An Optimal Algorithm for Strongly Convex Min-Min Optimization

Dmitry Kovalev[1,2] **Alexander Gasnikov**[3,4,5] **Grigory Malinovsky**[6]

[1]Yandex Research
[2]Ivannikov Institute for System Programming
[3]AI Research Center, Innopolis University
[4]Skolkovo Institute of Science and Technology (Skoltech)
[5]Moscow Institute of Physics and Technology (MIPT)
[6]King Abdullah University of Science and Technology (KAUST)

Abstract

We consider the problem of minimizing a function $f(x, y)$, where f is a smooth and strongly convex function with respect to both variables, being μ_x-strongly convex in x and μ_y-strongly convex in y. The optimal accelerated gradient method of Yurii Nesterov achieves a convergence rate that requires approximately $\mathcal{O}((\min(\mu_x, \mu_y))^{-1/2})$ evaluations of the partial gradients $\nabla_x f$ and $\nabla_y f$. In this paper, we propose a novel optimization algorithm that improves upon this complexity by requiring only $\mathcal{O}(\mu_x^{-1/2})$ computations of $\nabla_x f$ and $\mathcal{O}(\mu_y^{-1/2})$ computations of $\nabla_y f$. This improvement is particularly advantageous in scenarios where there is a significant disparity between the strong convexity parameters, specifically when $\mu_x \gg \mu_y$. Furthermore, in practical applications where the computation of $\nabla_y f$ is considerably more efficient than that of $\nabla_x f$, the proposed method leads to a substantial reduction in the overall wall-clock time required for optimization. As a key application, we consider Partially Local Federated Learning, a setting in which the model is partitioned into a local component and a global component. We demonstrate how our proposed method can be effectively applied in this framework, highlighting its practical advantages in improving computational efficiency.

1 INTRODUCTION

The development of optimal ("black-box") algorithms for fundamental classes of convex optimization problems dates back several decades [Nemirovski and Yudin, 1983]. Contemporary research, however, often exploits the additional structural properties of optimization problems, effectively "looking inside the black box" [Nesterov, 2018]. Many notable results in this direction focus on problems with a composite structure[2], formulated as

$$\min_x F(x) := f(x) + g(x), \tag{1}$$

where the complexity of the problem can be "split" into two components: approximately $\sqrt{L_f/\mu}$ evaluations of ∇f and $\sqrt{L_g/\mu}$ evaluations of ∇g [Lan, 2016, Ivanova et al., 2022, Kovalev et al., 2022].

However, there remains a significant gap in the literature regarding "optimal results" for the so-called *min-min* problem:

$$\min_{x,y} f(x, y), \tag{2}$$

where the smoothness constants, strong convexity parameters, and computational complexities of $\nabla_x f$ and $\nabla_y f$ can vary significantly between the variables x and y, as well as in their respective dimensionalities.

Such problems frequently arise in various applications, including transportation modeling, where they play a crucial role in combined trip distribution and assignment [De Cea et al., 2005, Gasnikov et al., 2014], as well as in soft clustering [Nesterov, 2020]. A particularly relevant application in Machine Learning can be seen in the Yahoo! Click-Prediction model proposed by [Dvurechensky et al., 2022]:

$$\min_{x\in\mathbb{R}^d} f(x) := \frac{1}{m}\sum_{k=1}^{m} \log\left(1 + \exp\left(-\eta^k \langle \xi^k, x\rangle\right)\right) + \lambda_S \sum_{i\in I_S} x_i^2 + \lambda_D \sum_{i\in I_D} x_i^2,$$

where $I_S \cup I_D = \{1, ..., d\}$, $I_S \cap I_D = \varnothing$, with $|I_D| \gg |I_S|$ and $\lambda_S \gg \lambda_D$.

In this context, it is natural to define $x := \{x_i\}_{i\in I_S}$ and $y := \{x_i\}_{i\in I_D}$ in the formulation of (2), highlighting the distinct structural differences in optimization complexities between these two variable groups.

[2]The function F is μ-strongly convex, while both f and g have Lipschitz-continuous gradients with constants L_f and L_g, respectively.

1.1 PROBLEM SETUP AND OVERVIEW OF MAIN RESULT

In this paper, we consider the following class of optimization problems:

$$\min_{x\in\mathbb{R}^{d_x}, y\in\mathbb{R}^{d_y}} f(x,y), \tag{3}$$

where $f(x,y)\colon \mathbb{R}^{d_x}\times\mathbb{R}^{d_y}\to\mathbb{R}$ is a convex function satisfying the assumptions outlined below. We impose the following standard smoothness and strong convexity conditions on $f(x,y)$:

Assumption 1.1. The function $f(x,y)$ is (L_x, L_y)-smooth with constants $L_x, L_y > 0$. That is, for all $x_1, x_2 \in \mathbb{R}^{d_x}$ and $y_1, y_2 \in \mathbb{R}^{d_y}$, the following inequality holds:

$$\begin{aligned} f(x_2,y_2) \le f(x_1,y_1) &+ \langle \nabla_x f(x_1,y_1), x_2 - x_1\rangle \\ &+ \langle \nabla_y f(x_1,y_1), y_2 - y_1\rangle \\ &+ \frac{L_x}{2}\|x_2 - x_1\|^2 + \frac{L_y}{2}\|y_2 - y_1\|^2. \end{aligned} \tag{4}$$

This condition implies that the gradients $\nabla_x f$ and $\nabla_y f$ are Lipschitz continuous with constants L_x and L_y, respectively, up to a factor of 2.

Assumption 1.2. The function $f(x,y)$ is (μ_x, μ_y)-strongly convex with constants $\mu_x, \mu_y > 0$. That is, for all $x_1, x_2 \in \mathbb{R}^{d_x}$ and $y_1, y_2 \in \mathbb{R}^{d_y}$, the function satisfies:

$$\begin{aligned} f(x_2,y_2) \ge f(x_1,y_1) &+ \langle \nabla_x f(x_1,y_1), x_2 - x_1\rangle \\ &+ \langle \nabla_y f(x_1,y_1), y_2 - y_1\rangle \\ &+ \frac{\mu_x}{2}\|x_2 - x_1\|^2 + \frac{\mu_y}{2}\|y_2 - y_1\|^2. \end{aligned} \tag{5}$$

This assumption ensures that $f(x,y)$ exhibits strong convexity in both x and y, which is crucial for achieving fast convergence rates using accelerated methods.

The main contribution of this paper is the introduction of the Block Accelerated Method (BAM) (see Section 2), which efficiently solves problem (3) to a relative precision ϵ with the following computational complexity:

$$\boxed{\begin{array}{c} \mathcal{O}\left(\sqrt{\frac{L_x}{\mu_x}}\log\frac{1}{\epsilon}\right) \text{ evaluations of } \nabla_x f \\ \text{and} \\ \mathcal{O}\left(\sqrt{\frac{L_y}{\mu_y}}\log\frac{1}{\epsilon}\right) \text{ evaluations of } \nabla_y f. \end{array}}$$

These complexity bounds match the known lower bounds for strongly convex smooth optimization, as established in classical results by Nemirovski and Yudin [1983] and Nesterov [2018]. Therefore, our method is optimal in terms of the number of gradient evaluations required for solving (3).

Moreover, when $f(x,y)$ is convex but not strongly convex in one or both blocks, we can apply a regularization technique (see, e.g., Gasnikov et al. [2016]) to transform the problem into a strongly convex one. Specifically, by introducing a small regularization term, we can ensure strong convexity with a parameter of approximately $\mu_\circ \sim \epsilon/R^2$, where R is the Euclidean distance between the initial point and the closest optimal solution. This allows us to extend the applicability of our method to a broader class of convex problems.

1.2 RELATED WORKS

The problem formulation we consider in this paper is closely related to those studied in the context of accelerated coordinate-descent methods [Nesterov, 2012, Richtárik and Takáč, 2014, Nesterov and Stich, 2017, Ivanova et al., 2021]. However, while the formulation is similar, our results differ significantly.

In particular, prior work on accelerated coordinate-descent methods has established that, with probability at least $1-\delta$, the complexity bounds for solving our problem (3) using randomized coordinate-wise acceleration are:

$$\begin{array}{c} \mathcal{O}\left(\sqrt{\frac{L_x}{\min\{\mu_x,\mu_y\}}}\log\frac{1}{\epsilon}\log\frac{1}{\delta}\right) \text{ evaluations of } \nabla_x f \\ \text{and} \\ \mathcal{O}\left(\sqrt{\frac{L_y}{\min\{\mu_x,\mu_y\}}}\log\frac{1}{\epsilon}\log\frac{1}{\delta}\right) \text{ evaluations of } \nabla_y f, \end{array}$$

where L_x and L_y are the Lipschitz constants of $\nabla_x f(x,y)$ with respect to x and $\nabla_y f(x,y)$ with respect to y, respectively.

The first accelerated coordinate-descent methods [Nesterov, 2012, Richtárik and Takáč, 2014] did not yield these results directly. The breakthrough came with [Nesterov, May 14, 2015], which introduced a specialized coordinate-wise randomization scheme with probabilities $p_x \sim \sqrt{L_x}$ and $p_y \sim \sqrt{L_y}$. This approach was further developed in subsequent works, leading to various algorithmic refinements [Gasnikov et al., 2015, Allen-Zhu et al., 2016, Nesterov and Stich, 2017].

Similar complexity bounds, albeit with slightly worse smoothness constants, have also been derived for **accelerated alternating methods** [Beck, 2017, Diakonikolas and Orecchia, 2018, Guminov et al., 2021, Tupitsa et al., 2021].

An alternative way to analyze the complexity of solving (3) is through a variable re-scaling approach. Specifically, by introducing a re-scaled variable $y' := \sqrt{\mu_y/\mu_x}y$, we can equalize the strong convexity constants such that $\mu_x = \mu_{y'}$. Applying the accelerated coordinate-descent method from [Nesterov and Stich, 2017] to the re-scaled problem, we obtain the following complexity bounds in the original variables:

Algorithm 1 Block **A**ccelerated **M**ethod (BAM)

Parameters: $\eta_x, \eta_y > 0$, $\theta_x, \theta_y > 0$, $\alpha \in (0,1)$
Input: $x^0 = \overline{x}^0 \in \mathbb{R}^{d_x}$, $y^0 = \overline{y}^0$
for $k = 0, 1, \ldots, K-1$ **do**
$\underline{x}^k = \alpha x^k + (1-\alpha)\overline{x}^k$
$\underline{y}^k = \alpha y^k + (1-\alpha)\overline{y}^k$
find $\overline{y}^{k+1}$ such that

$$\|\nabla_y f(\underline{x}^k, \overline{y}^{k+1}) + (\eta_y \alpha)^{-1}(\overline{y}^{k+1} - \underline{y}^k)\| \leq (\eta_y \alpha)^{-1}\|\overline{y}^{k+1} - \underline{y}^k\|. \tag{6}$$

$\overline{x}^{k+1} = \underline{x}^k - \eta_x \alpha \nabla_x f(\underline{x}^k, \overline{y}^{k+1})$
$x^{k+1} = x^k + \alpha(\underline{x}^k - x^{k+1}) - \eta_x \nabla_x f(\underline{x}^k, \overline{y}^{k+1})$
$y^{k+1} = y^k + \alpha(\overline{y}^{k+1} - y^{k+1}) - \eta_y \nabla_y f(\underline{x}^k, \overline{y}^{k+1})$
end for

$$\mathcal{O}\left(\sqrt{\tfrac{L_x}{\mu_x}} \log \tfrac{1}{\epsilon} \log \tfrac{1}{\delta}\right) \text{ evaluations of } \nabla_x f$$
and
$$\mathcal{O}\left(\sqrt{\tfrac{L_y}{\mu_y}} \log \tfrac{1}{\epsilon} \log \tfrac{1}{\delta}\right) \text{ evaluations of } \nabla_y f.$$

These results are closely related to our findings, but our approach provides an important advantage: our method is fully deterministic, which eliminates the additional logarithmic dependence on δ present in randomized methods. Moreover, our derivation is based on fundamentally different theoretical principles, further distinguishing our work from prior research.

