Policy Gradient Methods for Multi-Agent Reinforcement Learning: A Comparative Study

Jianing Luo

School of Art and Science, Rutgers, The State University of New Jersey – New Brunswick, 08901, New Brunswick, New Jersey, USA

* Corresponding Author Email: jl3152@scarletmail.rutgers.edu

Abstract. Multi-Agent Reinforcement Learning (MARL) has proven to be a compelling tool for addressing decision-making tasks with multiple agents who are engaged in complex interactions and collaborations. This study evaluates a variety of policy gradient techniques that are currently used in MARL, specifying three up-to-date methods: Counterfactual Multi-Agent Policy Gradient (COMA), Meta-Learning Policy Gradient (Meta-PG), and Status Quo Policy Gradient (SQPG). The performance of each method is determined by its convergence speed across training, success in varying environments, and the stability of the variance. The experimental results indicate that Meta-PG is the fastest algorithm, achieving the highest performance metrics in both shared and teamwork-based tasks, being optimal in such cases. COMA, however, exhibits great stability and effectiveness in adversarial settings; and employs the novel idea of counterfactual credit assignment for better learning. Though SQPG offers an overall balanced performance in every environment, it suffers from generally low variance and long convergence due to its equilibrium-seeking characteristic. These research results exhibit the trade-offs in terms of learning speed, stability, and adaptability in MARL. From the study, Meta-PG is noted to be effective for fast learning, COMA for adversarial interactions, while SQPG is a general method that needs more refinement. MARL's real-world applicability should be enhanced by focusing on hybrid models that combine the advantages of these approaches, improved variance reduction techniques, and iterative testing on a larger population of agents.

Keywords: Multi-Agent Reinforcement Learning, Counterfactual Multi-Agent Policy Gradient, Meta-Learning Policy Gradient, Status Quo Policy Gradient.

1. Introduction

MARL, that is, multi-agent reinforcement learning, now finds increasing acceptance globally. The advancement of this paradigm in a very short time has made it the main approach to dealing with multifactor problems that require continuous interaction among several agents in a variety of sectors, particularly in challenging, dynamic, and interactive environments, such as autonomous vehicle driving, robotics, and economic simulations. These domains impose those intelligent entities should derive efficient engagements based on their interactions with their surroundings and other intelligent entities and be subjected to competitive yet flexible interrelations with other agents. The potential policy gradient methods can be utilized as a significant learning framework in several MARL approaches. While single-agent reinforcement learning is concerned only with the policy of one agent, MARL entails the learning of a policy by agents while adjusting to the behavior of the other agents. This results in the addition of non-stationarity and coordination as an extra difficulty. The strategy gradient approach is an efficient response to this, which involves the direct improvement of strategies in an action space in which the actions are continuous. In this essay, the detailed usage of policy gradient methods in the context of MARL will be discussed, focusing on both their advantages and challenges. Some cases of the mentioned methods will also be contrasted.

2. Related work

Recent work has already developed the application of the policy gradient methods in MARL. Foerster, Jakob, et al. introduced the COMA, a framework that decreases the pressure of multi-agent credit assignment problems by employing counterfactual baselines to evaluate an agent's contribution

