Parameter Free Dual Averaging: Optimizing Lipschitz Functions in a Single Pass

Aaron Defazio
Meta AI, Fundamental AI Research team

Konstantin Mishchenko
CNRS, ENS, INRIA SIERRA team

Abstract

Both gradient descent and dual averaging for convex Lipschitz functions have convergence rates that are highly dependent on the choice of learning rate. Even when the Lipschitz constant is known, setting the learning rate to achieve the optimal convergence rate requires knowing the distance from the initial point to the solution set D. A number of approaches are known that relax this requirement, but they either require line searches, restarting (hyper-parameter grid search), or do not derive from the gradient descent or dual averaging frameworks (coin-betting). In this work we describe a single pass method, with no back-tracking or line searches, derived from dual averaging, which does not require knowledge of D yet asymptotically achieves the optimal rate of convergence for the complexity class of Convex Lipschitz functions.

1. Introduction

We consider the class of unconstrained convex Lipschitz functions with Lipschitz constant G. Starting at a point x_0, at each step we may query a subgradient g_k and function value $f(x_k)$ at a query point x_k. Let x_* be any minimizer of f, and denote $f_* = f(x_*)$.

For this class, classical convergence results for sub-gradient descent:

$$x_{k+1} = x_k - \gamma_k g_k,$$

with a fixed step size require knowledge of the distance to solution term $D = \|x_0 - x_*\|$ in order to achieve the optimal rate of convergence. Using a step size:

$$\gamma_{k+1} = \frac{D}{G\sqrt{k+1}},$$

the average iterate $\hat{x}_n$ converges in terms of function value at an inverse-sqrt rate:

$$f(\hat{x}_n) - f_* = \mathcal{O}(DG/\sqrt{n+1}).$$

This rate is worse case optimal for this complexity class [6]. Knowledge of the constant G can be removed by using AdaGrad [3] step sizes:

$$\gamma_k = \frac{D}{\sqrt{\sum_{i=0}^{k} \|g_i\|^2}}.$$

Under the mild assumption that loose lower and upper bounds are known on D, just doing a hyper-parameter grid search on a log spaced scale from d_0 to $d_{\max}$ gives a rate:

$$f(x_n) - f_* = \mathcal{O}\left(\frac{DG\log(d_{\max}/d_0)}{\sqrt{n+1}}\right),$$

which is the approach taken in practice in most applications. In this work, we describe a modification of dual averaging that achieves the optimal rate, for sufficiently large n, by maintaining and updating a lower bound on D, which is then used as part of the step size. Using this lower bound on D is provably sufficient to achieve the optimal rate of convergence. The method we describe is "parameter-free" according to the standard usage of the term [7], as it requires knowledge of G but not D.

2. Related Work

There are a number of approaches to optimization of Lipschitz functions that achieve independence of problem parameters, we review the major classes of approaches below.

2.1. Polyak step size

We can trade the requirement of knowledge of D to knowledge of f_*, by using the Polyak step size[10]:

$$\gamma_k = \frac{f(x_k) - f_*}{\|g_k\|^2}.$$

This gives the optimal rate of convergence without any additional log factors. Using estimates or approximations of f_* tend to result in unstable convergence, however a restarting scheme that maintains lower bounds on f_* can be shown to converge within a multiplicative log factor of the optimal rate [5].

2.2. Exact line searches

The following method relying on an exact line search also gives the optimal rate, without requiring any knowldge of problem parameters [2, 4]:

$$\begin{aligned}
s_{k+1} &= s_k + g_k, \\
\gamma_{k+1} &= \arg\min f_{k+1}\left(\frac{k+1}{k+2}x_k + \frac{1}{k+2}\left(z_0 - \gamma_{k+1}s_{k+1}\right)\right), \\
z_{k+1} &= z_0 - \gamma_{k+1}s_{k+1}, \\
x_{k+1} &= \frac{k+1}{k+2}x_k + \frac{1}{k+2}z_{k+1}.
\end{aligned}$$

Relaxing this exact line search to an approximate line search is non-trivial, and will potentially introduce additional dependencies on problem constants.

2.3. Bisection

Instead of running sub-gradient descent on every grid-point on a log spaced grid from d_0 to $d_{\max}$, we can use more sophisticated techniques to instead run a bisection algorithm on the same grid, giving an improvement of an additional log factor[1]:

$$f(x_n) - f_* = \mathcal{O}\left(\frac{DG \log\log(d_{\max}/d_0)}{\sqrt{n+1}}\right),$$

This can be further improved by estimating $d_{\max}$, which allows us to replace $d_{\max}$ with D in this bound.