An alternative line of research approaches problem (3) using a nested optimization framework, where the outer optimization is performed over x, and the inner problem in y is solved approximately to provide an inexact gradient oracle. This methodology has been explored in a series of works [Bolte et al., 2020, Gladin et al., 2021a,b, Ostroukhov, 2022], where the objective function is reformulated as:

$$\begin{aligned} \min_x F(x) &:= \min_y f(x,y), \\ \nabla F(x) &= \nabla_x f(x, y(x)) \\ &= \frac{\partial f}{\partial x}(x,y)\Big|_{y=y(x)}, \end{aligned}$$

where $y(x)$ is defined as the solution to the inner minimization problem $\min_y f(x,y)$.

The most practical results in this framework have been obtained for problems where x belongs to a low-dimensional set $Q \subset \mathbb{R}^{d_x}$, where d_x is relatively small. In this case, the established complexity bounds are:

$$\tilde{\mathcal{O}}\left(d_x \log \tfrac{1}{\epsilon}\right) \text{ evaluations of } \nabla_x f$$
and
$$\tilde{\mathcal{O}}\left(d_x \sqrt{\tfrac{L_y}{\min\{\mu_x, \mu_y\}}} \log^2 \tfrac{1}{\epsilon}\right) \text{ evaluations of } \nabla_y f.$$

Interestingly, the known lower bounds for this setting suggest:

$$\mathcal{O}\left(d_x \log \tfrac{1}{\epsilon}\right) \text{ evaluations of } \nabla_x f$$
and
$$\mathcal{O}\left(\sqrt{\tfrac{L_y}{\min\{\mu_x, \mu_y\}}} \log \tfrac{1}{\epsilon}\right) \text{ evaluations of } \nabla_y f.$$

However, it remains unclear whether this lower bound is tight, leaving room for potential improvements in future research.

This nested optimization framework has also been extended to scenarios involving various types of inner problem oracles: Gradient-free approaches [Gladin et al., 2021b], Randomized variance-reduced methods [Gladin et al., 2021a], Higher-order tensor methods [Ostroukhov, 2022]

Despite these advances, the performance of these methods deteriorates significantly when d_x is large. In this case, the outer method must be accelerated, leading to complexity bounds of:

$$\mathcal{O}\left(\sqrt{\tfrac{L_x}{\mu_x}} \log \tfrac{1}{\epsilon}\right) \text{ evaluations of } \nabla_x f$$
and
$$\mathcal{O}\left(\sqrt{\tfrac{L_x L_y}{\mu_x \mu_y}} \log^2 \tfrac{1}{\epsilon}\right) \text{ evaluations of } \nabla_y f.$$

This bound is significantly worse in terms of the number of $\nabla_y f$ evaluations compared to our method.

To summarize, prior to our work, no known deterministic optimization algorithm could achieve an independent complexity bound for each block without sacrificing theoretical guarantees. Our method provides the first fully deterministic approach that effectively decouples the complexities into separate terms for x and y, achieving an optimal rate without requiring coordinate-wise randomization or nested optimization frameworks.

The removal of the logarithmic factor in our analysis constitutes a substantial theoretical contribution. Eliminating stochasticity—and thus the associated logarithmic overhead—not only simplifies the analysis but also enhances the stability of the method by removing the need to average over multiple runs to estimate convergence behavior, a com-

mon requirement in stochastic settings. Our proof technique diverges significantly from standard analyses of accelerated coordinate methods that rely on coordinate sampling, thereby advancing the theoretical foundations for such methods. Historically, the elimination of logarithmic factors has marked several key breakthroughs in optimization. For instance, Accelerated Gradient Descent [1], while often seen as a novel application of momentum, can also be interpreted as eliminating logarithmic terms from the complexity of the conjugate gradient method [2]. Katyusha [3], a milestone in stochastic optimization, achieved direct acceleration with variance reduction, essentially refining the log-dependent Catalyst framework [4]. Similar log-factor removals underpinned major advances in online learning [5–7] and resolved a long-standing problem in the multi-armed bandit setting [8]. These precedents underscore the theoretical depth and practical value of eliminating such terms, highlighting the significance of our contribution.

2 MAIN ALGORITHM

The development of the Block Accelerated Method (BAM) was influenced by a series of recent advancements in optimization, particularly those presented by Kovalev et al. [2022], Kovalev and Gasnikov [2022a,b] (see also [Ivanova et al., 2021, Gasnikov et al., 2021, Carmon et al., 2022]). These works leverage inner-loop acceleration techniques, akin to catalyst-type methods, to derive optimal accelerated algorithms for saddle-point problems and high-order optimization methods.

However, it is important to emphasize that BAM represents a fundamentally different approach. While previous methods primarily focus on achieving optimal acceleration through nested iterations or high-order techniques, BAM is explicitly designed to decouple the complexities associated with different variable blocks. This distinction is crucial because splitting the computational burden into independent complexity bounds for each block is nontrivial and requires a novel algorithmic framework. Unlike existing methods that rely on uniform acceleration across all variables, BAM introduces a tailored acceleration mechanism that optimally balances the computational effort required for different blocks, ensuring efficiency without resorting to coordinate-wise randomization or nested optimization schemes.

Let us provide a detailed description of the BAM method. The first step involves computing convex combinations for both coordinate blocks. This operation can be interpreted as a form of momentum, which plays a central role in achieving acceleration:

$$\underline{x}^k = \alpha x^k + (1-\alpha)\bar{x}^k, \quad \underline{y}^k = \alpha y^k + (1-\alpha)\bar{y}^k.$$

Next, we solve a subproblem to ensure the following condition is satisfied:

$$\left\|\nabla_y f\left(\underline{x}^k, \bar{y}^{k+1}\right) + (\eta_y\alpha)^{-1}\left(\bar{y}^{k+1} - \underline{y}^k\right)\right\| \leq (\eta_y\alpha)^{-1}\left\|\bar{y}^{k+1} - \underline{y}^k\right\|.$$

This step is crucial for separating the complexity of different components and for enabling acceleration; it is also essential for the theoretical analysis. Subsequently, the method performs a gradient step on the server block:

$$\bar{x}^{k+1} = \underline{x}^k - \eta_x\alpha\nabla_x f\left(\underline{x}^k, \bar{y}^{k+1}\right).$$

Finally, the algorithm updates both coordinate blocks using gradient steps that incorporate the difference between iterations. This mechanism is another key component contributing to acceleration:

$$x^{k+1} = x^k + \alpha\left(\underline{x}^k - x^{k+1}\right) - \eta_x\nabla_x f\left(\underline{x}^k, \bar{y}^{k+1}\right),$$
$$y^{k+1} = y^k + \alpha\left(\bar{y}^{k+1} - y^{k+1}\right) - \eta_y\nabla_y f\left(\underline{x}^k, \bar{y}^{k+1}\right).$$

The theoretical guarantees and complexity bounds established in this work are fundamentally dependent on a key technical result, which we formalize in the core lemma below.

Lemma 2.1. *Let η_x satisfy $\eta_x \leq (\alpha L_x)^{-1}$. Then, the following inequality holds:*

$$-f(\underline{x}^k, \overline{y}^{k+1}) \leq -f(\overline{x}^{k+1}, \overline{y}^{k+1}) - \frac{\eta_x\alpha}{2}\|\nabla_x f(\underline{x}^k, \overline{y}^{k+1})\|^2. \tag{7}$$

We now formally present our main theoretical result in the theorem stated below. This theorem encapsulates the core contribution of our work, providing a rigorous statement of the achieved complexity bounds and demonstrating the effectiveness of the proposed algorithm.

Theorem 2.2. *Let $\mathcal{R}_x^k = \|x^k - x^*\|^2$, $\mathcal{R}_y^k = \|y^k - y^*\|^2$. Let Ψ^k be the following Lyapunov function:*

$$\Psi^k = (1+\alpha)\left(\eta_x^{-1}\mathcal{R}_x^k + \eta_y^{-1}\mathcal{R}_y^k\right) + \frac{2}{\alpha}\left(f(\overline{x}^k, \overline{y}^k) - f(x^*, y^*)\right). \tag{8}$$

Let parameters η_x, η_y, α be defined as follows:

$$\alpha = \sqrt{\frac{\mu_x}{L_x}}, \qquad \eta_x = \frac{1}{\sqrt{\mu_x L_x}}, \qquad \eta_y = \frac{1}{\mu_y}\sqrt{\frac{\mu_x}{L_x}}. \tag{9}$$

Then, iterations of Algorithm 1 satisfy the following inequality:

$$\Psi^{k+1} \leq (1+\alpha)^{-1}\Psi^k. \tag{10}$$

Algorithm 2 Optimized **G**radient **M**ethod (OGM-G)

Parameters: stepsize γ, matrix $\tilde{\theta}_i$:

$$\tilde{\theta}_i = \begin{cases} \frac{1+\sqrt{1+8\tilde{\theta}_{i+1}^2}}{2}, & i = 0, \\ \frac{1+\sqrt{1+4\tilde{\theta}_{i+1}^2}}{2}, & i = 1, \ldots, N-1, \\ 1, & i = N, \end{cases}$$

Input: $x^0 = y^0 \in \mathbb{R}^d$

for $i = 0, 1, \ldots, N-1$ **do**

$y_{i+1} = x_i - \gamma \nabla f(x_i)$

$x_{i+1} = y_{i+1} + \frac{(\tilde{\theta}_i - 1)(2\tilde{\theta}_{i+1} - 1)}{\tilde{\theta}_i (2\tilde{\theta}_i - 1)} (y_{i+1} - y_i) + \frac{2\tilde{\theta}_{i+1} - 1}{2\tilde{\theta}_i - 1} (y_{i+1} - x_i)$

end for

3 INNER ALGORITHM

We define the auxiliary function $A^k(y) \colon \mathbb{R}^{d_y} \to \mathbb{R}$ as follows:

$$A^k(y) = f(\underline{x}^k, y) + \frac{1}{2\eta_y \alpha} \|y - \underline{y}^k\|^2. \tag{11}$$

Then, the condition in (6) from Algorithm 1 can be equivalently written as:

$$\|\nabla A^k(\overline{y}^{k+1})\| \leq (\eta_y \alpha)^{-1} \|\overline{y}^{k+1} - \underline{y}^k\|. \tag{12}$$

To find $\overline{y}^{k+1}$ that satisfies this condition, we apply an optimal algorithm for gradient norm reduction [Diakonikolas and Wang, 2022, Kim and Fessler, 2021] to the minimization problem:

$$\min_{y \in \mathbb{R}^{d_y}} A^k(y). \tag{13}$$

The following theorem, taken from Remark 1 of [Nesterov et al., 2021], applies to this setup.

Theorem 3.1. *There exists an algorithm that, when applied to problem* (13) *with starting point* $\underline{y}^k$*, produces* $\overline{y}^{k+1}$ *satisfying:*

$$\|\nabla A^k(\overline{y}^{k+1})\| \leq \frac{C \max\{L_y, (\eta_y \alpha)^{-1}\} \|\underline{y}^k - \underline{y}^*\|}{T^2}, \tag{14}$$

where T *is the number of calls to* $\nabla A^k(y)$*,* $\underline{y}^* = \arg\min_{y \in \mathbb{R}^{d_y}} A^k(y)$*, and* $C > 0$ *is a universal constant.*

Corollary 3.2. *To output* $\overline{y}^{k+1}$ *that satisfies condition* (12)*, the inner algorithm requires the following number of iterations:*

$$T = \sqrt{2C} \max\left\{1, \sqrt{\eta_y \alpha L_y}\right\}. \tag{15}$$

A simple approach to achieve the optimal rate $\mathcal{O}\left(\frac{1}{T^2}\right)$ for gradient norm reduction under the initial distance condition involves running Nesterov Accelerated Gradient for the first $N/2$ iterations and then applying the OGM-G algorithm (Algorithm 2) for the remaining $N/2$ iterations.

