Exploring Reinforcement Learning Algorithms: A Comparative Review of SAC, A2LS, PETS, and Policy Gradient Approaches

Shreyans Solanki
Data Science Mukesh Patel School of Technology, Management & Engineering
Mumbai, India
shreyans.solanki01@gmail.com

Shubham Vishwakarma
Data Science Mukesh Patel School of Technology, Management & Engineering
Mumbai, India
vishwakarmashubham.2503@gmail.com

Shashwat Sharma
Data Science Mukesh Patel School of Technology, Management & Engineering
Mumbai, India
sharmashashwat2002@outlook.com

Swati Vaishnav
Data Science Mukesh Patel School of Technology, Management & Engineering
Mumbai, India
swati.vaishnav@nmims.edu

***Abstract*—Reinforcement Learning (RL) has burst onto the scene as a game-changer in machine learning, teaching agents to tackle tough decisions by learning from their own stumbles and successes. Whether it's guiding robots, dominating video games, or shaking up finance, RL's influence is hard to miss. Still, it's not all smooth sailing—it wrestles with headaches like needing tons of data, shaky training processes, and trouble applying what it learns to new situations. This paper zooms in on four awesome algorithms—Soft Actor-Critic (SAC), Automated Auxiliary Loss Search (A2LS), Probabilistic Ensembles with Trajectory Sampling (PETS), and Proximal Policy Optimization (PPO)—each bringing its own clever fix to the table. SAC keeps things lively with its entropy trick to encourage exploration, A2LS gets creative by auto-crafting loss functions, PETS makes every bit of data count with its model-based smarts, and PPO keeps things steady with its clipped updates. Together, they show off RL's incredible range and light the way for exciting new advances.**

***Index Terms*—Reinforcement Learning, Soft Actor-Critic, A2LS, PETS, PPO, Sample Efficiency**

I. Introduction

Reinforcement Learning (RL) has emerged as a powerful framework within machine learning, offering a way to solve complex, step-by-step decision-making puzzles. Its impact is increasingly visible across a wide array of fields, from enabling robots to navigate dynamic environments and guiding self-driving cars, to optimizing financial trading strategies and achieving superhuman performance in complex games. At its core, RL empowers an agent to learn through direct interaction with its environment, gradually discovering strategies that maximize long-term rewards. This process of trial and error is fundamental to its success, allowing it to tackle problems where optimal solutions are not immediately obvious.

However, despite its remarkable successes, RL is not without its challenges. The journey to developing an effective RL agent often involves significant hurdles that researchers are actively working to overcome. Many algorithms require a vast number of practice runs to learn effectively, which can be costly and time-consuming in real-world applications.

Furthermore, the training process can be notoriously unstable, with performance fluctuating dramatically due to unpredictable environmental factors and reward signals. Perhaps the most classic challenge is striking the perfect balance between exploration—trying new actions to discover potentially better strategies—and exploitation—sticking with known actions that have yielded good results in the past. Addressing these core issues is crucial for unlocking the next wave of advancements in the field.

A. Statement of the Problem

Effective data collection is crucial in research, yet challenges arise due to unfamiliarity with collection techniques and difficulties in selecting representative samples. These challenges can affect the quality and reliability of research results. For a comprehensive overview of RL and its challenges, see [1].

B. Aims of the Study

The study aims to:

1) Provide a comprehensive review of four seminal studies in RL.
2) Analyze and compare the methods proposed in these studies.
3) Identify strengths, weaknesses, and potential research directions in RL.

C. Research Questions

This paper addresses the following questions:

1) What are the main contributions of the SAC, A2LS, PETS, and Policy Gradient Methods?
2) How do these algorithms compare in terms of sample efficiency, stability, and applicability?
3) What are the key challenges and future directions in RL research?

D. Data Collection and Procedure

This review builds on 17 articles from conferences, journals, and arXiv, focusing on SAC, A2LS, PETS and PPO. We analyzed their methods and impacts for a balanced RL perspective.

