

Title of Thesis is here

by

Thiruvenkadam Sivaprakasam Radhakrishnan

THESIS

Submitted as partial fulfillment of the requirements
for the degree of Master's in Computer Science
in the Graduate College of the
University of Illinois at Chicago, 2023

Chicago, Illinois

Defense Committee:

Prof. Ian Kash, Chair and Advisor

Prof. Anastasios Sidiropoulos

Prof. Ugo Buy

ACKNOWLEDGMENTS

The thesis has been completed. .. (INSERT YOUR TEXTS)

YOUR INITIAL

TABLE OF CONTENTS

<u>CHAPTER</u>		<u>PAGE</u>
1	INTRODUCTION	1
1.1	Outline	2
2	BACKGROUND	3
2.1	Reinforcement Learning	3
2.1.1	Policy gradient methods	3
2.1.1.1	Reinforce	3
2.1.1.2	Softmax Policy Gradients	3
2.1.2	PPO	3
2.2	Game Theory	4
2.2.1	Problem Representations	4
2.2.2	Solution Concepts	6
2.3	Online Learning	6
2.3.1	FoReL	6
2.3.2	Hedge	6
2.3.3	Gradient Descent	6
3	MAGNETIC MIRROR DESCENT	7
3.1	Mirror Descent	7
3.1.1	MDPO	7
3.2	MMD	7
3.2.1	Connection between Variational Inequalities and QREs	8
3.2.2	MMD Algorithm	9
3.2.3	Behavioral form MMD	11
3.2.4	Equivalence of MMD and MDPO	12
4	MODIFIED UPDATES FOR MIRROR-DESCENT BASED METH- ODS	13
4.1	Neural Replicator Dynamics (NeuRD)	13
4.2	Extragradient updates	14
4.3	Optimism	14
4.3.1	Optimistic Mirror Descent	14
5	EXPERIMENTS	15
5.1	Experimental Domains	15
5.2	Evaluation Methods	16
5.3	Results	16

TABLE OF CONTENTS (Continued)

<u>CHAPTER</u>		<u>PAGE</u>
5.3.1	Tabular Experiments	16
5.3.2	Neural Experiments	18
APPENDICES	19
Appendix A	20
Appendix B	21
CITED LITERATURE	22

LIST OF TABLES

<u>TABLE</u>		<u>PAGE</u>
I	Convergence in Perturbed RPS	17

LIST OF FIGURES

FIGURE

PAGE

LIST OF NOTATIONS

θ Parameters of a function approximator.

SUMMARY

Put your summary of thesis here.

CHAPTER 1

INTRODUCTION

In this work, we study two mirror-descent based reinforcement learning algorithms and propose novel improvements to them.

1.1 Outline

The rest of the thesis is organized as follows. We begin by providing some background and definitions in section 2 that are useful for the understanding of the algorithms and methods described in section 3 and 4. Section 3 introduces Mirror Decsent and expands on Mirror Descent based methods for solving Reinforcement learning problems. Section 4 discusses combining novel improvements on top of these methods and discusses their structure and the expected effects. In Section 5, we dive into some experimental results and discuss the performance of these algorithms in different settings. We then close the thesis with some discussion.

CHAPTER 2

BACKGROUND

We begin by providing some necessary background in Reinforcement learning , Game Theory, and Online Learning to make the reader familiar with the concepts required to follow the ideas discussed in the following sections.

2.1 Reinforcement Learning

2.1.1 Policy gradient methods

- One way of doing RL is to directly manipulate the policy to maximize some reward. - Then use the reward and compute a gradient with respect to policy. - Reward might not be typically differentiable, so some workaround is needed to arrive at a gradient-based update rule.

2.1.1.1 Reinforce

- Widely popular PG algorithm, many instantiations possible including variants with baselines to reduce variance. - Convergence is proven using PG theorem.

