TIADA: A TIME-SCALE ADAPTIVE ALGORITHM FOR NONCONVEX MINIMAX OPTIMIZATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Adaptive gradient methods have shown their ability to adjust the stepsizes on the fly in a parameter-agnostic manner, and empirically achieve faster convergence for solving minimization problems. When it comes to nonconvex minimax optimization, however, current convergence analyses of gradient descent ascent (GDA) combined with adaptive stepsizes require careful tuning of hyper-parameters and the knowledge of problem-dependent parameters. Such a discrepancy arises from the primal-dual nature of minimax problems and the necessity of delicate *time-scale separation* between the primal and dual updates in attaining convergence. In this work, we propose a *single-loop* adaptive GDA algorithm called TiAda for nonconvex minimax optimization that automatically adapts to the time-scale separation. Our algorithm is *fully parameter-agnostic* and can achieve *near-optimal complexities* simultaneously in deterministic and stochastic settings of nonconvex-strongly-concave minimax problems. The effectiveness of the proposed method is further justified numerically for a number of machine learning applications.

1 INTRODUCTION

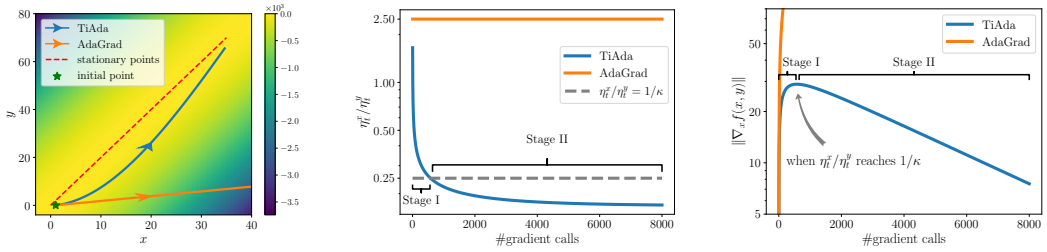
Adaptive gradient methods, such as AdaGrad (Duchi et al., 2011), Adam (Kingma & Ba, 2015) and AMSGrad (Reddi et al., 2018), have become the default choice of optimization algorithms in many machine learning applications owing to their robustness to hyper-parameter selection and fast empirical convergence. These advantages are especially prominent in nonconvex regime with success in training deep neural networks (DNN). Classic analyses of gradient descent for smooth functions require the stepsize to be less than $2/l$, where l is the smoothness parameter and often unknown for complicated models like DNN. Many adaptive schemes, usually with diminishing stepsizes based on cumulative gradient information, can adapt to such parameters and thus reducing the burden of hyper-parameter tuning (Ward et al., 2020; Xie et al., 2020). Such tuning-free algorithms are called *parameter-agnostic*, as they do not require any prior knowledge of problem-specific parameters, e.g., the smoothness or strong-convexity parameter.

In this work, we aim to bring the benefits of adaptive stepsizes to solving the following problem:

$$\min_{x\in\mathbb{R}^{d_1}} \max_{y\in\mathcal{Y}} f(x,y) = \mathbb{E}_{\xi\in P}\left[F(x,y;\xi)\right], \tag{1}$$

where P is an unknown distribution from which we can drawn i.i.d. samples, $\mathcal{Y}\subset\mathbb{R}^{d_2}$ is closed and convex, and $f:\mathbb{R}^{d_1}\times\mathbb{R}^{d_2}\to\mathbb{R}$ is nonconvex in x. We call x the primal variable and y the dual variable. This minimax formulation has found vast applications in modern machine learning, notably generative adversarial networks (Goodfellow et al., 2014; Arjovsky et al., 2017), adversarial learning (Goodfellow et al., 2015; Miller et al., 2020), reinforcement learning (Dai et al., 2017; Modi et al., 2021), sharpness-aware minimization (Foret et al., 2021), domain-adversarial training (Ganin et al., 2016), etc. Albeit theoretically underexplored, adaptive methods are widely deployed in these applications in combination with popular minimax optimization algorithms such as (stochastic) gradient descent ascent (GDA), extragradient (EG) (Korpelevich, 1976), and optimistic GDA (Popov, 1980; Rakhlin & Sridharan, 2013); see, e.g., (Gulrajani et al., 2017; Daskalakis et al., 2018; Mishchenko et al., 2020; Reisizadeh et al., 2020), just to list a few.

While it seems natural to directly extend adaptive stepsizes to minimax optimization algorithms, a recent work by Yang et al. (2022a) pointed out that such schemes may not always converge without knowing problem-dependent parameters. Unlike the case of minimization, convergent analyses of



(a) trajectory (b) effective stepsize ratio (c) convergence

Figure 1: Comparison between TiAda and vanilla GDA with AdaGrad stepsizes (labeled as AdaGrad) on the quadratic function (2) with $L = 2$ under a poor initial stepsize ratio, i.e., $\eta^x/\eta^y = 5$. Here, η_t^x and η_t^y are the effective stepsizes respectively for x and y, and κ is the condition number[1]. (a) shows the trajectory of the two algorithms and the background color demonstrates the function value $f(x, y)$. In (b), while the effective stepsize ratio stays unchanged for AdaGrad, TiAda adapts to the desired *time-scale separation* $1/\kappa$, which divides the training process into two stages. In (c), after entering Stage II, TiAda converges fast, whereas AdaGrad diverges.

GDA and EG for nonconvex minimax optimization are subject to *time-scale separation* (Boţ & Böhm, 2020; Lin et al., 2020a; Sebbouh et al., 2022; Yang et al., 2022b) — the stepsize ratio of primal and dual variables needs to be smaller than a problem-dependent threshold — which is recently shown to be necessary even when the objective is strongly concave in y with true gradients (Li et al., 2022). Moreover, Yang et al. (2022a) showed that GDA with standard adaptive stepsizes, that chooses the stepsize of each variable based only on the (moving) average of its own past gradients, fails to adapt to the time-scale separation requirement. Take the following nonconvex-strongly-concave function as a concrete example:

$$f(x, y) = -\frac{1}{2}y^2 + Lxy - \frac{L^2}{2}x^2, \tag{2}$$

where $L > 0$ is a constant. Yang et al. (2022a) proved that directly using adaptive stepsizes like AdaGrad, Adam and AMSGrad will fail to converge if the ratio of initial stepsizes of x and y (denoted as η^x and η^y) is large. We illustrate this phenomenon in Figures 1(a) and 1(c), where AdaGrad diverges. To sum up, adaptive stepsizes designed for minimization, are not *time-scale adaptive* for minimax optimization and thus not *parameter-agnostic*.

To circumvent this time-scale separation bottleneck, Yang et al. (2022a) introduced an adaptive algorithm, NeAda, for problem (1) with nonconvex-strongly-concave objectives. NeAda is a two-loop algorithm built upon GDmax (Lin et al., 2020a) that after one primal variable update, updates the dual variable for multiple steps until a stopping criterion is satisfied in the inner loop. Although the algorithm is agnostic to the smoothness and strong-concavity parameters, there are several limitations that may undermine its performance in large-scale training: (a) In the stochastic setting, it gradually increases the number of inner loop steps (k steps for the k-th outer loop) to improve the inner maximization problem accuracy, resulting in a possible waste of inner loop updates if the maximization problem is already well solved; (b) NeAda needs a large batchsize of order $\Omega\left(\epsilon^{-2}\right)$ to achieve the near-optimal convergence rate in theory; (c) It is not fully adaptive to the gradient noise, since it deploys different strategies for deterministic and stochastic settings.

In this work, we address all of the issues above by proposing TiAda (**Ti**me-scale **Ada**ptive Algorithm), a single-loop algorithm with time-scale adaptivity for minimax optimization. Specifically, one of our major modifications is setting the effective stepsize, i.e., the scale of (stochastic) gradient used in the updates, of the primal variable to the reciprocal of the *maximum* between the primal and dual variables' second moments, i.e., the sums of their past gradient norms. This ensures the effective stepsize ratio of x and y being upper bounded by a decreasing sequence, which eventually reaches the desired time-scale separation. Taking the test function (2) as an example, Figure 1 illustrates the time-scale adaptivity of TiAda: In Stage I, the stepsize ratio quickly decreases below the threshold; in Stage II, the ratio is stabilized and the gradient norm starts to converge fast.

We focus on the minimax optimization (1) that is strongly-concave in y, since other nonconvex regimes are far less understood even without adaptive stepsizes. Moreover, near stationary point

[1]Please refer to Section 2 for formal definitions of initial stepsize and effective stepsize. Note that the initial stepsize ratio, η^x/η^y, does not necessarily equal to the first effective stepsize ratio, η_0^x/η_0^y.

may not exist in nonconvex-nonconcave (NC-NC) problems and finding first-order local minimax point is already PPAD-complete (Daskalakis et al., 2021). We consider a constraint for the dual variable, which is common in convex optimization with adaptive stepsizes (Levy, 2017; Levy et al., 2018) and in the minimax optimization with non-adaptive stepsizes (Lin et al., 2020a). In summary, our contributions are as follows:

- We introduce the first *single-loop* and *fully parameter-agnostic* adaptive algorithm, TiAda, for nonconvex-strongly-concave (NC-SC) minimax optimization. It adapts to the necessary time-scale separation without large batchsize or any knowledge of problem-dependant parameters or target accuracy. TiAda finds an ϵ-stationary point with an optimal complexity of $\mathcal{O}\left(\epsilon^{-2}\right)$ in the deterministic case, and a near-optimal sample complexity of $\mathcal{O}\left(\epsilon^{-(4+\delta)}\right)$ for any small $\delta > 0$ in the stochastic case. It shaves off the extra logarithmic terms in the complexity of NeAda with AdaGrad stepsize for both primal and dual variables (Yang et al., 2022a). TiAda is proven to be noise-adaptive, which is the first of its kind among nonconvex minimax optimization algorithms.
- While TiAda is based on AdaGrad stepsize, we generalize TiAda with other existing adaptive schemes, and conduct experiments on several tasks. The tasks include 1) test functions by Yang et al. (2022a) for showing the nonconvergence of GDA with adaptive schemes under poor initial stepsize ratios, 2) distributional robustness optimization (Sinha et al., 2018) on MNIST dataset with a NC-SC objective, and 3) training the NC-NC generative adversarial networks on CIFAR-10 dataset. In all tasks, we show that TiAda converges faster and is more robust compared with NeAda or GDA with other existing adaptive stepsizes.

1.1 Related Work

Adaptive gradient methods. AdaGrad brings about an adaptive mechanism for gradient-based optimization algorithm that adjusts its stepsize by keeping the avaraged past gradients. The original AdaGrad was introduced for online convex optimization and maintains coordinate-wise stepsizes. In nonconvex stochastic optimization, AdaGrad-Norm with one learning rate for all directions is shown to achieve the same complexity as SGD (Ward et al., 2020; Li & Orabona, 2019), even with the high probability bound (Kavis et al., 2022; Li & Orabona, 2020). In comparison, RMSProp (Hinton et al., 2012) and Adam (Kingma & Ba, 2015) use the decaying moving average of past gradients, but may suffer from divergence (Reddi et al., 2018). Many variants of Adam are proposed, and a wide family of them, including AMSGrad, are provided with convergence guarantees (Zhou et al., 2018; Chen et al., 2018; Défossez et al., 2020; Zhang et al., 2022b). One of the distinguishing traits of adaptive algorithms is that they can achieve order-optimal rates without knowledge about the problem parameters, such as smoothness and variance of the noise, even in nonconvex optimization (Ward et al., 2020; Levy et al., 2021; Kavis et al., 2019).

Adaptive minimax optimization algorithms. The adaptive stepsize schemes are naturally extended to minimax optimization, both in theory and practice, notably in the training of GANs (Goodfellow, 2016; Gidel et al., 2018). In the convex-concave regime, several adaptive algorithms are designed based on EG and AdaGrad stepsize, and they inherit the parameter-agnostic characteristic (Bach & Levy, 2019; Antonakopoulos et al., 2019). In sharp contrast, when the objective function is nonconvex about one variable, most existing adaptive algorithms require knowledge of the problem parameters (Huang & Huang, 2021; Huang et al., 2021; Guo et al., 2021). Very recently, it was proved that a parameter-dependent ratio between two stepsizes is necessary for GDA in NC-SC minimax problems with non-adaptive stepsize (Li et al., 2022) and most existing adaptive stepsizes (Yang et al., 2022a). Heusel et al. (2017) shows the two-time-scaled GDA with non-adaptive stepsize or Adam will converge, but assuming the existence of an asymptotically stable attractor.

Other NC-SC minimax optimization algorithms. In the NC-SC setting, the most popular algorithms are GDA and GDmax, in which one primal variable update is followed by one or multiple steps of dual variable updates Both of them can achieve $\mathcal{O}(\epsilon^{-2})$ complexity in the deterministic setting and $\mathcal{O}(\epsilon^{-4})$ sample complexity in the stochastic setting (Lin et al., 2020a; Chen et al., 2021; Nouiehed et al., 2019), which are not improvable in the dependency on ϵ given the existing lower complexity bounds (Zhang et al., 2021; Li et al., 2021). Later, several works further improved the dependency on the condition number with more complicated algorithms in deterministic (Yang et al., 2022b; Lin et al., 2020b) and stochastic settings (Zhang et al., 2022a). All of the algorithms above do not use adaptive stepsizes and rely on prior knowledge of the problem parameters.

1.2 NOTATIONS

We denote l as the smoothness parameter, μ as the strong-concavity parameter, whose formal definitions will be introduced in Assumptions 3.1 and 3.2, and $\kappa := l/\mu$ as the condition number. We assume access to stochastic gradient oracle returning $[\nabla_x F(x, y; \xi), \nabla_y F(x, y; \xi)]$. For the minimax problem (1), we denote $y^*(x) := \arg\max_{y \in \mathcal{Y}} f(x, y)$ as the solution of the inner maximization problem, $\Phi(x) := f(x, y^*(x))$ as the primal function, and $\mathcal{P}_{\mathcal{Y}}(\cdot)$ as projection operator onto set $\mathcal{Y}$. For notational simplicity, we will use the name of an existing adaptive algorithm to refer to the simple combination of GDA and it, i.e., setting the stepsize of GDA to that adaptive scheme separately for both x and y. For instance "AdaGrad" for minimax problems stands for the algorithm that uses AdaGrad stepsizes separately for x and y in GDA.

2 METHOD

We formally introduce the TiAda method in Algorithm 1, and the major difference with AdaGrad lies in line 5. Like AdaGrad, TiAda stores the accumulated squared (stochastic) gradient norm of the primal and dual variables in v_t^x and v_t^y, respectively. We refer to hyper-parameters η^x and η^y as the *initial stepsizes*, and the actual stepsizes for updating in line 5 as *effective stepsizes* which are denoted by η_t^x and η_t^y. TiAda adopts effective stepsizes $\eta_t^x = \eta^x / \max\left\{v_{t+1}^x, v_{t+1}^y\right\}^\alpha$ and $\eta_t^y = \eta^y / \left(v_{t+1}^y\right)^\beta$, while AdaGrad uses $\eta^x / \left(v_{t+1}^x\right)^{1/2}$ and $\eta^y / \left(v_{t+1}^y\right)^{1/2}$. In Section 3, our theoretical analysis suggests to choose $\alpha > 1/2 > \beta$. We will also illustrate in the next subsection that the $\max$ structure and different α, β make our algorithm adapt to the desired time-scale separation.

For simplicity of analysis, similar to AdaGrad-Norm (Ward et al., 2020), we use the norms of gradients for updating the effective stepsizes. A more practical coordinate-wise variant that can be used for high-dimensional models is presented in Section 4.1.

Algorithm 1 TiAda (Time-scale Adaptive Algorithm)

1: **Input:** (x_0, y_0), $v_0^x > 0$, $v_0^y > 0$, $\eta^x > 0$, $\eta^y > 0$, $\alpha > 0$, $\beta > 0$ and $\alpha > \beta$.
2: **for** $t = 0, 1, 2, ...$ **do**
3: sample i.i.d. ξ_t^x and ξ_t^y, and let $g_t^x = \nabla_x F(x_t, y_t; \xi_t^x)$ and $g_t^y = \nabla_y F(x_t, y_t; \xi_t^y)$
4: $v_{t+1}^x = v_t^x + \|g_t^x\|^2$ and $v_{t+1}^y = v_t^y + \|g_t^y\|^2$
5: $x_{t+1} = x_t - \frac{\eta^x}{\max\left\{v_{t+1}^x, v_{t+1}^y\right\}^\alpha} g_t^x$ and $y_{t+1} = \mathcal{P}_{\mathcal{Y}}\left(y_t + \frac{\eta^y}{\left(v_{t+1}^y\right)^\beta} g_t^y\right)$
6: **end for**

2.1 THE TIME-SCALE ADAPTIVITY OF TIADA

Current analyses of GDA with non-adaptive stepsizes require the time-scale, η_t^x / η_t^y, to be smaller than a threshold depending on problem constants such as the smoothness and the strong-concavity parameter (Lin et al., 2020a; Yang et al., 2022b). The intuition is that we should not aggressively update x if the inner maximization problem has not yet been solved accurately, i.e., we have not found a good approximation of $y^*(x)$. Therefore, the effective stepsize of x should be small compared with that of y. It is tempting to expect adaptive stepsizes to automatically find a suitable time-scale separation. However, the quadratic example (2) given by Yang et al. (2022a) shattered the illusion. In this example, the effective stepsize ratio stays the same along the run of existing adaptive algorithms, including AdaGrad (see Figure 1(b)), Adam and AMSGrad, and they fail to converge if the initial stepsizes are not carefully chosen (see Yang et al. (2022a) for details). As v_t^x and v_t^y only separately contain the gradients of x and y, the effective stepsizes of two variables in these adaptive methods depend on their own history, which prevents them from cooperating to adjust the ratio.

Now we explain how TiAda adapts to both the required time-scale separation and small enough stepsizes. First, the ratio of our modified effective stepsizes is upper bounded by a decreasing sequence when $\alpha > \beta$:

$$\frac{\eta_t^x}{\eta_t^y} = \frac{\eta^x / \max\left\{v_{t+1}^x, v_{t+1}^y\right\}^\alpha}{\eta^y / \left(v_{t+1}^y\right)^\beta} \leq \frac{\eta^x / \left(v_{t+1}^y\right)^\alpha}{\eta^y / \left(v_{t+1}^y\right)^\beta} = \frac{\eta^x}{\eta^y \left(v_{t+1}^y\right)^{\alpha-\beta}}, \tag{3}$$

as v_t^y is the sum of previous gradient norms and is increasing. Regardless of the initial stepsize ratio η^x/η^y, we expect the effective stepsize ratio to eventually drop below the desirable threshold for convergence. On the other hand, the effective stepsizes for the primal and dual variables are also upper bounded by decreasing sequences, $\eta^x/\left(v_{t+1}^x\right)^\alpha$ and $\eta^y/\left(v_{t+1}^y\right)^\beta$, respectively. Similar to AdaGrad, such adaptive stepsizes will reduce to small enough, e.g., $\mathcal{O}(1/l)$, to ensure convergence.

Another way to look at the effective stepsize of x is

$$\eta_t^x = \frac{\eta^x}{\max\left\{v_{t+1}^x, v_{t+1}^y\right\}^\alpha} = \frac{\left(v_{t+1}^x\right)^\alpha}{\max\left\{v_{t+1}^x, v_{t+1}^y\right\}^\alpha} \cdot \frac{\eta^x}{\left(v_{t+1}^x\right)^\alpha}. \tag{4}$$

If the gradients of y are small (i.e., $v_{t+1}^y < v_{t+1}^x$), meaning the inner maximization problem is well solved, then the first factor becomes 1 and the effective stepsize of x is just the second factor, similar to the AdaGrad updates. If the term v_{t+1}^y dominates over v_{t+1}^x, the first factor would be smaller than 1, allowing to slow down the update of x and waiting for a better approximation of $y^*(x)$.