II. Literature Review

Reinforcement Learning is all about an agent figuring things out by engaging with its surroundings, aiming to nail down the best strategy for racking up the most rewards over time. You've got different methods like value-based techniques (think Q- Learning or DQN), policy-based ones (like REINFORCE or PPO), and actor-critic hybrids (such as A3C or SAC), each bringing its own spin to the table.

A. Value-based Methods

Value-based methods in Reinforcement Learning (RL) focus on estimating the value function, which predicts the expected cumulative reward for a state or action. These methods improve decision-making by using the value function instead of directly optimizing the policy.

1) Q-Learning: Q-Learning is an off-policy algorithm that uses temporal difference learning. It relies on epsilon-greedy to explore new actions or stick with known good ones [2].

2) Double Q-Learning: Standard Q-Learning can overestimate values by using the same estimates for choosing and evaluating actions. Double Q-Learning fixes this with two separate Q-value estimators, boosting performance and stability [3].

3) Deep Q-Networks (DQN): DQN, introduced by Mnih et al. [4], builds on Q-Learning by using deep neural networks to estimate the Q-function, handling complex state spaces like images. It uses experience replay to stabilize training [5].

TABLE I
VALUE-BASED RL METHODS

Feature	Q-Learning	Double Q	DQN
Type	Off-Policy, Tabular	Off-Policy, Tabular	Off-Policy, Neural
Policy	ϵ-Greedy	ϵ-Greedy	ϵ-Greedy
Stability	Moderate	Improved	Improved
Sample Efficiency	Moderate	Moderate	High
Advantages	Simple Design	Reduces Bias	Handles Images
Limitations	Overestimates	Tabular Only	Resource Intensive

B. Policy-based Methods

Policy-based methods directly learn a policy, mapping states to actions, without relying on a value function. They excel in continuous action spaces and stochastic policies, serving as a foundation for advanced reinforcement learning techniques [6].

1) REINFORCE: It is a foundational policy gradient method introduced by Williams in 1992 [7].

2) It optimizes policies by estimating gradients through Monte Carlo sampling, adjusting actions to maximize expected rewards.

3) Proximal Policy Optimization (PPO): PPO which is proposed by Schulman et al. in 2017 [8], is a widely-used policy gradient algorithm that builds on earlier methods like TRPO [9]. It enhances training stability by employing clipped objective functions, effectively constraining policy updates and ensuring robust performance across a range of tasks.

4) Deep Visuomotor Policies: Deep Visuomotor Policies, developed by Levine et al., target robotic applications [10]. They employ convolutional neural networks to map raw visual inputs directly to motor actions for tasks like manipulation.

TABLE II
Comparison of Policy-Based RL Methods

Feature	REINFORCE	PPO	Visuomotor
Type	On-Policy PG	On-Policy PG	End-to-End
Policy	Stochastic	Stochastic	Stochastic
Stability	Low	Moderate–High	Moderate
Sample Efficiency	Low	Moderate	Low
Advantages	Simple, unbiased	Stable via clipping	Integrates perception & control
Limitations	High variance, slow	Clipping sensitive	Data intensive; risk of overfitting

C. Actor-Critic Methods

Actor–Critic methods combine policy-based and value-based approaches by maintaining two separate networks: the *Actor* (policy network) which selects actions, and the *Critic* (value network) which evaluates them. They benefit from variance reduction via the critic's TD updates and can be applied in both on-policy and off-policy settings.

1) ***Advantage Actor–Critic (A2C/A3C):*** A2C synchronizes multiple parallel workers to compute gradients of the advantage function, reducing variance. A3C extends this with truly asynchronous updates, improving exploration and wall-clock efficiency [11].

2) ***Soft Actor–Critic (SAC):*** SAC is an off-policy ac- tor–critic that maximizes a trade-off between expected return and policy entropy, yielding robust, sample-efficient learning with improved exploration [12].