2.1.1.2 Softmax Policy Gradients

- Parametrizing policies - One way of projecting parametrized policy to a probability simplex is Softmax

PG with softmax parameterization is referred to as Softmax Policy gradients.

2.1.2 PPO

- Trust region methods - PPO an approximation of TRPO with heuristic objective

2.2 Game Theory

Game theory is the mathematical study of interaction between agents to produce outcomes while trying to uphold certain individual or group preferences. It has a wide range of applications including economics, biology, and computer science.

In this work, we mainly focus on a branch of game theory called non-cooperative game theory in which each agent has their own individual preference.

2.2.1 Problem Representations

In discussing agent interactions, and preferences, we need a formal notion of how agents act, and how agent preferences can be defined. In game theory, agent preferences are formalized using Utility theory, where each agent has a utility function that maps the agents preferences over outcomes to a real value.

The primary way of modeling problems in game theory is through a *game* that encodes information about the agents, possible actions agents can take in different situations, their preferences, and the outcome of an interaction. There are many types of such representations, a few relevant of which we introduce below. Before we introduce such representations, we first cover some preliminary concepts that will be helpful in formally defining those representations and agent preferences.

- what are utilities, and strategies?

Normal-Form Games

Normal-Form games are a popular way of representing situations in which all agents act simultaneously and the outcome is revealed after each agent has taken their action. A few

popular games that can be represented in this form are rock-paper-scissors, matching pennies, prisoner's dilemma etc. A more formal definition of a normal-form game is as follows:

Definition 1 (Normal-form games) *A (N, A, u) tuple is a n -player normal form game, where N is the set of players, $A = A_1 \times A_2 \dots \times A_n$, with A_i being the set of actions available to player i , and $u = (u_i \forall i \in N)$ is the set of utility functions that map an action profile to a real utility value for each agent, $u_i : A \mapsto \mathbb{R}$.*

Normal-form games are typically represented using a n -dimensional tensor, where each dimension represents the possible actions available to each agent, and every entry represents an outcome. The actual entries of the

Sequential Games

Although normal-form games provide a neat representation, many real-world scenarios necessitate agents act sequentially which is difficult to represent as a matrix. These problems require a tree-like representation where each node is an agent's turn to make a choice, and each edge is a possible action. There are a few ways to represent such scenarios, one being normal-form games themselves. A downside is that the size of the normal-form representation for sequential games explode exponentially in the size of the game tree. Other possible representations include the Extensive-form, and Sequence-form.

Definition 2 (Extensive-form games)

Definition 3 (Sequence form games)

2.2.2 Solution Concepts

Now that we have - what are solution concepts? - what are the common solution concepts?
 Nash equilibrium, Quantal response equilibrium. - what are the relevant information related
 to solution concepts for this work? Existence of a nash equilibrium Uniqueness of QRE

2.3 Online Learning

- what is online learning?

Online learning is the study of designing algorithms that use historical knowledge in predicting actions for future rounds while trying to minimize some loss function in an adaptive (possibly adversarial) setting.

- why is it useful? - why is it relevant here?

2.3.1 FoReL

- what is forel? - relevant info?

2.3.2 Hedge

- what is hedge?

2.3.3 Gradient Descent

Gradient descent as a FTRL variant

CHAPTER 3

MAGNETIC MIRROR DESCENT

3.1 Mirror Descent

In this section we discuss about Mirror Descent, and two mirror descent-based reinforcement learning algorithms (MDPO, and MMD).

A disadvantage of FoReL 2.3.1 in solving online learning problems is that, there is a minimization at every step. Mirror descent overcomes this by using a recursive update rule that does not required us to perform a minimization at every step.

There are different views of arriving at Mirror Descent as an optimization algorithm, some of which include the FTRL view, the mirror map view, ...

- Mirror descent algorithm - a Mirror descent convergence guarantee. - Mirror descent applied to game theory and marl.