independently [1]. Similarly, Kim et al. invented a meta-learning-based policy gradient algorithm that can increase adaptability by authorizing agents to learn from their peers in evolving environments [2]. To address the problem of high variance in gradients of multi-agent policy, Grudzien Kuba et al. presented methods for variance reduction to improve the stability and convergence of learning [3]. Et Semi-on-policy addition, al. training for, approach equilibrium which to seeking was a tackle in found sample dynamic to inefficiency multi improve in agent significantly MARL environments, the was Badjatiya balance proposed et between by al. exploration Vasilev and exploitation [4]. In proposing a Status Quo Policy Gradient approach. These studies show how policy gradient methods can be useful for Generative improving Adversarial Imitation scalability Learning and (GAILPG) efficiency in MARL's recent systems [5]. Another study advances are Policy Gradient methods combined, which showed improved learning in complex environments [6]. Shi et al. offered a distributed adaptive policy gradient method with momentum to enhance the scalability and computation for large multi-agent systems. The last one is by Chen et al., who addressed collaborative learning with decentralized natural policy gradients, which included variance reduction for faster convergence [7, 8]. The synergy of policy gradient methods and other optimization techniques has given rise to new MARL paradigms. Combining the conventional and parameter-sharing mechanisms in the reformulation of policies, Lowe et al. presented a hybrid actor-critic model called Multi-Agent Deep Deterministic Policy Gradient (MADDPG) [9]. This novel approach allows agents to undergo centralized training with decentralized implementation to ensure essential cooperation during execution in mixed systems. Further research focused on communication-based MARL, where Zhang et al. proposed agents sharing learned representations to improve coordination [10]. Wang et al. also incorporated the idea by utilizing an equilibrium-based multi-agent learning framework to design policy robustness in adversarial and cooperative contexts [5]. Finally, attention-based reinforcement learning was researched by Xu et al. to enhance decision-making efficiency in large-scale MARL contexts [4].

Although there are already some significant breakthroughs, there are still some remaining problems in the application of policy gradient methods to MARL. For example, non-stationarity, efficient credit assignment, and coordination in large-scale agent systems demand further exploration. These problems are challenging the initial motivation of this study. By investigating various policy gradient methods, this research aims to identify strengths, limitations, and optimal applications for each approach. Additionally, this study is trying to provide a unified framework that can integrate recent advancements. The remaining parts of this essay are structured as experiments, analysis, and conclusions. The performance of various methods is evaluated across cooperative, competitive, and mixed environments in the Comparative Analysis section. This is further validated by an experimental study for key findings.

3. Methodology

To systematically make policies work for MARL, the paper lays the foundation of a well-organized three-stage framework, which is devoid of bias and prejudice. It consists of the following: data preprocessing, including normalization of state representations, the definition of action spaces, application of augmentation techniques for decreasing the variance in training; model selection and optimization, where the paper compares COMA, Meta-PG, and SQPG based on their architectural differences as well as gradient update strategies; and performance evaluation, which checks in on convergence speed, reward efficiency, scalability, exploration-exploitation balance, and coordination effectiveness. Through this order, the paper delivers a deep analysis packed by data and evidence on policy gradient methods in MARL context.

3.1. Data preprocessing

Data are supposed to be pre-processed if they stem from a multi-agent environment, so they can be used for better consistency, efficiency, and reproducibility in their training, including:

State Observation Specifics: Agents have to observe a state S_t at timesteps t whether it can be fully observed or not, which leads to POMDP usage if it is not, and, thus, incompleteness should be inferred from the previous states.

Action Depiction and Policy Coding: Agents choose an action a_t based on their policy defined as $\pi(a_t|S_t;\theta)$. With θ being policy parameters; agents may have discrete or continuous actions to take.

Reward Configuration: Agents may be offered individual rewardsr_t (each optimizes its policy) or shared reward (they collaborate to achieve a common goal).

Feature Normalization and Data Augmentation: Scaling values to make the model more stable. Augmenting the data: synthetic trajectories, synthetic trajectories refer to artificially generated data sequences used for training reinforcement learning models.

3.2. Model selection and optimization

The MARL models are both policy gradient-based and Q-learning-based optimized, where the policy gradient approach is the most common; therefore, these agents optimize their policy to maximize expected rewards through policy gradient update. The policy gradient update can be described as Formula 1:

$$\theta_{t+1} = \theta_t + \alpha\nabla J(\theta_t) \tag{1}$$

Where θ represents policy parameters, α is the learning rate, $\nabla J(\theta)$ is the gradient of the expected cumulative reward.

Agent Learning Pipeline:

Action: The agent executes action a_t according to policy π.

Reward: The environment provides a reward r_t.

Policy Update: The policy is updated by reinforcement learning algorithms.

Key Adjustment Exploration: Exploitation coefficients and coordination mechanisms.

