Proceedings of Machine Learning Research vol ???:1–35, 2025 38th Annual Conference on Learning Theory

Gradient Methods with Online Scaling

Wenzhi Gao GWZ@STANFORD.EDU
ICME, Stanford University

Ya-Chi Chu YCCHU97@STANFORD.EDU
Department of Mathematics, Stanford University

Yinyu Ye YYYE@STANFORD.EDU
Department of Management Science and Engineering, Stanford University
& Shanghai Jiao Tong University & Shanghai Institute for Mathematics and Interdisciplinary Sciences

Madeleine Udell UDELL@STANFORD.EDU
Department of Management Science and Engineering, Stanford University

Editors: Nika Haghtalab and Ankur Moitra

Abstract

We introduce a framework to accelerate the convergence of gradient-based methods with online learning. The framework learns to scale the gradient at each iteration through an online learning algorithm and provably accelerates gradient-based methods asymptotically. In contrast with previous literature, where convergence is established based on worst-case analysis, our framework provides a strong convergence guarantee with respect to the optimal stepsize for the *iteration trajectory*. For smooth strongly convex optimization, our framework provides an $\mathcal{O}(\kappa^\star \log(1/\varepsilon))$ asymptotic complexity result, where $\kappa^\star$ is the condition number achievable by the optimal preconditioner, improving on the previous $\mathcal{O}(\sqrt{n}\kappa^\star \log(1/\varepsilon))$ result. In particular, a variant of our method achieves superlinear convergence on convex quadratics. For smooth convex optimization, we obtain the first convergence guarantee for the widely used hypergradient descent heuristic.

1. Introduction

We consider unconstrained smooth strongly convex optimization problem

$$\underset{x \in \mathbb{R}^n}{\text{minimize}} \quad f(x),$$

where $f(x) : \mathbb{R}^n \to \mathbb{R}$ is L-smooth and μ-strongly convex with $f(x^\star) := \min_x f(x) > -\infty$. It is known that gradient descent with stepsize $1/L$ converges in $\mathcal{O}(\kappa \log(1/\varepsilon))$ iterations, where $\kappa = L/\mu$ is the condition number of the problem. The dependence on the condition number κ is unfortunate since the condition number can be very large and substantially slows down convergence. Two major techniques have been developed in the literature to accelerate gradient descent. One is to improve the dependence on κ through accelerated gradient descent (Necoara et al., 2019; Nesterov, 2013), which achieves $\mathcal{O}(\sqrt{\kappa} \log(1/\varepsilon))$ complexity; the other is through preconditioning: a positive definite matrix stepsize P, known as *preconditioner*, premultiplies the gradient:

$$x^{k+1} = x^k - P\nabla f(x^k). \tag{1}$$

Preconditioning has been a standard tool in convex optimization and numerical linear algebra to improve the convergence of gradient descent (Li, 2017; Maddison et al., 2021; Li et al., 2016;

Frangella et al., 2022, 2023a) or other iterative methods (Saad, 2003), and it is closely related to the well-known adaptive gradient methods (Duchi et al., 2011; Kingma, 2014; Zhang et al., 2024), either for online learning or for a general optimization problem. Some recent results quantify the effect of adaptive gradient methods on problem conditioning (Das et al., 2024). In the context of machine learning, adaptive preconditioner is also relevant to hyperparameter tuning (Hospedales et al., 2021) and learning rate scheduling (Defazio et al., 2024).

Despite the empirical success of adaptive gradient methods in practice, they usually cannot improve the theoretical complexity as a function of the condition number. Recently, Kunstner et al. (2024) showed that hypergradient, the gradient of objective f with respect to the preconditioner, can be used to improve the convergence rate of gradient descent. Kunstner et al. (2024) use a cutting plane subroutine to update the (diagonal) preconditioner and obtain the complexity result $\mathcal{O}(\sqrt{n}\kappa^\star \log(1/\varepsilon))$, where n is the variable dimension and $\kappa^\star$ is the condition number of the optimally preconditioned problem, formally defined as the optimal value of the following semidefinite program:

$$\kappa^\star := \min_{P \in \mathcal{P}_+, \kappa} \kappa \quad \text{subject to} \quad \tfrac{1}{\kappa} I \preceq P^{1/2} \nabla^2 f(x) P^{1/2} \preceq I \quad \text{for all } x, \tag{2}$$

where $\mathcal{P}_+$ is a subset of positive definite matrices. Although the complexity achieved by Kunstner et al. (2024) has a $\sqrt{n}$ term associated with $\kappa^\star$ and requires a nontrivial subroutine to update the preconditioner, the work provides a valuable new direction to theoretically improve the performance of first-order adaptive methods. Whether a simple adaptive first-order method can achieve $\mathcal{O}(\kappa^\star \log(1/\varepsilon))$ complexity or even stronger guarantees remains open. This paper answers this question affirmatively by proposing the online scaled gradient method, a framework that accelerates gradient-based methods through online convex optimization.

Contributions.

- We develop a framework that accelerates gradient-based algorithms through online learning. Unlike previous work, our framework guarantees convergence with respect to the scaling matrix optimized for the iteration trajectory, rather than the worst-case analysis.
- We propose a simple adaptive first-order gradient-based method with asymptotic $\mathcal{O}(\kappa^\star \log(1/\varepsilon))$ complexity, improving on $\mathcal{O}(\sqrt{n}\kappa^\star \log(1/\varepsilon))$ in literature, where $\kappa^\star$ is defined as in (15). In particular, one realization of our framework achieves superlinear convergence on strongly convex quadratics using first-order information.
- For the first time, we show that the hypergradient descent heuristic improves on the convergence of gradient descent (Almeida et al., 1999).

1.1. Related literature

Preconditioned iterative methods. Preconditioning is a well-established technique to enhance the convergence of iterative algorithms in both optimization (Frangella et al., 2023a; O'donoghue et al., 2016; Applegate et al., 2021; Deng et al., 2024) and numerical linear algebra (Saad, 2003; Qu et al., 2024; Gao et al., 2023; Frangella et al., 2023b; Doan and Wolkowicz, 2011). By applying a linear transformation to the optimization variables, preconditioning aims to reduce the heterogeneity of the optimization landscape. Recent research has focused on understanding the properties of optimal preconditioners (Qu et al., 2024; Gao et al., 2023; Jambulapati et al., 2020). While these methods demonstrate empirical success, identifying a good preconditioner can be computationally intensive and often depends on the specific structure of the problem.

Hypergradient descent heuristic. Our method is closely related to the hypergradient descent heuristic (Almeida et al., 1999; Baydin et al., 2018; Chandra et al., 2022), which updates the stepsize (hyperparameters) using the gradient of the optimization objective with respect to it. Despite strong empirical results (Baydin et al., 2018; Chandra et al., 2022), the theoretical understanding of hypergradient descent remains limited. The existing results from Rubio (2017) cannot fully justify the observed improvements. Recently, Kunstner et al. (2024) provides the first theoretical justification for hypergradient descent with a novel multi-dimensional backtracking approach. However, their approach only applies to diagonal preconditioners, requires solving a cutting plane subproblem at every iteration, and incurs a $\sqrt{n}$ dimension dependence in the complexity. No theoretical proof exists that the original hypergradient descent heuristic accelerates gradient-based methods. Our paper provides the first proof that quantifies the acceleration effect of the hypergradient descent heuristic. After this paper comes out, two follow-up works (Gao et al., 2025; Chu et al., 2025) have refined and improved our analyses.

Adaptive preconditioner. Adaptive stepsize is a well-established technique to enhance the convergence of optimization algorithms. The most notable one is `AdaGrad` (Duchi et al., 2011; McMahan and Streeter, 2010), which provides strong theoretical guarantees in the context of online convex optimization. Other methods, such as `Adam` (Kingma, 2014) and `RMSProp` (Hinton et al., 2012), have demonstrated competitive empirical performance, though they generally yield weaker online regret bounds. Our approach also leverages online learning techniques to accelerate gradient-based methods, with a particular focus on improving the dependence on problem conditioning. The idea of applying online learning to learn a preconditioner also appeared in recent work (Jiang et al., 2023; Jiang and Mokhtari, 2024; Jiang et al., 2024) in the context of quasi-Newton methods. However, to our knowledge, the only prior work in a similar spirit is (Zhuang et al., 2019), which adapts the stepsize to noise in stochastic nonconvex optimization using a surrogate loss function.

Learning to optimize and meta-learning. Learning to optimize (Li and Malik, 2016; Chen et al., 2022) and meta-learning (Hospedales et al., 2021; Chen and Hazan, 2024; Finn et al., 2019) literatures also use online learning to improve algorithm performance. These approaches are typically designed to solve a sequence of related optimization problems, providing performance guarantees across multiple tasks. In contrast, our work applies online learning to improve first-order methods over the course of solving a single optimization instance.

2. Background and preliminaries

Notations. Let $\|\cdot\|$ denote vector Euclidean norm or matrix spectral norm, and $\langle \cdot, \cdot \rangle$ denote Euclidean inner product. Letters A and a denote matrices and vectors, respectively. The Frobenius norm is denoted by $\|A\|_F = \sqrt{\sum_{ij} a_{ij}^2}$. Given two vectors $a, b \in \mathbb{R}^n$, let $a \odot b$ denote the element-wise product. The Clarke subdifferential of f at x is defined by $\partial f(x) := \{v \in \mathbb{R}^n : f(y) \geq f(x) + \langle v, y - x \rangle + o(\|x - y\|), y \to x\}$. Let $f'(x) \in \partial f(x)$ denote a subgradient. For symmetric matrices A, B, we say $A \succeq B$ if $A - B \in \mathbb{S}_+^n$ is positive semidefinite. Given a closed convex set C, $\Pi_C[x]$ denotes the orthogonal projection of x onto C. We use $\mathcal{L}_\alpha := \{x : f(x) \leq \alpha\}$ to denote the α-sublevel set of f and $\mathcal{X}^\star$ to denote the optimal set of f. A point x is an ε-optimal solution if $f(x) \leq f(x^\star) + \varepsilon$ and x is an ε-critical point if $\|\nabla f(x)\| \leq \varepsilon$.

Assumptions. We make the following two assumptions throughout the paper.

A1: $f(x)$ is L-smooth. $|f(x) - f(y) - \langle \nabla f(y), x - y\rangle| \leq \frac{L}{2}\|x - y\|^2$.

A2: $f(x)$ is μ-strongly convex ($\mu \geq 0$). $f(x) - f(y) - \langle \nabla f(y), x - y\rangle \geq \frac{\mu}{2}\|x - y\|^2$.

We also assume that f is twice-differentiable for simplicity. However, our algorithm does not necessarily require twice-differentiability to work. In addition, μ-strong convexity can be relaxed to weaker conditions such as convexity with quadratic growth (Necoara et al., 2019).

2.1. Preconditioned and scaled gradient method

In literature, the preconditioner P in (1) is typically positive definite. This paper allows P to be an arbitrary matrix from a closed convex set $\mathcal{P} \subseteq \mathbb{R}^{n\times n}$ and also allows P to vary across iterations:

$$x^{k+1} = x^k - P_k \nabla f(x^k). \tag{3}$$

We call the update (3) *scaled gradient method*: P only serves as a (not necessarily positive definite or symmetric) scaling matrix. Preconditioned gradient descent can be viewed as a special case of the scaled gradient method. We define $\mathcal{P}_+ := \mathcal{P} \cap \mathbb{S}^n_+$ and make the following assumption on $\mathcal{P}$:

A3: Closed convex set $\mathcal{P}$ satisfies $0 \in \mathcal{P}$, $L^{-1}I \in \mathcal{P}$ and $\operatorname{diam}(\mathcal{P}) \leq D$,

where $\operatorname{diam}(\mathcal{P}) := \max_{P_1, P_2 \in \mathcal{P}} \|P_1 - P_2\|_F$ is defined with respect to the Frobenius norm. In practice, D is an algorithm parameter and should be proportional to $1/L$ (e.g., $2025/L$).

2.2. Monotone descent oracle

When P is not positive definite, a scaled gradient update will not necessarily decrease the function value. To guarantee convergence under weak assumptions, the scaled gradient method optionally uses a *monotone descent oracle* $\mathcal{M}$, defined below.

Definition 1. *Given the scaled gradient update $x^+ = x - P\nabla f(x)$, $\mathcal{M}_{\varphi,P} : \mathbb{R}^n \to \mathbb{R}^n$ is called a monotone descent oracle associated with the update and measure φ if its output $\mathcal{M}_{\varphi,P}(x)$ satisfies*

$$\varphi(\mathcal{M}_{\varphi,P}(x)) \leq \min\{\varphi(x), \varphi(x^+)\}.$$

We use $\mathcal{M}(x)$ to denote the oracle when the context is clear. Three typical realizations of $\mathcal{M}$ are:

- *Null step.* Let $\mathcal{M}(x) = x^+$ if $\varphi(x^+) \leq \varphi(x)$; otherwise $\mathcal{M}(x) = x$.
- *Back-tracking line-search.* Take $\mathcal{M}(x) = x + \alpha(x^+ - x)$ such that $\varphi(\mathcal{M}(x)) \leq \varphi(x)$. Additional regularity conditions, such as $\mathcal{P} = \mathcal{P}_+$, are required to ensure that line-search stops in finite steps.
- *Exact line-search.* Take $\mathcal{M}(x) = x + \alpha(x^+ - x)$ and $\alpha = \arg\min_\alpha \varphi(x + \alpha(x^+ - x))$.

3. Online scaled gradient methods

This section introduces our main methodology, which relates the scaled gradient method to online convex optimization with P as the decision variable.

3.1. Scaled gradient method and online-to-offline reduction

We introduce two reductions *in the hyperparameter space* that link the regret of online learning algorithms with offline algorithm convergence (Theorem **1** and Lemma **7**). This section introduces the first reduction, which is universally applicable to linearly convergent algorithms. Let $\varphi(x)$ be a non-negative measure that characterizes the optimality of x. e.g., function value gap $\varphi(x) = f(x) - f(x^\star)$ and gradient norm $\varphi(x) = \|\nabla f(x)\|$ are common measures. The progress of an algorithm at step $K + 1$ with respect to measure φ can be expressed as the telescoping product

$$\varphi(x^{K+1}) = \varphi(x^1) \prod_{k=1}^{K} \tfrac{\varphi(x^{k+1})}{\varphi(x^k)}.$$

Then the arithmetic-geometric mean inequality bounds $\varphi(x^{K+1})$:

Theorem 1 (Online-to-offline reduction)**.** *Given a non-negative function $\varphi(x) : \mathbb{R}^n \to \mathbb{R}_+$ and a sequence of iterations $\{x^k\}$ such that $\varphi(x^k) \neq 0$ for all k,*

$$\varphi(x^{K+1}) \leq \varphi(x^1)\big(\tfrac{1}{K} \sum_{k=1}^{K} \tfrac{\varphi(x^{k+1})}{\varphi(x^k)}\big)^K.$$

The quantity $\frac{1}{K} \sum_{k=1}^{K} \frac{\varphi(x^{k+1})}{\varphi(x^k)}$ on the right-hand side is the averaged contraction factor across all previous iterates: a smaller contraction factor ensures stronger convergence. Suppose the iterates $\{x^k\}_{k\geq 2}$ are generated by the scaled gradient method in (3). Then

$$\tfrac{1}{K} \sum_{k=1}^{K} \tfrac{\varphi(x^{k+1})}{\varphi(x^k)} = \tfrac{1}{K} \sum_{k=1}^{K} \tfrac{\varphi(x^k - P_k \nabla f(x^k))}{\varphi(x^k)}. \tag{4}$$

To maximize the progress in the scaled gradient method, we aim to minimize the quantity in (4) over the choice of scaling matrices P_k with online learning. We will show that online convex optimization can learn a sequence of $\{P_k\}$ that asymptotically accelerates gradient-based methods. Define the *surrogate loss*

$$\ell_x(P) := \tfrac{\varphi(x - P\nabla f(x))}{\varphi(x)}$$

with respect to the measure φ. Note that ℓ_{x^k} only depends on x^1 and all previous scaling matrices $\{P_j\}_{j\leq k-1}$. Online learning generates a sequence $\{P_k\}$ such that the regret is bounded by ρ_K:

$$\sum_{k=1}^{K} \ell_{x^k}(P_k) - \min_{P\in\mathcal{P}} \sum_{k=1}^{K} \ell_{x^k}(P) \leq \rho_K. \tag{5}$$

Existing results in online optimization can guarantee sublinear regret if the losses $\{\ell_{x^k}\}$ are convex and are either Lipschitz continuous or have Lipschitz continuous gradient (Orabona, 2019). In this case, we say the family of surrogate losses $\{\ell_{x^k}\}$ is *online-learnable*. The definition of regret ρ_K and Theorem **1** imply

$$\varphi(x^{K+1}) \leq \varphi(x^1)\big(\tfrac{1}{K} \sum_{k=1}^{K} \ell_{x^k}(P_k)\big)^K \leq \varphi(x^1)\big(\min_{P\in\mathcal{P}} \tfrac{1}{K} \sum_{k=1}^{K} \ell_{x^k}(P) + \tfrac{\rho_K}{K}\big)^K. \tag{6}$$

When the regret ρ_K grows sublinearly in K, the bound in (6) suggests that for large enough K,

$$\varphi(x^{K+1}) \leq \varphi(x^1)\big(\min_{P\in\mathcal{P}} \tfrac{1}{K} \sum_{k=1}^{K} \ell_{x^k}(P) + \tfrac{\rho_K}{K}\big)^K \approx \varphi(x^1)\big(\min_{P\in\mathcal{P}} \tfrac{1}{K} \sum_{k=1}^{K} \ell_{x^k}(P)\big)^K.$$

This result is powerful: it shows that a scaled gradient method, in the long run, can achieve convergence that is competitive with any fixed scaling matrix optimized for the iteration trajectory. To the best of our knowledge, this trajectory-based convergence guarantee is rare in the literature. Moreover, as long as there exists some pre-specified scaling matrix $P^\star$ (or simply stepsize $P^\star = \alpha I$) such that $\ell_x(P^\star) \leq \theta^\star < 1$ for any x, we obtain the global convergence guarantee

$$\varphi(x^{K+1}) \leq \varphi(x^1)(\theta^\star + \tfrac{\rho_K}{K})^K \approx \varphi(x^1)(\theta^\star)^K \tag{7}$$

The algorithm, which updates the scaling matrix P_k on the fly, is called a realization of the *online scaled gradient method* (OSGM).