2.4. Coin-betting

If we assume knowledge of G but not D, coin betting approaches can be used. Coin-betting [9] is normally analyzed in the online-convex optimization framework, which is more general then our setting and for that class, coin-betting methods achieve optimal regret among methods without knowledge of D, which is a log factor worse than the best possible regret with knowledge of D [7]:

$$\text{Regret}_n = \mathcal{O}\left(DG\sqrt{(n+1)\log(1+D)}\right).$$

Using online to batch conversion gives a rate of convergence in function value of

$$f(\hat{x}_n) - f_* = \mathcal{O}\left(\frac{DG \log(1+D)}{\sqrt{n+1}}\right).$$

3. Algorithm

The algorithm we propose is Algorithm 1. It is a modification of the AdaGrad step-size applied to weighted dual averaging, also known as linear follow-the-regularized-leader in the online learning community. The key idea is simple. At each step, we construct a lower bound $\hat{d}_k$ on D. If this bound is less than twice our current best estimate of d_k of D, we continue to use d_k. Otherwise, we replace d_k in our step size by $\hat{d}_k$, and proceed normally. To construct the lower bound, we use the technique of analyzing a "phantom" point:

$$x'_k = x_0 - \gamma_* s_k,$$

which differs from the x_k sequence used in the algorithm by using a different fixed step size sequence $\gamma_* = \gamma_{n+1}/2$. By using this point, we are able to gain a additional negative term $-\frac{1}{4}\gamma_{n+1}\left\|s_{n+1}\right\|^2$ in our upper bound on the weighted sum of the function values:

$$\sum_{k=0}^{n} d_k\left(f(x_k) - f_*\right) \leq D\left\|s_{n+1}\right\| + \sum_{k=0}^{n} \frac{\gamma_k}{2} d_k^2 \left\|g_k\right\|^2 - \frac{\gamma_{n+1}}{4}\left\|s_{n+1}\right\|^2.$$

Using the fact that $\sum_{k=0}^{n} d_k\left(f(x_k) - f_*\right) \geq 0$, we have:

$$0 \leq D\left\|s_{n+1}\right\| + \sum_{k=0}^{n} \frac{\gamma_k}{2} d_k^2 \left\|g_k\right\|^2 - \frac{\gamma_{n+1}}{4}\left\|s_{n+1}\right\|^2,$$

Algorithm 1 Parameter Free Dual Averaging

Input: d_0, x_0

$s_0 = 0$

for $k = 0$ **to** n **do**

$g_k \in \partial f(x_k)$

$s_{k+1} = s_k + d_k g_k$

$$\gamma_{k+1} = \frac{1}{\sqrt{G^2 + \sum_{i=0}^{k} \|g_i\|^2}}$$

$$\hat{d}_{k+1} = \frac{\frac{\gamma_{k+1}}{2} \|s_{k+1}\|^2 - \sum_{i=0}^{k} \gamma_i d_i^2 \|g_i\|^2}{2 \|s_{k+1}\|}$$

if $\hat{d}_{k+1} > 2d_k$ **then**

$d_{k+1} = \hat{d}_{k+1}$

else

$d_{k+1} = d_k$

end if

$x_{k+1} = x_0 - \gamma_{k+1} s_{k+1}$

end for

Return $\hat{x}_n = \frac{1}{\sum_{k=0}^{n} d_k} \sum_{k=0}^{n} d_k x_k$

which can be rearranged to yield a lower bound on D, involving only known quantities:

$$D \geq \hat{d}_{n+1} = \frac{\frac{\gamma_{n+1}}{2} \|s_{n+1}\|^2 - \sum_{k=0}^{n} \gamma_k d_k^2 \|g_k\|^2}{2 \|s_{n+1}\|}.$$

This bound is potentially vacuous if $\|s_{n+1}\|^2$ is small in comparison to $\sum_{k=0}^{n} \gamma_k d_k^2 \|g_k\|^2$, however we are able to show that the algorithm converges rapidly in that case, and so further increases of d_k are not needed.

Theorem 1 *For a convex G-Lipschitz function f. Algorithm 1 returns a point $\hat{x}_n$ such that:*

$$f(\hat{x}_n) - f(x_*) = \mathcal{O}\left(\frac{DG}{\sqrt{n+1}}\right),$$

as $n \to \infty$, where $D = \|x_0 - x_*\|$ for any x_* in the set of minimizers of f, as long as $d_0 \leq D$. We provide a proof in the Appendix.

