Fast Stochastic Bregman Gradient Methods: Sharp Analysis and Variance Reduction

Radu-Alexandru Dragomir 1,2 Mathieu Even 1,2 Mathieu Hendrikx 1,2

Abstract

We study the problem of minimizing a relatively-smooth convex function using stochastic Bregman gradient methods. We first prove the convergence of Bregman Stochastic Gradient Descent (BSGD) to a region that depends on the noise (magnitude of the gradients) at the optimum. In particular, BSGD with a constant step-size converges to the exact minimizer when this noise is zero (interpolation setting, in which the data is fit perfectly). Otherwise, when the objective has a finite sum structure, we show that variance reduction can be used to counter the effect of noise. In particular, fast convergence to the exact minimizer can be obtained under additional regularity assumptions on the Bregman reference function. We illustrate the effectiveness of our approach on two key applications of relative smoothness: tomographic reconstruction with Poisson noise and statistical pre-conditioning for distributed optimization.

1. Introduction

We are interested in solving the minimization problem

$$\min_{x \in C} f(x), \text{ where } f(x) = E_\xi [f_\xi(x)],$$

(1)

where $C$ is a closed convex subset of $\mathbb{R}^d$ and $f_\xi$ are differentiable convex functions. These problems typically arise in machine learning when performing (empirical) risk minimization, in which case $f_\xi$ is for instance a loss function for some random sample $\xi$. Problem (1) is also encountered in signal processing applications such as image deblurring or tomographic reconstruction inverse problems, in which the goal is to recover an unknown signal from a large number of noisy observations. First-order methods are often very efficient for solving problems such as (1), but computing a gradient $\nabla f$ might be very expensive for large-scale problems (large number of components $f_\xi$), and even impossible in the case of true risk minimization (infinite number of $f_\xi$). In this case, stochastic gradient methods have proven to be particularly effective thanks to their low cost per iteration. The simplest one, Stochastic Gradient Descent (SGD), consists in updating $x_t$ as

$$x_{t+1} = \arg \min_{x \in C} \left\{ \eta_t g_t^\top x + \frac{1}{2} \| x - x_t \|^2 \right\},$$

where $g_t$ is a gradient estimate such that $E [g_t] = \nabla f(x_t)$. In our case, a natural choice would be $g_t = \nabla f_\xi(x_t)$ for some $\xi_t$. The choice of the step size $\eta_t$ is crucial for obtaining good performances and is typically related to the smoothness of $f$ with respect to the Euclidean norm.

Beyond simply adapting the step size, a powerful generalization of SGD consists in refining the geometry and performing instead Bregman gradient (a.k.a mirror) steps as

$$x_{t+1} = \arg \min_{x \in C} \left\{ \eta_t g_t^\top x + D_h(x, x_t) \right\},$$

(2)

where the Euclidean distance has been replaced by the Bregman divergence with respect to a reference function $h$, which writes:

$$D_h(x, y) = h(x) - h(y) - \nabla h(y)^\top (x - y),$$

(3)

for all $x \in \text{dom } h, y \in \text{int } \text{dom } h$. We make the following blanket assumptions on $h$ throughout the article, which guarantee well-posedness of the update (2).

Assumption 1. The function $h : \mathbb{R}^n \to \mathbb{R} \cup \{\infty\}$ is twice continuously differentiable and strictly convex on int $C$. Moreover, for every $y \in \mathbb{R}^d$, the problem

$$\min_{x \in C} h(x) - x^\top y$$

has a unique solution, which lies in int $C$.

The standard SGD algorithm corresponds to the case where $h = \frac{1}{2} \| \cdot \|^2$. However, a different choice of $h$ might better fit...
the geometry of the set $C$ and the curvature of the function, allowing the algorithm to take larger steps in directions where the objective gradient changes slowly. This choice is guided by the notion of relative smoothness and strong convexity, introduced in Bauschke et al. (2017); Lu et al. (2018). Instead of the squared Euclidean norm for standard smoothness, relative regularity is measured with respect to the reference function $h$.

**Definition 1.** The function $f$ is said to be $L$-relatively smooth and $\mu$-relatively strongly convex with respect to $h$ if it is differentiable and for all $x, y \in \text{int dom } h$,

$$\mu D_h(x, y) \leq D_f(x, y) \leq LD_h(x, y),$$

(4)

where $D_f$ is defined similarly to (3). Note that if $\mu = 0$, the left-hand side inequality reduces to assuming convexity of $f$. Similarly, if $h = \frac{1}{2} \| \cdot \|^2$, then $D_h(x, y) = \frac{1}{2} \| x - y \|^2$, and the usual notions of smoothness and strong convexity are recovered. If both functions are two times differentiable, Equation (4) can be turned into an equivalent condition on the Hessians: $\mu \nabla^2 h(x) \preceq \nabla^2 f(x) \preceq L \nabla^2 h(x)$. Throughout the article, we will generally write $\mu f/h$ and $L f/h$ to insist on the relative aspect.

Writing the optimality conditions for the minimization problem of Equation (2), we obtain the following equivalent iteration, which is in the alternative Mirror Descent form (Nemirovsky and Yudin, 1983):

$$\nabla h(x_{t+1}) = \nabla h(x_t) - \eta_t g_t.$$  
(5)

Although these updates have a closed-form solution for many choices of the reference function $h$, they may be harder to perform than standard gradient steps, since they require solving the subproblem defined in (2). Yet, this may be worth doing in some cases to reduce the overall iteration complexity, if the resulting majorization in (4) is much tighter than with the Euclidean distance. Let us list some applications of relative regularity:

**Problems with unbounded curvature.** Some problems have singularities at some boundary points in $C$ where the Hessian grows arbitrarily large. In this situation, smoothness with respect to the Euclidean norm does not hold globally, and standard gradient methods become inefficient as they necessitate excessively small step sizes or costly line search procedures. A typical example arises in inverse problems with Poisson noise, which are used in particular for image deblurring (Bertero et al., 2009) or tomographic reconstruction (Kak and Slaney, 2001). In this case, the objective function involves the Kullback-Leibler divergence, which becomes singular as one of its arguments approaches 0. However, by choosing the reference function $h(x) = -\sum_{i=1}^d \log(x^{(i)})$, one can show that relative smoothness holds globally (Bauschke et al., 2017). For more examples, see Lu et al. (2018); Bolte et al. (2018); Nesterov (2019); Mishchenko (2019).