3.2. Framework of online scaled gradient method

The online scaled gradient method is determined by the components below:

- *Optimality measure*. A measure φ to characterize the convergence of OSGM.
- *Surrogate loss*. A surrogate loss $\ell_x(P)$ that relates φ with an online learning problem in P.
- *Online learning algorithm*. An online learning algorithm $\mathcal{A}$ that guarantees sublinear ρ_K in (5).
- (Optional) *Monotone oracle*. An oracle $\mathcal{M}$ (Definition 1) that guarantees monotonicity.
- (Theory) *Hindsight scaling matrix*. A hindsight scaling matrix $P^\star$ to ensure global convergence.

Algorithm 1: Online scaled gradient method (OSGM)

```
input $x^1, P_1, \varphi, \ell, \mathcal{A}, \mathcal{M}$
for $k$ = 1, 2,... do
    if $\mathcal{M} = \varnothing$ then
        $x^{k+1} = x^k - P_k \nabla f(x^k)$
    else
        $x^{k+1} = \mathcal{M}_{\varphi, P_k}(x^k)$
    end
    $P_{k+1} = \mathcal{A}(\ell_{x^k}, P_k)$
end
output $x^{\text{best}}$ with minimum objective value
```

The tuple $(\varphi, \ell, \mathcal{A}, \mathcal{M})$ determines a realization of the online scaled gradient method (Algorithm 1). In the rest of the paper, we provide several realizations of the framework for different function classes, summarized in Table 1.

$\varphi(x)$	Surrogate ℓ	Strong convexity	$\mathcal{A}$	$\mathcal{M}$ oracle	Complexity	Reference
$f(x) - f(x^\star)$	$r_x(P) = \frac{f(x^+) - f(x^\star)}{f(x) - f(x^\star)}$	Yes	OGD	Optional	$\mathcal{O}(\kappa^\star \log(\frac{1}{\varepsilon}))$	Section 4
$\|\nabla f(x)\|$	$g_x(P) = \frac{\|\nabla f(x^+)\|}{\|\nabla f(x)\|}$	Yes		Required	$\mathcal{O}(\lambda^\star \log(\frac{1}{\varepsilon}))$	Section 5
$f(x) - f(x^\star)$	$h_x(P) = \frac{f(x^+) - f(x)}{\|\nabla f(x)\|^2}$	Yes		Required	$\mathcal{O}(\frac{1}{2\mu\gamma^\star} \log(\frac{1}{\varepsilon}))$	Section 6
		No		Required	$\mathcal{O}(\frac{1}{\gamma^\star \varepsilon})$	

Table 1: Instantiations of OSGM. OGD: online (sub)gradient method. $\kappa^\star, \lambda^\star, \gamma^\star$ and their optimal scaling matrix $P^\star$ will be defined in the next sections.

4. Function value ratio surrogate

The first surrogate, function value *ratio surrogate*, is defined by:

$$r_x(P) := \tfrac{f(x^+)-f(x^\star)}{f(x)-f(x^\star)} = \tfrac{f(x-P\nabla f(x))-f(x^\star)}{f(x)-f(x^\star)}, \tag{8}$$

which is well-defined for $x \notin \mathcal{X}^\star$. The ratio surrogate r_x measures the contraction factor of the function value gap between two consecutive `OSGM` steps. We assume strong convexity ($\mu > 0$) throughout this section. The ratio surrogate r_x assumes the optimal value $f(x^\star)$ is known, and this assumption will be relaxed later in this section. The monotone oracle is optional for r_x. We present the results without a monotone oracle: $\mathcal{M} = \varnothing$.

4.1. Surrogate loss

The function value ratio r_x in (8) can be viewed as a surrogate loss since its average along `OSGM` iterates upperbounds the function value gap. Substituting $\varphi(x) = f(x) - f(x^\star)$ in Theorem **1** and plugging in the definition of r_x, `OSGM` iterates $\{x^k\}$ satisfy the following relation:

Lemma 1 (Online-to-offline reduction)**.** *For all $K \geq 1$, `OSGM` satisfies*

$$f(x^{K+1}) - f(x^\star) \leq (f(x^1) - f(x^\star))\big(\tfrac{1}{K}\textstyle\sum_{k=1}^K r_{x^k}(P_k)\big)^K. \tag{9}$$

The ratio surrogate r_x inherits several important properties from f, which are summarized in Proposition **1**. These properties are crucial for the online learnability of $\{r_{x^k}\}$.

Proposition 1 (Properties of r_x)**.** *Under* **A1** *and* **A2***, for any fixed $x \notin \mathcal{X}^\star$, the surrogate loss $r_x(P)$ defined in (8) is convex, non-negative, and $2L^2$-smooth as a function in P. In addition, the derivative of r_x takes the form*

$$\nabla r_x(P) = -\tfrac{\nabla f(x-P\nabla f(x))\nabla f(x)^\top}{f(x)-f(x^\star)}. \tag{10}$$

4.2. Online learning algorithm

Online gradient descent is known to ensure sublinear regret for a family of smooth, convex, and lower-bounded losses (Orabona, 2019), which is the case for ratio surrogate loss r_x by Proposition **1**. We tailor the classical $L^\star$ regret bound from online convex optimization literature (Orabona, 2019) to our settings in Lemma **2** below.

Lemma 2 (Learnability)**.** *Given* **A1**, **A2***, and the surrogate $\{r_{x^k}\}$, online gradient descent*

$$P_{k+1} = \Pi_{\mathcal{P}}[P_k - \eta \nabla r_{x^k}(P_k)] \tag{11}$$

with stepsize $\eta \leq 1/(4L^2)$ generates a sequence of scaling matrices $\{P_k\}_{k\geq 2}$ such that

$$\textstyle\sum_{k=1}^K r_{x^k}(P_k) - \sum_{k=1}^K r_{x^k}(P) \leq \tfrac{1}{\eta}\|P - P_1\|_F^2 + 4L^2\eta \sum_{k=1}^K r_{x^k}(P) \quad \textit{for any } P \in \mathcal{P}. \tag{12}$$

In particular, if **A3** *is further assumed, the choice of stepsize $\eta = \min\big\{\frac{1}{4L^2}, \frac{D}{2L(1+LD)\sqrt{K}}\big\}$ ensures*

$$\textstyle\sum_{k=1}^K r_{x^k}(P_k) - \min\limits_{P\in\mathcal{P}} \sum_{k=1}^K r_{x^k}(P) \leq \rho_K := \max\big\{4LD(1+LD)\sqrt{K}, 8L^2D^2\big\}. \tag{13}$$

Remark 1. We use online gradient descent with constant stepsize for simplicity. In practice, it is recommended to use varying stepsize $\eta_k = \mathcal{O}(1/\sqrt{k})$ or adaptive online algorithms (with a similar $\mathcal{O}(\sqrt{K})$ regret from Orabona (2019)) to provide anytime convergence guarantees.

Remark 2. The relation (11) suggests additional complexity from a rank-one update with an orthogonal projection. But we can choose $\mathcal{P}$ to have arbitrary sparsity (e.g., diagonal), and it is only necessary to update the nonzero elements. Moreover, the orthogonal projection is easy to compute since we do not require P_k to be positive semidefinite (see Appendix **B** for more details).

4.3. Algorithm design and analysis

We now state a realization of OSGM with the ratio surrogate loss r_x, denoted by OSGM-R. We choose the optimality measure φ, the surrogate loss ℓ, and the online learning algorithm $\mathcal{A}$ to be

$$\varphi(x) := f(x) - f(x^\star), \quad \ell_x(P) := r_x(P), \quad \mathcal{A} := \text{online gradient descent in (11)},$$

and the monotone oracle $\mathcal{M}$ is optional. Algorithm **2** presents OSGM-R without the monotone oracle, the trajectory-based convergence guarantee of which is established in Theorem **2**.

Algorithm 2: Online scaled gradient method with ratio surrogate (OSGM-R)

```
input $x^1, P_1 \in \mathcal{P}$, online gradient stepsize $\eta > 0$
for $k$ = 1, 2,... do
    $x^{k+1} = x^k - P_k \nabla f(x^k)$
    $P_{k+1} = \Pi_{\mathcal{P}}[P_k - \eta \nabla r_{x^k}(P_k)]$
end
output $x^{\text{best}}$ with minimum objective value
```

Theorem 2 (Trajectory-based convergence)**.** *Under* **A1** *to* **A3***, Algorithm* **2** *(OSGM-R) with* $\eta = \min\left\{\frac{1}{4L^2}, \frac{D}{2L(1+LD)\sqrt{K}}\right\}$ *satisfies*

$$f(x^{K+1}) - f(x^\star) \leq (f(x^1) - f(x^\star))(\theta_K^\star + \tfrac{\rho_K}{K})^K, \tag{14}$$

where $\theta_K^\star := \min_{P \in \mathcal{P}} \frac{1}{K} \sum_{k=1}^K r_{x^k}(P)$ *and* $\rho_K = \max\left\{4LD(1 + LD)\sqrt{K}, 8L^2D^2\right\}$.

From (14), when K is large enough, $\frac{\rho_K}{K}$ is negligible, and OSGM-R behaves like an algorithm with linear convergence rate $\theta_K^\star$. Note that $\theta_K^\star$ is based on the optimization trajectory, and the behavior of OSGM-R is competitive with the scaling matrix that minimizes $\theta_K^\star$. To guarantee global convergence, we need $\theta_K^\star < 1$, which follows from the existence of the *optimal preconditioner*.

4.4. Hindsight and global convergence

Let $P_r^\star$ be the scaling matrix that solves the semidefinite optimization problem

$$\kappa^\star := \min_{P \in \mathcal{P}_+} \kappa \quad \text{subject to} \quad \tfrac{1}{\kappa} I \preceq P^{1/2} \nabla^2 f(x) P^{1/2} \preceq I \quad \text{for all } x. \tag{15}$$

which is known as the *universal optimal preconditioner* in the literature (Qu et al., 2024). The optimal value $\kappa^\star$ is called the *optimal condition number* with respect to subset $\mathcal{P}_+ = \mathcal{P} \cap \mathbb{S}_+^n$.

Assumption **A3** assumes $L^{-1}I \in \mathcal{P}$ and hence guarantees $\kappa^\star \leq \kappa$. Gradient descent with preconditioner $P_r^\star$ converges as if the condition number of the optimization problem reduces from $\kappa = L/\mu$ to $\kappa^\star$. Moreover, the descent lemma and strong convexity ensure that

$$f(x - P_r^\star \nabla f(x)) - f(x^\star) \leq (1 - \tfrac{1}{\kappa^\star})(f(x) - f(x^\star)) \quad \text{for all } \; x, \tag{16}$$

which can be equivalently expressed in terms of the ratio surrogate loss r_x in the lemma below:

Lemma 3 (Hindsight)**.** *Instate* **A1** *to* **A3***. Then $r_x(P_r^\star) \leq 1 - \frac{1}{\kappa^\star}$ for all $x \notin \mathcal{X}^\star$.*

Combining Theorem **2** and $\theta_K^\star \leq 1 - \frac{1}{\kappa^\star}$ from Lemma **3**, the asymptotic linear convergence of `OSGM-R` follows immediately.

Corollary 1 (Global convergence)**.** *Under the same assumptions as Theorem* **2***, $\theta_K^\star \leq 1 - \frac{1}{\kappa^\star}$ and the asymptotic complexity of `OSGM-R` to find an ε-optimal point is $\mathcal{O}(\kappa^\star \log(1/\varepsilon))$, where $\kappa^\star$ is the optimal condition number defined in (15). In particular, for any $\varepsilon > 0$, the iteration complexity of `OSGM-R` to reach ε-optimal solution is bounded by*

$$K_\varepsilon := \left\lceil 2\kappa^\star \log\left(\tfrac{f(x^1) - f(x^\star)}{\varepsilon}\right) + 64[\kappa^\star LD(LD+1)]^2 \right\rceil. \tag{17}$$

In fact, a slightly better convergence result can be obtained by evaluating the regret bound (12) at $P = P_r^\star$, which we state as the following theorem.

Theorem 3 (Refined global convergence)**.** *Under* **A1** *and* **A2***, Algorithm* **2** *(`OSGM-R`) with $\eta = \min\{\frac{1}{4L^2}, \frac{\|P_r^\star - P_1\|_F}{2L\sqrt{K}}\}$ satisfies*

$$f(x^{K+1}) - f(x^\star) \leq (f(x^1) - f(x^\star))\big(1 - \tfrac{1}{\kappa^\star} + \max\big\{\tfrac{4L\|P_r^\star - P_1\|_F}{\sqrt{K}}, \tfrac{8L^2\|P_r^\star - P_1\|_F^2}{K}\big\}\big)^K. \tag{18}$$

Remark 3. Theorem **3** gives a $(1 - \frac{1}{\kappa^\star} + \mathcal{O}(\frac{1}{\sqrt{K}}))^K$ rate. The convergence curve (as a function of K) will be continuous and concave: function value first increases, and decreases when $K = \mathcal{O}((\kappa^\star)^2)$.

Remark 4. Note that Theorem **3** does not depend on D. Therefore, `OSGM-R` can be applied even if $\mathcal{P} = \mathbb{R}^{n \times n}$, and there is no need to project P onto $\mathcal{P}$.

The asymptotic linear convergence rate of `OSGM-R` is comparable to that of preconditioned gradient descent using the universal optimal preconditioner $P_r^\star$. This result removes the dimension dependence from the $\mathcal{O}(\sqrt{n}\kappa^\star \log(1/\varepsilon))$ result in Kunstner et al. (2024), which only applies to diagonal preconditioners. As previously remarked, the practical convergence behavior of `OSGM-R` could be even better: the linear convergence rate $\theta_K^\star$ is determined by the best possible choice of $P \in \mathcal{P}$ optimized for the iteration trajectory $\{x^k\}$, while the universal optimal preconditioner $P_r^\star$ is chosen against all possible x in the domain. Since no practical algorithm will visit every $x \in \mathbb{R}^n$, the trajectory-based convergence more precisely characterizes the practical performance of `OSGM`.
For convex quadratics, we have the following equivalent characterization of $P_r^\star$ through r_x:

Proposition 2 (Relation between $P_r^\star$ and r_x)**.** *For $f(x) = \frac{1}{2}\langle x, Ax\rangle$, $A \in \mathbb{S}_{++}^n$, the optimal solutions to the following two problems coincide:*

$$\min_{P \in \mathcal{P}_+} \quad \kappa \quad \text{subject to} \quad \tfrac{1}{\kappa} I \preceq P^{1/2} A P^{1/2} \preceq I; \tag{19}$$

$$\min_{P \in \mathcal{P}_+} \max_{x \notin \mathcal{X}^\star} \quad r_x(P). \tag{20}$$

Since no practical algorithm will visit every $x \in \mathbb{R}^n$, the trajectory-based convergence guarantee more precisely characterizes the practical performance of `OSGM`.

