Eligibility Traces in Pacman: Reinforcement Learning using Semi-Gradient TD(λ) and True Online TD(λ)

First A. Ashwin Rao, ASU, Second B. Gautham Damodaran Jagath Kumar, ASU, Third C. Julian Lin, ASU and Fourth D. Nikunj Phutela, ASU

***Abstract*—This study explores the application of advanced temporal difference (TD) learning methods—Semi-Gradient TD(λ) and True Online TD(λ)—within the reinforcement learning (RL) [1] framework in the dynamic Pacman environment [2]. Utilizing linear function approximation, these methods are compared against a baseline Approximate Q-learning model. Our experiments in various dynamically generated Pacman environments reveal that both TD(λ) methods significantly perform better than the baseline. This comparative analysis not only underscores the potential of these advanced TD(λ) methods in addressing real-world complexities but also paves the way for their application in broader RL tasks characterized by uncertainty and dynamic changes.**

I. INTRODUCTION

The motivation for this project stems from the need to enhance the learning efficiency and decision-making accuracy in reinforcement learning (RL) applications [1]. Recent advancements in RL methodologies have shown promising results, particularly in enhancing learning efficiency and decision-making accuracy. This project, by focusing on the Pacman game domain [2], aims to contribute uniquely to this body of knowledge by employing two sophisticated temporal difference (TD) learning methods—Semi-Gradient TD(λ) and True Online TD(λ).

Pacman serves not only as a testbed for algorithmic innovation but also as a bridge to real-world applications. It encompasses key challenges such as decision-making in the face of adversaries, planning under uncertainty, and learning from a combination of immediate and delayed rewards - elements that are ubiquitous in practical RL deployments. This study, therefore, not only explores the technical execution of Semi-Gradient TD(λ) and True Online TD(λ) within this framework but also sets the stage for broader application of these methods in RL tasks characterized by similar complexities.

Previous work laid the groundwork by integrating linear function approximation into RL tasks [8, 9], demonstrating its potential to streamline state representation and facilitate learning. Building upon this foundation, the current project explores the application of Semi-Gradient TD(λ) and True Online TD(λ) algorithms in an active learning context [4,7]. Adapted to embody the principles of the Sarsa algorithm for direct control, these methods are meticulously integrated into the Pacman game, a domain characterized by its complexity and the need for strategic planning and execution.

This report delves into the technical intricacies of the algorithms' implementation, their experimental evaluation against the Approximate Q-learning baseline, and the resultant enhancements in learning efficiency and policy optimization. By casting light on the comparative advantages of Semi-Gradient and True Online TD(λ) approaches, this investigation contributes valuable insights to the RL community, advocating for their broader utilization in tasks that mirror the dynamic and unpredictable nature of real-world challenges.

II. TECHNICAL APPROACH

Implementing Semi-Gradient TD(λ) and True Online TD(λ) in the Pacman domain required adapting passive RL algorithms for active control [3,7]. Our agent's design was centered around enabling real-time decision-making and policy refinement, essential for the dynamic and stochastic nature of the Pacman environment.

Traditional passive reinforcement learning algorithms were adapted to suit an active learning paradigm. This adaptation involved updating the agent's value function weights in response to actions and immediate rewards. A key feature of our implementation is the introduction of eligibility traces that decay according to the parameter λ [4].

Eligibility traces are a mechanism that temporarily mark recently visited states or taken actions, indicating their potential contribution to future rewards. This mechanism facilitates an incremental learning process that integrates insights from both past and present interactions with the environment, enabling continuous policy refinement. Upon execution of an action by the agent, the algorithm updates the eligibility traces for all features to reflect their recency and frequency. Subsequently, the value function weights are adjusted based on the temporal-difference error, modulated by the eligibility traces, which allows for a sophisticated learning process attuned to the sequence of visited states. The incorporation of eligibility traces enables the agent to learn from a series of events over time, providing a mechanism for the agent to develop a strategic understanding of the environment that accounts for both immediate and future consequences of actions.

A. Algorithm Adaptation

Prerequisites: Our initial phase entailed the adaptation of the existing algorithms to the domain of active reinforcement learning. This adaptation involved the incorporation of a policy improvement mechanism, deviating from the predetermined policy outlined in the algorithm pseudocode. Additionally, we introduced the epsilon-greedy strategy to foster exploration of the state space, enhancing the agent's capacity to discover optimal actions within its environment.

