Methods for Optimization Problems with Markovian Stochasticity and Non-Euclidean Geometry

Vladimir Solodkin[1, 2], **Andrey Veprikov**[1,2, 3], **Alexander Chernyavskiy**[1], **Aleksandr Beznosikov**[1, 2, 4]
[1]Moscow Independent Research Institute of Artificial Intelligence (MIRAI)
[2]Basic Research of Artificial Intelligence Laboratory (BRAIn Lab)
[3]SB AI Lab
[4]Innopolis University
veprikov.as.18@gmail.com

Abstract

This paper examines a variety of classical optimization problems, including well-known minimization tasks and more general variational inequalities. We consider a stochastic formulation of these problems and, unlike most previous work, we take into account the complex Markov nature of the noise. We also consider the geometry of the problem in an arbitrary non-Euclidean setting and propose four methods based on the Mirror Descent iteration technique. The theoretical analysis is provided for smooth and convex minimization problems and variational inequalities with Lipschitz and monotone operators. The convergence guarantees obtained are optimal for first-order stochastic methods, as evidenced by the lower bound estimates provided in this paper. In order to validate the theoretical results, we present the relevant numerical experiments on various reinforcement learning tasks.

Code — https://github.com/brain-lab-research/MLMC_RL

1 Introduction

In the quest to solve complex real-world problems, optimization plays a crucial role in various fields, including artificial intelligence (Creswell et al. 2018; Zhang 2018; Hashem et al. 2024), finance (Cornuejols and Tütüncü 2006), operations research (Rardin and Rardin 1998), and reinforcement learning (Sutton and Barto 2018). While traditional deterministic optimization methods assume that all problem data is known and fixed, practical scenarios often involve uncertainty and variability. Stochastic optimization proves to be a powerful method (Robbins and Monro 1951) for addressing these challenges. This approach incorporates randomness into the optimization process, allowing it to handle the unpredictable nature of application problems more effectively, thus enabling the generation of more robust and reliable solutions. The sources of stochastic behavior can include noise (Huang and Becker 2021), sampling methods (Hedar, Allam, and Fahim 2020), and environmental factors (Gorbunov, Danilova, and Gasnikov 2020). To describe this randomness, one of the simplest and most common assumptions is that noise variables are independent and identically distributed (i.i.d.) (Bach and Moulines 2013; Zhou et al. 2017; Johnson and Zhang 2013; Nazykov et al. 2024). However, in modern applications, there are an increasing number of problems where the noise depends on a certain background. Therefore, the assumption of independence is not always satisfied. For instance, this occurs in reinforcement learning (Bhandari, Russo, and Singal 2018; Srikant and Ying 2019; Durmus et al. 2021) or in distributed optimization (Lopes and Sayed 2007; Dimakis et al. 2010; Even 2023). Thus, we naturally arrive at the statement of the problem in a more general form, where the noise variables are the realizations of an ergodic Markov chain.

In the first paper on Markovian optimization (Duchi et al. 2012), the authors incorporate this type of stochasticity in arbitrary geometry; however, they only consider non-smooth problems. Recently, for the general case of Markovian noise, the finite-time analysis of non-accelerated SGD-type algorithms has been studied in (Sun, Sun, and Yin 2018) for the convex case and in (Doan 2022) for the non-convex and strongly convex settings. Alongside this, (Dorfman and Levy 2022) propose a random batch size algorithm that adapts to the mixing time of the underlying chain for non-convex optimization with a compact domain. In the exploration of accelerated SGD, the paper (Doan et al. 2020a) obtains estimates, and (Beznosikov et al. 2024) improves these results for both non-convex and strongly convex settings. At the same time, numerous papers have appeared that deal with the special scenario of distributed optimization (Even 2023; Doan 2022). Unfortunately, all of these works only consider the Euclidean formulation, while it is interesting to investigate the arbitrary geometry setting as in (Duchi et al. 2012), but in the smooth case, since it is possible to design accelerated methods for it (Nesterov 1983).

The classical algorithm that utilizes non-Euclidean properties is Mirror Descent (MD) (Nemirovsky, Yudin, and Dawson 1983). Various modifications of this method have been extensively studied, including Stochastic MD (Zhou et al. 2017; Jin and Sidford 2020a), Composite MD (Duchi et al. 2010; Lei and Tang 2018), Accelerated MD (Lan 2012; Krichene, Bayen, and Bartlett 2015; Lan, Li, and Zhou 2019), Coordinate MD (Gao et al. 2020; Hanzely and Richtárik 2021), Coupled MD (Allen-Zhu and Orecchia 2014) and even Zero-order MD (Gorbunov, Dvurechensky, and Gasnikov 2022). Recently, the first work combining MD with Markovian noise has appeared (Xiao et al. 2024). However, this paper considers only a token algorithm (finite-sum stochasticity) in the specific setting of decentralized optimization and proposes a

non-accelerated method.

When considering the generalization of optimization problems, it is possible to do this not only from the perspective of randomness or geometry but also from the overall formulation of the problem to be solved. For instance, we can consider a wide class of variational inequalities (VIs). VIs encompass important special cases, including minimization over a convex domain, saddle point/min-max, and fixed point problems with a wide variety of applications in supervised (Joachims 2005; Bach et al. 2012) and unsupervised (Xu et al. 2004; Bach, Mairal, and Ponce 2008) learning, image denoising (Chambolle and Pock 2011), computational game theory (Facchinei and Pang 2003), and GANs (Gidel et al. 2018). As demonstrated in (Goodfellow 2016), the classical Gradient/Mirror Descent algorithms can diverge for the VI problem; therefore, more sophisticated techniques need to be introduced. One of the first algorithms dealing with VI is the Extragradient method (Korpelevich 1976), followed by Dual Extragradient (Nesterov 2003) and Mirror-Prox (MP) (Nemirovski 2004). As shown in (Nemirovski 2004), the Mirror-Prox algorithm attains an optimal convergence rate for monotone variational inequalities with Lipschitz continuous operators. The pioneering work dealing with MP in the stochastic setting was done in (Juditsky, Nemirovskii, and Tauvel 2011). Later, the analysis of stochastic finite-sum variational inequalities or saddle point problems has been extensively studied by many authors, but mostly in the i.i.d. setting (Chavdarova et al. 2019; Alacaoglu and Malitsky 2022; Beznosikov et al. 2023). In the Markovian noise setup, (Wang et al. 2022) considered the finite-sum case in the narrowing of the saddle point problem only. Whereas, (Beznosikov et al. 2024) examines the general formulation of VI, yet again in the Euclidean setup.

Motivated by these research gaps, we arrive at the following results:

- We present new algorithms based on Mirror Descent and Mirror-Prox methods for minimization problems and for variational inequalities, respectively. The analysis is provided in the general setup of arbitrary norms and compatible Bregman divergences, which are uncommon for the Markovian case.
- We provide lower bounds for minimization and VI problems with Markovian noise (Proposition 2.11 and 2.17), allowing us to show the optimality of convergence rates for Algorithms 2 and 3 using batches of size $\widetilde{\mathcal{O}}(1)$ only (Theorems 2.9 and 2.15).
- For the non-batched version of the algorithm that solves the VI problem (Algorithm 4), we show the convergence under the assumption that the variance is bounded only in expectation with respect to the stationary distribution (Theorem E.4), which has not been achieved with Markovian noise before.
- We present a series of numerical experiments conducted on reinforcement learning tasks (see Section 3). The results verify the competitiveness of our approach compared to the baselines.
- As a byproduct of our main results, we provide a novel deviation bound for the mean of realizations of the geometrically ergodic Markov chain (Lemma 2.7) in an arbitrary norm. To the best of our knowledge, this result has previously only been obtained in the Euclidean setup (Paulin 2015).

