Policy search and policy value iteration algorithm in Reinforcement Learning

M. Jahnavi *CSE(AIML)* 221FA18104

*Vignan University*

Vadlamudi,Guntur

T.Brahmaih *CSE(AIML)* 221FA18114

*Vignan University*

Vadlamudi,Guntur

Ram sai *CSE(AIML)* 221FA18124

*Vignan University*

Vadlamudi,Guntur

Aditya *CSE(AIML)* 221FA18175

*Vignan University*

Vadlamudi,Guntur

***Abstract*—Reinforcement Learning (RL) focuses on optimiz- ing decision-making in uncertain environments, often modeled as Markov Decision Processes (MDPs). Policy Search directly optimizes policies, making it suitable for high-dimensional spaces, while Policy Iteration iteratively evaluates and improves policies. Value Iteration, a more computationally efficient approach, updates value functions directly using the Bellman Optimality Equation. This paper explores these methods, comparing their strengths, applications, and suitability for different problem domains.**

I. Introduction

spaces where value-based methods become computationally infeasible.

1. *Gradient-Based Policy Search*

One popular approach is the policy gradient method, where the policy is parameterized by a set of parameters *θ*. The optimization follows the policy gradient theorem:

∇ *J*(*θ*) = *E* "Σ ∇ log *π* (*a* |*s* )*R* # (1)

*θ*

*t*

*θ*

*θ*

*t*

*t*

*t*

Reinforcement Learning (RL) is an area of artificial in- telligence that trains agents to make sequential decisions in dynamic environments. Many RL problems are modeled using **Markov Decision Processes (MDPs)**, which consist of states, actions, transition probabilities, rewards, and discount factors. The primary goal in RL is to find an *optimal policy*, a mapping from states to actions, that maximizes the expected cumulative reward over time.

Several approaches exist for solving MDPs, including **Pol- icy Search**, **Policy Iteration**, and **Value Iteration**. *Policy Search* directly optimizes the policy function and is partic- ularly useful for large-scale and continuous action spaces. *Policy Iteration* refines policies iteratively through cycles of policy evaluation and improvement, ensuring convergence to an optimal solution. *Value Iteration*, a computationally effi- cient alternative, updates value functions using the Bellman Optimality Equation, accelerating convergence to the optimal policy.

Understanding the strengths and trade-offs of these methods is essential for selecting the appropriate approach in real- world applications, such as robotics, autonomous systems, and financial decision-making. This paper provides a detailed dis- cussion of these three techniques, their theoretical foundations, implementation, and comparative advantages.

II. Policy Search

Policy Search methods aim to find the optimal policy directly, without computing explicit value functions. These methods are particularly useful in large or continuous action

where *J*(*θ*) is the expected reward and *πθ* is the policy

function.

1. *Gradient-Free Policy Search*

Methods like Evolutionary Strategies and Genetic Algo- rithms fall into this category. These algorithms do not require gradient computation but instead rely on sampling-based op- timization.

III. Policy Value Iteration Algorithm

Policy Value Iteration is a combination of Policy Iteration and Value Iteration, aimed at finding an optimal policy effi- ciently. The key steps include:

1. *Policy Iteration*
   1. **Policy Evaluation:** Compute the value function *V π* for the current policy:

Σ Σ

*V π*(*s*) = *π*(*a*|*s*) *P* (*s′*|*s, a*)[*R*(*s, a, s′*)+*γV π*(*s′*)] (2)

*a s′*

* 1. **Policy Improvement:** Update the policy using:

Σ

*π′*(*s*) = arg max *P* (*s′*|*s, a*)[*R*(*s, a, s′*) + *γV π*(*s′*)] (3)

*a*

*s′*

1. *Value Iteration*

Instead of full policy evaluation, Value Iteration updates value functions directly:

Σ

*V* (*s*) = max *P* (*s′*|*s, a*)[*R*(*s, a, s′*) + *γV* (*s′*)] (4)

*a*

*s′*

IV. Conclusion

This paper discussed three fundamental approaches for solv- ing Markov Decision Processes in Reinforcement Learning: Policy Search, Policy Iteration, and Value Iteration. Policy Search is useful in high-dimensional spaces, Policy Itera- tion provides iterative policy refinement, and Value Iteration efficiently updates value functions. The choice of method depends on computational feasibility and the complexity of the environment. Future research may explore hybrid approaches to leverage the strengths of each method in real-world appli- cations.

References

[1] R. Sutton and A. Barto, *Reinforcement Learning: An Introduction*, MIT Press, 2018.

M. van Otterlo and M. Wiering, *Reinforcement Learning and Markov Decision Processes*, Springer, 2012.

D. Silver, *Lecture Notes on Reinforcement Learning*, Univer- sity College London, 2015.