A Closer Look at Sparse Training in Deep Reinforcement Learning

Muhammad Athar Ganaie
University of Calgary
muhammadathar.ganaie@ucalgary.ca

Vincent Michalski
Université de Montréal
michalskivince@gmail.com

Samira Ebrahimi Kahou
University of Calgary
samira.ebrahimi.kahou@gmail.com

Yani Ioannou
University of Calgary
yani.ioannou@ucalgary.ca

Abstract

Deep neural networks have enabled remarkable progress in reinforcement learning across a variety of domains, yet advancements in model architecture, especially involving sparse training, remain under-explored. Sparse architectures hold potential for reducing computational overhead in deep reinforcement learning (DRL), where prior studies suggest that parameter under-utilization may create opportunities for efficiency gains. This work investigates adaptation of sparse training methods from supervised learning to DRL, specifically examining pruning and the RigL algorithm in value-based agents like DQN. In our experiments across multiple Atari games, we study factors neglected in supervised sparse training which are of relevance to DRL, such as the impact of the bias parameter in high-sparsity regimes and the dynamics of dormant neurons under sparse conditions. The results reveal that RigL, despite its adaptability in supervised contexts, under-performs relative to pruning in DRL. Strikingly, removing bias parameters enhances RigL's performance, reduces dormant neurons and improves stability in high sparsity, while pruning suffers the opposite effect. These empirical observations underscore the imperative to re-evaluate sparse training methodologies, particularly within the context of DRL paradigms. The results elucidate the necessity for further investigation into the applicability of sparse training techniques across more expansive architectural frameworks and diverse environments.

1 Introduction

Deep neural networks are now fundamental to scalable reinforcement learning (RL), driving achievements across domains such as Atari gameplay [Mnih et al., 2015a], strategic games like Go [Silver et al., 2016], complex simulations in Dota 2 [Berner et al., 2019], and real-time control in applications like stratospheric balloons and plasma systems [Bellemare et al., 2020, Degrave et al., 2022]. Despite these successes, the field has primarily concentrated on algorithmic advancements in deep reinforcement learning (DRL), leaving architectural refinements, including sparse training under-explored. However, recent findings indicate that DRL agents often operate with underutilized parameters during training [Kumar et al., 2021, Sokar et al., 2022], suggesting that sparsity could offer significant benefits by reducing training costs and enhancing performance in latency-sensitive settings. Obando-Ceron et al. [2024] demonstrate that a pruned model obtained from gradual magnitude pruning is better for most online DRL settings.

Recent work has taken initial steps towards incorporating sparse architectures within DRL by applying sparse training algorithms originally developed for supervised learning [Graesser et al.,

38th Conference on Neural Information Processing Systems (NeurIPS 2024).

2022, Obando-Ceron et al., 2024]. However, in some aspects these approaches implicitly assume supervised training dynamics can directly transfer to DRL. Unlike supervised learning, DRL training involves a stochastic, reward-driven feedback loop that can be delayed and unstable, affecting model expressivity and training stability. Notably, the observation of Graesser et al. [2022] that the RigL [Evci et al., 2021] algorithm underperforms compared to pruning [Zhu and Gupta, 2017], are surprising and counter-intuitive given that dynamic sparse training (DST) methods like RigL are dynamic in nature and should intuitively benefit the distribution shift characteristic of DRL training [Sokar et al., 2022].

In this paper, we posit that directly transferring supervised sparse training methods to DRL is suboptimal, highlighting the need to re-evaluate strategies that accommodate the dynamic and stochastic characteristics of DRL. We systematically investigate Sparse Training methods for DRL, focusing on value-based agents (DQN [Mnih et al., 2013]) in the online setting and examining two primary approaches: pruning [Zhu and Gupta, 2017] and RigL [Evci et al., 2021]. Our experiments reveal several DRL-specific observations, such as the impact of model bias-parameter in high sparsity regimes, and the variation of dormant neurons in different sparse training algorithms. We observe that simply architectural decisions that are implicitly made by common architectures (e.g. ResNets) used in supervised learning algorithms, like disabling the bias terms for some layers, can actually enable DST (RigL) algorithms to outperform pruning in DRL. This underscores the importance of approaching sparse training through the lens of DRL, rather than carrying over assumptions from supervised learning paradigms.