3) ***Deep Deterministic Policy Gradient (DDPG):*** DDPG is an off-policy actor-critic method designed for handling continuous actions, relying on a deterministic policy gradient to make decisions. It uses tricks like experience replay and target networks to keep training steady, though it sometimes trips over itself by overestimating values [13].

4) ***Twin Delayed DDPG (TD3):*** TD3 builds on DDPG by using clipped double Q-learning, delayed policy updates, and target policy smoothing to significantly reduce overestimation and improve stability [14].

TABLE III
COMPARISON OF ACTOR–CRITIC METHODS

Feature	A3C	SAC	DDPG	TD3
Type	On-policy	Off policy	Off policy	Off policy
Policy	Stochastic	Stochastic	Deterministic	Deterministic
Stability	Moderate	High (entropy)	Low–Moderate	Moderate–High
Sample Efficiency	Low	High	High	High
Advantages	Parallel updates, better exploration	Robust, efficient, exploratory	Suited for continuous control	Reduces overestimation, stable
Limitations	High variance, low efficiency	Sensitive to reward scaling	Unstable, overestimation	High compute, tuning sensitive

III. ADVANCED REINFORCEMENT LEARNING ALGORITHMS

This section examines four prominent reinforcements learning (RL) algorithms: SAC, A2LS, PETS, and PPO. These algorithms offer distinct approaches to addressing critical RL challenges, including exploration, training stability, and data efficiency. Their contributions are pivotal in advancing both theoretical research and practical applications.

A. Soft Actor-Critic (SAC)

SAC, introduced by Haarnoja et al. [12], is particularly effective for continuous control tasks. Its defining characteristic is entropy regularization, which optimizes both the expected reward and the policy's entropy. This dual objective encourages exploration, preventing convergence to suboptimal policies and yielding robust, adaptable strategies.

Operating within an off-policy actor-critic framework, SAC employs two Q-networks to estimate the value function, reducing the overestimation bias prevalent in RL methods. The policy, parameterized by a neural network, generates a stochastic distribution over actions, with entropy regularization balancing exploration and exploitation. This design enhances SAC's sample efficiency by leveraging past experiences effectively. In practice, SAC demonstrates strong performance in domains such as robotics, where stability and rapid learning from limited data are essential.

B. Automated Auxiliary Loss Search (A2LS)

A2LS represents a significant advancement in RL by automating the identification of auxiliary losses to improve learning outcomes [15]. These supplementary objectives complement the primary RL goal, often accelerating convergence or enhancing performance in complex tasks. A2LS systematically selects optimal auxiliary losses tailored to specific problems, offering a customized approach to optimization.

The implementation of A2LS varies depending on the context, but its strength lies in its adaptability. In environments where conventional RL objectives are inadequate, A2LS uncovers latent efficiencies, making it a versatile tool. This method is particularly valuable in scenarios where manual loss design is impractical, although its success hinges on the robustness of the search process.

C. Probabilistic Ensembles with Trajectory Sampling (PETS)

PETS, developed by Chua et al. [16], is a model-based RL algorithm distinguished by its data efficiency. It employs an ensemble of probabilistic dynamics models to predict future states and rewards, accounting for environmental uncertainty. By sampling trajectories from these models, PETS identifies actions that maximize long-term returns. This uncertainty- aware planning enhances model-based RL, especially in data- scarce settings, building on earlier methods like PILCO [17] with improved scalability.

The probabilistic and ensemble-based approach of PETS mitigates the impact of model inaccuracies, a frequent challenge in model-based RL. Consequently, it achieves reliable performance with fewer environment interactions, which is advantageous for applications such as robotics or simulations where data collection is costly. PETS exemplifies how predictive modeling can optimize RL efficiency.

D. Policy Gradient Approaches: Proximal Policy Optimization (PPO)

PPO, proposed by Schulman et al. [8], is a leading algorithm within policy gradient methods. It mitigates the instability of policy updates through a clipped objective function, which constrains adjustments to a stable range. This mechanism ensures a balance between learning progress and training consistency.