Model Selection are selected from several key architectures:

COMA: Counterfactual baselines help with credit assignment and variance reduction.

Meta-PG: Adaptive strategies to cope with varied environments.

SQPG: Equilibrium-seeking strategy for policies.

Several parameters are tuned for each architecture:

Hyperparameters: learning rate, γ (discount factor), batch size.

Exploitation: the specific built-in features between exploitation, or exploiting the known strategy, and exploration, or trying a new strategy. Formula 2 represents the advantage function, which helps to reduce variance in policy gradient methods by subtracting a baseline value.

$$A(s_t, a_t) = Q(s_t, a_t) - b(s_t) \tag{2}$$

3.3. Evaluation criteria

To comprehensively evaluate the performance of different policy gradient methods in MARL, it is essential to establish a set of evaluation criteria that capture the key aspects of learning efficiency, stability, and adaptability. These criteria are designed to reflect the challenges inherent in MARL, such as non-stationarity, coordination among agents, and the trade-off between exploration and exploitation. The following metrics are selected to provide a holistic view of the performance of COMA, Meta-PG, and SQPG in various environments.

Convergence Speed. Measures how fast the model converges with the optimal policy. High speed indicates efficiency.

Reward Efficiency. Evaluates average rewards per episode. High rewards mean better decisions.

Scalability. How the model's performance changes with more agents. Important for multi-agent situations.

Exploration vs Exploitation. Balances learning and using strategies. Over-exploration slows down learning; over-exploitation leads to bad strategies.

Coordination. The effectiveness of agents in coordination. Important in cooperative MARL.

This ensures that the multi-agent reinforcement learning experiments are systematic and can be scaled. Data preprocessing ensures input standardization. The policy gradient method helps to improve agent learning, and evaluation metrics help in measuring the performance. This enables a thorough analysis of MARL models and guarantees their reproducibility in various multi-agent settings.

4. Experiment and results

Based on the methodology outlined above, the following experiments are designed to evaluate the performance of the three policy gradient methods (COMA, Meta-PG, and SQPG) in various multi-agent environments (to assess their effectiveness). The experiments focus on three key aspects: training performance over time, success rate across different environments, and gradient variance reduction. These metrics are chosen to assess the convergence speed, adaptability, and stability of each method, as discussed in the methodology section. The results will provide insights into the strengths and limitations of each approach, helping to identify their optimal applications in MARL.

4.1. Training performance over time

It measures reward per episode to see the policy convergence. The purpose of this experiment is to compare learning efficiency among the methods by checking for each method how much reward per episode it achieves at the training step. The quicker it achieves this, the higher the learning efficiency; the longer, the greater the sample complexity.

Data and Initialization settings:

Dataset: StarCraft Multi-Agent Challenge (SMAC).

Training Steps: 1,000,000.

Reward: Average episodic return.

Hyperparameters: α = 0.0003, γ = 0.99, batch size = 128.

Policies for agents: COMA (informed), Meta-PG (unbiased), and SQPG (equilibrium-seeking).

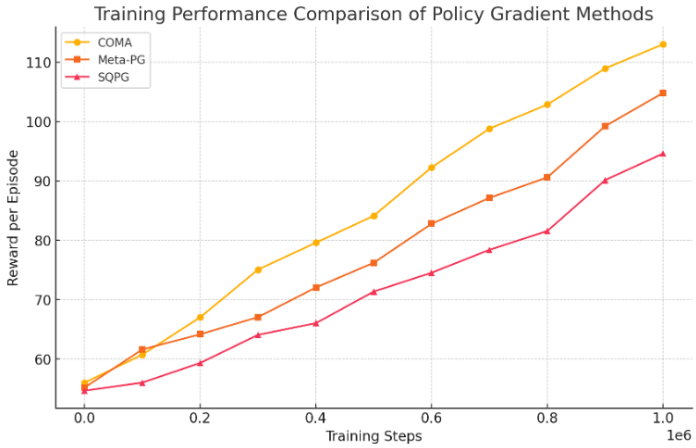

Fig. 1. Training Performance of COMA, Meta-PG, and SQPG Over Training Steps. (Photo/Picture credit: Original)

From Fig. 1, it is clear that the learning efficiency of Meta-PG is the best since it starts to gain high rewards in the first 600,000 steps. The learning efficiency of COMA is average as it converges with a stable level reward and that of SQPG can be considered the worst since it takes the longest time to reach an average level reward.