2 Related Work

Deep neural networks have been instrumental in enabling RL to achieve remarkable results, however, scaling these networks in RL has proven challenging due to inherent training instabilities [van Hasselt et al., 2015, Zha et al., 2019]. Researchers have proposed a variety of methods to mitigate these instabilities and enable larger architectures, including transitioning from traditional CNNs to ResNet-based [He et al., 2015] architectures [Mnih et al., 2015b, Espeholt et al., 2018, Cobbe et al., 2020, Schwarzer et al., 2021] and applying normalization techniques [Bjorck et al., 2022].

Sparse neural networks have recently emerged as a promising approach to reducing training costs and enhancing scalability in DRL. Traditional dense-to-sparse approaches, such as gradual pruning, have demonstrated success in deep learning by systematically removing parameters over time [Han et al., 2015, Zhu and Gupta, 2017]. DST methods like SET [Mocanu et al., 2018] and RigL [Evci et al., 2021] improve upon static sparse methods by allowing the network's sparse topology to evolve, the latter leveraging gradient signals to optimize connectivity. It has been observed that DRL networks often contain dormant neurons, leading to under-utilization of network capacity [Sokar et al., 2023]. Dynamic sparse training approaches that periodically reinitialize and update network connections [Sokar et al., 2022, Graesser et al., 2022, Obando-Ceron et al., 2024] have been shown to maintain network plasticity, sometimes outperforming their dense counterparts.

In this paper, we build upon these insights by examining sparse training techniques, focusing on factors which influence their performance in DRL. We employ pruning [Zhu and Gupta, 2017, Obando-Ceron et al., 2024] and RigL [Evci et al., 2021] as primary methods for dynamic sparse training in value-based DRL agents.

3 Empirical Evaluation

3.1 Experimental Setup

Our work centers on value-based algorithms, focusing specifically on DQN [Mnih et al., 2015b] to explore sparse training approaches within DRL. The ALE Atari suite [Bellemare et al., 2013] provides the test environments, with five games (Pong, Qbert, Breakout, MSPacman, SpaceInvaders) selected to capture varying levels of difficulty. Each experiment was conducted over 40 million frames and repeated five times to account for variance, reporting the average reward over the final 10% of evaluations, supplemented by 95% confidence intervals in all plots. This comparison includes dense-to-sparse (Pruning [Zhu and Gupta, 2017]) and sparse training (RigL [Evci et al., 2021]) approaches. Prior work [Obando-Ceron et al., 2024] suggests that pruning represents the State-Of-The-Art (SOTA) in sparse training for DRL settings, whereas RigL has shown promise in supervised learning contexts. Unlike the sparse training schedule outlined by Graesser et al. [2022], we commenced sparse training after 20% of training progress both for RigL and pruning to establish a more equitable basis for comparison.