3.1.1 Mirror Descent Policy Optimization

3.2 Magnetic Mirror Descent

Magnetic Mirror Descent [1] is a non-euclidean proximal gradient method that can more generally be applied to solve Variational Inequality problems with certain structures. Sokota et al. [1] cast solving for QREs in two-player zero-sum games as a variational inequality problem and apply MMD to solve for QREs with linear last-iterate convergence guarantees.

3.2.1 Connection between Variational Inequalities and QREs

Variational inequalities are a general class of problems that have a wide range of applications. A Variational Inequality problem is generally of the following form (for convenience, we use the same notation and symbols as in [1]):

Definition 4 *Given $\mathcal{Z} \subseteq \mathbb{R}^n$ and mapping $G : \mathcal{Z} \rightarrow \mathbb{R}^n$, the variational inequality problem VI (\mathcal{Z}, G) is to find $z_* \in \mathcal{Z}$ such that,*

$$\langle G(z_*), z - z_* \rangle \geq 0 \quad \forall z \in \mathcal{Z}.$$

Solving for QREs in two-player zero-sum games can be represented as the following bilinear saddle point problem:

$$\min_{x \in \mathcal{X}} \max_{y \in \mathcal{Y}} \alpha g_1(x) + f(x, y) + \alpha g_2(y), \quad (3.1)$$

where $\mathcal{X} \subset \mathbb{R}^n$, $\mathcal{Y} \subset \mathbb{R}^m$ are closed and convex, and $g_1 : \mathbb{R}^n \rightarrow \mathbb{R}$, $g_2 : \mathbb{R}^m \rightarrow \mathbb{R}$, $f : \mathbb{R}^n \times \mathbb{R}^m \rightarrow \mathbb{R}$.

The solution (x_*, y_*) to Equation 3.1, is a Nash equilibrium in the regularized game with the following best response conditions and optimality conditions:

$$\langle \nabla g_1(x_*) + \nabla_{x_*} f(x_*, y_*), x - x_* \rangle \geq 0, \forall x \in \mathcal{X}; \langle \nabla g_2(y_*) + \nabla_{y_*} f(x_*, y_*), y - y_* \rangle \geq 0, \forall y \in \mathcal{Y}. \quad (3.2)$$

$$x_* \in \arg \min_{x \in \mathcal{X}} \alpha g_1(x) + f(x, y_*) \quad (3.3)$$

$$y_* \in \arg \min_{y \in \mathcal{Y}} \alpha g_2(y) + f(x_*, y) \quad (3.4)$$

The optimality conditions are equivalent to VI(\mathcal{Z}, G), where $G = F + \alpha \nabla g$, $\mathcal{Z} = \mathcal{X} \times \mathcal{Y}$, and $g : \mathcal{Z} \rightarrow \mathbb{R}$.

Now, the solution the the VI problem ($z_* = (x_*, y_*)$), corresponds to the solution of the saddle point problem stated in Equation 3.1, and satisfies the best response conditions Equation 3.3 and Equation 3.4.

3.2.2 MMD Algorithm

The algorithm that the authors propose is a non-Euclidean proximal gradient method to solve the VI ($\mathcal{Z}, F + \alpha \nabla g$) problem with the following update at each iteration:

$$z_{t+1} = \arg \min_{z \in \mathcal{Z}} \eta(\langle F(z_t), z \rangle + \alpha g(z)) + B_\psi(z; z_t). \quad (3.5)$$

The MMD algorithm uses the same update as Equation 3.5, with g taken to be ψ or $B_\psi(\cdot; z')$ for some z' . We now restate the main algorithm as stated in Sokota et. al, [1],

With the following assumptions, z_{t+1} is well defined:

- ψ is 1-strongly convex with respect to $\|\cdot\|$ over \mathcal{Z} , and for any l , stepsize $\eta > 0$, $\alpha > 0$,

$$z_{t+1} = \arg \min_{z \in \mathcal{Z}} \eta(\langle l, z \rangle + \alpha g(z)) + B_\psi(z; z_t) \in \text{int dom } \psi.$$