Notations

We use $\|\cdot\|_*$ to denote the norm conjugate to $\|\cdot\|$, in particular $\|\cdot\|_* := \max_{z:\|z\|\leq 1}\langle\cdot, z\rangle$. We denote $\|\cdot\|_{\text{TV}}$ as the total variation distance. We define $poly(x)$ as a notation of polynomial dependence, that is, $poly(x) := x^k$ for some $k \geq 0$. Let $(\mathsf{Z}, \mathsf{d}_\mathsf{Z})$ be a complete separable metric space endowed with its Borel σ-field $\mathcal{Z}$. We denote by $(\mathsf{Z}^\mathbb{N}, \mathcal{Z}^{\otimes\mathbb{N}})$ the corresponding canonical process. Let $\mathcal{Q}$ be the Markov kernel defined on $\mathsf{Z}\times\mathcal{Z}$, and denote by $\mathbb{P}_\xi$ and $\mathbb{E}_\xi$ the corresponding probability distribution and the expected value with initial distribution ξ. Let $(Z_k)_{k\in\mathbb{N}}$ be the corresponding canonical process. For $\xi = \delta_z$, $z \in \mathsf{Z}$, we simply write $\mathbb{P}_z$ and $\mathbb{E}_z$ instead of $\mathbb{P}_{\delta_z}$ and $\mathbb{E}_{\delta_z}$. For $x^0, \ldots, x^t$ being the iterates of any algorithm, we denote $\mathcal{F}_t = \sigma(x^j, j \leq t)$ and write $\mathbb{E}_t$ to denote $\mathbb{E}[\cdot|\mathcal{F}_t]$.

2 Main Results

Technical Preliminaries

In this section, we study the optimization problem of the form

$$f^* := \min_{x\in\mathcal{X}} \{f(x) := \mathbb{E}_\pi[F(x, Z)]\}, \tag{1}$$

where π is usually an unknown distribution and $\mathcal{X} \subseteq \mathbb{R}^d$ is a normed space with a dual space $\mathcal{X}^*$ and a pair of primal and dual norms $\|\cdot\|$, $\|\cdot\|_*$. Let $\omega(\cdot)$ be a differentiable and 1-strongly convex function with respect to $\|\cdot\|$ on $\mathcal{X}$. Then, for any $x, y \in \mathcal{X}$ we can define the Bregman divergence as $V(x, y) := \omega(x) - \omega(y) - \langle\nabla\omega(y), x - y\rangle$. We assume that for all optimization problems that are considered in this paper, there exists a Bregman divergence V with respect to a norm $\|\cdot\|$ on $\mathcal{X}$. We also assume that $\mathcal{X}$ is compact and $D^2 := \max_{x,y\in\mathcal{X}} V(x, y)$. From the 1-strong convexity of ω, it follows that $\|x - y\|^2 \leq 2D^2$ for all $x, y \in \mathcal{X}$.

For all $\xi \in \mathbb{R}^d$ we also define the *prox mapping* $P_x(\xi)$ as

$$P_x(\xi) := \arg\min_{y\in\mathcal{X}} \{V(x, y) + \langle\xi, y\rangle\}. \tag{2}$$

We now introduce the common assumptions, required for the analysis of solving (1).

Assumption 2.1. The function f is L-smooth on $\mathcal{X}$ with respect to $\|\cdot\|$ norm, i.e., there exists $L > 0$ such that for any $x, y \in \mathcal{X}$ the following inequality holds: $\|\nabla f(x) - \nabla f(y)\|_* \leq L\|x - y\|$.

Assumption 2.2. The function f is convex on $\mathcal{X}$, i.e., it is differentiable and for any $x, y \in \mathcal{X}$ the following inequality holds: $f(y) \leq f(x) - \langle\nabla f(y), x - y\rangle$.

We assume that access to the function f and its gradient is only available through the noisy oracles $F(x, Z)$ and $\nabla_x F(x, Z)$, respectively. Henceforth, the notation $\nabla F(x, Z)$ will be used for brevity. Now, we make assumptions on the noise variables $\{Z_t\}_{t=0}^{+\infty}$.

Assumption 2.3. Let $\{Z_t\}_{t=0}^{\infty}$ be a stationary Markov chain on $(\mathsf{Z}, \mathcal{Z})$ with Markov kernel $\mathcal{Q}$ and a unique invariant distribution π. Let $\{Z_t\}_{t=0}^{\infty}$ be uniformly geometrically ergodic with a mixing time τ_{mix}, i.e. for all $t > 0$ the following holds: $\sup_{z,z' \in \mathsf{Z}} \|\mathcal{Q}^t(z,\cdot) - \mathcal{Q}^t(z',\cdot)\|_{\text{TV}} \lesssim (1/2)^{t/\tau_{\text{mix}}}$.

Assumption 2.3 is a common occurrence in the literature on Markovian noise (Creswell et al. 2018; Doan et al. 2020a; Dorfman and Levy 2022; Beznosikov et al. 2024). This assumption covers finite Markov chains with an irreducible and aperiodic transition matrix (Even 2023). The mixing time τ_{mix} is the number of steps of the Markov chain required for the distribution of the current state to be close to the stationary distribution π. We now provide an assumption on the stochastic gradient.

Assumption 2.4. For all $x \in \mathbb{R}^d$ it holds that $\mathbb{E}_{\pi}[\nabla F(x, Z)] = \nabla f(x)$. Moreover, for all $Z \in \mathsf{Z}$ and $x \in \mathbb{R}^d$ the following inequality holds $\|\nabla F(x, Z) - \nabla f(x)\|_*^2 \leq \sigma^2$.

In Markovian noise problems, we are forced to bound the noise uniformly rather than only in expectation, as is done in the i.i.d. case (Agarwal et al. 2011; Bach and Perchet 2016; Akhavan, Pontil, and Tsybakov 2020; Dvurechensky, Gorbunov, and Gasnikov 2021). This complication arises in many works (Duchi et al. 2012; Sun, Sun, and Yin 2018; Doan et al. 2020a; Doan 2022; Dorfman and Levy 2022; Even 2023; Beznosikov et al. 2024), and the authors have not yet worked out how to avoid this assumption.

Motivating Example from Reinforcement Learning

We now present a motivating example from reinforcement learning (RL) in which Markovian noise has to be taken into account. In this setting, a learning agent interacts with an environment, represented as a finite average-reward Markov decision process (MDP) (Wan, Naik, and Sutton 2021; Jin and Sidford 2020b) $\mathcal{M} = (\mathcal{S}, \mathcal{A}, R, \mathcal{P}, \gamma)$, where $\mathcal{S}$ is a set of states, $\mathcal{A}$ is a set of actions, $R : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ is the reward function, and $\mathcal{P} : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to [0, 1]$ denotes the transition dynamics of the environment. At each discrete time step t, the agent receives a state of the MDP $s_t \in \mathcal{S}$. Based on the state, it selects an action $a_t \in \mathcal{A}$, using a stochastic policy $\mu \in \mathcal{S} \times \Delta(\mathcal{A})$ defined over the set of states $\mathcal{S}$ and the $(|\mathcal{A}| - 1)$-dimensional probability simplex. Then, the agent gets a reward $r_t = R(s_t, a_t)$ from the environment, and the environment proceeds to the next state with a probability, defined by the transition function $\mathcal{P}$.

The goal of the learning process is to maximize average reward via interaction with the environment:

$$\mu^* = \arg \max_{\mu \in \mathcal{S} \times \Delta(\mathcal{A})} J(\mu), \text{ such that} \tag{3}$$

$$J(\mu) := \mathbb{E}_{\mathcal{T}(\mu)} \left[\sum_{t=0}^{\infty} \gamma^t r_t | s_0 \sim \rho, a_0 \sim \mu(\cdot|s_0) \right],$$

where ρ is an initial state distribution for the environment and $\mathbb{E}_{\mathcal{T}(\mu)}[\cdot] := \mathbb{E}_{s_1 \sim \mathcal{P}(\cdot|a_0,s_0)} \mathbb{E}_{a_1 \sim \mu(\cdot|s_1)} \ldots [\cdot]$ describes the environment transitions.

Notably, it is often assumed (Liu et al. 2020; Lee et al. 2021) that the optimal policy μ^* corresponds to the stationary distribution π of the Markov chain, and by construction, its transition kernel $\mathcal{Q}(s'|s) = \sum_{a \in \mathcal{A}} \mathcal{P}(s'|s, a)\mu(a|s)$ falls under Assumptions 2.3 and 2.4 (Dong et al. 2024).