As an on-policy algorithm, PPO refines the policy iteratively using data from the current policy. Its simplicity and reliability have made it widely adopted across tasks, from gaming to robotic control. Building on foundational methods like REINFORCE, PPO's clipped updates enhance its practicality, establishing it as a benchmark in modern RL.

IV. COMPARATIVE ANALYSIS

This section evaluates the strengths, limitations, and applications of SAC, A2LS, PETS, and PPO, focusing on their approaches to RL challenges—exploration, resource efficiency, and deployment.

Each algorithm has unique characteristics. SAC uses entropy regularization to enhance exploration, suited for robotic manipulation, but its dual Q-networks require substantial computation.

A2LS automates auxiliary loss selection for complex tasks like multi-agent coordination, minimizing manual effort, though its search process is computationally intensive.

PETS employs probabilistic models to optimize data use, ideal for costly scenarios like autonomous vehicle testing. Its ensemble approach ensures robustness, but stochastic environments challenge model accuracy.

PPO provides versatility with clipped updates, excelling in virtual and physical tasks. Its simplicity aids implementation, yet its on-policy nature demands more data than model-based methods.

Exploration differs across methods: SAC promotes diverse actions, PPO refines cautiously, PETS tests within models, and A2LS adapts via tailored losses. SAC and PPO are easily deployable, while PETS needs environmental insight, and A2LS requires significant computational resources.

The table below summarizes these attributes:

TABLE IV
ADVANCED RL ALGORITHMS COMPARISON

Feature	SAC	A2LS	PETS	PPO
Type	MF, Off-policy	Hybrid	MB	MF, On-policy
Policy	Stochastic	Varies	Deterministic	Stochastic
Sample Efficiency	High	Varies	Very High	Moderate
Stability	High	Varies	Moderate	High
Applicability	Continuous Actions	Varies	Continuous Actions	Both

Insights: SAC, A2LS, PETS, and PPO each provide distinct reinforcement learning approaches, addressing specific challenges. Their strengths underscore their value in diverse real-world applications, reinforcing their importance in this review.

- **SAC** employs entropy regularization to boost exploration, ideal for precision tasks like robotic manipulation (e.g., circuit assembly). Its off-policy efficiency uses past experiences, but dual Q-networks demand significant computation.
- **A2LS** creates tailored auxiliary losses for complex scenarios, such as drones in dynamic disaster zones. Its adaptability is a key strength, though finding optimal losses can be resource heavy.
- **PETS** uses probabilistic models to optimize data, critical for data-scarce settings like self-driving car simulations. Its model-based efficiency shines, yet stochastic environments pose challenges.
- **PPO** ensures stable learning with clipped updates, excelling in game AI tasks. Its versatility suits many applications, but it needs more data than model-based methods.

Together, these algorithms showcase reinforcement learning's versatility—SAC for precision, A2LS for complexity, PETS for efficiency, and PPO for reliability—guiding the development of smarter solutions.

V. CONCLUSION

Examining SAC, A2LS, PETS, and PPO reveals a suite of advanced reinforcement learning algorithms, each contributing unique capabilities. SAC's entropy regularization promotes exploration, making it well-suited for high-precision tasks such as robotic control. A2LS automates the design of auxiliary losses to address multifaceted challenges, such as coordinating multiple agents.

PETS leverages model-based planning to maximize data efficiency, which is critical in resource-constrained settings like autonomous vehicle testing. Meanwhile, PPO serves as a versatile and reliable method, performing consistently across diverse domains—from game AI to physical control systems—through its clipped objective function. These algorithms do not merely mitigate reinforcement learning's limitations; they expand its potential by demonstrating how varied strategies can overcome issues like sample inefficiency and training instability. Looking forward, they provide a foundation for developing more robust and practical reinforcement learning systems capable of addressing complex real-world problems

REFERENCES

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

[2] C. J. C. H. Watkins and P. Dayan, "Q-learning," *Mach. Learn.*, vol. 8, no. 3–4, pp. 279–292, May 1992.