The above result is asymptotic due to the potential of worst-case functions. For any fixed choice of n, a function could be constructed such that Algorithm 1 run for n steps has a dependence on d_0. The next theorem shows that even in the worst case this dependence only results in a $\log_2(D/d_0)$ worst rate of convergence, significantly better than the D/d_0 worse rate that sub-gradient-descent incurs:

Theorem 2 *Consider Algorithm 1 run for n steps, if we return the point $\hat{x}_t = \frac{1}{\sum_k^t d_k} \sum_{k=0}^{t} d_k x_k$ where t is chosen to be:*

$$t = \arg\min_{t \leq n} \frac{d_{t+1}}{\sum_{k=0}^{t} d_k},$$

Then:

$$f(\hat{x}_t) - f_* \leq 11 \frac{\log_2(D/d_0)}{n+1} D \sqrt{\sum_{k=0}^{t} \|g_k\|^2}.$$

4. Discussion

Our analysis applies to a very restricted problem setting of convex Lipschitz functions. In Carmon and Hinder [1], an approach for the same setting is extended to the stochastic setting in high probability. The same extension may also be applicable here.

Our approach has an undesirable dependence on the constant G, as it appears in the denominator of the step size. This dependence is typical for dual averaging methods, and approaches have been developed to remove the dependence [8]. In general, this dependence is mild in practice, as the G term only has a significant effect at the early stages of optimization, where it might dominate the sum in the denominator of the step size:

$$\gamma_{k+1} = \frac{1}{\sqrt{G^2 + \sum_{i=0}^{k} \|g_i\|^2}}.$$

Our algorithm requires an initial lower bound d_0 on D. The value of d_0 does not appear in the convergence rate bound as it's contribution goes to zero as $k \to \infty$, and hence is suppressed when big-$\mathcal{O}$ notation is used. In practice very small values can be used, as d_k will grow exponentially with k when d_0 is extremely small.

5. Conclusion

We have presented a simple approach to achieving parameter free learning of convex Lipshitz functions, by constructing succesively better lower bounds on the key unknown quantity: the distance to solution $\|x_0 - x_*\|$. Our approach for constructing these lower bounds may be of independent interest.

References

[1] Yair Carmon and Oliver Hinder. Making sgd parameter-free. Technical report, Tel Aviv University, 2022.

[2] Yoel Drori and Adrien B. Taylor. Efficient first-order methods for convex minimization: a constructive approach. *Mathematical Programming*, 2020.

[3] John Duchi, Elad Hazan, and Yoram Singer. Adaptive subgradient methods for online learning and stochastic optimization. *Journal of Machine Learning Research*, 12(61):2121–2159, 2011.

[4] Baptiste Goujaud, Adrien Taylor, and Aymeric Dieuleveut. Optimal first-order methods for convex functions with a quadratic upper bound. Technical report, INRIA, 2022.

[5] Elad Hazan and Sham M. Kakade. Revisiting the polyak step size. Technical report, Google AI Princeton, 2019.

[6] Yurii Nesterov. *Lectures on Convex Optimization*. Springer Nature, 2018.

[7] Francesco Orabona. A modern introduction to online learning. Technical report, Boston University, 2019.

[8] Francesco Orabona and David Pal. Scale-free online learning. Technical report, Yahoo Research, 2016.

[9] Francesco Orabona and Tatiana Tommasi. Training deep networks without learning rates through coin betting. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

[10] Boris T. Polyak. *Introduction to optimization*. Optimization Software, Inc., 1987.

[11] Rachel Ward, Xiaoxia Wu, and Léon Bottou. Adagrad stepsizes: Sharp convergence over nonconvex landscapes. *J. Mach. Learn. Res.*, 21(1), jan 2020. ISSN 1532-4435.

