but of fundamental structural issues within its framework. Relying on function approximators introduces significant challenges, particularly in continuous state spaces where the true Q-function often exhibits sharp gradients or discontinuities. These features are notoriously difficult, if not impossible, to model accurately. This mismatch between the approximated and true Q-function causes inconsistency, where policies do not reproduce the Q-values predicted by the learned Q-function [5, 6, 7].

This paper introduces a novel framework to tackle the inherent instabilities of Q-learning in offline RL with continuous state spaces. The proposed method substitutes Q-value targets with model-based rollouts and addresses their recurring collapse by replacing the iteration-wise target setting with a stabilization technique that computes point-wise Q-value maxima across all iterations. By eliminating bootstrapping, the framework ensures convergence by mitigating the recursive amplification of errors like overestimation bias. Additionally, it links Q-values to policy performance, enabling reliable insights throughout training. To illustrate the effectiveness of the proposed approach, we apply it on the well-established cart-pole benchmark. The experiments show that our method stabilizes Q-value estimation, yielding consistent and reliable policies and addressing key limitations of traditional Q-learning.

2 A unified perspective on Q-values

Q-learning theory To better understand its structural limitations, we revisit the theoretical underpinnings of Q-learning. For readability, all equations are presented for deterministic policies and transitions. The Q-function, defined recursively by the Bellman equation, represents the expected cumulative reward of executing a given action and subsequently following a specified policy:

$$Q^{\pi}(s_t, a_t) = r_t + \gamma Q^{\pi}(s_{t+1}, \pi(s_{t+1})). \tag{1}$$

Q-learning seeks to iteratively learn the optimal Q-function Q^* for the optimal policy π^* by leveraging the contraction property of the Bellman operator, yielding the following update rule:

$$Q_{i+1}(s_t, a_t) \leftarrow r_t + \gamma \max_{a_{t+1}} Q_i(s_{t+1}, a_{t+1}). \tag{2}$$

These updates utilize bootstrapping, where prior Q-value estimates are used to propagate information from future rewards back to earlier states. In table-based MDPs, this process benefits from robust convergence guarantees, provided there are adequate exploration and properly tuned learning schedules.

However, when Q-learning is extended to continuous or large state-action spaces with function approximation, it suffers from critical limitations. Errors inherent to function approximators propagate through the bootstrapping mechanism, compounding across iterations. Their dependency can be directly seen:

$$Q^{\epsilon}_{i+1}(s_t, a_t) \leftarrow r_t + \gamma \max_{a_{t+1}} \tilde{Q}_i(s_{t+1}, a_{t+1}), \text{ with } \tilde{Q}_i(s, a) = Q^{\epsilon}_i(s, a) + \epsilon_i(s, a). \tag{3}$$

ESANN 2026 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium) and online event, 22-24 April 2026, i6doc.com publ., ISBN 9782875870964. Available from http://www.i6doc.com/en/.

This recursive error can amplify itself and destabilize learning, especially when paired with the max operator. Thus, not only leading to extrapolation errors, but also introducing amplified overestimation: random errors in Q-value predictions lead to systematically inflated estimates. The iterative nature of Q-learning further compounds this bias in scenarios involving function approximation, particularly in long-horizon problems with high discount factors ($\gamma \approx 1$).

Existing heuristics Although prior research has targeted these issues, existing methods primarily tackle symptoms rather than the underlying structural causes. Heuristic approaches like delayed updates, double Q-learning, and target clipping, offer partial remedies for overestimation bias [8]. Delayed updates slow down the propagation of errors, double Q-learning leverages the minimum of two approximated Q-functions for the bootstrapping in Equation (2) and clipping strategies can make sure errors do not accumulate above certain thresholds. Similarly, regularization-based techniques, such as CQL [9], aim to mitigate these issues by attempting to learn a lower bound for Q-values. Behavior-constraining methods like IQL [10], BCQ [11] and MOOSE [12] confine actions to remain within the training dataset. However, these approaches fail to fundamentally resolve the instability and inaccuracies in calculating Q-value targets.