Algorithm 1: MAMD without batching

Parameters: stepsizes $\{\gamma_t\}$, momentums $\{\beta_t\}$ and number of iterations T.
Initialization: choose $x^0 = x_f^0 \in \mathcal{X}$.
for $t = 0, 1, 2, \ldots, T$ **do**
  $x_g^t = \beta_t^{-1} x^t + (1 - \beta_t^{-1}) x_f^t$
  $x^{t+1} = P_{x^t}(\gamma_t \nabla F(x_g^t, Z_t))$
  $x_f^{t+1} = \beta_t^{-1} x^{t+1} + (1 - \beta_t^{-1}) x_f^t$
end for

Many reinforcement learning algorithms that solve the problem (3) utilize a policy gradient technique and update the policy μ iteratively using a stochastic first-order oracle $\nabla_\mu J(\mu^t)$. In order to perform optimization not on the complex set of $\mathcal{S} \times \Delta(\mathcal{A})$ of policies μ, one can use a transition to a dual space. Such a technique is called Mirror Descent policy gradient optimization (Yang et al. 2022; Alfano, Yuan, and Rebeschini 2024). According to the Policy Gradient Theorem (Sutton et al. 1999), the problem (3) fits under Assumptions 2.1 and 2.2 with respect to policies μ (Dong et al. 2024).

In the case of finite $\mathcal{S}$ and $\mathcal{A}$, we can store a table of size $|\mathcal{S}| \times |\mathcal{A}|$ in memory, then the step of the idealized MD algorithm (Nesterov 1983; Sutton et al. 1999) is of the form $\mu_s^{t+1} = P_{\mu_s^t}\left[-\gamma_t Q_s^{\mu^t}\right]$ for all $s \in \mathcal{S}$, where P is defined in (2), γ_t is a stepsize, $\mu_s^t := \mu^t(\cdot|s) \in \Delta(\mathcal{A}), Q_s^{\mu^t} := Q^\mu(s, \cdot) \in \mathbb{R}^d$ and $Q^\mu : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ denotes the associated state-action value function, or Q-function, associated with a policy μ: $Q^\mu(s, a) := \mathbb{E}_{\mathcal{T}(\mu)}\left[\sum_{t=0}^{\infty} r_t | s_0 = s, a_0 = a\right]$. However, in practice, we cannot compute the Q-function in the MD step exactly; therefore it takes the form $\mu_s^{t+1} = P_{\mu_s^t}\left[-\gamma_t \widehat{Q}_s^{\mu^t}\right]$ for all $s \in \mathcal{S}$, where $\widehat{Q}^{\mu^t}(s, a)$ is an estimation of the true Q^{μ^t}-function. Here, the stochasticity of the Mirror Descent algorithm is contained only in the estimation $\widehat{Q}$. In this paper, we provide two methods of such estimation. The first one does not use the batching technique and considers only one rollout from the system (see Algorithm 1), the second method uses the MLMC batching technique (Giles 2015), which takes into account the fact that in RL problems we deal with Markovian noise (see Algorithm 2).

Markovian Accelerated Mirror Descent

To obtain optimal theoretical estimates, we consider accelerated methods to analyze the Markovian noise setting. We present two different techniques for gradient approximation. One technique utilizes a batching construction similar to that used in (Dorfman and Levy 2022; Beznosikov et al. 2024). The other technique does not involve any batching. Firstly, let us introduce a method that uses only one sample of Z at each iteration (Algorithm 1). This algorithm is similar to vanilla Accelerated Mirror Descent (Lan 2012), but with the Markovian noise. The convergence rate of Algorithm 1 is explored in the theorem below.

Theorem 2.5 (Convergence of MAMD without batching (Algorithm 1))**.** *Let Assumptions 2.1, 2.2, 2.3, 2.4 be satisfied. Let the problem* (1) *be solved by Algorithm 1. Assume that the stepsizes γ_t and momentums β_t are chosen such that $0 \leq (\beta_{t+1}-1)\gamma_{t+1} \leq \beta_t\gamma_t$, $\beta_t \geq 2\gamma_t L$ for all $t \geq \tau_{mix}$ and $\beta_{\tau_{mix}} = 1$. Then, for all $T \geq \tau_{mix}$ it holds that*

$$(\beta_T - 1)\gamma_T \mathbb{E}\left[f(x_f^T) - f^*\right] = \widetilde{\mathcal{O}}\left(D^2 + \tau_{mix}^3\sigma^2\sum_{t=\tau}^{T}\gamma_t^2\cdot\right).$$

We can specify the choice of γ_t and β_t to achieve the accelerated convergence of Algorithm 1.

Corollary 2.6 (Parameters tuning for Theorem 2.5)**.** *Under the conditions of Theorem 2.5, choosing β_t and γ_t as $\beta_t = \max\left\{\frac{t-\tau_{mix}}{2}+1; 1\right\}, \gamma_t = \left(\frac{t-\tau_{mix}}{2}+1\right)\cdot \min\left\{\frac{1}{2L}; \frac{D}{(T-\tau_{mix})^{3/2}\sigma\tau_{mix}^{3/2}}\right\}$, in order to achieve the ε-approximate solution in terms of $\mathbb{E}\left[f(x_f^T) - f^*\right] \leq \varepsilon$ it takes*

$$T = \widetilde{\mathcal{O}}\left(\max\left\{\sqrt{\frac{LD^2}{\varepsilon}}\ ;\ \frac{\tau_{mix}^3 D^2\sigma^2}{\varepsilon^2}\right\}\right)$$

iterations (oracle calls) of Algorithm 1.

The complete proofs of Theorem 2.5 and Corollary 2.6 are provided in Appendix B. The result of Theorem 2.5 in i.i.d. ($\tau_{\text{mix}} = 1$) case aligns with the result in (Lan 2012): $T = \mathcal{O}(\max\{\sqrt{(LD^2)/\varepsilon}; D^2\sigma^2/\varepsilon^2\})$. However, in the Markovian noise setup, we inevitably face the τ_{mix}-dependency. It cannot be removed because it appears in the lower bound for the convergence rate of the methods that utilize Markovian properties (see Proposition 2.11). Despite this, Algorithm 1 has a reasonably good polynomial dependence on τ_{mix}. There are several works (Doan et al. 2020b; Doan 2022), whose bounds contain terms that are even exponential in mixing time. A more detailed overview of the upper and lower estimates is presented after Corollary 2.10.

If we use the batching technique as in (Dorfman and Levy 2022; Beznosikov et al. 2024), we can relax the polynomial dependence on τ_{mix} to a linear one. Let us now introduce a modified version of Algorithm 1 that utilizes this technique.

Before analyzing the convergence of Algorithm 2, we provide two important lemmas. The first lemma provides a general result for batching in the context of Markovian noise for arbitrary norms. The second bounds the error between the gradient estimator g^k (line 6) and the true gradient $\nabla f(x_g^t)$.

Lemma 2.7. *Let $\{\xi^t\}_{t=0}^{\infty}$ be an arbitrary Markov chain, such that it satisfies Assumption 2.3 and*

$$\mathbb{E}_\pi[\xi^t] = 0 \ \text{ and } \ \|\xi^t\|_*^2 \leq \sigma^2.$$

Then for any $N \in \mathbb{N}$ it holds that

$$\mathbb{E}\left[\left\|N^{-1}\textstyle\sum_{t=1}^{N}\xi^t\right\|_*^2\right] \lesssim N^{-1}R^2\sigma^2\tau_{mix},$$

where $R^2 := \max_{x\in\mathbb{R}^d:\|x\|=1}\hat{V}(0,x)$ and $\hat{V}$ is an arbitrary Bregman divergence, not necessary V on $\mathcal{X}$.

Algorithm 2: MAMD with batching

1: **Parameters:** stepsizes $\{\gamma_t\}$, momentums $\{\beta_t\}$, number of iterations T, batchsize B, batchsize limit M.
2: **Initialization:** choose $x^0 = x_f^0 \in \mathcal{X}, N_0 = 0$.
3: **for** $t = 0, 1, 2, \ldots, T$ **do**
4: $\quad x_g^t = \beta_t^{-1}x^t + (1-\beta_t^{-1})x_f^t$
5: $\quad$ Sample $J_t \sim \text{Geom}\,(1/2)$
6: $\quad g^t = g_0^t + \begin{cases}2^{J_t}\left(g_{J_t}^t - g_{J_t-1}^t\right), & \text{if } 2^{J_t} \leq M\\ 0, & \text{otherwise}\end{cases}$, with $g_j^t = 2^{-j}B^{-1}\sum_{i=1}^{2^jB}\nabla F(x_g^t, Z_{N_t+i})$
7: $\quad x^{t+1} = P_{x^t}(\gamma_t g^t)$
8: $\quad x_f^{t+1} = \beta_t^{-1}x^{t+1} + (1-\beta_t^{-1})x_f^t$
9: $\quad N_{t+1} = N_t + 2^{J_t}B$
10: **end for**

In this bound, we again obtain terms of the form *poly*(τ_{mix}), which are related to the Markovian nature of the noise variables $\{\xi^t\}_{t=0}^{\infty}$. The term R^2, which is related to the choice of norm, on which the noise variance is bounded, can be removed for certain types of norms, which will be described below. This result is novel in the literature, since it was not proven even for the i.i.d. case. In previous works, the authors only considered the Euclidean norm (Paulin 2015) with an arbitrary ergodic Markov chain.

