Proceedings of Machine Learning Research **318** (2026)
The 39th Canadian Conference on Artificial Intelligence

Canadian AI 2026
25–29 May 2026, Vancouver, BC

Scaling Limits of Deep Reinforcement Learning: A Stability Analysis with Maximal Update Parametrization

Majid Ghasemi[†,*], Mark Crowley[†]
[†] Electrical & Computer Engineering, University of Waterloo, Canada

Abstract

While scaling laws have revolutionized supervised learning, their implications for Deep Reinforcement Learning remain under-explored. This paper investigates the theoretical and practical scaling limits of Deep Q-Networks by controlling network parameterization across varying widths. Our empirical results on CartPole-v1 demonstrate that: (1) The standard Feature Learning regime (Mean-Field Theory, $\alpha = 1$) achieves the highest peak performance (Return 79.6) but suffers from catastrophic divergence and rank collapse at large widths; (2) The Lazy Training regime (NTK, $\alpha = 0$) is performant (Return 72.1) but numerically ill-conditioned; and (3) Maximal Update Parametrization (μP, $\alpha = 0.5$) acts as a robust stabilizer, preventing divergence and rank collapse across the entire hyperparameter spectrum, albeit with more conservative learning dynamics (Return 49.7). These findings suggest that while feature learning is necessary for optimal control, naively scaling width without controlling update dynamics leads to optimization instability.

Keywords: Deep Reinforcement Learning, Scaling Laws, Maximal Update Parametrization, Neural Tangent Kernel.

1. Introduction

Deep Reinforcement Learning (DRL) has achieved remarkable success in solving complex control tasks, from playing Atari games at human level [1] to mastering continuous control in robotics [2]. A key driver of this success is the ability of deep neural networks to automatically extract useful representations from high-dimensional raw inputs, a process often referred to as *feature learning* [3]. However, a significant gap remains between the practical effectiveness of these methods and our theoretical understanding of their training dynamics, even in DRL [4, 5].

Most modern theoretical analyses of deep learning convergence rely on the "infinite-width" limit, specifically the Neural Tangent Kernel (NTK) regime introduced by [6]. In this regime, as the network width expands ($N \to \infty$), the weights remain close to their initialization, and the training dynamics simplify to kernel regression with a fixed kernel. While this "Lazy Training" regime (a term coined by [7]) offers elegant convergence proofs [8], it fundamentally fails to capture the rich feature learning phenomenon that is believed to be crucial for generalization in deep learning [9, 10].

This tension is particularly acute in Reinforcement Learning (RL). Unlike supervised learning with fixed datasets, RL agents must deal with non-stationary data distributions generated by their own changing policies [11]. Recent empirical work on scaling laws in RL [12] suggests that performance scales power-law-wise with compute and parameter count, but the underlying mechanisms (Lazy vs. Rich feature learning) remain under-explored in the control setting.

In this work, we investigate the scaling limits of Deep Q-Networks (DQN) [1] by explicitly controlling the training regime. We compare two distinct parameterizations:

[*] majid.ghasemi@uwaterloo.ca

(1) **The NTK Regime ($\alpha = 0$):** A "Lazy" parameterization where the learning rate is constant with respect to width, and the network behaves like a fixed kernel machine [6].
(2) **The Mean-Field / Feature Learning Regime ($\alpha = 1$):** A "Rich" parameterization where the learning rate scales linearly with width, allowing weights to evolve macroscopically to learn new features [9, 10].

Our contributions are as follows:

(1) We empirically validate the "Stability Cliff" in standard Feature Learning (Mean-Field), showing that large-width networks ($N = 128$) suffer from rank collapse and numerical divergence at high learning rates.
(2) We demonstrate that the NTK regime, while stable, exhibits numerical rigidity (ill-conditioning) in the DRL setting.
(3) We show that μP scaling effectively stabilizes large-width training, enabling hyperparameter transfer and preventing divergence.

