Adaptive First-Order Methods Revisited: Convex Optimization without Lipschitz Requirements

Abstract

We propose a new family of adaptive first-order methods for a class of convex minimization problems that may fail to be Lipschitz continuous or smooth in the standard sense. Specifically, motivated by a recent flurry of activity on non-Lipschitz (NoLips) optimization, we consider problems that are continuous or smooth relative to a reference Bregman function – as opposed to a global, ambient norm (Euclidean or otherwise). These conditions encompass a wide range of problems with singular objective, such as Fisher markets, Poisson tomography, D-design, and the like. In this setting, the application of existing order-optimal adaptive methods – like UNIXGRAD or ACCELEGRAD – is not possible, especially in the presence of randomness and uncertainty. The proposed method, *adaptive mirror descent* (ADAMIR), aims to close this gap by concurrently achieving min-max optimal rates in problems that are relatively continuous or smooth, including stochastic ones.

1 Introduction

Owing to their wide applicability and flexibility, first-order methods continue to occupy the forefront of research in learning theory and continuous optimization. Their analysis typically revolves around two basic regularity conditions for the problem at hand: (i) Lipschitz continuity of the problem's objective, and / or (ii) Lipschitz continuity of its gradient (also referred to as *Lipschitz smoothness*). Depending on which of these conditions holds, the lower bounds for first-order methods with perfect gradient input are $\Theta(1/\sqrt{T})$ and $\Theta(1/T^2)$ after T gradient queries, and they are achieved by gradient descent and Nesterov's fast gradient algorithm respectively [31, 32, 43]. By contrast, if the optimizer only has access to stochastic gradients (as is often the case in machine learning and distributed control), the corresponding lower bound is $\Theta(1/\sqrt{T})$ for both classes [11, 29, 32].

This disparity in convergence rates has led to a surge of interest in adaptive methods that can seamlessly interpolate between these different regimes. Two state-of-the-art methods of this type are the ACCELEGRAD and UNIXGRAD algorithms of Levy et al. [23] and Kavis et al. [22]: both algorithms simultaneously achieve an $\mathcal{O}(1/\sqrt{T})$ value convergence rate in non-smooth problems, an $\mathcal{O}(1/T^2)$ rate in smooth problems, and an $\mathcal{O}(1/\sqrt{T})$ average rate if run with bounded, unbiased stochastic gradients (the smoothness does not affect the rate in this case). In this regard, UNIXGRAD and ACCELEGRAD both achieve a "best of all worlds" guarantee which makes them ideal as off-the-shelf solutions for applications where the problem class is not known in advance – e.g., as in online traffic routing, game theory, etc.

At the same time, there have been considerable efforts in the literature to account for problems that do not adhere to these Lipschitz regularity requirements – such as quantum process tomography and ptychography, D-design, Poisson deconvolution / inverse problems, and many other examples [4, 7, 8, 26, 27, 41]. The reason that the Lipschitz framework fails in this case is that, even when the problem's domain is bounded, the objective function exhibits singularities at the boundary, so it cannot be Lipschitz continuous or smooth. As a result, no matter how small we pick the step-size of a standard gradient method (adaptive or otherwise), the existence of domains with singular gradients can – and *does* – lead to catastrophic oscillations (especially in the stochastic case).

Submitted to 35th Conference on Neural Information Processing Systems (NeurIPS 2021). Do not distribute.

A recent breakthrough in this area was provided by Birnbaum et al. [7] and, independently Bauschke et al. [4] and Lu et al. [27], who authors considered a "Lipschitz-like" gradient continuity condition for problems with singularities.[1] At around the same time, Lu [26] and Teboulle [41] introduced a "relative continuity" condition which plays the same role for the Lipschitz continuity of the objective. Collectively, instead of using a global norm as a metric, these conditions employ a Bregman divergence as a – possibly asymmetric – measure of distance, and they replace gradient descent with *mirror descent* [11, 29].

In these extended problem classes, *non-adaptive* mirror descent methods achieve an $\mathcal{O}(1/\sqrt{T})$ value convergence rate in relatively continuous problems [1, 26], an $\mathcal{O}(1/T)$ rate in relatively smooth problems [4, 7], and an $\mathcal{O}(1/\sqrt{T})$ average rate if run with stochastic gradients in relatively continuous problems [1, 26]. Importantly, the $\mathcal{O}(1/T)$ rate for relatively smooth problems *does not match* the $\mathcal{O}(1/T^2)$ rate for standard Lipschitz smooth problems: in fact, even though [18] proposed a tentative path towards faster convergence in certain non-Lipschitz problems, Dragomir et al. [15] recently established an $\Omega(1/T)$ lower bound for problems that are relatively-but-not-Lipschitz smooth.

Our contributions. Our aim in this paper is to provide an *adaptive, parameter-agnostic* method that simultaneously achieves order-optimal rates in the above "non-Lipschitz" framework. By design, the proposed method – which we call *adaptive mirror descent* (AdaMir) – has the following desirable properties:

1. When run with perfect gradients, the trajectory of queried points converges, and the method's rate of convergence in terms of function values interpolates between $\mathcal{O}(1/\sqrt{T})$ and $\mathcal{O}(1/T)$ for relatively continuous and relatively smooth problems respectively.
2. When run with stochastic gradients, the method attains an $\mathcal{O}(1/\sqrt{T})$ average rate of convergence.
3. The method applies to both constrained and unconstrained problems, without requiring a finite Bregman diameter or knowledge of a compact domain containing a solution.

The enabling apparatus for these properties is an adaptive step-size policy in the spirit of [3, 16, 28]. However, a major difference – and technical difficulty – is that the relevant definitions cannot be stated in terms of global norms, because the variation of non-Lipschitz function explodes (put differently, gradients may be unbounded even over bounded domains). For this reason, our policy relies on the aggregation of a suitable sequence of "Bregman residuals" that stabilizes seamlessly when approaching a smooth solution, thus enabling the method to achieve faster convergence rates.

Related work. Beyond the references cited above, problems with singular objectives were treated in a recent series of papers in the context of online and stochastic optimization [1, 45]; however, the proposed methods are not adaptive, and do not interpolate between different problem classes.