The OGM-G algorithm utilizes a triangular matrix $\tilde{\theta}_i$, which determines coefficients for the iterations. The first step of the algorithm is a gradient step, while the second step is an acceleration step using previous points and the coefficients $\tilde{\theta}_i$.

4 TOTAL COMPLEXITY

Let us now formulate and summarize the key results, followed by an analysis of the total computational complexity.

From Theorem 2.2, we can conclude that to find an ϵ-accurate solution to problem (3), Algorithm 1 requires the following number of calls to $\nabla_x f(x, y)$:

$$K = \mathcal{O}\left(\sqrt{\frac{L_x}{\mu_x}} \log \frac{1}{\epsilon}\right). \tag{16}$$

Additionally, Corollary 3.2, in conjunction with the parameter choices in Algorithm 1 as derived from Theorem 2.2, implies that the number of inner iterations is:

$$\begin{aligned} T &= \mathcal{O}\left(\max\{1, \sqrt{\eta_y \alpha L_y}\}\right) \\ &= \mathcal{O}\left(\max\left\{1, \sqrt{\frac{L_y \mu_x}{L_x \mu_y}}\right\}\right). \end{aligned} \tag{17}$$

Thus, the total number of calls to $\nabla_y f(x, y)$ is:

$$\begin{aligned} K \times T &= \mathcal{O}\left(\sqrt{\frac{L_x}{\mu_x}} \log \frac{1}{\epsilon}\right) \times \mathcal{O}\left(\max\left\{1, \sqrt{\frac{L_y \mu_x}{L_x \mu_y}}\right\}\right) \\ &= \mathcal{O}\left(\max\left\{\sqrt{\frac{L_x}{\mu_x}}, \sqrt{\frac{L_y}{\mu_y}}\right\} \log \frac{1}{\epsilon}\right) \end{aligned}$$

This expression provides a concise description of the total complexity required for solving the problem to ϵ-accuracy, considering the number of gradient evaluations in both blocks.

5 FEDERATED LEARNING APPLICATION

5.1 COLLABORATIVE LEARNING

Federated learning is a robust machine learning paradigm in which multiple clients (or workers) collaborate to train a shared model in a distributed environment, while ensuring that the clients' local data remains private [McMahan et al., 2017]. This privacy is critical, as it allows for training on sensitive or proprietary data without the need to share it across participants. Typically, the data is distributed across numerous clients, and communication occurs only with a central server in the centralized regime [Konečnỳ et al., 2016]. In contrast, in the decentralized regime [Koloskova et al., 2020], clients interact based on a predefined communication graph, without relying on a central coordinator or server, enabling more flexible communication architectures. A key example of federated learning is in developing machine learning models for applications such as text prediction in mobile keyboards, where sensitive user data (such as typed text) is never shared between clients or with the server, maintaining privacy.

Federated learning is deployed in a variety of settings, including both cross-device and cross-silo environments. In cross-device settings, such as mobile devices or IoT devices, data is typically distributed across a large number of individual devices, and federated learning allows for the creation of global models without transferring sensitive data [Hard et al., 2018]. In cross-silo settings, such as corporate or institutional collaborations, data is distributed across a smaller number of entities (e.g., hospitals or banks), where federated learning facilitates model training across organizations while ensuring privacy and compliance with regulations [Rieke et al., 2020].

In standard federated learning approaches, a single global model is trained using local updates from clients. One of the most commonly used algorithms is FedAvg [Khaled et al., 2020, Woodworth et al., 2020], which reduces communication costs—typically the major bottleneck in federated learning—by allowing clients to perform several local gradient steps before sending their updates to the central server for aggregation. While this approach helps reduce communication frequency, it suffers from poor convergence guarantees in the presence of data heterogeneity, especially when no additional assumptions about data similarity are made. To overcome these limitations, several enhanced methods have been proposed [Karimireddy et al., 2020, Mitra et al., 2021, Gorbunov et al., 2021], which achieve linear convergence rates in deterministic settings. However, despite these improvements, the communication complexity of these methods still does not outperform vanilla gradient descent (GD) because of the small step sizes required in the analysis.

In a more recent advancement [Mishchenko et al., 2022], it was shown that incorporating local steps into the training process can indeed accelerate communication, offering a promising approach for improving the efficiency of federated learning. This has been further developed in subsequent works that extend this mechanism to various problem settings [Malinovsky et al., 2022, Grudzień et al., 2022, Condat et al., 2022]. These studies provide valuable insights into how local optimization strategies can complement global model aggregation, thereby enhancing the overall communication efficiency without sacrificing convergence speed.

However, global model training can be prohibited in some settings even without sharing data due to privacy constraints. For example, using client-specific embeddings can reveal user identity, which is not allowed by a privacy policy. In order to fix this issue, a concept of partial federated learning was introduced [Singhal et al., 2021]. In this approach, models have two blocks of parameters: global block x and local blocks y_i, which never leave the clients. This technique enables to have interpolation between distributed and non-distributed learning. Partial federated learning is closely connected to personalizing and meta-learning algorithms. The most popular meta-learning algorithm is MAML [Finn et al., 2017], and connection to federated learning was established in several works [Nichol et al., 2018, Chen et al., 2018, Fallah et al., 2020].

5.2 FEDERATED RECONSTRUCTION

Let us describe the baseline of partial federated learning known as Federated Reconstruction [Singhal et al., 2021]. In this framework, there are two blocks of coordinates: user-specific parameters y_i and non-user-specific parameters x. During each communication round, the server sends the global part of the parameters x_t to all clients. Each client then reconstructs its local parameters y_t^i using the current global model x_t. This reconstruction process generally requires several steps. Once the local model is restored, each client updates its copy of the global parameters and sends only the updated copies back to the server. The server then aggregates these updates and generates the next iterate x_{t+1}.

The newly proposed BAM algorithm can be extended to minimize $f(x, y_1, \ldots, y_M)$ in a distributed setting and can be applied to Federated Reconstruction [Singhal et al., 2021]. Since the communication complexity depends on the number of calls to $\nabla_x f(x, y_1, \ldots, y_M)$, the communication complexity of this method is $\mathcal{O}\left(\sqrt{\frac{L_x}{\mu_x}} \log \frac{1}{\varepsilon}\right)$. This bottleneck in communication can be alleviated when the condition number of the local parameters is small. Furthermore, this communication complexity is optimal.

We now elaborate on how to apply the Block Alternating Minimization (BAM) method in a distributed setting under the Federated Reconstruction framework. We consider a federated system with n clients and the following objective

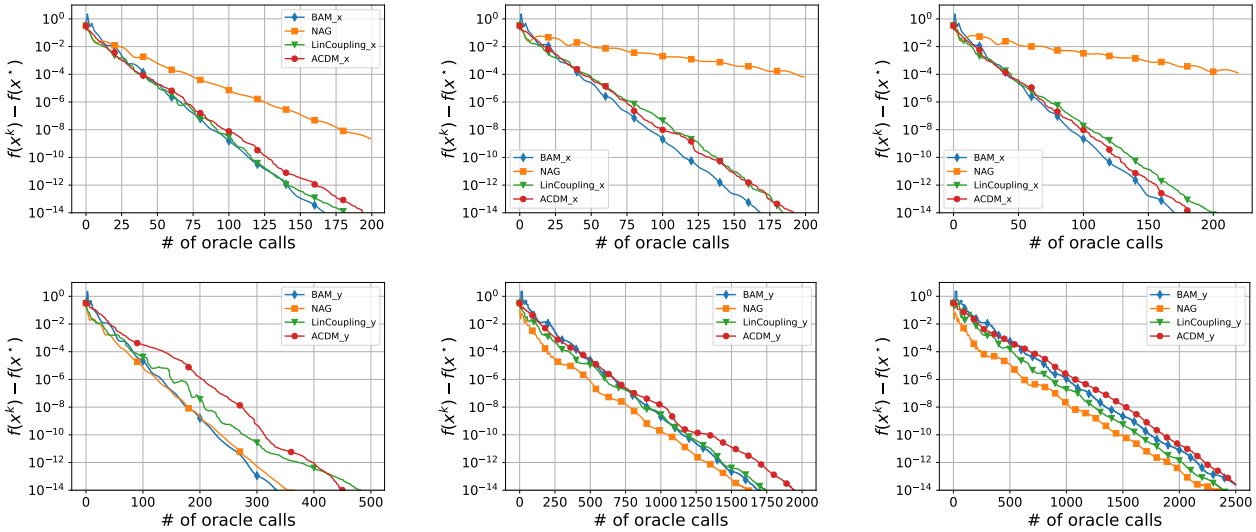

Figure 1: Comparison of Block Accelerated Method (BAM), Nesterov Accelerated Method (NAG), Accelerated Coordinate Descent Method (ACDM), and Linear Coupling method (LinCoupling) on logistic regression loss functions with two different l_2 regularizers. The first line represents the rate in terms of the $\nabla_x f(x, y)$ oracle calls, and the second one represents the rate in terms of the $\nabla_y f(x, y)$ oracle calls. We set $\mu_y = 0.002$ (left column), $\mu_y = 0.0001$ (middle column) and $\mu_y = 0.00005$ (right column).

function:

$$f(x, y_1, \ldots, y_n) = \frac{1}{n} \sum_{i=1}^{n} f_i(x, y_i),$$

where x denotes the global (server-specific) model block, and y_i represents the local block associated with client i. Each local loss function f_i depends only on the global variable x and the corresponding local variable y_i.

We now describe the Federated BAM algorithm. At the beginning of each communication round, the server computes the extrapolated global model:

$$\underline{x}^k = \alpha x^k + (1 - \alpha)\bar{x}^k.$$

The server then broadcasts $\underline{x}^k$ to all clients. Upon receiving this model, each client computes the extrapolated local model:

$$\underline{y}_i^k = \alpha y_i^k + (1 - \alpha)\bar{y}_i^k.$$

Each client then solves a local subproblem of the form (6) to find $\bar{y}_i^{k+1}$ such that

$$\left\| \nabla_y f\left(\underline{x}^k, \bar{y}_i^{k+1}\right) + (\eta_y \alpha)^{-1} \left(\bar{y}_i^{k+1} - \underline{y}_i^k\right) \right\| \leq (\eta_y \alpha)^{-1} \left\| \bar{y}_i^{k+1} - \underline{y}_i^k \right\|.$$

After solving this subproblem, the client updates its local variable as follows:

$$y_i^{k+1} = y_i^k + \alpha\left(\bar{y}_i^{k+1} - y_i^k\right) - \eta_y \nabla_y f\left(\underline{x}^k, \bar{y}_i^{k+1}\right).$$

Each client also computes the gradient with respect to the global variable: $\nabla_x f_i\left(\underline{x}^k, \bar{y}_i^{k+1}\right)$, and sends it to the server. The server aggregates these gradients to compute the full global gradient:

$$\nabla_x f\left(\underline{x}^k, \bar{y}_1^{k+1}, \ldots, \bar{y}_n^{k+1}\right) = \frac{1}{n} \sum_{i=1}^{n} \nabla_x f_i\left(\underline{x}^k, \bar{y}_i^{k+1}\right).$$

Using the aggregated gradient, the server updates the global model as follows:

$$\bar{x}^{k+1} = \underline{x}^k - \eta_x \alpha \nabla_x f\left(\underline{x}^k, \bar{y}^{k+1}\right),$$
$$x^{k+1} = x^k + \alpha\left(\underline{x}^k - x^k\right) - \eta_x \nabla_x f\left(\underline{x}^k, \bar{y}^{k+1}\right).$$

The process repeats until convergence, with local parameters y_i staying on clients and only the global parameter x shared with the server.