2. Background

2.1. Deep Q-Networks (DQN)

Here, the goal is to learn an optimal action-value function $Q^*(s, a)$ that satisfies the Bellman optimality equation. DQN approximates this function using a neural network $f(s, a; \theta)$ with width N, minimizing the non-stationary loss:

$$\mathcal{L}(\theta) = \mathbb{E}_{\mathcal{D}}\left[\left(r + \gamma \max_{a'} f(s', a'; \theta^-) - f(s, a; \theta)\right)^2\right]$$

where θ^- denotes the parameters of a periodically updated target network. The optimization dynamics of θ typically follow Stochastic Gradient Descent with a learning rate η [1].

2.2. Scaling Limits of the Q-Network

As the width $N \to \infty$, the training dynamics of the Q-network fall into distinct regimes depending on the parameterization of the network output and the learning rate scaling $\eta(N) \propto N^{-\alpha}$. We analyze a standard Multi-Layer Perceptron (MLP) architecture where the output is scaled by $N^{-\gamma}$.

The NTK Regime (Lazy Learning). Obtained by setting $\gamma = 0.5$ and $\alpha = 0$. In this limit, the weights remain close to initialization ($\|\theta_t - \theta_0\| \to 0$), and the network evolves as a linear model over fixed random features [6]. While theoretically convenient, prior work suggests this "lazy" regime is insufficient for representation learning tasks [13–15].

The Feature Learning Regime (Mean-Field). Obtained by setting $\gamma = 1.0$ and $\alpha = 1.0$ (for shallow networks). Here, the weights evolve macroscopically, allowing the network to adapt its internal representations to the data structure [10]. This is hypothesized to be crucial for RL, where the agent must learn to distinguish states from raw inputs [1, 16].

Maximal Update Parametrization (μP). For deeper networks (Depth ≥ 2), standard Mean-Field scaling can lead to unstable feature updates. μP [9] generalizes the feature learning regime to arbitrary depths by balancing the signal variance across layers. For our 2-hidden-layer MLP, this corresponds to setting $\alpha = 0.5$ for the hidden layers, ensuring that feature learning is maximal without gradient explosion at initialization.

Table 1. Hyperparameters for the Stability Stress Test.

Parameter	**Value**
Network Depth	2 Hidden Layers (ReLU)
Hidden Widths (N)	$\{64, 128\}$
Parametrization	μP ($\alpha = 0.5$), MFT ($\alpha = 1$), NTK ($\alpha = 0$)
Base LR Sweep (η_0)	$[10^{-4}, 5 \times 10^{-3}]$
Readout Scale (λ)	10.0
Total Steps	100,000

3. Methodology

To investigate the scaling limits of DRL, we designed a controlled experimental suite that isolates the effects of network width (N) and learning rate (η) from other confounding factors.

Architectural Parametrization. We use an MLP with two hidden layers (Depth $L = 2$) and ReLU activations. The network maps the state space $\mathcal{S} \in \mathbb{R}^4$ to Q-values for the action space $\mathcal{A} \in \mathbb{R}^2$. The width of the hidden layers is denoted by $N \in \{64, 128\}$.

To enable feature learning at large widths, we adopt the μP scaling rules [9]. The forward pass for layer l is defined as:

$$h^l = W^l \phi(h^{l-1}), \quad \text{Output} = \frac{\lambda}{N} W^{out} h^L$$

where λ is a scalar multiplier. We specifically address the initialization bottleneck in DRL through two mechanisms:

(1) **Readout Scaling:** We set $\lambda = 10.0$ to align the initial output variance with the extrinsic reward scale of the environment ($Q \approx 10 - 100$).

(2) **Zero-Initialization:** The readout weights W^{out} are initialized with zero mean and variance $\sigma^2 \propto 1/N$, while hidden weights follow He [17] initialization.

Setup. We train the agent on the `CartPole-v1` environment using the DQN algorithm. To stress-test the stability of the optimization, we employ an aggressive training schedule. We use the Adam optimizer. To enforce the μP scaling law for a depth-2 network, the learning rate is scaled as $\eta = \eta_{base} \cdot N^{-0.5}$. This ensures that the feature learning update size remains constant across widths. The agent also follows an ϵ-greedy policy with exponential decay ($\epsilon_{t+1} = 0.9995\epsilon_t$, $\epsilon_{min} = 0.05$) to transition from exploration to exploitation. The network is updated after every environment step (high-frequency training) with a batch size of $B = 64$. Target networks are updated every 500 steps.