Unknown optimal value. Our method can be extended to the case where $f(x^\star)$ is unknown, but instead, a lower bound $z < f(x^\star)$ is available. In this case, we can define the auxiliary surrogate loss $r_x^z(P) := \frac{f(x-P\nabla f(x))-z}{f(x)-z}$, which is obtained by replacing $f(x^\star)$ in the surrogate loss r_x with lower bound z. Using an additional outer loop to update the lower bound z, the resulting algorithm (Algorithm 6 in Appendix E) can achieve $\mathcal{O}(\kappa^\star \log^2(1/\varepsilon))$ iteration complexity (Theorem 7 in Appendix E).

Asymptotic convergence and dimension dependence. OSGM-R achieves an improved asymptotic convergence rate (in ε) that is independent of problem dimension n. However, our theory only guarantees this faster rate is realized after the algorithm has learned a nontrivial scaling matrix: according to Corollary 1, this warm-up phase can take up to $(\kappa^\star)^2 L^4 D^4$ iterations. Here, LD can depend on the dimension n due to the use of the Frobenius norm in our analysis. We believe this dependence is an artifact of the analysis, but leave further improvement of this asymptotic result to future work. Our experiments (Appendix B) indicate this warm-up phase is very short in practice. A similar warm-up phase appears in our analysis of the algorithm using other surrogate losses. Recently, a nonasymptotic convergence analysis has been developed by Gao et al. (2025) to address the above issues.

Convex quadratics. For strongly convex quadratics $f(x) = \frac{1}{2}\langle x, Ax\rangle - \langle b, x\rangle$, $P_r^\star = A^{-1}$ gives $r_x(A^{-1}) = 0$ for all $x \notin \mathcal{X}^\star$. This implies the following superlinear convergence guarantee.

Theorem 4 (Superlinear convergence on quadratics)**.** *For strongly convex quadratics with* $\nabla^2 f(x) \equiv A \in \mathbb{S}^n_{++}$, *OSGM-R with* $\mathcal{P} = \mathbb{R}^{n\times n}$ *and* $\eta = \frac{1}{4L^2}$ *satisfies* $f(x^{K+1}) - f(x^\star) \leq (f(x^1) - f(x^\star))(\frac{4L^2\|P_1 - A^{-1}\|_F^2}{K})^K$.

5. Gradient norm surrogate

The second surrogate, *gradient norm surrogate*, is defined by the contraction of the gradient norm:

$$g_x(P) := \frac{\|\nabla f(x^+)\|}{\|\nabla f(x)\|} = \frac{\|\nabla f(x - P\nabla f(x))\|}{\|\nabla f(x)\|}. \tag{21}$$

We assume strong convexity ($\mu > 0$) throughout this section. The gradient norm surrogate g_x can be evaluated without $f(x^\star)$. However, it is more challenging to establish the learnability of g_x, which requires the following extra assumption:

A4: $f(x)$ has H-Lipschitz Hessian. $\|\nabla^2 f(x) - \nabla^2 f(y)\| \leq H\|x - y\|$ for all x, y.

This section assumes a nonempty monotone oracle with respect to gradient norm $\mathcal{M}_{\|\nabla f(x)\|,P} \neq \varnothing$.

5.1. Surrogate loss

Substituting $\varphi(x)$ in Theorem 1 with $\varphi(x) = \|\nabla f(x)\|$ and applying the definition of the gradient norm surrogate g_x, we obtain the following lemma.

Lemma 4 (Online-to-offline reduction)**.** *For all* $K \geq 1$, *OSGM with nonempty monotone oracle* $\mathcal{M}_{\|\nabla f(x)\|,P} \neq \varnothing$ *satisfies* $\|\nabla f(x^{K+1})\| \leq \|\nabla f(x^1)\|(\frac{1}{K}\sum_{k=1}^K g_{x^k}(P_k))^K$.

Although the gradient norm surrogate g_x can be nonconvex, Proposition 4 shows that g_x can be approximated by an L-Lipschitz continuous convex function.

Proposition 3 (Properties of g_x)**.** *Under* **A1** *to* **A4***, for any fixed $x \notin \mathcal{X}^\star$, the surrogate loss $g_x(P)$ defined in* (21) *is L-Lipschitz continuous as a function in P and $|g_x(P) - \hat{g}_x(P)| \leq \frac{H}{2}\|\nabla f(x)\|\|P\|^2$, where $\hat{g}_x(P) = \big\|\frac{\nabla f(x)}{\|\nabla f(x)\|} - \nabla^2 f(x) P \frac{\nabla f(x)}{\|\nabla f(x)\|}\big\|$ is convex and L-Lipschitz continuous. In particular,*

$$g_x(P_1) - g_x(P_2) - \langle \hat{g}_x'(P_2), P_1 - P_2 \rangle \geq -HD^2 \|\nabla f(x)\|.$$

In addition, if $x - P\nabla f(x) \notin \mathcal{X}^\star$, the loss $g_x(P)$ is differentiable at P and its derivative takes the form

$$\nabla g_x(P) = -\tfrac{\nabla^2 f(x - P\nabla f(x)) \nabla f(x - P\nabla f(x)) \nabla f(x)^\top}{\|\nabla f(x)\| \cdot \|\nabla f(x - P\nabla f(x))\|}. \tag{22}$$

Proposition **3** bounds the nonconvexity of g_x by $\|\nabla f(x)\|$, the non-stationarity at x. We can still apply online learning algorithms to g_x using these properties.

5.2. Online learning algorithm

Given Lipschitz loss functions whose nonconvexity can be bounded, online subgradient method gives the following regret guarantee.

Lemma 5 (Learnability)**.** *Given* **A1** *to* **A3** *and the surrogate $\{g_{x^k}\}$, online subgradient method*

$$P_{k+1} = \Pi_{\mathcal{P}}[P_k - \eta g_{x^k}'(P_k)] \tag{23}$$

with stepsize $\eta = \frac{2D}{L\sqrt{K}}$ generates a sequence of scaling matrices $\{P_k\}_{k \geq 2}$ such that

$$\textstyle\sum_{k=1}^K g_{x^k}(P_k) - \min\limits_{P \in \mathcal{P}} \sum_{k=1}^K g_{x^k}(P) \leq \rho_K := 2LD\sqrt{K} + \frac{HD^2}{2}\|\nabla f(x^1)\|K. \tag{24}$$

5.3. Algorithm design and analysis

We now state a realization of `OSGM` with the gradient norm surrogate loss g_x, denoted by `OSGM-G`. We choose the optimality measure φ, the surrogate loss ℓ, and the online learning algorithm $\mathcal{A}$ to be

$$\varphi(x) := \|\nabla f(x)\|, \quad \ell_x(P) := g_x(P), \quad \mathcal{A} := \text{online subgradient method in (23)},$$

and the monotone oracle $\mathcal{M}$ is necessary. Algorithm **3** presents the pseudocode for `OSGM-G`, the trajectory-based convergence guarantee of which is established in Theorem **5**.

Algorithm 3: Online scaled gradient method with gradient norm surrogate (`OSGM-G`)

```
input $x^1, P_1 \in \mathcal{P}$, online gradient stepsize $\eta > 0$, nonempty $\mathcal{M}_{\|\nabla f(x)\|, P}$
for $k$ = 1, 2,... do
    $x^{k+1} = \mathcal{M}_{\|\nabla f(x^k)\|, P_k}(x^k)$
    $P_{k+1} = \Pi_{\mathcal{P}}[P_k - \eta g_{x^k}'(P_k)]$
end
output $x^{\text{best}}$ with minimum objective value
```

Theorem 5 (Trajectory-based convergence)**.** *Under* **A1** *to* **A4***, Algorithm* **3** *(`OSGM-G`) with stepsize $\eta = \frac{2D}{L\sqrt{K}}$ satisfies $\|\nabla f(x^{K+1})\| \leq \|\nabla f(x^1)\|(\theta_K^\star + \frac{\rho_K}{K})^K$, where $\theta_K^\star := \min_{P \in \mathcal{P}} \frac{1}{K}\sum_{k=1}^K g_{x^k}(P)$ and $\rho_K = 2LD\sqrt{K} + \frac{HD^2}{2}\|\nabla f(x^1)\|K$.*

Theorem 5 itself does not necessarily yield convergence. The regret ρ_K contains $\frac{HD^2}{2}\|\nabla f(x^1)\|K$ and thus is linear in K. One solution is to start the algorithm at a near-stationary point with sufficiently small $\|\nabla f(x^1)\|$. This strategy leads to a two-stage algorithm, and our main result is based on this strategy for brevity of exposition.

5.4. Hindsight and global convergence

Define $P_g^\star$ to be the gradient norm scaling matrix that solves $\omega^\star := \min_{P \in \mathcal{P}} \max_x \|I - \nabla^2 f(x)P\|$ and define $\lambda^\star := \frac{1}{1-\omega^\star}$. The definition is motivated by

$$\|\nabla f(x - P\nabla f(x))\| = \|\nabla f(x) - \int_0^1 \nabla^2 f(x(t))P\nabla f(x)\mathrm{d}t\| \leq [\int_0^1 \|I - \nabla^2 f(x(t))P\|\mathrm{d}t] \cdot \|\nabla f(x)\|.$$

where $x(t) := x - tP\nabla f(x)$. A contraction is established if $\|I - \nabla^2 f(x)P\| < 1$ for all x.

Lemma 6 (Hindsight)**.** *Instate A1 to A3. Then $g_x(P_g^\star) \leq 1 - \frac{1}{\lambda^\star}$ for all $x \notin \mathcal{X}^\star$ and $\lambda^\star \leq \frac{L}{\mu} = \kappa$.*

Corollary 2 (Global convergence)**.** *Instate the assumptions in Theorem 5. Then $\theta_K^\star \leq 1 - \frac{1}{\lambda^\star}$. Moreover, if $\|\nabla f(x^1)\| \leq \frac{1}{HD^2\lambda^\star}$, the asymptotic complexity of* `OSGM-G` *to find an ε-critical point is $\mathcal{O}(2\lambda^\star \log(1/\varepsilon))$.*

Remark 5. `OSGM-G` can also output an ε-optimal solution since $f(x) - f(x^\star) \leq \frac{1}{2\mu}\|\nabla f(x)\|^2$.

6. Hypergradient surrogate

The last surrogate loss, *hypergradient surrogate*, is defined as follows:

$$h_x(P) := \frac{f(x^+)-f(x)}{\|\nabla f(x)\|^2} = \frac{f(x-P\nabla f(x))-f(x)}{\|\nabla f(x)\|^2}. \tag{25}$$

The name hypergradient comes from Baydin et al. (2018), by which the hypergradient descent heuristic improves the convergence of gradient-based methods. Unlike the ratio surrogate r_x or the gradient norm surrogate g_x, the hypergradient surrogate h_x itself is not directly derived from $\varphi(x)$ using a telescopic product, but instead motivated by the descent lemma:

$$f\big(x - \tfrac{1}{L}\nabla f(x)\big) - f(x) \leq -\tfrac{1}{2L}\|\nabla f(x)\|^2.$$

Dividing both sides of the inequality by $\|\nabla f(x)\|^2$ for $x \notin \mathcal{X}^\star$ gives h_x. The descent lemma does not depend on the strong convexity coefficient μ, so the hypergradient surrogate h_x applies to general convex (non-strongly convex) optimization problems. Throughout this section, we assume a nonempty monotone oracle with respect to function value gap $\mathcal{M}_{f(x)-f(x^\star),P} \neq \varnothing$.

6.1. Surrogate loss

Analysis of the hypergradient surrogate h_x requires a different online-to-offline reduction, established in Lemma 7.

Lemma 7 (Online-to-offline reduction)**.** *Under A1, A2, for all $K \geq 1$,* `OSGM` *with nonempty monotone oracle $\mathcal{M}_{f(x)-f(x^\star),P} \neq \varnothing$ satisfies:*
If $\mu > 0$, then

$$f(x^{K+1}) - f(x^\star) \leq (f(x^1) - f(x^\star))(1 - 2\mu \max\{\tfrac{1}{K}\textstyle\sum_{k=1}^K -h_{x^k}(P_k), 0\})^K. \tag{26}$$

If $\mu \geq 0$, then

$$f(x^{K+1}) - f(x^\star) \leq \tfrac{\Delta^2}{K \max\{\frac{1}{K}\sum_{k=1}^K -h_{x^k}(P_k), 0\}}, \quad \min_{1 \leq k \leq K} \|\nabla f(x^k)\|^2 \leq \tfrac{f(x^1) - f(x^\star)}{K \max\{\frac{1}{K}\sum_{k=1}^K -h_{x^k}(P_k), 0\}} \tag{27}$$

where $\Delta = \max_{x \in \mathcal{L}_{f(x^1)}} \min_{x^\star \in \mathcal{X}^\star} \|x - x^\star\|$.

Remark 6. The $\max\{\cdot, 0\}$ terms arise from the monotone oracle. Here, we slightly abuse the notation: if the denominator in Lemma 7 is 0, the bound simplifies to a trivial infinity bound.

Now we establish the properties of the hypergradient surrogate h_x.

Proposition 4 (Properties of h_x)**.** *Under* **A1** *to* **A3***, for any fixed $x \notin \mathcal{X}^\star$, the surrogate loss $h_x(P)$ defined in* (25) *is convex and $(LD + 1)$-Lipschitz continuous as a function in P. In addition, the derivative of h_x takes the form*

$$\nabla h_x(P) = -\tfrac{\nabla f(x - P\nabla f(x)) \nabla f(x)^\top}{\|\nabla f(x)\|^2}. \tag{28}$$

6.2. Online learning algorithm

Online gradient descent with hypergradient surrogate gives the following regret guarantee.

Lemma 8 (Learnability)**.** *Given* **A1** *to* **A3** *and the surrogate $\{h_{x^k}\}$, online gradient descent*

$$P_{k+1} = \Pi_{\mathcal{P}}[P_k - \eta \nabla h_{x^k}(P_k)] \tag{29}$$

with stepsize $\eta = \frac{2D}{(LD+1)\sqrt{K}}$ generates a sequence of scaling matrices $\{P_k\}_{k \geq 2}$ such that

$$\textstyle\sum_{k=1}^K h_{x^k}(P_k) - \min_{P \in \mathcal{P}} \sum_{k=1}^K h_{x^k}(P) \leq \rho_K := 2D(LD + 1)\sqrt{K}. \tag{30}$$

6.3. Algorithm design and analysis

We now state a realization of `OSGM` with the hypergradient surrogate loss h_x, denoted by `OSGM-H`. We choose the optimality measure φ, the surrogate loss ℓ, and the online learning algorithm $\mathcal{A}$ to be

$$\varphi(x) := f(x) - f(x^\star) \text{ or } \|\nabla f(x)\|, \quad \ell_x(P) := h_x(P), \quad \mathcal{A} := \text{online gradient descent in (29)},$$

and the monotone oracle $\mathcal{M}$ is necessary. Algorithm 4 presents the pseudocode for `OSGM-H`, the trajectory-based convergence guarantee of which is established in Theorem 6.

Algorithm 4: Online scaled gradient method with hypergradient surrogate (`OSGM-H`)

```
input $x^1, P_1 \in \mathcal{P}$, online gradient stepsize $\eta > 0$, nonempty oracle $\mathcal{M}_{f(x)-f(x^\star),P}$
for $k$ = 1, 2,... do
    $x^{k+1} = \mathcal{M}_{f(x)-f(x^\star),P_k}(x^k)$
    $P_{k+1} = \Pi_{\mathcal{P}}[P_k - \eta \nabla h_{x^k}(P_k)]$
end
output $x^{\text{best}}$ with minimum objective value
```

Theorem 6 (Trajectory-based convergence)**.** *Under* **A1** *to* **A3***, Algorithm 4 (*`OSGM-H`*) with stepsize $\eta = \frac{2D}{(LD+1)\sqrt{K}}$ satisfies*

- *If* $\mu > 0$*, then*

$$f(x^{K+1}) - f(x^\star) \leq (f(x^1) - f(x^\star))(1 - 2\mu \max\{-\theta_K^\star - \tfrac{\rho_K}{K}, 0\})^K.$$

- *If* $\mu \geq 0$*, then*

$$\min_{1 \leq k \leq K} \|\nabla f(x^k)\|^2 \leq \tfrac{f(x^1) - f(x^\star)}{K \max\{-\theta_K^\star - \frac{\rho_K}{K}, 0\}} \quad \textit{and} \quad f(x^{K+1}) - f(x^\star) \leq \tfrac{\Delta^2}{K \max\{-\theta_K^\star - \frac{\rho_K}{K}, 0\}},$$

where $\theta_K^\star := \min_{P \in \mathcal{P}} \frac{1}{K} \sum_{k=1}^K h_{x^k}(P)$*,* Δ *is defined in Lemma 7 and* $\rho_K = 2D(LD+1)\sqrt{K}$*.*

6.4. Hindsight and global convergence

Define $P_h^\star$ to be the hypergradient scaling matrix that solves

$$\gamma^\star := \max_{P \in \mathcal{P}} \min_{x \in \mathcal{L}_{f(x^1)} \setminus \mathcal{X}^\star} -h_x(P) = \tfrac{f(x) - f(x - P\nabla f(x))}{\|\nabla f(x)\|^2}.$$

Intuitively, $\gamma^\star$ maximizes the function value progress with respect to gradient norm and can be viewed as the inverse of local smoothness constant. The descent lemma gives a lower bound on $\gamma^\star$.