**Distributed optimization.** When $h$ approximates $f$ in the sense of (4), Bregman methods can be used to speed up convergence by performing non-uniform preconditioning (Shamir et al., 2014; Reddi et al., 2016; Yuan and Li, 2020; Hendrikx et al., 2020b). Typically, $h$ is chosen as the objective function on a smaller portion of the dataset of size $n_{\text{prec}}$ (e.g., the dataset of the server), which improves the conditioning by a factor of up to $n_{\text{prec}}$ compared to Euclidean methods, while naturally taking advantage of an eventually small effective dimension of the dataset (Even and Massoulié, 2021). In this case, forming the gradient $g_t$ requires communication with the workers (where most of the data is held), and is thus expensive. Although the updates may not have a simple expression, the inner problem of Equation (2) can be solved locally at the server without additional communications. Therefore, Bregman methods allow to drastically reduce the communication cost by reducing the overall iteration complexity.

Despite these applications, there are still many gaps in our understanding of convergence guarantees of Bregman gradient methods. In particular, most existing results focus on the deterministic case $g_t = \nabla f(x_t)$, or do not leverage the relative regularity assumptions.

**Contributions and outline**

In this work, we develop convergence theorems for Bregman SGD in the relatively-smooth setting. Our variance condition depends on the magnitude of the stochastic gradients at the optimum, and can thus be much smaller than the one used in Hanzely et al. (2018), in particular for overparameritized models (which verify the interpolation condition that all stochastic gradients are equal to 0 at the optimum). Our analysis relies on the Bregman generalization of a few technical lemmas such as the celebrated $\|a + b\|^2 \leq 2(\|a\|^2 + \|b\|^2)$ inequality (Lemma 2) or the co-coercivity inequality (Lemma 3), which we believe to be of independent interest.

Then, we show that variance-reduction techniques, which are widely used to accelerate traditional Euclidean stochastic methods when the objective has a finite-sum structure (Schmidt et al., 2013; Johnson and Zhang, 2013; Defazio et al., 2014; Allen-Zhu, 2017), can be adapted to the Bregman setting. Although this generally requires stronger regularity assumptions (such as global smoothness of $h$ and Lipschitz continuity of $\nabla^2 h^*$), we show that the asymptotical rate of convergence solely depends on relative regularity constants. The same type of results (asymptotic speedup under additional smoothness assumptions) is observed when applying Nesterov-type acceleration to Bregman gradient methods (Hanzely et al., 2018; Dragomir et al., 2021; Hend-


Stochastic Bregman first-order methods and variance reduction

drix et al., 2020b). We provide a summary of the rates proven in this paper in the appendix.

We start by discussing the related work in Section 2. Then, Section 3 presents the results for stochastic gradient descent, along with the main technical lemmas. Section 4 develops a Bregman version of the standard SAGA algorithm (Defazio et al., 2014). Finally, Section 5 illustrates the efficiency of the proposed methods on several applications, including Poisson inverse problems, tomographic reconstruction and distributed optimization.

2. Related work

The Bregman gradient method was first introduced as the Mirror Descent scheme1 (Nemirovsky and Yudin, 1983; Beck and Teboulle, 2003) for minimizing convex nonsmooth functions, and enjoyed notable success in online learning (Bubeck, 2011). More recently, the introduction of relative smoothness (Bauschke et al., 2017; Lu et al., 2018; Bolte et al., 2018) has also brought interest in applying Bregman methods to differentiable objectives. This condition guides the choice of a well-suited reference function \( h \) which can greatly improve efficiency over standard gradient descent. While the vanilla Bregman descent method yields the same convergence rate as the Euclidean counterpart, subsequent work has focused on obtaining better rates with acceleration schemes (Hanzely et al., 2018). However, lower bounds show that the rates for relatively smooth optimization cannot be accelerated in general (Dragomir et al., 2021), and that additional regularity assumptions are needed. Similar notions of relative regularity have also been investigated for non-differentiable functions, such as relative continuity (Lu, 2019; Antonakopoulos et al., 2019). Zhou et al. (2020) also study non-differentiable functions, but in the online setting and without relative continuity.

Stochastic optimization methods, and in particular SGD, are very efficient when the number of samples is high (Bottou, 2012) and are often referred to as “the workhorse of machine learning”. The problem with SGD is that, in general, it only converges to a neighbourhood of the optimum unless a diminishing step-size is used. Variance reduction can be used to counter this problem, and many variance-reduced methods have been developed, such as SAG (Schmidt et al., 2013), SDCA (Shalev-Shwartz and Zhang, 2013; Shalev-Shwartz, 2016), SVRG (Johnson and Zhang, 2013) or SAGA (Defazio et al., 2014).