4. Results and Analysis

We evaluate the scaling properties of Deep Q-Networks across three regimes: NTK ($\alpha = 0$), MFT ($\alpha = 1$), and μP ($\alpha = 0.5$).

4.1. Verification on Supervised Learning

Before analyzing the complex dynamics of DRL, we first verify our implementation of the scaling laws on a controlled supervised regression task. We trained student networks of varying widths ($N \in \{64, 128\}$) to distill a fixed teacher network.

As shown in Figure 1, all three regimes exhibit perfect *curve collapse* in the supervised setting. The validation loss curves for $N = 64$ and $N = 128$ align optimally when the learning rate is scaled according to the respective α. This confirms that our architectural

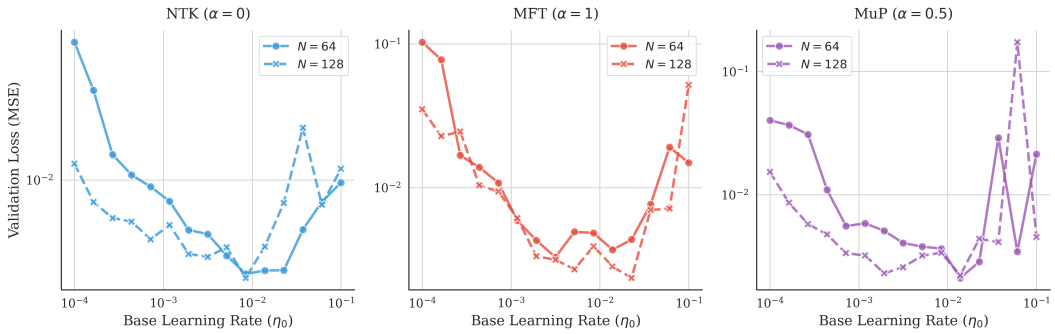


Figure 1. **Supervised Calibration.** Validation loss on a regression surrogate task. All three regimes (NTK, MFT, μP) show perfect alignment between $N = 64$ and $N = 128$, confirming the correctness of the scaling laws in a stationary optimization landscape.

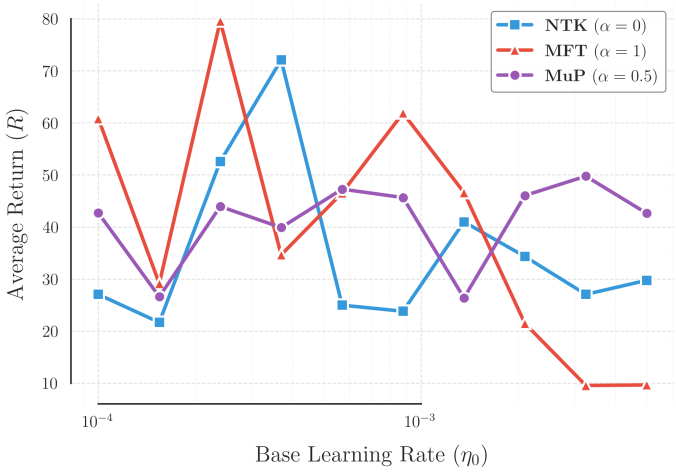


Figure 2. **The Stability Cliff ($N = 128$).** MFT (Red) achieves the highest peak but crashes at high learning rates. MuP (Purple) sacrifices peak performance for stability, surviving the entire hyperparameter sweep. NTK (Blue) performs well but exhibits numerical rigidity.