To demonstrate the time-scale adaptivity of TiAda, we conducted experiments on the quadratic minimax example (2) with $L = 2$. As shown in Figure 1(b), while the effective stepsize ratio of AdaGrad stays unchanged for this particular function, TiAda progressively decreases the ratio. According to Lemma 2.1 of Yang et al. (2022a), $1/\kappa$ is the threshold where GDA starts to converge. We label the time period before reaching this threshold as Stage I, during which as shown in Figure 1(c), the gradient norm for TiAda increases. However, as soon as it enters Stage II, i.e., when the ratio drops below $1/\kappa$, TiAda converges fast to the stationary point. In contrast, since the stepsize ratio of AdaGrad never reaches this threshold, the gradient norm keeps growing.

3 Theoretical Analysis of TiAda

In this section, we study the convergence of TiAda under NC-SC setting with both deterministic and stochastic gradient oracles. We make the following assumptions to develop our convergence results.

Assumption 3.1 (smoothness)**.** *Function $f(x, y)$ is l-smooth ($l > 0$) in both x and y, that is, for any $x_1, x_2 \in \mathbb{R}^{d_1}$ and $y_1, y_2 \in \mathcal{Y}$, we have*

$$\max\{\|\nabla_x f(x_1, y_1) - \nabla_x f(x_2, y_2)\|, \|\nabla_y f(x_1, y_1) - \nabla_y f(x_2, y_2)\|\} \leq l\left(\|x_1 - x_2\| + \|y_1 - y_2\|\right).$$

Assumption 3.2 (strong-concavity in y)**.** *Function $f(x, y)$ is μ-strongly-concave ($\mu > 0$) in y, that is, for any $x \in \mathbb{R}^{d_1}$ and $y_1, y_2 \in \mathcal{Y}$, we have*

$$f(x, y_1) \geq f(x, y_2) + \langle \nabla_y f(x, y_2), y_1 - y_2 \rangle + \frac{\mu}{2}\|y_1 - y_2\|^2.$$

Assumption 3.3 (interior optimal point)**.** *For any $x \in \mathbb{R}^{d_1}$, $y^*(x)$ is in the interior of $\mathcal{Y}$.*

Remark 3.1. The last assumption ensures $\nabla_y f(x, y^*(x)) = 0$, which is important for AdaGrad-like stepsizes that use the sum of squared norms of past gradients in the denominator. If the gradient about y is not 0 at $y^*(x)$, the stepsize will keep decreasing even near the optimal point, leading to slow convergence. This assumption could be potentially alleviated by using generalized AdaGrad stepsizes (Bach & Levy, 2019).

We aim to find a near stationary point for the minimax problem (1). Here, (x, y) is defined to be an ϵ stationary point if $\|\nabla_x f(x, y)\| \leq \epsilon$ and $\|\nabla_y f(x, y)\| \leq \epsilon$ in the deterministic setting, or $\mathbb{E}\|\nabla_x f(x, y)\|^2 \leq \epsilon^2$ and $\mathbb{E}\|\nabla_y f(x, y)\|^2 \leq \epsilon^2$ in the stochastic setting, where the expectation is taken over all the randomness in the algorithm. This stationarity notion can be easily translated to the near-stationarity of the primal function $\Phi(x) = \max_{y \in \mathcal{Y}}(x, y)$ (Yang et al., 2022b). Under our analyses, TiAda is able to achieve the optimal $\mathcal{O}\left(\epsilon^{-2}\right)$ complexity in the deterministic setting and a near-optimal $\mathcal{O}\left(\epsilon^{-(4+\delta)}\right)$ sample complexity for any small $\delta > 0$ in the stochastic setting.

3.1 Deterministic Setting

In this subsection, we assume to have access to the exact gradients of $f(\cdot, \cdot)$, and therefore we can replace $\nabla_x F(x_t, y_t; \xi_t^x)$ and $\nabla_y F(x_t, y_t; \xi_t^y)$ by $\nabla_x f(x_t, y_t)$ and $\nabla_y f(x_t, y_t)$ in Algorithm 1.

Theorem 3.1 (deterministic setting)**.** *Under Assumptions 3.1 to 3.3, Algorithm 1 with deterministic gradient oracles satisfies that for any* $0 < \beta < \alpha < 1$*, after* T *iterations,*

$$\frac{1}{T}\sum_{t=0}^{T-1}\|\nabla_x f(x_t, y_t)\|^2 + \frac{1}{T}\sum_{t=0}^{T-1}\|\nabla_y f(x_t, y_t)\|^2 \leq \mathcal{O}\left(\frac{1}{T}\right).$$

This theorem implies that for any initial stepsizes, TiAda finds an ϵ-stationary point within $\mathcal{O}(\epsilon^{-2})$ iterations. Such complexity is comparable to that of nonadaptive methods, such as vanilla GDA (Lin et al., 2020a), and is optimal in the dependency of ϵ (Zhang et al., 2021). Like NeAda (Yang et al., 2022a), TiAda does not need any prior knowledge about μ and l, but it improves over NeAda by removing the logarithmic term in the complexity. Notably, we provide a unified analysis for a wide range of α and β, while most existing literature on AdaGrad-like stepsizes only validates a specific hyper-parameter, e.g., $\alpha = 1/2$ in minimization problems (Ward et al., 2020; Kavis et al., 2019).

3.2 Stochastic Setting

In this subsection, we assume the access to a stochastic gradient oracle, that returns unbiased noisy gradients, $\nabla_x F(x, y; \xi)$ and $\nabla_y F(x, y; \xi)$. Also, we make the following additional assumptions.

Assumption 3.4 (stochastic gradients)**.** *For* $z \in \{x, y\}$*, we have* $\mathbb{E}_\xi\left[\nabla_z F(x, y, \xi)\right] = \nabla_z f(x, y)$*. In addition, there exists a constant* G *such that* $\|\nabla_z F(x, y, \xi)\| \leq G$ *for any* $x \in \mathbb{R}^{d_1}$ *and* $y \in \mathcal{Y}$*.*

Assumption 3.5 (bounded primal function value)**.** *There exists a constant* $\Phi_{\max} \in \mathbb{R}$ *such that for any* $x \in \mathbb{R}^{d_1}$*,* $\Phi(x)$ *is upper bounded by* $\Phi_{\max}$*.*

Remark 3.2. The bounded gradients and function value are assumed in many works on adaptive algorithms (Kavis et al., 2022; Levy et al., 2021). This implies the domain of y is bounded, which is also assumed in the analyses of AdaGrad (Levy, 2017; Levy et al., 2018). In neural networks with rectified activations, because of its scale-invariance property (Dinh et al., 2017), imposing boundedness of y does not affect the expressiveness. Wasserstein GANs (Arjovsky et al., 2017) also use projections on the critic to restrain the weights on a small cube around the origin.

Assumption 3.6 (second order Lipschitz continuity for y)**.** *For any* $x_1, x_2 \in \mathbb{R}^{d_1}$ *and* $y_1, y_2 \in \mathcal{Y}$*, there exists constant* L *such that* $\left\|\nabla^2_{xy} f(x_1, y_1) - \nabla^2_{xy} f(x_2, y_2)\right\| \leq L\left(\|x_1 - x_2\| + \|y_1 - y_2\|\right)$ *and* $\left\|\nabla^2_{yy} f(x_1, y_1) - \nabla^2_{yy} f(x_2, y_2)\right\| \leq L\left(\|x_1 - x_2\| + \|y_1 - y_2\|\right)$*.*

Remark 3.3. Chen et al. (2021) also impose this assumption to achieve the optimal $\mathcal{O}\left(\epsilon^{-4}\right)$ complexity for GDA with non-adaptive stepsizes for solving NC-SC minimax problems. Together with Assumption 3.3, we can show that $y^*(\cdot)$ is smooth. Nevertheless, without this assumption, Lin et al. (2020a) only show a worse complexity of $\mathcal{O}\left(\epsilon^{-5}\right)$ for GDA without large batchsize.

Theorem 3.2 (stochastic setting)**.** *Under Assumptions 3.1 to 3.6, Algorithm 1 with stochastic gradient oracles satisfies that for any* $0 < \beta < \alpha < 1$*, after* T *iterations,*

$$\frac{1}{T}\mathbb{E}\left[\sum_{t=0}^{T-1}\|\nabla_x f(x_t, y_t)\|^2 + \sum_{t=0}^{T-1}\|\nabla_y f(x_t, y_t)\|^2\right] \leq \mathcal{O}\left(T^{\alpha-1} + T^{-\alpha} + T^{\beta-1} + T^{-\beta}\right).$$

TiAda can achieve the complexity arbitrarily close to the optimal sample complexity, $\mathcal{O}\left(\epsilon^{-4}\right)$ (Li et al., 2021), by choosing α and β arbitrarily close to 0.5. Specifically, TiAda achieves a complexity of $\mathcal{O}\left(\epsilon^{-(4+\delta)}\right)$ for any small $\delta > 0$ if we set $\alpha = 0.5 + \delta/(8 + 2\delta)$ and $\beta = 0.5 - \delta/(8 + 2\delta)$. Notably, this matches the complexity of NeAda with AdaGrad stepsizes for both variables (Yang et al., 2022a). NeAda may attain $\widetilde{\mathcal{O}}(\epsilon^{-4})$ complexity with more complicated subroutines for y.

Theorem 3.2 implies that TiAda is fully agnostic to problem parameters, e.g., μ, l and σ. GDA with non-adaptive stepsizes (Lin et al., 2020a) and vanilla single-loop adaptive methods (Huang & Huang, 2021), such as AdaGrad and AMSGrad, all require knowledge of these parameters. Compared with the only parameter-agnostic algorithm, NeAda, our algorithm has several advantages. First, TiAda is a single-loop algorithm, while NeAda (Yang et al., 2022a) needs increasing inner-loop steps and a huge batchsize of order $\Omega\left(\epsilon^{-2}\right)$ to achieve its best complexity. Second, our stationary guarantee is for $\mathbb{E}\|\nabla_x f(x, y)\|^2 \leq \epsilon^2$, which is stronger than $\mathbb{E}\|\nabla_x f(x, y)\| \leq \epsilon$ guarantee in NeAda. Last but not least, although NeAda does not need to know the exact value of variance σ in the stochastic setting when $\sigma > 0$, NeAda uses a different stopping criterion for the inner loop in the deterministic

setting when $\sigma = 0$, so it still needs partial information about σ. In comparison, TiAda achieves the (near) optimal complexity in both settings with the same strategy.

Consistent with the intuition of time-scale adaptivity in Section 2.1, the convergence result can be derived in two stages. In Stage I, according to the upper bound of the ratio in Equation (3), we expect the term $1/\left(v_{t+1}^y\right)^{\alpha-\beta}$ reduces to a constant c, a desirable time-scale separation. This means that v_{t+1}^y has to grow to nearly $(1/c)^{1/(\alpha-\beta)}$. In Stage II, when the time-scale separation is satisfied, TiAda converges at a speed specified in Theorem 3.2. This indicates that the proximity between α and β affects the speed trade-off between Stage I and II. When α and β are close, we have a faster overall convergence rate close to the optimality, but suffer from a longer transition phase in Stage I, albeit by only a constant term. We also present an empirical ablation study on the convergence behavior with different choices of α and β in Appendix A.2.

Remark 3.4. In TiAda, the update of x requires to know the gradients of y (or v_{t+1}^y). However, in some applications that concern about privacy, one player might not access the information about the other player (Koller & Pfeffer, 1995; Foster & Young, 2006; He et al., 2016). Therefore, we also consider a variant of TiAda without taking the maximum of gradient norms, i.e., setting the effective stepsize of x in Algorithm 1 to $\eta^x/\left(v_{t+1}^x\right)^{\alpha}$. This variant achieves a sub-optimal complexity of $\widetilde{\mathcal{O}}\left(\epsilon^{-6}\right)$. This result further justifies the importance of coordination between adaptive stepsizes of two players for achieving faster convergence in minimax optimization. The algorithm and convergence results are presented in Appendix C.4.

4 EXPERIMENTS

In this section, we first present extensions of TiAda that accommodate other adaptive schemes besides AdaGrad and are more practical in deep models. Then we present empirical results of TiAda and compare it with (i) simple combinations of GDA and adaptive stepsizes, which are commonly used in practice, and (ii) NeAda with different adaptive mechanism (Yang et al., 2022a). Our experiments include test functions proposed by Yang et al. (2022a), the NC-SC distributional robustness optimization (Sinha et al., 2018), and training the NC-NC Wasserstein GAN with gradient penalty (Gulrajani et al., 2017). We believe that this not only validates our theoretical results but also shows the potential of our algorithm in real-world scenarios. To show the strength of being parameter-agnostic of TiAda, in all the experiments, we merely select $\alpha = 0.6$ and $\beta = 0.4$ without further tuning those two hyper-parameters. All experimental details including the neural network structure and hyper-parameters are described in Appendix A.1.

4.1 EXTENSIONS TO OTHER ADAPTIVE STEPSIZES AND HIGH-DIMENSIONAL MODELS

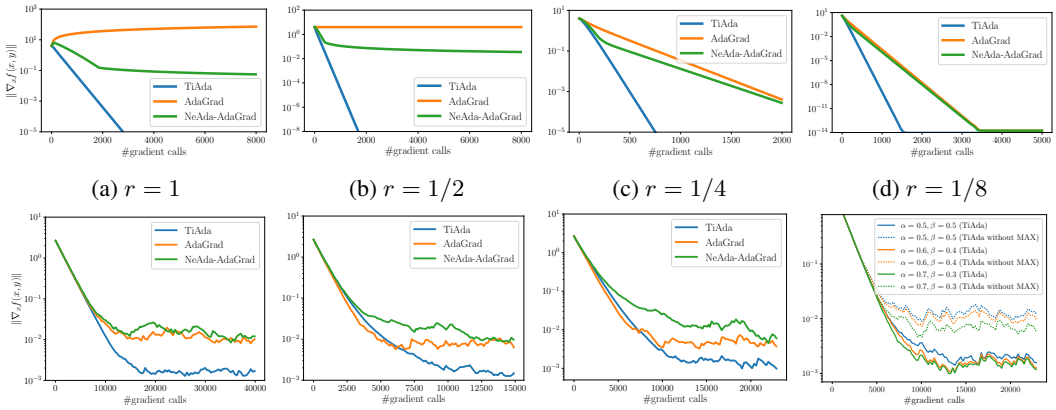
Although we design TiAda upon AdaGrad-Norm, it is easy and intuitive to apply other adaptive schemes like Adam and AMSGrad. To do so, for $z \in \{x, y\}$, we replace the definition of g_t^z and v_{t+1}^z in line 3 and 4 of Algorithm 1 to

$$g_t^z = \beta_t^z g_{t-1}^z + (1-\beta_t^z)\nabla_z F(x_t, y_t; \xi_t^z), \quad v_{t+1}^z = \psi\left(v_0, \left\{\|\nabla_z F(x_i, y_i; \xi_i^z)\|^2\right\}_{i=0}^t\right),$$

where $\{\beta_t^z\}$ is the momentum parameters and ψ is the second moment function. Some common stepsizes that fit in this generalized framework can be seen in Table 1 in the appendix. Since Adam is widely used in many deep learning tasks, we also implement generalized TiAda with Adam stepsizes in our experiments for real-world applications, and we label it "TiAda-Adam".

Besides generalizing TiAda to accommodate different stepsize schemes, for high-dimensional models, we also provide a coordinate-wise version of TiAda. Note that we cannot simply change everything in Algorithm 1 to be coordinate-wise, because we use the gradients of y in the stepsize of x and there are no corresponding relationships between the coordinates of x and y. Therefore, in light of our intuition in Equation (4), we use the global accumulated gradient norms to dynamically adjust the stepsize of x. Denote the second moment (analogous to v_{t+1}^x in Algorithm 1) for the i-th coordinate of x at the t-th step as $v_{t+1,i}^x$ and globally $v_{t+1}^x := \sum_{i=1}^{d_1} v_{t+1,i}^x$. We also use similar notations for y. Then, the update for the i-th parameter, i.e., x^i and y^i, can be written as

$$\begin{cases} x_{t+1}^i = x_t^i - \frac{\left(v_{t+1}^x\right)^{\alpha}}{\max\left\{v_{t+1}^x, v_{t+1}^y\right\}^{\alpha}} \cdot \frac{\eta^x}{\left(v_{t+1,i}^x\right)^{\alpha}} \nabla_{x^i} f(x_t, y_t) \\ y_{t+1}^i = y_t^i + \frac{\eta^y}{\left(v_{t+1,i}^y\right)^{\beta}} \nabla_{y^i} f(x_t, y_t). \end{cases}$$



(a) $r = 1$ (b) $r = 1/2$ (c) $r = 1/4$ (d) $r = 1/8$

(e) $r = 1/0.01$ (f) $r = 1/0.03$ (g) $r = 1/0.05$ (h) ablation study

Figure 2: Comparison of the algorithms on test functions. $r = \eta^x/\eta^y$ is the initial stepsize ratio. In the first row, we use the quadratic function (2) with $L = 2$ under deterministic gradient oracles. For the second row, we test the methods on the McCormick function with noisy gradients.

Our results in the following subsections provide strong empirical evidence for the effectiveness of these TiAda variants, and developing convergence guarantees for them would be an interesting future work. We believe our proof techniques for TiAda, together with existing convergence results for coordinate-wise AdaGrad and AMSGrad (Zhou et al., 2018; Chen et al., 2018; Défossez et al., 2020), can shed light on the theoretical analyses of these variants.

4.2 Test Functions

Firstly, we examine TiAda on the quadratic function (2) that shows the non-convergence of simple combinations of GDA and adaptive stepsizes (Yang et al., 2022a). Since our TiAda is based on AdaGrad, we compare it with GDA with AdaGrad stepsize and NeAda-AdaGrad (Yang et al., 2022a). The results are shown in the first row of Figure 2. When the initial ratio is poor, TiAda and NeAda-AdaGrad always converge while AdaGrad diverges. NeAda also suffers from slow convergence when the initial ratio is poor, e.g., 1 and $1/2$ after 2000 iterations. In contrast, TiAda automatically balances the stepsizes and converges fast under all ratios.

For the stochastic case, we follow Yang et al. (2022a) and conduct experiments on the McCormick function which is more complicated and 2-dimensional: $f(x,y) = \sin(x_1+x_2)+(x_1-x_2)^2-\frac{3}{2}x_1+\frac{5}{2}x_2+1+x_1y_1+x_2y_2-\frac{1}{2}(y_1^2+y_2^2)$. TiAda consistently outperforms AdaGrad and NeAda-AdaGrad as demonstrated in the second row of Figure 2 regardless of the initial ratio. In this function, we also run an ablation study on the effect of our design that uses max-operator in the update of x. We compare TiAda with and its variant without the max-operator, TiAda without MAX (Algorithm 2 in the appendix) whose effective stepsizes of x are $\eta^x/\left(v_{t+1}^x\right)^\alpha$. According to Figure 2(h), TiAda converges to smaller gradient norms under all configurations of α and β.