I. Semi-Gradient TD(λ): The Semi-Gradient TD(λ) algorithm, while powerful in enhancing the learning capabilities of the agent through eligibility traces and incremental updates, is designed with computational efficiency in mind. The updates to the weights and eligibility traces, though iterative and occurring at each step of an episode, are computationally straightforward. This efficiency arises from the linear function approximation [8,9] and the direct method of applying the temporal-difference error to update the value function's weights.

The computational simplicity of semi gradient TD lambda makes it an appealing choice for applications where resources are limited or when the agent needs to operate in real-time environments. However, it is important to note that the choice of features and the size of the feature set can significantly influence the computational load. Careful feature selection is therefore paramount to maintaining the balance between learning efficacy and computational demand.

$$z \leftarrow \gamma\lambda\, z \;+\; \nabla\hat{v}(S, w) \qquad (1)$$

II. True Online TD(λ): True Online TD(λ) builds upon the foundations of the Semi-Gradient method, enhancing its update equations to better accommodate overlapping data observations from similar states [4]. Unlike the Semi-Gradient TD(λ) algorithm, which updates the trace using the partial derivative of the feature vector, True Online TD(λ) introduces a more sophisticated approach, where the trace vector is updated directly with the components of the feature vector itself. This refinement facilitates faster assimilation of experience and adjustments to the policy.

$$z \leftarrow \gamma\lambda\, z \;+\; (1 \;-\; \alpha\gamma\lambda\, z^{T} x)\, x \qquad (2)$$

$$w \leftarrow w \;+\; \alpha(\delta \;+\; V \;-\; Vold)z \;-\; \alpha(V \;-\; Vold)x \qquad (3)$$

However, this enhancement comes at the cost of increased memory requirements, approximately 50% higher than those of the Semi-Gradient method [5]. This is attributed to the additional inner product involved in the eligibility trace update. The computational complexity of each update step however remains consistent at O(d), where d represents the dimensionality of the feature vector. Consequently, True Online TD(λ) exhibits enhanced efficiency in learning and policy optimization compared to Semi-Gradient TD(λ), justifying the trade-off of increased computational cost for improved performance.

B. Linear Function Approximation

To manage the expansive state space of Pacman, both semi-gradient, and true online TD λ algorithms employ linear function approximation [8, 9]. This technique enables the representation of the value function as a weighted sum of selected features, which in our case include Pacman's proximity to the closest food pellets and number of ghosts 1 step away. Effective feature engineering was crucial in this context, given its significant impact on the learning efficacy and the agent's capability to generalize across various game states.

C. Environment Dynamics and Testing

Testing Environment: We conducted extensive testing across a variety of generated environments, varying in layout complexity and adversary presence. The layouts were generated using the recursive backtracking algorithm, which generates a grid, and carves a passage through the grid, and recursively does this for all cells with uncarved adjacent walls, until all cells are visited. The food pellets are added by substituting the walls for food pellets while carving a path through the grid. The size of the grid is varied randomly, and the number of the ghosts generated is dependent upon the area of the maze and a difficulty factor, to control the difficulty of the game.

Performance Metrics: Key metrics include the average score across different environments, speed of convergence and the stability of the policy across different settings.

D. Computational Considerations

Optimizing computational efficiency was crucial to our approach. Operations critical to function approximation were streamlined, and resource utilization was carefully managed. This focus on efficiency facilitated extensive testing within practical timeframes, enabling a thorough examination of our agents' performance across numerous scenarios.

The successful application of Semi-Gradient TD(λ) and True Online TD(λ) methodologies in an interactive reinforcement learning setting underscores their potential for broader applicability in complex, dynamic systems.

III. Results, Analyses, and Discussions

The comparative analysis of Semi-Gradient TD(λ), True Online TD(λ), and the benchmark Approximate Q-learning algorithm yielded insightful results. Our simulations, spanning a multitude of randomly generated environments, provided a comprehensive assessment ground for each method's performance [2].

A. Convergence and Stability

We present an analysis of the convergence rates exhibited by Semi-Gradient TD(λ), True Online TD(λ), and the benchmark Approximate Q-learning algorithm, depicted in Figure 1. This graph represents the variance in average scores sampled at intervals of 100 episodes plotted against the total number of episodes.