Surprisingly, stochastic Bregman gradients algorithms have received less attention. Hanzely and Richtárik (2018); Gao et al. (2020); Hendrikx et al. (2020a) study Bregman coordinate descent methods, and Zhang and He (2018) study the non-convex non-smooth setting. Antonakopoulos et al. (2020) study stochastic algorithms for online optimization, under Riemann–Lipschitz continuity. In contrast, our work focuses on Bregman SGD for relatively-smooth objectives. Hanzely and Richtárik (2018) study the same setting and obtain comparable convergence rates, but with a much looser notion of variance, which we discuss more in details in the next section. This is problematic since their bound on the variance is thus proportional to the magnitude of the gradients along the trajectory, and may thus be very large when far from the optimum if \( f \) is strongly convex. In contrast, our definition of variance leverages the stochastic gradients at the optimum, which allows us to obtain significant results without bounded gradients and in the interpolation regime (zero gradients at the optimum). In particular, our analysis can be seen as a Bregman generalization of the analysis from Gower et al. (2019). Davis et al. (2018) also analyze a similar setting, but again with more restrictive assumptions on the noise and boundedness of the gradients. Besides, to the best of our knowledge, variance reduction for Bregman stochastic methods was only studied in Shi et al. (2017) in the context of stochastic saddle-point optimization, but without leveraging relative regularity assumptions like we do in this work.

When preparing the final version of this paper, we became aware of the recent work by (Latafat et al., 2021), who also study Bregman stochastic algorithms for finite-sum minimization. However, the setting they consider is more restricted, as the convergence rates depend on the strong convexity modulus of \( h \) as well as the Euclidean smoothness constant of \( f \).

3. Bregman Stochastic Gradient Descent

3.1. Preliminaries

We start by introducing a few technical lemmas, which are Bregman analogs to well-known Euclidean results, and which are at the heart of our analysis. All missing proofs can be found in Appendix A.

Recall that the conjugate \( h^\ast \) is defined for \( y \in \mathbb{R}^d \) as \( h^\ast(y) = \sup_{x \in \mathbb{R}^d} x^\top y - h(x) \). In particular, under Assumption 1, \( h^\ast \) is convex and differentiable on \( \mathbb{R}^d \) (Bauschke and Combettes, 2011, Cor. 18.12), and \( \nabla h^\ast(\nabla h(y)) = y \) for \( y \in \text{int } C \), which implies the following result:

**Lemma 1** (Duality). For \( x, y \in \text{int } \text{dom } h \), we have \( D_h(x, y) = D_{h^\ast}(\nabla h(y), \nabla h(x)) \).
Yet, it prevents the analysis from applying to coordinate descent methods for instance, in which \(g_i = \nabla_i f(x_t)\), with \(i \in \{1, \ldots, d\}\). However, in this case, the extra structure can also be leveraged to obtain similar results (Hanzely and Richtárik, 2018; Hendrikx et al., 2020a; Gao et al., 2020).

For the variance, Assumption 2 is a Bregman adaptation of the usual definition at the optimum definition used for instance in Bach and Moulines (2011); Gower et al. (2019). Note that if \(h^*\) is \(\mu_h\)-strongly convex with respect to the Euclidean norm, then the assumption is verified for instance when the variance is bounded in \(\ell_2\) norm, as \(\|\nabla f_{\xi_t}(x^*)\|/\ell_{2\mu_h}(z_t) \leq \mu_h\|\nabla f_{\xi_t}(x^*)\|^2\) (we used the fact that if \(h\) is \(\mu_h\)-strongly convex, then \(h^*\) is \(1/\mu_h\)-smooth, see e.g., Kakade et al. (2009)).

We now compare our noise assumption with (Hanzely and Richtárik, 2018, Assumption 5.1.), which writes:

\[
\frac{1}{\eta_t} \mathbb{E}_{\xi_t} \left[ \|\nabla f(x_t) - \nabla f_{\xi_t}(x_t)\|^2 (x_{t+1} - \bar{x}_{t+1}) \right] \leq \sigma^2, \tag{6}
\]

for \(t \geq 0\), where \(g_t\) is the stochastic gradient estimate and \(x_{t+1}\) is the output of the (theoretical) Bregman gradient step taken with the true gradient, that is, \(\nabla h(\bar{x}_{t+1}) = \nabla h(x_t) - \eta_t \nabla f(x_t)\). Thus, their condition can be written:

\[
\frac{1}{\eta_t} \mathbb{E}_{\xi_t} \left[ D_h(x_{t+1}, \bar{x}_{t+1}) + D_h(\bar{x}_{t+1}, x_{t+1}) \right] \leq \sigma^2,
\]

so that \(\sigma^2\) bounds at each step the distance (in the Bregman sense) between \(x_{t+1}\) and \(\bar{x}_{t+1}\), the point that would be obtained by the expected (deterministic) gradient update. To illustrate why our assumption is weaker, let us consider the case where \(h\) is \(\mu_h\)-strongly convex. In this setting, a sufficient condition for (6) to hold is that

\[
\frac{1}{\mu_h} \mathbb{E}_{\xi_t} \left[ \|\nabla f(x_t) - \nabla f_{\xi_t}(x_t)\|^2 \right] \leq \sigma^2, \tag{7}
\]

while a sufficient condition for our variance definition to hold is (using that \(\nabla f(x^*) = 0\)):

\[
\frac{1}{\mu_h} \mathbb{E}_{\xi_t} \left[ \|\nabla f(x^*) - \nabla f_{\xi_t}(x^*)\|^2 \right] \leq \sigma^2, \tag{8}
\]

which only depends on the magnitude of the gradients at the optimum instead of the variance along the full trajectory since \(x_t\) is replaced by \(x^*\). In particular, in the interpolation setting where \(\nabla f_{\xi_t}(x^*) = 0\) for every \(\xi\), \(\sigma^2 = 0\) with our condition. Besides, if \(f\) is strongly convex then the norm of its gradients increases when far from the optimum, and so one needs to restrict \(x_t\) to a compact set of \(\mathbb{R}^d\) for a condition such as (7) to hold. In contrast, the condition from (8) can hold globally without further assumptions.

### 3.3. Convergence results

We now prove the actual convergence theorems for Bregman SGD. To avoid notation clutter, we generally omit with respect to which variable expectations are taken when clear from the context.