Appendix A. Theory

Let the weighting of the gradient in the s_k sum be λ_k:

$$s_{k+1} = s_k + \lambda_k g_k,$$

We use the theory technique of a phantom point

$$x'_k = x_0 - \gamma_* s_k,$$

using a Lyapunov function:

$$V_k = \frac{1}{2\gamma_*} \left\| x'_k - x_* \right\|^2.$$

A.1. Lemmas

Lemma 3 *(From Carmon and Hinder [1]) For any points x_0, x_T, (not necessarily iterates):*

$$\|x_0 - x_*\|^2 - \|x_T - x_*\|^2 \leq 2 \|x_0 - x_*\| \|x_0 - x_T\|.$$

Proof We apply a case analysis. Suppose that $\|x_T - x_*\| \geq \|x_0 - x_*\|$, then clearly $\|x_0 - x_*\|^2 - \|x_T - x_*\|^2$ is negative and thus trivially less than $2 \|x_0 - x_*\|^2 \|x_0 - x_T\|$.

So next consider the case that $\|x_0 - x_*\|^2 \geq \|x_T - x_*\|^2$. Then:

$$\begin{aligned}
&\|x_0 - x_*\|^2 - \|x_T - x_*\|^2 \\
&= (\|x_0 - x_*\| + \|x_T - x_*\|) (\|x_0 - x_*\| - \|x_T - x_*\|) \\
&\leq (\|x_0 - x_*\| + \|x_T - x_*\|) \|x_0 - x_T\| \\
&\leq 2 \|x_0 - x_*\| \|x_0 - x_T\|.
\end{aligned}$$

The first inequality is an application of the triangle inequality in the form $\|x_0 - x_*\| \leq \|x_T - x_*\| + \|x_0 - x_T\|$. The second inequality uses our case assumption. ■

Lemma 4 *The inner product $\gamma_k \lambda_k \langle g_k, s_k \rangle$ is a key quantity that occurs in our theory, where $s_{k+1} = s_k + \lambda_k g_k$ for some choice of λ_k. We can bound the sum of these inner products over time by considering the following expansion, where γ_k is any sequence of weights.*

$$-\sum_{k=0}^{n} \gamma_k \lambda_k \langle g_k, s_k \rangle = -\frac{\gamma_{n+1}}{2} \|s_{n+1}\|^2 + \sum_{k=0}^{n} \frac{\gamma_k}{2} \lambda_k^2 \|g_k\|^2 + \frac{1}{2} \sum_{k=0}^{n} (\gamma_{k+1} - \gamma_k) \|s_{k+1}\|^2.$$

This simplifies when the weighting sequence is flat:

$$-\gamma_{n+1} \sum_{k=0}^{n} \langle g_k, s_k \rangle = -\frac{\gamma_{n+1}}{2} \|s_{n+1}\|^2 + \frac{\gamma_{n+1}}{2} \sum_{k=0}^{n} \|g_k\|^2,$$

with λ weights:

$$-\gamma_{n+1} \sum_{k=0}^{n} \lambda_k \langle g_k, s_k \rangle = -\frac{\gamma_{n+1}}{2} \|s_{n+1}\|^2 + \frac{\gamma_{n+1}}{2} \sum_{k=0}^{n} \lambda_k^2 \|g_k\|^2.$$

Proof This is straight-forward to show by induction (it's a consequence of standard DA proof techniques, where $\|s_n\|^2$ is expanded).

$$\begin{aligned}\frac{\gamma_{n+1}}{2}\|s_{n+1}\|^2 &= \frac{\gamma_n}{2}\|s_{n+1}\|^2 + \frac{1}{2}(\gamma_{n+1}-\gamma_n)\|s_{n+1}\|^2 \\ &= \frac{\gamma_n}{2}\|s_n\|^2 + \gamma_n\lambda_n\langle g_n, s_n\rangle + \frac{\gamma_n}{2}\lambda_n^2\|g_n\|^2 + \frac{1}{2}(\gamma_{n+1}-\gamma_n)\|s_{n+1}\|^2 .\end{aligned}$$

Therefore

$$-\gamma_n\lambda_n\langle g_n, s_n\rangle = \frac{\gamma_n}{2}\|s_n\|^2 - \frac{\gamma_{n+1}}{2}\|s_{n+1}\|^2 + \frac{\gamma_n}{2}\lambda_n^2\|g_n\|^2 + \frac{1}{2}(\gamma_{n+1}-\gamma_n)\|s_{n+1}\|^2 .$$

Telescoping

$$-\sum_{k=0}^{n}\gamma_n\lambda_n\langle g_k, s_k\rangle = -\frac{\gamma_{n+1}}{2}\|s_{n+1}\|^2 + \sum_{k=0}^{n}\frac{\gamma_k}{2}\lambda_k^2\|g_k\|^2 + \frac{1}{2}\sum_{k=0}^{n}(\gamma_{k+1}-\gamma_k)\|s_{k+1}\|^2 .$$