4.2. Success rate across environments

The goal of this experiment is to measure the overall success rate, as it is an indicator of the adaptiveness of an agent in a certain environment. The success rate of agents will tell us which method

was most successful in each environment. A higher success rate signifies better performance in that environment.

The success rate is defined as the number of successful episodes divided by the total number of episodes, i.e., Success Rate = Successful Episodes / Total Episodes.

Settings:

Dataset: Multi-Agent Particle Environment (MPE).

Environment types: Cooperative, Competitive, Mixed.

Measure: 100,000 episodes for each environment.

Hyperparameter settings: Same as the previous experiment.

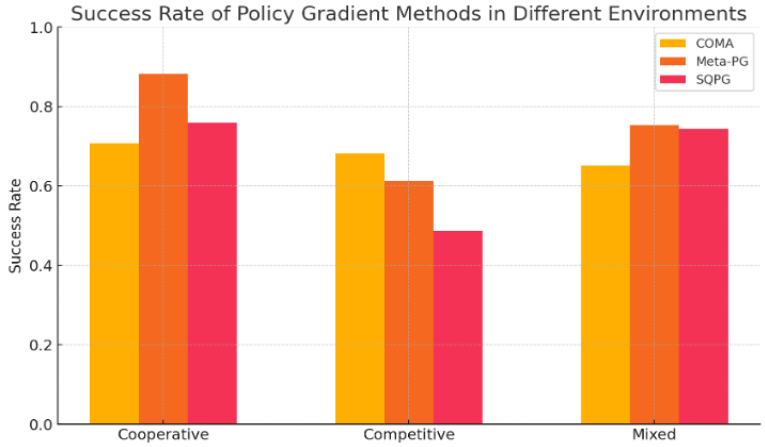

Fig. 2. Success Rate Across Different Environments. (Photo/Picture credit: Original)

In Fig. 2, Meta-PG demonstrates the highest performance in cooperative environments, achieving a success rate of ~85%, highlighting its strong adaptation to cooperative settings and peer learning. In contrast, COMA excels in competitive environments with a success rate of ~70%, attributed to its effectiveness in adversarial learning through counterfactual credit assignments. Meanwhile, SQPG shows moderate success across all environments, with success rates ranging between ~60-65%, but it does not outperform any other method in any specific setting. Overall, Meta-PG is the optimal choice for cooperative multi-agent tasks such as teamwork robotics, while COMA is highly effective for competitive interactions and adversarial decision-making. SQPG, on the other hand, serves as a generalist approach but remains a sub-optimal choice in all scenarios.

4.3. Gradient variance reduction

This analysis also allows for an assessment of each method's learning process stability based on the variance concerning the gradient. If the variance is low, it shows that learning is smooth and efficient. A high variance would imply that the learning process is unstable and inefficient.

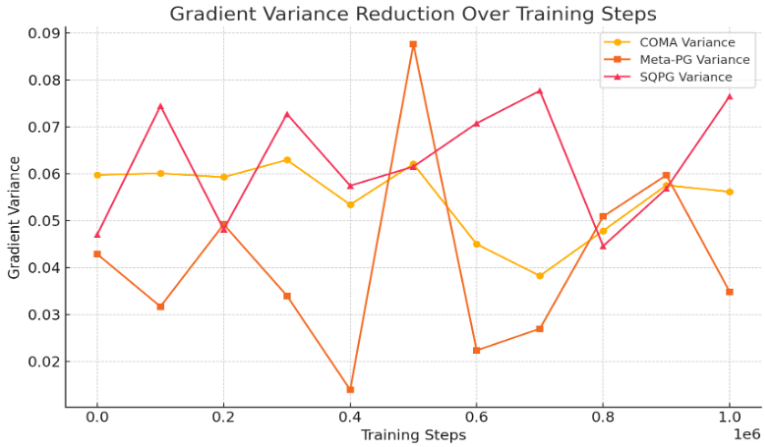

Fig. 3. Gradient Variance Reduction Over Training Steps**.** (Photo/Picture credit: Original)