In the context of adaptive methods, the widely used AdaGrad algorithm of Duchi et al. [16] and McMahan and Streeter [28] was recently shown to interpolate between an $\mathcal{O}(1/\sqrt{T})$ and $\mathcal{O}(1/T)$ rate of convergence [23, 24]. More precisely, Li and Orabona [24] showed that a specific, "one-lag-behind" variant of AdaGrad with prior knowledge of the smoothness modulus achieves an $\mathcal{O}(1/T)$ rate in smooth, unconstrained problems; concurrently, Levy et al. [23] obtained the same rate in a parameter-agnostic context. In both cases, AdaGrad achieves an $\mathcal{O}(1/\sqrt{T})$ rate of convergence in stochastic problems (though with somewhat different assumptions for the randomness).

In terms of rate optimality for Lipschitz smooth problems, AdaGrad is outperformed by AcceleGrad [23] and UnixGrad [22]: these methods both achieve an $\mathcal{O}(1/T^2)$ rate in Lipschitz smooth problems, and an $\mathcal{O}(1/\sqrt{T})$ in stochastic problems with bounded gradient noise. By employing an efficient line-search step, the *universal primal gradient descent* (UPGD) algorithm of Nesterov [34] achieves order-optimal guarantees in the wider class of Hölder continuous problems (which includes the Lipschitz continuous and smooth cases as extreme cases); however, UPGD does not cover stochastic problems or problems with relatively continuous / smooth objectives.

[1] This condition was first examined by Birnbaum et al. [7] in the specific context of Fisher markets. The analysis of Bauschke et al. [4] and Lu et al. [27] is much more general, but many ideas are already present in [7].

		Constr. / Uncon.	Stoch. (L)	Order-optimal	RC	RS	Stoch. (R)
AdaGrad	[16]	✓/✓	✓	×	×	×	×
AcceleGrad	[23]	×/ ✓	✓	✓	×	×	×
UnixGrad	[22]	✓ / ×	✓	✓	×	×	×
UPGD	[34]	✓ / ✓	×	✓	×	×	×
GMP	[40]	✓ / ✓	×	partial	×	$1/T$	×
AdaProx	[2]	✓ / ✓	×	✓	partial	partial	×
AdaMir	[ours]	✓ / ✓	✓	✓	$1/\sqrt{T}$	$1/T$	$1/\sqrt{T}$

Table 1: Overview of related work. For the purposes of this table, (L) refers to "Lipschitz" and (R) to "relative" continuity or smoothness respectively. "Order-optimal" means that the algorithm attains the best rates for the worst instance in the class it was designed for (see cited papers for the details). Logarithmic factors are ignored throughout; we also note that the $\mathcal{O}(1/T)$ rate in the RS column is, in general, unimprovable [15].

As far as we are aware, the closest works to our own are the *generalized mirror-prox* (GMP) algorithm of [40] and the AdaProx method of [1], both designed for variational inequality problems. The GMP algorithm can achieve interpolation between different classes of Hölder continuous problems and can adapt to the problem's relative smoothness modulus, but it does not otherwise interpolate between the relatively smooth and relatively continuous classes. Moreover, GMP requires knowledge of a "domain of interest" containing a solution of the problem; in this regard, it is similar to AcceleGrad [23] (though it does not require an extra projection step). The recently proposed AdaProx method of Antonakopoulos et al. [2] also achieves a similar interpolation result under a set of assumptions that are closely related – but not equivalent – to the relatively continuous/smooth setting of our paper. In any case, neither of these papers covers the stochastic case; to the best of our knowledge, AdaMir is the first method in the literature capable of adaptiving to relatively continuous / smooth objectives, even under uncertainty. For convenience, we detail these related works in Table 1 above.

2 Problem setup and preliminaries

Problem statement. Throughout the sequel, we will focus on convex minimization problems of the general form

$$\begin{aligned} \text{minimize} \quad & f(x), \\ \text{subject to} \quad & x \in \mathcal{X}. \end{aligned} \tag{Opt}$$

In the above, $\mathcal{X}$ is a convex subset of a normed d-dimensional space $\mathcal{V} \cong \mathbb{R}^d$, and $f \colon \mathcal{V} \to \mathbb{R} \cup \{\infty\}$ is a proper lower semi-continuous (l.s.c.) convex function with $\operatorname{dom} f = \{x \in \mathcal{V} : f(x) < \infty\} = \mathcal{X}$. Compared to standard formulations, we stress that $\mathcal{X}$ *is not assumed to be compact, bounded, or even closed*. This lack of closedness will be an important feature for our analysis because we are interested in objectives that may develop singularities at the boundary; for a class of relevant examples of this type, see [1, 4, 6–8, 27, 45] and references therein.

To formalize our assumptions for (Opt), we will write $\partial f(x)$ for the *subdifferential* of f at x, and $\mathcal{X}_\circ \equiv \operatorname{dom} \partial f = \{x \in \mathcal{X} : \partial f(x) \neq \varnothing\}$ for the *domain of subdifferentiability* of f. Formally, elements of ∂f will be called subgradients, and we will treat them throughout as elements of the dual space $\mathcal{V}^*$ of $\mathcal{V}$. By standard results, we have $\operatorname{ri} \mathcal{X} \subseteq \mathcal{X}_\circ \subseteq \mathcal{X}$, and any solution x^* of (Opt) belongs to $\mathcal{X}_\circ$ [37, Chap. 26]; to avoid trivialities, we will make the following blanket assumption.

Assumption 1. The solution set $\mathcal{X}^* \equiv \arg\min f \subseteq \mathcal{X}_\circ$ of (Opt) is nonempty.

Two further assumptions that are standard in the literature (but which we relax in the sequel) are:

1. *Lipschitz continuity:* there exists some $G > 0$ such that

$$|f(x') - f(x)| \leq G\|x' - x\| \quad \text{for all } x, x' \in \mathcal{X}. \tag{LC}$$

2. *Lipschitz smoothness:* there exists some $L > 0$ such that

$$f(x') \leq f(x) + \langle v, x' - x \rangle + \frac{L}{2}\|x' - x\|^2 \quad \text{for all } x, x' \in \mathcal{X} \text{ and all } v \in \partial f(x). \tag{LS}$$

Remark. For posterity, we note that the smoothness requirement (LS) *does not* imply that $\partial f(x)$ is a singleton. The reason for this more general definition is that we want to concurrently treat problems with smooth and non-smooth objectives, and also feasible domains that are contained in lower-dimensional subspaces of $\mathcal{V}$.[2] We also note that we will be mainly interested in cases where the above requirements all *fail* because f and/or its derivatives blow up at the boundary of $\mathcal{X}$. By this token, we will not treat (LC)/(LS) as "blanket assumptions"; we discuss this in detail in the sequel.