AlgorithmMMD [1, (Algorithm 3.6)] Starting with $z_1 \in \text{int dom } \psi \cap \mathcal{Z}$, at each iteration

t do

$$z_{t+1} = \arg \min_{z \in \mathcal{Z}} \eta(\langle F(z_t), z \rangle + \alpha \psi(z)) + B_\psi(z; z_t).$$

or, Given some z' , do

$$z_{t+1} = \arg \min_{z \in \mathcal{Z}} \eta(\langle F(z_t), z \rangle + \alpha B_\psi(z; z')) + B_\psi(z, z_t).$$

- F is monotone and L -smooth with respect to $\|\cdot\|$ and g is 1-strongly convex relative to ψ over \mathcal{Z} with g differentiable over $\text{int dom } \psi$.

Algorithm 1 provides the following convergence guarantees.

Theorem 1 [1, Theorem 3.4] *Assuming that the solution z_* to the problem $VI(\mathcal{Z}, F + \alpha \nabla g)$ lies in the int dom ψ , then*

$$B_\psi(z_*; z_{t+1}) \leq \left(\frac{1}{1 + \eta\alpha} \right)^t B_\psi(z_*; z_1),$$

if $\alpha > 0$, and $\eta \leq \frac{\alpha}{L^2}$.

3.2.3 Behavioral form MMD

While Algorithm 1 gives a concrete update rule, it involves solving an optimization at each iteration. This can be done either in closed form for a given set of parameters or by performing stochastic gradient descent at each step to approximate the closed form. This approximation is especially useful when it is not possible to compute the closed form such as in function approximation cases.

In the two form of updates from Algorithm 1, the second form involves a magnet policy z' , that is used to guide the learning at each iteration. This can either be a reference policy such as the ones used in Imitation learning, or one that is trailing the current policy to stabilize learning. Alternatively, one can also use a uniform policy in which case the term reduces to entropy and acts as a regularization term and encourages exploration. For the rest of our discussion on MMD, we assume the magnet to always be a uniform policy.

With ψ taken to be negative entropy, the behavioral form of MMD is to perform the following update at each information state,

$$\pi_{t+1} = \arg \max_{\pi} \mathbb{E}_{A \sim \pi} q_t(A) + \alpha H(\pi) - \frac{1}{\eta} KL(\pi, \pi_t), \quad (3.6)$$

where π_t is the current policy, q_t is a vector containing the q-values following the policy π_t , and $H(\pi)$ is the entropy of the policy being optimized.

Here is a sketch of an algorithm for on-policy MMD using the behavioral form update in RL terms similar to

AlgorithmMMD-BF

Gradient of MMD

Remarks: In single-agent settings MMD’s performance is competitive with PPO in Atari and MuJoCo environments. And, in the multi-agent setting the performance of tabular MMD is on par with CFR, but worse than CFR+.

3.2.4 Equivalence of MMD and MDPO

- MDPO with an added entropy term is equivalent to MMD with negative entropy mirror map and uniform magnet.

CHAPTER 4

MODIFIED UPDATES FOR MIRROR-DESCENT BASED METHODS

We propose applying a few modifications to the methods discussed in the previous section and investigate the effect of these modifications in terms of their performance and convergence behaviors in two-player zero-sum games. We first introduce the proposed modifications to be applied on the methods discussed in the previous section, and then provide a description of how these modifications fit into those algorithms.

4.1 Neural Replicator Dynamics (NeuRD)

Neural Replicator Dynamics (NeuRD) [2] is a model-free sample-based algorithm that is an application of Replicator Dynamics to Policy Gradients. Replicator Dynamics is an idea from Evolutionary game theory (EGT) that defines operators to update the dynamics of a population in order to maximize some pay-off defined by a fitness function.

The single-population replicator dynamics is defined by the following system of differential equations:

$$\dot{\pi}(a) = \pi(a)[u(a, \pi) - \bar{u}(\pi)], \forall a \in \mathcal{A} \quad (4.1)$$

Hennes et.al, [2] show equivalence between Softmax Policy gradients 2.1.1.2 and continuous-time Replicator Dynamics [2, THEOREM 1, on p5].