Experimental results for partial federated learning can be found in Singhal et al. [2021], Mishchenko et al. [2023]. A detailed study of the practical application of BAM to Partial Personalized Federated Learning with deep learning models is left for future work.

6 EXPERIMENTS

In all of our experiments, we compare the proposed Block Accelerated Method (BAM) with several well-established

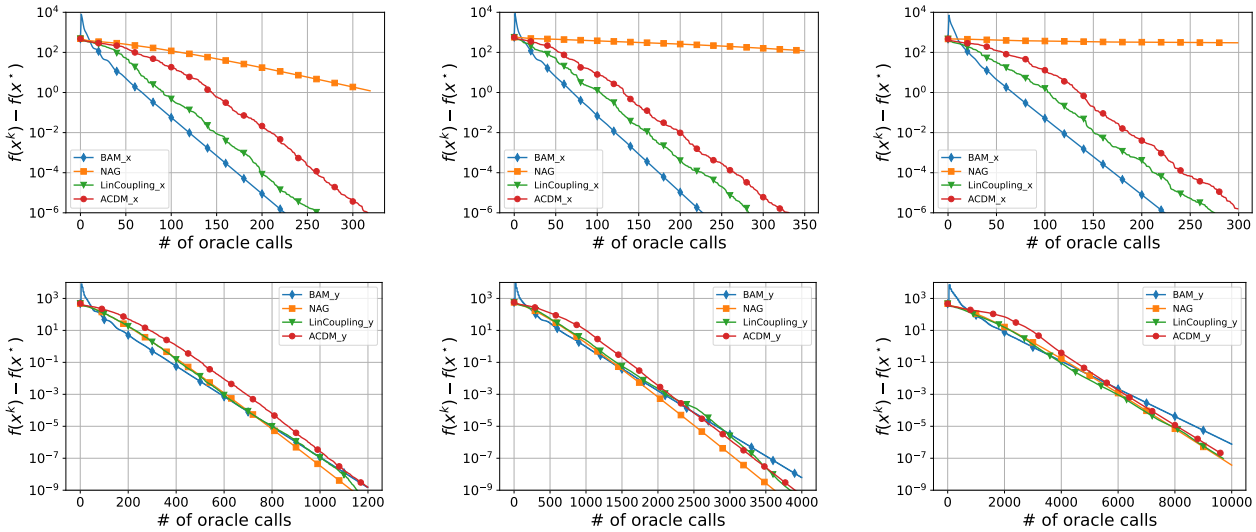

Figure 2: Comparison of Block Accelerated Method (BAM), Nesterov Accelerated Method (NAG), Accelerated Coordinate Descent Method (ACDM) and Linear Coupling method (LinCoupling) on quadratic functions. First line represents rate in terms of the $\nabla_x f(x, y)$ oracle calls and the second one represents rate in terms of the $\nabla_y f(x, y)$ oracle calls. We set $L_y = 500$ (left column), $L_y = 5000$ (middle column) and $L_y = 50000$ (right column).

optimization methods to assess its performance and effectiveness. These methods include the Nesterov Accelerated Method (NAG) [Nesterov, 1983], which is a classical approach for smooth convex optimization, the Accelerated Coordinate Descent Method (ACDM) [Nesterov and Stich, 2017], known for its efficiency in coordinate-wise optimization, and the Linear Coupling Method (LinCoupling) [Allen-Zhu et al., 2016, Gasnikov et al., 2015], which provides a framework for optimizing coupled problems.

6.1 QUADRATIC OBJECTIVES

In our experiments, we begin by considering quadratic functions of the form:

$$f(z) = z^\top A z + b^\top z,$$

where $z = (x, y)^\top$ represents a joint vector consisting of two blocks: x and y. The matrix spectrum is uniformly generated for each block, with eigenvalues for the block x ranging from μ_x to L_x, and eigenvalues for the block y ranging from μ_y to L_y. For this setup, we set $\mu_x = \mu_y = 0.1$, and $L_y = 50$. The dimensions of the blocks are set to $d_x = 100$ for x and $d_y = 10$ for y.

To analyze the impact of varying condition numbers, we adjust the parameter L_y to generate different values of the condition number κ_y. Throughout the experiments, we focus on comparing the number of oracle calls for $\nabla_x f(x, y)$ and $\nabla_y f(x, y)$ across several optimization methods. This allows us to evaluate the efficiency of each method under controlled settings.

6.2 LOGISTIC REGRESSION

In our experiments, we also investigate the logistic regression loss function with two l_2 regularizers for a click-prediction model, defined as:

$$f(x, y) := \frac{1}{n} \sum_{k=1}^{n} \log\left(1 + \exp\left(-\eta^k \langle \xi^k, (x, y) \rangle\right)\right) + \lambda_x \|x\|^2 + \lambda_y \|y\|^2.$$

For this experiment, we used the "a1a" dataset from the LIBSVM collection [Chang and Lin, 2011]. The datasets analyzed in this study are available in the LIBSVM repository. The smoothness constant for this dataset is estimated as $L = 1.567$. We set $d_x = 100$, $d_y = 19$, and $\mu_x = 0.01$. To explore condition numbers, we vary the parameter μ_y. Also, we consider the number of oracle calls to $\nabla_x f(x, y)$ and $\nabla_y f(x, y)$ for comparison across different methods.

6.3 RESULTS

In our experiments, as illustrated in the plots, the new Block Accelerated Method (BAM) demonstrates superior performance in terms of the number of $\nabla_x f(x, y)$ oracle calls for both objective functions across all tested condition numbers. This indicates that the new method is more efficient in terms of computational resources for these oracle calls. Additionally, all accelerated coordinate methods outperform the Nesterov Gradient Method (NAG) by a significant mar-

gin, which serves to validate the theoretical bounds established for these methods.

When considering $\nabla_y f(x, y)$ oracle calls, the performance of BAM is approximately the same as that of other accelerated coordinate methods and the Nesterov Gradient Method. This shows that BAM does not incur a performance penalty when evaluating $\nabla_y f(x, y)$. In scenarios where oracle calls to $\nabla_x f(x, y)$ are particularly costly, BAM can be particularly advantageous due to its reduced communication complexity. Furthermore, the method's ability to be generalized to distributed and federated settings further enhances its practical utility, suggesting that BAM has significant potential for practical applications.

7 DISCUSSION

In this paper, we address a convex optimization problem with a min-min structure:

$$\min_{x,y} f(x, y).$$

Under the assumption that f is L-smooth and μ_x-strongly convex in x, and μ_y-strongly convex in y, we propose a new algorithm, BAM, which requires $\mathcal{O}\left(\sqrt{L/\mu_x}\log\frac{1}{\epsilon}\right)$ calculations of $\nabla_x f$ and $\mathcal{O}\left(\sqrt{L/\mu_y}\log\frac{1}{\epsilon}\right)$ calculations of $\nabla_y f$ to achieve an ϵ-accurate solution. Furthermore, we demonstrate the applicability of BAM to Federated Learning, showing its potential to reduce communication costs while maintaining high efficiency in decentralized settings.

The approach proposed in this paper offers several possibilities for further generalizations. For instance, it can be adapted to mixed oracles, as introduced in [Gladin et al., 2021b], where instead of computing $\nabla_y f$, only the function value $f(x, y)$ is available. Another possible extension is increasing the number of blocks in the optimization problem (currently, we consider only two blocks, x and y) for more complex scenarios. Additionally, BAM can be combined with other techniques, such as composite sliding methods [Lan, 2016, Kovalev et al., 2022], which were mentioned at the outset of the introduction. These possible extensions present promising directions for future research and could lead to further improvements in efficiency and applicability across various domains. Furthermore, the proposed method opens up new avenues for exploring optimization in large-scale distributed systems, where the challenges of data heterogeneity and communication constraints are critical.

Acknowledgements

This work of A. Gasnikov was supported by the Ministry of Economic Development of the RF (code 25-139-66879-1-0003).

References

Zeyuan Allen-Zhu, Zheng Qu, Peter Richtárik, and Yang Yuan. Even faster accelerated coordinate descent using non-uniform sampling. In *International Conference on Machine Learning*, pages 1110–1119. PMLR, 2016.

Amir Beck. *First-order methods in optimization*. SIAM, 2017.

Jérôme Bolte, Lilian Glaudin, Edouard Pauwels, and Mathieu Serrurier. Ah\" olderian backtracking method for min-max and min-min problems. *arXiv preprint arXiv:2007.08810*, 2020.

Yair Carmon, Arun Jambulapati, Yujia Jin, and Aaron Sidford. Recapp: Crafting a more efficient catalyst for convex optimization. In *International Conference on Machine Learning*, pages 2658–2685. PMLR, 2022.

Chih-Chung Chang and Chih-Jen Lin. LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2:27:1–27:27, 2011. Software available at `http://www.csie.ntu.edu.tw/~cjlin/libsvm`.

Fei Chen, Mi Luo, Zhenhua Dong, Zhenguo Li, and Xiuqiang He. Federated meta-learning with fast convergence and efficient communication. *arXiv preprint arXiv:1802.07876*, 2018.

Laurent Condat, Ivan Agarsky, and Peter Richtárik. Provably doubly accelerated federated learning: The first theoretically successful combination of local training and compressed communication. *arXiv preprint arXiv:2210.13277*, 2022.

Joaquin De Cea, J Enrique Fernández, Valerie Dekock, and Alexandra Soto. Solving network equilibrium problems on multimodal urban transportation networks with multiple user classes. *Transport Reviews*, 25(3):293–317, 2005.

Jelena Diakonikolas and Lorenzo Orecchia. Alternating randomized block coordinate descent. In *International Conference on Machine Learning*, pages 1224–1232. PMLR, 2018.

Jelena Diakonikolas and Puqian Wang. Potential function-based framework for minimizing gradients in convex and min-max optimization. *SIAM Journal on Optimization*, 32(3):1668–1697, 2022.

Pavel Dvurechensky, Dmitry Kamzolov, Aleksandr Lukashevich, Soomin Lee, Erik Ordentlich, César A Uribe, and Alexander Gasnikov. Hyperfast second-order local solvers for efficient statistically preconditioned distributed optimization. *EURO Journal on Computational Optimization*, 10:100045, 2022.

Alireza Fallah, Aryan Mokhtari, and Asuman Ozdaglar. Personalized federated learning: A meta-learning approach. *arXiv preprint arXiv:2002.07948*, 2020.

Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *International conference on machine learning*, pages 1126–1135. PMLR, 2017.

Alexander Gasnikov, Pavel Dvurechensky, and Ilnura Usmanova. About accelerated randomized methods. *arXiv preprint arXiv:1508.02182*, 2015.

Alexander Vladimirovich Gasnikov, Yutii Vladimirovich Dorn, Yurii Evgen'evich Nesterov, and Sergei Valer'evich Shpirko. On the three-stage version of stable dynamic model. *Matematicheskoe modelirovanie*, 26(6):34–70, 2014.

Alexander Vladimirovich Gasnikov, EB Gasnikova, Yu E Nesterov, and AV Chernov. Efficient numerical methods for entropy-linear programming problems. *Computational Mathematics and Mathematical Physics*, 56(4): 514–524, 2016.

Alexander Vladimirovich Gasnikov, Darina Mikhailovna Dvinskikh, PE Dvurechensky, DI Kamzolov, Vladislav V Matyukhin, Dmitry A Pasechnyuk, Nazarii Konstantinovich Tupitsa, and Aleksey Vladimirovich Chernov. Accelerated meta-algorithm for convex optimization problems. *Computational Mathematics and Mathematical Physics*, 61(1):17–28, 2021.

Egor Gladin, M Alkousa, and A Gasnikov. Solving convex min-min problems with smoothness and strong convexity in one group of variables and low dimension in the other. *Automation and Remote Control*, 82(10):1679–1691, 2021a.

Egor Gladin, Abdurakhmon Sadiev, Alexander Gasnikov, Pavel Dvurechensky, Aleksandr Beznosikov, and Mohammad Alkousa. Solving smooth min-min and min-max problems by mixed oracle algorithms. In *International Conference on Mathematical Optimization Theory and Operations Research*, pages 19–40. Springer, 2021b.