■

A.2. Main Theory

Theorem 5 *For Algorithm 1, for all steps k it holds that:*

$$\lambda_k[f(x_k) - f_*] + V_{k+1} \le V_k + \lambda_k\langle g_k, x_k - x'_k\rangle + \frac{\gamma_*}{2}\lambda_k^2\|g_k\|^2 .$$

Proof The bound simply relies on convexity:

$$\begin{aligned}V_{k+1} &= \frac{1}{2\gamma_*}\|x'_{k+1} - x_*\|^2 \\ &= \frac{1}{2\gamma_*}\|x'_k - x_* - \gamma_*\lambda_k g_k\|^2 \\ &= \frac{1}{2\gamma_*}\|x'_k - x_*\|^2 - \lambda_k\langle g_k, x'_k - x_*\rangle + \frac{\gamma_*}{2}\lambda_k^2\|g_k\|^2 \\ &= V_k - \lambda_k\langle g_k, x'_k - x_k + x_k - x_*\rangle + \frac{\gamma_*}{2}\lambda_k^2\|g_k\|^2 \\ &= V_k - \lambda_k\langle g_k, x'_k - x_k\rangle - \lambda_k\langle g_k, x_k - x_*\rangle + \frac{\gamma_*}{2}\lambda_k^2\|g_k\|^2 \\ &\le V_k - \lambda_k\langle g_k, x'_k - x_k\rangle - \lambda_k[f(x_k) - f_*] + \frac{\gamma_*}{2}\lambda_k^2\|g_k\|^2 .\end{aligned}$$

■

Theorem 6 *Theorem 5 can be telescoped and simplified to give:*

$$\sum_{k=0}^{n}\lambda_k(f(x_k) - f_*) \le \|x_0 - x_*\|\,\|s_{n+1}\| + \sum_{k=0}^{n}\frac{\gamma_k}{2}\lambda_k^2\|g_k\|^2 - \frac{\gamma_{n+1}}{4}\|s_{n+1}\|^2 .$$

Proof Plugging in the phantom step size:

$$\begin{aligned}-\lambda_k \left\langle g_k, x'_k - x_k \right\rangle &= -\lambda_k \left\langle g_k, x_0 - \gamma_* s_k - x_0 + \gamma_k s_k \right\rangle \\ &= -\lambda_k \left(\gamma_k - \gamma_*\right) \left\langle g_k, s_k \right\rangle .\end{aligned}$$

So:

$$\lambda_k \left[f(x_k) - f_*\right] + V_{k+1} \leq V_k - \lambda_k \left(\gamma_k - \gamma_*\right) \left\langle g_k, s_k \right\rangle + \frac{\gamma_*}{2} \lambda_k^2 \left\|g_k\right\|^2 .$$

Then telescoping:

$$V_{k+1} \leq V_0 - \sum_k^n \gamma_k \lambda_k \left\langle g_k, s_k \right\rangle + \gamma_* \sum_{k=0}^n \lambda_k \left\langle g_k, s_k \right\rangle + \frac{\gamma_*}{2} \lambda_k \sum_{k=0}^n \left\|g_k\right\|^2 .$$

Therefore:

$$\begin{aligned}\sum_{k=0}^n \lambda_k \left(f(x_k) - f_*\right) + \frac{1}{2\gamma_*} \left\|x'_{n+1} - x_*\right\|^2 \leq \frac{1}{2\gamma_*} \left\|x_0 - x_*\right\|^2 - \sum_k^n \gamma_k \lambda_k \left\langle g_k, s_k \right\rangle \\ + \gamma_* \sum_{k=0}^n \lambda_k \left\langle g_k, s_k \right\rangle + \frac{\gamma_*}{2} \sum_{k=0}^n \lambda_k^2 \left\|g_k\right\|^2 .\end{aligned}$$

We now apply Lemmas 3 as follows:

$$\begin{aligned}\frac{1}{2\gamma_*} \left\|x_0 - x_*\right\|^2 - \frac{1}{2\gamma_*} \left\|x'_{n+1} - x_*\right\|^2 &\leq \frac{1}{\gamma_*} \left\|x_0 - x_*\right\| \left\|x_0 - x'_{n+1}\right\| \\ &\leq \left\|x_0 - x_*\right\| \left\|s_{n+1}\right\| ,\end{aligned}$$

and 4 to simplify

$$\gamma^* \sum_{k=0}^n \lambda_k \left\langle g_k, s_k \right\rangle = \frac{\gamma^*}{2} \left\|s_{n+1}\right\|^2 - \frac{\gamma^*}{2} \sum_{k=0}^n \lambda_k^2 \left\|g_k\right\|^2 .$$