Lemma 9 (Hindsight)**.** *Instate A1 to A3. Then* $-h_x(P_h^\star) \geq \gamma^\star \geq \frac{1}{2L}$ *for all* $x \notin \mathcal{X}^\star$*.*

Corollary 3 (Global convergence)**.** *Instate the assumptions in Theorem 6. Then* $\theta_K^\star \leq -\gamma^\star$ *and the asymptotic complexity of* `OSGM-H` *to find an* ε*-optimal point is* ***1)*** *if* $\mu > 0$*, then* $\mathcal{O}(\frac{1}{2\mu\gamma^\star} \log(1/\varepsilon))$ ***2)*** *if* $\mu = 0$*, then* $\mathcal{O}(\frac{1}{\gamma^\star \varepsilon})$*.*

Remark 7. Given $-\theta_K^\star \geq \frac{1}{2L}$, the complexity of `OSGM-H` is no worse than vanilla gradient descent and can provide acceleration if $-\theta_K^\star > \frac{1}{2L}$. Our results for the first time show that the hypergradient heuristic, when combined with a monotone oracle $\mathcal{M}$, provably accelerates gradient descent. We leave it to future work to establish further convergence properties of hypergradient descent. We refer the interested readers to a follow-up work (Chu et al., 2025) for more details on the theoretical analysis of the hypergradient surrogate.

7. Conclusions and future directions

In this paper, we propose `OSGM`, a general framework that allows online convex optimization algorithms to provably accelerate gradient-based algorithms. Our framework achieves a strong trajectory-based convergence guarantee and explains the success of the popular hypergradient descent heuristic. Future directions include extending the results to accelerated gradient descent, stochastic gradient descent, nonconvex, nonsmooth, and constrained optimization, and to other iterative algorithms where a scaled update affects the algorithm performance.

Acknowledgments

We appreciate the constructive feedback from Professor Qi Deng from Shanghai Jiaotong University, Zikai Xiong from MIT, and the anonymous reviewers. MU, YC, and WG gratefully acknowledge support from the National Science Foundation (NSF) Award IIS-2233762, the Office of Naval Research (ONR) Awards N000142212825, N000142412306, and N000142312203, the Alfred P. Sloan Foundation, and from IBM Research as a founding member of Stanford Institute for Human-centered Artificial Intelligence (HAI).

References

Luís B Almeida, Thibault Langlois, José D Amaral, and Alexander Plakhov. Parameter adaptation in stochastic optimization. In *On-line learning in neural networks*, pages 111–134. 1999.

David Applegate, Mateo Diaz, Oliver Hinder, Haihao Lu, Miles Lubin, Brendan O'Donoghue, and Warren Schudy. Practical large-scale linear programming using primal-dual hybrid gradient. *Advances in Neural Information Processing Systems*, 34:20243–20257, 2021.

Atilim Gunes Baydin, Robert Cornish, David Martinez Rubio, Mark Schmidt, and Frank Wood. Online learning rate adaptation with hypergradient descent. In *International Conference on Learning Representations*, 2018. URL https://openreview.net/forum?id=BkrsAzWAb.

Kartik Chandra, Audrey Xie, Jonathan Ragan-Kelley, and Erik Meijer. Gradient descent: The ultimate optimizer. *Advances in Neural Information Processing Systems*, 35:8214–8225, 2022.

Chih-Chung Chang and Chih-Jen Lin. Libsvm: a library for support vector machines. *ACM transactions on intelligent systems and technology (TIST)*, 2(3):1–27, 2011.

Tianlong Chen, Xiaohan Chen, Wuyang Chen, Howard Heaton, Jialin Liu, Zhangyang Wang, and Wotao Yin. Learning to optimize: A primer and a benchmark. *Journal of Machine Learning Research*, 23(189):1–59, 2022.

Xinyi Chen and Elad Hazan. Online control for meta-optimization. *Advances in Neural Information Processing Systems*, 36, 2024.

Ya-Chi Chu, Wenzhi Gao, Yinyu Ye, and Madeleine Udell. Provable and practical online learning rate adaptation with hypergradient descent. *arXiv preprint arXiv:2502.11229*, 2025.

Rudrajit Das, Naman Agarwal, Sujay Sanghavi, and Inderjit S Dhillon. Towards quantifying the preconditioning effect of adam. *arXiv preprint arXiv:2402.07114*, 2024.

Alexandre d'Aspremont, Damien Scieur, Adrien Taylor, et al. Acceleration methods. *Foundations and Trends® in Optimization*, 5(1-2):1–245, 2021.

Aaron Defazio, Harsh Mehta, Konstantin Mishchenko, Ahmed Khaled, Ashok Cutkosky, et al. The road less scheduled. *arXiv preprint arXiv:2405.15682*, 2024.

Qi Deng, Qing Feng, Wenzhi Gao, Dongdong Ge, Bo Jiang, Yuntian Jiang, Jingsong Liu, Tianhao Liu, Chenyu Xue, Yinyu Ye, et al. An enhanced alternating direction method of multipliers-based interior point method for linear and conic optimization. *INFORMS Journal on Computing*, 2024.

X Doan and Henry Wolkowicz. Numerical computations and the ω-condition number. Technical report, Citeseer, 2011.

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

Chelsea Finn, Aravind Rajeswaran, Sham Kakade, and Sergey Levine. Online meta-learning. In *International conference on machine learning*, pages 1920–1930. PMLR, 2019.

Zachary Frangella, Pratik Rathore, Shipu Zhao, and Madeleine Udell. Sketchysgd: Reliable stochastic optimization via robust curvature estimates. *arXiv preprint arXiv:2211.08597*, 2022.

Zachary Frangella, Pratik Rathore, Shipu Zhao, and Madeleine Udell. Promise: Preconditioned stochastic optimization methods by incorporating scalable curvature estimates. *arXiv preprint arXiv:2309.02014*, 2023a.

Zachary Frangella, Joel A Tropp, and Madeleine Udell. Randomized nyström preconditioning. *SIAM Journal on Matrix Analysis and Applications*, 44(2):718–752, 2023b.

Wenzhi Gao, Zhaonan Qu, Madeleine Udell, and Yinyu Ye. Scalable approximate optimal diagonal preconditioning. *arXiv preprint arXiv:2312.15594*, 2023.

Wenzhi Gao, Ya-Chi Chu, Yinyu Ye, and Madeleine Udell. Gradient methods with online scaling part i. theoretical foundations. *arXiv preprint arXiv:2505.23081*, 2025.

Elad Hazan and Sham Kakade. Revisiting the polyak step size. *arXiv preprint arXiv:1905.00313*, 2019.

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.

Timothy Hospedales, Antreas Antoniou, Paul Micaelli, and Amos Storkey. Meta-learning in neural networks: A survey. *IEEE transactions on pattern analysis and machine intelligence*, 44(9): 5149–5169, 2021.

Arun Jambulapati, Jerry Li, Christopher Musco, Aaron Sidford, and Kevin Tian. Fast and near-optimal diagonal preconditioning. *arXiv preprint arXiv:2008.01722*, 2020.

Ruichen Jiang and Aryan Mokhtari. Online learning guided quasi-newton methods with global non-asymptotic convergence. *arXiv preprint arXiv:2410.02626*, 2024.

Ruichen Jiang, Qiujiang Jin, and Aryan Mokhtari. Online learning guided curvature approximation: A quasi-newton method with global non-asymptotic superlinear convergence. In *The Thirty Sixth Annual Conference on Learning Theory*, pages 1962–1992. PMLR, 2023.

Ruichen Jiang, Aryan Mokhtari, and Francisco Patitucci. Improved complexity for smooth nonconvex optimization: A two-level online learning approach with quasi-newton methods. *arXiv preprint arXiv:2412.02175*, 2024.

Diederik P Kingma. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Frederik Kunstner, Victor Sanches Portella, Mark Schmidt, and Nicholas Harvey. Searching for optimal per-coordinate step-sizes with multidimensional backtracking. *Advances in Neural Information Processing Systems*, 36, 2024.

Yuh-Jye Lee and Olvi L Mangasarian. Ssvm: A smooth support vector machine for classification. *Computational optimization and Applications*, 20:5–22, 2001.

Chunyuan Li, Changyou Chen, David Carlson, and Lawrence Carin. Preconditioned stochastic gradient langevin dynamics for deep neural networks. In *Proceedings of the AAAI conference on artificial intelligence*, volume 30, 2016.

Ke Li and Jitendra Malik. Learning to optimize. *arXiv preprint arXiv:1606.01885*, 2016.

Xi-Lin Li. Preconditioned stochastic gradient descent. *IEEE transactions on neural networks and learning systems*, 29(5):1454–1466, 2017.

Chris J Maddison, Daniel Paulin, Yee Whye Teh, and Arnaud Doucet. Dual space preconditioning for gradient descent. *SIAM Journal on Optimization*, 31(1):991–1016, 2021.

H Brendan McMahan and Matthew Streeter. Adaptive bound optimization for online convex optimization. *arXiv preprint arXiv:1002.4908*, 2010.

Ion Necoara, Yu Nesterov, and Francois Glineur. Linear convergence of first order methods for non-strongly convex optimization. *Mathematical Programming*, 175:69–107, 2019.

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

Brendan O'donoghue, Eric Chu, Neal Parikh, and Stephen Boyd. Conic optimization via operator splitting and homogeneous self-dual embedding. *Journal of Optimization Theory and Applications*, 169:1042–1068, 2016.

Francesco Orabona. A modern introduction to online learning. *arXiv preprint arXiv:1912.13213*, 2019.

Zhaonan Qu, Wenzhi Gao, Oliver Hinder, Yinyu Ye, and Zhengyuan Zhou. Optimal diagonal preconditioning. *Operations Research*, 2024.

David Martinez Rubio. Convergence analysis of an adaptive method of gradient descent. *University of Oxford, Oxford, M. Sc. thesis*, 2017.

Yousef Saad. *Iterative methods for sparse linear systems*. SIAM, 2003.

Yushun Zhang, Congliang Chen, Ziniu Li, Tian Ding, Chenwei Wu, Yinyu Ye, Zhi-Quan Luo, and Ruoyu Sun. Adam-mini: Use fewer learning rates to gain more. *arXiv preprint arXiv:2406.16793*, 2024.

Zhenxun Zhuang, Ashok Cutkosky, and Francesco Orabona. Surrogate losses for online learning of stepsizes in stochastic non-convex optimization. In *International Conference on Machine Learning*, pages 7664–7672. PMLR, 2019.

Appendix A. Practical considerations

This section considers practical aspects of implementing OSGM.

A.1. Gradient evaluations for r_x and h_x using null step oracle

A first look at Algorithm 2 and Algorithm 4 suggests an additional gradient evaluation at every iteration. Both ∇r_x and ∇h_x need to evaluate two gradients in every iteration. However, with $x^{k+1/2} := x^k - P_k \nabla f(x^k)$ and null step (Section 2.2) as the monotone oracle, the gradient $\nabla r_{x^k}(P)$ in Algorithm 2 can be expressed as

$$\nabla r_{x^k}(P) = -\tfrac{\nabla f(x^{k+1/2}) \nabla f(x^k)^\top}{f(x^k) - f(x^\star)}. \tag{31}$$

If $x^{k+1} = x^{k+1/2}$, then $\nabla f(x^{k+1/2})$ can be reused in the next iteration; if $x^{k+1} = x^k$, then $\nabla f(x^k)$ can be reused. Therefore, null step oracle ensures that the total number of gradient evaluations in OSGM-R is the same as that of gradient descent but simply requires an additional cache to store $\nabla f(x^{k+1})$. Regarding ∇g_x, its implementation needs a Hessian-gradient product, which can be efficiently computed in practice.

A.2. Efficient scaling matrix updates

The subset $\mathcal{P}$ can be chosen to have a simple structure, such as diagonal matrices or sparse matrices with some prespecified sparsity pattern. Then the scaling matrix P_{k+1} can be efficiently updated in the cost of $\mathcal{O}(\mathtt{supp}(\mathcal{P}))$ since it suffices to compute the non-zero entries of the sparsity pattern in $\mathcal{P}$. For example, if $\mathcal{P}$ is the set of diagonal matrices, then (31) simplifies to $\nabla r_{x^k}(P) = -\frac{\nabla f(x^k - P_k \nabla f(x^k)) \odot \nabla f(x^k)}{f(x^k) - f(x^\star)}$, where $\odot$ denotes the element-wise product. A simpler structure in $\mathcal{P}$ provides more efficient scaling matrix updates. However, more freedom in $\mathcal{P}$ may provide smaller $\theta_K^\star$, enhancing the convergence of OSGM.

A.3. Choice of candidate set of scaling matrices $\mathcal{P}$

We propose two heuristics for choosing a subset $\mathcal{P}$ of sparse matrices. Sparsity refers to either entries sparsity or spectrum sparsity. We assume that some Hessian matrix $\nabla^2 f(x) = A \succ 0$ is known.

- *Nonzero sparsity pattern.*
  A preconditioner can be viewed as a cutting plane in the difference of extremal spectrum (Gao et al., 2023). Let $v_{\min}$ and $v_{\max}$ be two extremal eigenvectors of A. Then $|v_{\max} v_{\max}^\top - v_{\min} v_{\min}^\top|$, an $n \times n$ grid with nonnegative entries, serves as a score function for the most critical sparsity pattern. The large-magnitude entries in $|v_{\max} v_{\max}^\top - v_{\min} v_{\min}^\top|$ strongly affect the conditioning of the system.

- *Spectral sparsity (low rank).*
  It is common to consider diagonal plus low-rank preconditioners, and randomized preconditioners have proved to be very efficient (Frangella et al., 2023b). Given a low-rank matrix M, we can parameterize $\mathcal{P} = \{\mathrm{diag}(d) + \alpha M : (d, \alpha) \in \mathbb{R}^{n+1}\}$ to be the linear combination between diagonal matrices and M.

A.4. Choice of online learning algorithm $\mathcal{A}$

Our convergence analyses show that a good online learning algorithm $\mathcal{A}$ benefits convergence of `OSGM`. For simplicity, the simplest possible online learning algorithms are adopted in the theoretical analysis. However, `OSGM` is compatible with more advanced online learning algorithms such as `AdaGrad`. Using advanced online algorithms often greatly improves the robustness and practical performance of `OSGM`. In particular, our results on the hypergradient surrogate loss provide new insights into improving the hypergradient descent heuristics.

Appendix B. Numerical experiments

In this section, we conduct experiments to show the performance of online scaled gradient methods. We test on standard strongly convex optimization problems, including least squares and support vector machine.

B.1. Experiment setup

Synthetic data. For the least squares problem $f(x) = \frac{1}{2}\|Ax - b\|^2$, $A \in \mathbb{R}^{n \times n} = CDC^\top + \sigma I$ with C is element-wise generated from $0.01 \times \mathcal{N}(0, 1)$ and an identity matrix I is added to it; D is a diagonal matrix with $\mathcal{U}[0, 1]^n$ diagonals; $b \in \mathbb{R}^m$ is generated from $\mathcal{U}[0, 1]^n$.

Real data. We use datasets from `LIBSVM` (Chang and Lin, 2011) for support vector machine (SVM) problem $f(x) = \frac{1}{m}\sum_{i=1}^m f_i(x) + \frac{\lambda}{2}\|x\|^2$, where f_i is the squared hinge loss (Lee and Mangasarian, 2001). We set $\lambda = 5/n$.