Eduard Gorbunov, Filip Hanzely, and Peter Richtárik. Local sgd: Unified theory and new efficient methods. In *International Conference on Artificial Intelligence and Statistics*, pages 3556–3564. PMLR, 2021.

Michał Grudzień, Grigory Malinovsky, and Peter Richtárik. Can 5th generation local training methods support client sampling? yes! *arXiv preprint arXiv:2212.14370*, 2022.

Sergey Guminov, Pavel Dvurechensky, Nazarii Tupitsa, and Alexander Gasnikov. On a combination of alternating minimization and nesterov's momentum. In *International Conference on Machine Learning*, pages 3886–3898. PMLR, 2021.

Andrew Hard, Kanishka Rao, Rajiv Mathews, Swaroop Ramaswamy, Françoise Beaufays, Sean Augenstein, Hubert Eichner, Chloé Kiddon, and Daniel Ramage. Federated learning for mobile keyboard prediction. *arXiv preprint arXiv:1811.03604*, 2018.

Anastasiya Ivanova, Dmitry Pasechnyuk, Dmitry Grishchenko, Egor Shulgin, Alexander Gasnikov, and Vladislav Matyukhin. Adaptive catalyst for smooth convex optimization. In *International Conference on Optimization and Applications*, pages 20–37. Springer, 2021.

Anastasiya Ivanova, Pavel Dvurechensky, Evgeniya Vorontsova, Dmitry Pasechnyuk, Alexander Gasnikov, Darina Dvinskikh, and Alexander Tyurin. Oracle complexity separation in convex optimization. *Journal of Optimization Theory and Applications*, 193(1):462–490, 2022.

Sai Praneeth Karimireddy, Satyen Kale, Mehryar Mohri, Sashank Reddi, Sebastian Stich, and Ananda Theertha Suresh. Scaffold: Stochastic controlled averaging for federated learning. In *International Conference on Machine Learning*, pages 5132–5143. PMLR, 2020.

Ahmed Khaled, Konstantin Mishchenko, and Peter Richtárik. Tighter theory for local sgd on identical and heterogeneous data. In *International Conference on Artificial Intelligence and Statistics*, pages 4519–4529. PMLR, 2020.

Donghwan Kim and Jeffrey A Fessler. Optimizing the efficiency of first-order methods for decreasing the gradient of smooth convex functions. *Journal of optimization theory and applications*, 188(1):192–219, 2021.

Anastasia Koloskova, Nicolas Loizou, Sadra Boreiri, Martin Jaggi, and Sebastian Stich. A unified theory of decentralized sgd with changing topology and local updates. In *International Conference on Machine Learning*, pages 5381–5393. PMLR, 2020.

Jakub Konečnỳ, H Brendan McMahan, Felix X Yu, Peter Richtárik, Ananda Theertha Suresh, and Dave Bacon. Federated learning: Strategies for improving communication efficiency. *arXiv preprint arXiv:1610.05492*, 2016.

Dmitry Kovalev and Alexander Gasnikov. The first optimal algorithm for smooth and strongly-convex-strongly-concave minimax optimization. In *Advances in Neural Information Processing Systems*, 2022a.

Dmitry Kovalev and Alexander Gasnikov. The first optimal acceleration of high-order methods in smooth convex optimization. In *Advances in Neural Information Processing Systems*, 2022b.

Dmitry Kovalev, Aleksandr Beznosikov, Ekaterina Dmitrievna Borodich, Alexander Gasnikov,

and Gesualdo Scutari. Optimal gradient sliding and its application to optimal distributed optimization under similarity. In *Advances in Neural Information Processing Systems*, 2022.

Guanghui Lan. Gradient sliding for composite optimization. *Mathematical Programming*, 159(1):201–235, 2016.

Grigory Malinovsky, Kai Yi, and Peter Richtárik. Variance reduced proxskip: Algorithm, theory and application to federated learning. *arXiv preprint arXiv:2207.04338*, 2022.

Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. Communication-efficient learning of deep networks from decentralized data. In *Artificial intelligence and statistics*, pages 1273–1282. PMLR, 2017.

Konstantin Mishchenko, Grigory Malinovsky, Sebastian Stich, and Peter Richtárik. Proxskip: Yes! local gradient steps provably lead to communication acceleration! finally! In *International Conference on Machine Learning*, pages 15750–15769. PMLR, 2022.

Konstantin Mishchenko, Rustem Islamov, Eduard Gorbunov, and Samuel Horváth. Partially personalized federated learning: Breaking the curse of data heterogeneity. *arXiv preprint arXiv:2305.18285*, 2023.

Aritra Mitra, Rayana Jaafar, George J Pappas, and Hamed Hassani. Linear convergence in federated learning: Tackling client heterogeneity and sparse gradients. *Advances in Neural Information Processing Systems*, 34:14606–14619, 2021.

A Nemirovski and D Yudin. *Problem complexity and method efficiency in Optimization*. J. Wiley & Sons, 1983.

Yu Nesterov. Efficiency of coordinate descent methods on huge-scale optimization problems. *SIAM Journal on Optimization*, 22(2):341–362, 2012.

Yurii Nesterov. *Lectures on convex optimization*, volume 137. Springer, 2018.

Yurii Nesterov. Soft clustering by convex electoral model. *Soft Computing*, 24(23):17609–17620, 2020.

Yurii Nesterov. Structural optimization: New perspectives for increasing efficiency of numerical schemes. *nternational conference "Optimization and Applications in Control and Data Science" on the occasion of Boris Polyak's 80th birthday*, May 14, 2015. URL `https://www.mathnet.ru/php/presentation.phtml?option_lang=rus&presentid=11909`.

Yurii Nesterov and Sebastian U Stich. Efficiency of the accelerated coordinate descent method on structured optimization problems. *SIAM Journal on Optimization*, 27 (1):110–123, 2017.

Yurii Nesterov, Alexander Gasnikov, Sergey Guminov, and Pavel Dvurechensky. Primal–dual accelerated gradient methods with small-dimensional relaxation oracle. *Optimization Methods and Software*, 36(4):773–810, 2021.

Yurii Evgen'evich Nesterov. A method of solving a convex programming problem with convergence rate $\mathcal{O}(1/k^2)$. In *Doklady Akademii Nauk*, volume 269, pages 543–547. Russian Academy of Sciences, 1983.

Alex Nichol, Joshua Achiam, and John Schulman. On first-order meta-learning algorithms. *arXiv preprint arXiv:1803.02999*, 2018.

Petr A Ostroukhov. Tensor methods inside mixed oracle for min-min problems. *Computer Research and Modeling*, 14(2):377–398, 2022.

Peter Richtárik and Martin Takáč. Iteration complexity of randomized block-coordinate descent methods for minimizing a composite function. *Mathematical Programming*, 144(1):1–38, 2014.

Nicola Rieke, Jonny Hancox, Wenqi Li, Fausto Milletari, Holger R Roth, Shadi Albarqouni, Spyridon Bakas, Mathieu N Galtier, Bennett A Landman, Klaus Maier-Hein, et al. The future of digital health with federated learning. *NPJ digital medicine*, 3(1):119, 2020.

Karan Singhal, Hakim Sidahmed, Zachary Garrett, Shanshan Wu, John Rush, and Sushant Prakash. Federated reconstruction: Partially local federated learning. *Advances in Neural Information Processing Systems*, 34: 11220–11232, 2021.

Nazarii Tupitsa, Pavel Dvurechensky, Alexander Gasnikov, and Sergey Guminov. Alternating minimization methods for strongly convex optimization. *Journal of Inverse and Ill-posed Problems*, 29(5):721–739, 2021.

Blake Woodworth, Kumar Kshitij Patel, Sebastian Stich, Zhen Dai, Brian Bullins, Brendan Mcmahan, Ohad Shamir, and Nathan Srebro. Is local sgd better than minibatch sgd? In *International Conference on Machine Learning*, pages 10334–10343. PMLR, 2020.

An Optimal Algorithm for Strongly Convex Min-Min Optimization (Supplementary Material)

Dmitry Kovalev[1,2] **Alexander Gasnikov**[3,4,5] **Grigory Malinovsky**[6]

[1]Yandex Research
[2]Ivannikov Institute for System Programming
[3]AI Research Center, Innopolis University
[4]Skolkovo Institute of Science and Technology (Skoltech)
[5]Moscow Institute of Physics and Technology (MIPT)
[6]King Abdullah University of Science and Technology (KAUST)

A PROOFS

Proof of Lemma 2.1. Using Assumption 1.1, we get

$$\begin{aligned}
f(\overline{x}^{k+1},\overline{y}^{k+1}) &\leq f(\underline{x}^k,\overline{y}^{k+1}) + \langle \nabla_x f(\underline{x}^k,\overline{y}^{k+1}), \overline{x}^{k+1}-\underline{x}^k\rangle + \frac{L_x}{2}\|\overline{x}^{k+1}-\underline{x}^k\|^2 \\
&= f(\underline{x}^k,\overline{y}^{k+1}) + \eta_x\alpha\left(\frac{\eta_x\alpha L_x}{2}-1\right)\|\nabla_x f(\underline{x}^k,\overline{y}^{k+1})\|^2 \\
&\leq f(\underline{x}^k,\overline{y}^{k+1}) + \eta_x\alpha\left(\frac{1}{2}-1\right)\|\nabla_x f(\underline{x}^k,\overline{y}^{k+1})\|^2 \\
&\leq f(\underline{x}^k,\overline{y}^{k+1}) - \frac{\eta_x\alpha}{2}\|\nabla_x f(\underline{x}^k,\overline{y}^{k+1})\|^2
\end{aligned}$$

□

Proof of Theorem 2.2. We start our derivation of upper bound from considering the following term: $\eta_x^{-1}\mathcal{R}_x^{k+1} + \eta_y^{-1}\mathcal{R}_y^{k+1}$. Using definition of $\mathcal{R}_x^{k+1}$ and $\mathcal{R}_y^{k+1}$ in Theorem 2.2 we have

$$\eta_x^{-1}\mathcal{R}_x^{k+1} + \eta_y^{-1}\mathcal{R}_y^{k+1} = \eta_x^{-1}\|x^{k+1}-x^*\|^2 + \eta_y^{-1}\|y^{k+1}-y^*\|^2.$$

Let us consider the squared norm of difference $\|x^{k+1}-x^*\|^2$:

$$\begin{aligned}
\|x^{k+1}-x^\star\|^2 &= \|x^{k+1}-x^k+x^k-x^\star\|^2 \\
&= \|x^{k+1}-x^k\|^2 + 2<x^{k+1}-x^k, x^k-x^\star> + \|x^k-x^\star\|^2 \\
&= \|x^{k+1}-x^k\|^2 + 2<x^{k+1}-x^k, x^k-x^{k+1}+x^{k+1}-x^\star> + \|x^k-x^\star\|^2 \\
&= \|x^{k+1}-x^k\|^2 - 2<x^{k+1}-x^k, x^{k+1}-x^k> + 2<x^{k+1}-x^k, x^{k+1}-x^\star> + \|x^k-x^\star\|^2 \\
&= -\|x^{k+1}-x^k\|^2 + 2<x^{k+1}-x^k, x^{k+1}-x^\star> + \|x^k-x^\star\|^2.
\end{aligned}$$

Similarly, we have

$$\|y^{k+1}-y^\star\|^2 = -\|y^{k+1}-y^k\|^2 + 2<y^{k+1}-y^k, y^{k+1}-y^\star> + \|y^k-y^\star\|^2.$$