If we take $\|\cdot\| = \|\cdot\|_p$, $1 \leq p \leq 2$, then the result of Lemma 2.7 is consistent with the classical result for the Euclidean norm (Paulin 2015) (in this case $\hat{V}(x,y) = 1/2\|x-y\|_2^2$, $R^2 = 1/2$, hence $\log(d)$ disappears),d but in the case of $1 \leq p \lesssim \log(d)^{-1}$, the term $\log(d)$ cannot be removed (Ben-Tal and Nemirovski 2001). We regard this as a negligible charge for the generalization of the result to an arbitrary norm and further omit this term in our theoretical estimates.

Lemma 2.8. *Consider Assumptions 2.3 and 2.4. Then for the gradient estimates g^t from line 6 of Algorithm 2, it holds that $\mathbb{E}_t[g^t] = \mathbb{E}_t[g^t_{\lfloor\log M\rfloor}]$. Moreover,*

$$\mathbb{E}_t[\|\nabla f(x_g^t) - g^t\|_*^2] \lesssim B^{-1}\tau_{mix}\log(M)\sigma^2,$$
$$\|\nabla f(x_g^t) - \mathbb{E}_t[g^t]\|_*^2 \lesssim B^{-1}\tau_{mix}M^{-1}\sigma^2.$$

The full proofs of Lemmas 2.7 and 2.8 are provided in Appendix C. Note that Lemma 2.8 follows directly from Lemma 2.7. Moreover, it provides insight about the trade-off between parameters M and B that control the number of calls to the oracle $\nabla F(x, Z)$. The expected number of oracle calls required to compute g^t is $\mathcal{O}(B\log(M))$. Thus, M affects it as $\log(M)$ while the bias term decreases as M^{-1}. Hence, increasing this parameter is not significantly expensive in terms of computational complexity but is very helpful in terms of accuracy. At the same time, the parameter B linearly increases the number of oracle calls. Therefore, we take $B = 1$ in our algorithms.

We are now ready to explore the convergence rate of Algorithm 2.

Theorem 2.9 (Convergence of MAMD with batching (Algorithm 2))**.** *Let Assumptions 2.1, 2.2, 2.3, 2.4 be satisfied with*

Method	Small batches	Arbitrary norm	Acceleration	Oracle complexity
`EMD` (Duchi et al. 2012)	✓	✓	✗	$\tilde{\mathcal{O}}\big(\tau_{\text{mix}} G^2 D^2/\varepsilon^4\big)$
`MARCHON` (Zhao 2023)	✓	✗	✗	$\widetilde{\mathcal{O}}\big(\max\{\tau_{\text{mix}}^2 L^2 D^2/\varepsilon\ ;\ \tau_{\text{mix}} D^2 G^2/\varepsilon^2\}\big)$
`MC SGD` (Sun, Sun, and Yin 2018)	✓	✗	✗	$\tilde{\mathcal{O}}\big(h(G,L)(\tau_{\text{mix}}/\varepsilon^2)^{1/(1-q)}\big)$
`MC SGD` (Doan 2022)	✓	✗	✗	$\widetilde{\mathcal{O}}\big(e^{\tau_{\text{mix}}} L^2 D^2/\varepsilon^2\big)$
`ASGD`(Doan et al. 2020a)	✓	✗	✓	$\widetilde{\mathcal{O}}\big(\max\{\sqrt{L} D^3/\varepsilon\ ;\ \tau_{\text{mix}}^2 D^2 G^2/\varepsilon^2\}\big)$
`MAG` (Dorfman and Levy 2022)	✗	✗	✗	$\tilde{\mathcal{O}}\big(\tau_{\text{mix}} L^2 G^2/\varepsilon^2\big)$
`MC SGD` (Even 2023)	✓	✗	✗	$\widetilde{\mathcal{O}}\big(\tau_{\text{mix}} \max\{LD^2/\varepsilon\ ;\ D^2\sigma^2/\varepsilon^2\}\big)$
`RASGD` (Beznosikov et al. 2024)	✗	✗	✓	$\widetilde{\mathcal{O}}\big(\tau_{\text{mix}} \max\{\sqrt{LD^2/\varepsilon}\ ;\ D^2\sigma^2/\varepsilon^2\}\big)$
`MMD (Algorithm 2)`	✓	✓	✓	$\tilde{\mathcal{O}}\big(\max\{\sqrt{LD^2/\varepsilon}\ ;\ \tau_{\text{mix}} D^2\sigma^2/\varepsilon^2\}\big)$

Table 1: Summary of the results on first-order method with Markovian noise.

$\|\cdot\| = \|\cdot\|_p$, $1 \le p \le 2$. *Let the problem* (1) *be solved by Algorithm 2. Assume that the stepsizes γ_t and momentums β_t are chosen such that $0 \le (\beta_{t+1}-1)\gamma_{t+1} \le \beta_t\gamma_t$, $\beta_t \ge 2\gamma_t L$ for all $t \ge 0$ and $\beta_0 = 1$. Then, for all $T \ge 0$ it holds that*

$$(\beta_T - 1)\gamma_T \mathbb{E}\left[f(x_f^T) - f^*\right] = \\ = \widetilde{\mathcal{O}}\Big(D^2 + \tau_{mix} B^{-1}\left(TM^{-1} + \log M\right)\sigma^2 \sum_{t=0}^{T-1} \gamma_t^2\Big).$$

Similarly, as outlined in Corollary 2.6, we can specify the choice of all parameters of Algorithm 2.

Corollary 2.10 (Parameters tuning for Theorem 2.9)**.** *Under the conditions of Theorem 2.9, choosing β_t, γ_t, M and B as $\beta_t = \frac{t}{2}+1$, $\gamma_t = \left(\frac{t}{2}+1\right)\min\left\{\frac{1}{2L}\ ;\ \frac{D}{T^{3/2}\sigma\tau_{mix}^{1/2}}\right\}$, $M = T$ and $B = 1$, in order to achieve the ε-approximate solution in terms of $\mathbb{E}\left[f(x_f^T) - f^*\right] \le \varepsilon$ it takes*

$$T = \widetilde{\mathcal{O}}\left(\max\left\{\sqrt{\frac{LD^2}{\varepsilon}}\ ;\ \frac{\tau_{mix} D^2\sigma^2}{\varepsilon^2}\right\}\right)$$

iterations (oracle calls) of Algorithm 2.

The complete proofs of Theorem 2.9 and Corollary 2.10 are provided in Appendix D.

Discussion. In Corollary 2.10, we obtain the result directly in terms of oracle complexity, because using g^t as a gradient estimator requires to make $\mathcal{O}(B\log(M))$ first-order oracle calls at each iteration. And since in Corollary 2.10 we set $M = T$ and $B = 1$, we can conclude that oracle complexity is equal to iteration complexity in terms of $\widetilde{\mathcal{O}}(\cdot)$. The results of Theorem 2.9 and Corollary 2.10 again align with the results from (Lan 2012) in the i.i.d. ($\tau_{\text{mix}} = 1$) case. The usage of Markovian batching with batchsize $B = \widetilde{\mathcal{O}}(1)$ (Algorithm 2) helps us to achieve better performance compared to Algorithm 1 (see Corollary 2.6). In particular, the degree of dependence on τ_{mix} in the stochastic (second) term decreases from cubic to linear. If we consider $B \sim \varepsilon, \tau_{\text{mix}}, \sigma^2, L$, then, according to Theorem 2.9, we can achieve the same convergence rate as in the deterministic case: $T = \widetilde{\mathcal{O}}(\sqrt{(LD^2)/\varepsilon})$. However, the oracle complexity remains the same as in Corollary 2.10. If we do not know τ_{mix}, to compute γ_t from Corollary 2.10, we can consider the step size of the form $\gamma_t := (t/2+1)\cdot\min\left\{1/(2L)\ ;\ D/(T^{3/2}\sigma)\right\}$. This prevents us from explicitly estimating the mixing time; however, in this case, we obtain a different convergence rate: $T = \widetilde{\mathcal{O}}(\max\{\sqrt{LD^2/\varepsilon}\ ;\ \tau_{\text{mix}}^2 D^2\sigma^2/\varepsilon^2\})$. Since for many types of Markovian noise the exact calculation of τ_{mix} is a challenging task (Hsu, Kontorovich, and Szepesvári 2015; Wolfer 2020), it is not always possible to use it when choosing the step γ_t, therefore, one has to slightly sacrifice the convergence rate in the stochastic term.