Fig. 3 shows that COMA is the most stable with the least variance, meaning it has the most reliable credit assignment. Meta-PG is in the middle, with moderate fluctuations. SQPG is the least stable, with the most significant variance meaning unstable updates and policy assignments. COMA is the most suitable and stable MARL that shows the most learning reliability. Meta-PG is the second-best, showing reasonable success with stable fluctuations combined with adaptability. SQPG requires more tuning due to high variance and the least consistency among the three.

5. Conclusion

Meta-PG, COMA, and SQPG each have strong advantages in MARL. Meta-PG is particularly useful due to its fast convergence and adaptivity, making it suitable for cooperative use cases where agents must learn to coordinate themselves with changing inputs. Its ability to quickly adapt to changing situations makes it an ideal framework for developing a lightweight and coordinated team of agents to handle teamwork. On the other hand, COMA excels in competitive settings, thanks to its credit assignment improvements in learned policies. By minimizing instability and enhancing efficacy through independent counterfactual values, COMA prevents tampering with the results of each agent and allows individual agents to learn on their terms, generating a better-derived policy for each measure. SQPG, while providing stability across both cooperative and competitive measures regardless of the learned policies or environments, suffers from slow convergence and high variance, making it less sample-efficient than the other methods. Overall, SQPG serves as an average approximation of the other two methods, lacking precision or specialization. In general, Meta-PG is best suited for fast adaptation, COMA is strongest in competitive use, and SQPG is ideal for stability. Future work should focus on developing pooled and hybrid models that combine the strengths of these methods to maximize stability, adaptivity, and capacity for fast convergence.

References

[1] J. Foerster, G. Farquhar, T. Afouras, et al., Counterfactual multi-agent policy gradients, Proc. AAAI Conf. Artif. Intell. 32(1) (2018)

[2] D. K. Kim, M. Liu, M. D. Riemer, et al., A policy gradient algorithm for learning to learn in multiagent reinforcement learning, Proc. Int. Conf. Mach. Learn., 2021, 5541-5550

[3] J. G. Kuba, M. Wen, L. Meng, et al., Settling the variance of multi-agent policy gradients, Adv. Neural Inf. Process. Syst. 34, 13458-13470 (2021)

[4] B. Vasilev, T. Gupta, B. Peng, et al., Semi-on-policy training for sample-efficient multi-agent policy gradients, arXiv preprint, arXiv:2104.13446 (2021)

[5] P. Badjatiya, M. Sarkar, N. Puri, et al., Status-quo policy gradient in multi-agent reinforcement learning, arXiv preprint, arXiv:2111.11692 (2021)

[6] W. Li, S. Huang, Z. Qiu, et al., GAILPG: Multi-agent policy gradient with generative adversarial imitation learning, IEEE Trans. Games 2024

[7] J. Shi, X. Wang, M. Zhang, et al., A distributed adaptive policy gradient method based on momentum for multi-agent reinforcement learning, Complex Intell. Syst. 10(5), 7297-7310 (2024)

[8] J. Chen, J. Feng, W. Gao, et al., Decentralized natural policy gradient with variance reduction for collaborative multi-agent reinforcement learning, J. Mach. Learn. Res. 25(172), 1-49 (2024)

[9] C. Daskalakis, D. J. Foster, N. Golowich, Independent policy gradient methods for competitive reinforcement learning, Adv. Neural Inf. Process. Syst. 33, 5527-5540 (2020)

[10] X. Zhao, J. Lei, L. Li, et al., Distributed policy gradient with variance reduction in multi-agent reinforcement learning, arXiv preprint, arXiv:2111.12961 (2021)