Combining these equations together we obtain

$$\eta_x^{-1}\mathcal{R}_x^{k+1} + \eta_y^{-1}\mathcal{R}_y^{k+1} = \eta_x^{-1}\mathcal{R}_x^{k} + \eta_y^{-1}\mathcal{R}_y^{k} - \eta_x^{-1}\left\|x^{k+1} - x^k\right\|^2 - \eta_y^{-1}\left\|y^{k+1} - y^k\right\|^2 + 2\eta_x^{-1}\left\langle x^{k+1} - x^k, x^{k+1} - x^*\right\rangle + 2\eta_y^{-1}\left\langle y^{k+1} - y^k, y^{k+1} - y^*\right\rangle.$$

Next, we recall that the update rules for new iterates are the following:

$$x^{k+1} = x^k + \alpha\left(\underline{x}^k - x^{k+1}\right) - \eta_x \nabla_x f\left(\underline{x}^k, \bar{y}^{k+1}\right)$$
$$y^{k+1} = y^k + \alpha\left(\bar{y}^{k+1} - y^{k+1}\right) - \eta_y \nabla_y f\left(\underline{x}^k, \bar{y}^{k+1}\right).$$

We can extract the difference between iterates:

$$x^{k+1} - x^k = \alpha\left(\underline{x}^k - x^{k+1}\right) - \eta_x \nabla_x f\left(\underline{x}^k, \bar{y}^{k+1}\right) \qquad y^{k+1} - y^k = \alpha\left(\bar{y}^{k+1} - y^{k+1}\right) - \eta_y \nabla_y f\left(\underline{x}^k, \bar{y}^{k+1}\right).$$

Now we can plug these identities into previous our main equation and obtain

$$\begin{aligned}\eta_x^{-1}\mathcal{R}_x^{k+1} + \eta_y^{-1}\mathcal{R}_y^{k+1} = {} & \eta_x^{-1}\mathcal{R}_x^{k} + \eta_y^{-1}\mathcal{R}_y^{k} - \eta_x^{-1}\left\|x^{k+1} - x^k\right\|^2 - \eta_y^{-1}\left\|y^{k+1} - y^k\right\|^2 \\ & + 2\eta_x^{-1}\alpha\left\langle \underline{x}_k - x^{k+1}, x^{k+1} - x^*\right\rangle + 2\eta_y^{-1}\left\langle \bar{y}^{k+1} - y^{k+1}, y^{k+1} - y^*\right\rangle \\ & - 2\left\langle \nabla_x f\left(\underline{x}^k, \bar{y}^{k+1}\right), x^{k+1} - x^*\right\rangle - 2\left\langle \nabla_y f\left(\underline{x}^k, \bar{y}^{k+1}\right), y^{k+1} - y^*\right\rangle.\end{aligned}$$

Next, we need to use standard algebraic trick:

$$2 < (a-b), (b-c) >= \|a-c\|^2 - \|b-c\|^2 - \|b-a\|^2.$$

We can quickly proof this statement:

$$\begin{aligned}\|a-c\|^2 - \|b-c\|^2 - \|b-a\|^2 &= \|a\|^2 - 2 < a, c > + \|c\|^2 \\ &\quad - \left(\|b\|^2 - 2 < b, c > + \|c\|^2\right) - \left(\|b\|^2 - 2 < a, b > + \|a\|^2\right) \\ &= -2 < a, c > + 2 < b, c > + 2 < a, b > - 2 < b, b > \\ &= -2 < a-b, c > + 2 < a-b, b >= 2 < a-b, b-c >\end{aligned}$$

.

We apply this identity to our main equation for $2\eta_x^{-1}\alpha\left\langle \underline{x}_k - x^{k+1}, x^{k+1} - x^*\right\rangle + 2\eta_y^{-1}\left\langle \bar{y}^{k+1} - y^{k+1}, y^{k+1} - y^*\right\rangle$ and obtain the following:

$$\begin{aligned}\eta_x^{-1}\mathcal{R}_x^{k+1} + \eta_y^{-1}\mathcal{R}_y^{k+1} = {} & \eta_x^{-1}\mathcal{R}_x^{k} + \eta_y^{-1}\mathcal{R}_y^{k} - \eta_x^{-1}\left\|x^{k+1} - x^k\right\|^2 - \eta_y^{-1}\left\|y^{k+1} - y^k\right\|^2 \\ & + \eta_x^{-1}\alpha\left(\left\|\underline{x}^k - x^*\right\|^2 - \left\|x^{k+1} - x^*\right\|^2 - \left\|x^{k+1} - \underline{x}^k\right\|^2\right) \\ & + \eta_y^{-1}\alpha\left(\left\|\bar{y}^{k+1} - y^*\right\|^2 - \left\|y^{k+1} - y^*\right\|^2 - \left\|y^{k+1} - \bar{y}^{k+1}\right\|^2\right) \\ & - 2\left\langle \nabla_x f\left(\underline{x}^k, \bar{y}^{k+1}\right), x^{k+1} - x^*\right\rangle - 2\left\langle \nabla_y f\left(\underline{x}^k, \bar{y}^{k+1}\right), y^{k+1} - y^*\right\rangle.\end{aligned}$$

Since the norm of vector is nonnegative, then we have $A - \|b\| \leq A$, so we can get rid of $-\left\|x^{k+1} - \underline{x}^k\right\|^2$ and $-\left\|y^{k+1} - \bar{y}^{k+1}\right\|^2$:

$$\begin{aligned}\eta_x^{-1}\mathcal{R}_x^{k+1} + \eta_y^{-1}\mathcal{R}_y^{k+1} \leq {} & \eta_x^{-1}\mathcal{R}_x^{k} + \eta_y^{-1}\mathcal{R}_y^{k} - \eta_x^{-1}\left\|x^{k+1} - x^k\right\|^2 - \eta_y^{-1}\left\|y^{k+1} - y^k\right\|^2 \\ & + \eta_x^{-1}\alpha\left(\left\|\underline{x}^k - x^*\right\|^2 - \left\|x^{k+1} - x^*\right\|^2\right) \\ & + \eta_y^{-1}\alpha\left(\left\|\bar{y}^{k+1} - y^*\right\|^2 - \left\|y^{k+1} - y^*\right\|^2\right) \\ & - 2\left\langle \nabla_x f\left(\underline{x}^k, \bar{y}^{k+1}\right), x^{k+1} - x^*\right\rangle - 2\left\langle \nabla_y f\left(\underline{x}^k, \bar{y}^{k+1}\right), y^{k+1} - y^*\right\rangle.\end{aligned}$$

Starting from previous bound on $\eta_x^{-1}\mathcal{R}_x^{k+1} + \eta_y^{-1}\mathcal{R}_y^{k+1}$ and let us open the brackets:

$$\begin{aligned}\eta_x^{-1}\mathcal{R}_x^{k+1} + \eta_y^{-1}\mathcal{R}_y^{k+1} \leq{}& \eta_x^{-1}\mathcal{R}_x^{k} + \eta_y^{-1}\mathcal{R}_y^{k} - \eta_x^{-1}\left\|x^{k+1} - x^k\right\|^2 - \eta_y^{-1}\left\|y^{k+1} - y^k\right\|^2 \\ &+ \eta_x^{-1}\alpha\left\|\underline{x}^k - x^*\right\|^2 - \eta_x^{-1}\alpha\left\|x^{k+1} - x^*\right\|^2 \\ &+ \eta_y^{-1}\alpha\left\|\bar{y}^{k+1} - y^*\right\|^2 - \eta_y^{-1}\alpha\left\|y^{k+1} - y^*\right\|^2 \\ &- 2\left\langle\nabla_x f\left(\underline{x}^k, \bar{y}^{k+1}\right), x^{k+1} - x^*\right\rangle - 2\left\langle\nabla_y f\left(\underline{x}^k, \bar{y}^{k+1}\right), y^{k+1} - y^*\right\rangle.\end{aligned}$$

Next, we put $-\eta_x^{-1}\alpha\left\|x^{k+1} - x^*\right\|^2$ and $-\eta_y^{-1}\alpha\left\|y^{k+1} - y^*\right\|^2$ to the left side, and this leads to

$$\begin{aligned}(1+\alpha)\left(\eta_x^{-1}\mathcal{R}_x^{k+1} + \eta_y^{-1}\mathcal{R}_y^{k+1}\right) \leq{}& \eta_x^{-1}\mathcal{R}_x^{k} + \eta_y^{-1}\mathcal{R}_y^{k} - \eta_x^{-1}\left\|x^{k+1} - x^k\right\|^2 - \eta_y^{-1}\left\|y^{k+1} - y^k\right\|^2 \\ &+ \eta_x^{-1}\alpha\left\|\underline{x}^k - x^*\right\|^2 + \eta_y^{-1}\alpha\left\|\bar{y}^{k+1} - y^*\right\|^2 \\ &- 2\left\langle\nabla_x f\left(\underline{x}^k, \bar{y}^{k+1}\right), x^{k+1} - x^*\right\rangle - 2\left\langle\nabla_y f\left(\underline{x}^k, \bar{y}^{k+1}\right), y^{k+1} - y^*\right\rangle.\end{aligned}$$

Next we add and subtract vectors x^k and y^k in inner products and use the following identity: $-2 < a, b - c > -2 < a, c - d >= 2 < a, b - d >$, so we have

$$\begin{aligned}(1+\alpha)\left(\eta_x^{-1}\mathcal{R}_x^{k+1} + \eta_y^{-1}\mathcal{R}_y^{k+1}\right) \leq{}& \eta_x^{-1}\mathcal{R}_x^{k} + \eta_y^{-1}\mathcal{R}_y^{k} - \eta_x^{-1}\left\|x^{k+1} - x^k\right\|^2 - \eta_y^{-1}\left\|y^{k+1} - y^k\right\|^2 \\ &+ \eta_x^{-1}\alpha\left\|\underline{x}^k - x^*\right\|^2 + \eta_y^{-1}\alpha\left\|\bar{y}^{k+1} - y^*\right\|^2 \\ &- 2\left\langle\nabla_x f\left(\underline{x}^k, \bar{y}^{k+1}\right), x^{k+1} - x^k\right\rangle - 2\left\langle\nabla_x f\left(\underline{x}^k, \bar{y}^{k+1}\right), x^k - x^*\right\rangle \\ &- 2\left\langle\nabla_y f\left(\underline{x}^k, \bar{y}^{k+1}\right), y^{k+1} - y^k\right\rangle - 2\left\langle\nabla_y f\left(\underline{x}^k, \bar{y}^{k+1}\right), y^k - y^*\right\rangle.\end{aligned}$$

Next, we need to apply Young's inequality for inner products (also known as the Peter–Paul inequality):

$$< a, b >\leq \frac{\|a\|^2}{2c_1} + \frac{\|b\|^2 c_1}{2}.$$

If we use $a' = -a$, then we also have

$$- < a, b >\leq \frac{\|a\|^2}{2c_1} + \frac{\|b\|^2 c_1}{2}.$$

We apply this inequality and obtain

$$\begin{aligned}(1+\alpha)\left(\eta_x^{-1}\mathcal{R}_x^{k+1} + \eta_y^{-1}\mathcal{R}_y^{k+1}\right) \leq{}& \eta_x^{-1}\mathcal{R}_x^{k} + \eta_y^{-1}\mathcal{R}_y^{k} - \eta_x^{-1}\left\|x^{k+1} - x^k\right\|^2 - \eta_y^{-1}\left\|y^{k+1} - y^k\right\|^2 \\ &+ \eta_x^{-1}\alpha\left\|\underline{x}^k - x^*\right\|^2 + \eta_y^{-1}\alpha\left\|\bar{y}^{k+1} - y^*\right\|^2 \\ &+ \eta_x^{-1}\left\|x^{k+1} - x^k\right\|^2 + \eta_x\left\|\nabla_x f\left(\underline{x}^k, \bar{y}^{k+1}\right)\right\|^2 - 2\left\langle\nabla_x f\left(\underline{x}^k, \bar{y}^{k+1}\right), x^k - x^*\right\rangle \\ &+ \eta_y^{-1}\left\|y^{k+1} - y^k\right\|^2 + \eta_y\left\|\nabla_y f\left(\underline{x}^k, \bar{y}^{k+1}\right)\right\|^2 - 2\left\langle\nabla_y f\left(\underline{x}^k, \bar{y}^{k+1}\right), y^k - y^*\right\rangle.\end{aligned}$$