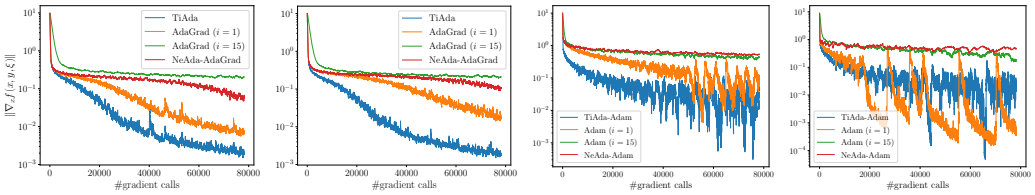
4.3 Distributional Robustness Optimization

In this subsection, we consider the distributional robustness optimization (Sinha et al., 2018). We target training the model weights, the primal variable x, to be robust to the perturbations in the image inputs, the dual variable y. The problem can be formulated as:

$$\min_x \max_{y=[y_1,\ldots,y_n]} \frac{1}{n}\sum_{i=1}^n f_i(x,y_i) - \gamma\|y_i - v_i\|^2, \tag{5}$$

where f_i is the loss function of the i-th sample, v_i is the i-th input image, and y_i is the corresponding perturbation. There are a total of n samples and γ is a trade-off hyper-parameter between the original loss and the penalty of the perturbations. If γ is large enough, the problem is NC-SC.

We conduct the experiments on the MNIST dataset (LeCun, 1998). In the left two plots of Figure 3, we compare TiAda with AdaGrad and NeAda-AdaGrad in terms of convergence. Since it is common in practice to update y 15 times after each x update (Sinha et al., 2018) for better generalization error,



(a) $\eta^x = 0.1, \eta^y = 0.05$ (b) $\eta^x = 0.1, \eta^y = 0.1$ (c) $\eta^x = 0.001, \eta^y = 0.1$ (d) $\eta^x = 0.001, \eta^y = 0.001$

Figure 3: Comparison of the algorithms on distributional robustness optimization (5). We use i in the legend to indicate the number of inner loops. Here we present two sets of stepsize configurations for the comparisons of AdaGrad-like and Adam-like algorithms. Please refer to Appendix A.3 for more extensive experiments on larger ranges of stepsizes, and it will be shown that TiAda is the best among all stepsize combinations in our grid.

we implement AdaGrad using both single and 15 iterations of inner loop (update of y). In order to show that TiAda is more robust to the initial stepsize ratio, we compare two sets of initial stepsize configurations with two different ratios. In both cases, TiAda outperforms NeAda and AdaGrad, especially when $\eta^x = \eta^y = 0.1$, the performance gap is large. In the right two plots of Figure 3, the Adam variants are compared. In this case, we find that TiAda is not only faster, but also more stable comparing to Adam with single inner loop iteration.

4.4 Generative Adversarial Networks

Another successful and popular application of minimax optimization is generative adversarial networks. In this task, a discriminator (or critic) is trained to distinguish whether an image is from the dataset. At the same time, a generator is mutually trained to synthesize samples with the same distribution as the training dataset so as to fool the discriminator. We use WGAN-GP loss (Gulrajani et al., 2017), which imposes the discriminator to be a 1-Lipschitz function, with CIFAR-10 dataset (Krizhevsky et al., 2009) in our experiments.

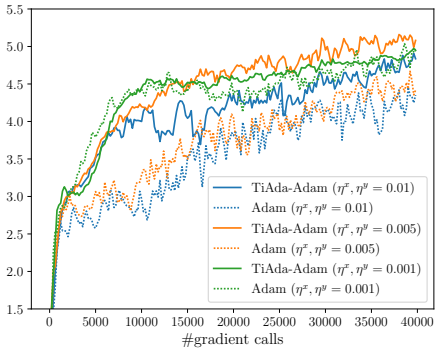


Figure 4: Inception score on WGAN-GP.

Since TiAda is a single-loop algorithm, for fair comparisons, we also update the discriminator only once for each generator update in Adam. In Figure 4, we plot the inception scores (Salimans et al., 2016) of TiAda-Adam and Adam under different initial stepsizes. We use the same color for the same initial stepsizes, and different line styles to distinguish the two methods, i.e., solid lines for TiAda-Adam and dashed lines for Adam. For all the three initial stepsizes we consider, TiAda-Adam achieves higher inception scores. Also, TiAda-Adam is more robust to initial stepsize selection, as the gap between different solid lines at the end of training is smaller than the dashed lines.

5 Conclusion

In this work, we bring in adaptive stepsizes to nonconvex minimax problems in a parameter-agnostic manner. We designed the first time-scale adaptive algorithm, TiAda, which progressively adjusts the effective stepsize ratio and reaches the desired time-scale separation. TiAda is also noise adaptive and does not require large batchsizes compared with the existing parameter-agnostic algorithm for nonconvex minimax optimization. Furthermore, TiAda is able to achieve optimal and near-optimal complexities respectively wtih deterministic and stochastic gradient oracles. We also empirically showcased the advantages of TiAda over NeAda and GDA with adaptive stepsizes on several tasks, including simple test functions, as well as NC-SC and NC-NC real-world applications. It remains an interesting problem to study whether TiAda can escape stationary points that are not local optimum, like adaptive methods for minimization problems (Staib et al., 2019).

REFERENCES

Kimon Antonakopoulos, Veronica Belmega, and Panayotis Mertikopoulos. An adaptive mirror-prox method for variational inequalities with singular operators. *NeurIPS*, 32, 2019.

Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein generative adversarial networks. In *ICML*, pp. 214–223. PMLR, 2017.

Francis Bach and Kfir Y Levy. A universal algorithm for variational inequalities adaptive to smoothness and noise. In *Conference on Learning Theory*, pp. 164–194. PMLR, 2019.

Radu Ioan Boţ and Axel Böhm. Alternating proximal-gradient steps for (stochastic) nonconvex-concave minimax problems. *arXiv preprint arXiv:2007.13605*, 2020.

Tianyi Chen, Yuejiao Sun, and Wotao Yin. Closing the gap: Tighter analysis of alternating stochastic gradient methods for bilevel problems. 2021.

Xiangyi Chen, Sijia Liu, Ruoyu Sun, and Mingyi Hong. On the convergence of a class of adam-type algorithms for non-convex optimization. *arXiv preprint arXiv:1808.02941*, 2018.

Bo Dai, Niao He, Yunpeng Pan, Byron Boots, and Le Song. Learning from conditional distributions via dual embeddings. In *AISTATS*, pp. 1458–1467. PMLR, 2017.

Constantinos Daskalakis, Andrew Ilyas, Vasilis Syrgkanis, and Haoyang Zeng. Training GANs with optimism. In *ICLR*, 2018.

Constantinos Daskalakis, Stratis Skoulakis, and Manolis Zampetakis. The complexity of constrained min-max optimization. In *Proceedings of the 53rd Annual ACM SIGACT Symposium on Theory of Computing*, pp. 1466–1478, 2021.

Alexandre Défossez, Léon Bottou, Francis Bach, and Nicolas Usunier. A simple convergence proof of adam and adagrad. *arXiv preprint arXiv:2003.02395*, 2020.

Laurent Dinh, Razvan Pascanu, Samy Bengio, and Yoshua Bengio. Sharp minima can generalize for deep nets. In *ICML*, pp. 1019–1028. PMLR, 2017.

John Duchi, Elad Hazan, and Yoram Singer. Adaptive subgradient methods for online learning and stochastic optimization. *Journal of machine learning research*, 12(7), 2011.

Pierre Foret, Ariel Kleiner, Hossein Mobahi, and Behnam Neyshabur. Sharpness-aware minimization for efficiently improving generalization. In *ICLR*, 2021.

Dean Foster and Hobart Peyton Young. Regret testing: Learning to play nash equilibrium without knowing you have an opponent. *Theoretical Economics*, 1(3):341–367, 2006.

Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky. Domain-adversarial training of neural networks. *The journal of machine learning research*, 17(1):2096–2030, 2016.

Gauthier Gidel, Hugo Berard, Gaëtan Vignoud, Pascal Vincent, and Simon Lacoste-Julien. A variational inequality perspective on generative adversarial networks. *arXiv preprint arXiv:1802.10551*, 2018.

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

Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. *NeurIPS*, 27, 2014.

Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. In *ICLR*, 2015.

Green9. Pytorch code for gan models. `https://github.com/Zeleni9/pytorch-wgan`, 2018.

Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, and Aaron C Courville. Improved training of wasserstein gans. *NeurIPS*, 30, 2017.

Zhishuai Guo, Yi Xu, Wotao Yin, Rong Jin, and Tianbao Yang. A novel convergence analysis for algorithms of the adam family. *arXiv preprint arXiv:2112.03459*, 2021.

He He, Jordan Boyd-Graber, Kevin Kwok, and Hal Daumé III. Opponent modeling in deep reinforcement learning. In *ICML*, pp. 1804–1813. PMLR, 2016.

Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *Advances in neural information processing systems*, 30, 2017.

Geoffrey Hinton, Nitish Srivastava, and Kevin Swersky. Neural networks for machine learning lecture 6a overview of mini-batch gradient descent. *Cited on*, 14(8):2, 2012.

Feihu Huang and Heng Huang. Adagda: Faster adaptive gradient descent ascent methods for minimax optimization. *arXiv preprint arXiv:2106.16101*, 2021.

Feihu Huang, Xidong Wu, and Heng Huang. Efficient mirror descent ascent methods for nonsmooth minimax problems. *NeurIPS*, 34:10431–10443, 2021.

Ali Kavis, Kfir Y Levy, Francis Bach, and Volkan Cevher. Unixgrad: A universal, adaptive algorithm with optimal guarantees for constrained optimization. *NeurIPS*, 32, 2019.

Ali Kavis, Kfir Levy, and Volkan Cevher. High probability bounds for a class of nonconvex algorithms with adagrad stepsize. In *ICLR*, 2022.

Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *ICLR*, 2015.

Daphne Koller and Avi Pfeffer. Generating and solving imperfect information games. In *IJCAI*, pp. 1185–1193. Citeseer, 1995.

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

Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009.

Yann LeCun. The mnist database of handwritten digits. *http://yann. lecun. com/exdb/mnist/*, 1998.

Kfir Levy. Online to offline conversions, universality and adaptive minibatch sizes. *NeurIPS*, 30, 2017.

Kfir Levy, Ali Kavis, and Volkan Cevher. Storm+: Fully adaptive sgd with recursive momentum for nonconvex optimization. *NeurIPS*, 34:20571–20582, 2021.

Kfir Y Levy, Alp Yurtsever, and Volkan Cevher. Online adaptive methods, universality and acceleration. *NeurIPS*, 31, 2018.

Haochuan Li, Yi Tian, Jingzhao Zhang, and Ali Jadbabaie. Complexity lower bounds for nonconvex-strongly-concave min-max optimization. *NeurIPS*, 34:1792–1804, 2021.

Haochuan Li, Farzan Farnia, Subhro Das, and Ali Jadbabaie. On convergence of gradient descent ascent: A tight local analysis. In *ICML*, pp. 12717–12740. PMLR, 2022.

Xiaoyu Li and Francesco Orabona. On the convergence of stochastic gradient descent with adaptive stepsizes. In *The 22nd international conference on artificial intelligence and statistics*, pp. 983–992. PMLR, 2019.

Xiaoyu Li and Francesco Orabona. A high probability analysis of adaptive sgd with momentum. *arXiv preprint arXiv:2007.14294*, 2020.

Tianyi Lin, Chi Jin, and Michael Jordan. On gradient descent ascent for nonconvex-concave minimax problems. In *ICML*, pp. 6083–6093. PMLR, 2020a.

Tianyi Lin, Chi Jin, and Michael I Jordan. Near-optimal algorithms for minimax optimization. In *Conference on Learning Theory*, pp. 2738–2779. PMLR, 2020b.

Louis Lv. Reproducing "certifying some distributional robustness with principled adversarial training". `https://github.com/Louis-udm/Reproducing-certifiable-distributional-robustness`, 2019.

David J Miller, Zhen Xiang, and George Kesidis. Adversarial learning targeting deep neural network classification: A comprehensive review of defenses against attacks. *Proceedings of the IEEE*, 108 (3):402–433, 2020.

Konstantin Mishchenko, Dmitry Kovalev, Egor Shulgin, Peter Richtárik, and Yura Malitsky. Revisiting stochastic extragradient. In *International Conference on Artificial Intelligence and Statistics*, pp. 4573–4582. PMLR, 2020.

Aditya Modi, Jinglin Chen, Akshay Krishnamurthy, Nan Jiang, and Alekh Agarwal. Model-free representation learning and exploration in low-rank mdps. *arXiv preprint arXiv:2102.07035*, 2021.

Yurii Nesterov. *Introductory lectures on convex optimization: A basic course*, volume 87. Springer Science & Business Media, 2003.

Maher Nouiehed, Maziar Sanjabi, Tianjian Huang, Jason D Lee, and Meisam Razaviyayn. Solving a class of non-convex min-max games using iterative first order methods. *NeurIPS*, 32, 2019.

Leonid Denisovich Popov. A modification of the arrow-hurwicz method for search of saddle points. *Mathematical notes of the Academy of Sciences of the USSR*, 28(5):845–848, 1980.

Alexander Rakhlin and Karthik Sridharan. Online learning with predictable sequences. In *Conference on Learning Theory*, pp. 993–1019. PMLR, 2013.

Sashank J Reddi, Satyen Kale, and Sanjiv Kumar. On the convergence of adam and beyond. In *ICLR*, 2018.

Amirhossein Reisizadeh, Farzan Farnia, Ramtin Pedarsani, and Ali Jadbabaie. Robust federated learning: The case of affine distribution shifts. *Advances in Neural Information Processing Systems*, 33:21554–21565, 2020.

Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. Improved techniques for training gans. *NeurIPS*, 29, 2016.

Othmane Sebbouh, Marco Cuturi, and Gabriel Peyré. Randomized stochastic gradient descent ascent. In *AISTATS*, pp. 2941–2969. PMLR, 2022.

Aman Sinha, Hongseok Namkoong, and John Duchi. Certifiable distributional robustness with principled adversarial training. In *ICLR*, 2018.

Matthew Staib, Sashank Reddi, Satyen Kale, Sanjiv Kumar, and Suvrit Sra. Escaping saddle points with adaptive gradient methods. In *International Conference on Machine Learning*, pp. 5956–5965. PMLR, 2019.

Rachel Ward, Xiaoxia Wu, and Leon Bottou. Adagrad stepsizes: Sharp convergence over nonconvex landscapes. *The Journal of Machine Learning Research*, 21(1):9047–9076, 2020.

Yuege Xie, Xiaoxia Wu, and Rachel Ward. Linear convergence of adaptive stochastic gradient descent. In *AISTATS*, pp. 1475–1485. PMLR, 2020.

Junchi Yang, Xiang Li, and Niao He. Nest your adaptive algorithm for parameter-agnostic nonconvex minimax optimization. *arXiv preprint arXiv:2206.00743*, 2022a.

Junchi Yang, Antonio Orvieto, Aurelien Lucchi, and Niao He. Faster single-loop algorithms for minimax optimization without strong concavity. In *AISTATS*, pp. 5485–5517. PMLR, 2022b.

Siqi Zhang, Junchi Yang, Cristóbal Guzmán, Negar Kiyavash, and Niao He. The complexity of nonconvex-strongly-concave minimax optimization. In *Uncertainty in Artificial Intelligence*, pp. 482–492. PMLR, 2021.

Xuan Zhang, Necdet Serhat Aybat, and Mert Gurbuzbalaban. Sapd+: An accelerated stochastic method for nonconvex-concave minimax problems. *arXiv preprint arXiv:2205.15084*, 2022a.

Yushun Zhang, Congliang Chen, Naichen Shi, Ruoyu Sun, and Zhi-Quan Luo. Adam can converge without any modification on update rules. *arXiv preprint arXiv:2208.09632*, 2022b.

Dongruo Zhou, Jinghui Chen, Yuan Cao, Yiqi Tang, Ziyan Yang, and Quanquan Gu. On the convergence of adaptive gradient methods for nonconvex optimization. *arXiv preprint arXiv:1808.05671*, 2018.

A Supplementary to Experiments

Table 1: Stepsize schemes fit in generalized TiAda. See also (Yang et al., 2022a).

Algorithms	first moment parameter β_t	second moment function $\psi\left(v_0, \{u_i^2\}_{i=0}^t\right)$
AdaGrad (TiAda)	$\beta_t = 0$	$v_0 + \sum_{i=0}^t u_i^2$
GDA	$\beta_t = 0$	1
Adam	$0 < \beta_t < 1$	$\gamma^{t+1} v_0 + (1-\gamma)\sum_{i=0}^t \gamma^{t-i} u_i^2$
AMSGrad	$0 < \beta_t < 1$	$\max_{m=0,\ldots,t} \gamma^{m+1} v_0 + (1-\gamma)\sum_{i=0}^m \gamma^{m-i} u_i^2$

A.1 Experimental Details

In this section, we will summarize the experimental settings and hyper-parameters used. As we mentioned, since we try to develop a parameter-agnostic algorithm without tuning the hyper-parameters much, if not specified, we simply use $\alpha = 0.6$ and $\beta = 0.4$ for all experiments. For fair comparisons, we used the same hyper-parameters when comparing our TiAda with other algorithms.

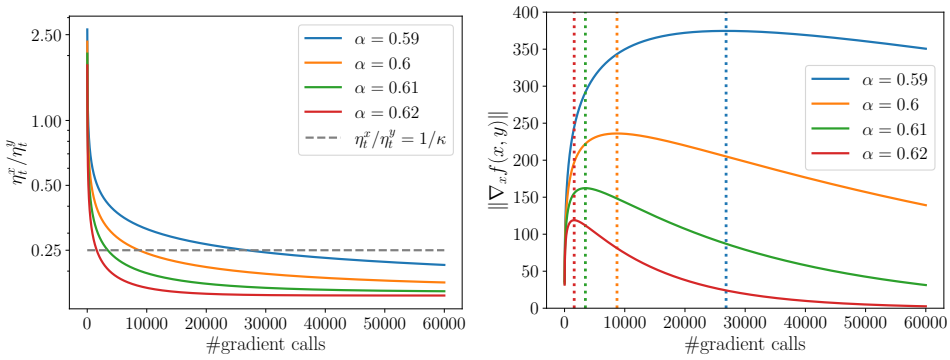
Test Functions For Figure 1 and the first row of Figure 2, we conduct experiments on problem (2) with $L = 2$. We use initial stepsize $\eta^y = 0.2$ and initial point $(1, 0.01)$ for all runs. As for the McCormick function used in the second row of Figure 2, we chose $\eta^y = 0.01$. and the noises added to the gradients are from zero-mean Gaussian distribution with variance 0.01.

Distributional Robustness Optimization For results shown in Figures 3, 6 and 7, we adapt code from Lv (2019), and used the same hyper-parameter setting as Sinha et al. (2018); Sebbouh et al. (2022), i.e., $\gamma = 1.3$. The model we used is a three layer convolutional neural network (CNN) with a final fully-connected layer. For each layer, batch normalization and ELU activation are used. The width of each layer is $(32, 64, 128, 512)$. The setting is the same as Sinha et al. (2018); Yang et al. (2022a). We set the batchsize as 128, and for the Adam-like optimizers, including Adam, NeAda-Adam and TiAda-Adam, we use $\beta_1 = 0.9, \beta_2 = 0.999$ for the first moment and second moment parameters.

Generative Adversarial Networks For this part, we use the code adapted from Green9 (2018). To produce the results in Figure 4, a four layer CNN and a four layer CNN with transpose convolution layers are used respectively for the discriminator and generator. Following a similar setting as Daskalakis et al. (2018), we set batchsize as 512, the dimension of latent variable as 50 and the weight of gradient penalty term as 10^{-4}. For the Adam-like optimizers, we set $\beta_1 = 0.5, \beta_2 = 0.9$. To get the inception score, we feed the pre-trained inception network with 8000 synthesized samples.