We now proceed to compare the convergence rate in Theorem 2.9 with the results reported in previous works on Markovian noise. The comparison is made in terms of oracle calls. The results obtained in terms of iteration complexity were subsequently rewritten in terms of oracle complexity and are summarized in Table 1.

Comparison. It is worth noting that most of the works dealing with Markovian stochasticity consider non-accelerated methods (without momentum). However, in both practice and theory, the momentum technique provides significant improvements. In particular, the acceleration technique improves the deterministic (first) convergence term (Nesterov 1983). We now provide a brief commentary on the convergence rates presented in Table 1. In the papers (Duchi et al. 2012; Zhao 2023; Sun, Sun, and Yin 2018; Doan 2022; Dorfman and Levy 2022; Even 2023), the authors do not consider acceleration; therefore, the bounds are not optimal in terms of ε-dependency. Moreover, these results are not optimal in τ_{mix}-dependency, as evidenced by the lower bound we provide below (see Proposition 2.11). The results in (Doan et al. 2020a) have quadratic dependence on mixing time, while our work provides linear one. In the work (Beznosikov et al. 2024), deterministic and stochastic parts are the same as in Corollary 2.10, but the first term is multiplied by τ_{mix}. This occurs because the authors used a batch of size $\widetilde{\mathcal{O}}(\tau_{\text{mix}})$, in-

stead of $\widetilde{\mathcal{O}}(1)$, which is much worse in terms of oracle calls compared to our work. Equally important, all of the aforementioned works, apart from (Duchi et al. 2012), consider only the Euclidean setup, as mentioned in Section 1.

Note that the better results for the deterministic term are as important as those for the stochastic term. Numerous existing studies demonstrate that the deterministic term influences the main convergence to a neighborhood of the optimum, while the stochastic term gives the proximity to the optimum within this neighborhood (Dieuleveut, Durmus, and Bach 2018). We present an ablation study showing that the influence of the deterministic term is significant even for small values of τ_{mix} (see Section J).

Lower bound. Combining two lower bounds from (Nesterov 2013) and from (Duchi et al. 2012), we provide the following result.

Proposition 2.11 (Lower bound for problem (1))**.** *There exists an instance of the optimization problem* (1) *satisfying Assumptions 2.1, 2.2, 2.3, 2.4 with arbitrary* $L > 0, \sigma^2 \geq 0, \tau_{mix} \in \mathbb{N}$*, such that for any stochastic first-order gradient method it takes at least*

$$T = \Omega\left(\max\left\{\sqrt{\frac{LD^2}{\varepsilon}}\,;\,\frac{\tau_{mix}D^2\sigma^2}{\varepsilon^2}\right\}\right)$$

oracle calls in order to achieve $\mathbb{E}\left[f(x^T) - f^*\right] \leq \varepsilon$.

The full proof of Proposition 2.11 is provided in Appendix G. As follows from Proposition 2.11 and Corollary 2.10, Algorithm 2 is optimal in the class of first-order methods with Markovian noise.

Markovian Mirror-Prox

In this section, we are interested in the optimization problem of the following form

$$\text{Find } x^* \in \mathcal{X} \text{ such that } \langle F(x^*), x - x^*\rangle \geq 0 \;\; \forall x \in \mathcal{X}. \quad (4)$$

Here, $F : \mathcal{X} \to \mathcal{Y}$ is an operator, and $\mathcal{X}$ is a compact convex set in Euclidean space $\mathcal{Y}$. We again assume that access to the objective operator $F(x)$ is available only through the noisy oracle $F(x, Z)$. For convex minimization, F is the gradient of the objective function, while for the convex-concave saddle point problem F is composed of a gradient and a negative gradient of the objective with respect to the primal and dual variables.

We first provide several assumptions required for the analysis. Assumptions 2.12 and 2.13 are similar to Assumptions 2.1 and 2.2 for the $\nabla f(x)$ in the minimization problem.

Assumption 2.12. The operator F is L-Lipschitz continuous on $\mathcal{X}$, i.e., for all $x, y \in \mathcal{X}$ the following inequality holds: $\|F(x) - F(y)\|_* \leq L\|x - y\|$.

Assumption 2.13. The operator F is monotone on $\mathcal{X}$, i.e., for all $x, y \in \mathcal{X}$ the following inequality holds: $\langle F(x) - F(y), x - y\rangle \geq 0$.

Now, we proceed to examine the assumption on noise boundedness. In Section E, we introduced Algorithm 2, which uses only one sample of the Markov chain at each iteration. Without employing the batching technique, we obtain convergence results under the assumption that the noise is bounded in expectation with respect to a stationary distribution only. Nevertheless, when proving its convergence rate, we require more stringent analogs of Assumptions 2.12 and 2.13 that for all $Z \in \mathsf{Z}$ the operators $F(x, Z)$ are L-Lipschitz and monotone (see Assumptions E.1, E.2, Theorem E.4 and Corollary E.5). In the case of Lipschitzness and monotonicity of the noise-free (objective) operator only, we bind over to the uniformly bounded noise, similar to Section 2.

Algorithm 3: MMP with batching

Parameters: stepsize $\gamma > 0$, number of iterations T, batchsize B, batchsize limit M.
Initialization: choose $x^0 \in \mathcal{X}, N_0 = 0$
for $t = 0, 1, 2, \ldots, T$ **do**
  $g^{t+1/2} = B^{-1}\sum_{i=1}^{B} F(x^t, Z_{N_t+i})$
  $x^{t+1/2} = P_{x^t}\left(\gamma g^{t+1/2}\right)$
  $N_{t+1/2} = N_t + B$
  Sample $J_t \sim \text{Geom}\,(1/2)$
  $g^t = g_0^t + \begin{cases} 2^{J_t}\left(g_{J_t}^t - g_{J_t-1}^t\right), & \text{if } 2^{J_t} \leq M \\ 0, & \text{otherwise} \end{cases}$, with
  $g_j^t = 2^{-j}B^{-1}\sum\limits_{i=1}^{2^j B} F(x^{t+1/2}, Z_{N_{t+1/2}+i})$
  $x^{t+1} = P_{x^t}\left(\gamma g^t\right)$
  $N_{t+1} = N_{t+1/2} + 2^{J_t}B$
end for

Assumption 2.14. For all $x \in \mathbb{R}^d$, we have $\mathbb{E}_\pi[F(x, Z)] = F(x)$. Moreover, for all $Z \in \mathsf{Z}$ and $x \in \mathbb{R}^d$ the following inequality holds: $\|F(x, Z) - F(x)\|_*^2 \leq \sigma^2$.

Now, drawing on (Nesterov 2003, 2005) and (Juditsky, Nemirovskii, and Tauvel 2011), if F is monotone, the quality of a candidate solution $x \in \mathcal{X}$ can be assessed via the *error (or merit)* function

$$\text{Err}_{\text{VI}}(x) := \max_{u \in \mathcal{X}} \langle F(u), x - u\rangle. \quad (5)$$

Let us now introduce an algorithm that solves the problem (4) and utilizes the batching technique for Markovian noise.

Theorem 2.15 (Convergence of MMP)**.** *Let Assumptions 2.3, 2.12, 2.13, 2.14 be satisfied with* $\|\cdot\| = \|\cdot\|_p$, $1 \leq p \leq 2$. *Let the problem* (4) *be solved by Algorithm 3. Assume that the step size* γ *is chosen such that* $\gamma \leq 1/(2L)$. *Then, for all* $T \geq 0$ *it holds that*

$$\mathbb{E}[\text{Err}_{\text{VI}}(\widehat{x}^T)] = \widetilde{\mathcal{O}}\left(\frac{D^2}{\gamma T} + \frac{\gamma\tau_{mix}}{B}\left(\frac{T}{M} + \log(M)\right)\sigma^2\right).$$

Corollary 2.16 (Parameters tuning for Theorem 2.15)**.** *Under the conditions of Theorem 2.15, choosing* γ, M *and* B *as* $\gamma = \min\left\{\frac{1}{2L}\,;\,\frac{D}{T^{1/2}\sigma\tau_{mix}^{1/2}}\right\}$, $M = T$ *and* $B = 1$, *in order to achieve the* ε*-approximate solution in terms of* $\mathbb{E}[\text{Err}_{\text{VI}}(\widehat{x}^T)] \leq \varepsilon$, *it takes*

$$T = \widetilde{\mathcal{O}}\left(\max\left\{\frac{LD^2}{\varepsilon}\,;\,\frac{\tau_{mix}D^2\sigma^2}{\varepsilon^2}\right\}\right)$$

iterations (oracle calls) of Algorithm 3.