Benchmark algorithms. Eight algorithms are compared:

- (`GD`) Gradient descent with $1/L$ stepsize.
- (`OptDiagGD`) Gradient descent with the universal optimal diagonal preconditioner (Qu et al., 2024; Gao et al., 2023).
- (`OSGM-R`) Online scaled gradient method with ratio surrogate.
- (`OSGM-G`) Online scaled gradient method with gradient norm surrogate.
- (`OSGM-H`) Online scaled gradient method with hypergradient surrogate.
- (`AdaGrad`) Adaptive (sub)gradient method (Duchi et al., 2011).
- (`AGD`) Accelerated gradient descent for general convex problems (d'Aspremont et al., 2021).
- (`SAGD`) Accelerated gradient descent for strongly convex problems (d'Aspremont et al., 2021).

`OptDiagGD` and `OSGM-G` are only tested on problems with fixed Hessian.

Algorithm configurations. We configure the algorithms as follows.

1) *Dataset generation*. For synthetic data, we pick $n = 100$ and $\sigma \in \{10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}\}$.

2) *Initial point*. Initial points x^1 for all the algorithms are generated from standard normal $\mathcal{N}(0, I_n)$ and scaled to have unit length. Initial scaling matrix $P_0 = 0$.

3) *Maximum iteration*. The maximum iteration is set to $K = 10000$.

4) *Stopping criterion*. Algorithm stops when $\|\nabla f(x^k)\| \leq 10^{-10}$.

5) *Stepsize configuration.* (AdaGrad) uses the optimal stepsize among $\{10^{-3}, 10^{-2}, 10^{-1}, 1, 10\}$.

6) *Monotone oracle.* All OSGM methods use null step (Section 2.2) as the monotone oracle.

7) *Online learning algorithm.* All OSGM methods use $\mathcal{A} =$ (AdaGrad) with the optimal stepsize among $\{10^{-3}, 10^{-2}, 10^{-1}, 1, 10\}$.

8) *Choice of candidate scaling matrix $\mathcal{P}$.* We choose $\mathcal{P}$ as the set of diagonal matrices in the experiments.

9) *Knowledge of optimal value.* When the exact optimal value is unknown, OSGM-R uses the auxiliary surrogate r_x^z. But we allow z to be an arbitrary guess of $f(x^\star)$, and if $z > f(x^k)$, we heuristically adjust $z \leftarrow f(x^k) - \min\{5(z - f(x^k)), 1\}$ to update the lower bound. This strategy is not theoretically justified but performs well in practice.

B.2. Toy example: near diagonal convex quadratic

This section verifies the convergence behavior of OSGM on a toy least squares problem with near diagonal Hessian. The problem has $\kappa \approx 68$ and $\kappa^\star \approx 4.7 < \sqrt{\kappa}$. Theory predicts that OSGM should outperform SAGD asymptotically. Figure 1 (left) illustrates the performance of the eight tested algorithms, with OSGM-R and OSGM-H showing the most competitive performance. In particular, the linear convergence rates (slope) of three OSGM algorithms are better than that of SAGD, which aligns with our theory. Moreover, OSGM-R and OSGM-H both converge faster than gradient descent using the universal diagonal preconditioner (OptDiagGD). This is also consistent with our theory and suggests that we can still gain from being adaptive, even on a convex quadratic with fixed curvature. Notably, AdaGrad also achieves competitive performance on this problem.

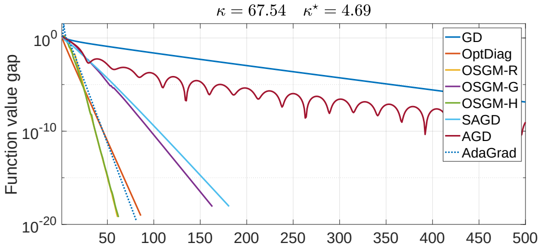

Figure 1: Comparison of benchmark algorithms on toy quadratic problem.

B.3. More comparison between the algorithms

This section compares different algorithms on the aforementioned datasets. Figure 2 shows the results on synthetic least squares problems with different condition numbers.

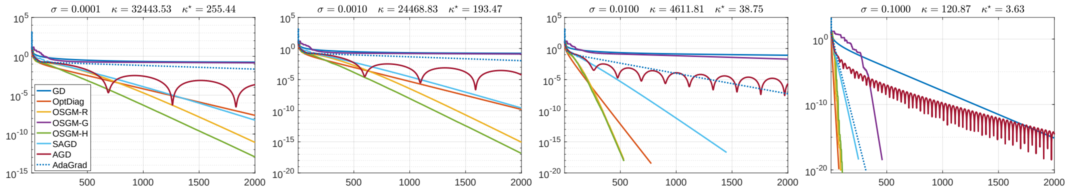

Figure 2: Function value gap on least squares problem with $\sigma \in \{10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}\}$

Figure 2 suggests that when $\kappa^\star \ll \sqrt{\kappa}$, OSGM tends to outperform accelerated methods. On the other hand, if $\kappa^\star > \sqrt{\kappa}$, OSGM is often less competitive compared to SAGD on quadratics.

Figure 3 shows the results on the SVM problems from LIBSVM, and we observe similar competitive performance of OSGM-R and OSGM-H.

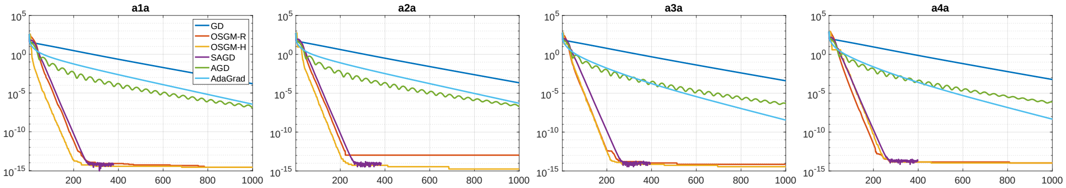

Figure 3: Function value gap on the support vector machine problems

Appendix C. Proof of results in Section 3

C.1. Proof of Theorem 1

Since the measure φ is non-negative. Applying arithmetic-geometric mean inequality

$$\big(\textstyle\prod_{k=1}^K a_k\big)^{1/K} \leq \frac{1}{K}\sum_{k=1}^K a_k$$

completes the proof.

Appendix D. Proof of results in Section 4

D.1. Auxiliary results

Lemma 10 ((Orabona, 2019)). *Let $r(P)$ be a τ-smooth function with $\inf_{P\in\mathbb{R}^{n\times n}} r(P) \geq 0$. Then $r(P) \geq \frac{1}{2\tau}\|\nabla r(P)\|^2$ for all $P \in \mathcal{P}$.*

Lemma 11. *Given a family of non-negative, convex, and τ-smooth losses $\{r_k\}$, online gradient descent*

$$P_{k+1} = \Pi_{\mathcal{P}}[P_k - \eta\nabla r_k(P_k)] \tag{32}$$

with stepsize $\eta \leq 1/(2\tau)$ generates a sequence of scaling matrices $\{P_k\}_{k\geq 2}$ such that

$$\textstyle\sum_{k=1}^K r_k(P_k) - \sum_{k=1}^K r_k(P) \leq \frac{1}{\eta}\|P - P_1\|_F^2 + 2\tau\eta\sum_{k=1}^K r_k(P) \quad \textit{for any } P \in \mathcal{P}. \tag{33}$$

Proof. The proof follows the standard proof of the $L^\star$ regret bound (Orabona, 2019) in online convex optimization but is tailored to our settings. For any $P \in \mathcal{P}$, we have

$$\begin{aligned}\|P_{k+1} - P\|_F^2 &= \|\Pi_{\mathcal{P}}[P_k - \eta\nabla r_k(P_k)] - P\|_F^2 \\ &\leq \|P_k - P - \eta\nabla r_k(P_k)\|_F^2 \qquad (34)\\ &= \|P_k - P\|_F^2 - 2\eta\langle\nabla r_k(P_k), P_k - P\rangle + \eta^2\|\nabla r_k(P_k)\|_F^2, \qquad (35)\end{aligned}$$

where (34) uses non-expansiveness of projection. With convexity $r_k(P) - r_k(P_k) \geq \langle\nabla r_k(P_k), P - P_k\rangle$,

$$\|P_{k+1} - P\|_F^2 \leq \|P_k - P\|_F^2 + 2\eta(r_k(P) - r_k(P_k)) + \eta^2\|\nabla r_k(P_k)\|_F^2.$$

Re-arrangement yields $r_k(P_k) - r_k(P) \leq \frac{\eta}{2}\|\nabla r_k(P_k)\|^2 + \frac{1}{2\eta}[\|P_k - P\|_F^2 - \|P_{k+1} - P\|_F^2]$. Telescoping over k and dropping the term $-\frac{1}{2\eta}\|P_{K+1} - P\|_F^2$ to obtain

$$\textstyle\sum_{k=1}^K r_k(P_k) - \sum_{k=1}^K r_k(P) \leq \frac{1}{2\eta}\|P_1 - P\|_F^2 + \frac{\eta}{2}\sum_{k=1}^K \|\nabla r_k(P_k)\|_F^2. \tag{36}$$

Using Lemma **10**, we have $\|\nabla r_k(P_k)\|_F^2 \leq 2\tau r_k(P_k)$. Plugging this bound into (36) gives

$$\textstyle\sum_{k=1}^K r_k(P_k) - \sum_{k=1}^K r_k(P) \leq \frac{1}{2\eta}\|P_1 - P\|_F^2 + \tau\eta\sum_{k=1}^K r_k(P_k).$$

Re-arrangement gives

$$(1-\tau\eta)\big[\textstyle\sum_{k=1}^K r_k(P_k) - \sum_{k=1}^K r_k(P)\big] \leq \frac{1}{2\eta}\|P_1 - P\|_F^2 + \tau\eta\sum_{k=1}^K r_k(P). \tag{37}$$

For $\eta \leq \frac{1}{2\tau}$, we may divide both sides of (37) by $1-\tau\eta$ and plug in the bound $\frac{1}{1-\tau\eta} \leq 2$ to obtain

$$\textstyle\sum_{k=1}^K r_k(P_k) - \sum_{k=1}^K r_k(P) \leq \frac{1}{\eta}\|P - P_1\|_F^2 + 2\tau\eta\sum_{k=1}^K r_k(P), \tag{38}$$

and this completes the proof.

□

D.2. Proof of Lemma 1

Since $f(x) - f(x^\star) \geq 0$, applying Theorem **1** with $\varphi(x) = f(x) - f(x^\star)$ completes the proof.

D.3. Proof of Proposition 1

Denote $u_x(P) := f(x - P\nabla f(x))$. As a function of P, $u_x(P) = f(x - P\nabla f(x))$ is a composition of convex function $f(x)$ and affine transformation $x - P\nabla f(x)$. Hence u_x is a convex function. The chain rule gives

$$\nabla u_x(P) = \nabla f(x - P\nabla f(x))\nabla f(x)^\top.$$

For any $P_1, P_2 \in \mathcal{P}$, we can successively deduce that

$$\begin{aligned}
\|\nabla u_x(P_1) - \nabla u_x(P_2)\|_F &= \|\nabla f(x - P_1\nabla f(x))\nabla f(x)^\top - \nabla f(x - P_2\nabla f(x))\nabla f(x)^\top\|_F \\
&= \|[\nabla f(x - P_1\nabla f(x)) - \nabla f(x - P_2\nabla f(x))]\nabla f(x)^\top\|_F \\
&\leq \|\nabla f(x - P_1\nabla f(x)) - \nabla f(x - P_2\nabla f(x))\| \cdot \|\nabla f(x)\| \qquad (39)\\
&\leq L\|(P_1 - P_2)\nabla f(x)\| \cdot \|\nabla f(x)\| \qquad (40)\\
&\leq L\|\nabla f(x)\|^2\|P_1 - P_2\| \\
&\leq L\|\nabla f(x)\|^2\|P_1 - P_2\|_F,
\end{aligned}$$

where (39) uses the submultiplicativity of Frobenius norm $\|AB\|_F \leq \|A\|_F\|B\|_F$; and (40) uses L-smoothness of $f(x)$. Hence u_x is $L\|\nabla f(x)\|^2$-smooth. Since the surrogate loss $r_x(P) = \frac{u_x(P) - f(x^\star)}{f(x) - f(x^\star)}$ is a positive-scaled convex function u_x with translation, and hence r_x is also convex. Next, since $x \notin \mathcal{X}^\star$, the denominator of $r_x(P)$ must be positive, and hence $r_x(P) \geq 0$ for all P. Lastly, since $r_x(P) = \frac{u_x(P) - f(x^\star)}{f(x) - f(x^\star)}$ and u_x is $L\|\nabla f(x)\|^2$-smooth, $r_x(P)$ is also smooth with smoothness constant no greater than $2L^2$:

$$\tfrac{L\|\nabla f(x)\|^2}{f(x) - f(x^\star)} = \tfrac{L\|\nabla f(x) - \nabla f(x^\star)\|^2}{f(x) - f(x^\star)} \leq 2L^2,$$

where the last inequality invokes L-smoothness of $f(x)$. This completes the proof.

D.4. Proof of Lemma 2

For simplicity we denote $r_k(P) := r_{x^k}(P)$. By Proposition 1, the surrogate losses $\{r_k\}$ are $2L^2$-smooth and non-negative. Then using Lemma 11 with $\tau = 2L^2$, we get

$$\textstyle\sum_{k=1}^K r_k(P_k) - \sum_{k=1}^K r_k(P) \leq \frac{1}{\eta}\|P - P_1\|_F^2 + 4L^2\eta \sum_{k=1}^K r_k(P), \tag{41}$$

which proves (12). Suppose further $\mathrm{diam}(\mathcal{P}) \leq D$. Then

$$\begin{aligned} r_x(P) &= \tfrac{f(x-P\nabla f(x))-f(x^\star)}{f(x)-f(x^\star)} \\ &\leq \tfrac{f(x)-f(x^\star)+\langle \nabla f(x),(-P+\frac{L}{2}P^\top P)\nabla f(x)\rangle}{f(x)-f(x^\star)} \\ &\leq 1 + \|\tfrac{L}{2}P^\top P - P\| \tfrac{\|\nabla f(x)\|^2}{f(x)-f(x^\star)} \\ &\leq 1 + 2L(\tfrac{L}{2}D^2 + D) \\ &= (1+LD)^2. \end{aligned}$$

Therefore, (41) implies: for $\eta \leq \frac{1}{4L^2}$,

$$\textstyle\sum_{k=1}^K r_k(P_k) - \sum_{k=1}^K r_k(P) \leq \frac{1}{\eta}D^2 + 4L^2(1+LD)^2K\eta, \tag{42}$$

in which the right-hand side is minimized (as a function of η) at $\eta = \frac{D}{2L(1+LD)\sqrt{K}}$. By taking the stepsize $\eta = \min\big\{\frac{1}{4L^2}, \frac{D}{2L(1+LD)\sqrt{K}}\big\}$ and then minimizing over $P \in \mathcal{P}$, we conclude

$$\textstyle\sum_{k=1}^K r_{x^k}(P_k) \leq \min\limits_{P\in\mathcal{P}} \sum_{k=1}^K r_{x^k}(P) + \max\big\{4LD(1+LD)\sqrt{K}, 8L^2D^2\big\}.$$

and this completes the proof.

D.5. Proof of Theorem 2

By Lemma 2, we have

$$\tfrac{1}{K}\textstyle\sum_{k=1}^K r_{x^k}(P_k) \leq \tfrac{1}{K}\sum_{k=1}^K r_{x^k}(P) + \tfrac{\rho_K}{K}$$

for all $P \in \mathcal{P}$. and plugging the relation into Lemma 1 completes the proof.

D.6. Proof of Lemma 3

For any fixed $x \notin \mathcal{X}^\star$, the result $r_x(P_r^\star) \leq 1 - \frac{1}{\kappa^\star}$ is a direct consequence of (16).

D.7. Proof of Corollary 1

Using Lemma 3 and Theorem 2, $\theta_K^\star \leq 1 - \frac{1}{\kappa^\star}$ and plugging the bound back into Theorem 2 completes the proof. To show the specific iteration complexity, note that when $K \geq 64\lceil \kappa^\star LD(LD+1)]^2\rceil$, we have $\frac{4LD(LD+1)}{\sqrt{K}} \leq \frac{1}{2\kappa^\star}$ and we can upperbound

$$(1 - \tfrac{1}{\kappa^\star} + \tfrac{4LD(1+LD)}{\sqrt{K}})^K \leq (1 - \tfrac{1}{2\kappa^\star})^K.$$

Applying this upperbound completes the proof.