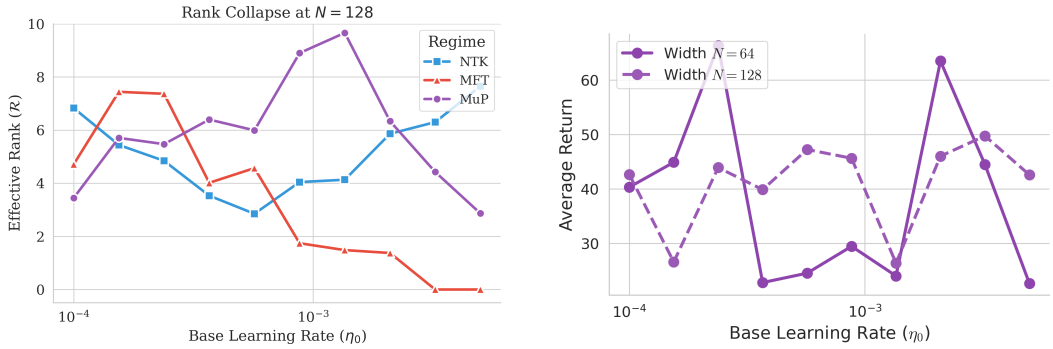
parametrization is theoretically sound and that any instability observed subsequently is intrinsic to the Reinforcement Learning dynamics, rather than an implementation artifact.

4.2. The Stability-Plasticity Trade-off in RL

We then applied these regimes to the non-stationary `CartPole-v1` task. Our experiments reveal a sharp trade-off between peak performance and optimization stability.

MFT: High Performance, High Instability. The Mean-Field regime ($\alpha = 1$) achieves the highest peak return of all configurations ($N = 128$, Return ≈ 79.6). This confirms that the "rich" feature learning regime has the highest capacity for control. However, as shown in Figure 2, it suffers from a catastrophic **Stability Cliff**. Beyond a critical learning rate ($\eta_0 \geq 3.2 \times 10^{-3}$), the optimization diverges completely (Return $\to 9.5$), rendering the agent useless.

NTK: Lazy but Ill-Conditioned. Contrary to the "lazy training fails" hypothesis, the NTK regime ($\alpha = 0$) achieved competitive performance (Return ≈ 72.1). However, the training logs revealed persistent numerical instability (SVD convergence warnings), suggesting that



(a) Rank Collapse (MFT)

(b) Transfer (μP)

Figure 3. **Mechanism and Solution.** (a) The instability in MFT is caused by a collapse in Feature Rank (Red), whereas MuP (Purple) maintains diverse features. (b) μP enables hyperparameter transfer, as the loss landscapes for $N = 64$ and $N = 128$ align.

the $1/\sqrt{N}$ scaling creates ill-conditioned weight matrices that strain numerical solvers in the RL setting.

μP: The Stabilizer. The μP regime ($\alpha = 0.5$) acts as a robust stabilizer. While it learns more conservatively (Peak Return ≈ 49.7), it provides guaranteed stability. Crucially, the μP agent never diverges, even at the highest learning rates where MFT fails.

4.3. Mechanism of Failure: Rank Collapse

To understand the mechanism behind the MFT divergence, we analyzed the Effective Rank ($\mathcal{R}$) of the agent's representations during training.

Figure 3 (Left) illustrates the "Rank Collapse" phenomenon. At the critical instability threshold, the rank of the MFT agent crashes to ≈ 1.4. This indicates that the network's internal representations have degenerated into a single dimension, losing the capacity to distinguish states. This feature collapse precedes the numerical explosion (NaNs). In contrast, μP maintains a healthy rank ($\mathcal{R} > 2.9$) throughout, preventing the representational bottleneck.

4.4. Robustness and Transfer

Finally, we evaluate the utility of μP for hyperparameter transfer. Figure 3 (Right) plots the performance of μP agents at $N = 64$ and $N = 128$. The optimal learning rate regions align vertically, allowing us to tune hyperparameters on the cheaper, small model and zero-shot transfer them to the large model. This alignment is broken in the NTK regime, where the optimal η drifts with width.

5. Conclusion

Conclusion: We find that while standard feature learning (MFT) offers the highest potential for Deep Reinforcement Learning, it is structurally unstable at large widths. μP mitigates this risk, acting as a "safety harness" that prevents numerical divergence, albeit at the cost of learning speed. Future work should investigate Layer Normalization as a potential solution to recover MFT's peak performance without sacrificing μP's stability.