A.2 Ablation Study on Convergence Behavior with Different α and β

We conduct experiments on the quadratic minimax problem (2) with $L = 2$ to study the effect of hyper-parameters α and β on the convergence behavior of TiAda. As discussed in Sections 1 and 3.2, we refer to the period before the stepsize ratio reduce to the convergence threshold as Stage I, and the period after that as Stage II. In order to accentuate the difference between these two stages, we pick a large initial stepsize ratio $\eta^x/\eta^y = 20$. We compare 4 different pairs of α and β: $\alpha \in \{0.59, 0.6, 0.61, 0.62\}$ and $\beta = 1-\alpha$. From Figure 5, we observed that as soon as TiAda enters Stage II, the norm of gradients start to drop. Moreover, the closer α and β are to 0.5, the more time TiAda remains in Stage I, which confirms the intuitions behind our analysis in Section 3.2.



(a) effective stepsize ratio

(b) convergence

Figure 5: Illustration of the effect of α and β on the two stages in TiAda's time-scale adaptation process. We set $\beta = 1 - \alpha$. The dashed line on the right plot represents the first iteration when the effective stepsize ratio is below $1/\kappa$.

A.3 Additional Experiments on Distributional Robustness Optimization

We use a grid of stepsize combinations to evaluate TiAda and compare it with NeAda and GDA with corresponding adaptive stepsizes. For AdaGrad-like algorithms, we use $\{0.1, 0.05, 0.01, 0.0005\}$ for both η^x and η^y, and the results are reported in Figure 6. For Adam-like algorithms, we use $\{0.001, 0.0005, 0.0001\}$ for η^x and $\{0.1, 0.05, 0.005, 0.001\}$ for η^y, and the results are shown in Figure 7. We note that since Adam uses the reciprocal of the moving average of gradient norms, it is extremely unstable when the gradients are small. Therefore, Adam-like algorithms often experience instability when they are near stationary points.

B Helper Lemmas

Lemma B.1. *Let $x_1, ..., x_T$ be a sequence of non-negative real numbers, $x_1 > 0$ and $0 < \alpha < 1$. Then we have*

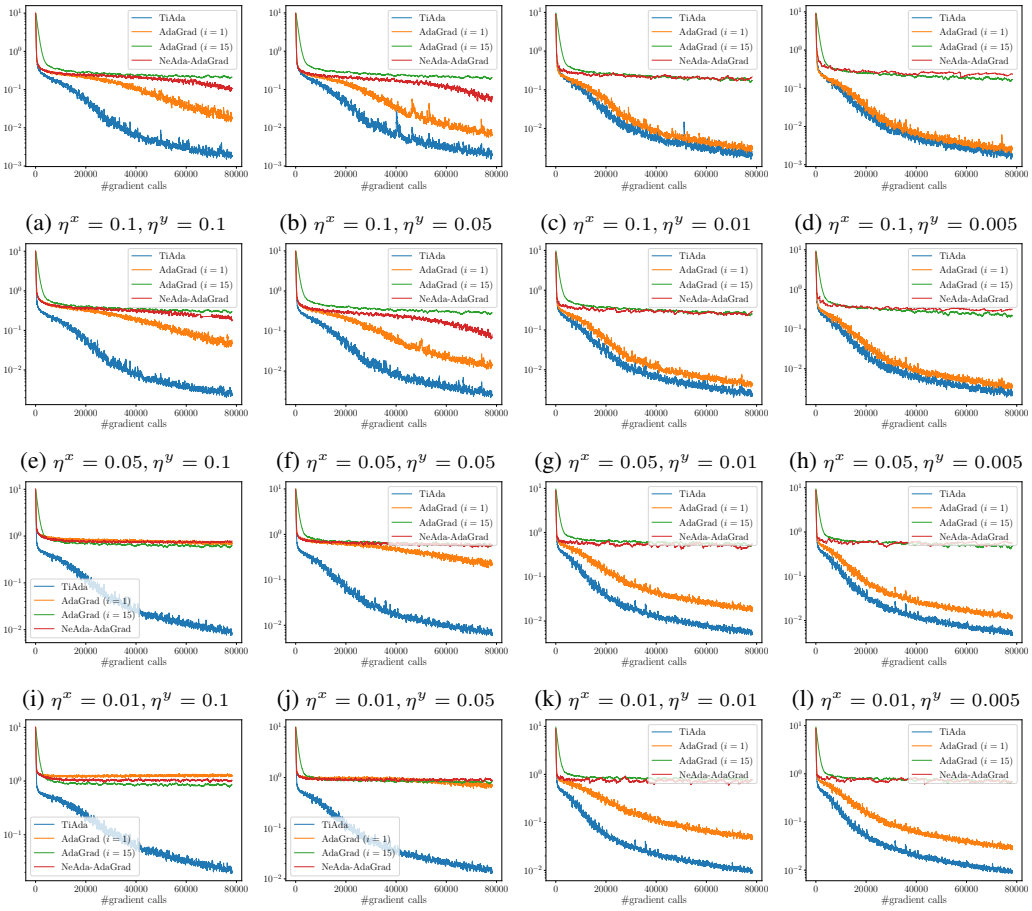
$$\left(\sum_{t=1}^{T} x_t\right)^{1-\alpha} \leq \sum_{t=1}^{T} \frac{x_t}{\left(\sum_{k=1}^{t} x_k\right)^{\alpha}} \leq \frac{1}{1-\alpha}\left(\sum_{t=1}^{T} x_t\right)^{1-\alpha}.$$

When $\alpha = 1$, we have

$$\sum_{t=1}^{T} \frac{x_t}{\left(\sum_{k=1}^{t} x_k\right)^{\alpha}} \leq 1 + \log\left(\frac{\sum_{t=1}^{t} x_t}{x_1}\right).$$

Proof. For the first inequality in the case $0 < \alpha < 1$, we have

$$\sum_{t=1}^{T} \frac{x_t}{\left(\sum_{k=1}^{t} x_k\right)^{\alpha}} \geq \sum_{t=1}^{T} \frac{x_t}{\left(\sum_{k=1}^{T} x_k\right)^{\alpha}} = \frac{\sum_{t=1}^{T} x_t}{\left(\sum_{t=1}^{T} x_t\right)^{\alpha}} = \left(\sum_{t=1}^{T} x_t\right)^{1-\alpha}.$$



(a) $\eta^x = 0.1, \eta^y = 0.1$ (b) $\eta^x = 0.1, \eta^y = 0.05$ (c) $\eta^x = 0.1, \eta^y = 0.01$ (d) $\eta^x = 0.1, \eta^y = 0.005$

(e) $\eta^x = 0.05, \eta^y = 0.1$ (f) $\eta^x = 0.05, \eta^y = 0.05$ (g) $\eta^x = 0.05, \eta^y = 0.01$ (h) $\eta^x = 0.05, \eta^y = 0.005$

(i) $\eta^x = 0.01, \eta^y = 0.1$ (j) $\eta^x = 0.01, \eta^y = 0.05$ (k) $\eta^x = 0.01, \eta^y = 0.01$ (l) $\eta^x = 0.01, \eta^y = 0.005$

(m) $\eta^x = 0.005, \eta^y = 0.1$ (n) $\eta^x = 0.005, \eta^y = 0.05$ (o) $\eta^x = 0.005, \eta^y = 0.01$ (p) $\eta^x = 0.005, \eta^y = 0.005$

Figure 6: Gradient norms in x of AdaGrad-like algorithms on distributional robustness optimization (5). We use i in the legend to indicate the number of inner loops.

The second inequality in the case $0 < \alpha < 1$ is proved in Lemma 3 of (Levy et al., 2021). For $\alpha = 1$, it is proved in Lemma 3.2 of Ward et al. (2020). □

Lemma B.2 (smoothness of $\Phi(\cdot)$ and Lipschitzness of $y^*(\cdot)$. Lemma 4.3 in Lin et al. (2020a))**.** *Under Assumptions 3.1 and 3.2, we have $\Phi(\cdot)$ is $(l + \kappa l)$-smooth with $\nabla\Phi(x) = \nabla_x f(x, y^*(x))$, and $y^*(\cdot)$ is κ-Lipschitz.*

Lemma B.3 (smoothness of $y^*(\cdot)$. Lemma 2 in Chen et al. (2021))**.** *Under Assumptions 3.1, 3.2 and 3.6, we have for some constant $\widehat{L}$,*

$$\|\nabla y^*(x_1) - \nabla y^*(x_2)\| \leq \widehat{L}\|x_1 - x_2\|.$$

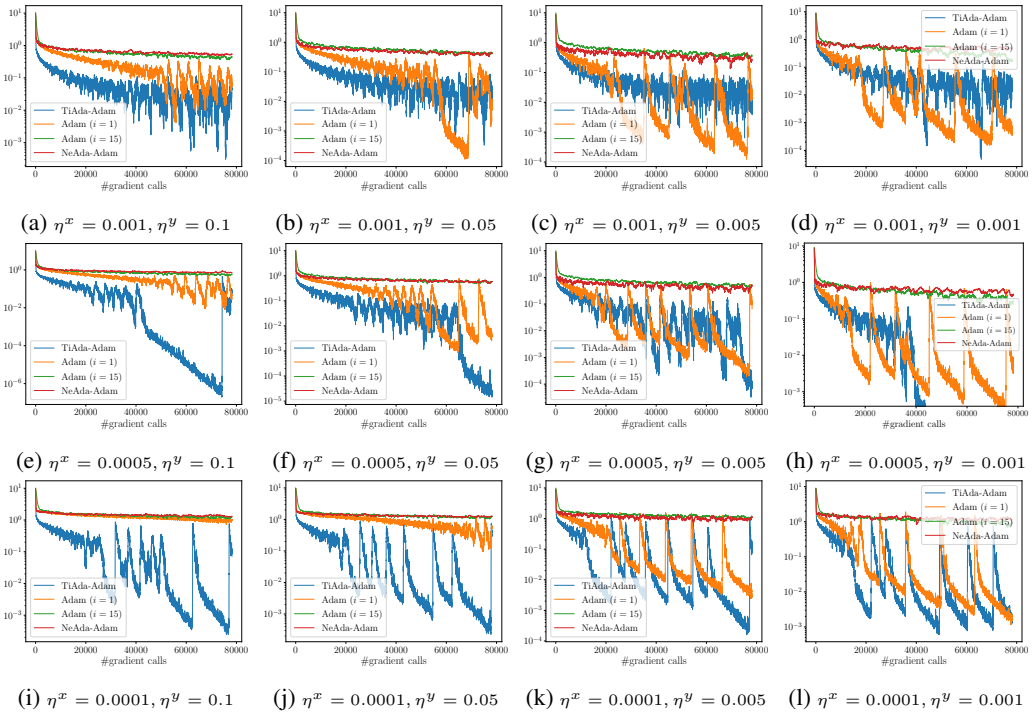
C PROOFS

For notational convenience in the proofs, we denote the stochastic gradient as $\nabla_x \widetilde{f}(x_t, y_t)$ and $\nabla_y \widetilde{f}(x_t, y_t)$. Also denote $y_t^* = y^*(x_t)$, $\eta_t = \frac{\eta^x}{\max\{v_{t+1}^x, v_{t+1}^y\}^\alpha}$ and $\gamma_t = \frac{\eta^y}{(v_{t+1}^y)^\beta}$.

C.1 PROOF OF THEOREM 3.1

Proof. Let us start from the smoothness of the primal function $\Phi(\cdot)$. By Lemma B.2,

$$\Phi(x_{t+1})$$



(a) $\eta^x = 0.001, \eta^y = 0.1$ (b) $\eta^x = 0.001, \eta^y = 0.05$ (c) $\eta^x = 0.001, \eta^y = 0.005$ (d) $\eta^x = 0.001, \eta^y = 0.001$

(e) $\eta^x = 0.0005, \eta^y = 0.1$ (f) $\eta^x = 0.0005, \eta^y = 0.05$ (g) $\eta^x = 0.0005, \eta^y = 0.005$ (h) $\eta^x = 0.0005, \eta^y = 0.001$

(i) $\eta^x = 0.0001, \eta^y = 0.1$ (j) $\eta^x = 0.0001, \eta^y = 0.05$ (k) $\eta^x = 0.0001, \eta^y = 0.005$ (l) $\eta^x = 0.0001, \eta^y = 0.001$

Figure 7: Gradient norms in x of Adam-like algorithms on distributional robustness optimization (5). We use i in the legend to indicate the number of inner loops.

$$
\begin{aligned}
&\leq \Phi(x_t) - \eta_t \langle \Phi(x_{t+1}), \nabla_x f(x_t, y_t) \rangle + kl\eta_t^2 \|\nabla_x f(x_t, y_t)\|^2 \\
&= \Phi(x_t) - \eta_t \|\nabla_x f(x_t, y_t)\|^2 + \eta_t \langle \nabla_x f(x_t, y_t) - \nabla \Phi(x_t), \nabla_x f(x_t, y_t) \rangle + kl\eta_t^2 \|\nabla_x f(x_t, y_t)\|^2 \\
&\leq \Phi(x_t) - \eta_t \|\nabla_x f(x_t, y_t)\|^2 + \frac{\eta_t}{2} \|\nabla_x f(x_t, y_t)\|^2 + \frac{\eta_t}{2} \|\nabla_x f(x_t, y_t) - \nabla \Phi(x_t)\|^2 + kl\eta_t^2 \|\nabla_x f(x_t, y_t)\|^2 \\
&= \Phi(x_t) - \frac{\eta_t}{2} \|\nabla_x f(x_t, y_t)\|^2 + kl\eta_t^2 \|\nabla_x f(x_t, y_t)\|^2 + \frac{\eta_t}{2} \|\nabla_x f(x_t, y_t) - \nabla \Phi(x_t)\|^2 \\
&= \Phi(x_t) - \frac{\eta_t}{2} \|\nabla_x f(x_t, y_t)\|^2 + kl\eta_t^2 \|\nabla_x f(x_t, y_t)\|^2 + \frac{\eta^x}{2 \max\left\{v_{t+1}^x, v_{t+1}^y\right\}^\alpha} \|\nabla_x f(x_t, y_t) - \nabla \Phi(x_t)\|^2 \\
&\leq \Phi(x_t) - \frac{\eta_t}{2} \|\nabla_x f(x_t, y_t)\|^2 + kl\eta_t^2 \|\nabla_x f(x_t, y_t)\|^2 + \frac{\eta^x}{2 \left(v_{t_0}^y\right)^{\alpha-\beta} \left(v_{t+1}^y\right)^\beta} \|\nabla_x f(x_t, y_t) - \nabla \Phi(x_t)\|^2 \\
&\leq \Phi(x_t) - \frac{\eta_t}{2} \|\nabla_x f(x_t, y_t)\|^2 + kl\eta_t^2 \|\nabla_x f(x_t, y_t)\|^2 + \frac{\eta^x \kappa^2}{2 \left(v_{t_0}^y\right)^{\alpha-\beta} \left(v_{t+1}^y\right)^\beta} \|\nabla_y f(x_t, y_t)\|^2 \\
&\leq \Phi(x_t) - \frac{\eta_t}{2} \|\nabla_x f(x_t, y_t)\|^2 + kl\eta_t^2 \|\nabla_x f(x_t, y_t)\|^2 + \frac{\eta^x \kappa^2}{2\eta^y \left(v_{t_0}^y\right)^{\alpha-\beta}} \cdot \gamma_t \|\nabla_y f(x_t, y_t)\|^2,
\end{aligned}
$$

where in the second to last inequality, we used the strong-concavity of $f(x, \cdot)$:

$$
\|\nabla_x f(x_t, y_t) - \nabla \Phi(x_t)\| \leq l \|y_t - y_t^*\| \leq \kappa \|\nabla_y f(x_t, y_t)\|.
$$

Telescoping and rearranging the terms, we have

$$
\begin{aligned}
&\sum_{t=0}^{T-1} \eta_t \|\nabla_x f(x_t, y_t)\|^2 \\
&\leq \underbrace{2\left(\Phi(x_0) - \Phi^*\right)}_{c_1} + 2\kappa l \sum_{t=0}^{T-1} \eta_t^2 \|\nabla_x f(x_t, y_t)\|^2 + \underbrace{\frac{\eta^x \kappa^2}{\eta^y \left(v_{t_0}^y\right)^{\alpha-\beta}}}_{c_2} \sum_{t=0}^{T-1} \gamma_t \|\nabla_y f(x_t, y_t)\|^2
\end{aligned}
$$

$$
\begin{aligned}
&= c_1 + \sum_{t=0}^{T-1} \frac{2\kappa l \eta^x}{\max\left\{v_{t+1}^x, v_{t+1}^y\right\}^{2\alpha}} \|\nabla_x f(x_t, y_t)\|^2 + c_2 \sum_{t=0}^{T-1} \gamma_t \|\nabla_y f(x_t, y_t)\|^2 \\
&\leq c_1 + \sum_{t=0}^{T-1} \frac{2\kappa l \eta^x}{\left(v_{t+1}^x\right)^{2\alpha}} \|\nabla_x f(x_t, y_t)\|^2 + c_2 \sum_{t=0}^{T-1} \gamma_t \|\nabla_y f(x_t, y_t)\|^2 \\
&\leq \mathcal{O}\left(\left(v_T^x\right)^{\max\{1-2\alpha, 0\}}\right) + c_2 \sum_{t=0}^{T-1} \gamma_t \|\nabla_y f(x_t, y_t)\|^2. \qquad (6)
\end{aligned}
$$

We proceed to bound $\sum_{t=0}^{T-1} \gamma_t \|\nabla_y f(x_t, y_t)\|^2$. Let t_0 be the first iteration such that $\left(v_{t_0+1}^y\right)^\beta > c_3 := \max\left\{\frac{4\eta^y \mu l}{\mu + l}, \eta^y(\mu + l)\right\}$. We have $v_{t_0}^y \leq c_3^{1/\beta}$, and for $t \geq t_0$,

$$
\begin{aligned}
&\left\|y_{t+1} - y_{t+1}^*\right\|^2 \\
&\leq (1 + \lambda_t)\|y_{t+1} - y_t^*\|^2 + \left(1 + \frac{1}{\lambda_t}\right)\left\|y_{t+1}^* - y_t^*\right\|^2 \\
&\leq \underbrace{(1 + \lambda_t)\left(\|y_t - y_t^*\|^2 + \frac{(\eta^y)^2}{\left(v_{t+1}^y\right)^{2\beta}}\|\nabla_y f(x_t, y_t)\|^2 + \frac{2\eta^y}{\left(v_{t+1}^y\right)^\beta}\langle y_t - y_t^*, \nabla_y f(x_t, y_t)\rangle\right)}_{\text{(A)}} \\
&\quad + \left(1 + \frac{1}{\lambda_t}\right)\left\|y_{t+1}^* - y_t^*\right\|^2,
\end{aligned}
$$

where $\lambda_t > 0$ will be determined later. For l-smooth and μ-strongly convex function $g(x)$, according to Theorem 2.1.12 in Nesterov (2003), we have

$$
\langle \nabla g(x) - \nabla g(y), x - y\rangle \geq \frac{\mu l}{\mu + l}\|x - y\|^2 + \frac{1}{\mu + l}\|\nabla g(x) - \nabla g(y)\|^2.
$$

Therefore,

$$
\begin{aligned}
&\text{Term (A)} \\
&\leq (1 + \lambda_t)\left(\left(1 - \frac{2\eta^y \mu l}{(\mu + l)\left(v_{t+1}^y\right)^\beta}\right)\|y_t - y_t^*\|^2 + \left(\frac{(\eta^y)^2}{\left(v_{t+1}^y\right)^{2\beta}} - \frac{2\eta^y}{(\mu + l)\left(v_{t+1}^y\right)^\beta}\right)\|\nabla_y f(x_t, y_t)\|^2\right).
\end{aligned}
$$