D.8. Proof of Theorem 3

Combining Lemma 1 and (12) from Lemma 2 and using the relation $r_x(P_r^\star) \leq 1 - \frac{1}{\kappa^\star}$ from Lemma 3, we have, for $\eta \leq \frac{1}{4L^2}$, that

$$\begin{aligned} & f(x^{K+1}) - f(x^\star) \\ \leq {} & (f(x^1) - f(x^\star))\big(\tfrac{1}{K}\textstyle\sum_{k=1}^K r_{x^k}(P_r^\star) + \tfrac{1}{\eta K}\|P_r^\star - P_1\|_F^2 + \tfrac{4L^2\eta}{K}\textstyle\sum_{k=1}^K r_{x^k}(P_r^\star)\big)^K \\ \leq {} & (f(x^1) - f(x^\star))\big(1 - \tfrac{1}{\kappa^\star} + \tfrac{1}{\eta K}\|P_r^\star - P_1\|_F^2 + 4L^2\eta\big)^K. \end{aligned} \tag{43}$$

Take the stepsize $\eta = \min\big\{\frac{1}{4L^2}, \frac{\|P_r^\star - P_1\|_F}{2L\sqrt{K}}\big\}$. The bound (43) implies the desired result:

$$f(x^{K+1}) - f(x^\star) \leq (f(x^1) - f(x^\star))\big(1 - \tfrac{1}{\kappa^\star} + \max\big\{\tfrac{4L\|P_r^\star - P_1\|_F}{\sqrt{K}}, \tfrac{8L^2\|P_r^\star - P_1\|_F^2}{K}\big\}\big)^K.$$

D.9. Proof of Proposition 2

Given the optimization problem (19),

$$\min_{P \in \mathcal{P}_+} \quad \kappa \quad \text{subject to} \quad \tfrac{1}{\kappa} I \preceq P^{1/2} A P^{1/2} \preceq I,$$

we can define $\tau = 1/\kappa$ and reduce it to a standard semidefinite optimization problem (SDP)

$$\max_{P \in \mathcal{P}_+} \quad \tau = \kappa^{-1} \quad \text{subject to} \quad A^{-1}\tau \preceq P \preceq A^{-1}. \tag{44}$$

On the other hand, using $f(x) = \frac{1}{2}\langle x, A, x\rangle$, we can explicitly write

$$r_x(P) = \frac{\frac{1}{2}\langle x, A(PAP - 2P)Ax\rangle}{\frac{1}{2}\langle x, A, x\rangle}$$

and r_x is degree-zero homogeneous in x. Therefore, we can consider the following optimization problem

$$\min_{P \in \mathcal{P}_+} \max_{\langle x, Ax\rangle = 1} \quad \langle x, A(PAP - 2P)Ax\rangle,$$

which can be further re-written as

$$\max_{P \in \mathcal{P}_+} \quad \lambda \quad \text{subject to} \quad 2A^{1/2}PA^{1/2} - A^{1/2}PAPA^{1/2} \succeq \lambda I$$

Next we do variable replacement by letting $M := A^{1/2}PA^{1/2}$ and $\mathcal{P}_+' = \{M = A^{1/2}PA^{1/2} : P \in \mathcal{P}_+\}$ and it suffices to show the equivalence between the following two problems.

$$\max_{M \in \mathcal{P}_+'} \quad \tau \quad \text{subject to} \quad \tau I \preceq M \preceq I \tag{SDP}$$

$$\max_{M \in \mathcal{P}_+'} \quad \lambda \quad \text{subject to} \quad 2M - M^2 \succeq \lambda I \tag{Minimax}$$

Given optimal solution $(M_1^\star, \tau^\star)$ to (SDP), we have $\tau^\star I \preceq M_1^\star \preceq I$ and let $M_1^\star = Q\Lambda_1 Q^\top$. Plugging $M_1^\star$ into the constraint,

$$2M_1^\star - (M_1^\star)^2 = Q(2\Lambda_1 - \Lambda_1^2)Q^\top,$$

which corresponds to $\lambda = 2\tau^\star - (\tau^\star)^2$. On the other hand, given optimal solution $(M_2^\star, \lambda^\star)$ to (Minimax), $2M_2^\star - M_2^{\star 2} \succeq \lambda^\star I$ and similarly we let $M_2^\star = Q\Lambda_2 Q^\top$. Then

$$Q(2\Lambda_2 - \Lambda_2^2)Q^\top \succeq \lambda^\star I$$

and there exists some q_j such that $2\lambda_{2j} - \lambda_{2j}^2 = \lambda^\star$. It corresponds to $\tau = 2\tau^2 - \tau = \lambda^\star$. This establishes the equivalence between the two problems and completes the proof.

D.10. Proof of Theorem 4

Recall that by (43) we have

$$f(x^{K+1}) - f(x^\star) \leq (f(x^1) - f(x^\star))(\tfrac{1}{K}\textstyle\sum_{k=1}^K r_{x^k}(P_r^\star) + \tfrac{1}{\eta}\|P_r^\star - P_1\|_F^2 + 4L^2\eta \textstyle\sum_{k=1}^K r_{x^k}(P_r^\star)).$$

Using $r_{x^k}(P_r^\star) = 0, P_r^\star = A^{-1}$ and taking $\eta = 1/(4L^2)$, we get

$$f(x^{K+1}) - f(x^\star) \leq (f(x^1) - f(x^\star))(\tfrac{4L^2\|P_1 - A^{-1}\|_F^2}{K})^K.$$

This completes the proof.

Appendix E. Function value ratio surrogate with optimal value lower bound

This section analyzes the sub-optimal ratio surrogate loss $r_x^z(P)$ defined by

$$r_x^z(P) := \tfrac{f(x - P\nabla f(x)) - z}{f(x) - z} = \tfrac{f(x^+) - z}{f(x) - z}, \tag{45}$$

where $z < f(x^\star)$ is a lower bound for the optimal objective value. The challenging part of the analysis when $z < f(x^\star)$ is that the algorithm is only guaranteed to converge to some suboptimal solution whose suboptimality is determined by $f(x^\star) - z$, the accuracy of the lower bound. The analysis in this section is more involved than in Section 4, and for clarity, we *only present the global convergence result*. The main idea is to search for $f(x^\star)$ through a double-loop procedure. The analysis is motivated by Hazan and Kakade (2019).

E.1. Surrogate loss

Lemma 12 (Surrogate loss and measure)**.** *For all $K \geq 1$, the online scaled gradient method satisfies*

$$f(x^{K+1}) - z \leq (f(x^1) - z)\big(\tfrac{1}{K}\textstyle\sum_{k=1}^K r_{x^k}^z(P_k)\big)^K. \tag{46}$$

Proposition 5 (Properties of $r_{x^k}^z$)**.** *Let $z < f(x^\star)$ be a given lower bound. Under* **A1** *and* **A2***, for any fixed x, the surrogate loss $r_x^z(P)$ defined in* (45) *is convex, non-negative, and $2L^2$-smooth as a function in P. In addition, the derivative of r_x^z takes the form*

$$\nabla r_x^z(P) = -\tfrac{\nabla f(x - P\nabla f(x))\nabla f(x)^\top}{f(x) - z}.$$

E.2. Online learning algorithm

Lemma 13 (Learnability)**.** *Given* **A1**, **A2**, *and the ratio surrogate losses* $\{r^z_{x^k}\}$, *online gradient descent*

$$P_{k+1} = \Pi_{\mathcal{P}}[P_k - \eta \nabla r^z_{x^k}(P_k)] \tag{47}$$

with stepsize $\eta \leq 1/(4L^2)$ *generates a sequence of scaling matrices* $\{P_k\}_{k\geq 2}$ *such that*

$$\textstyle\sum_{k=1}^K r^z_{x^k}(P_k) - \sum_{k=1}^K r^z_{x^k}(P) \leq \frac{1}{\eta}\|P - P_1\|_F^2 + 4L^2\eta \sum_{k=1}^K r^z_{x^k}(P) \quad \text{for any } P \in \mathcal{P}. \tag{48}$$

E.3. Algorithm design and analysis

In this section, we show how to obtain an $\mathcal{O}(\kappa^\star \log^2(1/\varepsilon))$ complexity through a double-loop algorithm. Since the double-loop algorithm deviates from our framework, only global convergence is established for brevity. We start by specifying the OSGM-RZ, a subroutine that will be invoked in the inner loop.

We choose the optimality measure φ, the surrogate loss ℓ, and the online learning algorithm $\mathcal{A}$ to be

$$\varphi(x) := f(x) - f(x^\star), \quad \ell_x(P) := r^z_x(P), \quad \mathcal{A} := \text{online gradient descent in (47)},$$

and the monotone oracle $\mathcal{M}$ is optional. Algorithm **5** presents OSGM-RZ without the monotone oracle.

Algorithm 5: Online scaled gradient method with lower bound ratio surrogate (OSGM-RZ)

```
input $x^1, P_1 \in \mathcal{P}, \eta > 0, z < f(x^\star)$
for k = 1, 2,... do
    $x^{k+1} = x^k - P_k \nabla f(x^k)$
    $P_{k+1} = \Pi_{\mathcal{P}}[P_k - \eta \nabla r^z_{x^k}(P_k)]$
end
output $x^{\text{best}}$ with minimum objective value
```

Theorem **7** characterizes the convergence behavior of OSGM-RZ.

Theorem 7 (Global convergence with lower bound)**.** *Under* **A1** *to* **A3**, *Algorithm* **5** *(OSGM-RZ) with* $\eta = \min\{\frac{1}{4L^2}, \frac{\|P_r^\star - P_1\|_F}{2L\sqrt{K}}\}$ *satisfies*

$$\min_{1\leq k\leq K+1} f(x^k) - f(x^\star) \leq \tfrac{1}{2}(f(x^\star) - z) + (f(x^1) - f(x^\star))(1 - \tfrac{1}{2\kappa^\star} + \tfrac{\rho_K}{K})^K,$$

where $\rho_K := \max\{4L\sqrt{K}\|P_r^\star - P_1\|_F, 8L^2\|P_r^\star - P_1\|_F^2\}$.

Lemma 14 (Lower bound update)**.** *Under the same assumptions and parameter choice as Theorem* **7** *and denote*

$$z^+ = \frac{1}{2}\Big[\min_{1\leq k\leq K+1} f(x^k) + z\Big].$$

Then exactly one of the cases below happens:

- $f(x^{K+1}) - f(x^\star) \leq (f(x^1) - f(x^\star))(1 - \frac{1}{2\kappa^\star} + \frac{\rho_K}{K})^K$, *or*
- $f(x^\star) - z^+ \leq \frac{1}{2}(f(x^\star) - z)$ *and* $z^+ \leq f(x^\star)$.

Algorithm 6: Online scaled gradient method with ratio surrogate and lower bound update

```
input $x^1, P_1 \in \mathcal{P}, \eta > 0, z^1 < f(x^\star)$
for t = 1, 2,... do
    $x^{t+1} = \texttt{OSGM-RZ}(x^t, P_1, \eta, z^t)$
    $z^{t+1} = \frac{1}{2}(f(x^{t+1}) + z^t)$
end
output $x^{\text{best}}$ with minimum objective value
```

Lemma 14 suggests that the output of `OSGM-RZ` either already satisfies the desirable convergence result, or the accuracy of the lower bound can be improved by a factor of 2. This motivates the idea of running `OSGM-RZ` multiple times and outputting the best solution, as presented in Algorithm 6. Theorem 8 provides the final convergence result.

Theorem 8. *Under the same assumptions and parameter choices as* ***Theorem 7***, ***Algorithm 6*** *attains $f(x^{\text{best}}) - f(x^\star) \leq \varepsilon$ in at most $\mathcal{O}(\kappa^\star \log^2(1/\varepsilon))$ scaled gradient iterations.*

E.4. Proof of Lemma 12

Since $f(x) - z > 0$, applying Theorem 1 with $\varphi(x) = f(x) - z$ completes the proof.

E.5. Proof of Proposition 5

Since $z < f(x)$, both the numerator and the denominator of r_x^z are positive. Following the proof of Proposition 1, we can show that r_x^z is convex. Since u_x is $L\|\nabla f(x)\|^2$-smooth, r_x^z is $\frac{L\|\nabla f(x)\|^2}{f(x)-z}$-smooth, and

$$\tfrac{L\|\nabla f(x)\|^2}{f(x)-z} < \tfrac{L\|\nabla f(x)\|^2}{f(x)-f(x^\star)} = \tfrac{L\|\nabla f(x)-\nabla f(x^\star)\|^2}{f(x)-f(x^\star)} \leq 2L^2,$$

which completes the proof.

E.6. Proof of Lemma 13

By Proposition 5, the surrogate losses $\{r_{x^k}^z\}$ are $2L^2$-smooth and non-negative. Applying Lemma 11 with $\tau = 2L^2$ completes the proof.

E.7. Proof of Theorem 7

Using Lemma 12 and Lemma 13 with $P = P_r^\star$,

$$\begin{aligned} f(x^{K+1}) - z \leq{}& (f(x^1) - z)(\tfrac{1}{K}\textstyle\sum_{k=1}^K r_{x^k}^z(P_k))^K \\ \leq{}& (f(x^1) - z)(\tfrac{1}{K}\textstyle\sum_{k=1}^K r_{x^k}^z(P_r^\star) + \tfrac{1}{K}[\tfrac{1}{\eta}\|P - P_r^\star\|_F^2 + 4L^2\eta\textstyle\sum_{k=1}^K r_{x^k}^z(P_r^\star)])^K \\ \leq{}& (f(x^1) - z)(\tfrac{1}{K}\textstyle\sum_{k=1}^K r_{x^k}^z(P_r^\star) + \tfrac{1}{K}[\tfrac{1}{\eta}\|P - P_r^\star\|_F^2 + 4L^2\eta])^K, \end{aligned} \tag{49}$$

where (49) uses $f(x - P_r^\star \nabla f(x)) \leq f(x)$ and that $r_x^z(P_r^\star) = \frac{f(x-P_r^\star\nabla f(x))-z}{f(x)-z} \leq 1$.

Taking $\eta = \min\{\frac{1}{4L^2}, \frac{\|P_r^\star - P_1\|_F}{2L\sqrt{K}}\}$ gives

$$\begin{aligned} f(x^{K+1}) - z &\leq (f(x^1) - z)\big(\tfrac{1}{K}\textstyle\sum_{k=1}^K r_{x^k}^z(P_r^\star) + \max\big\{\tfrac{4L\|P_r^\star - P_1\|_F}{\sqrt{K}}, \tfrac{8L^2\|P_r^\star - P_1\|_F^2}{K}\big\}\big)^K \\ &= (f(x^1) - z)\big(\tfrac{1}{K}\textstyle\sum_{k=1}^K r_{x^k}^z(P_r^\star) + \tfrac{\rho_K}{K}\}\big)^K. \end{aligned}$$

Next we analyze $\frac{1}{K}\sum_{k=1}^K r_{x^k}^z(P_r^\star)$, and using

$$f(x - P_r^\star \nabla f(x)) - f(x^\star) \leq (1 - \tfrac{1}{\kappa^\star})(f(x) - f(x^\star)),$$

we deduce that

$$\begin{aligned} f(x - P_r^\star\nabla f(x)) - z &= f(x - P_r^\star\nabla f(x)) - f(x^\star) + f(x^\star) - z \\ &\leq (1 - \tfrac{1}{\kappa^\star})[f(x) - f(x^\star)] + f(x^\star) - z \\ &= (1 - \tfrac{1}{\kappa^\star})[f(x) - z] - (1 - \tfrac{1}{\kappa^\star})[f(x^\star) - z] + f(x^\star) - z \\ &= (1 - \tfrac{1}{\kappa^\star})[f(x) - z] + \tfrac{1}{\kappa^\star}[f(x^\star) - z] \end{aligned}$$

Dividing both sides by $f(x) - z$ gives

$$\tfrac{f(x - P_r^\star\nabla f(x)) - z}{f(x) - z} = (1 - \tfrac{1}{\kappa^\star}) + \tfrac{1}{\kappa^\star}\tfrac{f(x^\star) - z}{f(x) - z} = 1 - \tfrac{1}{\kappa^\star}\tfrac{f(x) - f(x^\star)}{f(x) - z}.$$

Hence $\frac{1}{K}\sum_{k=1}^K r_{x^k}^z(P_r^\star) \leq 1 - \frac{1}{\kappa^\star}(\frac{1}{K}\sum_{k=1}^K \frac{f(x^k) - f(x^\star)}{f(x^k) - z})$. Now, we do case analysis

Case 1. Suppose $\frac{f(x^k) - f(x^\star)}{f(x^k) - z} \geq \frac{1}{2}$ for all $1 \leq k \leq K$, then $\frac{1}{K}\sum_{k=1}^K r_{x^k}^z(P_r^\star) \leq 1 - \frac{1}{2\kappa^\star}$ and

$$\begin{aligned} \min_{1 \leq k \leq K+1} f(x^k) - f(x^\star) &\leq f(x^{K+1}) - f(x^\star) \\ &\leq (f(x^1) - f(x^\star))\big(1 - \tfrac{1}{2\kappa^\star} + \tfrac{\rho_K}{K}\big)^K. \end{aligned}$$

Case 2. Otherwise, there is some $1 \leq j \leq K$ such that $\frac{f(x^j) - f(x^\star)}{f(x^j) - z} \leq \frac{1}{2}$, a re-arrangement gives $2f(x^j) - 2f(x^\star) \leq f(x^j) - z$ and

$$\min_{1 \leq k \leq K+1} f(x^k) - f(x^\star) \leq f(x^j) - f(x^\star) \leq \tfrac{1}{2}(f(x^j) - z).$$

Putting the two cases together, we complete the proof.