The oracle model. For an algorithmic point of view, we aim to solve (Opt) by using iterative methods that require access to a *stochastic first-order oracle* (SFO) [32]. This means that, at each stage of the process, the optimizer can query a black-box mechanism that returns an estimate of the objective's gradient (or subgradient) at the queried point. Formally, when called at $x \in \mathcal{X}$, an SFO is assumed to return a random (dual) vector $g(x;\omega) \in \mathcal{V}^*$ where ω belongs to some (complete) probability space $(\Omega, \mathcal{F}, \mathbb{P})$. In practice, the oracle will be called repeatedly at a (possibly) random sequence of points $X_t \in \mathcal{X}$ generated by the algorithm under study. Thus, once X_t has been generated at stage t, the oracle draws an i.i.d. sample sample $\omega_t \in \Omega$ and returns the dual vector:

$$g_t \equiv g(X_t;\omega_t) = \nabla f(X_t) + U_t \tag{1}$$

with $U_t \in \mathcal{V}^*$ denoting the "measurement error" relative to some selection $\nabla f(X_t)$ of $\partial f(X_t)$. In terms of measurability, we will write $\mathcal{F}_t$ for the history (natural filtration) of X_t, so X_t is $\mathcal{F}_t$-adapted, but ω_t, g_t and U_t are not. Finally, we will also make the statistical assumption that

$$\mathbb{E}[U_t \mid \mathcal{F}_t] = 0 \quad \text{and} \quad \|U_t\|_*^2 \leq \sigma^2 \quad \text{for all } t = 1, 2, \ldots \tag{SFO}$$

This assumption is standard in the analysis of parameter-agnostic adaptive methods, cf. [3, 22, 23, 42] and references therein. For concreteness, we will refer to the case $\sigma = 0$ as deterministic – since, in that case, $U_t = 0$ for all t. Otherwise, if $\liminf_t \|U_t\|_* > 0$, the noise will be called *persistent* and the model will be called *stochastic*.

3 Relative Lipschitz continuity and relative Lipschitz smoothness

3.1. Bregman functions. We now proceed to describe a flexible template extending the standard Lipschitz continuity and Lipschitz smoothness conditions – (LC) and (LS) – to functions that are possibly singular at the boundary points of $\mathcal{X}$. The main idea of this extension revolves around the *non-Lipschitz* (NoLips) framework that was first studied by Birnbaum et al. [7] and then rediscovered independently by Bauschke et al. [4] and Lu et al. [27]. The key notion in this setting is that of a suitable "reference" *Bregman function* which provides a geometry-adapted measure of divergence on $\mathcal{X}$. This is defined as follows:

Definition 1. A convex l.s.c. function $h\colon \mathcal{V} \to \mathbb{R} \cup \{\infty\}$ is a *Bregman function* on $\mathcal{X}$, if

1. $\operatorname{dom} \partial h \subseteq \mathcal{X} \subseteq \operatorname{dom} h$.
2. The subdifferential of h admits a continuous selection $\nabla h(x) \in \partial h(x)$ for all $x \in \operatorname{dom} \partial h$.
3. h is strongly convex, i.e., there exists some $K > 0$ such that

$$h(x') \geq h(x) + \langle \nabla h(x), x' - x \rangle + \frac{K}{2}\|x' - x\|^2 \tag{2}$$

for all $x \in \operatorname{dom} \partial h$, $x' \in \operatorname{dom} \partial h$.

The induced *Bregman divergence* of h is then defined for all $x \in \operatorname{dom} \partial h$, $x' \in \operatorname{dom} h$ as

$$D(x', x) = h(x') - h(x) - \langle \nabla h(x), x' - x \rangle. \tag{3}$$

Remark. The notion of a Bregman function was first introduced by Bregman [10]. Our definition follows [20, 30, 33] and leads to the smoothest presentation, but there are variant definitions where h is not necessarily assumed strongly convex, cf. [1, 12, 13] and references therein.

[2] For example, the function $f\colon \mathbb{R}^2 \to \mathbb{R}$ with $f(x_1, 0) = x_1$ and $f(x_1, x_2) = \infty$ for $x_2 \neq 0$ is perfectly smooth on its domain ($x_2 = 0$); however, $\partial f(x_1, 0) = \{(1, v_2) : v_2 \in \mathbb{R}\}$, and this set is never a singleton.

Some standard examples of Bregman functions are as follows:

- **Euclidean regularizer:** Let $\mathcal{X}$ be a convex subset of $\mathbb{R}^d$ endowed with the Euclidean norm $\|\cdot\|_2$. Then, the *Euclidean regularizer* on $\mathcal{X}$ is defined as $h(x) = \|x\|_2^2/2$ and the induced Bregman divergence is the standard square distance $D(x', x) = \|x' - x\|_2^2$ for all $x, x' \in \mathcal{X}$
- **Entropic regularizer:** Let $\mathcal{X} = \{x \in \mathbb{R}_+^d : \sum_{i=1}^d x_i = 1\}$ be the unit simplex of $\mathbb{R}^d$ endowed with the L^1-norm $\|\cdot\|_1$. Then, the *entropic regularizer* on $\mathcal{X}$ is $h(x) = \sum_i x_i \log x_i$ and the induced divergence is the relative entropy $D(x', x) = \sum_i x_i' \log(x_i'/x_i)$ for all $x' \in \mathcal{X}$ $x \in \operatorname{ri} \mathcal{X}$.
- **Log-barrier:** Let $\mathcal{X} = \mathbb{R}_{++}^d$ denote the (open) positive orthant of $\mathbb{R}^d$. Then, the *log-barrier* on $\mathcal{X}$ is defined as $h(x) = -\sum_{i=1}^d \log x_i$ for all $x \in \mathbb{R}_{++}^d$. The corresponding divergence is known as the *Itakura-Saito divergence* and is given by $D(x, x') = \sum_{i=1}^d [-\log(x_i/x_i') + x_i/x_i' - 1]$ [13].