NeuRD can be implemented as a single line change to SPG. This can be seen as applying a fix to the update of the SPG algorithm to make it more responsive to changes in a non-stationary environment. We refer to this as the “NeuRD-fix”. Since the typical neural network representation of policies use softmax projection on the logits, NeuRD-fix is a more general idea that is applicable to algorithms beyond just SPG. It has been adapted into other algorithms to improve performance or induce convergence in competitive and cooperative settings. Chhablani et.al, [3] showed improved performance in identical-interest games by applying the NeuRD fix to COMA [4]. Perolat et.al, [5] used a NeuRD based loss function along with adaptive regularization [6] to induce last-iterate convergence in the game of Stratego.

4.2 Extragradient updates

The Extragradient method was first introduced by G.M.Korpelevich [7] as a modification of gradient descent methods in solving saddle point problems. Extragradient is a classical method for solving smooth and strongly convex-concave bilinear saddle point problems with a linear rate of convergence. Extragradient and Optimistic Gradient Descent Ascent methods have been shown to be approximations of proximal-point method for solving saddle point methods [8]. In this work, we extend MMD, and FMMD with extragradient updates to investigate the effect of extragredients in these settings in terms of inducing convergence and also speeding up the rate of convergence.

4.3 Optimism

Optimism is another proximal

4.3.1 Optimistic Mirror Descent

CHAPTER 5

EXPERIMENTS

We now evaluate our proposed methods empirically in various settings. Through the experiments, we aim to answer the following questions:

- How does the addition of the NeuRD-fix, Extragradient updates, Optimism affect the convergence rate of these algorithms?
- What is the last-iterate vs average-iterate convergence behavior of these algorithms in the presence of these modifications?
- Do these performance improvements scale well with the size of the game?

5.1 Experimental Domains

In answering the above questions, we consider both tabular and function approximation settings. For the tabular experiments we evaluate the methods using the following environments - Perturbed RPS.

For the function approximation setting, we evaluate the algorithms on Kuhn Poker, Abrupt Dark Hex, and Phantom Tic-tac-toe. Kuhn Poker is a smaller extensive form game that allows for more introspection and accurate measurement of performance. Whereas, Abrupt Dark Hex, and Phantom TTT are large games that test the stability of these modifications in settings with a large state-space.

5.2 Evaluation Methods

Typically, to measure the convergence rate of an algorithm we can use either a a measure of distance from an equilibrium if the equilibrium point is known or, a measure of exploitability as an indication of the policies reaching the equilibrium.

For tabular settings since the equilibrium solutions are known, we use both exact exploitability and KL-divergence to the equilibrium to evaluate the algorithms. For PerturbedRPS, we use the QRE solutions computed through Gambit and, we also derive the unique Nash Equilibrium (please refer to ?? in the appendix for the derivation) for PerturbedRPS.

-Define exact exploitability

For the function approximation settings we use the above metrics for the smaller games Kuhn Poker, and 2x2 Dark Hex which also have equilibrium solutions computed through Gambit. For the larger games, since exact exploitability computation is not possible and we do not know the equilibrium solution, we use an approximate measure of exploitability by using a DQN to approximate the best response computation similar to [1].

5.3 Results

5.3.1 Tabular Experiments

Below, we give an overview of Last, and Average iterate convergences of 3 different algorithms - MMD, MDPO, SPG, and their variants obtained by applying combinations of the modified updates. We give the convergence results for the more interesting behaviors observed, a more exhaustive list of all combinations and their convergence behaviors can be found in the appendix.