E.8. Proof of Lemma 14

The argument is the same as in **Theorem** 7. In **Case 1**, we get the first convergence result. Otherwise, we know that there exists some $1 \leq j \leq K$ such that $\frac{f(x^j) - f(x^\star)}{f(x^j) - z} \leq \frac{1}{2}$ and since $\min_{1 \leq k \leq K} f(x^k) \leq f(x^j)$, we have

$$\tfrac{\min_{1 \leq k \leq K} f(x^k) - f(x^\star)}{\min_{1 \leq k \leq K} f(x^k) - z} \leq \tfrac{f(x^j) - f(x^\star)}{f(x^j) - z} \leq \tfrac{1}{2}.$$

Rearranging the relation, we have $z^+ = \frac{1}{2}[\min_{1 \leq k \leq K} f(x^k) + z] \leq f(x^\star)$ and

$$\begin{aligned} f(x^\star) - \tfrac{1}{2}\big[\min_{1 \leq k \leq K} f(x^k) + z\big] &= \tfrac{1}{2}[f(x^\star) - \min_{1 \leq k \leq K} f(x^k)] + \tfrac{1}{2}[f(x^\star) - z] \\ &\leq \tfrac{1}{2}[f(x^\star) - z]. \end{aligned}$$

This completes the proof.

E.9. Proof of Theorem 8

Denote x^{t+1} as the output of `OSGM-RZ` in iteration t of **Algorithm 6**. If we fall into **Case 1** in **Lemma 14** after running `OSGM-RZ` for K iterations, then

$$f(x^{t+1}) - f(x^\star) \leq (f(x^t) - f(x^\star))\big(1 - \tfrac{1}{2\kappa^\star} + \tfrac{\rho_K}{K}\big)^K \tag{50}$$

since x^t is the initial point of `OSGM-RZ` in iteration t. Using the fact that z^1 is a lower bound for $f(x^\star)$, algebraic manipulation shows that the right-hand side of (50) is less than ε whenever

$$K \geq \tfrac{128\|P_r^\star - P_1\|_F^2 L^2}{\log^2(1 - \frac{1}{2\kappa^\star})} + 2\kappa^\star \log\big(\tfrac{f(x^1) - z^1}{\varepsilon}\big) =: K_0.$$

We claim that if we run `OSGM-RZ` for K_0 iterations at each iteration t in **Algorithm 6** and run **Algorithm 6** for $T := \frac{1}{\log 2}\log\big(\frac{4(f(x^1)-z^1)}{\varepsilon}\big)$ iterations, then we have $f(x^{\text{best}}) - f(x^\star) \leq \varepsilon$ where x^{best} is the point in $\{x^t : t = 1, \ldots, T+1\}$ with the smallest function value. Hence, **Algorithm 6** takes at most $K_0 T = \mathcal{O}(\kappa^\star \log^2(1/\varepsilon))$ scaled gradient iterations.

We will show that at least one of the iterates in $\{x^t : t = 1, \ldots, T+1\}$ from our algorithm satisfies $f(x^t) - f(x^\star) < \varepsilon$. If we fall into **Case 1** in **Lemma 14** for some iteration t, then we have $f(x^{t+1}) - f(x^\star) \leq \varepsilon$ by (50). Otherwise, we fall into **Case 2** in **Lemma 14** for all $t \leq T$. In this case, we halve the distance between z^t and $f(x^\star)$ after every outer iteration, so that after $T := \lceil \frac{1}{\log 2}\log\big(\frac{4(f(x^1)-z^1)}{\varepsilon}\big)\rceil$ iterations, we have

$$|z^T - f(x^\star)| \leq \big(\tfrac{1}{2}\big)^{T-1}(f(x^\star) - z^1) \leq \big(\tfrac{1}{2}\big)^{T-1}(f(x^1) - z^1) \leq \tfrac{\varepsilon}{2}.$$

Since $z^{T+1} = \frac{1}{2}(f(x^{t+1}) + z^T)$ and we fall into **Case 2** at iteration T, we have

$$|z^{T+1} - f(x^\star)| = \big|\tfrac{f(x^{T+1}) + z^T}{2} - f(x^\star)\big| \leq \tfrac{1}{2}|z^T - f(x^\star)| \leq \tfrac{\varepsilon}{4}.$$

Rearranging the relation, we have $f(x^{T+1}) \leq f(x^\star) + \frac{1}{2}\varepsilon + (f(x^\star) - z^T) \leq f(x^\star) + \varepsilon$. This completes the proof.

Appendix F. Proof of results in Section 5

F.1. Proof Lemma 4

Given monotone oracle $\mathcal{M}$ with respect to gradient norm and by definition of g_x,

$$\|\nabla f(x^{k+1})\| = \|\nabla f(\mathcal{M}(x^k))\| \leq \|\nabla f(x^k - P_k \nabla f(x^k))\| = g_{x^k}(P_k)\|\nabla f(x^k)\|.$$

Hence, through the same argument as in Theorem 1, we deduce that

$$\tfrac{\|\nabla f(x^{K+1})\|}{\|\nabla f(x^1)\|} = \textstyle\prod_{k=1}^K \tfrac{\|\nabla f(x^{k+1})\|}{\|\nabla f(x^k)\|} \leq (\tfrac{1}{K}\sum_{k=1}^K \tfrac{\|\nabla f(x^{k+1})\|}{\|\nabla f(x^k)\|})^K \leq (\tfrac{1}{K}\sum_{k=1}^K g_{x^k}(P_k))^K$$

and this completes the proof.

F.2. Proof of Proposition 3

Lipschitzness of g_x is straight-forward:

$$\begin{aligned}|g_x(P_1) - g_x(P_2)| &= \left|\tfrac{\|\nabla f(x-P_1\nabla f(x))\|}{\|\nabla f(x)\|} - \tfrac{\|\nabla f(x-P_2\nabla f(x))\|}{\|\nabla f(x)\|}\right| \\ &\leq \tfrac{\|\nabla f(x-P_1\nabla f(x))-\nabla f(x-P_2\nabla f(x))\|}{\|\nabla f(x)\|} \qquad (51)\\ &\leq \tfrac{L\|P_1-P_2\|\cdot\|\nabla f(x)\|}{\|\nabla f(x)\|} \qquad (52)\\ &= L\|P_1 - P_2\| \leq L\|P_1 - P_2\|_F\end{aligned}$$

where (51) uses the triangle inequality $|\|a\| - \|b\|| \leq \|a - b\|$ and (52) uses L-smoothness of f. Next consider $|g_x(P) - \hat{g}_x(P)|$ and we deduce that

$$\begin{aligned}&|g_x(P) - \hat{g}_x(P)| \\ = &\left|\left\|\tfrac{\nabla f(x)}{\|\nabla f(x)\|} - \textstyle\int_0^1 \nabla^2 f(x - tP\nabla f(x))P\tfrac{\nabla f(x)}{\|\nabla f(x)\|}\,\mathrm{d}t\right\| - \left\|\tfrac{\nabla f(x)}{\|\nabla f(x)\|} - \nabla^2 f(x)P\tfrac{\nabla f(x)}{\|\nabla f(x)\|}\right\|\right| \\ \leq &\ \| \textstyle\int_0^1 \nabla^2 f(x - tP\nabla f(x))P\tfrac{\nabla f(x)}{\|\nabla f(x)\|} - \nabla^2 f(x)P\tfrac{\nabla f(x)}{\|\nabla f(x)\|}\,\mathrm{d}t\| \qquad (53)\\ \leq &\ \textstyle\int_0^1 \|\nabla^2 f(x - tP\nabla f(x)) - \nabla^2 f(x)\|\,\mathrm{d}t \cdot \big(\tfrac{\|P\|\|\nabla f(x)\|}{\|\nabla f(x)\|}\big) \qquad (54)\\ \leq &\ H \textstyle\int_0^1 \|P\nabla f(x)\|t\,\mathrm{d}t \cdot \|P\| \qquad (55)\\ \leq &\ \tfrac{1}{2}H\|P\|^2\|\nabla f(x)\| \qquad (56)\end{aligned}$$

where (53) again uses $|\|a\| - \|b\|| \leq \|a - b\|$ and (54) uses the Lispchitzness of the Hessian; (55) uses $\|AB\| \leq \|A\|\|B\|$. Convexity of $\hat{g}_x$ is straight-forward since $\hat{g}_x$ is a composition of linear function (in P) with norm $\|\cdot\|$. To show L-Lipschitzness of $\hat{g}_x$, we have

$$\begin{aligned}|\hat{g}_x(P_1) - \hat{g}_x(P_2)| &= \left|\left\|\tfrac{\nabla f(x)}{\|\nabla f(x)\|} - \nabla^2 f(x)P_1\tfrac{\nabla f(x)}{\|\nabla f(x)\|}\right\| - \left\|\tfrac{\nabla f(x)}{\|\nabla f(x)\|} - \nabla^2 f(x)P_2\tfrac{\nabla f(x)}{\|\nabla f(x)\|}\right\|\right| \\ &\leq \tfrac{1}{\|\nabla f(x)\|}\|\nabla^2 f(x)(P_1 - P_2)\nabla f(x)\| \qquad (57)\\ &\leq L\|P_1 - P_2\|, \qquad (58)\end{aligned}$$

where (57) again uses $|\|a\| - \|b\|| \leq \|a - b\|$ and (58) uses $\|\nabla^2 f(x)\| \leq L$. Last, we combine the convex subgradient lower bound of $\hat{g}_x$ with the approximation

$$\begin{aligned}g_x(P_1) &\geq \hat{g}_x(P_1) - \tfrac{1}{2}H\|P_1\|^2\|\nabla f(x)\| \qquad (59)\\ &\geq \hat{g}_x(P_2) + \langle \hat{g}_x'(P_2), P_1 - P_2\rangle - \tfrac{1}{2}H\|P_1\|^2\|\nabla f(x)\| \qquad (60)\\ &\geq g_x(P_2) + \langle \hat{g}_x'(P_2), P_1 - P_2\rangle - \tfrac{1}{2}H[\|P_1\|^2 + \|P_2\|^2]\|\nabla f(x)\| \qquad (61)\\ &\geq g_x(P_2) + \langle \hat{g}_x'(P_2), P_1 - P_2\rangle - HD^2\|\nabla f(x)\|,\end{aligned}$$

where (59) uses (56), (60) uses convexity of $\hat{g}_x$ and (61) again applies (56). This completes the proof.

F.3. Proof of Lemma 5

Denote $g_k(P_k) := g_{x^k}(P_k)$. For any $P \in \mathcal{P}$, we have

$$\begin{aligned}\|P_{k+1} - P\|_F^2 &= \|\Pi_{\mathcal{P}}[P_k - \eta g_k'(P_k)]\|_F^2 \\ &\leq \|P_k - P - \eta g_k'(P_k)\|_F^2 \\ &= \|P_k - P\|_F^2 - 2\eta\langle g_k'(P_k), P_k - P\rangle + \eta^2\|g_k'(P_k)\|_F^2 \\ &\leq \|P_k - P\|_F^2 - 2\eta[g_k(P_k) - g_k(P)] + \eta^2 L^2 + \eta H D^2\|\nabla f(x^k)\|, \qquad (62)\end{aligned}$$

where (62) invokes Proposition 3. Dividing both sides by η and re-arranging the terms,

$$\begin{aligned}g_k(P_k) - g_k(P) &\leq \tfrac{\|P_k - P\|_F^2}{2\eta} - \tfrac{\|P_{k+1} - P\|_F^2}{2\eta} + \tfrac{\eta}{2}L^2 + \tfrac{HD^2}{2}\|\nabla f(x^k)\| \\ &\leq \tfrac{\|P_k - P\|_F^2}{2\eta} - \tfrac{\|P_{k+1} - P\|_F^2}{2\eta} + \tfrac{\eta}{2}L^2 + \tfrac{HD^2}{2}\|\nabla f(x^1)\|, \qquad (63)\end{aligned}$$

where (63) uses we use the fact that $\|\nabla f(x^k)\| \leq \|\nabla f(x^1)\|$. Summing both sides from $k = 1, \ldots, K$, we get the desired result:

$$\textstyle\sum_{k=1}^K g_k(P_k) - \sum_{k=1}^K g_k(P) \leq \frac{\|P_1 - P\|_F^2}{2\eta} + \frac{\eta}{2}L^2 K + \frac{HD^2}{2}\|\nabla f(x^1)\|K. \qquad (64)$$

Finally, using the bound $\|P_1 - P\|_F^2 \leq 4D^2$ and plugging in the stepsize $\eta = \frac{2D}{L\sqrt{K}}$ yield (24).

F.4. Proof of Theorem 5

By Lemma 5, we have

$$\tfrac{1}{K}\textstyle\sum_{k=1}^K g_{x^k}(P_k) \leq \min\limits_{P \in \mathcal{P}} \tfrac{1}{K}\sum_{k=1}^K g_{x^k}(P) + \tfrac{\rho_K}{K}$$

Plugging the relation into Lemma 4 completes the proof.

F.5. Proof of Lemma 6

We show an alternative version of Lemma 6.

Lemma 15 (Hindsight). *Instate A1 to A3. Then the followings hold:*

- *Contraction.* $\|\nabla f(x - P_g^\star \nabla f(x))\| \leq (1 - \frac{1}{\lambda^\star})\|\nabla f(x)\|$ *for all* x.
- *Conditioning.* $\lambda^\star \leq \frac{L}{\mu} = \kappa$.
- *Surrogate loss bound.* $g_x(P_g^\star) \leq 1 - \frac{1}{\lambda^\star}$ *for all* $x \notin \mathcal{X}^\star$.