Note that $\eta_x^{-1}\left\|x^{k+1} - x^k\right\|^2$ and $\eta_y^{-1}\left\|y^{k+1} - y^k\right\|^2$ cancel out, since we also have the terms $-\eta_x^{-1}\left\|x^{k+1} - x^k\right\|^2$ and $-\eta_y^{-1}\left\|y^{k+1} - y^k\right\|^2$, so we have

$$\begin{aligned}(1+\alpha)\left(\eta_x^{-1}\mathcal{R}_x^{k+1} + \eta_y^{-1}\mathcal{R}_y^{k+1}\right) \leq{}& \eta_x^{-1}\mathcal{R}_x^{k} + \eta_y^{-1}\mathcal{R}_y^{k} + \eta_x^{-1}\alpha\left\|\underline{x}^k - x^*\right\|^2 + \eta_y^{-1}\alpha\left\|\bar{y}^{k+1} - y^*\right\|^2 \\ &+ \eta_x\left\|\nabla_x f\left(\underline{x}^k, \bar{y}^{k+1}\right)\right\|^2 + \eta_y\left\|\nabla_y f\left(\underline{x}^k, \bar{y}^{k+1}\right)\right\|^2 \\ &- 2\left\langle\nabla_x f\left(\underline{x}^k, \bar{y}^{k+1}\right), x^k - x^*\right\rangle - 2\left\langle\nabla_y f\left(\underline{x}^k, \bar{y}^{k+1}\right), y^k - y^*\right\rangle.\end{aligned}$$

Next, we need to add and subtract vectors $\underline{x}^k$ and $\bar{y}^{k+1}$ in inner products, so we have

$$\begin{aligned}(1+\alpha)\left(\eta_x^{-1}\mathcal{R}_x^{k+1}+\eta_y^{-1}\mathcal{R}_y^{k+1}\right) \leq\ & \eta_x^{-1}\mathcal{R}_x^{k}+\eta_y^{-1}\mathcal{R}_y^{k}+\eta_x^{-1}\alpha\left\|\underline{x}^k-x^*\right\|^2+\eta_y^{-1}\alpha\left\|\bar{y}^{k+1}-y^*\right\|^2\\ &+\eta_x\left\|\nabla_x f\left(\underline{x}^k,\bar{y}^{k+1}\right)\right\|^2+\eta_y\left\|\nabla_y f\left(\underline{x}^k,\bar{y}^{k+1}\right)\right\|^2\\ &-2\left\langle\nabla_x f\left(\underline{x}^k,\bar{y}^{k+1}\right),\underline{x}^k-x^*\right\rangle-2\left\langle\nabla_y f\left(\underline{x}^k,\bar{y}^{k+1}\right),\bar{y}^{k+1}-y^*\right\rangle\\ &-2\left\langle\nabla_x f\left(\underline{x}^k,\bar{y}^{k+1}\right),x^k-\underline{x}^k\right\rangle-2\left\langle\nabla_y f\left(\underline{x}^k,\bar{y}^{k+1}\right),y^k-\bar{y}^{k+1}\right\rangle.\end{aligned}$$

Now we are ready to apply strong convexity Assumption 1.2 specifically for terms $-2\left\langle\nabla_x f\left(\underline{x}^k,\bar{y}^{k+1}\right),\underline{x}^k-x^*\right\rangle-2\left\langle\nabla_y f\left(\underline{x}^k,\bar{y}^{k+1}\right),\bar{y}^{k+1}-y^*\right\rangle$. This allows us to obtain the following inequality:

$$\begin{aligned}(1+\alpha)\left(\eta_x^{-1}\mathcal{R}_x^{k+1}+\eta_y^{-1}\mathcal{R}_y^{k+1}\right) \leq\ & \eta_x^{-1}\mathcal{R}_x^{k}+\eta_y^{-1}\mathcal{R}_y^{k}+\eta_x^{-1}\alpha\left\|\underline{x}^k-x^*\right\|^2+\eta_y^{-1}\alpha\left\|\bar{y}^{k+1}-y^*\right\|^2\\ &+\eta_x\left\|\nabla_x f\left(\underline{x}^k,\bar{y}^{k+1}\right)\right\|^2+\eta_y\left\|\nabla_y f\left(\underline{x}^k,\bar{y}^{k+1}\right)\right\|^2\\ &+2\left(f\left(x^*,y^*\right)-f\left(\underline{x}^k,\bar{y}^{k+1}\right)\right)-\mu_x\left\|\underline{x}^k-x^*\right\|^2-\mu_y\left\|\bar{y}^{k+1}-y^*\right\|^2\\ &-2\left\langle\nabla_x f\left(\underline{x}^k,\bar{y}^{k+1}\right),x^k-\underline{x}^k\right\rangle-2\left\langle\nabla_y f\left(\underline{x}^k,\bar{y}^{k+1}\right),y^k-\bar{y}^{k+1}\right\rangle.\end{aligned}$$

After rearranging terms we obtain

$$\begin{aligned}(1+\alpha)\left(\eta_x^{-1}\mathcal{R}_x^{k+1}+\eta_y^{-1}\mathcal{R}_y^{k+1}\right) \leq\ & \eta_x^{-1}\mathcal{R}_x^{k}+\eta_y^{-1}\mathcal{R}_y^{k}+\left(\eta_x^{-1}\alpha-\mu_x\right)\left\|\underline{x}^k-x^*\right\|^2+\left(\eta_y^{-1}\alpha-\mu_y\right)\left\|\bar{y}^{k+1}-y^*\right\|^2\\ &+\eta_x\left\|\nabla_x f\left(\underline{x}^k,\bar{y}^{k+1}\right)\right\|^2+\eta_y\left\|\nabla_y f\left(\underline{x}^k,\bar{y}^{k+1}\right)\right\|^2+2\left(f\left(x^*,y^*\right)-f\left(\underline{x}^k,\bar{y}^{k+1}\right)\right)\\ &-2\left\langle\nabla_x f\left(\underline{x}^k,\bar{y}^{k+1}\right),x^k-\underline{x}^k\right\rangle-2\left\langle\nabla_y f\left(\underline{x}^k,\bar{y}^{k+1}\right),y^k-\bar{y}^{k+1}\right\rangle.\end{aligned}$$

Next, we use the update rules: $\underline{x}^k=\alpha x^k+(1-\alpha)\overline{x}^k$ and $\underline{y}^k=\alpha y^k+(1-\alpha)\overline{y}^k$. From these lines we derive $x^k=\frac{\underline{x}^k-(1-\alpha)\bar{x}^k}{\alpha}$ and $y^k=\frac{\underline{y}^k-(1-\alpha)\bar{y}^k}{\alpha}$ and obtain

$$\begin{aligned}(1+\alpha)\left(\eta_x^{-1}\mathcal{R}_x^{k+1}+\eta_y^{-1}\mathcal{R}_y^{k+1}\right) \leq\ & \eta_x^{-1}\mathcal{R}_x^{k}+\eta_y^{-1}\mathcal{R}_y^{k}+(\eta_x^{-1}\alpha-\mu_x)\|\underline{x}^k-x^*\|^2+(\eta_y^{-1}\alpha-\mu_y)\|\overline{y}^{k+1}-y^*\|^2\\ &+\eta_x\|\nabla_x f(\underline{x}^k,\overline{y}^{k+1})\|^2+\eta_y\|\nabla_y f(\underline{x}^k,\overline{y}^{k+1})\|^2+2\left(f(x^*,y^*)-f(\underline{x}^k,\overline{y}^{k+1})\right)\\ &+\frac{2(1-\alpha)}{\alpha}\langle\nabla_x f(\underline{x}^k,\overline{y}^{k+1}),\overline{x}^k-\underline{x}^k\rangle\\ &-\langle\nabla_y f(\underline{x}^k,\overline{y}^{k+1}),\frac{2}{\alpha}(\underline{y}^k-(1-\alpha)\overline{y}^k)-2\overline{y}^{k+1}\rangle\\ =\ & \eta_x^{-1}\mathcal{R}_x^{k}+\eta_y^{-1}\mathcal{R}_y^{k}+(\eta_x^{-1}\alpha-\mu_x)\|\underline{x}^k-x^*\|^2+(\eta_y^{-1}\alpha-\mu_y)\|\overline{y}^{k+1}-y^*\|^2\\ &+\eta_x\|\nabla_x f(\underline{x}^k,\overline{y}^{k+1})\|^2+\eta_y\|\nabla_y f(\underline{x}^k,\overline{y}^{k+1})\|^2+2\left(f(x^*,y^*)-f(\underline{x}^k,\overline{y}^{k+1})\right)\\ &+\frac{2(1-\alpha)}{\alpha}\langle\nabla_x f(\underline{x}^k,\overline{y}^{k+1}),\overline{x}^k-\underline{x}^k\rangle\\ &+\frac{2}{\alpha}\langle\nabla_y f(\underline{x}^k,\overline{y}^{k+1}),\overline{y}^{k+1}-\underline{y}^k+(1-\alpha)\overline{y}^k-(1-\alpha)\overline{y}^{k+1}\rangle\\ =\ & \eta_x^{-1}\mathcal{R}_x^{k}+\eta_y^{-1}\mathcal{R}_y^{k}+(\eta_x^{-1}\alpha-\mu_x)\|\underline{x}^k-x^*\|^2+(\eta_y^{-1}\alpha-\mu_y)\|\overline{y}^{k+1}-y^*\|^2\\ &+\eta_x\|\nabla_x f(\underline{x}^k,\overline{y}^{k+1})\|^2+\eta_y\|\nabla_y f(\underline{x}^k,\overline{y}^{k+1})\|^2+2\left(f(x^*,y^*)-f(\underline{x}^k,\overline{y}^{k+1})\right)\\ &+\frac{2(1-\alpha)}{\alpha}\left(\langle\nabla_x f(\underline{x}^k,\overline{y}^{k+1}),\overline{x}^k-\underline{x}^k\rangle+\langle\nabla_y f(\underline{x}^k,\overline{y}^{k+1}),\overline{y}^k-\overline{y}^{k+1}\rangle\right)\\ &+\frac{2}{\alpha}\langle\nabla_y f(\underline{x}^k,\overline{y}^{k+1}),\overline{y}^{k+1}-\underline{y}^k\rangle.\end{aligned}$$

Next, we use convexity of $f(x,y)$ and apply for the term $\big(\langle\nabla_x f(\underline{x}^k,\overline{y}^{k+1}),\overline{x}^k-\underline{x}^k\rangle+\langle\nabla_y f(\underline{x}^k,\overline{y}^{k+1}),\overline{y}^k-\overline{y}^{k+1}\rangle\big)$:

$$\begin{aligned}(1+\alpha)\left(\eta_x^{-1}\mathcal{R}_x^{k+1}+\eta_y^{-1}\mathcal{R}_y^{k+1}\right) &\leq \eta_x^{-1}\mathcal{R}_x^{k}+\eta_y^{-1}\mathcal{R}_y^{k}+(\eta_x^{-1}\alpha-\mu_x)\|\underline{x}^k-x^*\|^2+(\eta_y^{-1}\alpha-\mu_y)\|\overline{y}^{k+1}-y^*\|^2\\ &+\eta_x\|\nabla_x f(\underline{x}^k,\overline{y}^{k+1})\|^2+\eta_y\|\nabla_y f(\underline{x}^k,\overline{y}^{k+1})\|^2+2\left(f(x^*,y^*)-f(\underline{x}^k,\overline{y}^{k+1})\right)\\ &+\frac{2(1-\alpha)}{\alpha}\left(f(\overline{x}^k,\overline{y}^k)-f(\underline{x}^k,\overline{y}^{k+1})\right)+\frac{2}{\alpha}\langle\nabla_y f(\underline{x}^k,\overline{y}^{k+1}),\overline{y}^{k+1}-\underline{y}^k\rangle.\end{aligned}$$