Algorithm	Nash		QRE ($\alpha=0.5$)	
	Avg	Last	Avg	Last
MMD (cf)	y	y	y	y
FMMD (bf)	y	y	y	y
FMMD-EG	y	y	y	y
FMMD-N	y	y	y	y
FMMD-N-EG	y	y	y	y
FMMD-N-OPT	y	y	y	y
MDPO-EG	y	x	x	x
MDPO-EG-OPT	y	x	x	x
MDPO-N-EG	y	y	x	x
MDPO-N-EG-OPT	y	y	x	x

TABLE I

Convergence in Perturbed RPS

Apart from the final convergence results, we also make a few observations regarding the speed of convergence for these variants. From ??, we note the following

- MDPO with extragradient updates have superlinear last iterate convergence to Nash equilibrium.
- MMD with NeuRD fix and extragradient

5.3.2 Neural Experiments

Based on the observations from the Tabular NFG experiments, we evaluate the most promising combinations of these algorithms in the function approximation setting. As discussed in [1], due to its similarity MMD can be implemented with minor changes to PPO.

- Implementation details (RLlib, PPO modifications, GAE) - Neural network architecture, hyperparameters

APPENDICES

Appendix A

SOME ANCILLARY STUFF

Ancillary material should be put in appendices.

Appendix B

SOME MORE ANCILLARY STUFF

[?]

CITED LITERATURE

1. Sokota, S., D’Orazio, R., Kolter, J. Z., Loizou, N., Lanctot, M., Mitliagkas, I., Brown, N., and Kroer, C.: A Unified Approach to Reinforcement Learning, Quantal Response Equilibria, and Two-Player Zero-Sum Games. In *The Eleventh International Conference on Learning Representations* , February 2023.
2. Hennes, D., Morrill, D., Omidshafiei, S., Munos, R., Perolat, J., Lanctot, M., Gruslys, A., Lespiau, J.-B., Parmas, P., Duenez-Guzman, E., and Tuyls, K.: Neural Replicator Dynamics, February 2020.
3. Chhablani, C. and Kash, I. A.: Counterfactual Multiagent Policy Gradients and Regret Minimization in Cooperative Settings. In *AAAI* , 2021.
4. Foerster, J., Farquhar, G., Afouras, T., Nardelli, N., and Whiteson, S.: Counterfactual Multi-Agent Policy Gradients. *Proceedings of the AAAI Conference on Artificial Intelligence* , 32(1), April 2018.
5. Perolat, J., de Vylder, B., Hennes, D., Tarassov, E., Strub, F., de Boer, V., Muller, P., Connor, J. T., Burch, N., Anthony, T., McAleer, S., Elie, R., Cen, S. H., Wang, Z., Gruslys, A., Malysheva, A., Khan, M., Ozair, S., Timbers, F., Pohlen, T., Eccles, T., Rowland, M., Lanctot, M., Lespiau, J.-B., Piot, B., Omidshafiei, S., Lockhart, E., Sifre, L., Beauguerlange, N., Munos, R., Silver, D., Singh, S., Hassabis, D., and Tuyls, K.: Mastering the Game of Stratego with Model-Free Multiagent Reinforcement Learning. *Science* , 378(6623):990–996, December 2022.
6. Perolat, J., Munos, R., Lespiau, J.-B., Omidshafiei, S., Rowland, M., Ortega, P., Burch, N., Anthony, T., Balduzzi, D., Vylder, B. D., Piliouras, G., Lanctot, M., and Tuyls, K.: From Poincaré Recurrence to Convergence in Imperfect Information Games: Finding Equilibrium via Regularization. In *Proceedings of the 38th International Conference on Machine Learning* , pages 8525–8535. PMLR, July 2021.
7. Korpelevich, G. M.: The extragradient method for finding saddle points and other problems. 1976.
8. Mokhtari, A., Ozdaglar, A., and Pattathil, S.: A Unified Analysis of Extra-gradient and Optimistic Gradient Methods for Saddle Point Problems: Proximal Point Approach.

In *Proceedings of the Twenty Third International Conference on Artificial Intelligence and Statistics* , pages 1497–1507. PMLR, June 2020.