The first relation follows from

$$\begin{aligned}\|\nabla f(x - P_g^\star \nabla f(x))\| &\leq [\textstyle\int_0^1 \|I - \nabla^2 f(x - tP_g^\star \nabla f(x))P_g^\star\|\mathrm{d}t] \cdot \|\nabla f(x)\| \\ &\leq \omega^\star\|\nabla f(x)\| = (1 - \tfrac{1}{\lambda^\star})\|\nabla f(x)\|. \qquad (65)\end{aligned}$$

The fact that $\frac{\mu}{L}I \preceq L^{-1}\nabla^2 f(x) \preceq I$ for all x implies $\|I - L^{-1}\nabla^2 f(x)\| \leq 1 - \frac{\mu}{L}$ for all x. Hence, by taking $P = \frac{1}{L}I \in \mathcal{P}$, we conclude

$$\omega^\star := \min_{P \in \mathcal{P}} \max_x \|I - \nabla^2 f(x)P\| \leq \max_x \|I - L^{-1}\nabla^2 f(x)\| \leq 1 - \tfrac{\mu}{L}.$$

The desired inequality $\lambda^\star \le \frac{L}{\mu}$ immediately follows from the definition $\lambda^\star = \frac{1}{1-\omega^\star}$. Finally, rearranging (65) gives the desired bound on gradient norm surrogate loss:

$$g_x(P_g^\star) = \tfrac{\|\nabla f(x - P_g^\star \nabla f(x))\|}{\|\nabla f(x)\|} \le 1 - \tfrac{1}{\lambda^\star}.$$

Remark 8. We can link $\lambda^\star$ and $\kappa^\star$ through two relations below:

$$\begin{aligned} \|\nabla f(x - P_g^\star \nabla f(x))\| &\le (1 - \tfrac{1}{\lambda^\star})\|\nabla f(x)\|, \\ \|\nabla f(x - P_r^\star \nabla f(x))\|_{P_r^\star} &\le (1 - \tfrac{1}{\kappa^\star})\|\nabla f(x)\|_{P_r^\star}. \end{aligned} \tag{66}$$

The second relation (66) holds by simple algebraic derivation: since $\frac{1}{\kappa^\star} I \preceq (P_r^\star)^{1/2} \nabla^2 f(x) (P_r^\star)^{1/2} \preceq I$ for all x, we deduce that

$$\begin{aligned} &\|\nabla f(x - P_r^\star \nabla f(x))\|_{P_r^\star}^2 \\ =\, &\|\nabla f(x) - \textstyle\int_0^1 \nabla^2 f(x - tP_r^\star \nabla f(x)) P_r^\star \nabla f(x) \mathrm{d}t\|_{P_r^\star}^2 \\ =\, &\| \textstyle\int_0^1 (I - \nabla^2 f(x - tP_r^\star \nabla f(x)) P_r^\star) \nabla f(x)\, \mathrm{d}t\|_{P_r^\star}^2 \\ =\, &\langle \textstyle\int_0^1 (I - \nabla^2 f(x - tP_r^\star \nabla f(x)) P_r^\star) \nabla f(x)\, \mathrm{d}t, P_r^\star \int_0^1 (I - \nabla^2 f(x - tP_r^\star \nabla f(x)) P_r^\star) \nabla f(x)\, \mathrm{d}t \rangle \\ =\, &\langle (P_r^\star)^{1/2} \nabla f(x), (\textstyle\int_0^1 (I - M_t)\, \mathrm{d}t)^2 (P_r^\star)^{1/2} \nabla f(x) \rangle, \end{aligned}$$

where $M_t := (P_r^\star)^{1/2} \nabla^2 f(x - tP^\star \nabla f(x)) (P_r^\star)^{1/2}$. Using the fact that $\frac{1}{\kappa^\star} \preceq M_t \preceq I$, we have $\int_0^1 (I - M_t)^2 \mathrm{d}t \preceq (1 - \frac{1}{\kappa^\star})^2 I$ and hence

$$\|\nabla f(x - P_r^\star \nabla f(x))\|_{P_r^\star}^2 \le (1 - \tfrac{1}{\kappa^\star})^2 \|\nabla f(x)\|_{P_r^\star}^2.$$

Taking square root on both sides gives the desired relation. However, since evaluating $\|\cdot\|_{P_r^\star}$ requires knowledge of $P_r^\star$, we have to define auxiliary quantity $\lambda^\star$ and $P_g^\star$.

F.6. Proof of Corollary 2

With convergence results from standard gradient descent, it takes $\mathcal{O}(\kappa \log(HD^2\lambda^\star))$ iterations to output $\hat{x}$ such that $\|\nabla f(\hat{x})\| \le \frac{1}{HD^2\lambda^\star}$. Next let $x^1 = \hat{x}$. Using Lemma 6 and Theorem 5, $\theta_K^\star \le 1 - \frac{1}{\lambda^\star}$ and

$$\begin{aligned} \|\nabla f(x^{K+1})\| &\le \|\nabla f(x^1)\| (1 - \tfrac{1}{\lambda^\star} + \tfrac{2DL}{\sqrt{K}} + \tfrac{HD^2}{2} \|\nabla f(x^1)\|)^K \\ &\le \|\nabla f(x^1)\| (1 - \tfrac{1}{\lambda^\star} + \tfrac{2DL}{\sqrt{K}} + \tfrac{1}{2\lambda^\star})^K \\ &\le \|\nabla f(x^1)\| (1 - \tfrac{1}{2\lambda^\star} + \tfrac{2DL}{\sqrt{K}})^K, \end{aligned} \tag{67}$$

where (67) uses the assumption that $\|\nabla f(x^1)\| \le \frac{1}{HD^2\lambda^\star}$. This completes the proof.

Appendix G. Proof of results in Section 6

G.1. Proof of Lemma 7

For convenience we denote $x^{k+1/2} := x^k - P_k \nabla f(x^k)$. By definition of the monotone oracle, we always have

$$f(x^{k+1}) = f(\mathcal{M}(x^k)) \le \min\{f(x^k), f(x^{k+1/2})\}.$$

Proof of relation (26). Suppose $\mu \neq 0$. By definition of $h_x(P)$, we can write

$$\begin{aligned} f(x^{k+1/2}) - f(x^\star) &= f(x^k) - f(x^\star) + h_{x^k}(P_k)\|\nabla f(x^k)\|^2 \\ &= (f(x^k) - f(x^\star))\Big[1 + \tfrac{h_{x^k}(P_k)\|\nabla f(x^k)\|^2}{f(x^k)-f(x^\star)}\Big]. \end{aligned} \tag{68}$$

Since $f(x^{k+1}) = f(\mathcal{M}(x^k)) \leq \min\{f(x^k), f(x^{k+1/2})\}$, we have

$$\begin{aligned} f(x^{k+1}) - f(x^\star) \leq{} & \min\{f(x^k) - f(x^\star), f(x^{k+1/2}) - f(x^\star)\} \\ ={} & (f(x^k) - f(x^\star))\Big[1 + \min\Big\{\tfrac{h_{x^k}(P_k)\|\nabla f(x^k)\|^2}{f(x^k)-f(x^\star)}, 0\Big\}\Big], \end{aligned} \tag{69}$$

where (69) uses (68). We successively deduce that

$$\begin{aligned} \tfrac{f(x^{K+1})-f(x^\star)}{f(x^1)-f(x^\star)} &= \textstyle\prod_{k=1}^K \tfrac{f(x^{k+1})-f(x^\star)}{f(x^k)-f(x^\star)} \\ &\leq \big(\tfrac{1}{K}\textstyle\sum_{k=1}^K \tfrac{f(x^{k+1})-f(x^\star)}{f(x^k)-f(x^\star)}\big)^K \\ &\leq \Big(1 + \tfrac{1}{K}\textstyle\sum_{k=1}^K \min\Big\{\tfrac{h_{x^k}(P_k)\|\nabla f(x^k)\|^2}{f(x^k)-f(x^\star)}, 0\Big\}\Big)^K \\ &\leq \big(1 + \tfrac{2\mu}{K}\textstyle\sum_{k=1}^K \min\{h_{x^k}(P_k), 0\}\big)^K, \end{aligned} \tag{70}$$

where (70) is by $\frac{1}{2\mu}\|\nabla f(x^k)\|^2 \geq f(x^k) - f(x^\star)$ and $\min\{h_{x^k}(P_k), 0\} \leq 0$:

$$\min\Big\{\tfrac{h_{x^k}(P_k)\|\nabla f(x^k)\|^2}{f(x^k)-f(x^\star)}, 0\Big\} = \tfrac{\|\nabla f(x^k)\|^2}{f(x^k)-f(x^\star)} \cdot \min\{h_{x^k}(P_k), 0\} \leq 2\mu \min\{h_{x^k}(P_k), 0\}.$$

By concavity of $\min\{\cdot, 0\}$, we have

$$1 + \tfrac{2\mu}{K}\textstyle\sum_{k=1}^K \min\{h_{x^k}(P_k), 0\} \leq 1 + 2\mu \min\{\tfrac{1}{K}\textstyle\sum_{k=1}^K h_{x^k}(P_k), 0\}$$

and using the identity $\max\{\cdot, 0\} = -\min\{-(\cdot), 0\}$ completes the proof.

Proof of relation (27). To get the relation for gradient norm, again by definition of $h_x(P)$, $f(x^{k+1/2}) - f(x^k) = h_{x^k}(P_k)\|\nabla f(x^k)\|^2$ and

$$f(x^{k+1}) - f(x^k) \leq \min\big\{f(x^{k+1/2}) - f(x^k), f(x^k) - f(x^k)\big\} = \min\{h_{x^k}(P_k), 0\}\|\nabla f(x^k)\|^2.$$

Summing the inequality from $k = 1$ to K, we have

$$f(x^{K+1}) - f(x^1) \leq \textstyle\sum_{k=1}^K \min\{h_{x^k}(P_k), 0\}\|\nabla f(x^k)\|^2.$$

Re-arrangement gives

$$\begin{aligned} & (\textstyle\sum_{k=1}^K \max\{-h_{x^k}(P_k), 0\}) \cdot \min_{1\leq k\leq K} \|\nabla f(x^k)\|^2 \\ \leq{} & \textstyle\sum_{k=1}^K \max\{-h_{x^k}(P_k), 0\}\|\nabla f(x^k)\|^2 \\ \leq{} & f(x^1) - f(x^{K+1}) \\ \leq{} & f(x^1) - f(x^\star) \end{aligned}$$

Last, using convexity of $\max\{\cdot,0\}$,

$$\begin{aligned}\min_{1\le k\le K}\|\nabla f(x^k)\|^2 \le{}& \tfrac{f(x^1)-f(x^\star)}{K}\tfrac{1}{\frac{1}{K}\sum_{k=1}^K \max\{-h_{x^k}(P_k),0\}}\\ \le{}& \tfrac{f(x^1)-f(x^\star)}{K}\tfrac{1}{\max\{\frac{1}{K}\sum_{k=1}^K -h_{x^k}(P_k),0\}}\end{aligned}$$

and this completes the proof for the gradient norm.

To get the relation for function value gap, take $x^\star$ to be the equilibrium of the inner problem $\max_{x\in\mathcal{L}_{f(x^1)}}\min_{x^\star\in\mathcal{X}^\star}\|x-x^\star\|$, we deduce that

$$\begin{aligned}f(x^{k+1})-f(x^\star) \le{}& f(x^k)-f(x^\star)+\min\{h_{x^k}(P_k),0\}\|\nabla f(x^k)\|^2\\ ={}& f(x^k)-f(x^\star)+\min\{h_{x^k}(P_k),0\}\tfrac{\|\nabla f(x^k)\|^2\|x^k-x^\star\|^2}{(f(x^k)-f(x^\star))^2}\tfrac{[f(x^k)-f(x^\star)]^2}{\|x^k-x^\star\|^2}\\ \le{}& f(x^k)-f(x^\star)+\min\{h_{x^k}(P_k),0\}\tfrac{[f(x^k)-f(x^\star)]^2}{\|x^k-x^\star\|^2}, \qquad (71)\end{aligned}$$

where the last inequality uses $f(x^k)-f(x^\star)\le\|\nabla f(x^k)\|\|x^k-x^\star\|$ and that

$$\min\{h_{x^k}(P_k),0\}\cdot\tfrac{\|\nabla f(x^k)\|^2\|x^k-x^\star\|^2}{[f(x^k)-f(x^\star)]^2}\tfrac{[f(x^k)-f(x^\star)]^2}{\|x^k-x^\star\|^2}\le\min\{h_{x^k}(P_k),0\}\cdot\tfrac{[f(x^k)-f(x^\star)]^2}{\|x^k-x^\star\|^2}$$

since $\min\{h_{x^k}(P_k),0\}\le 0$. Re-arranging the terms, we get

$$\begin{aligned}\tfrac{1}{f(x^{k+1})-f(x^\star)}-\tfrac{1}{f(x^k)-f(x^\star)} ={}& \tfrac{f(x^k)-f(x^\star)-[f(x^{k+1})-f(x^\star)]}{[f(x^{k+1})-f(x^\star)][f(x^k)-f(x^\star)]}\\ \ge{}& \tfrac{-\min\{h_{x^k}(P_k),0\}\frac{[f(x^k)-f(x^\star)]^2}{\|x^k-x^\star\|^2}}{[f(x^{k+1})-f(x^\star)][f(x^k)-f(x^\star)]} \qquad (72)\\ ={}& \tfrac{-\min\{h_{x^k}(P_k),0\}[f(x^k)-f(x^\star)]}{[f(x^{k+1})-f(x^\star)]\|x^k-x^\star\|^2}\\ \ge{}& \tfrac{-\min\{h_{x^k}(P_k),0\}}{\|x^k-x^\star\|^2}\ge-\tfrac{1}{\Delta^2}\min\{h_{x^k}(P_k),0\}, \qquad (73)\end{aligned}$$

where (72) plugs in (71); (73) uses the fact that $f(x^k)\le f(x^1)$ and that

$$\|x^k-x^\star\|\le\max_{x\in\mathcal{L}_{f(x^1)}}\|x-x^\star\|=\Delta.$$

Finally, we telescope the relation

$$\tfrac{1}{f(x^{k+1})-f(x^\star)}-\tfrac{1}{f(x^k)-f(x^\star)}\ge-\tfrac{1}{\Delta^2}\min\{h_{x^k}(P_k),0\}$$

from $k=1$ to K and get

$$\begin{aligned}\tfrac{1}{f(x^{K+1})-f(x^\star)}-\tfrac{1}{f(x^1)-f(x^\star)} ={}& \textstyle\sum_{k=1}^K\tfrac{1}{f(x^{k+1})-f(x^\star)}-\tfrac{1}{f(x^k)-f(x^\star)}\\ \ge{}& -\tfrac{1}{\Delta^2}\textstyle\sum_{k=1}^K\min\{h_{x^k}(P_k),0\}\\ ={}& \tfrac{1}{\Delta^2}\textstyle\sum_{k=1}^K\max\{-h_{x^k}(P_k),0\}.\end{aligned}$$

Re-arranging the terms and using convexity of $\max\{\cdot,0\}$,

$$f(x^{K+1})-f(x^\star)\le\tfrac{\Delta^2}{\sum_{k=1}^K\max\{-h_{x^k}(P_k),0\}}\le\tfrac{\Delta^2}{K}\tfrac{1}{\max\{-\frac{1}{K}\sum_{k=1}^K h_{x^k}(P_k),0\}}$$

and this completes the proof.

G.2. Proof of Proposition 4

To show the Lipschitzness of h_x, it suffices to show the gradient is bounded. Given $\nabla h_x(P) = \frac{\nabla f(x - P\nabla f(x))\nabla f(x)^\top}{\|\nabla f(x)\|^2}$, we deduce that

$$\begin{aligned}
\|\nabla h_x(P)\|_F ={}& \tfrac{\|\nabla f(x-P\nabla f(x))\nabla f(x)^\top\|_F}{\|\nabla f(x)\|^2} \\
={}& \tfrac{\|\nabla f(x-P\nabla f(x))\|}{\|\nabla f(x)\|} && (74)\\
\leq{}& \tfrac{\|\nabla f(x-P\nabla f(x))-\nabla f(x)\|+\|\nabla f(x)\|}{\|\nabla f(x)\|} && (75)\\
\leq{}& \tfrac{L\|P\nabla f(x)\|}{\|\nabla f(x)\|} + 1 && (76)\\
\leq{}& L\|P\| + 1 \leq LD + 1,
\end{aligned}$$

where (74) uses $\|ab^\top\|_F = \|a\| \cdot \|b\|$ and (76) applies L-Lispschitzness of $\nabla f(x)$.

G.3. Proof of Lemma 8

The proof is again a direct application of the results in online convex optimization. For any $P \in \mathcal{P}$, (36) gives

$$\textstyle\sum_{k=1}^K h_{x^k}(P_k) - \sum_{k=1}^K h_{x^k}(P) \leq \frac{1}{2\eta}\|P_1 - P\|_F^2 + \frac{\eta}{2}\sum_{k=1}^K \|\nabla h_{x^k}(P_k)\|_F^2 \leq \frac{2D^2}{\eta} + \frac{\eta(LD+1)^2}{2}K,$$

where the last inequality $\|P_1 - P\|_F \leq \|P_1\|_F + \|P\|_F \leq 2D$ and the bounded gradient $\|\nabla h_{x^k}(P)\|_F \leq L(D+1)$. Taking η to minimize the right-hand side completes the proof.

G.4. Proof of Theorem 6

By Lemma 8, we have

$$\tfrac{1}{K}\textstyle\sum_{k=1}^K h_{x^k}(P_k) \leq \frac{1}{K}\sum_{k=1}^K h_{x^k}(P) + \frac{\rho_K}{K}$$

for all $P \in \mathcal{P}$ and plugging $-\frac{1}{K}\sum_{k=1}^K h_{x^k}(P_k) \geq -\theta_P^\star - \frac{\rho_K}{K}$ into Lemma 7 completes the proof.

G.5. Proof of Lemma 9

According to A3, $L^{-1}I \in \mathcal{P}$ and descent lemma gives, for all $x \notin \mathcal{X}^\star$, that

$$h_x(L^{-1}I) = \tfrac{f(x - \frac{1}{L}\nabla f(x)) - f(x)}{\|\nabla f(x)\|^2} \leq -\tfrac{1}{2L}$$

and this completes the proof.

G.6. Proof of Corollary 3

Using Lemma 9 and Theorem 6, $\theta_K^\star \leq -\gamma^\star$ and plugging the bound back into Theorem 6 completes the proof.