3.2. Relative continuity. With this background in hand, we are in a position to discuss how to extend the Lipschitz regularity assumptions of Section 2 to account for problems with singular objective functions. We begin with the notion of *relative continuity* (RC), as introduced by Lu [26] and extended further in a recent paper by Zhou et al. [45]:

Definition 2. A convex l.s.c. function $f \colon \mathcal{V} \to \mathbb{R} \cup \{\infty\}$ is said to be *relatively continuous* if there exists some $G > 0$ such that

$$f(x) - f(x') \leq \langle \nabla f(x), x - x' \rangle \leq G\sqrt{2D(x', x)} \quad \text{for all } x \in \operatorname{dom} h, x' \in \operatorname{dom} \partial h. \tag{RC}$$

In the literature, there have been several extensions of (LC) to problems with singular objectives. Below we discuss some of them and how they can be integrated in the setting of Definition 2.

► **Example 1** (W$[f, h]$ continuity)**.** This notion intends to single out sufficient conditions for the convergence of "proximal-like" methods like mirror descent. Specifically, following Teboulle [41], f is said to be W$[f, h]$-continuous relative to h on $\mathcal{X}$ (read: "f is weakly h-continuous") if there exists some $G > 0$ such that, for all $t > 0$, we have

$$t\langle \nabla f(x), x - x' \rangle - D(x', x) \leq \frac{t^2}{2} G^2 \quad \text{for all } x' \in \operatorname{dom} h, x \in \operatorname{dom} \partial h. \tag{W}$$

By rearranging the above quadratic polynomial in t, we note that its discriminant is $\Delta = [\langle \nabla f(x), x - x' \rangle]^2 - 2G^2 D(x', x)$, so it is immediate to check that (RC) holds.

► **Example 2** (Riemann–Lipschitz continuity)**.** Concurrently to the above, Antonakopoulos et al. [1] introduced a Riemann-Lipschitz continuity condition extending (LC) as follows. Let $\|\cdot\|_x$ be a family of local norms on $\mathcal{X}$ (possibly induced by an appropriate Riemannian metric), and let $\|v\|_{x,*} = \max_{\|x'\|_x \leq 1} \langle v, x' \rangle$ denote the corresponding dual norm. Then, f is *Riemann–Lipschitz continuous* relative to $\|\cdot\|_x$ if there exists some $G > 0$ such that:

$$\|\nabla f(x)\|_{x,*} \leq G \quad \text{for all } x \in \mathcal{X}. \tag{RLC}$$

As we show in the paper's supplement, (RLC) $\Longrightarrow$ (RC) so (RC) is more general in this regard.

3.3. Relative smoothness. As discussed above, the notion of *relative smoothness* (RS) was introduced by [7] and independently rediscovered by [4, 27]. It is defined as follows:

Definition 3. A convex l.s.c. function $f \colon \mathcal{V} \to \mathbb{R} \cup \{\infty\}$ is said to be *relatively smooth* if there exists some $L > 0$ such that

$$Lh - f \quad \text{is convex.} \tag{RS}$$

The main motivation behind this elegant definition is the following variational characterizations:

Proposition 1. *The following statements are equivalent:*

1. *f satisfies* (RS).
2. *f satisfies the inequality* $f(x) \leq f(x') + \langle \nabla f(x'), x - x' \rangle + LD(x, x')$,
3. *f satisfies the inequality* $\langle \nabla f(x) - \nabla f(x'), x - x' \rangle \leq L\left[D(x, x') + D(x', x)\right]$.

A close variant of Proposition 1 appears in [4, 7, 27], so we do not prove it here. Instead, we discuss below a different extension of (LS) that turns out to be a special case of (RS).

► **Example 3** (Metric smoothness)**.** Similar in spirit to (RLC), Antonakopoulos et al. [2] introduced an extension of (LS) that replaces the global norm $\|\cdot\|$ with a local norm $\|\cdot\|_x$, $x \in \mathcal{X}_\circ$. In particular, given such a norm, we say that f is *metrically smooth* (relative to $\|\cdot\|_x$) if

$$\|\nabla f(x) - \nabla f(x')\|_{x,*} \leq L\|x - x'\|_{x'} \quad \text{for all } x, x' \in \operatorname{dom} \partial f. \tag{MS}$$

An observation that seems to have been overlooked by [2] is that (MS) $\Longrightarrow$ (RS), so (RS) is more general. We prove this observation in the appendix.

3.4. More examples. Some concrete examples of optimization problems satisfying (RC), (RS) or both (but not their Euclidean counterparts) are Fisher markets [7, 39], Poisson inverse problems [1, 4], support vector machines [26, 45], D-design [8, 27], etc. Because of space constraints, we do not detail these examples here; however, we provide an in-depth presentation of a Fisher market model in the appendix, along with a series of numerical experiments used to validate the analysis to come.

4 Adaptive mirror descent

We are now in a position to define the proposed *adaptive mirror descent* (ADAMIR) method. In abstract recursive form, ADAMIR follows the basic mirror descent template

$$x^+ = P_x(-\gamma g), \tag{MD}$$

where P is a generalized Bregman proximal operator induced by h (see below for the detailed definition), g is a search direction determined by a (sub)gradient of f at x, and $\gamma > 0$ is a step-size parameter. We discuss these elements in detail below, starting with the prox-mapping P.

4.1. The prox-mapping. Given a Bregman function h, its induced *prox-mapping* is defined as

$$P_x(v) = \arg\min_{x' \in \mathcal{X}} \{\langle v, x - x'\rangle + D(x', x)\} \quad \text{for all } x \in \operatorname{dom} \partial h,\ v \in \mathcal{V}^*, \tag{4}$$

where $D(x', x)$ denotes the Bregman divergence of h. Of course, in order for (4) to be well-defined, the $\arg\min$ must be attained in $\operatorname{dom} \partial h$. Indeed, we have:

Proposition 2. *The recursion* (MD) *satisfies* $x^+ \in \operatorname{dom} \partial h$ *for all* $x \in \operatorname{dom} \partial h$ *and all* $g \in \mathcal{V}^*$.