The complete proofs of Theorem 2.15 and Corollary 2.16 are provided in Appendix F.

Discussion. In Corollary 2.16, we obtain the result in terms of oracle complexity, since the usage g^t requires $\mathcal{O}(B\log(M))$ oracle calls at each iteration. The results of Theorem 2.15 and Corollary 2.16 align with the results of (Juditsky, Nemirovskii, and Tauvel 2011) in the i.i.d. ($\tau_{\text{mix}} = 1$) case.

Let us now compare the convergence rate for Theorem 2.15 with the previous works on Markovian noise in the variational inequality problem (4). We again compare with existing methods based on the oracle complexity criterion.

Comparison. To the best of our knowledge, there exist only two works on the topic of VI with Markovian stochasticity. In the first paper (Wang et al. 2022), the authors consider only saddle point problems and provide a result of the form $T = \widetilde{\mathcal{O}}\left((G^2 + \tau_{\text{mix}}^2 G^4)/\varepsilon^2\right)$, where G is the uniform bound of the stochastic operator. This estimate is much worse than the one observed in Corollary 2.16. The deterministic term has a ε^{-2} dependence, while the stochastic term not only contains G^2 rather than σ^2, but is also multiplied by τ_{mix}^2. The second work (Beznosikov et al. 2024) provides a guarantee of the form $T = \widetilde{\mathcal{O}}\left(\tau_{\text{mix}} \max\left\{LD^2/\varepsilon\,;\; D^2\sigma^2/\varepsilon^2\right\}\right)$. This result is almost the same as in Corollary 2.16, but both terms are multiplied by τ_{mix}, because the authors again used the batch of size $\widetilde{\mathcal{O}}(\tau_{\text{mix}})$. And as previously mentioned, both papers consider only the Euclidean setup.

Lower bound. Combining lower bounds from (Ouyang and Xu 2021) and (Duchi et al. 2012), we provide the following result.

Proposition 2.17 (Lower bound for (4))**.** *There exists an instance of the optimization problem* (4) *satisfying Assumptions 2.3, 2.12, 2.13, 2.14 with arbitrary $L > 0, \sigma^2 \geq 0, \tau_{mix} \in \mathbb{N}$, such that for any stochastic first-order gradient method it takes at least*

$$T = \Omega\left(\max\left\{\frac{LD^2}{\varepsilon}\,;\; \frac{\tau_{mix} D^2\sigma^2}{\varepsilon^2}\right\}\right)$$

oracle calls in order to achieve $\mathbb{E}\left[\text{Err}_{\text{VI}}(x^T)\right] \leq \varepsilon$.

The full proof of Proposition 2.17 is provided in Appendix G. As follows from Proposition 2.17 and Corollary 2.16, Algorithm 3 is optimal in the class of convex VI problems (4) with Markovian noise.

3 Experiments

To investigate our approaches in practice, we consider the reinforcement learning task (3) from Section 2.

Description of the environments. We use `navix` (Pignatelli et al. 2024) with navigation setups in grid-like environments. We consider 4 tasks with 6×6 grids: 1) `Empty`: navigation from down left to upper right corner, 2) `LavaGap`: navigation without falling into lava, 3) `Dynamic-Obstacles`: navigation between dynamically moving obstacles, 4) `GoToDoor`: navigation from down left to a door (not fixed goal location). In particular, we conduct experiments, utilising a set of challenging maps from

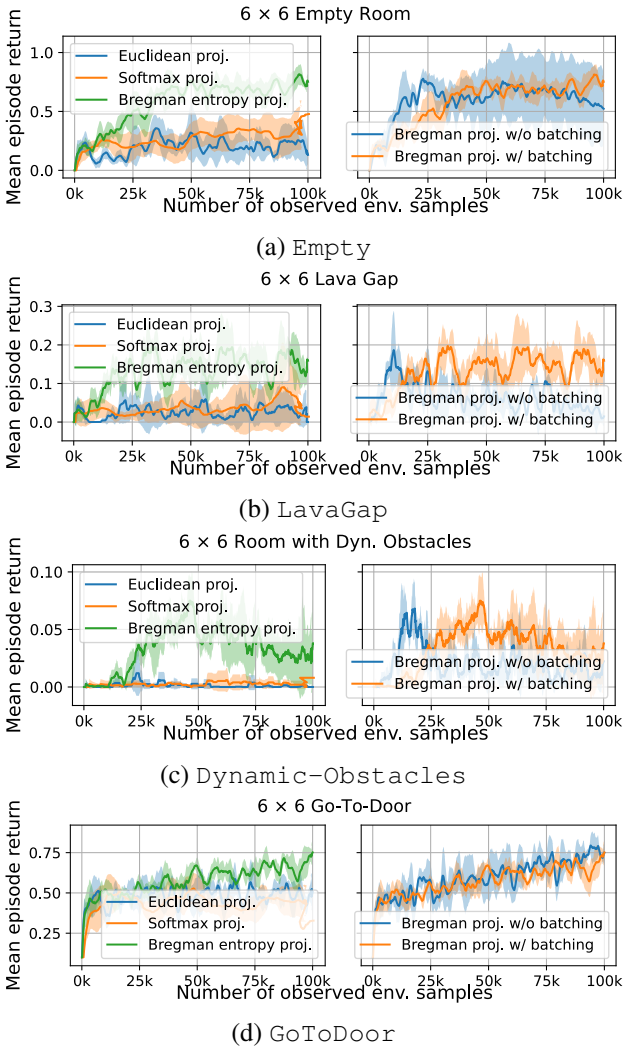

Figure 1: Convergence results on different RL environments with different types of projection **(left)**, with/without MLMC batching **(right)**.

grid-based navigation suite (Pignatelli et al. 2024) to investigate, how the method behaves in case of non-linear mixing time dependence caused by necessity of the environment exploration. It is also worth noting that, in contrast to similar environments, `navix` uses the Markovian reward function (the agent is rewarded only upon completing the goal). This is the exact setting that was investigated in the theoretical part of the paper. Due to space restrictions, we describe more details in Appendix H.

Experiment results. In our first experiment, we investigate the effect of taking into account the geometry of the problem. We use three types of projections: Euclidean and softmax as the baselines, and the Bregman projection illustrating our approach. See Figure 1(left). The method with Bregman projection consistently outperforms both the softmax and Euclidean projections across all tested environments. In particular, we observe that the Bregman projection achieves a higher mean

episodic return and demonstrates faster convergence, especially in more challenging tasks such as `Empty`.

In the second set of experiments, we evaluate the impact of the MLMC batching technique compared to the classical no-batching approach. As shown in Figure 1(right), MLMC batching consistently accelerates convergence. The improvement is major in environments with higher stochasticity, such as `LavaGap`, where the MLMC-batched methods achieve stable performance with fewer environment samples.

Reinforcement algorithm setup. The results are reported on average over 5 random seeds, with a 95 % confidence interval. Given that Algorithm 2 has the randomized batch size bound M and the randomized roll-out length parameter B, we set the computational budget not to a number of training iterations, but rather to the total number of observed environment samples during the training process. We set this number to $T = 10^6$ as the number of oracle calls needed to reach the stationary distribution of the MDP.

Additional experiments. We provide several supporting experiments. In Appendix I we present an ablation study on the effect of batching parameters B and M in the RL setup. In Appendix J we investigate the impact of the mixing time parameter in Markovian noise models. In Appendix K we cover VI problems (4) with various projections.

Conclusion. This paper examines stochastic optimization problems with Markovian noise in non-Euclidean settings. We propose Mirror Descent and Mirror-Prox algorithms that achieve optimal convergence rates and provide matching lower bounds for minimization and variational inequality problems. Experiments on reinforcement learning tasks confirm the theoretical findings and demonstrate the efficiency of the proposed methods in environments with correlated noise.