After reshuffling of terms we obtain

$$\begin{aligned}&(1+\alpha)\left(\eta_x^{-1}\mathcal{R}_x^{k+1}+\eta_y^{-1}\mathcal{R}_y^{k+1}\right)\\ &\qquad\leq \eta_x^{-1}\mathcal{R}_x^{k}+\eta_y^{-1}\mathcal{R}_y^{k}+(\eta_x^{-1}\alpha-\mu_x)\|\underline{x}^k-x^*\|^2+(\eta_y^{-1}\alpha-\mu_y)\|\overline{y}^{k+1}-y^*\|^2\\ &\qquad+\eta_x\|\nabla_x f(\underline{x}^k,\overline{y}^{k+1})\|^2+\eta_y\|\nabla_y f(\underline{x}^k,\overline{y}^{k+1})\|^2\\ &\qquad+2f(x^*,y^*)+\frac{2(1-\alpha)}{\alpha}f(\overline{x}^k,\overline{y}^k)-\frac{2}{\alpha}f(\underline{x}^k,\overline{y}^{k+1})\\ &\qquad+\frac{2}{\alpha}\langle\nabla_y f(\underline{x}^k,\overline{y}^{k+1}),\overline{y}^{k+1}-\underline{y}^k\rangle.\end{aligned}$$

Using Lemma 2.1, we obtain

$$\begin{aligned}(1+\alpha)\left(\eta_x^{-1}\mathcal{R}_x^{k+1}+\eta_y^{-1}\mathcal{R}_y^{k+1}\right) &\leq \eta_x^{-1}\mathcal{R}_x^{k}+\eta_y^{-1}\mathcal{R}_y^{k}+(\eta_x^{-1}\alpha-\mu_x)\|\underline{x}^k-x^*\|^2+(\eta_y^{-1}\alpha-\mu_y)\|\overline{y}^{k+1}-y^*\|^2\\ &+\eta_x\|\nabla_x f(\underline{x}^k,\overline{y}^{k+1})\|^2+\eta_y\|\nabla_y f(\underline{x}^k,\overline{y}^{k+1})\|^2\\ &+2f(x^*,y^*)+\frac{2(1-\alpha)}{\alpha}f(\overline{x}^k,\overline{y}^k)\\ &-\frac{2}{\alpha}\left(f(\overline{x}^{k+1},\overline{y}^{k+1})+\frac{\eta_x\alpha}{2}\|\nabla_x f(\underline{x}^k,\overline{y}^{k+1})\|^2\right)\\ &+\frac{2}{\alpha}\langle\nabla_y f(\underline{x}^k,\overline{y}^{k+1}),\overline{y}^{k+1}-\underline{y}^k\rangle.\end{aligned}$$

Since $2f(x^*,y^*)=\frac{2}{\alpha}f(x^*,y^*)-\frac{2(1-\alpha)}{\alpha}f(x^*,y^*)$ we have

$$\begin{aligned}(1+\alpha)\left(\eta_x^{-1}\mathcal{R}_x^{k+1}+\eta_y^{-1}\mathcal{R}_y^{k+1}\right) &\leq \eta_x^{-1}\mathcal{R}_x^{k}+\eta_y^{-1}\mathcal{R}_y^{k}+(\eta_x^{-1}\alpha-\mu_x)\|\underline{x}^k-x^*\|^2+(\eta_y^{-1}\alpha-\mu_y)\|\overline{y}^{k+1}-y^*\|^2\\ &+\frac{2(1-\alpha)}{\alpha}\left(f(\overline{x}^k,\overline{y}^k)-f(x^*,y^*)\right)-\frac{2}{\alpha}\left(f(\overline{x}^{k+1},\overline{y}^{k+1})-f(x^*,y^*)\right)\\ &+\eta_y\|\nabla_y f(\underline{x}^k,\overline{y}^{k+1})\|^2+\frac{2}{\alpha}\langle\nabla_y f(\underline{x}^k,\overline{y}^{k+1}),\overline{y}^{k+1}-\underline{y}^k\rangle.\end{aligned}$$

Next, we use $\frac{\eta_y}{\eta_y}=1$ and obtain

$$\begin{aligned}(1+\alpha)\left(\eta_x^{-1}\mathcal{R}_x^{k+1}+\eta_y^{-1}\mathcal{R}_y^{k+1}\right) &\leq \eta_x^{-1}\mathcal{R}_x^{k}+\eta_y^{-1}\mathcal{R}_y^{k}+(\eta_x^{-1}\alpha-\mu_x)\|\underline{x}^k-x^*\|^2+(\eta_y^{-1}\alpha-\mu_y)\|\overline{y}^{k+1}-y^*\|^2\\ &+\frac{2(1-\alpha)}{\alpha}\left(f(\overline{x}^k,\overline{y}^k)-f(x^*,y^*)\right)-\frac{2}{\alpha}\left(f(\overline{x}^{k+1},\overline{y}^{k+1})-f(x^*,y^*)\right)\\ &+\eta_y\|\nabla_y f(\underline{x}^k,\overline{y}^{k+1})\|^2+2\eta_y\langle\nabla_y f(\underline{x}^k,\overline{y}^{k+1}),(\eta_y\alpha)^{-1}(\overline{y}^{k+1}-\underline{y}^k)\rangle.\end{aligned}$$

Using the fact that $2\langle a,b\rangle=\|a+b\|^2-\|a\|^2-\|b\|^2$ we get

$$\begin{aligned}(1+\alpha)\left(\eta_x^{-1}\mathcal{R}_x^{k+1}+\eta_y^{-1}\mathcal{R}_y^{k+1}\right) &\leq \eta_x^{-1}\mathcal{R}_x^{k}+\eta_y^{-1}\mathcal{R}_y^{k}+(\eta_x^{-1}\alpha-\mu_x)\|\underline{x}^k-x^*\|^2+(\eta_y^{-1}\alpha-\mu_y)\|\overline{y}^{k+1}-y^*\|^2\\ &+\frac{2(1-\alpha)}{\alpha}\left(f(\overline{x}^k,\overline{y}^k)-f(x^*,y^*)\right)-\frac{2}{\alpha}\left(f(\overline{x}^{k+1},\overline{y}^{k+1})-f(x^*,y^*)\right)\\ &+\eta_y\|\nabla_y f(\underline{x}^k,\overline{y}^{k+1})\|^2+\eta_y\|\nabla_y f(\underline{x}^k,\overline{y}^{k+1})+(\eta_y\alpha)^{-1}(\overline{y}^{k+1}-\underline{y}^k)\|^2\\ &-\eta_y\|\nabla_y f(\underline{x}^k,\overline{y}^{k+1})\|^2-\eta_y^{-1}\alpha^{-2}\|\overline{y}^{k+1}-\underline{y}^k\|^2.\end{aligned}$$

After rearranging terms we get

$$\begin{aligned}(1+\alpha)\left(\eta_x^{-1}\mathcal{R}_x^{k+1}+\eta_y^{-1}\mathcal{R}_y^{k+1}\right) \leq \eta_x^{-1}\mathcal{R}_x^{k}+\eta_y^{-1}\mathcal{R}_y^{k}+(\eta_x^{-1}\alpha-\mu_x)\|\underline{x}^k-x^*\|^2+(\eta_y^{-1}\alpha-\mu_y)\|\overline{y}^{k+1}-y^*\|^2\\ +\frac{2(1-\alpha)}{\alpha}\left(f(\overline{x}^k,\overline{y}^k)-f(x^*,y^*)\right)-\frac{2}{\alpha}\left(f(\overline{x}^{k+1},\overline{y}^{k+1})-f(x^*,y^*)\right)\\ +\eta_y\left(\|\nabla_y f(\underline{x}^k,\overline{y}^{k+1})+(\eta_y\alpha)^{-1}(\overline{y}^{k+1}-\underline{y}^k)\|^2-(\eta_y\alpha)^{-2}\|\overline{y}^{k+1}-\underline{y}^k\|^2\right).\end{aligned}$$

Using inequality (6), we get

$$\begin{aligned}(1+\alpha)\left(\eta_x^{-1}\mathcal{R}_x^{k+1}+\eta_y^{-1}\mathcal{R}_y^{k+1}\right) \leq \eta_x^{-1}\mathcal{R}_x^{k}+\eta_y^{-1}\mathcal{R}_y^{k}+(\eta_x^{-1}\alpha-\mu_x)\|\underline{x}^k-x^*\|^2+(\eta_y^{-1}\alpha-\mu_y)\|\overline{y}^{k+1}-y^*\|^2\\ +\frac{2(1-\alpha)}{\alpha}\left(f(\overline{x}^k,\overline{y}^k)-f(x^*,y^*)\right)-\frac{2}{\alpha}\left(f(\overline{x}^{k+1},\overline{y}^{k+1})-f(x^*,y^*)\right).\end{aligned}$$

Using the choice of parameters η_x, η_y, α, we get

$$\begin{aligned}(1+\alpha)\left(\eta_x^{-1}\mathcal{R}_x^{k+1}+\eta_y^{-1}\mathcal{R}_y^{k+1}\right) \leq \eta_x^{-1}\mathcal{R}_x^{k}+\eta_y^{-1}\mathcal{R}_y^{k}+\frac{2(1-\alpha)}{\alpha}\left(f(\overline{x}^k,\overline{y}^k)-f(x^*,y^*)\right)\\ -\frac{2}{\alpha}\left(f(\overline{x}^{k+1},\overline{y}^{k+1})-f(x^*,y^*)\right).\end{aligned}$$

After rearranging, we get

$$\begin{aligned}\Psi^{k+1} &\leq \eta_x^{-1}\mathcal{R}_x^{k}+\eta_y^{-1}\mathcal{R}_y^{k}+\frac{2(1-\alpha)}{\alpha}\left(f(\overline{x}^k,\overline{y}^k)-f(x^*,y^*)\right)\\ &\leq \eta_x^{-1}\mathcal{R}_x^{k}+\eta_y^{-1}\mathcal{R}_y^{k}+\frac{2(1+\alpha)^{-1}}{\alpha}\left(f(\overline{x}^k,\overline{y}^k)-f(x^*,y^*)\right)\\ &= (1+\alpha)^{-1}\Psi^k.\end{aligned}$$

□

Proof of Corollary 3.2. Using inequality (14) and (15), we get

$$\begin{aligned}\|\nabla A^k(\overline{y}^{k+1})\| &\leq \frac{(\eta_y\alpha)^{-1}}{2}\|\underline{y}^k-\arg\min_{y\in\mathbb{R}^{d_y}}A^k(y)\|\\ &\leq \frac{(\eta_y\alpha)^{-1}}{2}\|\overline{y}^{k+1}-\underline{y}^k\|+\frac{(\eta_y\alpha)^{-1}}{2}\|\overline{y}^{k+1}-\arg\min_{y\in\mathbb{R}^{d_y}}A^k(y)\|.\end{aligned}$$

Function $A^k(y)$ is $(\eta_y\alpha)^{-1}$-strongly convex which implies

$$(\eta_y\alpha)^{-1}\|\overline{y}^{k+1}-\arg\min_{y\in\mathbb{R}^{d_y}}A^k(y)\| \leq \|\nabla A^k(\overline{y}^{k+1})-\nabla A^k(\arg\min_{y\in\mathbb{R}^{d_y}}A^k(y))\| = \|\nabla A^k(\overline{y}^{k+1})\|. \tag{18}$$

Hence,

$$\|\nabla A^k(\overline{y}^{k+1})\| \leq \frac{(\eta_y\alpha)^{-1}}{2}\|\overline{y}^{k+1}-\underline{y}^k\|+\frac{1}{2}\|\nabla A^k(\overline{y}^{k+1})\|.$$

Rearranging concludes the proof. □