To streamline our discussion, we postpone the proof of Proposition 2 to Appendix A. For now, we only note that it implies that the abstract recursion (MD) is *well-posed*, i.e., it can be iterated for $t = 1, 2, \dots$ to generate a sequence $X_t \in \mathcal{X}$.

4.2. The method's step-size. The next important element of (MD) is the method's step-size. In the unconstrained case, a popular adaptive choice is the so-called "inverse-sum-of-squares" policy

$$\gamma_t = 1 \Big/ \sqrt{\textstyle\sum_{s=1}^t \|\nabla f(X_s)\|_*^2}, \tag{5}$$

where X_t is the series of iterates produced by the algorithm. In our case, this definition encounters two crucial issues. First, because the gradient of f is unbounded (even over a bounded domain), the denominator of (5) may grow at an uncontrollable rate, leading to a step-size policy that vanishes too fast to be of any practical use. The second is that, if the problem is constrained, the extra terms entering the denominator of γ_t do not vanish as the algorithm approaches a solution, so the (5) may still be unable to exploit the smoothness of the objective.

We begin by addressing the second issue. In the Euclidean case, the key observation is that the difference $\|x^+ x - x\|$ must always vanish near a solution (even near the boundary), so we can use it as a proxy for $\nabla f(x)$ in constrained problems. This idea is formalized by the notion of the *gradient mapping* [32] that can be defined here as

$$\delta = \|x^+ - x\| / \gamma. \tag{6}$$

On the other hand, in a Bregman setting, the prox-mapping tends to deflate gradient steps, so the norm difference between two successive iterates x^+ and x of (MD) could be very small relative to the oracle signal that was used to generate the update. As a result, the Euclidean residual (6) could lead to a disproportionately large step-size that would be harmful for convergence. For this reason, we consider a gradient mapping that takes into account the Bregman geometry of the method, namely

$$\delta = \sqrt{D(x, x^+) + D(x^+, x)}/\gamma. \tag{7}$$

Obviously, when $h(x) = (1/2)\|x\|_2^2$, we readily recover the definition of the Euclidean gradient mapping (6). In general however, by the strong convexity of h, the value of this "Bregman residual mapping" exceeds the corresponding Euclidean definition, so the induced step-size exhibits smoother variations that are more adapted to the framework in hand.

4.3. The AdaMir algorithm. Putting everything together, we are finally in a position to define the *adaptive mirror descent* (AdaMir) method. More precisely, combining the abstract template (MD) with the Bregman residual and "inverse-sum-of-squares" approach discussed above, we will consider the recursive policy

$$X_{t+1} = P_{X_t}(-\gamma_t g_t) \tag{8}$$

with g_t, $t = 1, 2, \dots$, coming from a generic oracle model of the form (SFO), and with γ_t defined as

$$\gamma_t = \frac{1}{\sqrt{\sum_{s=0}^{t-1} \delta_s^2}} \qquad \text{with } \delta_s^2 = \frac{D(X_s, X_{s+1}) + D(X_{s+1}, X_s)}{\gamma_s^2}. \tag{AdaMir}$$

In the sequel, we will use the term "AdaMir" to refer interchangeably to the update $X_t \leftarrow X_{t+1}$ and the specific step-size policy used within. The convergence properties of AdaMir are discussed in detail in the next two sections (in both deterministic and stochastic problems); in the supplement, we also perform a numerical validation of the method in the context of a Fisher market model.

5 Deterministic analysis and results

We are now in a position to state our main convergence results for AdaMir. We begin with the deterministic analysis ($\sigma = 0$), treating both the method's "averaged" as well as the induced trajectory of query points; the analysis for the stochastic case ($\sigma > 0$) is presented in the next section.

5.1. Ergodic convergence and rate interpolation. We begin with the convergence of the method's "time-averaged" state, i.e., $\bar{X}_T = (1/T) \sum_{t=1}^{T} X_t$.

Theorem 1. *Let X_t, $t = 1, 2, \dots$, denote the sequence of iterates generated by* AdaMir*, and let $D_1 = D(x^*, X_1)$. Then,* AdaMir *enjoys the following simultaneous guarantees:*

1. *If f satisfies* (RC)*, we have:*

$$f(\bar{X}_T) - \min f \leq \frac{\sqrt{2}G\big[D_1 + 8G^2/\delta_0^2 + 2\log(1 + 2G^2T/\delta_0^2)\big]}{\sqrt{T}} + \frac{3\sqrt{2}G + 4G^2/\delta_0^2}{T}. \tag{9}$$

2. *If f satisfies* (RS)*, we have $f(\bar{X}_T) - \min f = \mathcal{O}(D_1/T)$.*

3. *If f satisfies* (RS) *and* (RC)*, we have:*

$$f(\bar{X}_T) - \min f \leq \left[f(X_1) - \min f + \left(2 + \frac{8G^2}{\delta_0^2} + 2\log\frac{4L^2}{\delta_0^2}\right)L\right]^2 \frac{D_1}{T}. \tag{10}$$

Theorem 1 shows that, up to logarithmic factors, AdaMir achieves the min-max optimal bounds for functions in the $RC \cup RS$ oracle complexity class.[3] The key element of the proof (which we detail in Appendix B), is the following regret bound:

[3] We recall here that, in contrast to (LS), the $\mathcal{O}(1/T)$ rate is optimal in (RS), cf. Dragomir et al. [15].

Proposition 3. *With notation as in Theorem 1,* AdaMir *enjoys the regret bound*

$$\sum_{t=1}^{T}[f(X_t) - f(x^*)] \leq \frac{D_1}{\gamma_T} + \frac{\sum_{t=1}^{T} \gamma_t^2 \delta_t^2}{\gamma_T} + \sum_{t=1}^{T} \gamma_t \delta_t^2. \tag{11}$$

The proof of Proposition 3 hinges on the specific definition of AdaMir's step-size, and the exact functional form of the regret bound (11) plays a crucial role in the sequel. Specifically, under the regularity conditions (RC) and (RS), we respectively obtain the following key lemmas:

Lemma 1. *Under* (RC)*, the sequence of the Bregman residuals* δ_t *of* AdaMir *is bounded as* $\delta_t^2 \leq 2G^2$ *for all* $t \geq 1$.