Let $\lambda_t = \frac{\eta^y \mu l}{(\mu+l)\left(v_{t+1}^y\right)^\beta - 2\eta^y \mu l}$. Note that $\lambda_t > 0$ after t_0. Then

$$
\begin{aligned}
&\text{Term (A)} \\
&\leq \left(1 - \frac{\eta^y \mu l}{(\mu + l)\left(v_{t+1}^y\right)^\beta}\right)\|y_t - y_t^*\|^2 + (1 + \lambda_t)\left(\frac{(\eta^y)^2}{\left(v_{t+1}^y\right)^{2\beta}} - \frac{2\eta^y}{(\mu + l)\left(v_{t+1}^y\right)^\beta}\right)\|\nabla_y f(x_t, y_t)\|^2 \\
&\leq \|y_t - y_t^*\|^2 + \underbrace{(1 + \lambda_t)\left(\frac{(\eta^y)^2}{\left(v_{t+1}^y\right)^{2\beta}} - \frac{2\eta^y}{(\mu + l)\left(v_{t+1}^y\right)^\beta}\right)}_{\text{(B)}}\|\nabla_y f(x_t, y_t)\|^2.
\end{aligned}
$$

As $1 + \lambda_t \geq 1$ and $\left(v_{t+1}^y\right)^\beta \geq \eta^y(\mu + l)$, we have term (B) $\leq -\frac{\eta^y}{(\mu+l)\left(v_{t+1}^y\right)^\beta}$. Putting them back, we can get

$$
\begin{aligned}
&\left\|y_{t+1} - y_{t+1}^*\right\|^2 \\
&\leq \|y_t - y_t^*\|^2 - \frac{\eta^y}{(\mu + l)\left(v_{t+1}^y\right)^\beta}\|\nabla_y f(x_t, y_t)\|^2 + \left(1 + \frac{1}{\lambda_t}\right)\left\|y_{t+1}^* - y_t^*\right\|^2 \\
&\leq \|y_t - y_t^*\|^2 - \frac{\eta^y}{(\mu + l)\left(v_{t+1}^y\right)^\beta}\|\nabla_y f(x_t, y_t)\|^2 + \frac{(\mu + l)\left(v_{t+1}^y\right)^\beta}{\eta^y \mu l}\left\|y_{t+1}^* - y_t^*\right\|^2
\end{aligned}
$$

$$\leq \|y_t - y_t^*\|^2 - \frac{\eta^y}{(\mu+l)\left(v_{t+1}^y\right)^\beta}\|\nabla_y f(x_t, y_t)\|^2 + \frac{(\mu+l)\kappa^2\left(v_{t+1}^y\right)^\beta}{\eta^y \mu l}\|x_{x+1} - x_t\|^2$$

$$= \|y_t - y_t^*\|^2 - \frac{\eta^y}{(\mu+l)\left(v_{t+1}^y\right)^\beta}\|\nabla_y f(x_t, y_t)\|^2 + \frac{(\mu+l)\kappa^2\left(v_{t+1}^y\right)^\beta \eta_t^2}{\eta^y \mu l}\|\nabla_x f(x_t, y_t)\|^2.$$

Then, by telescoping, we have

$$\sum_{t=t_0}^{T-1} \frac{\eta^y}{(\mu+l)\left(v_{t+1}^y\right)^\beta}\|\nabla_y f(x_t, y_t)\|^2 \leq \left\|y_{t_0} - y_{t_0}^*\right\|^2 + \sum_{t=t_0}^{T-1} \frac{(\mu+l)\kappa^2\left(v_{t+1}^y\right)^\beta \eta_t^2}{\eta^y \mu l}\|\nabla_x f(x_t, y_t)\|^2. \tag{7}$$

For the first term in the RHS, using Young's inequality with τ to be determined later, we have

$$\begin{aligned}
\left\|y_{t_0} - y_{t_0}^*\right\|^2 &\leq 2\left\|y_{t_0} - y_{t_0-1}^*\right\|^2 + 2\left\|y_{t_0}^* - y_{t_0-1}^*\right\|^2 \\
&= 2\left\|\mathcal{P}_{\mathcal{Y}}\left(y_{t_0-1} + \gamma_{t_0-1}\nabla_y f(x_{t_0-1}, y_{t_0-1})\right) - y_{t_0-1}^*\right\|^2 + 2\left\|y_{t_0}^* - y_{t_0-1}^*\right\|^2 \\
&\leq 2\left\|y_{t_0-1} + \gamma_{t_0-1}\nabla_y f(x_{t_0-1}, y_{t_0-1}) - y_{t_0-1}^*\right\|^2 + 2\left\|y_{t_0}^* - y_{t_0-1}^*\right\|^2 \\
&\leq 4\left(\left\|y_{t_0-1} - y_{t_0-1}^*\right\|^2 + \gamma_{t_0-1}^2\|\nabla_y f(x_{t_0-1}, y_{t_0-1})\|^2\right) + 2\left\|y_{t_0}^* - y_{t_0-1}^*\right\|^2 \\
&\leq 4\left(\frac{1}{\mu^2}\|\nabla_y f(x_{t_0-1}, y_{t_0-1})\|^2 + \gamma_{t_0-1}^2\|\nabla_y f(x_{t_0-1}, y_{t_0-1})\|^2\right) + 2\left\|y_{t_0}^* - y_{t_0-1}^*\right\|^2 \\
&= 4\left(\frac{1}{\mu^2} + \gamma_{t_0-1}^2\right)\|\nabla_y f(x_{t_0-1}, y_{t_0-1})\|^2 + 2\left\|y_{t_0}^* - y_{t_0-1}^*\right\|^2 \\
&\leq 4\left(\frac{1}{\mu^2} + \gamma_0^2\right) v_{t_0}^y + 2\left\|y_{t_0}^* - y_{t_0-1}^*\right\|^2 \\
&\leq 4\left(\frac{1}{\mu^2} + \frac{\eta^y}{\left(v_{t_0}^y\right)^\beta}\right) c_3^{1/\beta} + 2\left\|y_{t_0}^* - y_{t_0-1}^*\right\|^2 \\
&\leq 4\left(\frac{1}{\mu^2} + \frac{\eta^y}{\left(v_{t_0}^y\right)^\beta}\right) c_3^{1/\beta} + 2\kappa^2\|x_{t_0} - x_{t_0-1}\|^2 \\
&\leq 4\left(\frac{1}{\mu^2} + \frac{\eta^y}{\left(v_{t_0}^y\right)^\beta}\right) c_3^{1/\beta} + 2\kappa^2\eta_{t_0-1}^2\|\nabla_x f(x_{t_0-1}, y_{t_0-1})\|^2 \\
&\leq 4\left(\frac{1}{\mu^2} + \frac{\eta^y}{\left(v_{t_0}^y\right)^\beta}\right) c_3^{1/\beta} + \frac{2\kappa^2\left(v_{t+1}^y\right)^\beta}{\left(v_0^y\right)^\beta}\eta_{t_0-1}^2\|\nabla_x f(x_{t_0-1}, y_{t_0-1})\|^2.
\end{aligned}$$

Combined with Equation (7), we have

$$\begin{aligned}
&\sum_{t=t_0}^{T-1} \frac{\eta^y}{\left(v_{t+1}^y\right)^\beta}\|\nabla_y f(x_t, y_t)\|^2 \\
&\leq \underbrace{4(\mu+l)\left(\frac{1}{\mu^2} + \frac{\eta^y}{\left(v_{t_0}^y\right)^\beta}\right) c_3^{1/\beta}}_{c_4} + \underbrace{(\mu+l)\left(\frac{2\kappa^2}{\left(v_0^y\right)^\alpha} + \frac{(\mu+l)\kappa^2}{\eta^y \mu l}\right)}_{c_5} \sum_{t=t_0-1}^{T-1} \left(v_{t+1}^y\right)^\beta \eta_t^2\|\nabla_x f(x_t, y_t)\|^2.
\end{aligned}$$

By adding terms from 0 to $t_0 - 1$ and $\frac{\eta^y v_0^y}{\left(v_0^y\right)^\beta}$ from both sides, we have

$$\begin{aligned}
&\frac{\eta^y v_0^y}{\left(v_0^y\right)^\beta} + \sum_{t=0}^{T-1} \frac{\eta^y}{\left(v_{t+1}^y\right)^\beta}\|\nabla_y f(x_t, y_t)\|^2 \\
&\leq c_4 + \frac{\eta^y v_0^y}{\left(v_0^y\right)^\beta} + c_5 \sum_{t=0}^{T-1} \left(v_{t+1}^y\right)^\beta \eta_t^2\|\nabla_x f(x_t, y_t)\|^2 + \sum_{t=t=0}^{t_0-1} \frac{\eta^y}{\left(v_{t+1}^y\right)^\beta}\|\nabla_y f(x_t, y_t)\|^2
\end{aligned}$$

$$
\begin{aligned}
&\leq c_4 + \frac{\eta^y v_0^y}{(v_0^y)^\beta} + c_5 \sum_{t=0}^{T-1} \left(v_{t+1}^y\right)^\beta \eta_t^2 \|\nabla_x f(x_t, y_t)\|^2 + \frac{\eta^y v_0^y}{(v_0^y)^\beta} + \sum_{t=t=0}^{t_0-1} \frac{\eta^y}{\left(v_{t+1}^y\right)^\beta} \|\nabla_y f(x_t, y_t)\|^2 \\
&\leq c_4 + \frac{\eta^y v_0^y}{(v_0^y)^\beta} + c_5 \sum_{t=0}^{T-1} \left(v_{t+1}^y\right)^\beta \eta_t^2 \|\nabla_x f(x_t, y_t)\|^2 + \frac{\eta^y}{1-\beta} v_{t_0}^{1-\beta} \\
&\leq c_4 + \frac{\eta^y v_0^y}{(v_0^y)^\beta} + c_5 \sum_{t=0}^{T-1} \left(v_{t+1}^y\right)^\beta \eta_t^2 \|\nabla_x f(x_t, y_t)\|^2 + \frac{\eta^y c_3^{\frac{1-\beta}{\beta}}}{1-\beta} \\
&= c_4 + \frac{\eta^y v_0^y}{(v_0^y)^\beta} + \frac{\eta^y c_3^{\frac{1-\beta}{\beta}}}{1-\beta} + c_5 \left(\eta^x\right)^2 \sum_{t=0}^{T-1} \frac{\left(v_{t+1}^y\right)^\beta}{\max\left\{v_{t+1}^x, v_{t+1}^y\right\}^{2\alpha}} \|\nabla_x f(x_t, y_t)\|^2 \\
&= c_4 + \frac{\eta^y v_0^y}{(v_0^y)^\beta} + \frac{\eta^y c_3^{\frac{1-\beta}{\beta}}}{1-\beta} + c_5 \left(\eta^x\right)^2 \sum_{t=0}^{T-1} \frac{1}{\left(v_{t+1}^x\right)^{2\alpha-\beta}} \|\nabla_x f(x_t, y_t)\|^2 \\
&\leq \mathcal{O}\left(\left(v_T^x\right)^{\max\{1-2\alpha+\beta, 0\}}\right).
\end{aligned}
$$

The LHS can be bounded by $\left(v_T^y\right)^{1-\beta}$ by Lemma B.1. Then we get two useful inequalities from above:

$$
\begin{cases}
\sum_{t=0}^{T-1} \gamma_t \|\nabla_y f(x_t, y_t)\|^2 \leq \mathcal{O}\left(\left(v_T^x\right)^{\max\{1-2\alpha+\beta, 0\}}\right) \\
v_T^y \leq \mathcal{O}\left(\left(v_T^x\right)^{\max\{\frac{1-2\alpha+\beta}{1-\beta}, 0\}}\right).
\end{cases}
\tag{8}
$$

Now bring it back to Equation (6), we get

$$
\sum_{t=0}^{T-1} \eta_t \|\nabla_x f(x_t, y_t)\|^2 \leq \mathcal{O}\left(\left(v_T^x\right)^{\max\{1-2\alpha, 0\}} + \left(v_T^x\right)^{\max\{1-2\alpha+\beta, 0\}}\right) \leq \mathcal{O}\left(\left(v_T^x\right)^{\max\{1-2\alpha+\beta, 0\}}\right).
$$

For the LHS, we have

$$
\sum_{t=0}^{T-1} \eta_t \|\nabla_x f(x_t, y_t)\|^2 = \sum_{t=0}^{T-1} \frac{\eta^x}{\max\left\{v_{t+1}^x, v_{t+1}^y\right\}^\alpha} \|\nabla_x f(x_t, y_t)\|^2 \geq \frac{\eta^x}{\max\left\{v_T^x, v_T^y\right\}^\alpha} \sum_{t=0}^{T-1} \|\nabla_x f(x_t, y_t)\|^2
$$

Now we consider two cases: 1) if $v_T^y \leq v_T^x$, then

$$
\sum_{t=0}^{T-1} \|\nabla_x f(x_t, y_t)\|^2 \leq \mathcal{O}\left(\left(v_T^x\right)^{\max\{1-\alpha+\beta, 0\}}\right),
$$

or 2) if $v_T^y \leq v_T^x$, then

$$
\sum_{t=0}^{T-1} \|\nabla_x f(x_t, y_t)\|^2 \leq \mathcal{O}\left(\left(v_T^x\right)^{\max\left\{\frac{(1-2\alpha+\beta)\alpha}{1-\beta} + 1 - 2\alpha + \beta, 0\right\}}\right).
$$

Note that since $\alpha > \beta$, we have

$$
\frac{(1-2\alpha+\beta)\alpha}{1-\beta} + 1 - 2\alpha + \beta \leq \frac{(1-\alpha)\alpha}{1-\beta} + 1 - \alpha = 1 + \frac{\alpha(\beta-\alpha)}{1-\beta} < 1.
$$

In both two cases, the order of RHS is strictly less than 1 while the order of LHS is 1. Therefore, we conclude that

$$
\sum_{t=0}^{T-1} \|\nabla_x f(x_t, y_t)\|^2 \leq \mathcal{O}(1).
$$

Because of Equation (8), we also have

$$
\sum_{t=0}^{T-1} \|\nabla_y f(x_t, y_t)\|^2 \leq \mathcal{O}(1).
$$

□

C.2 INTERMEDIATE LEMMAS FOR THEOREM 3.2

Lemma C.1. *Under the same setting as Theorem 3.2, if for $t = t_0$ to $t_1 - 1$ and any $\lambda_t > 0$, S_t,*

$$\left\|y_{t+1} - y_{t+1}^*\right\|^2 \leq (1+\lambda_t)\left\|y_{t+1} - y_t^*\right\|^2 + S_t,$$

then we have

$$\mathbb{E}\left[\sum_{t=t_0}^{t_1-1}\left(f(x_t, y_t^*) - f(x_t, y_t)\right)\right] \leq \mathbb{E}\left[\sum_{t=t_0+1}^{t_1-1}\left(\frac{1-\gamma_t\mu}{2\gamma_t}\|y_t - y_t^*\|^2 - \frac{1}{2\gamma_t(1+\lambda_t)}\left\|y_{t+1} - y_{t+1}^*\right\|^2\right)\right] + \mathbb{E}\left[\sum_{t=t_0}^{t_1-1}\frac{\gamma_t}{2}\left\|\nabla_y \widetilde{f}(x_t, y_t)\right\|^2\right] + \mathbb{E}\left[\sum_{t=t_0}^{t_1-1}\frac{S_t}{2\gamma_t(1+\lambda_t)}\right].$$

Proof. Letting $\lambda_t := \frac{\mu\eta^y}{2\left(v_{t+1}^y\right)^\beta}$, we have

$$\begin{aligned}
&\left\|y_{t+1} - y_{t+1}^*\right\|^2 \\
&\leq (1+\lambda_t)\|y_{t+1} - y_t^*\|^2 + S_t \\
&= (1+\lambda_t)\left\|\mathcal{P}_{\mathcal{Y}}\left(y_t + \gamma_t\nabla_y\widetilde{f}(x_t, y_t)\right) - y_t^*\right\|^2 + S_t \\
&\leq (1+\lambda_t)\left\|y_t + \gamma_t\nabla_y\widetilde{f}(x_t, y_t) - y_t^*\right\|^2 + S_t \\
&= (1+\lambda_t)\left(\|y_t - y_t^*\|^2 + \gamma_t^2\left\|\nabla_y\widetilde{f}(x_t, y_t)\right\|^2 + 2\gamma_t\left\langle\nabla_y\widetilde{f}(x_t, y_t), y_t - y_t^*\right\rangle\right) + S_t \\
&= (1+\lambda_t)\Bigg(\|y_t - y_t^*\|^2 + \gamma_t^2\left\|\nabla_y\widetilde{f}(x_t, y_t)\right\|^2 + 2\gamma_t\left\langle\nabla_y\widetilde{f}(x_t, y_t), y_t - y_t^*\right\rangle \\
&\quad + \gamma_t\mu\|y_t - y_t^*\|^2 - \gamma_t\mu\|y_t - y_t^*\|^2\Bigg) + S_t
\end{aligned}$$

By multiplying $\frac{1}{\gamma_t(1+\lambda_t)}$ and rearranging the terms, we can get

$$\begin{aligned}
&2\left\langle\nabla_y\widetilde{f}(x_t, y_t), y_t^* - y_t\right\rangle - \mu\|y_t - y_t^*\|^2 \\
&\leq \frac{1-\gamma_t\mu}{\gamma_t}\|y_t - y_t^*\|^2 - \frac{1}{\gamma_t(1+\lambda_t)}\left\|y_{t+1} - y_{t+1}^*\right\|^2 + \gamma_t\left\|\nabla_y\widetilde{f}(x_t, y_t)\right\|^2 + \frac{S_t}{\gamma_t(1+\lambda_t)}.
\end{aligned}$$

By telescoping from $t = t_0$ to $t_1 - 1$, we have

$$\begin{aligned}
&\sum_{t=t_0}^{t_1-1}\left(\left\langle\nabla_y\widetilde{f}(x_t, y_t), y_t^* - y_t\right\rangle - \frac{\mu}{2}\|y_t - y_t^*\|^2\right) \\
&\leq \sum_{t=t_0+1}^{t_1-1}\left(\frac{1-\gamma_t\mu}{2\gamma_t}\|y_t - y_t^*\|^2 - \frac{1}{2\gamma_t(1+\lambda_t)}\left\|y_{t+1} - y_{t+1}^*\right\|^2\right) + \sum_{t=t_0}^{t_1-1}\frac{\gamma_t}{2}\left\|\nabla_y\widetilde{f}(x_t, y_t)\right\|^2 \\
&\quad + \sum_{t=t_0}^{t_1-1}\frac{S_t}{2\gamma_t(1+\lambda_t)}.
\end{aligned}$$

Now we take the expectation and get

$$\begin{aligned}
\mathbb{E}\left[\text{LHS}\right] &\geq \mathbb{E}\left[\sum_{t=t_0}^{t_1-1}\mathbb{E}_{\xi_t^y}\left[\left(\left\langle\nabla_y\widetilde{f}(x_t, y_t), y_t^* - y_t\right\rangle - \frac{\mu}{2}\|y_t - y_t^*\|^2\right)\right]\right] \\
&= \mathbb{E}\left[\sum_{t=t_0}^{t_1-1}\left(\left\langle\nabla_y f(x_t, y_t), y_t^* - y_t\right\rangle - \frac{\mu}{2}\|y_t - y_t^*\|^2\right)\right]
\end{aligned}$$

$$\geq \mathbb{E}\left[\sum_{t=t_0}^{t_1-1}\left(f(x_t,y_t^*)-f(x_t,y_t)\right)\right],$$

where we used strong-concavity in the last inequality.