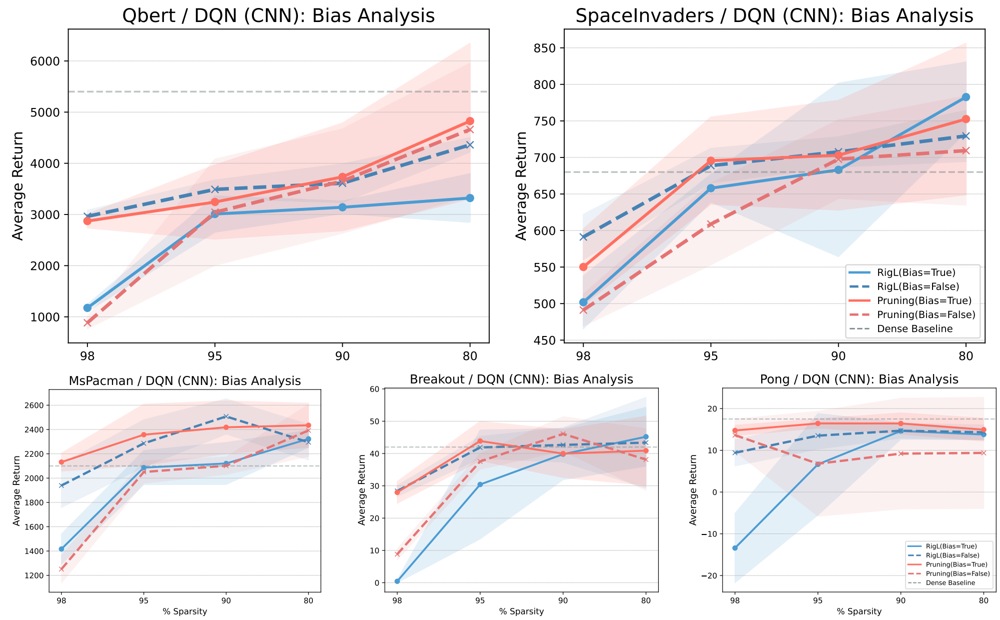

Figure 1: Comparison of final reward as a function of sparsity (i.e., the percentage of network weights set to zero) for RigL and pruning techniques across various games. Dashed lines indicate experiments where bias parameters were excluded. Sparse training methods were evaluated at sparsity levels from 80% to 98%. The dense baseline rewards are represented by horizontal lines. Shaded regions denote 95% confidence intervals, capturing variability across multiple runs.

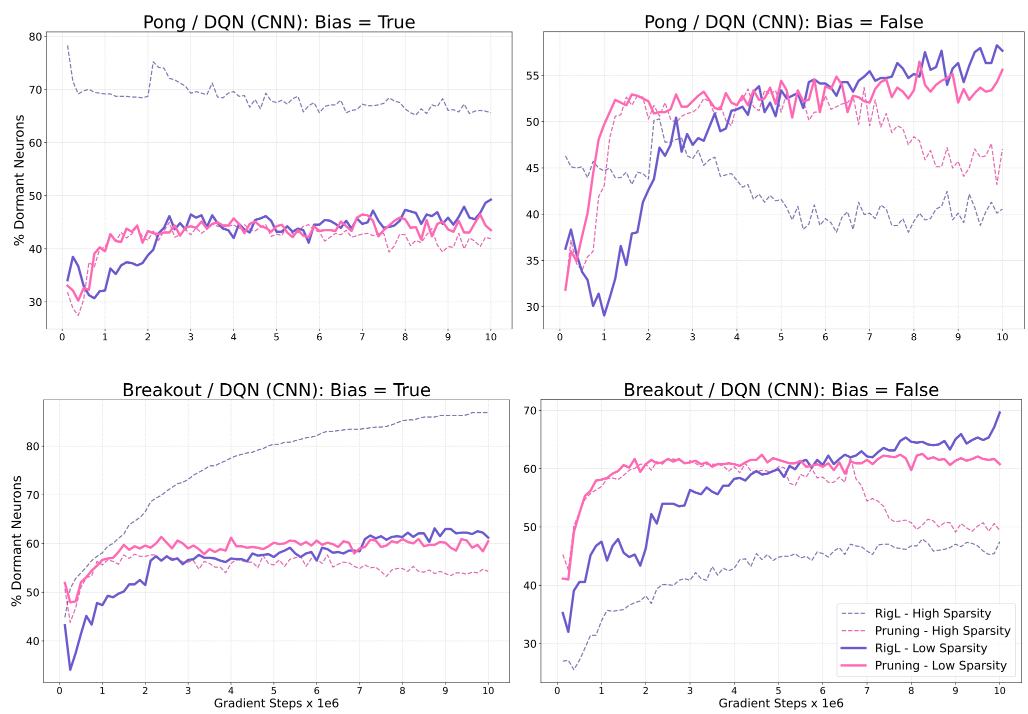

Figure 2: Dormant neurons evolution in Pong and Breakout, bias enabled in the left column and disabled in the right column. Dashed lines represent high sparsity(98%), and solid lines indicate low sparsity(80%).

The training is concluded at 80%, after which the sparse network remained fixed for the duration of training. We employed the JAX [Bradbury et al., 2018] implementations of the Dopamine library [Castro et al., 2018] for DQN agent. Sparse training was facilitated via the JaxPruner library [Lee et al., 2023].

3.2 Early Training

A critical observation pertains to the starting point of sparse training. Pruning in [Graesser et al., 2022] typically initiate sparse training at 20% of the full training timeline, whereas RigL commences from the first step. RigL training was adjusted to begin at the same 20% mark. This adjustment (see A.1) closed the performance gap in low-sparsity scenarios between RigL and pruning, underscoring the influence of early training on sparse training outcomes.