Acknowledgements

The work was supported by the Ministry of Economic Development of the Russian Federation (agreement No. 139-15-2025-013, dated June 20, 2025, IGK 000000C313925P4B0002).

References

Agarwal, A.; Foster, D. P.; Hsu, D. J.; Kakade, S. M.; and Rakhlin, A. 2011. Stochastic convex optimization with bandit feedback. *Advances in Neural Information Processing Systems*, 24.

Akhavan, A.; Pontil, M.; and Tsybakov, A. 2020. Exploiting higher order smoothness in derivative-free optimization and continuous bandits. *Advances in Neural Information Processing Systems*, 33: 9017–9027.

Alacaoglu, A.; and Malitsky, Y. 2022. Stochastic variance reduction for variational inequality methods. In *Conference on Learning Theory*, 778–816. PMLR.

Alfano, C.; Yuan, R.; and Rebeschini, P. 2024. A novel framework for policy mirror descent with general parameterization and linear convergence. *Advances in Neural Information Processing Systems*, 36.

Allen-Zhu, Z.; and Orecchia, L. 2014. Linear coupling: An ultimate unification of gradient and mirror descent. *arXiv preprint arXiv:1407.1537*.

Bach, F.; Jenatton, R.; Mairal, J.; Obozinski, G.; et al. 2012. Optimization with sparsity-inducing penalties. *Foundations and Trends® in Machine Learning*, 4(1): 1–106.

Bach, F.; Mairal, J.; and Ponce, J. 2008. Convex sparse matrix factorizations. *arXiv preprint arXiv:0812.1869*.

Bach, F.; and Moulines, E. 2013. Non-strongly-convex smooth stochastic approximation with convergence rate O (1/n). *Advances in neural information processing systems*, 26.

Bach, F.; and Perchet, V. 2016. Highly-smooth zero-th order online optimization. In *Conference on Learning Theory*, 257–283. PMLR.

Ben-Tal, A.; and Nemirovski, A. 2001. *Lectures on modern convex optimization: analysis, algorithms, and engineering applications*. SIAM.

Beznosikov, A.; Gorbunov, E.; Berard, H.; and Loizou, N. 2023. Stochastic gradient descent-ascent: Unified theory and new efficient methods. In *International Conference on Artificial Intelligence and Statistics*, 172–235. PMLR.

Beznosikov, A.; Samsonov, S.; Sheshukova, M.; Gasnikov, A.; Naumov, A.; and Moulines, E. 2024. First order methods with markovian noise: from acceleration to variational inequalities. *Advances in Neural Information Processing Systems*, 36.

Bhandari, J.; Russo, D.; and Singal, R. 2018. A finite time analysis of temporal difference learning with linear function approximation. In *Conference on learning theory*, 1691–1692. PMLR.

Chambolle, A.; and Pock, T. 2011. A first-order primal-dual algorithm for convex problems with applications to imaging. *Journal of mathematical imaging and vision*, 40: 120–145.

Chavdarova, T.; Gidel, G.; Fleuret, F.; and Lacoste-Julien, S. 2019. Reducing noise in GAN training with variance reduced extragradient. *Advances in Neural Information Processing Systems*, 32.

Cornuejols, G.; and Tütüncü, R. 2006. *Optimization methods in finance*, volume 5. Cambridge University Press.

Creswell, A.; White, T.; Dumoulin, V.; Arulkumaran, K.; Sengupta, B.; and Bharath, A. A. 2018. Generative adversarial networks: An overview. *IEEE signal processing magazine*, 35(1): 53–65.

Dieuleveut, A.; Durmus, A.; and Bach, F. 2018. Bridging the Gap between Constant Step Size Stochastic Gradient Descent and Markov Chains. arXiv:1707.06386.

Dimakis, A. G.; Kar, S.; Moura, J. M.; Rabbat, M. G.; and Scaglione, A. 2010. Gossip algorithms for distributed signal processing. *Proceedings of the IEEE*, 98(11): 1847–1864.

Doan, T. T. 2022. Finite-time analysis of markov gradient descent. *IEEE Transactions on Automatic Control*, 68(4): 2140–2153.

Doan, T. T.; Nguyen, L. M.; Pham, N. H.; and Romberg, J. 2020a. Convergence rates of accelerated markov gradient

descent with applications in reinforcement learning. *arXiv preprint arXiv:2002.02873*.

Doan, T. T.; Nguyen, L. M.; Pham, N. H.; and Romberg, J. 2020b. Finite-Time Analysis of Stochastic Gradient Descent under Markov Randomness. arXiv:2003.10973.

Dong, Y.; Zhang, H.; Wang, G.; Cui, S.; and Hu, X. 2024. Heavy-ball momentum accelerated actor-critic with function approximation. *arXiv preprint arXiv:2408.06945*.

Dorfman, R.; and Levy, K. Y. 2022. Adapting to mixing time in stochastic optimization with markovian data. In *International Conference on Machine Learning*, 5429–5446. PMLR.

Duchi, J. C.; Agarwal, A.; Johansson, M.; and Jordan, M. I. 2012. Ergodic mirror descent. *SIAM Journal on Optimization*, 22(4): 1549–1578.

Duchi, J. C.; Shalev-Shwartz, S.; Singer, Y.; and Tewari, A. 2010. Composite objective mirror descent. In *COLT*, volume 10, 14–26. Citeseer.

Durmus, A.; Moulines, E.; Naumov, A.; Samsonov, S.; and Wai, H.-T. 2021. On the stability of random matrix product with markovian noise: Application to linear stochastic approximation and td learning. In *Conference on Learning Theory*, 1711–1752. PMLR.

Dvurechensky, P.; Gorbunov, E.; and Gasnikov, A. 2021. An accelerated directional derivative method for smooth stochastic convex optimization. *European Journal of Operational Research*, 290(2): 601–621.

Even, M. 2023. Stochastic gradient descent under markovian sampling schemes. In *International Conference on Machine Learning*, 9412–9439. PMLR.

Facchinei, F.; and Pang, J.-S. 2003. *Finite-dimensional variational inequalities and complementarity problems*. Springer.

Gao, T.; Lu, S.; Liu, J.; and Chu, C. 2020. Randomized bregman coordinate descent methods for non-lipschitz optimization. *arXiv preprint arXiv:2001.05202*.

Gidel, G.; Berard, H.; Vignoud, G.; Vincent, P.; and Lacoste-Julien, S. 2018. A variational inequality perspective on generative adversarial networks. *arXiv preprint arXiv:1802.10551*.

Giles, M. B. 2015. Multilevel monte carlo methods. *Acta numerica*, 24: 259–328.

Goodfellow, I. 2016. Nips 2016 tutorial: Generative adversarial networks. *arXiv preprint arXiv:1701.00160*.

Gorbunov, E.; Danilova, M.; and Gasnikov, A. 2020. Stochastic optimization with heavy-tailed noise via accelerated gradient clipping. *Advances in Neural Information Processing Systems*, 33: 15042–15053.

Gorbunov, E.; Dvurechensky, P.; and Gasnikov, A. 2022. An accelerated method for derivative-free smooth stochastic convex optimization. *SIAM Journal on Optimization*, 32(2): 1210–1238.

Hanzely, F.; and Richtárik, P. 2021. Fastest rates for stochastic mirror descent methods. *Computational Optimization and Applications*, 79: 717–766.

Hashem, I. A.; Alaba, F. A.; Jumare, M. H.; Ibrahim, A. O.; and Abulfaraj, A. W. 2024. Adaptive Stochastic Conjugate Gradient Optimization for Backpropagation Neural Networks. *IEEE Access*.

Hedar, A.-R.; Allam, A. A.; and Fahim, A. 2020. Estimation of distribution algorithms with fuzzy sampling for stochastic programming problems. *Applied Sciences*, 10(19): 6937.

Holmstrom, L.; Koistinen, P.; et al. 1992. Using additive noise in back-propagation training. *IEEE transactions on neural networks*, 3(1): 24–38.

Hsu, D. J.; Kontorovich, A.; and Szepesvári, C. 2015. Mixing time estimation in reversible markov chains from a single sample path. *Advances in neural information processing systems*, 28.

Huang, Z.; and Becker, S. 2021. Stochastic gradient Langevin dynamics with variance reduction. In *2021 International Joint Conference on Neural Networks (IJCNN)*, 1–8. IEEE.