[3] H. van Hasselt, "Double Q-learning," in *Proc. Adv. Neural Inf. Process. Syst. (NeurIPS)*, Vancouver, BC, Canada, Dec. 2010, pp. 2613–2621.

[4] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, et al., "Human-level control through deep reinforcement learning," *Nature*, vol. 518, no. 7540, pp. 529–533, Feb. 2015.

[5] J. Fan, Z. Wang, Y. Xie, and Z. Yang, "A theoretical analysis of deep Q-learning," in *Proc. 2nd Annu. Conf. Learn. Dyn. Control (L4DC)*, vol. 120, Berkeley, CA, USA: PMLR, Jun. 2020, pp. 1–4.

[6] R. S. Sutton, D. McAllester, S. Singh, and Y. Mansour, "Policy gradient methods for reinforcement learning with function approximation," in *Proc. Adv. Neural Inf. Process. Syst. (NeurIPS)*, Denver, CO, USA, Nov./Dec. 2000, pp. 1057–1063.

[7] R. J. Williams, "Simple statistical gradient-following algorithms for connectionist reinforcement learning," *Mach. Learn.*, vol. 8, no. 3–4, pp. 229–256, May 1992.

[8] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal policy optimization algorithms," *arXiv preprint arXiv:1707.06347*, Jul. 2017.

[9] J. Schulman, S. Levine, P. Abbeel, M. Jordan, and P. Moritz, "Trust region policy optimization," in *Proc. Int. Conf. Mach. Learn. (ICML)*, Lille, France, Jul. 2015, pp. 1889–1897.

[10] S. Levine, C. Finn, T. Darrell, and P. Abbeel, "End-to-end training of deep visuomotor policies," *J. Mach. Learn. Res. (JMLR)*, vol. 17, no. 1, pp. 1334–1373, Jan. 2016.

[11] V. Mnih, A. P. Badia, M. Mirza, A. Graves, T. Lillicrap, T. Harley, B. Silver, and K. Kavukcuoglu, "Asynchronous methods for deep reinforcement learning," in *Proc. Int. Conf. Mach. Learn. (ICML)*, New York, NY, USA, Jun. 2016, pp. 1928–1937.

[12] T. Haarnoja, A. Zhou, P. Abbeel, and S. Levine, "Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor," in *Proc. Int. Conf. Mach. Learn. (ICML)*, Stockholm, Sweden, Jul. 2018, pp. 1861–1870.

[13] T. P. Lillicrap, J. J. Hunt, A. Pritzel, N. Heess, T. Erez, Y. Tassa, D. Silver, and D. Wierstra, "Continuous control with deep reinforcement learning," *arXiv preprint arXiv:1509.02971*, Sep. 2015.

[14] S. Fujimoto, H. V. Hoof, and D. Meger, "Addressing function approximation error in actor-critic methods," in *Proc. Int. Conf. Mach. Learn. (ICML)*, Stockholm, Sweden, Jul. 2018, pp. 1587–1596.

[15] T. He, Y. Zhang, K. Ren, M. Liu, C. Wang, W. Zhang, Y. Yang, and D. Li, "Reinforcement learning with automated auxiliary loss search," in *Proc. Adv. Neural Inf. Process. Syst. (NeurIPS)*, New Orleans, LA, USA, Dec. 2022, pp. 1–13.

[16] K. Chua, R. Calandra, R. McAllister, and S. Levine, "Deep reinforcement learning in a handful of trials using probabilistic dynamics models," in *Proc. Adv. Neural Inf. Process. Syst. (NeurIPS)*, Montreal, QC, Canada, Dec. 2018, pp. 4754–4765.

[17] M. P. Deisenroth and C. E. Rasmussen, "PILCO: A model-based and data-efficient approach to policy search," in *Proc. Int. Conf. Mach. Learn. (ICML)*, Bellevue, WA, USA, Jun./Jul. 2011, pp. 465–472.