Lemma 2. *Under* (RS)*, the sequence of the Bregman residuals* δ_t *of* AdaMir *is square-summable, i.e.,* $\sum_t \delta_t^2 < \infty$*. Consequently, the method's step-size converges to a strictly positive limit* $\gamma_\infty > 0$.

As we explain below, the boundedness estimate of Lemma 1 is necessary to show that the iterates of the method do not explode; however, without further assumptions, it is not possible to sharpen this bound. The principal technical difficulty – and an important novelty of our analysis – is the stabilization of the step-size to a strictly positive limit in Lemma 2. This property of AdaMir plays a crucial role because the method is not slowed down near a solution. To the best of our knowledge, there is no comparable result for the step-size of parameter-agnostic methods in the literature.[4]

Armed with these two lemmas, we obtain the following series of estimates:

1. Under (RC), the terms in the RHS of (11) can be bounded respectively as $\mathcal{O}(G\sqrt{T})$, $\mathcal{O}(\log(G^2T)\sqrt{T})$, and $\mathcal{O}(G\sqrt{T})$. As a result, we obtain an $\tilde{\mathcal{O}}(1/\sqrt{T})$ rate of convergence.
2. Under (RS), all the right-hand side (RHS) terms in (11) can be bounded as $\mathcal{O}(1)$, so we obtain an $\mathcal{O}(1/T)$ convergence rate for $\bar{X}_T$.

For the details of these calculations (including the explicit constants and logarithmic terms that appear in the statement of Theorem 1), we refer the reader to Appendix B.

5.2. Other modes of convergence. In complement to the analysis above, we provide below a spinoff result for the method's "last iterate", i.e., the actual trajectory of queried points. The formal statement is as follows.

Theorem 2. *Suppose that* f *satisfies* (RC) *or* (RS)*. Then* X_t *converges to* $\arg\min f$.

The main idea of the proof (which we detail in the appendix) consists of two steps. The first key step is to show that, under (RC) $\cup$ (RS), the iterates of AdaMir have $\liminf f(X_t) = \min f$; we show this in Proposition C.1. Given the existence of a convergent subsequence, the rest of our proof strategy branches out depending on whether f satisfies (RC) or (RS). Under (RS), the analysis relies on arguments that involve a quasi-Fejér argument as in [9, 14]. However, under (RC), the quasi-Fejér property fails, so we prove the convergence of X_t via a novel induction argument that shows that the method's iterates remain trapped within a Bregman neighborhood of x^* if they enter it with a sufficiently small step-size; we provide the relevant details in Appendix C.

Non-convex objectives. We close this section with two remarks on non-convex objectives. First, Theorem 2 applies verbatim to non-convex objectives f satisfying the "secant condition" [9, 47]

$$\inf\{\langle \nabla f(x), x - x^* \rangle : x^* \in \arg\min f, x \in \mathcal{K}\} > 0 \tag{12}$$

for every closed subset $\mathcal{K}$ of $\mathcal{X}$ that is separated by neighborhoods from $\arg\min f$. In Appendix C, our results have all been derived based on this more general condition (it is straightforward to verify that (SI) always holds for convex functions).

[4] In more detail, Levy et al. [23], Li and Orabona [24] and Kavis et al. [22] establish the summability of a suitable residual sequence to sharpen the $\mathcal{O}(1/\sqrt{T})$ rate in their respective contexts, but this does not translate to a step-size stabilization result. Under (RC)/(RS), controlling the method's step-size is of vital importance because the gradients that enter the algorithm may be unbounded even over a bounded domain; this crucial difficulty does not arise in any of the previous works on adaptive methods for ordinary Lipschitz problems.

Even more generally, Lemma 2 also allows us to derive results for general non-convex problems. Indeed, the proof of Proposition 1 shows that $\min_{1\leq t\leq T} \delta_t^2 = \mathcal{O}(1/T)$ *without* requiring any properties on f other than relative smoothness. As a result, we conclude that the "best iterate" of the method – i.e., the iterate with the least residual – decays as $\mathcal{O}(1/T)$. This fact partially generalizes a similar result obtained in [24, 42] for AdaGrad applied to non-convex problems; however, an in-depth discussion of this property would take us too far afield, so we do not attempt it.

6 Stochastic analysis

In this last section, we focus on the stochastic case ($\sigma > 0$). Our main results here are as follows.

Theorem 3. *Let X_t, $t = 1, 2, \dots$, denote the sequence of iterates generated by* AdaMir*, and let $D_1 = D(x^*, X_1)$, $G_\sigma = G + \sigma/\sqrt{K}$. Then, under (RC), we have*

$$\mathbb{E}\left[f(\bar{X}_T) - f(x^*)\right] \leq (D_1 + H)\sqrt{\frac{\delta_0^2 + 2G_\sigma^2}{T}} \tag{13}$$

where $H = 8G_\sigma^2/\delta_0^2 + 2\log(1 + 2G_\sigma^2 T/\delta_0^2)$.

Finally, if (RS) kicks in, we have the sharper guarantee:

Theorem 4. *With notation as above, if f satisfies (RS),* AdaMir *enjoys the bound*

$$\mathbb{E}[f(\bar{X}_T) - f(x^*)] \leq (2 + D_1 + H)\left[\frac{A}{T} + \frac{B\sigma}{\sqrt{T}}\right] \tag{14}$$

where:

a) $A = \delta_0 + 2[f(X_1) - \min f] + L\big(2 + 8G_\sigma^2/\delta_0^2 + 2\log(4L^2/\delta_0^2)\big)$. (15a)

b) $B = \sqrt{(4 + 2H)/K}$. (15b)

The full proof of Theorems 3 and 4 is relegated to the supplement, but the key steps are as follows:

Step 1: We first show that, under (RC), the method's residuals are bounded $\delta_t^2 \leq 2G_\sigma^2$ (a.s.).