□

Lemma C.2. *Under the same setting as Theorem 3.2, if $v_{t+1}^y \leq C$ for $t=0,...,t_0-1$, then we have*

$$\begin{aligned}
&\mathbb{E}\left[\sum_{t=0}^{t_0-1}\left(f(x_t,y_t^*)-f(x_t,y_t)\right)\right]\\
&\leq \mathbb{E}\left[\sum_{t=0}^{t_0-1}\left(\frac{1-\gamma_t\mu}{2\gamma_t}\|y_t-y_t^*\|^2-\frac{1}{\gamma_t(2+\mu\gamma_t)}\left\|y_{t+1}-y_{t+1}^*\right\|^2\right)\right]+\mathbb{E}\left[\sum_{t=0}^{t_0-1}\frac{\gamma_t}{2}\left\|\nabla_y\widetilde{f}(x_t,y_t)\right\|^2\right]\\
&\quad+\frac{\kappa^2\left(\mu\eta^yC^\beta+2C^{2\beta}\right)\left(\eta^x\right)^2}{2\mu\left(\eta^y\right)^2}\mathbb{E}\left[\frac{1+\log v_{t_0}^x-\log v_0^x}{\left(v_0^x\right)^{2\alpha-1}}\cdot\mathbf{1}_{\alpha\geq 0.5}+\frac{\left(v_{t_0}^x\right)^{1-2\alpha}}{1-2\alpha}\cdot\mathbf{1}_{\alpha<0.5}\right],
\end{aligned}$$

where $\mathbf{1}$ *is the indicator function.*

Proof. By Young's inequality, we have

$$\left\|y_{t+1}-y_{t+1}^*\right\|^2\leq(1+\lambda_t)\|y_{t+1}-y_t^*\|^2+\left(1+\frac{1}{\lambda_t}\right)\left\|y_{t+1}^*-y_t^*\right\|^2.$$

Then letting $\lambda_t=\frac{\mu\gamma_t}{2}$ and by Lemma C.1, we have

$$\begin{aligned}
&\mathbb{E}\left[\sum_{t=0}^{t_0-1}\left(f(x_t,y_t^*)-f(x_t,y_t)\right)\right]\\
&\leq \mathbb{E}\left[\sum_{t=0}^{t_0-1}\left(\frac{1-\gamma_t\mu}{2\gamma_t}\|y_t-y_t^*\|^2-\frac{1}{\gamma_t(2+\mu\gamma_t)}\left\|y_{t+1}-y_{t+1}^*\right\|^2\right)\right]\\
&\quad+\mathbb{E}\left[\sum_{t=0}^{t_0-1}\frac{\gamma_t}{2}\left\|\nabla_y\widetilde{f}(x_t,y_t)\right\|^2\right]+\mathbb{E}\left[\sum_{t=0}^{t_0-1}\frac{\left(1+\frac{2}{\mu\gamma_t}\right)}{\gamma_t(2+\mu\gamma_t)}\left\|y_{t+1}^*-y_t^*\right\|^2\right].
\end{aligned}$$

We now remain to bound the last term:

$$\begin{aligned}
&\mathbb{E}\left[\sum_{t=0}^{t_0-1}\frac{\left(1+\frac{2}{\mu\gamma_t}\right)}{\gamma_t(2+\mu\gamma_t)}\left\|y_{t+1}^*-y_t^*\right\|^2\right]\\
&\leq \mathbb{E}\left[\sum_{t=0}^{t_0-1}\frac{\left(1+\frac{2}{\mu\gamma_t}\right)}{2\gamma_t}\left\|y_{t+1}^*-y_t^*\right\|^2\right]\\
&=\mathbb{E}\left[\sum_{t=0}^{t_0-1}\frac{\mu\eta^y\left(v_{t+1}^y\right)^\beta+2\left(v_{t+1}^y\right)^{2\beta}}{2\mu\left(\eta^y\right)^2}\left\|y_{t+1}^*-y_t^*\right\|^2\right]\\
&\leq\frac{\mu\eta^yC^\beta+2C^{2\beta}}{2\mu\left(\eta^y\right)^2}\mathbb{E}\left[\sum_{t=0}^{t_0-1}\left\|y_{t+1}^*-y_t^*\right\|^2\right].
\end{aligned}$$

By Lemma B.2 we have

$$\begin{aligned}
\sum_{t=0}^{t_0-1}\left\|y_{t+1}^*-y_t^*\right\|^2&\leq\kappa^2\sum_{t=0}^{t_0-1}\|x_{t+1}-x_t\|^2\\
&=\kappa^2\sum_{t=0}^{t_0-1}\eta_t^2\left\|\nabla_x\widetilde{f}(x_t,y_t)\right\|^2
\end{aligned}$$

$$
\begin{aligned}
&= \kappa^2 \left(\eta^x\right)^2 \sum_{t=0}^{t_0-1} \frac{1}{\max\left\{v_{t+1}^x, v_{t+1}^y\right\}^{2\alpha}} \left\|\nabla_x \widetilde{f}(x_t, y_t)\right\|^2 \\
&\leq \kappa^2 \left(\eta^x\right)^2 \sum_{t=0}^{t_0-1} \frac{1}{\left(v_{t+1}^x\right)^{2\alpha}} \left\|\nabla_x \widetilde{f}(x_t, y_t)\right\|^2 \\
&\leq \kappa^2 \left(\eta^x\right)^2 \left(\frac{v_0^x}{\left(v_0^x\right)^{2\alpha}} + \sum_{t=0}^{t_0-1} \frac{1}{\left(v_{t+1}^x\right)^{2\alpha}} \left\|\nabla_x \widetilde{f}(x_t, y_t)\right\|^2\right) \\
&\leq \kappa^2 \left(\eta^x\right)^2 \left(\frac{1 + \log v_{t_0}^x - \log v_0^x}{\left(v_0^x\right)^{2\alpha-1}} \cdot \mathbf{1}_{\alpha \geq 0.5} + \frac{\left(v_{t_0}^x\right)^{1-2\alpha}}{1-2\alpha} \cdot \mathbf{1}_{\alpha < 0.5}\right)
\end{aligned}
$$

where we applied Lemma B.1 in the last inequality. Bringing back this result, we finish the proof.

□

Lemma C.3. *Under the same setting as Theorem 3.2, if t_0 is the first iteration such that $v_{t_0+1}^y > C$, then we have*

$$
\begin{aligned}
&\mathbb{E}\left[\sum_{t=t_0}^{T-1} \left(f(x_t, y_t^*) - f(x_t, y_t)\right)\right] \\
&\leq \mathbb{E}\left[\sum_{t=t_0}^{T-1} \left(\frac{1-\gamma_t\mu}{2\gamma_t}\|y_t - y_t^*\|^2 - \frac{1}{\gamma_t(2+\mu\gamma_t)}\left\|y_{t+1} - y_{t+1}^*\right\|^2\right)\right] + \mathbb{E}\left[\sum_{t=t_0}^{T-1} \frac{\gamma_t}{2}\left\|\nabla_y \widetilde{f}(x_t, y_t)\right\|^2\right] \\
&\quad + \left(\kappa^2 + \frac{\widehat{L}^2 G^2 \left(\eta^x\right)^2}{\mu\eta^y \left(v_0^y\right)^{2\alpha-\beta}}\right) \frac{\left(\eta^x\right)^2}{2(1-\alpha)\eta^y \left(v_0^y\right)^{\alpha-\beta}} \mathbb{E}\left[\left(v_T^x\right)^{1-\alpha}\right] \\
&\quad + \frac{2\kappa^2 \left(\eta^x\right)^2}{\mu \left(\eta^y\right)^2 C^{2\alpha-2\beta}} \mathbb{E}\left[\sum_{t=t_0}^{T-1} \|\nabla_x f(x_t, y_t)\|^2\right] + \left(\frac{1}{\mu} + \frac{\eta^y}{\left(v_0^y\right)^\beta}\right) \frac{4\kappa\eta^x G^2}{\eta^y \left(v_0^y\right)^\alpha} \mathbb{E}\left[\left(v_T^y\right)^\beta\right].
\end{aligned}
$$

Proof. By the Lipschitzness of $y^*(\cdot)$ as in Lemma B.2, we have

$$
\begin{aligned}
\left\|y_{t+1} - y_{t+1}^*\right\|^2 &= \|y_{t+1} - y_t^*\|^2 + \left\|y_t^* - y_{t+1}^*\right\|^2 + 2\left\langle y_{t+1} - y_t^*, y_t^* - y_{t+1}^*\right\rangle \\
&\leq \|y_{t+1} - y_t^*\|^2 + \kappa^2\eta_t^2\left\|\nabla_x \widetilde{f}(x_t, y_t)\right\|^2 + 2\left\langle y_{t+1} - y_t^*, y_t^* - y_{t+1}^*\right\rangle \\
&\leq \|y_{t+1} - y_t^*\|^2 + \kappa^2\eta_t^2\left\|\nabla_x \widetilde{f}(x_t, y_t)\right\|^2 \underbrace{-2\left(y_{t+1} - y_t^*\right)^\intercal \nabla y^*(x_t)\left(x_{t+1} - x_t\right)}_{\text{(C)}} \\
&\quad + \underbrace{2\left(y_{t+1} - y_t^*\right)^\intercal \left(y_t^* - y_{t+1}^* + \nabla y^*(x_t)\left(x_{t+1} - x_t\right)\right)}_{\text{(D)}}.
\end{aligned}
$$

For Term (C), by the Cauchy-Schwarz and Lipschitzness of $y^*(\cdot)$,

$$
\begin{aligned}
&-2\left(y_{t+1} - y_t^*\right)^\intercal \nabla y^*(x_t)\left(x_{t+1} - x_t\right) \\
&= 2\eta_t\left(y_{t+1} - y_t^*\right)^\intercal \nabla y^*(x_t)\nabla_x f(x_t, y_t) + 2\eta_t\left(y_{t+1} - y_t^*\right)^\intercal \nabla y^*(x_t)\left(\nabla_x \widetilde{f}(x_t, y_t) - \nabla_x f(x_t, y_t)\right) \\
&\leq 2\eta_t\|y_{t+1} - y_t^*\|\|\nabla y^*(x_t)\|\|\nabla_x f(x_t, y_t)\| + 2\eta_t\left(y_{t+1} - y_t^*\right)^\intercal \nabla y^*(x_t)\left(\nabla_x \widetilde{f}(x_t, y_t) - \nabla_x f(x_t, y_t)\right) \\
&\leq 2\|y_{t+1} - y_t^*\|\kappa\eta_t\|\nabla_x f(x_t, y_t)\| + 2\eta_t\left(y_{t+1} - y_t^*\right)^\intercal \nabla y^*(x_t)\left(\nabla_x \widetilde{f}(x_t, y_t) - \nabla_x f(x_t, y_t)\right) \\
&\leq \lambda_t\|y_{t+1} - y_t^*\|^2 + \frac{\kappa^2\eta_t^2}{\lambda_t}\|\nabla_x f(x_t, y_t)\|^2 + 2\eta_t\left(y_{t+1} - y_t^*\right)^\intercal \nabla y^*(x_t)\left(\nabla_x \widetilde{f}(x_t, y_t) - \nabla_x f(x_t, y_t)\right),
\end{aligned}
$$

where we used Young's inequality in the last step and $\lambda_t > 0$ will be determined later.

For Term (D), according to Cauchy-Schwarz and the smoothness of $y^*(\cdot)$ as shown in Lemma B.3,

$$
2\left(y_{t+1} - y_t^*\right)^\intercal \left(y_t^* - y_{t+1}^* + \nabla y^*(x_t)\left(x_{t+1} - x_t\right)\right)
$$

$$
\begin{aligned}
&\leq 2\|y_{t+1} - y_t^*\| \left\| y_t^* - y_{t+1}^* + \nabla y^*(x_t)\,(x_{t+1} - x_t) \right\| \\
&\leq 2\|y_{t+1} - y_t^*\| \cdot \frac{\widehat{L}}{2} \|x_{t+1} - x_t\|^2 \\
&= \widehat{L}\eta_t^2 \|y_{t+1} - y_t^*\| \left\| \nabla_x \widetilde{f}(x_t, y_t) \right\|^2 \\
&\leq \widehat{L}\eta_t^2 \|y_{t+1} - y_t^*\| G \cdot \left\| \nabla_x \widetilde{f}(x_t, y_t) \right\| \\
&\leq \frac{\tau \widehat{L} G^2 \eta_t^2}{2} \|y_{t+1} - y_t^*\|^2 + \frac{\widehat{L}\eta_t^2}{2\tau} \left\| \nabla_x \widetilde{f}(x_t, y_t) \right\|^2,
\end{aligned}
$$

where in the last step we used Young's inequality and $\tau > 0$.

Therefore, in total, we have

$$
\begin{aligned}
\left\| y_{t+1} - y_{t+1}^* \right\|^2 \leq{}& \left(1 + \lambda_t + \frac{\tau \widehat{L} G^2 \eta_t^2}{2} \right) \|y_{t+1} - y_t^*\|^2 + \left(\kappa^2 + \frac{\widehat{L}}{2\tau} \right) \eta_t^2 \left\| \nabla_x \widetilde{f}(x_t, y_t) \right\|^2 \\
&+ \frac{\kappa^2 \eta_t^2}{\lambda_t} \|\nabla_x f(x_t, y_t)\|^2 + 2\eta_t \left(y_{t+1} - y_t^* \right)^\intercal \nabla y^*(x_t) \left(\nabla_x \widetilde{f}(x_t, y_t) - \nabla_x f(x_t, y_t) \right).
\end{aligned}
$$

Note that we can upper bound η_t by

$$
\eta_t = \frac{\eta^x}{\max\left\{ v_{t+1}^x, v_{t+1}^y \right\}^\alpha} \leq \frac{\eta^x}{\left(v_{t+1}^y \right)^\alpha} \leq \frac{\eta^x}{\left(v_0^y \right)^\alpha},
$$

and

$$
\eta_t \leq \frac{\eta^x}{\left(v_{t+1}^y \right)^\alpha} = \frac{\eta^x}{\left(v_{t+1}^y \right)^{\alpha - \beta} \left(v_{t+1}^y \right)^\beta} \leq \frac{\eta^x}{\left(v_0^y \right)^{\alpha - \beta} \left(v_{t+1}^y \right)^\beta},
$$

which, plugged into the previous result, implies

$$
\begin{aligned}
\left\| y_{t+1} - y_{t+1}^* \right\|^2 \leq{}& \left(1 + \lambda_t + \frac{\tau \widehat{L} G^2 \left(\eta^x \right)^2}{2 \left(v_0^y \right)^{2\alpha - \beta} \left(v_{t+1}^y \right)^\beta} \right) \|y_{t+1} - y_t^*\|^2 + \left(\kappa^2 + \frac{\widehat{L}}{2\tau} \right) \eta_t^2 \left\| \nabla_x \widetilde{f}(x_t, y_t) \right\|^2 \\
&+ \frac{\kappa^2 \eta_t^2}{\lambda_t} \|\nabla_x f(x_t, y_t)\|^2 + 2\eta_t \left(y_{t+1} - y_t^* \right)^\intercal \nabla y^*(x_t) \left(\nabla_x \widetilde{f}(x_t, y_t) - \nabla_x f(x_t, y_t) \right).
\end{aligned}
$$

Now we choose $\lambda_t = \frac{\mu \eta^y}{4 \left(v_{t+1}^y \right)^\beta}$ and $\tau = \frac{\mu \eta^y \left(v_0^y \right)^{2\alpha - \beta}}{2 \widehat{L} G^2 \left(\eta^x \right)^2}$, and get

$$
\begin{aligned}
&\left\| y_{t+1} - y_{t+1}^* \right\|^2 \\
\leq{}& \left(1 + \frac{\mu \eta^y}{2 \left(v_{t+1}^y \right)^\beta} \right) \|y_{t+1} - y_t^*\|^2 + \left(\kappa^2 + \frac{\widehat{L}^2 G^2 \left(\eta^x \right)^2}{\mu \eta^y \left(v_0^y \right)^{2\alpha - \beta}} \right) \eta_t^2 \left\| \nabla_x \widetilde{f}(x_t, y_t) \right\|^2 \\
&+ \frac{4\kappa^2 \left(v_{t+1}^y \right)^\beta \eta_t^2}{\mu \eta^y} \|\nabla_x f(x_t, y_t)\|^2 + 2\eta_t \left(y_{t+1} - y_t^* \right)^\intercal \nabla y^*(x_t) \left(\nabla_x \widetilde{f}(x_t, y_t) - \nabla_x f(x_t, y_t) \right).
\end{aligned}
$$

Then Lemma C.1 gives us

$$
\begin{aligned}
&\mathbb{E} \left[\sum_{t=t_0}^{T-1} \left(f(x_t, y_t^*) - f(x_t, y_t) \right) \right] \\
\leq{}& \mathbb{E} \left[\sum_{t=t_0}^{T-1} \left(\frac{1 - \gamma_t \mu}{2\gamma_t} \|y_t - y_t^*\|^2 - \frac{1}{\gamma_t (2 + \mu\gamma_t)} \left\| y_{t+1} - y_{t+1}^* \right\|^2 \right) \right] + \mathbb{E} \left[\sum_{t=t_0}^{T-1} \frac{\gamma_t}{2} \left\| \nabla_y \widetilde{f}(x_t, y_t) \right\|^2 \right] \\
&+ \underbrace{\mathbb{E} \left[\sum_{t=t_0}^{T-1} \frac{1}{\gamma_t (2 + \mu\gamma_t)} \left(\kappa^2 + \frac{\widehat{L}^2 G^2 \left(\eta^x \right)^2}{\mu \eta^y \left(v_0^y \right)^{2\alpha - \beta}} \right) \eta_t^2 \left\| \nabla_x \widetilde{f}(x_t, y_t) \right\|^2 \right]}_{\text{(E)}}
\end{aligned}
$$

$$+\underbrace{\mathbb{E}\left[\sum_{t=t_0}^{T-1}\frac{4\kappa^2\left(v_{t+1}^y\right)^\beta\eta_t^2}{\gamma_t(2+\mu\gamma_t)\mu\eta^y}\|\nabla_x f(x_t,y_t)\|^2\right]}_{\text{(F)}}$$

$$+\underbrace{\mathbb{E}\left[\sum_{t=t_0}^{T-1}\frac{2\eta_t}{\gamma_t(2+\mu\gamma_t)}\left(y_{t+1}-y_t^*\right)^\intercal\nabla y^*(x_t)\left(\nabla_x\widetilde{f}(x_t,y_t)-\nabla_x f(x_t,y_t)\right)\right]}_{\text{(G)}}$$

Now we proceed to bound each term.

Term (E)

$$\begin{aligned}
\text{Term (E)} &\le \left(\kappa^2+\frac{\widehat{L}^2G^2\left(\eta^x\right)^2}{\mu\eta^y\left(v_0^y\right)^{2\alpha-\beta}}\right)\mathbb{E}\left[\sum_{t=t_0}^{T-1}\frac{\eta_t^2}{2\gamma_t}\left\|\nabla_x\widetilde{f}(x_t,y_t)\right\|^2\right]\\
&= \left(\kappa^2+\frac{\widehat{L}^2G^2\left(\eta^x\right)^2}{\mu\eta^y\left(v_0^y\right)^{2\alpha-\beta}}\right)\mathbb{E}\left[\sum_{t=t_0}^{T-1}\frac{\left(\eta^x\right)^2\left(v_{t+1}^y\right)^\beta}{2\eta^y\max\left\{v_{t+1}^x,v_{t+1}^y\right\}^{2\alpha}}\left\|\nabla_x\widetilde{f}(x_t,y_t)\right\|^2\right]\\
&\le \left(\kappa^2+\frac{\widehat{L}^2G^2\left(\eta^x\right)^2}{\mu\eta^y\left(v_0^y\right)^{2\alpha-\beta}}\right)\mathbb{E}\left[\sum_{t=t_0}^{T-1}\frac{\left(\eta^x\right)^2\left(v_{t+1}^y\right)^\beta}{2\eta^y\left(v_{t+1}^y\right)^\beta\left(v_{t+1}^y\right)^{\alpha-\beta}\left(v_{t+1}^x\right)^\alpha}\left\|\nabla_x\widetilde{f}(x_t,y_t)\right\|^2\right]\\
&\le \left(\kappa^2+\frac{\widehat{L}^2G^2\left(\eta^x\right)^2}{\mu\eta^y\left(v_0^y\right)^{2\alpha-\beta}}\right)\mathbb{E}\left[\sum_{t=t_0}^{T-1}\frac{\left(\eta^x\right)^2}{2\eta^y\left(v_0^y\right)^{\alpha-\beta}\left(v_{t+1}^x\right)^\alpha}\left\|\nabla_x\widetilde{f}(x_t,y_t)\right\|^2\right]\\
&\le \left(\kappa^2+\frac{\widehat{L}^2G^2\left(\eta^x\right)^2}{\mu\eta^y\left(v_0^y\right)^{2\alpha-\beta}}\right)\mathbb{E}\left[\frac{\left(\eta^x\right)^2}{2\eta^y\left(v_0^y\right)^{\alpha-\beta}}\left(\frac{v_0^x}{\left(v_0^x\right)^\alpha}+\sum_{t=0}^{T-1}\frac{1}{\left(v_{t+1}^x\right)^\alpha}\left\|\nabla_x\widetilde{f}(x_t,y_t)\right\|^2\right)\right]\\
&\le \left(\kappa^2+\frac{\widehat{L}^2G^2\left(\eta^x\right)^2}{\mu\eta^y\left(v_0^y\right)^{2\alpha-\beta}}\right)\frac{\left(\eta^x\right)^2}{2(1-\alpha)\eta^y\left(v_0^y\right)^{\alpha-\beta}}\mathbb{E}\left[\left(v_T^x\right)^{1-\alpha}\right],
\end{aligned}$$

where we used Lemma B.1 in the last step.