**Theorem 1.** If \(f\) is \(L_{f/h}\)-smooth and \(\mu_{f/h}\)-strongly convex relative to \(h\) with \(\mu_{f/h} > 0\), and Assumptions 1 and 2 hold,
then for $\eta \leq 1/(2L_{f/h})$, the iterates produced by Bregman stochastic gradient (2) satisfy

$$\mathbb{E} \left[ D_h(x^*, x_t) \right] \leq (1 - \eta f/h)^t D_h(x^*, x_0) + \eta \frac{\sigma^2}{f/h}. \quad (9)$$

Note that since we are in a Bregman setting, convergence is measured in terms of $D_h(x^*, x_t)$, the distance between $x^*$ and $x_t$ in the metric induced by $h$. If $h$ is $\mu_h$-strongly convex, then $D_h(x^*, x_t) \geq \frac{\mu_h}{2} \|x_t - x^*\|^2$ and convergence in $\ell_2$ distance is recovered.

**Proof.** By using Lemma 4 from Appendix A, we obtain:

$$\mathbb{E}_{\xi_t} \left[ D_h(x^*, x_{t+1}) \right] = D_h(x^*, x_t) - \eta D_f(x^*, x_t) - \eta D_f(x_t, x^*) + \mathbb{E}_{\xi_t} \left[ D_h(x_t, x_{t+1}) \right]. \quad (10)$$

Using Lemma 2, the last term can be bounded as $D_h(x_t, x_{t+1}) \leq \frac{1}{2} [D_1 + D_2]$. We use Lemma 3 (Bregman co-coercivity) to write:

$$D_1 = D_h(x^*, x_t) - 2\eta \left[ \nabla f_{\xi_t}(x_t) - \nabla f_{\xi_t}(x^*) \right] \cdot \nabla h(x_t) \leq 2\eta D_{f_{\xi_t}}(x_t, x^*),$$

so that $\mathbb{E}_{\xi_t} \left[ D_1/2 \right] \leq \eta D_f(x_t, x^*)$. Similarly,

$$D_2 = D_h(x^*, x_t) - 2\eta \nabla f_{\xi_t}(x^*) \cdot \nabla h(x_t), \quad (11)$$

so that $\mathbb{E}_{\xi_t} \left[ D_2/2 \right] \leq \eta^2 \sigma^2$. Thus, using the relative strong convexity of $f$ to bound the $D_f(x^*, x_t)$ term, we obtain:

$$\mathbb{E}_{\xi_t} \left[ D_h(x^*, x_{t+1}) \right] \leq (1 - \eta f/h) D_h(x^*, x_t) + \eta^2 \sigma^2, \quad (12)$$

which yields the desired result. \hfill \square

**Remark 1** (Interpolation). In the interpolation setting (when $\nabla f_{\xi_t}(x^*) = 0$ for all $\xi_t$, we have that $\sigma^2 = 0$. **Theorem 1** thus proves linear convergence in this case. For instance, when solving objectives of the form $D_{KL}(Ax, b)$ (which has applications in optimal transport (Mishchenko, 2019)) or $D_{KL}(b, Ax)$ (which has application in deblurring or tomographic reconstruction), then the variance as defined in Hanzely and Richtárik (2018) may be unbounded, whereas the variance as we define it is equal to 0 if there exists $z$ such that $Az = b$.

When $f$ is convex ($\mu_f/h = 0$), **Theorem 1** can be adapted to obtain a $1/T$ decrease of the error up to a noise region.

**Theorem 2** (Convex case). Under the same assumptions as **Theorem 1**, if $\mu = 0$, then

$$\mathbb{E} \left[ \frac{1}{T} \sum_{t=0}^{T} D_f(x^*, x_t) \right] \leq \frac{D_h(x^*, x_0)}{\eta T} + \eta \sigma^2 \quad (13)$$

Contrary to the Euclidean case, we do not obtain a guarantee on the average iterate in general. This is because the bound is on the average of $D_f(x_t, x_{t+1})$ instead of $D_f(x_t, x^*)$, and Bregman divergences are not necessarily convex in their second argument (except for the Euclidean distance and Kullback-Leibler divergence). Therefore, the final bound is obtained on $\min_{i} D_f(x_t, x^*)$, meaning that there is at least one $x_t$ such that this is true. Note that the nice properties regarding interpolation still hold in this setting.

**Proof.** We start from Lemma 4 and bound the $D_h(x_t, x_{t+1})$ in the same way as when $\mu > 0$, which yields:

$$\eta D_f(x^*, x_t) = D_h(x^*, x_t) - \mathbb{E}_{\xi_t} \left[ D_h(x^*, x_{t+1}) \right] + \eta^2 \sigma^2.$$

Averaging over $t$ and dividing by $\eta$ leads to (13). \hfill \square

### 4. Variance reduction

We have shown in the previous section that BSGD enjoys guarantees that are similar to that of its Euclidean counterpart, although the notion of variance needs to be adapted. We show in this section that it is also possible to apply variance reduction to accelerate convergence. To this end, we solve for $n \in \mathbb{N}^+$ and some convex functions $f_i$:

$$\min_{x \in C} f(x) := \frac{1}{n} \sum_{i=1}^{n} f_i(x). \quad (14)$$

The difference with Section 3 is that we now assume that $f$ is a finite sum, which is required for variance reduction. We also assume that the minimizer $x^*$ belongs to int $C$, so that $\nabla f(x^*) = 0$. The case where $x^*$ lies on the border of $C$ is more delicate, as $h$ might not be differentiable there (e.g., the log-barrier); this would require an involved technical analysis which we leave for future work.