The comparison reveals a greater variation in the score generated using True Online TD(λ) and Semi-Gradient TD(λ) in contrast to the Approximate Q-Learning algorithm. This variance appears to stem from the calculation methodology of delta values, which appears to be more sensitive to the behavior of the feature approximation function.

Furthermore, our analysis suggests a marginally quicker convergence of True Online TD(λ) and Semi-Gradient TD(λ) to a stable value compared to the Approximate Q Agent. Such visual comparison of algorithms help in understanding the practical implications of the theoretical differences among the approaches [6].

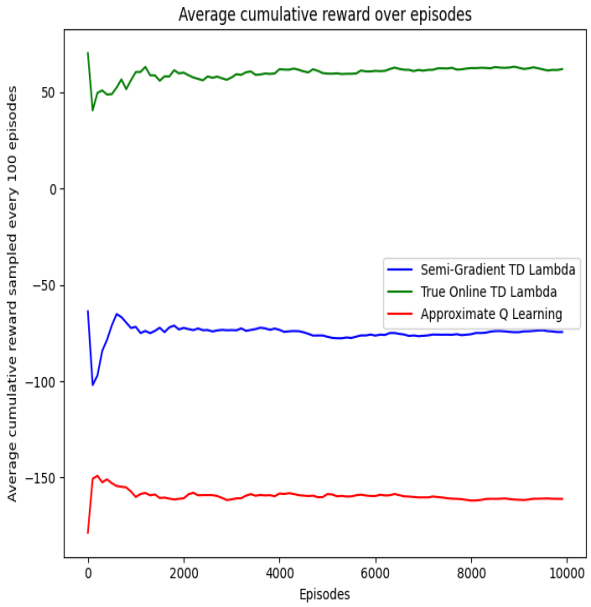

Fig 1. Convergence Rates of Learning Algorithms in Dynamically Generated Pacman Environments [2]

B. Comparative Analysis and Statistical Significance

Utilizing the paired Student's T-test to evaluate the performance metrics across different environmental setups confirmed that the enhancements in learning rates and policy optimization attributed to Semi-Gradient TD(λ) and True Online TD(λ) methods were not incidental.

The statistical significance of these improvements, with p-values consistently below the conventional threshold, validates the efficacy of these TD(λ) methods in fostering rapid and robust learning strategies. This rigorous statistical analysis provides a concrete foundation for asserting the superiority of TD(λ) methods over simpler RL approaches in environments characterized by high variability and dynamic elements.

The results derived from the Shapiro-Wilk test confirm the fulfillment of the normality assumption necessary for the Paired T-test across all comparisons as shown in Table I.

TABLE I
NORMALITY ASSUMPTION USING SHAPIRO WILK TEST

Algorithm	Shapiro statistic	P value
Semi gradient TD(λ) vs Approximate Q learning	0.98	0.16
True online TD(λ) vs Approximate Q learning	0.99	0.87
True online TD(λ) vs Semi gradient TD(λ)	0.99	0.94

The results obtained from the Paired T-test demonstrate a clear statistical significance in the scores achieved by training Pacman using distinct algorithms. Specifically, Pacman achieves notably higher scores when trained with the True Online TD(λ) algorithm compared to the Semi-gradient TD(λ) algorithm. Furthermore, both True Online TD(λ) and Semi-gradient TD(λ) consistently outperform the Approximate Q Learning agent, as evidenced by the statistical analysis in Table II.

TABLE II
STATISTICAL SIGNIFICANCE USING T TEST

Algorithm	T statistic	P value
Semi gradient TD(λ) vs Approximate Q learning	3.79	0.16
True online TD(λ) vs Approximate Q learning	5.78	9.35×10^{-8}
True onlineTD(λ) vs Semi gradient TD(λ)	2.16	0.03

Computation Time: The time taken to train the different algorithms ws compared, but there was no significant difference observed between their training times.This observation may be attributed to the constraints posed by our experimental domain, namely the Pacman game, where training time differences are not pronounced within small state spaces. Moreover, these observations vary across different environments, and our analysis did not yield sufficient

statistical evidence to draw conclusive findings within our experimental setting..

C. Discussion on Findings and Their Implications

Efficiency and Policy Optimization: The observed improvements in learning rates and policy stability under dynamic conditions highlight the effectiveness of eligibility traces combined with linear function approximation in complex environments like Pacman. True Online TD(λ)'s superior performance can be attributed to its more refined handling of eligibility traces, allowing for more precise updates that leverage overlapping state observations.

