# Preliminaries needed to understand Proximal Policy Optimization Algorithms

Notes on discussion and derivations from Sutton’s book and John Schulman’s articles

D. Gueorguiev 12/13/23

Table of Contents

[Preliminaries needed to understand Proximal Policy Optimization Algorithms 1](#_Toc153627034)

[A bit of theory on Policy Gradient Reinforcement Learning Methods 1](#_Toc153627035)

[Policy evaluation (Prediction) 1](#_Toc153627036)

[Appendix 2](#_Toc153627037)

[Solution of the Bellman system of equations for 2](#_Toc153627038)

[Bibliography 3](#_Toc153627039)

## A bit of theory on Policy Gradient Reinforcement Learning Methods

Assumptions:

The environment can be represented by a finite MDP

This is equivalent of saying that its state , action and reward sets are finite, and its dynamics is given by a set of probabilities , for all and ( is plus a terminal state if the problem is episodic).

We would like to compute value functions to organize and search for good policies.

The optimal value functions satisfying the Bellman’s optimality equations were derived and discussed in (Gueorguiev, 2023) (see Eq. (22) and (23)).

(1)

(2)

### Policy evaluation (Prediction)

**Definition**: *policy evaluation* - computation of the state-value function for a given policy . This is also known as the *prediction problem*.

Question: How to compute the state-value function for an arbitrary policy .

From (11) and (12) in (Gueorguiev, 2023) we can write:

(3)

(4)

(5)

where is the probability of taking action in state under policy , and the expectations are subscribed by to indicate that they are conditional on being followed.

The existence and uniqueness of are guaranteed as long as either or eventual termination is guaranteed from all states under the policy .

If the environment’s dynamics are completely known, then (5) is a system of simultaneous linear equations in unknowns (the ).

## Appendix

### Solution of the Bellman system of equations for

We notice that the Bellman system of equations with respect to (5) can be rewritten as:

(A1)

hence

(A2)

The left-hand side of (A2) can be rewritten as:

(A3)

The right-hand side of (A2) are rearranged as :

(A4)

we denote with and the following expressions:

(A5)

(A6)

(A7)

Using (A3)-(A7) in (A2) leads to :

(A8)

(A8) represents a linear system of equations with respect to the unknowns .

(A8) in matrix form:

(A9)

The solution of this linear system is given by:

# Bibliography

Gueorguiev, D. (2023, Nov 26). *Note on Q functions and V functions in Reinforcement Learning.* Retrieved from Reinforcement Learning and Game Theory Repository: https://github.com/dimitarpg13/reinforcement\_learning\_and\_game\_theory/blob/main/docs/Note\_on\_Q\_functions\_in\_Reinforcement\_Learning.pdf

Heeswijk, W. v. (2022, Apr 9). *Policy Gradients In Reinforcement Learning Explained.* Retrieved from Towards Data Science: https://towardsdatascience.com/policy-gradients-in-reinforcement-learning-explained-ecec7df94245

Heeswijk, W. v. (2022, Nov 29). *Proximal Policy Optimization (PPO) Explained.* Retrieved from Towards Data Science: https://towardsdatascience.com/proximal-policy-optimization-ppo-explained-abed1952457b

John Schulman, e. a. (2018, Oct 20). *High-Dimensional Continuous Control Using Generalized Advantage Estimation.* Retrieved from arxiv.org: https://github.com/dimitarpg13/reinforcement\_learning\_and\_game\_theory/blob/main/articles/ReinforcementLearning/High-Dimensional\_Continuous\_Control\_Using\_Generalized\_Advantage\_Estimation\_Schulman\_2018.pdf

John Schulman, F. W. (2017, Aug 28). *Proximal Policy Optimization Algorithms.* Retrieved from arxiv.org: https://github.com/dimitarpg13/reinforcement\_learning\_and\_game\_theory/blob/main/articles/ReinforcementLearning/Proximal\_Policy\_Optimization\_Algorithms\_Shulman\_2017.pdf

John Schulman, S. L. (2017, Apr 20). *Trust Region Policy Optimization.* Retrieved from arxiv.org: https://github.com/dimitarpg13/reinforcement\_learning\_and\_game\_theory/blob/main/articles/ReinforcementLearning/TrustRegionPolicyOptimization\_Schulman\_2015.pdf

OpenAI. (2017, July 20). *openai baselines proximal policy optimization.* Retrieved from openai.com: https://openai.com/research/openai-baselines-ppo

Richard D. Sutton, A. G. (2020, Oct 12). *Reinforcement Learning: An Introduction, Second Edition.* Retrieved from https://github.com/dimitarpg13/reinforcement\_learning\_and\_game\_theory/blob/main/books/ReinforcementLearningSuttonSecondEdition2020.pdf

Sham Kakade, J. L. (2002, July 8). *Approximately Optimal Approximate Reinforcement Learning.* Retrieved from ICML.cc: https://github.com/dimitarpg13/reinforcement\_learning\_and\_game\_theory/blob/main/articles/ReinforcementLearning/Approximate\_Optimal\_Approximate\_Reinforcement\_Learning\_KakadeLangford-icml2002.pdf

Volodymir Mnih, e. a. (2016, June 16). *Asynchronour Methods for Deep Reinforcement Learning.* Retrieved from https://github.com/dimitarpg13/reinforcement\_learning\_and\_game\_theory/blob/main/articles/ReinforcementLearning/Asynchronous\_Methods\_for\_Deep\_Reinforcement\_Learning\_Mnih\_2016.pdf

Volodymyr Mnih, e. a. (2015, Feb 26). *Human-level control through deep reinforcement learning.* Retrieved from Nature.com: https://github.com/dimitarpg13/reinforcement\_learning\_and\_game\_theory/blob/main/articles/ReinforcementLearning/Human-level\_control\_through\_deep\_reinforcement\_learning\_Mnih\_2015.pdf