Step 2: With this at hand, the workhorse for our analysis is the following boxing bound for the mean "weighted" regret $\sum_{t=1}^T \mathbb{E}[\gamma_t \langle \nabla f(X_t), X_t - x^* \rangle]$:

$$\mathbb{E}\left[\gamma_T \sum_{t=1}^T [f(X_t) - f(x^*)]\right] \leq \mathbb{E}\left[\sum_{t=1}^T \gamma_t \langle \nabla f(X_t), X_t - x^* \rangle\right] \leq D_1 + \mathbb{E}\left[\sum_{t=1}^T \gamma_t^2 \delta_t^2\right]$$

We prove this bound in the supplement, where we also show that $\mathbb{E}[\sum_{t=1}^T \gamma_t^2 \delta_t^2] = \mathcal{O}(\log T)$.

At this point the analysis between Theorems 3 and 4 branches out. First, in the case of Theorem 3, we show in the supplement that the method's step-size is bounded from below as $\gamma_t \geq 1/\sqrt{(\delta_0^2 + 2G_\sigma^2)t}$; the guarantee (13) then follows by the boxing bound. Instead, in the case of Theorem 4, the analysis is more involved, and relies on the lower bound $\gamma_t \geq 1/(A + B\sigma\sqrt{t})$. The bound (14) then follows by combining this lower bound for γ_t with the regret boxing bound above.

In the supplement, we also conclude a series of numerical experiments in random Fisher markets that illustrate the method's appealing adaptation properties in an archetypal non-Lipschitz problem.

7 Concluding remarks

Our theoretical analysis confirms that AdaMir concurrently achieves optimal rates of convergence in relatively continuous and relatively smooth problems, both stochastic or deterministic, constrained or unconstrained, and without requiring any prior knowledge of the problem's parameters. These appealing properties open the door to several future research directions, especially regarding the method's convergence properties in non-convex problems. Our "best-iterate" analysis that we hint at in Section 5 is a first step along the eay, but many questions and problems remain open in this direction, especially regarding the convergence of the method's "last iterate" in stochastic, non-convex settings. We defer these questions to future work.

References

[1] Kimon Antonakopoulos, E. Veronica Belmega, and Panayotis Mertikopoulos. Online and stochastic optimization beyond Lipschitz continuity: A Riemannian approach. In *ICLR '20: Proceedings of the 2020 International Conference on Learning Representations*, 2020.

[2] Kimon Antonakopoulos, E. Veronica Belmega, and Panayotis Mertikopoulos. Adaptive extra-gradient methods for min-max optimization and games. In *ICLR '21: Proceedings of the 2021 International Conference on Learning Representations*, 2021.

[3] Francis Bach and Kfir Yehuda Levy. A universal algorithm for variational inequalities adaptive to smoothness and noise. In *COLT '19: Proceedings of the 32nd Annual Conference on Learning Theory*, 2019.

[4] Heinz H. Bauschke, Jérôme Bolte, and Marc Teboulle. A descent lemma beyond Lipschitz gradient continuity: First-order methods revisited and applications. *Mathematics of Operations Research*, 42(2): 330–348, May 2017.

[5] Amir Beck and Marc Teboulle. Mirror descent and nonlinear projected subgradient methods for convex optimization. *Operations Research Letters*, 31(3):167–175, 2003.

[6] Mario Bertero, Patrizia Boccacci, Gabriele Desiderà, and Giuseppe Vicidomini. Image deblurring with Poisson data: from cells to galaxies. *Inverse Problems*, 25(12):123006, November 2009.

[7] Benjamin Birnbaum, Nikhil R. Devanur, and Lin Xiao. Distributed algorithms via gradient descent for Fisher markets. In *EC' 11: Proceedings of the 12th ACM Conference on Electronic Commerce*, 2011.

[8] Jérôme Bolte, Shoham Sabach, Marc Teboulle, and Yakov Vaisbourd. First order methods beyond convexity and Lipschitz gradient continuity with applications to quadratic inverse problems. *SIAM Journal on Optimization*, 28(3):2131–2151, 2018.

[9] Léon Bottou. Online learning and stochastic approximations. *On-line learning in neural networks*, 17(9): 142, 1998.

[10] Lev M. Bregman. The relaxation method of finding the common point of convex sets and its application to the solution of problems in convex programming. *USSR Computational Mathematics and Mathematical Physics*, 7(3):200–217, 1967.

[11] Sébastien Bubeck. Convex optimization: Algorithms and complexity. *Foundations and Trends in Machine Learning*, 8(3-4):231–358, 2015.

[12] Yair Censor and Arnold Lent. An iterative row action method for internal convex programming. *Journal of Optimization Theory and Applications*, 34:321–353, 1981.

[13] Gong Chen and Marc Teboulle. Convergence analysis of a proximal-like minimization algorithm using Bregman functions. *SIAM Journal on Optimization*, 3(3):538–543, August 1993.

[14] Patrick L. Combettes. Quasi-Fejérian analysis of some optimization algorithms. In Dan Butnariu, Yair Censor, and Simeon Reich, editors, *Inherently Parallel Algorithms in Feasibility and Optimization and Their Applications*, pages 115–152. Elsevier, New York, NY, USA, 2001.

[15] Radu-Alexandru Dragomir, Adrien B. Taylor, Alexandre d' Aspremont, and Jérôme Bolte. Optimal complexity and certification of Bregman first-order merthods. https://arxiv.org/pdf/1911.08510.pdf, November 2019.

[16] John Duchi, Elad Hazan, and Yoram Singer. Adaptive subgradient methods for online learning and stochastic optimization. *The Journal of Machine Learning Research*, 12:2121–2159, 2011.

[17] Francisco Facchinei and Jong-Shi Pang. *Finite-Dimensional Variational Inequalities and Complementarity Problems*. Springer Series in Operations Research. Springer, 2003.

[18] Filip Hanzely, Peter Richtárik, and Lin Xiao. Accelerated Bregman proximal gradient methods for relatively smooth convex optimization. https://arxiv.org/abs/1808.03045, 2018.

[19] Houyuan Jiang and Huifu Xu. Stochastic approximation approaches to the stochastic variational inequality problem. *IEEE Trans. Autom. Control*, 53(6):1462–1475, July 2008.

[20] Anatoli Juditsky, Arkadi Semen Nemirovski, and Claire Tauvel. Solving variational inequalities with stochastic mirror-prox algorithm. *Stochastic Systems*, 1(1):17–58, 2011.