To solve Problem (14), we consider Algorithm 1, which is a Bregman adaptation of the SAGA algorithm (Defazio et al., 2014). Following its Euclidean counterpart, Algorithm 1 stores the stochastic gradients computed at each iteration, and reuses them to estimate the full gradient. Therefore, only one stochastic gradient needs to be computed at each iteration, thus drastically reducing the iteration cost compared to batch gradient descent, at the expense of using more memory. Note that the stochastic updates are unbiased since $\mathbb{E}_{\xi_t} [g_{\xi_t}] = \nabla f(x_t)$, and at the optimum (when $x_t = \phi_{\xi_t} = x^*$ for all $i$), $g_{\xi_t} = \nabla f(x^*) = 0$ so the variance at the optimum is 0 (contrary to SGD). We now study the convergence guarantees of Algorithm 1 in more details.

#### 4.1. Convergence Results

For analyzing the Bregman-SAGA scheme, we first need to introduce, in addition to relative smoothness, an assumption on the regularity of $D_h$. 

Algorithm 1 Bregman-SAGA((η_t)_{t≥0}, x_0)

1: φ_i = x_0 for i = 1, ..., n
2: for t = 0, 1, 2, ... do
3:   Pick i_t ∈ {1, ..., n} uniformly at random
4:   g_t = ∇ f_{i_t}(x_t) − ∇ f_{i_t}(φ_{i_t}^t) + \frac{1}{2} \sum_{j=1}^n ∇ f_j(φ_{j}^t)
5:   x_{t+1} = arg min_{x} \{η_t g_{i_t}^t x + D_h(x, x_{i_t}) \}
6:   φ_{i_t}^{t+1} = x_t, and store ∇ f_{i_t}(φ_{i_t}^{t+1}).
7:   φ_j^{t+1} = φ_{j}^t for j ≠ i_t.
8: end for

Assumption 3. For all i ∈ {1, ..., n}, f_i is L_{f/h} relatively smooth w.r.t. h, and f is μ_{f/h} relatively strongly convex w.r.t. h. Moreover, there exists a gain function G such that for any x, y, v ∈ ℝ^d and λ ∈ [−1, 1],

\[ D_{h^\ast}(x + λv, x) ≤ G(x, y, v)λ^2 D_{h^\ast}(y + v, y). \]

Such structural assumptions appear to be essential for analyzing Bregman-type methods that use information provided by gradients of past iterates. The function G models the fact that the Bregman divergence D_{h^\ast}(x + v, x) is not homogeneous nor invariant to translation in x in general (except for the Euclidean case where it is equal to \|v\|^2/2). Note that such difficulties are also encountered for obtaining accelerated rates with inertial variants of Bregman descent, where similar assumptions are needed (Hanzely et al., 2018). This seems unavoidable, as suggested by the lower bound in Dragomir et al. (2021).

Although the gain function G is relatively abstract at this point, it plays a key role in defining the step-size, and convergence guarantees similar to those of Euclidean SAGA can be obtained provided G can be chosen small enough. We first state the general Theorem 3 (convergence proof for Algorithm 1), and then detail how G can be bounded in several interesting cases.

For \( t ≥ 0 \) and step-sizes \( η_t > 0 \), define \( H_t = \frac{1}{n} \sum_{i=1}^n D_f(φ_i^t, x^*) \), and the potential \( \psi_t \) as follows:

\[ \psi_t = \frac{1}{η_t} D_h(x^*, x_t) + \frac{n}{2} H_t. \] (15)

First note that by convexity of \( h \) and of the \( f_i, \psi_t ≥ 0 \) for all \( t \). Our goal in this section is to show that \( \{ψ_t\}_{t≥0} \) converges to 0 at a given speed. Indeed, since \( D_h(x_t, x^*) ≤ ψ_t \), this implies (as in Section 3) that \( x_t \) converges to \( x^* \) at the same rate. To ease notations, we define

\[ \tilde{α}_t = \frac{1}{n} \sum_{j=1}^n ∇ f_j(φ_j^t), \quad \check{α}_t = ∇ f_i(φ_i^t) − \tilde{α}_t. \] (16)

Theorem 3. Assume that Algorithm 1 is run with a step size sequence \( \{η_t\}_{t≥0} \) satisfying \( η_t = 1/(8L_{f/h}G_i) \) for every \( t ≥ 0 \), with \( G_i \) decreasing in \( t \) and such that for all \( j ∈ \{1, ..., n\} \):

\[ G_t ≥ G \left( ∇ h(x_t), ∇ h(x_t), \frac{1}{L_{f/h}} (∇ f_j(x_t) − ∇ f_j(x^*)) \right), \]

\[ G_t ≥ G \left( ∇ h(x_t) − 2η_t \check{α}_t, ∇ h(φ_i^t), \frac{1}{L_{f/h}} (∇ f_j(φ_j^t) − ∇ f_j(x^*)) \right). \]

Then, under Assumptions 1 and 3, the potential \( ψ_t \) satisfies

\[ \mathbb{E}_t [ψ_{t+1}] ≤ \left( 1 - \min \left( \frac{η_t μ_{f/h}}{2}, \frac{1}{2η_t} \right) \right) ψ_t, \] (17)

In the convex case (\( μ_{f/h} = 0 \)), we obtain that

\[ \mathbb{E} \left[ \frac{1}{4T} \sum_{t=1}^T D_f(x_t, x*) + H_t \right] ≤ \frac{ψ_0}{T}, \] (18)

Proof. Similarly to BSGD, we apply Lemma 4 (Appendix A), which yields

\[ \mathbb{E}_t [D_h(x^*, x_{t+1})] = D_h(x^*, x_t) − η_t D_f(x^*, x_t) \]

\[ −η_t D_f(x_t, x^*) + \mathbb{E}_t [D_h(x_t, x_{t+1})] \] (19)