Model-based Q-value estimation Model-based methods explicitly approximate transition dynamics, either using learned models or predefined equations. This approach enables Q-value estimation through rollouts [13], thus avoiding reliance on direct bootstrapping:

$$\tilde{Q}^{\pi}_{\mathrm{MB}}(s,a) = R(s,a,\tilde{s}_1) + \sum_{k=1}^{K-1} \gamma^k R(\tilde{s}_k, \pi(\tilde{s}_k), \tilde{s}_{k+1}), \tag{4}$$

where $\tilde{s}_{k+1} = M(\tilde{s}_k, \pi(\tilde{s}_k))$, with both transition model M and reward model R learned from the offline dataset. Thus the Q-learning update rule yields:

$$Q_{i+1}(s_t,a_t) \leftarrow r_t + \gamma \tilde{Q}^{\pi}_{\mathrm{MB}}(s_{t+1}, \pi(s_{t+1})), \text{ with } \pi(s) = \arg\max_a Q_i(s,a). \tag{5}$$

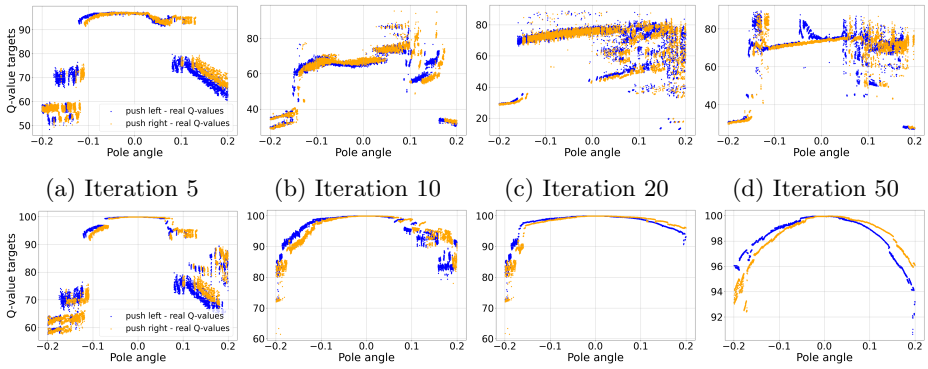
In [13], it has been shown that long rollouts on transition dynamics models can actually produce better Q-value estimates than model-free methods. Also, amplified overestimation and recursive error propagation are effectively mitigated.

3 Forcing Q-value target convergence

Even with perfect Q-value estimates, Q-learning suffers from iteration-wise instability, as highlighted in prior studies [5, 6]. The recurring collapse of targets and the underlying degradation in policy quality pose a significant challenge in stabilizing the algorithm. Rather than fitting the targets like in Equation (5) calculated for the last iteration's policy, we propose utilizing the point-wise maximum of all targets computed across iterations:

$$Q_{i+1}(s_t,a_t) \leftarrow r_t + \gamma \max_{j \le i} \tilde{Q}^{\pi}_{j,\mathrm{MB}}(s_{t+1}, \pi(s_{t+1})). \tag{6}$$

ESANN 2026 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium) and online event, 22-24 April 2026, i6doc.com publ., ISBN 9782875870964. Available from http://www.i6doc.com/en/.



(a) Iteration 5 (b) Iteration 10 (c) Iteration 20 (d) Iteration 50

(e) Iteration 5 (f) Iteration 10 (g) Iteration 20 (h) Iteration 50

Fig. 1: Each plot depicts the iteration's Q-value targets for 10,000 different pole angle values with cart position, cart velocity and pole velocity fixed at 0.0.