Combining gives:

$$\sum_{k=0}^n \lambda_k \left(f(x_k) - f_*\right) \leq \left\|x_0 - x_*\right\| \left\|s_{n+1}\right\| - \sum_k^n \gamma_k \lambda_k \left\langle g_k, s_k \right\rangle + \frac{\gamma^*}{2} \left\|s_{n+1}\right\|^2$$

We can further simplify with:

$$-\sum_{k=0}^n \gamma_k \lambda_k \left\langle g_k, s_k \right\rangle = -\frac{\gamma_{n+1}}{2} \left\|s_{n+1}\right\|^2 + \sum_{k=0}^n \frac{\gamma_k}{2} \lambda_k^2 \left\|g_k\right\|^2 + \frac{1}{2} \sum_{k=0}^n \left(\gamma_{k+1} - \gamma_k\right) \left\|s_{k+1}\right\|^2$$

Using the fact that $\gamma_{k+1} - \gamma_k \leq 0$ and that $\gamma_* = \gamma_{n+1}/2$ we have:

$$\begin{aligned}\sum_{k=0}^n \lambda_k \left(f(x_k) - f_*\right) &\leq \left\|x_0 - x_*\right\| \left\|s_{n+1}\right\| - \sum_k^n \gamma_k \lambda_k \left\langle g_k, s_k \right\rangle + \frac{\gamma^*}{2} \left\|s_{n+1}\right\|^2 \\ &= \left\|x_0 - x_*\right\| \left\|s_{n+1}\right\| - \sum_k^n \gamma_k \lambda_k \left\langle g_k, s_k \right\rangle + \frac{\gamma_{n+1}}{4} \left\|s_{n+1}\right\|^2 \\ &\leq \left\|x_0 - x_*\right\| \left\|s_{n+1}\right\| + \sum_{k=0}^n \frac{\gamma_k}{2} \lambda_k^2 \left\|g_k\right\|^2 - \frac{\gamma_{n+1}}{4} \left\|s_{n+1}\right\|^2\end{aligned}$$

■

Theorem 7 *The distance to solution error term can be lower bounded as follows*

$$D \geq \hat{d}_{n+1} = \frac{\frac{\gamma_{n+1}}{2} \|s_{n+1}\|^2 - \sum_{k=0}^{n} \gamma_k \lambda_k^2 \|g_k\|^2}{2 \|s_{n+1}\|}.$$

Proof The key idea is that the bound:

$$\sum_{k=0}^{n} \lambda_k \left(f(x_k) - f_*\right) \leq D \|s_{n+1}\| + \sum_{k=0}^{n} \frac{\gamma_k}{2} \lambda_k^2 \|g_k\|^2 - \frac{\gamma_{n+1}}{4} \|s_{n+1}\|^2,$$

gives some indication as to the magnitude of D in the case when the other terms on the right are negative. To proceed, we use $\sum_{k=0}^{n} \lambda_k \left(f(x_k) - f_*\right) \geq 0$, giving:

$$0 \leq D \|s_{n+1}\| + \sum_{k=0}^{n} \frac{\gamma_k}{2} \lambda_k^2 \|g_k\|^2 - \frac{\gamma_{n+1}}{4} \|s_{n+1}\|^2,$$

which we can rearrange to:

$$D \|s_{n+1}\| \geq \frac{\gamma_{n+1}}{4} \|s_{n+1}\|^2 - \sum_{k=0}^{n} \frac{\gamma_k}{2} \lambda_k^2 \|g_k\|^2.$$

Therefore:

$$D \geq \frac{\frac{\gamma_{n+1}}{4} \|s_{n+1}\|^2 - \sum_{k=0}^{n} \frac{\gamma_k}{2} \lambda_k^2 \|g_k\|^2}{\|s_{n+1}\|}.$$

■

Theorem 8 *The norm of s_{n+1} is bounded as:*

$$\|s_{n+1}\| \leq \frac{8 d_{n+1}}{\gamma_{n+1}} + \frac{2\sqrt{\sum_{k=0}^{n} \frac{\gamma_k}{2} \lambda_k^2 \|g_k\|^2}}{\sqrt{\gamma_{n+1}}}.$$