Lemmas 1 and 2 yield \( D_h(x_t, x_{t+1}) ≤ (D_1 + D_2)/2 \), with

\[ D_1 = D_{h^\ast}(∇ h(x_t) − 2η_t [∇ f_i(x_t) − ∇ f_i(x^*)], ∇ h(x_t)), \]

\[ D_2 = D_{h^\ast}(∇ h(x_t) − 2η_t [∇ f_i(x^*) − \tilde{α}_t], ∇ h(x_t)). \]

Using Assumption 3 together with Lemma 3, we obtain:

\[ D_1 ≤ 4η_t^2 L_{f/h}^2 G_i \]

\[ D_{h^\ast}(∇ h(x_t) − \frac{1}{L_{f/h}} (∇ f_i(x_t) − ∇ f_i(x^*)) , ∇ h(x_t)) \]

\[ ≤ 4η_t^2 L_{f/h}^2 G_i D_f(x_t, x^*). \]

To bound the second term, we use Lemma 5 (Pfau, 2013) which is a Bregman version of the bias-variance decomposition. We write \( V = 2η_t [∇ f_i(φ_i^t) − ∇ f_i(x^*)] \), so that

\[ \mathbb{E}_t [V] = 2η_t \check{α}_t \] and:

\[ \mathbb{E}_t [D_2] = \mathbb{E}_t [D_{h^\ast}(∇ h(x_t) − \mathbb{E}_t [V] + V, ∇ h(x_t))]

\[ ≤ \mathbb{E}_t [D_{h^\ast}(∇ h(x_t) − \mathbb{E}_t [V] + V, ∇ h(x_t) − \mathbb{E}_t [V])]

\[ ≤ 4η_t^2 L_{f/h}^2 G_i \mathbb{E}_t [∇ h(φ_i^t) − L_{f/h}^{-1} (∇ f_i(φ_i^t) − ∇ f_i(x^*))], ∇ h(φ_i^t)] \]

\[ ≤ 4η_t^2 G_i L_{f/h} \mathbb{E}_t [D_f(φ_i^t, x^*)] \]

where we used the gain function for translation and rescaling the step size. Following Hofmann et al. (2015), we write:

\[ \mathbb{E}_t [H_{t+1}] = \left( 1 - \frac{1}{n} \right) H_t + \frac{1}{n} D_f(x_t, x^*). \] (20)
Therefore, we can use the $-H_t/n$ term to control the excess term from bounding $D_h(x_t, x_{t+1})$. In the end, we obtain:

$$
E_t [\psi_{t+1} - \psi_t] \leq -D_f(x^*, x_t) - \left( \frac{1}{2} - 2\eta_t L_{f/h} G_t \right) H_t - \left( 1 - 2\eta_t L_{f/h} G_t - \frac{1}{2} \right) D_f(x_t, x^*).
$$

If we choose $\eta_t \leq 1/(8 L_{f/h} G_t)$ then the last term is positive and $1 - 4\eta_t L_{f/h} G_t \geq 1/2$. If $\mu_{f/h} > 0$ then we use the relative strong convexity of $f$ to obtain that the right hand side is proportional to $\psi_t$, thus leading to a linear convergence rate. Otherwise, we obtain a telescopic sum, leading to the $1/T$ rate of Equation (18). 

Note that the monotonicity of $\eta_t$ (through $G_t$) is a technical condition to ensure that the Lyapunov is non-increasing. Otherwise, $\psi_t$ could blow up even though $x_{t+1}$ is very close to $x_t$, simply because $\eta_t$ shrinks. It could be replaced by the condition that $\eta_t$ does not vary too much (not more than a factor $1 - O(1/n)$), which achieves the same goal. The rest of this section is devoted to shoring that non-trivial $G_t$ can be chosen in many cases, thus leading to strong convergence guarantees. In particular, the rate recovers that of Euclidean SAGA in case $h$ is a quadratic form.

**Corollary 1.** If $\nabla^2 h$ is constant ($h$ is quadratic), then Assumption 3 is satisfied with $G = 1$, so that

$$
E [\psi_t] \leq \left( 1 - \min \left( \frac{1}{8\kappa_f}, \frac{1}{2n} \right) \right)^t \psi_0,
$$

where $\kappa_f = L_f / \mu_{f/h}$ is the relative condition number.

If $h$ is not quadratic, but $f^*$ and $h^*$ are regular with respect to a norm, then strong guarantees can also be obtained:

**Corollary 2.** If $h^*$ is $\mu_{h^*}^{-1}$-smooth and $f^*$ is $L_f^{-1}$-strongly convex with respect to a norm $\| \cdot \|_2$, then the step size can be chosen constant as $\eta_t = \frac{\mu_{h^*}}{8L_f}$, and

$$
E [\psi_t] \leq \left( 1 - \min \left( \frac{\mu_{h^*} \mu_f}{8L_f}, \frac{1}{2n} \right) \right)^t \psi_0.
$$

Note that following Kakade et al. (2009), having $h^*$ be $\mu_{h^*}^{-1}$-smooth is equivalent to having $h$ be $\mu_h$ strongly-convex.

**Proof.** The proof follows the same step as the proof of Theorem 3, but the translation invariance and homogeneity are obtained by comparison with the norm, instead of using Assumption 3. Thus, we pay a factor $\mu_{h^*}^{-1}$ when bounding $D_h$, by the norm, and a factor $L_f$ when bounding the norm by $D_f$. It is also possible to directly use Assumption 3, but in this case the $L_f$ factor is replaced by $L_{f/h} L_h$, which is an upper bound on $L_f$, and may thus be slightly looser. 

Note that Corollary 1 is actually a consequence of Corollary 2, since $\mu_h = 1$ and $L_f = L_{f/h}$ if $D_h$ is a norm itself. Otherwise, the constant $G_t$ is chosen in a rather pessimistic way, and depends on the difference between directly bounding $D_f$ by $D_h$ (in which case we pay a factor $L_{f/h}$), or going through a norm $\| \cdot \|$ in the middle (in which case we pay $L_{f/h} / \mu_{h^*} \geq L_{f/h}$).