Term (F)

$$\begin{aligned}
\text{Term (F)} &\le \mathbb{E}\left[\sum_{t=t_0}^{T-1}\frac{2\kappa^2\left(v_{t+1}^y\right)^\beta\eta_t^2}{\gamma_t\mu\eta^y}\|\nabla_x f(x_t,y_t)\|^2\right]\\
&= \frac{2\kappa^2\left(\eta^x\right)^2}{\mu\left(\eta^y\right)^2}\mathbb{E}\left[\sum_{t=t_0}^{T-1}\frac{\left(v_{t+1}^y\right)^{2\beta}}{\max\left\{v_{t+1}^x,v_{t+1}^y\right\}^{2\alpha}}\|\nabla_x f(x_t,y_t)\|^2\right]\\
&\le \frac{2\kappa^2\left(\eta^x\right)^2}{\mu\left(\eta^y\right)^2}\mathbb{E}\left[\sum_{t=t_0}^{T-1}\frac{\left(v_{t+1}^y\right)^{2\beta}}{\left(v_{t+1}^y\right)^{2\alpha}}\|\nabla_x f(x_t,y_t)\|^2\right]\\
&\le \frac{2\kappa^2\left(\eta^x\right)^2}{\mu\left(\eta^y\right)^2}\mathbb{E}\left[\frac{1}{\left(v_{t_0+1}^y\right)^{2\alpha-2\beta}}\sum_{t=t_0}^{T-1}\|\nabla_x f(x_t,y_t)\|^2\right]\\
&\le \frac{2\kappa^2\left(\eta^x\right)^2}{\mu\left(\eta^y\right)^2C^{2\alpha-2\beta}}\mathbb{E}\left[\sum_{t=t_0}^{T-1}\|\nabla_x f(x_t,y_t)\|^2\right]
\end{aligned}$$

Term (G) For simplicity, denote $m_t := \frac{2}{\gamma_t(2+\mu\gamma_t)}\left(y_{t+1}-y_t^*\right)^\intercal\nabla y^*(x_t)\left(\nabla_x\widetilde{f}(x_t,y_t)-\nabla_x f(x_t,y_t)\right)$
Since $y^*(\cdot)$ is κ-Lipschitz as in Lemma B.2, $|m_t|$ can be upper bounded as

$$|m_t|\le\frac{1}{\gamma_t}\|y_{t+1}-y_t^*\|\|\nabla y^*(x_t)\|\left(\left\|\nabla_x\widetilde{f}(x_t,y_t)\right\|+\|\nabla_x f(x_t,y_t)\|\right)$$

$$
\begin{aligned}
&\leq \frac{\kappa}{\gamma_t}\|y_{t+1}-y_t^*\|\left(\left\|\nabla_x \widetilde{f}(x_t,y_t)\right\|+\|\nabla_x f(x_t,y_t)\|\right)\\
&\leq \frac{\kappa}{\gamma_t}\left\|\mathcal{P}_{\mathcal{Y}}\left(y_t+\gamma_t\nabla_y \widetilde{f}(x_t,y_t)\right)-y_t^*\right\|\left(\left\|\nabla_x \widetilde{f}(x_t,y_t)\right\|+\|\nabla_x f(x_t,y_t)\|\right)\\
&\leq \frac{\kappa}{\gamma_t}\left\|y_t+\gamma_t\nabla_y \widetilde{f}(x_t,y_t)-y_t^*\right\|\left(\left\|\nabla_x \widetilde{f}(x_t,y_t)\right\|+\|\nabla_x f(x_t,y_t)\|\right)\\
&\leq \frac{\kappa}{\gamma_t}\left(\|y_t-y_t^*\|+\left\|\gamma_t\nabla_y \widetilde{f}(x_t,y_t)\right\|\right)\left(\left\|\nabla_x \widetilde{f}(x_t,y_t)\right\|+\|\nabla_x f(x_t,y_t)\|\right)\\
&\leq \frac{\kappa}{\gamma_t}\left(\frac{1}{\mu}\|\nabla_y f(x_t,y_t)\|+\left\|\gamma_t\nabla_y \widetilde{f}(x_t,y_t)\right\|\right)\left(\left\|\nabla_x \widetilde{f}(x_t,y_t)\right\|+\|\nabla_x f(x_t,y_t)\|\right)\\
&\leq \underbrace{\frac{2G\kappa}{\gamma_{T-1}}\left(\frac{G}{\mu}+\frac{\eta^y G}{(v_0^y)^\beta}\right)}_{M}.
\end{aligned}
$$

Also note that γ_t and y_{t+1} does not depend on ξ_t^x, so $\mathbb{E}_{\xi_t^x}[m_t]=0$. Next, we look at Term (G).

$$
\begin{aligned}
\text{Term (G)} &= \mathbb{E}\left[\sum_{t=t_0}^{T-1}\eta_t m_t\right]\\
&= \mathbb{E}\left[\eta_{t_0}m_{t_0}+\sum_{t=t_0+1}^{T-1}\eta_{t-1}m_t+\sum_{t=t_0+1}^{T-1}(\eta_t-\eta_{t-1})\,m_t\right]\\
&\leq \mathbb{E}\left[\frac{\eta^x}{(v_0^y)^\alpha}M+\sum_{t=t_0+1}^{T-1}\eta_{t-1}\mathbb{E}_{\xi_t^x}[m_t]+\sum_{t=t_0+1}^{T-1}(\eta_{t-1}-\eta_t)\,(-m_t)\right]\\
&\leq \mathbb{E}\left[\frac{\eta^x}{(v_0^y)^\alpha}M+\sum_{t=t_0+1}^{T-1}(\eta_{t-1}-\eta_t)\,M\right]\\
&\leq \mathbb{E}\left[\frac{2\eta^x}{(v_0^y)^\alpha}M\right]\\
&= \left(\frac{1}{\mu}+\frac{\eta^y}{(v_0^y)^\beta}\right)\frac{4\kappa\eta^x G^2}{\eta^y\,(v_0^y)^\alpha}\mathbb{E}\left[(v_T^y)^\beta\right].
\end{aligned}
$$

Summarizing all the results, we finish the proof.

□

Lemma C.4. *Under the same setting as Theorem 3.2, we have*

$$
\begin{aligned}
&\mathbb{E}\left[\sum_{t=0}^{T-1}\left(\frac{1-\gamma_t\mu}{2\gamma_t}\|y_t-y_t^*\|^2-\frac{1}{\gamma_t(2+\mu\gamma_t)}\left\|y_{t+1}-y_{t+1}^*\right\|^2\right)\right]\\
\leq\ &\frac{(v_0^y)^\beta G^2}{2\mu^2\eta^y}+\frac{(2\beta G)^{\frac{1}{1-\beta}+2}G^2}{4\mu^{\frac{1}{1-\beta}+3}(\eta^y)^{\frac{1}{1-\beta}+2}(v_0^y)^{2-2\beta}}.
\end{aligned}
$$

Proof.

$$
\begin{aligned}
&\mathbb{E}\left[\sum_{t=0}^{T-1}\left(\frac{1-\gamma_t\mu}{2\gamma_t}\|y_t-y_t^*\|^2-\frac{1}{\gamma_t(2+\mu\gamma_t)}\left\|y_{t+1}-y_{t+1}^*\right\|^2\right)\right]\\
\leq\ &\left(\frac{(v_0^y)^\beta}{2\eta^y}-\frac{\mu}{2}\right)\|y_0-y_0^*\|^2+\frac{1}{2\eta^y}\sum_{t=1}^{T-1}\left((v_{t+1}^y)^\beta-\frac{\mu\eta^y}{2}-(v_t^y)^\beta-\frac{\mu^2(\eta^y)^2}{4(v_t^y)^\beta+2\mu\eta^y}\right)\|y_t-y_t^*\|^2\\
\leq\ &\frac{(v_0^y)^\beta G^2}{2\mu^2\eta^y}+\frac{1}{2\eta^y}\underbrace{\sum_{t=1}^{T-1}\left((v_{t+1}^y)^\beta-\frac{\mu\eta^y}{2}-(v_t^y)^\beta\right)\|y_t-y_t^*\|^2}_{\text{(H)}}.
\end{aligned}
$$

For Term (H),we will bound it using the same strategy as in (Yang et al., 2022a). The general idea is to show that $\left(v_{t+1}^y\right)^\beta - \frac{\mu\eta^y}{2} - \left(v_t^y\right)^\beta$ is positive for only a constant number of times. If the term is positive at iteration t, then we have

$$\begin{aligned}
0 &< \left(v_{t+1}^y\right)^\beta - \left(v_t^y\right)^\beta - \frac{\mu\eta^y}{2} \\
&= \left(v_t^y + \left\|\nabla_y \widetilde{f}(x_t, y_t)\right\|^2\right)^\beta - \left(v_t^y\right)^\beta - \frac{\mu\eta^y}{2} \\
&= \left(v_t^y\right)^\beta \left(1 + \frac{\left\|\nabla_y \widetilde{f}(x_t, y_t)\right\|^2}{v_t^y}\right)^\beta - \left(v_t^y\right)^\beta - \frac{\mu\eta^y}{2} \\
&\leq \left(v_t^y\right)^\beta \left(1 + \frac{\beta\left\|\nabla_y \widetilde{f}(x_t, y_t)\right\|^2}{v_t^y}\right) - \left(v_t^y\right)^\beta - \frac{\mu\eta^y}{2} \\
&= \frac{\beta\left\|\nabla_y \widetilde{f}(x_t, y_t)\right\|^2}{\left(v_t^y\right)^{1-\beta}} - \frac{\mu\eta^y}{2},
\end{aligned} \tag{9}$$

where in the last inequality we used Bernoulli's inequality. By rearranging the terms, we have the two following conditions

$$\begin{cases}
\left\|\nabla_y \widetilde{f}(x_t, y_t)\right\|^2 > \frac{\mu\eta^y}{2\beta}\left(v_t^y\right)^{1-\beta} \geq \frac{\mu\eta^y}{2\beta}\left(v_0^y\right)^{1-\beta} \\
\left(v_t^y\right)^{1-\beta} < \frac{2\beta}{\mu\eta^y}\left\|\nabla_y \widetilde{f}(x_t, y_t)\right\|^2 \leq \frac{2\beta G}{\mu\eta^y},
\end{cases}$$

This indicates that at each time the term is positive, the gradient norm must be large enough and the accumulated gradient norm, i.e., v_{t+1}^y, must be small enough. Therefore, we can have at most

$$\frac{\left(\frac{2\beta G}{\mu\eta^y}\right)^{\frac{1}{1-\beta}}}{\frac{\mu\eta^y}{2\beta}\left(v_0^y\right)^{1-\beta}}$$

constant number of iterations when the term is positive. When the term is positive, it is also upper bounded by using the result from Equation (9):

$$\begin{aligned}
\left(\left(v_{t+1}^y\right)^\beta - \frac{\mu\eta^y}{2} - \left(v_t^y\right)^\beta\right)\|y_t - y_t^*\|^2 &\leq \frac{\beta\left\|\nabla_y \widetilde{f}(x_t, y_t)\right\|^2}{\left(v_t^y\right)^{1-\beta}}\|y_t - y_t^*\|^2 \\
&\leq \frac{\beta G^2}{\left(v_0^y\right)^{1-\beta}}\|y_t - y_t^*\|^2 \\
&\leq \frac{\beta G^2}{\mu^2\left(v_0^y\right)^{1-\beta}}\|\nabla_y f(x_t, y_t)\|^2 \\
&\leq \frac{\beta G^4}{\mu^2\left(v_0^y\right)^{1-\beta}}
\end{aligned}$$

which is a constant. In total, Term (H) is bounded by

$$\frac{(2\beta G)^{\frac{1}{1-\beta}+2}\, G^2}{2\mu^{\frac{1}{1-\beta}+3}\left(\eta^y\right)^{\frac{1}{1-\beta}+1}\left(v_0^y\right)^{2-2\beta}}.$$

Bringing it back, we get the desired result. □

Lemma C.5. *Under the same setting as Theorem 3.2, for any constant C, we have*

$$
\begin{aligned}
&\mathbb{E}\left[\sum_{t=0}^{T-1}\left(f(x_t,y_t^*)-f(x_t,y_t)\right)\right]\\
\leq\ &\frac{2\kappa^2\left(\eta^x\right)^2}{\mu\left(\eta^y\right)^2C^{2\alpha-2\beta}}\mathbb{E}\left[\sum_{t=0}^{T-1}\|\nabla_x f(x_t,y_t)\|^2\right]+\frac{\eta^y}{2(1-\beta)}\mathbb{E}\left[\left(v_T^y\right)^{1-\beta}\right]\\
&+\left(\frac{G}{\mu}+\frac{\eta^yG}{\left(v_0^y\right)^\beta}\right)\frac{2\kappa\eta^x\left(G+G\right)}{\eta^y\left(v_0^y\right)^\alpha}\mathbb{E}\left[\left(v_T^y\right)^\beta\right]\\
&+\frac{\kappa^2\left(\mu\eta^yC^\beta+2C^{2\beta}\right)\left(\eta^x\right)^2}{2\mu\left(\eta^y\right)^2}\mathbb{E}\left[\frac{1+\log v_T^x-\log v_0^x}{\left(v_0^x\right)^{2\alpha-1}}\cdot\mathbf{1}_{\alpha\geq0.5}+\frac{\left(v_T^x\right)^{1-2\alpha}}{1-2\alpha}\cdot\mathbf{1}_{\alpha<0.5}\right]\\
&+\left(\kappa^2+\frac{\widehat{L}^2G^2\left(\eta^x\right)^2}{\mu\eta^y\left(v_0^y\right)^{2\alpha-\beta}}\right)\frac{\left(\eta^x\right)^2}{2(1-\alpha)\eta^y\left(v_0^y\right)^{\alpha-\beta}}\mathbb{E}\left[\left(v_T^x\right)^{1-\alpha}\right]\\
&+\frac{\left(v_0^y\right)^\beta G^2}{2\mu^2\eta^y}+\frac{\left(2\beta G\right)^{\frac{1}{1-\beta}+2}G^2}{4\mu^{\frac{1}{1-\beta}+3}\left(\eta^y\right)^{\frac{1}{1-\beta}+2}\left(v_0^y\right)^{2-2\beta}},
\end{aligned}
$$

where $\mathbf{1}$ is the indicator function.

Proof. By Lemma C.2 and Lemma C.3, we have for any constant C,

$$
\begin{aligned}
&\mathbb{E}\left[\sum_{t=0}^{T-1}\left(f(x_t,y_t^*)-f(x_t,y_t)\right)\right]\\
\leq\ &\mathbb{E}\left[\sum_{t=0}^{T-1}\left(\frac{1-\gamma_t\mu}{2\gamma_t}\|y_t-y_t^*\|^2-\frac{1}{\gamma_t(2+\mu\gamma_t)}\left\|y_{t+1}-y_{t+1}^*\right\|^2\right)\right]\\
&+\mathbb{E}\left[\sum_{t=0}^{T-1}\frac{\gamma_t}{2}\left\|\nabla_y\widetilde{f}(x_t,y_t)\right\|^2\right]+\frac{2\kappa^2\left(\eta^x\right)^2}{\mu\left(\eta^y\right)^2C^{2\alpha-2\beta}}\mathbb{E}\left[\sum_{t=0}^{T-1}\|\nabla_x f(x_t,y_t)\|^2\right]\\
&+\frac{\kappa^2\left(\mu\eta^yC^\beta+2C^{2\beta}\right)\left(\eta^x\right)^2}{2\mu\left(\eta^y\right)^2}\mathbb{E}\left[\frac{1+\log v_T^x-\log v_0^x}{\left(v_0^x\right)^{2\alpha-1}}\cdot\mathbf{1}_{\alpha\geq0.5}+\frac{\left(v_T^x\right)^{1-2\alpha}}{1-2\alpha}\cdot\mathbf{1}_{\alpha<0.5}\right]\\
&+\left(\kappa^2+\frac{\widehat{L}^2G^2\left(\eta^x\right)^2}{\mu\eta^y\left(v_0^y\right)^{2\alpha-\beta}}\right)\frac{\left(\eta^x\right)^2}{2(1-\alpha)\eta^y\left(v_0^y\right)^{\alpha-\beta}}\mathbb{E}\left[\left(v_T^x\right)^{1-\alpha}\right]\\
&+\left(\frac{1}{\mu}+\frac{\eta^y}{\left(v_0^y\right)^\beta}\right)\frac{4\kappa\eta^xG^2}{\eta^y\left(v_0^y\right)^\alpha}\mathbb{E}\left[\left(v_T^y\right)^\beta\right].
\end{aligned}
$$

The first term can be bounded by Lemma C.4. For the second term, we have

$$
\begin{aligned}
\mathbb{E}\left[\sum_{t=0}^{T-1}\frac{\gamma_t}{2}\left\|\nabla_y\widetilde{f}(x_t,y_t)\right\|^2\right]&=\mathbb{E}\left[\sum_{t=0}^{T-1}\frac{\eta^y}{2\left(v_{t+1}^y\right)^\beta}\left\|\nabla_y\widetilde{f}(x_t,y_t)\right\|^2\right]\\
&\leq\frac{\eta^y}{2}\mathbb{E}\left[\frac{v_0^y}{\left(v_0^y\right)^\beta}+\sum_{t=0}^{T-1}\frac{1}{\left(v_{t+1}^y\right)^\beta}\left\|\nabla_y\widetilde{f}(x_t,y_t)\right\|^2\right]\\
&\leq\frac{\eta^y}{2(1-\beta)}\mathbb{E}\left[\left(v_T^y\right)^{1-\beta}\right],
\end{aligned}
$$

where the last inequality follows from Lemma B.1. Then the proof is completed. □

C.3 Proof of Theorem 3.2

Proof. By smoothness of the primal function, we have

$$
\Phi(x_{t+1})-\Phi(x_t)\leq-\eta_t\left\langle\nabla\Phi(x_t),\nabla_x\widetilde{f}(x_t,y_t)\right\rangle+l\kappa\eta_t^2\left\|\nabla_x\widetilde{f}(x_t,y_t)\right\|^2.
$$