Proof By the definition of $\hat{d}_{n+1}$ as used in Theorem 7, we have:

$$\hat{d}_{n+1} \|s_{n+1}\| = \frac{\gamma_{n+1}}{4} \|s_{n+1}\|^2 - \sum_{k=0}^{n} \frac{\gamma_k}{2} \lambda_k^2 \|g_k\|^2,$$

and since $2 d_{n+1} \geq \hat{d}_{n+1}$,

$$d_{n+1} \|s_{n+1}\| \geq \frac{1}{2} \hat{d}_{n+1} \|s_{n+1}\| = \frac{1}{2} \left[\frac{\gamma_{n+1}}{4} \|s_{n+1}\|^2 - \sum_{k=0}^{n} \frac{\gamma_k}{2} \lambda_k^2 \|g_k\|^2 \right].$$

So:

$$2 d_{n+1} \|s_{n+1}\| - \frac{\gamma_{n+1}}{4} \|s_{n+1}\|^2 + \sum_{k=0}^{n} \frac{\gamma_k}{2} \lambda_k^2 \|g_k\|^2 \geq 0.$$

This is a quadratic equation in $\|s_{n+1}\|$ that we can solve explicitly. We have equality when:

$$\|s_{n+1}\| = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a},$$

where:

$$a = -\frac{\gamma_{n+1}}{4},$$
$$b = 2d_{n+1},$$
$$c = \sum_{k=0}^{n} \frac{\gamma_k}{2} \lambda_k^2 \|g_k\|^2 .$$

So we have equality at the value:

$$\|s_{n+1}\| = \frac{-2d_{n+1} \pm \sqrt{4d_{n+1}^2 + 4\frac{\gamma_{n+1}}{4} \sum_{k=0}^{n} \frac{\gamma_k}{2} \lambda_k^2 \|g_k\|^2}}{-\frac{\gamma_{n+1}}{2}},$$

$$\therefore \|s_{n+1}\| = \frac{4d_{n+1} \pm 2\sqrt{4d_{n+1}^2 + \gamma_{n+1} \sum_{k=0}^{n} \frac{\gamma_k}{2} \lambda_k^2 \|g_k\|^2}}{\gamma_{n+1}}.$$

By examination of the cases, we see that that the + case provides an upper bound:

$$\|s_{n+1}\| \leq \frac{4d_{n+1} + 2\sqrt{4d_{n+1}^2 + \gamma_{n+1} \sum_{k=0}^{n} \frac{\gamma_k}{2} \lambda_k^2 \|g_k\|^2}}{\gamma_{n+1}}.$$

We can further simplify this bound using subadditivity:

$$\|s_{n+1}\| \leq \frac{4d_{n+1} + 4d_{n+1} + 2\sqrt{\gamma_{n+1} \sum_{k=0}^{n} \frac{\gamma_k}{2} \lambda_k^2 \|g_k\|^2}}{\gamma_{n+1}},$$

therefore:

$$\|s_{n+1}\| \leq \frac{8d_{n+1}}{\gamma_{n+1}} + \frac{2\sqrt{\sum_{k=0}^{n} \frac{\gamma_k}{2} \lambda_k^2 \|g_k\|^2}}{\sqrt{\gamma_{n+1}}}.$$

■

Proposition 9 *(From Duchi et al. [3]) The gradient error term can be bounded as:*

$$\sum_{k=0}^{n} \frac{\|g_k\|^2}{\sqrt{G^2 + \sum_{i=0}^{k-1} \|g_i\|^2}} \leq 2\sqrt{\sum_{k}^{n} \|g_k\|^2},$$

and therefore:

$$\sum_{k=0}^{n} \frac{\gamma_k}{2} \|g_k\|^2 \leq \gamma_{n+1} \left(G^2 + \sum_{k}^{n} \|g_k\|^2 \right).$$

Appendix B. Putting it together

Theorem 10 *For Algorithm 1:*

$$\sum_{k=0}^{n} d_k \left(f(x_k) - f_*\right) \leq 8Dd_{n+1}\sqrt{\sum_{k=0}^{n} \|g_k\|^2} + \frac{2D\sqrt{\sum_{k=0}^{n} \frac{\gamma_k}{2}\lambda_k^2 \|g_k\|^2}}{\sqrt{\gamma_{n+1}}} + \sum_{k=0}^{n} \frac{\gamma_k}{2}\lambda_k^2 \|g_k\|^2 .$$