Jin, Y.; and Sidford, A. 2020a. Efficiently solving MDPs with stochastic mirror descent. In *International Conference on Machine Learning*, 4890–4900. PMLR.

Jin, Y.; and Sidford, A. 2020b. Efficiently Solving MDPs with Stochastic Mirror Descent. In III, H. D.; and Singh, A., eds., *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, 4890–4900. PMLR.

Joachims, T. 2005. A support vector method for multivariate performance measures. In *Proceedings of the 22nd international conference on Machine learning*, 377–384.

Johnson, R.; and Zhang, T. 2013. Accelerating stochastic gradient descent using predictive variance reduction. *Advances in neural information processing systems*, 26.

Juditsky, A.; Nemirovskii, A. S.; and Tauvel, C. 2011. Solving variational inequalities with Stochastic Mirror-Prox algorithm. arXiv:0809.0815.

Konda, V.; and Tsitsiklis, J. 1999. Actor-critic algorithms. *Advances in neural information processing systems*, 12.

Korpelevich, G. M. 1976. The extragradient method for finding saddle points and other problems. *Matecon*, 12: 747–756.

Krichene, W.; Bayen, A.; and Bartlett, P. L. 2015. Accelerated mirror descent in continuous and discrete time. *Advances in neural information processing systems*, 28.

Lan, G. 2012. An optimal method for stochastic composite optimization. *Mathematical Programming*, 133(1): 365–397.

Lan, G.; Li, Z.; and Zhou, Y. 2019. A unified variance-reduced accelerated gradient method for convex optimization. *Advances in Neural Information Processing Systems*, 32.

Lee, J.; Jeon, W.; Lee, B.; Pineau, J.; and Kim, K.-E. 2021. Optidice: Offline policy optimization via stationary distribution correction estimation. In *International Conference on Machine Learning*, 6120–6130. PMLR.

Lehtonen, J. 2016. The Lambert W function in ecological and evolutionary models. *Methods in Ecology and Evolution*, 7(9): 1110–1118.

Lei, Y.; and Tang, K. 2018. Stochastic composite mirror descent: Optimal bounds with high probabilities. *Advances in Neural Information Processing Systems*, 31.

Liu, Y.; Swaminathan, A.; Agarwal, A.; and Brunskill, E. 2020. Off-policy policy gradient with stationary distribution correction. In *Uncertainty in artificial intelligence*, 1180–1190. PMLR.

Lopes, C. G.; and Sayed, A. H. 2007. Incremental adaptive strategies over distributed networks. *IEEE transactions on signal processing*, 55(8): 4064–4077.

Ma, X.; Tang, X.; Xia, L.; Yang, J.; and Zhao, Q. 2021. Average-reward reinforcement learning with trust region methods. *arXiv preprint arXiv:2106.03442*.

Mania, H.; Pan, X.; Papailiopoulos, D.; Recht, B.; Ramchandran, K.; and Jordan, M. I. 2017. Perturbed iterate analysis for asynchronous stochastic optimization. *SIAM Journal on Optimization*, 27(4): 2202–2229.

Nazykov, R.; Shestakov, A.; Solodkin, V.; Beznosikov, A.; Gidel, G.; and Gasnikov, A. 2024. Stochastic Frank-Wolfe: Unified Analysis and Zoo of Special Cases. In *International Conference on Artificial Intelligence and Statistics*, 4870–4878. PMLR.

Nemirovski, A. 2004. Prox-method with rate of convergence O (1/t) for variational inequalities with Lipschitz continuous monotone operators and smooth convex-concave saddle point problems. *SIAM Journal on Optimization*, 15(1): 229–251.

Nemirovsky, A.; Yudin, D.; and Dawson, E. 1983. Wiley-Interscience Series in Discrete Mathematics.

Nesterov, Y. 1983. A method of solving a convex programming problem with convergence rate O (1/k** 2). *Doklady Akademii Nauk SSSR*, 269(3): 543.

Nesterov, Y. 2003. Dual extrapolation and its applications to solving variational inequalities and related problems. *Mathematical Programming*, 109: 319–344.

Nesterov, Y. 2005. Primal-dual subgradient methods for convex problems. *Mathematical Programming*, 120: 221–259.

Nesterov, Y. 2013. *Introductory lectures on convex optimization: A basic course*, volume 87. Springer Science & Business Media.

Ouyang, Y.; and Xu, Y. 2021. Lower complexity bounds of first-order methods for convex-concave bilinear saddle-point problems. *Mathematical Programming*, 185(1): 1–35.

Patel, B.; Suttle, W. A.; Koppel, A.; Aggarwal, V.; Sadler, B. M.; Manocha, D.; and Bedi, A. S. 2024. Towards global optimality for practical average reward reinforcement learning without mixing time oracles. In *Proceedings of the 41st International Conference on Machine Learning*, 39889–39907.

Paulin, D. 2015. Concentration inequalities for Markov chains by Marton couplings and spectral methods.

Pignatelli, E.; Liesen, J.; Lange, R. T.; Lu, C.; Castro, P. S.; and Toni, L. 2024. NAVIX: Scaling MiniGrid Environments with JAX. *arXiv preprint arXiv:2407.19396*.

Rardin, R. L.; and Rardin, R. L. 1998. *Optimization in operations research*, volume 166. Prentice Hall Upper Saddle River, NJ.

Robbins, H.; and Monro, S. 1951. A stochastic approximation method. *The annals of mathematical statistics*, 400–407.

Schulman, J.; Moritz, P.; Levine, S.; Jordan, M.; and Abbeel, P. 2015. High-dimensional continuous control using generalized advantage estimation. *arXiv preprint arXiv:1506.02438*.

Srikant, R.; and Ying, L. 2019. Finite-time error bounds for linear stochastic approximation andtd learning. In *Conference on Learning Theory*, 2803–2830. PMLR.

Sun, T.; Sun, Y.; and Yin, W. 2018. On markov chain gradient descent. *Advances in neural information processing systems*, 31.

Suttle, W. A.; Bedi, A.; Patel, B.; Sadler, B. M.; Koppel, A.; and Manocha, D. 2023. Beyond exponentially fast mixing in average-reward reinforcement learning via multi-level Monte Carlo actor-critic. In *International Conference on Machine Learning*, 33240–33267. PMLR.

Sutton, R. S.; and Barto, A. G. 2018. *Reinforcement learning: An introduction*. MIT press.

Sutton, R. S.; McAllester, D.; Singh, S.; and Mansour, Y. 1999. Policy gradient methods for reinforcement learning with function approximation. *Advances in neural information processing systems*, 12.

Tomar, M.; Shani, L.; Efroni, Y.; and Ghavamzadeh, M. 2020. Mirror descent policy optimization. *arXiv preprint arXiv:2005.09814*.

Wan, Y.; Naik, A.; and Sutton, R. S. 2021. Learning and planning in average-reward markov decision processes. In *International Conference on Machine Learning*, 10653–10662. PMLR.

Wang, P.; Lei, Y.; Ying, Y.; and Zhou, D.-X. 2022. Stability and generalization for markov chain stochastic gradient methods. *Advances in Neural Information Processing Systems*, 35: 37735–37748.

Wolfer, G. 2020. Mixing time estimation in ergodic markov chains from a single trajectory with contraction methods. In *Algorithmic Learning Theory*, 890–905. PMLR.

Xiao, L.; Zhang, Z.; Huang, K.; Jiang, J.; and Peng, Y. 2024. Noise optimization in artificial neural networks. *IEEE Transactions on Automation Science and Engineering*.

Xu, L.; Neufeld, J.; Larson, B.; and Schuurmans, D. 2004. Maximum margin clustering. *Advances in neural information processing systems*, 17.

Yang, L.; Zhang, Y.; Zheng, G.; Zheng, Q.; Li, P.; Huang, J.; and Pan, G. 2022. Policy optimization with stochastic mirror descent. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, 8823–8831.

Zhang, Z. 2018. Improved adam optimizer for deep neural networks. In *2018 IEEE/ACM 26th international symposium on quality of service (IWQoS)*, 1–2. Ieee.

Zhao, Y. 2023. Markov Chain Mirror Descent On Data Federation. *arXiv preprint arXiv:2309.14775*.

Zhou, Z.; Mertikopoulos, P.; Bambos, N.; Boyd, S.; and Glynn, P. W. 2017. Stochastic mirror descent in variationally coherent optimization problems. *Advances in Neural Information Processing Systems*, 30.