[21] Hamed Karimi, Julie Nutini, and Mark Schmidt. Linear convergence of gradient and proximal-gradient methods under the Polyak-Łojasiewicz condition. https://arxiv.org/abs/1608.04636, 2016.

[22] Ali Kavis, Kfir Yehuda Levy, Francis Bach, and Volkan Cevher. UnixGrad: A universal, adaptive algorithm with optimal guarantees for constrained optimization. In *NeurIPS '19: Proceedings of the 33rd International Conference on Neural Information Processing Systems*, 2019.

[23] Kfir Yehuda Levy, Alp Yurtsever, and Volkan Cevher. Online adaptive methods, universality and acceleration. In *NeurIPS '18: Proceedings of the 32nd International Conference of Neural Information Processing Systems*, 2018.

[24] Xiaoyu Li and Francesco Orabona. On the convergence of stochastic gradient descent with adaptive stepsizes. In *AISTATS '19: Proceedings of the 22nd International Conference on Artificial Intelligence and Statistics*, 2019.

[25] Lennart Ljung. Strong convergence of a stochastic approximation algorithm. *Annals of Statistics*, 6(3): 680–696, 1978.

[26] Haihao Lu. "Relative-continuity" for non-Lipschitz non-smooth convex optimization using stochastic (or deterministic) mirror descent. *INFORMS Journal on Optimization*, 1(4):288–303, June 2019.

[27] Haihao Lu, Robert M. Freund, and Yurii Nesterov. Relatively-smooth convex optimization by first-order methods and applications. *SIAM Journal on Optimization*, 28(1):333–354, 2018.

[28] H. Brendan McMahan and Matthew Streeter. Adaptive bound optimization for online convex optimization. In *COLT '10: Proceedings of the 23rd Annual Conference on Learning Theory*, 2010.

[29] Arkadi Semen Nemirovski and David Berkovich Yudin. *Problem Complexity and Method Efficiency in Optimization*. Wiley, New York, NY, 1983.

[30] Arkadi Semen Nemirovski, Anatoli Juditsky, Guanghui Lan, and Alexander Shapiro. Robust stochastic approximation approach to stochastic programming. *SIAM Journal on Optimization*, 19(4):1574–1609, 2009.

[31] Yurii Nesterov. A method for unconstrained convex minimization problem with the rate of convergence $\mathcal{O}(1/k^2)$. *Proceedings of the USSR Academy of Sciences*, 269(543-547), 1983.

[32] Yurii Nesterov. *Introductory Lectures on Convex Optimization: A Basic Course*. Number 87 in Applied Optimization. Kluwer Academic Publishers, 2004.

[33] Yurii Nesterov. Primal-dual subgradient methods for convex problems. *Mathematical Programming*, 120 (1):221–259, 2009.

[34] Yurii Nesterov. Universal gradient methods for convex optimization problems. *Mathematical Programming*, 152(1-2):381–404, 2015.

[35] M. B. Nevel'son and Rafail Z. Khasminskii. *Stochastic Approximation and Recursive Estimation*. American Mathematical Society, Providence, RI, 1976.

[36] Boris Teodorovich Polyak. *Introduction to Optimization*. Optimization Software, New York, NY, USA, 1987.

[37] Ralph Tyrrell Rockafellar. *Convex Analysis*. Princeton University Press, Princeton, NJ, 1970.

[38] Shai Shalev-Shwartz. Online learning and online convex optimization. *Foundations and Trends in Machine Learning*, 4(2):107–194, 2011.

[39] Vadim Ivanovich Shmyrev. An algorithm for finding equilibrium in the linear exchange model with fixed budgets. *Journal of Applied and Industrial Mathematics*, 3:505–518, 2009.

[40] Fedor Stonyakin, Alexander Gasnikov, Alexander Tyurin, Dmitry Pasechnyuk, Artem Agafonov, Pavel Dvurechensky, Darina Dvinskikh, Alexey Kroshnin, and Victorya Piskunova. Inexact model: A framework for optimization and variational inequalities. https://arxiv.org/abs/1902.00990, 2019.

[41] Marc Teboulle. A simplified view of first order methods for optimization. *Mathematical Programming*, 170:67–96, 2018.

[42] Rachel Ward, Xiaoxia Wu, and Léon Bottou. AdaGrad stepsizes: Sharp convergence over nonconvex landscapes, from any initialization. In *ICML '19: Proceedings of the 36th International Conference on Machine Learning*, 2019.

[43] Lin Xiao. Dual averaging methods for regularized stochastic learning and online optimization. *Journal of Machine Learning Research*, 11:2543–2596, October 2010.

[44] Hui Zhang and Wotao Yin. Gradient methods for convex minimization: Better rates under weaker conditions. https://arxiv.org/abs/1303.4645, 2013.

[45] Yihan Zhou, Victor S. Portella, Mark Schmidt, and Nicholas J. A. Harvey. Regret bounds without Lipschitz continuity: Online learning with relative Lipschitz losses. In *NeurIPS '20: Proceedings of the 34th International Conference on Neural Information Processing Systems*, 2020.

[46] Zhengyuan Zhou, Panayotis Mertikopoulos, Nicholas Bambos, Stephen P. Boyd, and Peter W. Glynn. Stochastic mirror descent for variationally coherent optimization problems. In *NIPS '17: Proceedings of the 31st International Conference on Neural Information Processing Systems*, 2017.

[47] Zhengyuan Zhou, Panayotis Mertikopoulos, Nicholas Bambos, Stephen P. Boyd, and Peter W. Glynn. On the convergence of mirror descent beyond stochastic convex programming. *SIAM Journal on Optimization*, 30(1):687–716, 2020.

Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes]
   (b) Did you describe the limitations of your work? [Yes] **We clearly state the assumptions under which our results are obtained (cf. Section 2).**
   (c) Did you discuss any potential negative societal impacts of your work? [N/A]
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [Yes] **cf. Section 2.**
   (b) Did you include complete proofs of all theoretical results? [Yes] **Complete proofs are provided in the supplement, with proof sketches in the main text.**
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes] **See the supplement.**
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [N/A]
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [Yes]
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [N/A]
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [N/A]
   (b) Did you mention the license of the assets? [N/A]
   (c) Did you include any new assets either in the supplemental material or as a URL? [N/A]
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [N/A]
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [N/A]
5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]
   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]
   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]