As stated at the beginning of this section, one of the problems is that Bregman divergences lack translation invariance and homogeneity. However, as the algorithm converges, one can expect these conditions to hold locally, as $D_{h^*} (x + \nu, x)$ is approximated by $\frac{1}{2} \| \nu \|^2_{\nabla^2 h^* (x^*)}$ for small enough $\nu$, and $x$ close enough to $x^*$. This is indeed what happens under enough regularity assumptions on $h$.

**Proposition 1.** If $h$ is $L_h$-smooth and the Hessian $\nabla^2 h^*$ is $M$-smooth, then the gain function can be chosen as:

$$
G(x, y, \nu) = 1 + 2ML_h (\| y - x \| + \| \nu \|).
$$

Note that, even if the regularity conditions of Proposition 1 do not hold globally (such as for problems with unbounded curvature), they are at least valid on every bounded subset of $\text{int } C$, as soon as $h$ is $C^3$ on $\text{int } C$. We now explicit a possible explicit choice for $G_t$ in this setting.

**Corollary 3.** Assume that $h$ is $L_h$-smooth, $\mu_h$-strongly convex and that the Hessian $\nabla^2 h^*$ is $M$-smooth. Then, there exists an explicit constant $C$ such that if Algorithm 1 is run with a step size $\eta_t = 1/(8 L_{f/h} G_t)$ with $G_t$ decreasing and satisfying

$$
G_t \geq \min \left( \frac{L_{f/h} L_h}{\mu_h}, \right.
$$

$$
1 + C \left( \sum_{j=1}^n \| x_t - \phi_j^* \| + \| \sum_{j=1}^n \nabla f_j (\phi_j^*) \| \right)
$$

then we have the convergence rate

$$
E [\psi_{t+1}] \leq \left( 1 - \min \left( \frac{1}{8G_t \kappa_f}, \frac{1}{2n} \right) \right)^t \psi_t,
$$

where $\lim_{t \to \infty} G_t = 1$, or, more precisely,

$$
E [G_t] \leq 1 + O \left( 1 - \min \left( \frac{1}{8G_t \kappa_f}, \frac{1}{2n} \right) \right)^t.
$$

The explicit expression for the constant $C$ is provided in Appendix B along with the proof. Although the result involves smoothness constants of $h$ which can be large in the relatively-smooth setting, this dependence disappears asymptotically. Hence, after some time $t$, which we can roughly estimate using Equation (25), we obtain that $G_t = O(1)$. Thus, we reach the same kind of convergence
rate as in the ideal quadratic case, which depends only on the relative condition number $\kappa_{f/h}$, but with more general functions $h$, and thus possibly much better conditioning. Besides, the order of magnitude required for $G_t$ can be estimated during the optimization process using Equation (23).

4.2. Remarks on adaptivity

Assumption 3 highlights the fact that the key difficulty is purely geometric, and that in general we need to make up for the lack of translation invariance and homogeneity of Bregman divergences. Although Corollary 3 gives a criterion for $G_t$ that can be evaluated throughout training (since the constant $C$ is explicit), several approximations are required to obtain it, and it may be loose overall. Yet, for the theory to hold, it suffices to have $\eta_t$ small enough such that:

$$E_i D_h(x_t, x_{t+1}) \leq \frac{\eta_t}{4} [D_f(x_t, x^*) + E_i D_{f_i}(\phi^i_t, x^*)].$$

Unfortunately, one would need to know $x^*$ to evaluate such a condition, which is thus hard to use in practice. For the sake of clarity, we have only presented results for Bregman SAGA in this section. Yet, similar results hold for SVRG-style variance reduction, and we present them in Appendix C. An important difference is that in this case, $\phi^i_t = \phi_t$ for all $i$, and so the last term becomes $E [D_{f_i}(\phi_t, x^*) | F_t] = f(\phi_t) - f(x^*)$ since $\nabla f(x^*) = 0$, so we only need to know $f(x^*)$ (or an estimation of it) in order to compute this criterion. In this case, we don’t need to know the relative smoothness constant of the problem and the step-size can be set adaptively, similarly to Barré et al. (2020). Although it may be expensive to compute $D_f(x_t, x^*)$ at each iteration, one can also approximate $f(x_i)$ on the fly, or only update $\eta_t$ periodically. On a side note, a similar criterion could be used for BSGD (without the second term in this case), in particular for over-parametrized problems for which we know that $f(x^*) = 0$.

5. Experiments

In order to show the effectiveness of our method, we consider the two key settings mentioned in the introduction: problems with unbounded curvature (inverse problems with Poisson noise) and preconditioned distributed optimization. The first setting corresponds to the convex case ($\mu_{f/h} = 0$), whereas the second one corresponds to the relatively strongly convex case ($\mu_{f/h} > 0$). We observe that leveraging stochasticity (and, when needed, variance reduction) drastically improves the performance of Bregman methods in both cases. Additional details on the setting (such as the precise formulation of the objective or the relative smoothness constants) are given in Appendix D.

5.1. Poisson inverse problems

Figure 1(a) considers the minimization problem $\min_{x \in \mathbb{R}^d} f(x) = D_{KL}(b, Ax)$, where $A \in \mathbb{R}^{n \times d}$ and $D_{KL}(u, v) = \sum_{i=1}^n u_i \log(u_i/v_i) - u_i + v_i$ is the Kullback-Leibler divergence. The goal is to recover an unknown signal $x_*$, observed through the matrix $A$ and corrupted by Poisson noise. This is a fundamental signal processing problem, with applications in astronomy and medicine (see Bertero et al. (2009) for a review). We use the log-barrier reference function, $h(x) = -\sum_i \log x_i$, for which relative smoothness holds with $L_{f/h} = \sum_{i=1}^n b_i/n$ (Bauschke et al., 2017).