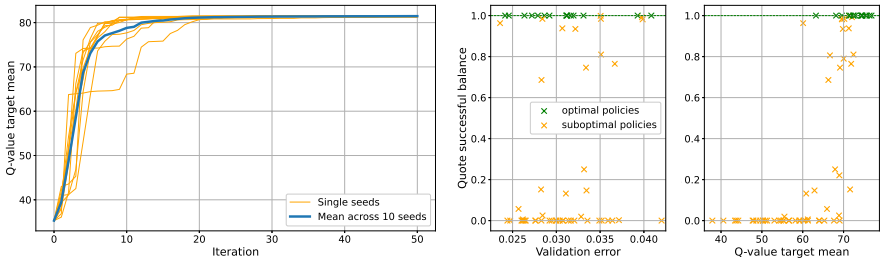
To evaluate the effectiveness of this approach, we turn to the well-established cart-pole benchmark. Its binary action space creates a Q-value landscape particularly prone to discontinuities and sharp gradients, caused by fast trajectory divergence from similar starting states. These discontinuities amplify the challenges faced when applying function approximation, intensifying instability and complicating convergence. The cart-pole environment features a four-dimensional continuous state space, *i.e.*, position x, velocity $\dot{x}$, angle θ, and angular velocity $\dot{\theta}$. A dataset of 20,000 observation tuples in the form (s_t, a_t, s_{t+1}, r_t) was generated by using a random policy on *Gymnasium*'s[1] *CartPole-v1*. The reward function assigns a value of 1 for an upright pole with the cart in the center and decreases quadratically along cart position and pole angle relative to their termination bounds, *i.e.*, $r = (1 - (x/2.4)^2 + 1 - (\theta/0.2095)^2)/2$.

Targets for each iteration were calculated using Equation (6) and Equation (5) aas baseline. To isolate the effect of our point-wise maximum approach, we used the benchmark's equations as error-free transition dynamics model M in Equation (4). The Q-values were approximated with NNs in a supervised learning setup. The Adam optimizer was employed with a learning rate of 0.01, a mini-batch size of 100, and mean squared error as the loss function. The NN followed a 5-64-1 architecture with ReLU activations, taking state-action pairs as inputs. The dataset was split into 70% training and 30% validation subsets. Early stopping was applied to prevent overfitting, halting when the validation error showed no improvements for 50 epochs, and persisting the best parameters found so far. To evaluate the effectiveness of the proposed method, training was conducted over 100 iterations with multiple random seeds. The impact of using point-wise maximization to set targets is illustrated in Figure 1.

As shown in Figure 1 (a)-(d) for the baseline, the Q-value targets, derived from the true Q-values of the previous iteration's policy, show considerable oscil-

[1] https://gymnasium.farama.org

ESANN 2026 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium) and online event, 22-24 April 2026, i6doc.com publ., ISBN 9782875870964. Available from http://www.i6doc.com/en/.



(a) Q-value target convergence (b) No correlation (c) Correlation

Fig. 2: The means of each iteration's Q-value targets are depicted in (a). Correlation of policy performance on fit error is shown in (b), and on the mean of Q-value targets in (c). The policy performance measures the quote of successful long-term balance attempts over 1,000 episodes, each with 5,000 steps.

lations and fail to stabilize across iterations. The inherent discontinuities in the true Q-function make it hard to fit using a function approximator. Consequently, the quality of the resulting policy remains unpredictable, marked by recurring performance collapses and sporadically high Q-values in certain subspaces, with no indication of convergence. In stark contrast, Figures 1 (e)-(h) highlight the effectiveness of employing point-wise maximization over all iterations' targets, mitigating the instability caused by discontinuities. The maximum operator enforces monotonic growth in the Q-value targets. By converging point-wise over iterations, the targets demonstrate enhanced stability and robustness, remaining unaffected by recurring policy degradations. Note, this is only achievable because the method avoids the notorious amplification of overestimation bias that inhibits any bootstrapping-based method from employing such a technique.