By multiplying $\frac{1}{\eta_t}$ on both sides and taking the expectation w.r.t. the noise of current iteration, we have

$$
\begin{aligned}
&\mathbb{E}\left[\frac{\Phi(x_{t+1})-\Phi(x_t)}{\eta_t}\right] \\
\leq& -\langle\nabla\Phi(x_t),\nabla_x f(x_t,y_t)\rangle + l\kappa\mathbb{E}\left[\eta_t\left\|\nabla_x\widetilde{f}(x_t,y_t)\right\|^2\right] \\
=& -\|\nabla_x f(x_t,y_t)\|^2 + \langle\nabla_x f(x_t,y_t)-\nabla\Phi(x_t),\nabla_x f(x_t,y_t)\rangle + l\kappa\mathbb{E}\left[\eta_t\left\|\nabla_x\widetilde{f}(x_t,y_t)\right\|^2\right] \\
\leq& -\|\nabla_x f(x_t,y_t)\|^2 + \frac{1}{2}\|\nabla_x f(x_t,y_t)-\nabla\Phi(x_t)\|^2 + \frac{1}{2}\|\nabla_x f(x_t,y_t)\|^2 + l\kappa\mathbb{E}\left[\eta_t\left\|\nabla_x\widetilde{f}(x_t,y_t)\right\|^2\right] \\
=& -\frac{1}{2}\|\nabla_x f(x_t,y_t)\|^2 + \frac{1}{2}\|\nabla_x f(x_t,y_t)-\nabla\Phi(x_t)\|^2 + l\kappa\mathbb{E}\left[\eta_t\left\|\nabla_x\widetilde{f}(x_t,y_t)\right\|^2\right]
\end{aligned}
$$

Summing over $t=0$ to $T-1$, rearranging and taking total expectation, we get

$$
\begin{aligned}
&\mathbb{E}\left[\sum_{t=0}^{T-1}\|\nabla_x f(x_t,y_t)\|^2\right] \\
\leq& \underbrace{2\mathbb{E}\left[\sum_{t=0}^{T-1}\frac{\Phi(x_t)-\Phi(x_{t+1})}{\eta_t}\right]}_{\text{(I)}} + \underbrace{2l\kappa\mathbb{E}\left[\sum_{t=0}^{T-1}\eta_t\left\|\nabla_x\widetilde{f}(x_t,y_t)\right\|^2\right]}_{\text{(J)}} + \underbrace{\mathbb{E}\left[\sum_{t=0}^{T-1}\|\nabla_x f(x_t,y_t)-\nabla\Phi(x_t)\|^2\right]}_{\text{(K)}}.
\end{aligned} \tag{10}
$$

Term (I)

$$
\begin{aligned}
2\mathbb{E}\left[\sum_{t=0}^{T-1}\frac{\Phi(x_t)-\Phi(x_{t+1})}{\eta_t}\right] &\leq 2\mathbb{E}\left[\frac{\Phi(x_0)}{\eta_0}+\sum_{t=1}^{T-1}\Phi(x_t)\left(\frac{1}{\eta_t}-\frac{1}{\eta_{t-1}}\right)\right] \\
&\leq 2\mathbb{E}\left[\frac{\Phi_{\max}}{\eta_0}+\sum_{t=1}^{T-1}\Phi_{\max}\left(\frac{1}{\eta_t}-\frac{1}{\eta_{t-1}}\right)\right] \\
&= 2\mathbb{E}\left[\frac{\Phi_{\max}}{\eta_{T-1}}\right] = 2\mathbb{E}\left[\frac{\Phi_{\max}}{\eta^x}\max\left\{v_T^x,v_T^y\right\}^\alpha\right].
\end{aligned}
$$

Term (J)

$$
\begin{aligned}
2l\kappa\sum_{t=0}^{T-1}\mathbb{E}\left[\eta_t\left\|\nabla_x\widetilde{f}(x_t,y_t)\right\|^2\right] &= 2l\kappa\mathbb{E}\left[\sum_{t=0}^{T-1}\frac{\eta^x}{\max\left\{v_{t+1}^x,v_{t+1}^y\right\}^\alpha}\left\|\nabla_x\widetilde{f}(x_t,y_t)\right\|^2\right] \\
&\leq 2l\kappa\eta^x\mathbb{E}\left[\sum_{t=0}^{T-1}\frac{1}{\left(v_{t+1}^x\right)^\alpha}\left\|\nabla_x\widetilde{f}(x_t,y_t)\right\|^2\right] \\
&\leq 2l\kappa\eta^x\mathbb{E}\left[\left(\frac{v_0^x}{\left(v_0^x\right)^\alpha}+\sum_{t=0}^{T-1}\frac{1}{\left(v_{t+1}^x\right)^\alpha}\left\|\nabla_x\widetilde{f}(x_t,y_t)\right\|^2\right)\right] \\
&\leq \frac{2l\kappa\eta^x}{1-\alpha}\mathbb{E}\left[\left(v_T^x\right)^{1-\alpha}\right].
\end{aligned}
$$

Term (K) According to the smoothness of $f(x_t,\cdot)$, we have

$$
\mathbb{E}\left[\sum_{t=0}^{T-1}\|\nabla_x f(x_t,y_t)-\nabla\Phi(x_t)\|^2\right] \leq l^2\mathbb{E}\left[\sum_{t=0}^{T-1}\|y_t-y_t^*\|^2\right] \leq 2l\kappa\mathbb{E}\left[\sum_{t=0}^{T-1}\left(f(x_t,y_t^*)-f(x_t,y_t)\right)\right],
$$

where the last inequality follows the strong-concavity of y. Now we let

$$C = \left(\frac{8l\kappa^3 \left(\eta^x\right)^2}{\mu \left(\eta^y\right)^2} \right)^{\frac{1}{2\alpha - 2\beta}},$$

and apply Lemma C.5, in total, we have

$$\begin{aligned}
&\mathbb{E}\left[\sum_{t=0}^{T-1} \|\nabla_x f(x_t, y_t)\|^2\right] \\
\leq\ & \frac{1}{2}\mathbb{E}\left[\sum_{t=0}^{T-1} \|\nabla_x f(x_t, y_t)\|^2\right] + 2\mathbb{E}\left[\frac{\Phi_{\max}}{\eta^x} \max\left\{v_T^x, v_T^y\right\}^\alpha\right] + \frac{2l\kappa\eta^x}{1-\alpha}\mathbb{E}\left[\left(v_T^x\right)^{1-\alpha}\right] \\
&+ \frac{l\kappa\eta^y}{1-\beta}\mathbb{E}\left[\left(v_T^y\right)^{1-\beta}\right] + \left(\frac{1}{\mu} + \frac{\eta^y}{\left(v_0^y\right)^\beta}\right) \frac{8l\kappa^2\eta^x G^2}{\eta^y \left(v_0^y\right)^\alpha}\mathbb{E}\left[\left(v_T^y\right)^\beta\right] \\
&+ \frac{\kappa^4 \left(\mu\eta^y C^\beta + 2C^{2\beta}\right)\left(\eta^x\right)^2}{\left(\eta^y\right)^2}\mathbb{E}\left[\frac{1 + \log v_T^x - \log v_0^x}{\left(v_0^x\right)^{2\alpha-1}} \cdot \mathbf{1}_{\alpha \geq 0.5} + \frac{\left(v_T^x\right)^{1-2\alpha}}{1-2\alpha} \cdot \mathbf{1}_{\alpha < 0.5}\right] \\
&+ \left(\kappa^2 + \frac{\widehat{L}^2 G^2 \left(\eta^x\right)^2}{\mu\eta^y \left(v_0^y\right)^{2\alpha-\beta}}\right) \frac{l\kappa \left(\eta^x\right)^2}{(1-\alpha)\eta^y \left(v_0^y\right)^{\alpha-\beta}}\mathbb{E}\left[\left(v_T^x\right)^{1-\alpha}\right] + \frac{\kappa^2 \left(v_0^y\right)^\beta G^2}{\mu\eta^y} \\
&+ \frac{l\kappa \left(2\beta G\right)^{\frac{1}{1-\beta}+2} G^2}{2\mu^{\frac{1}{1-\beta}+3} \left(\eta^y\right)^{\frac{1}{1-\beta}+2} \left(v_0^y\right)^{2-2\beta}}.
\end{aligned}$$

It remains to bound $\left(v_T^z\right)^m$ for $z \in \{x, y\}$ and $m \geq 0$:

$$\left(v_T^z\right)^m \leq \left(TG^2\right)^m \leq \mathcal{O}\left(T^m\right).$$

Therefore, we have

$$\mathbb{E}\left[\sum_{t=0}^{T-1} \|\nabla_x f(x_t, y_t)\|^2\right] \leq \mathcal{O}\left(T^{1-\alpha} + T^\alpha + T^{1-\beta} + T^\beta\right).$$

In addition, by Lemma C.5,

$$\mathbb{E}\left[\sum_{t=0}^{T-1} \|\nabla_y f(x_t, y_t)\|^2\right] \leq \mathcal{O}\left(\mathbb{E}\left[\sum_{t=0}^{T-1} \|\nabla_y f(x_t, y_t)\|^2\right]\right) + \mathcal{O}\left(T^{1-\alpha} + T^{1-\beta} + T^\beta\right),$$

which concludes our proof.

□

C.4 TiAda without Accessing Opponent's Gradients

The effective stepsize of x requires the knowledge of gradients of y, i.e., v_{t+1}^y. At the end of Section 3, we discussed the situation when such information is not available. Now we formally introduce the algorithm and present the convergence result.

Theorem C.1 (stochastic)**.** *Under Assumptions 3.1, 3.2, 3.4 and 3.5, Algorithm 2 with stochastic gradient oracles satisfies that for any $0 < \beta < \alpha < 1$, after T iterations,*

$$\begin{aligned}
&\frac{1}{T}\mathbb{E}\left[\sum_{t=0}^{T-1} \|\nabla_x f(x_t, y_t)\|^2 + \sum_{t=0}^{T-1} \|\nabla_y f(x_t, y_t)\|^2\right] \\
\leq\ & \mathcal{O}\left(T^{\alpha-1} + T^{-\alpha} + T^{-\beta}\right) + \mathcal{O}\left(T^{2\beta-2\alpha}\right) \cdot \mathbf{1}_{\alpha<0.5} + \widetilde{\mathcal{O}}\left(T^{2\beta-1}\right) \cdot \mathbf{1}_{\alpha \geq 0.5},
\end{aligned}$$

where $\mathbf{1}$ *is the indicator function.*

Remark C.1. The best rate achievable is $\widetilde{\mathcal{O}}\left(\epsilon^{-6}\right)$ by choosing $\alpha = 1/2$ and $\beta = 1/3$.

Algorithm 2 TiAda without MAX

1: **Input:** (x_0, y_0), $v_0^x > 0$, $v_0^y > 0$, $\eta^x > 0$, $\eta^y > 0$, $\alpha > 0$, $\beta > 0$ and $\alpha > \beta$.
2: **for** $t = 0, 1, 2, ...$ **do**
3: sample i.i.d. ξ_t^x and ξ_t^y, and let $g_t^x = \nabla_x F(x_t, y_t; \xi_t^x)$ and $g_t^y = \nabla_y F(x_t, y_t; \xi_t^y)$
4: $v_{t+1}^x = v_t^x + \|g_t^x\|^2$ and $v_{t+1}^y = v_t^y + \|g_t^y\|^2$
5: $x_{t+1} = x_t - \frac{\eta^x}{(v_{t+1}^x)^\alpha} g_t^x$ and $y_{t+1} = \mathcal{P}_{\mathcal{Y}}\left(y_t + \frac{\eta^y}{(v_{t+1}^y)^\beta} g_t^y\right)$
6: **end for**

Proof. Lemmas C.1 and C.4 can be directly used here because they do not have or expand the effective stepsize of x, i.e., η_t. This is also the case for the beginning part of Appendix C.3, the proof of Theorem 3.2, up to Equation (10). However, we need to bound Terms (I), (J) and (K) in Equation (10) differently. According to our assumption on bounded stochastic gradients, we know that v_{t+1}^x and v_{t+1}^y are both upper bounded by $\mathcal{O}(T)$, which we will use throughout the proof.

Term (I)

$$
\begin{aligned}
2\mathbb{E}\left[\sum_{t=0}^{T-1} \frac{\Phi(x_t) - \Phi(x_{t+1})}{\eta_t}\right] &\leq 2\mathbb{E}\left[\frac{\Phi(x_0)}{\eta_0} + \sum_{t=1}^{T-1} \Phi(x_t)\left(\frac{1}{\eta_t} - \frac{1}{\eta_{t-1}}\right)\right] \\
&\leq 2\mathbb{E}\left[\frac{\Phi_{\max}}{\eta_0} + \sum_{t=1}^{T-1} \Phi_{\max}\left(\frac{1}{\eta_t} - \frac{1}{\eta_{t-1}}\right)\right] \\
&= 2\mathbb{E}\left[\frac{\Phi_{\max}}{\eta_{T-1}}\right] = 2\mathbb{E}\left[\frac{\Phi_{\max}}{\eta^x}\left(v_T^x\right)^\alpha\right] \leq \mathcal{O}\left(T^\alpha\right).
\end{aligned}
$$

Term (J)

$$
\begin{aligned}
2l\kappa \sum_{t=0}^{T-1} \mathbb{E}\left[\eta_t \left\|\nabla_x \widetilde{f}(x_t, y_t)\right\|^2\right] &= 2l\kappa\eta^x \mathbb{E}\left[\sum_{t=0}^{T-1} \frac{1}{\left(v_{t+1}^x\right)^\alpha}\left\|\nabla_x \widetilde{f}(x_t, y_t)\right\|^2\right] \\
&\leq 2l\kappa\eta^x \mathbb{E}\left[\left(\frac{v_0^x}{(v_0^x)^\alpha} + \sum_{t=0}^{T-1} \frac{1}{\left(v_{t+1}^x\right)^\alpha}\left\|\nabla_x \widetilde{f}(x_t, y_t)\right\|^2\right)\right] \\
&\leq \frac{2l\kappa\eta^x}{1-\alpha}\mathbb{E}\left[\left(v_T^x\right)^{1-\alpha}\right] \leq \mathcal{O}\left(T^{1-\alpha}\right).
\end{aligned}
$$

Term (K) According to the smoothness and strong-concavity of $f(x_t, \cdot)$, we have

$$
\mathbb{E}\left[\sum_{t=0}^{T-1} \|\nabla_x f(x_t, y_t) - \nabla\Phi(x_t)\|^2\right] \leq l^2 \mathbb{E}\left[\sum_{t=0}^{T-1} \|y_t - y_t^*\|^2\right] \leq 2l\kappa \mathbb{E}\left[\sum_{t=0}^{T-1} \left(f(x_t, y_t^*) - f(x_t, y_t)\right)\right].
$$

To bound the RHS, we use Young's inequality and have

$$
\left\|y_{t+1} - y_{t+1}^*\right\|^2 \leq (1 + \lambda_t)\|y_{t+1} - y_t^*\|^2 + \left(1 + \frac{1}{\lambda_t}\right)\left\|y_{t+1}^* - y_t^*\right\|^2.
$$

Then applying Lemma C.1 with $\lambda_t = \frac{\mu\gamma_t}{2}$ gives us

$$
\begin{aligned}
&\mathbb{E}\left[\sum_{t=0}^{T-1} \left(f(x_t, y_t^*) - f(x_t, y_t)\right)\right] \\
&\leq \mathbb{E}\left[\sum_{t=0}^{T-1} \left(\frac{1 - \gamma_t\mu}{2\gamma_t}\|y_t - y_t^*\|^2 - \frac{1}{\gamma_t(2 + \mu\gamma_t)}\left\|y_{t+1} - y_{t+1}^*\right\|^2\right)\right] \\
&\quad + \underbrace{\mathbb{E}\left[\sum_{t=0}^{T-1} \frac{\gamma_t}{2}\left\|\nabla_y \widetilde{f}(x_t, y_t)\right\|^2\right]}_{\text{(L)}} + \underbrace{\mathbb{E}\left[\sum_{t=0}^{T-1} \frac{\left(1 + \frac{2}{\mu\gamma_t}\right)}{\gamma_t(2 + \mu\gamma_t)}\left\|y_{t+1}^* - y_t^*\right\|^2\right]}_{\text{(M)}},
\end{aligned}
$$

where the first term is $\mathcal{O}(1)$ according to Lemma C.4. The other two terms can be bounded as follow.

Term (L)

$$\leq \mathbb{E}\left[\frac{\eta^y}{2}\left(\frac{v_0^y}{(v_0^y)^\beta}+\sum_{t=0}^{T-1}\frac{1}{(v_{t+1}^y)^\beta}\left\|\nabla_y \widetilde{f}(x_t,y_t)\right\|^2\right)\right] \leq \mathbb{E}\left[\frac{\eta^y}{2(1-\beta)}\left(v_T^y\right)^{1-\beta}\right] \leq \mathcal{O}\left(T^{1-\beta}\right).$$

Term (M)

$$\begin{aligned}
&= \mathbb{E}\left[\sum_{t=0}^{T-1}\left(\frac{1}{(v_{t+1}^y)^\beta}+\frac{2}{\mu\eta^y}\right)\frac{(v_{t+1}^y)^{2\beta}}{2\eta^y(1+\lambda_t)}\left\|y_{t+1}^*-y_t^*\right\|^2\right] \\
&\leq \left(\frac{1}{2\left(v_0^y\right)^\beta\eta^y}+\frac{1}{\mu(\eta^y)^2}\right)\mathbb{E}\left[\sum_{t=0}^{T-1}\left(v_{t+1}^y\right)^{2\beta}\left\|y_{t+1}^*-y_t^*\right\|^2\right] \\
&\leq \left(\frac{1}{2\left(v_0^y\right)^\beta\eta^y}+\frac{1}{\mu(\eta^y)^2}\right)\mathbb{E}\left[\left(v_T^y\right)^{2\beta}\sum_{t=0}^{T-1}\left\|y_{t+1}^*-y_t^*\right\|^2\right] \\
&\leq \left(\frac{\kappa^2}{2\left(v_0^y\right)^\beta\eta^y}+\frac{\kappa^2}{\mu(\eta^y)^2}\right)\mathbb{E}\left[\left(v_T^y\right)^{2\beta}\sum_{t=0}^{T-1}\left\|x_{t+1}-x_t\right\|^2\right] \\
&= \left(\frac{(\eta^x)^2\kappa^2}{2\left(v_0^y\right)^\beta\eta^y}+\frac{(\eta^x)^2\kappa^2}{\mu(\eta^y)^2}\right)\mathbb{E}\left[\left(v_T^y\right)^{2\beta}\sum_{t=0}^{T-1}\frac{1}{(v_{t+1}^x)^{2\alpha}}\left\|\nabla_x \widetilde{f}(x_t,y_t)\right\|^2\right] \\
&\leq \left(\frac{(\eta^x)^2\kappa^2}{2\left(v_0^y\right)^\beta\eta^y}+\frac{(\eta^x)^2\kappa^2}{\mu(\eta^y)^2}\right)\mathbb{E}\left[\left(v_T^y\right)^{2\beta}\left(\frac{1+\log v_T^x-\log v_0^x}{(v_0^x)^{2\alpha-1}}\cdot\mathbf{1}_{\alpha\geq 0.5}+\frac{(v_T^x)^{1-2\alpha}}{1-2\alpha}\cdot\mathbf{1}_{\alpha<0.5}\right)\right] \\
&\leq \mathcal{O}\left(T^{1-2\alpha+2\beta}\right)\cdot\mathbf{1}_{\alpha<0.5}+\widetilde{\mathcal{O}}\left(T^{2\beta}\right)\cdot\mathbf{1}_{\alpha\geq 0.5},
\end{aligned}$$

where we used the the Lipschitzness of $y^*(\cdot)$ in the third inequality.

Summarizing all the terms, we finish the proof.

□