We verify experimentally in this section that SGD is fast when the gradients at optimum are zero by first studying a problem where $b = Ax^*$. $A \in \mathbb{R}^{n \times d}$ and $x \in \mathbb{R}^d$ are random (indices sampled uniformly between 0 and 1), with $n = 10000$ and $d = 1000$. We compare the results of the deterministic and stochastic versions of Bregman Gradient descent. We also compare to the Multiplicative Updates (MU) algorithm, also known as Lucy-Richardson or Expectation-Maximization (Shepp and Vardi, 1982), which is a standard baseline for this problem. We observe that BGD is by far the slowest algorithm, but that BSGD is faster than Lucy-Richardson thanks to the stochastic speedup. We also observe that BSGD does not plateau in a noise region and converges to the true solution, which is consistent with Theorem 2. The step-size for BGD and BSGD is chosen as $1/L_{f/h}$, whereas Lucy-Richardson is parameter-free.

Figure 1(b) considers experiments on the tomographic reconstruction problem on the standard Shepp-Logan phantom (Kak and Slaney, 2001). Due to space limitations, the main text mainly describes the results, but the setting details can be found in Appendix D. The step-size given by theory was rather conservative in this case, so we increased it by a factor of 5 for all Bregman algorithms (and even 10 for BGD). Figure 1(b) shows again that stochastic algorithms drastically outperform BGD. Yet, BSGD quickly reaches a plateau because of the noise. On the other hand, BSAGA enjoys variance reduction and fast convergence to the optimum. In this case, BSAGA is on par with MU, the state-of-the-art algorithm for this problem. This is because of the log barrier that allows relative smoothness to hold, but heavily slows down Bregman algorithms when coordinates are close to 0. Yet, these results are encouraging and one may hope for even faster convergence of BSAGA for tomographic reconstruction with a tighter reference function.

5.2. Statistically Preconditioned Distributed Optimization

In this section we consider the problem of solving a distributed optimization problem in which data is distributed...
among many workers. We closely follow the setting of Hendrikx et al. (2020a), and solve a logistic regression problem for the RCV1 dataset (Lewis et al., 2004). Function $h$ is taken as the same logistic regression objective as for the global objective $f$, but on a much smaller dataset of size $n_{\text{prec}} = 1000$ and with an added regularization $c_{\text{prec}} = 10^{-5}$. In this case, BGD corresponds to a widely used variant of DANE (Shamir et al., 2014), in which only the server performs the update. The stochastic updates in BSGD are obtained by subsampling a set of workers at each iteration, so that all the nodes do not have to participate in every iteration. Regularization is taken as $\lambda = 10^{-5}$, and there are $n = 100$ nodes with $N = 1000$ samples each. A fixed learning rate is used, and the best one is selected among $[0.025, 0.05, 0.1, 0.25, 0.5, 1]$. BGD uses $\eta = 0.5$ while SAGA and BSGD use $\eta = 0.05$. The x-axis represents the total number of communications (or number of passes over the dataset). Note that at each epoch, BGD communicates once with all workers (one round trip for each worker) whereas BSGD and BSAGA communicate $n$ times with one worker sampled uniformly at random each time. Therefore, BSAGA requires much less gradients from the workers to reach a given precision level, yet, it is at the cost of having to solve more local iterations.

Figure 1(c) first shows that BSAGA clearly outperforms BGD. BSGD on the other hand is as fast as BSAGA at the beginning of training, until it hits a variance region at which it saturates. This is consistent with the theory, and is similar to what can be observed in the Euclidean case. An interesting feature is that although the step-size has to be selected smaller than that of gradient descent (which is also the case in the Euclidean setting since $f$ is smoother than the least smooth $f_j$), choosing a constant step-size is enough to ensure convergence in this case, thus hinting at the fact that the analysis is rather conservative and that $G_t$ does not slow down the algorithm as much as we could have feared when far from the optimum. This is consistent with the results obtained by Hendrikx et al. (2020b) on acceleration.

6. Conclusion

Throughout the paper, we have (i) given tight convergence guarantees for Bregman SGD that allow to accurately describe its behaviour in the interpolation setting, and (ii) introduced and analyzed Bregman analogs to the standard variance-reduced algorithm SAGA. These convergence results require stronger assumptions on the objective than relative smoothness and strong convexity, but we show that fast rates can be obtained nonetheless when $h$ is nicely behaved (quadratic or Lipschitz Hessian). We also prove that these fast rates can be obtained for more general functions $h$ after a transient regime. Besides, we show experimentally that variance reduction greatly accelerates Bregman first-order methods for several key applications, including distributed optimization and tomographic reconstruction. In particular, there does not seem to be a slow transient regime in the applications considered, despite the lack of regularity of the objectives. This need for higher order regularity assumptions but great practical performance is consistent with the results obtained for acceleration in the Bregman setting. Better understanding the transient regime (in which $G_t$ can be high) and finding better reference functions $h$ for the tomographic reconstruction problem are two promising extensions of our work.

Acknowledgements

RD was supported by an AMX fellowship. RD would like to acknowledge support from the Air Force Office of Scientific Research, Air Force Material Command, USAF, under grant number FA9550-19-1-7026/191OE033 and FA9550-18-1-0226. HH was funded in part by the French government under management of Agence Nationale de la Recherche as part of the “Investissements d’avenir” program, reference ANR-19-P3IA-0001 (PRAIRIE 3IA Institute). HH also acknowledges support from the European Research Council (grant SEQUOIA 724063) and from the MSR-INRIA joint centre.
Stochastic Bregman first-order methods and variance reduction

References


Stochastic Bregman first-order methods and variance reduction


Shai Shalev-Shwartz. SDCA without duality, regularization, and individual convexity. In International Conference on Machine Learning, 2016.