4 Suitable performance metrics for policy selection

A key issue in policy iteration methods arises when using function approximators: learning a Q-function on sparse data and subsequently acting greedily with respect to the learned function does not necessarily yield a policy that replicates or improves the Q-values when evaluated [5, 6]. The new policy can either improve, stagnate, or even degrade in performance compared to the policies from earlier iterations. This originates not only from inaccuracies in fitting individual data points but also from the imperfect interpolation or extrapolation performed by the function approximator. Crucially, standard metrics such as train and validation errors are unable to predict the performance of the resulting policy, which is depicted in Figure 2 (b). This disconnect highlights the need for actionable performance metrics that bridge the gap between Q-function target dynamics and policy performance. To address this issue, we propose utilizing the mean of the point-wise Q-values. This is computation-wise efficient since they are already calculated as part of the policy evaluation. Figure 2 (c) depicts the correlation

ESANN 2026 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium) and online event, 22-24 April 2026, i6doc.com publ., ISBN 9782875870964. Available from http://www.i6doc.com/en/.

of the current Q-value target means and the performance of their underlying policy. As shown, they can serve as strong indicators of policy performance. Since they are available throughout training, the policy with the highest average Q-values can be selected post-training, consistently yielding an optimal policy in all experiments conducted.

5 Conclusion

This paper addresses fundamental challenges in offline Q-learning for continuous state-action spaces, where function approximation and bootstrapping often cause instability, error propagation, and overestimation bias. A model-based framework is introduced that replaces bootstrapping with rollout-based Q-value estimation using transition and reward models. It stabilizes learning by computing point-wise maximum Q-value targets across iterations. Its ability to produce monotonic converging Q-values handling discontinuities and being unaffected by degrading policies is demonstrated on a benchmark. Furthermore, a computationally efficient performance metric based on average point-wise Q-values is provided, enabling consistent selection of optimal policies post-training. This dual approach addresses crucial issues in Q-learning, contributing to reliable performance in offline and safety-critical environments.

References

[1] Christopher JCH Watkins and Peter Dayan. Q-learning. *Machine learning*, 8, 1992.

[2] Richard S. Sutton and Andrew G. Barto. *Reinforcement Learning: An Introduction*. MIT press Cambridge, 2018.

[3] Hado van Hasselt, Yotam Doron, Florian Strub, Matteo Hessel, Nicolas Sonnerat, and Joseph Modayil. Deep reinforcement learning and the deadly triad, 2018.

[4] Sebastian Thrun and Anton Schwartz. Issues in using function approximation for reinforcement learning. In *Proceedings of the 1993 connectionist models summer school*.

[5] Philipp Wissmann, Daniel Hein, Steffen Udluft, and Thomas Runkler. Is Q-learning an ill-posed problem? In *ESANN*, 2025.

[6] Amir-massoud Farahmand, Csaba Szepesvári, and Rémi Munos. Error propagation for approximate policy and value iteration. In *NeurIPS*, volume 23, 2010.

[7] Tao Wang, Sylvia Lee Herbert, and Sicun Gao. Fractal landscapes in policy optimization. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023.

[8] Scott Fujimoto, Herke Hoof, and David Meger. Addressing function approximation error in actor-critic methods. In *International conference on machine learning*. PMLR, 2018.

[9] Aviral Kumar, Aurick Zhou, George Tucker, and Sergey Levine. Conservative Q-learning for offline reinforcement learning. In *NeurIPS*, volume 33, 2020.

[10] Ilya Kostrikov, Ashvin Nair, and Sergey Levine. Offline reinforcement learning with implicit Q-learning. In *International Conference on Learning Representations*, 2021.

[11] Scott Fujimoto, David Meger, and Doina Precup. Off-policy deep reinforcement learning without exploration. In *International conference on machine learning*. PMLR, 2019.

[12] Phillip Swazinna, Steffen Udluft, and Thomas Runkler. Overcoming model bias for robust offline deep reinforcement learning. *Engineering Applications of Artificial Intelligence*, 2021.

[13] Philipp Wissmann, Daniel Hein, Steffen Udluft, and Volker Tresp. Why long model-based rollouts are no reason for bad Q-value estimates. In *ESANN*, 2024.