3.3 Effect of Bias on DST

In deep learning, each neuron's output is computed as a weighted sum of its inputs, followed by the application of an activation function. Mathematically, for a neuron i in layer l, the output (activation) $a_i^{(l)}$ can be expressed as:

$$a_i^{(l)} = \phi\left(\sum_{j=1}^{n} w_{ij}^{(l)} a_j^{(l-1)} + b_i^{(l)}\right),$$

where ϕ is the activation function, $w_{ij}^{(l)}$ is the weight for input activation $a_j^{(l-1)}$, and $b_i^{(l)}$ is the bias term for the neuron. Our work explores the impact of the bias term $b_i^{(l)}$ on sparse training methods RigL and pruning when applied to DRL. To understand this impact, we conducted experiments with the bias term disabled.

Adjusting the training start point for RigL yielded improved performance at lower sparsity levels; however, RigL continued to under-perform compared to pruning in high-sparsity regimes (i.e., above 90%). Previous studies [Konda et al., 2015] suggest that the presence of biases may impede the formation of effective sparse representations, particularly in context of auto-encoders. We examined the impact of disabling bias in both pruning and RigL methods. Notably, RigL exhibited better performance across all five games at high sparsity when bias was removed, as depicted in Figure 1, where dashed lines indicate the experiments conducted without bias. When bias was disabled, RigL performed comparably to pruning with bias enabled. In contrast, pruning showed a decline in performance without bias, suggesting the impact of bias being more pronounced in improving DST methods.

3.4 Dormant Neuron Analysis

The dormant neuron phenomenon, previously observed in both supervised and DRL settings [Sokar et al., 2023], results in diminished network expressivity and plasticity. Obando-Ceron et al. [2024] observed a reduction in dormant neurons with pruning compared to dense training; however, they did not include comparisons with other sparse training methods.

The methodology for defining and quantifying dormant neurons in our study follows the approach established by [Sokar et al., 2023]. Figure 2 illustrates the progression of dormant neurons throughout the training process. Our analysis focuses on two Atari games, Pong and Breakout, under varying sparsity conditions. The emergence of dormant neurons, particularly at high sparsity levels for RigL, is well depicted in the (left column) Figure 2. This observation provides insight into the performance decline between these approaches, specifically elucidating RigL's diminished performance at higher sparsities relative to pruning. As illustrated in Figure 2, the deactivation of bias (right column) in RigL resulted in a notable reduction of dormant neurons. This finding aligns with the performance enhancements we observed under these conditions. Conversely, pruning methods exhibited an increase in dormant neurons upon bias deactivation.

4 Conclusion

Our work examines the adaptation of sparse training techniques from supervised learning, specifically pruning and RigL, to DRL. While RigL's dynamic adaptability falters at high sparsity levels, pruning consistently demonstrates strong performance across various games. A key finding is that disabling the bias terms in the common neural network architectures used in the context of previous work significantly enhances RigL's performance at high sparsity, aligning it more closely with pruning.

Additionally, we show that high sparsity can lead to dormant neurons, but bias removal mitigates this effect for RigL. These insights prompt a re-evaluation of sparse training algorithms tailored specifically for DRL. Future work will explore larger architectures like Impala [Espeholt et al., 2018] and diverse environments to further assess the potential of sparse training in DRL.

Acknowledgments

We acknowledge the support of Alberta Innovates, the Natural Sciences and Engineering Research Council of Canada (NSERC), and Defence Research and Development Canada (DRDC). We are grateful for the computational resources made available to us by Denvr Dataworks, Amazon, and the Digital Research Alliance of Canada.

References

M. G. Bellemare, Y. Naddaf, J. Veness, and M. Bowling. The arcade learning environment: An evaluation platform for general agents. *Journal of Artificial Intelligence Research*, 47:253–279, jun 2013.

Marc G Bellemare, Salvatore Candido, Pablo Samuel Castro, Jun Gong, Marlos C Machado, Subhodeep Moitra, Sameera S Ponda, and Ziyu Wang. Autonomous navigation of stratospheric balloons using reinforcement learning. *Nature*, 588(7836):77–82, 2020.

Christopher Berner, Greg Brockman, Brooke Chan, Vicki Cheung, Przemyslaw Debiak, Christy Dennison, David Farhi, Quirin Fischer, Shariq Hashme, Christopher Hesse, Rafal Józefowicz, Scott Gray, Catherine Olsson, Jakub Pachocki, Michael Petrov, Henrique Pondé de Oliveira Pinto, Jonathan Raiman, Tim Salimans, Jeremy Schlatter, Jonas Schneider, Szymon Sidor, Ilya Sutskever, Jie Tang, Filip Wolski, and Susan Zhang. Dota 2 with large scale deep reinforcement learning. *CoRR*, abs/1912.06680, 2019. URL http://arxiv.org/abs/1912.06680.