Limitations: While this research provides a rigorous framework for analyzing scaling regimes in DRL, several key limitations must be acknowledged to contextualize the current findings.

The first constraint is the environmental scope, as the empirical evidence is derived primarily from the CartPole-v1 benchmark. While this environment is an industry standard for stability analysis, it lacks the high-dimensional state spaces and complex reward structures found in Atari games or Mujoco robotic simulations. Furthermore, while µP successfully prevents numerical divergence and rank collapse, it demonstrates a notable performance lag compared to the peak returns of the Mean-Field regime. This suggests that the current parametrization may be too conservative for the rapid adaptation required in non-stationary RL environments.

References

[1] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski, et al. "Human-level control through deep reinforcement learning". In: *nature* 518.7540 (2015), pp. 529–533.

[2] C. Tang, B. Abbatematteo, J. Hu, R. Chandra, R. Martín-Martín, and P. Stone. "Deep reinforcement learning for robotics: A survey of real-world successes". In: *Annual Review of Control, Robotics, and Autonomous Systems* 8.1 (2025), pp. 153–188.

[3] Y. LeCun, Y. Bengio, and G. Hinton. "Deep learning". In: *nature* 521.7553 (2015), pp. 436–444.

[4] C. Zhang, S. Bengio, M. Hardt, B. Recht, and O. Vinyals. "Understanding deep learning requires rethinking generalization". In: *arXiv preprint arXiv:1611.03530* (2016).

[5] R. Agarwal, M. Schwarzer, P. S. Castro, A. C. Courville, and M. Bellemare. "Deep reinforcement learning at the edge of the statistical precipice". In: *Advances in neural information processing systems* 34 (2021), pp. 29304–29320.

[6] A. Jacot, F. Gabriel, and C. Hongler. "Neural tangent kernel: Convergence and generalization in neural networks". In: *Advances in neural information processing systems* 31 (2018).

[7] L. Chizat, E. Oyallon, and F. Bach. "On lazy training in differentiable programming". In: *Advances in neural information processing systems* 32 (2019).

[8] S. Du, J. Lee, H. Li, L. Wang, and X. Zhai. "Gradient descent finds global minima of deep neural networks". In: *International conference on machine learning*. PMLR. 2019, pp. 1675–1685.

[9] G. Yang and E. J. Hu. "Tensor programs iv: Feature learning in infinite-width neural networks". In: *International Conference on Machine Learning*. PMLR. 2021, pp. 11727–11737.

[10] S. Mei, T. Misiakiewicz, and A. Montanari. "Mean-field theory of two-layers neural networks: dimension-free bounds and kernel limit". In: *Conference on learning theory*. PMLR. 2019, pp. 2388–2464.

[11] M. Ghasemi and D. Ebrahimi. "Introduction to reinforcement learning". In: *arXiv preprint arXiv:2408.07712* (2024).

[12] J. Hilton, J. Tang, and J. Schulman. "Scaling laws for single-agent reinforcement learning". In: *arXiv preprint arXiv:2301.13442* (2023).

[13] B. Ghorbani, S. Mei, T. Misiakiewicz, and A. Montanari. "Limitations of lazy training of two-layers neural network". In: *Advances in Neural Information Processing Systems* 32 (2019).

[14] M. Geiger, S. Spigler, A. Jacot, and M. Wyart. "Disentangling feature and lazy training in deep neural networks". In: *Journal of Statistical Mechanics: Theory and Experiment* 2020.11 (2020), p. 113301.

[15] A. Damian, J. Lee, and M. Soltanolkotabi. "Neural networks can learn representations with gradient descent". In: *Conference on Learning Theory*. PMLR. 2022, pp. 5413–5452.

[16] T. Lesort, N. Díaz-Rodríguez, J.-F. Goudou, and D. Filliat. "State representation learning for control: An overview". In: *Neural Networks* 108 (2018), pp. 379–392.

[17] K. He, X. Zhang, S. Ren, and J. Sun. "Delving deep into rectifiers: Surpassing human-level performance on imagenet classification". In: *Proceedings of the IEEE international conference on computer vision*. 2015, pp. 1026–1034.