Adaptability in Dynamic Environments: The robust performance of the TD(λ) methods, particularly under varied and unpredictable game configurations, underscores their suitability for real-world applications where environments are not static and require adaptive decision-making.

Feature Approximation Function: Through experimentation, it can be seen that the feature extractor plays a key role in the performance of the agent. The provided SimpleExtractor function, inherent to the Pacman project, has been meticulously tailored to optimize the performance of the Approximate Q Learning algorithm. However, its applicability to True Online TD(λ) and Semi-Gradient TD(λ) is limited. A more nuanced design of features holds promise for augmenting the efficacy of both True Online TD(λ) and Semi-Gradient TD(λ) algorithms.

Delta values: The rate of change of delta values tends to be higher in True Online TD(λ) and Semi-Gradient TD(λ) compared to Approximate Q-learning. This can be attributed to its efficient reduction of redundant computations and optimization of eligibility traces, in addition to its reliance on the feature approximation function.

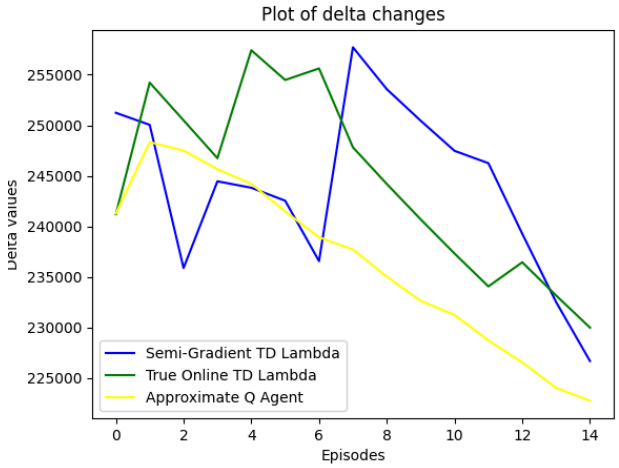

Fig 2. Changes in delta values across different episodes

D. Limitations of the Current Study

The potential improvements in this study include enhancements in the feature approximation function, which is evidenced to be a key metric that influences the performance of the agent.

IV. Conclusion

The project successfully demonstrates that both Semi-Gradient TD(λ) and True Online TD(λ) can be effectively adapted for active RL tasks and offer substantial improvements over traditional Approximate Q-learning in terms of learning speed and adaptability. The True Online TD(λ) algorithm, in particular, exhibited superior performance metrics across most tested environments in the pacman domain [2]. These results suggest that eligibility traces, when combined with linear function approximation [8,9], provide a powerful mechanism for enhancing learning in complex, dynamic systems like video games.

References

[1] R. S. Sutton and A. G. Barto, "Reinforcement Learning: An Introduction," 2nd ed., MIT Press, Cambridge, MA, 2018.

[2] J. DeNero and D. Klein, "The Pac-Man Projects," University of California, Berkeley. http://ai.berkeley.edu/project_overview.html.

[3] T. P. Lillicrap et al., "Continuous control with deep reinforcement learning," in Proc. Int. Conf. Learn. Represent., San Juan, Puerto Rico, 2016.

[4] H. van Seijen, A. R. Mahmood, P. M. Pilarski, M. C. Machado, and R. S. Sutton, "True Online Temporal-Difference Learning," Journal of Machine Learning Research, 2016.

[5] R. S. Sutton, "Learning to predict by the methods of temporal differences," Machine Learning, vol. 3, 1988.

[6] P. Dayan, "The convergence of TD(λ) for general λ," Machine Learning, vol. 8, 1992.

[7] H. van Hasselt and M. Wiering, "Reinforcement learning in continuous action spaces," in Proc. IEEE Symposium on Approximate Dynamic Programming and Reinforcement Learning, Nashville, TN, USA, 2009.

[8] J. Bhandari, D. Russo, and R. Singal, "A Finite Time Analysis of Temporal Difference Learning With Linear Function Approximation," 2018.

[9] T. Sun, H. Shen, T. Chen, and D. Li, "Adaptive Temporal Difference Learning with Linear Function Approximation," 2020.

[10] H. van Seijen and R. S. Sutton, "True Online TD(lambda)," in Proc. 31st Int. Conf. Mach. Learn., PMLR 32(1):692-700, 2014.