Johan Bjorck, Carla P. Gomes, and Kilian Q. Weinberger. Towards deeper deep reinforcement learning with spectral normalization, 2022. URL https://arxiv.org/abs/2106.01151.

James Bradbury, Roy Frostig, Peter Hawkins, Matthew James Johnson, Chris Leary, Dougal Maclaurin, George Necula, Adam Paszke, Jake VanderPlas, Skye Wanderman-Milne, and Qiao Zhang. JAX: composable transformations of Python+NumPy programs, 2018. URL http://github.com/jax-ml/jax.

Pablo Samuel Castro, Subhodeep Moitra, Carles Gelada, Saurabh Kumar, and Marc G. Bellemare. Dopamine: A Research Framework for Deep Reinforcement Learning. 2018. URL http://arxiv.org/abs/1812.06110.

Karl Cobbe, Christopher Hesse, Jacob Hilton, and John Schulman. Leveraging procedural generation to benchmark reinforcement learning, 2020. URL https://arxiv.org/abs/1912.01588.

Jonas Degrave, Federico Felici, Jonas Buchli, Michael Neunert, Brendan Tracey, Francesco Carpanese, Timo Ewalds, Roland Hafner, Abbas Abdolmaleki, Diego de Las Casas, et al. Magnetic control of tokamak plasmas through deep reinforcement learning. *Nature*, 602(7897):414–419, 2022.

Lasse Espeholt, Hubert Soyer, Remi Munos, Karen Simonyan, Volodymir Mnih, Tom Ward, Yotam Doron, Vlad Firoiu, Tim Harley, Iain Dunning, Shane Legg, and Koray Kavukcuoglu. Impala: Scalable distributed deep-rl with importance weighted actor-learner architectures, 2018. URL https://arxiv.org/abs/1802.01561.

Utku Evci, Trevor Gale, Jacob Menick, Pablo Samuel Castro, and Erich Elsen. Rigging the lottery: Making all tickets winners, 2021. URL https://arxiv.org/abs/1911.11134.

Laura Graesser, Utku Evci, Erich Elsen, and Pablo Samuel Castro. The state of sparse training in deep reinforcement learning, 2022. URL https://arxiv.org/abs/2206.10369.

Song Han, Jeff Pool, John Tran, and William J. Dally. Learning both weights and connections for efficient neural networks, 2015. URL https://arxiv.org/abs/1506.02626.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition, 2015. URL https://arxiv.org/abs/1512.03385.

Kishore Konda, Roland Memisevic, and David Krueger. Zero-bias autoencoders and the benefits of co-adapting features, 2015. URL https://arxiv.org/abs/1402.3337.

Aviral Kumar, Rishabh Agarwal, Dibya Ghosh, and Sergey Levine. Implicit under-parameterization inhibits data-efficient deep reinforcement learning, 2021. URL https://arxiv.org/abs/2010.14498.

Joo Hyung Lee, Wonpyo Park, Nicole Mitchell, Jonathan Pilault, Johan S. Obando-Ceron, Han-Byul Kim, Namhoon Lee, Elias Frantar, Yun Long, Amir Yazdanbakhsh, Shivani Agrawal, Suvinay Subramanian, Xin Wang, Sheng-Chun Kao, Xingyao Zhang, Trevor Gale, Aart J. C. Bik, Woohyun Han, Milen Ferev, Zhonglin Han, Hong-Seok Kim, Yann Dauphin, Karolina Dziugaite, Pablo Samuel Castro, and Utku Evci. Jaxpruner: A concise library for sparsity research. 2023.

Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Alex Graves, Ioannis Antonoglou, Daan Wierstra, and Martin Riedmiller. Playing atari with deep reinforcement learning, 2013. URL https://arxiv.org/abs/1312.5602.

Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A. Rusu, Joel Veness, Marc G. Bellemare, Alex Graves, Martin A. Riedmiller, Andreas Fidjeland, Georg Ostrovski, Stig Petersen, Charles Beattie, Amir Sadik, Ioannis Antonoglou, Helen King, Dharshan Kumaran, Daan Wierstra, Shane Legg, and Demis Hassabis. Human-level control through deep reinforcement learning. *Nat.*, 518(7540):529–533, 2015a. doi: 10.1038/NATURE14236. URL https://doi.org/10.1038/nature14236.

Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A. Rusu, Joel Veness, Marc G. Bellemare, Alex Graves, Martin A. Riedmiller, Andreas Kirkeby Fidjeland, Georg Ostrovski, Stig Petersen, Charlie Beattie, Amir Sadik, Ioannis Antonoglou, Helen King, Dharshan Kumaran, Daan Wierstra, Shane Legg, and Demis Hassabis. Human-level control through deep reinforcement learning. *Nature*, 518:529–533, 2015b. URL https://api.semanticscholar.org/CorpusID:205242740.

Decebal Constantin Mocanu, Elena Mocanu, Peter Stone, Phuong H. Nguyen, Madeleine Gibescu, and Antonio Liotta. Scalable training of artificial neural networks with adaptive sparse connectivity inspired by network science. *Nature Communications*, 9(1), June 2018. ISSN 2041-1723. doi: 10.1038/s41467-018-04316-3. URL http://dx.doi.org/10.1038/s41467-018-04316-3.

Johan Obando-Ceron, Aaron Courville, and Pablo Samuel Castro. In value-based deep reinforcement learning, a pruned network is a good network, 2024. URL https://arxiv.org/abs/2402.12479.

Max Schwarzer, Ankesh Anand, Rishab Goel, R Devon Hjelm, Aaron Courville, and Philip Bachman. Data-efficient reinforcement learning with self-predictive representations, 2021. URL https://arxiv.org/abs/2007.05929.

David Silver, Aja Huang, Chris J. Maddison, Arthur Guez, Laurent Sifre, George van den Driessche, Julian Schrittwieser, Ioannis Antonoglou, Vedavyas Panneershelvam, Marc Lanctot, Sander Dieleman, Dominik Grewe, John Nham, Nal Kalchbrenner, Ilya Sutskever, Timothy P. Lillicrap, Madeleine Leach, Koray Kavukcuoglu, Thore Graepel, and Demis Hassabis. Mastering the game of go with deep neural networks and tree search. *Nat.*, 529(7587):484–489, 2016. doi: 10.1038/NATURE16961. URL https://doi.org/10.1038/nature16961.

Ghada Sokar, Elena Mocanu, Decebal Constantin Mocanu, Mykola Pechenizkiy, and Peter Stone. Dynamic sparse training for deep reinforcement learning, 2022. URL https://arxiv.org/abs/2106.04217.

Ghada Sokar, Rishabh Agarwal, Pablo Samuel Castro, and Utku Evci. The dormant neuron phenomenon in deep reinforcement learning, 2023. URL https://arxiv.org/abs/2302.12902.

Hado van Hasselt, Arthur Guez, and David Silver. Deep reinforcement learning with double q-learning, 2015. URL https://arxiv.org/abs/1509.06461.

Daochen Zha, Kwei-Herng Lai, Kaixiong Zhou, and Xia Hu. Experience replay optimization, 2019. URL https://arxiv.org/abs/1906.08387.

Michael Zhu and Suyog Gupta. To prune, or not to prune: exploring the efficacy of pruning for model compression, 2017. URL https://arxiv.org/abs/1710.01878.

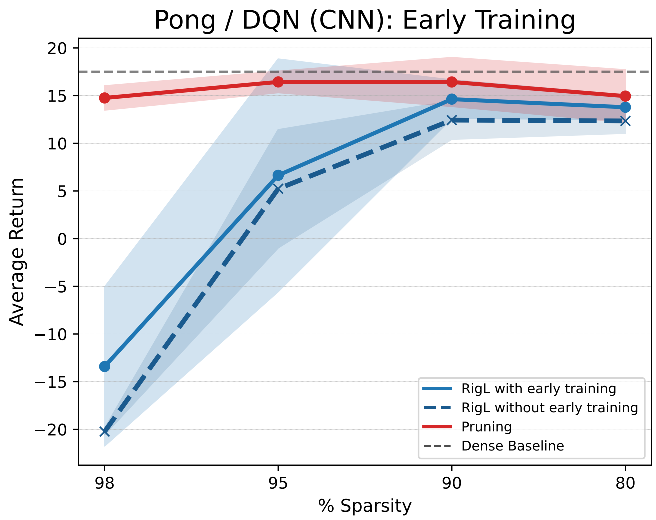

Figure 3: Impact of early training on RigL. RigL starting at 20% matches pruning performance in low-sparsity scenarios.

A Additional Results

A.1 Early Training