Proof Consider the key bound:

$$\sum_{k=0}^{n} \lambda_k \left(f(x_k) - f_*\right) \leq D\|s_{n+1}\| - \frac{\gamma_{n+1}}{4}\|s_{n+1}\|^2 + \sum_{k=0}^{n} \frac{\gamma_k}{2}\lambda_k^2 \|g_k, \|^2 ,$$

applying the bound from Theorem 8:

$$\|s_{n+1}\| \leq \frac{8d_{n+1}}{\gamma_{n+1}} + \frac{2\sqrt{\sum_{k=0}^{n} \frac{\gamma_k}{2}\lambda_k^2 \|g_k\|^2}}{\sqrt{\gamma_{n+1}}},$$

gives:

$$\begin{aligned}\sum_{k=0}^{n} \lambda_k \left(f(x_k) - f_*\right) \leq \frac{8Dd_{n+1}}{\gamma_{n+1}} + \frac{2D\sqrt{\sum_{k=0}^{n} \frac{\gamma_k}{2}\lambda_k^2 \|g_k\|^2}}{\sqrt{\gamma_{n+1}}} \\ - \frac{\gamma_{n+1}}{4}\|s_{n+1}\|^2 + \sum_{k=0}^{n} \frac{\gamma_k}{2}\lambda_k^2 \|g_k\|^2 .\end{aligned}$$

Now using $\lambda_k = d_k$, plugging in the step size, and dropping the $-\frac{\gamma_{n+1}}{4}\|s_{n+1}\|^2$ term:

$$\sum_{k=0}^{n} d_k \left(f(x_k) - f_*\right) \leq 8Dd_{n+1}\sqrt{\sum_{k=0}^{n} \|g_k\|^2} + \frac{2D\sqrt{\sum_{k=0}^{n} \frac{\gamma_k}{2}\lambda_k^2 \|g_k\|^2}}{\sqrt{\gamma_{n+1}}} + \sum_{k=0}^{n} \frac{\gamma_k}{2}\lambda_k^2 \|g_k\|^2 .$$

■

Theorem 11 *For the point returned by Algorithm 1, as $n \to \infty$:*

$$f(\hat{x}_n) - f_* = \mathcal{O}\left(\frac{DG}{\sqrt{n+1}}\right).$$

Proof Since the sequence d_k changes by doubling, and is upper bounded by D, there must exist some finite $\hat{n}$ such that after $\hat{n}$ steps, d_k no longer increases, i.e. $d_n = d_{\hat{n}}$ for all $n \geq \hat{n}$. Suppose that $n \geq 2\hat{n}$. Then we have that:

$$\sum_{k=0}^{n} d_k \geq \frac{1}{2}(n+1)d_{n+1},$$

$$\therefore \frac{1}{\sum_{k=0}^{n} d_k} \leq \frac{2}{(n+1)d_{n+1}}.$$

Then:

$$\frac{1}{\sum_{k=0}^{n} d_k} \sum_{k=0}^{n} d_k \left(f(x_k) - f_*\right) \leq \frac{16D}{(n+1)} \sqrt{\sum_{k=0}^{n} \|g_k\|^2} + \frac{4D\sqrt{\sum_{k=0}^{n} \frac{\gamma_k}{2} \|g_k\|^2}}{(n+1)\sqrt{\gamma_{n+1}}} + \frac{2\gamma_{n+1} d_{n+1} \sum_{k}^{n} \|g_k\|^2}{n+1}.$$

Then using $\sum_{k=0}^{n} \frac{\gamma_k}{2} \|g_k\|^2 \leq \gamma_{n+1} G^2 (n+2)$ to simplify further we get:

$$\frac{1}{\sum_{k=0}^{n} d_k} \sum_{k=0}^{n} d_k \left(f(x_k) - f_*\right) \leq \frac{16D}{(n+1)} \sqrt{\sum_{k=0}^{n} \|g_k\|^2} + \frac{4D}{(n+1)} \sqrt{G^2 + \sum_{k}^{n} \|g_k\|^2} + \frac{2D}{n+1} \sqrt{\sum_{k=0}^{n} \|g_k\|^2}.$$

So

$$\frac{1}{\sum_{k=0}^{n} d_k} \sum_{k=0}^{n} d_k \left(f(x_k) - f_*\right) \leq \frac{22D}{n+1} \sqrt{G^2 + \sum_{k}^{n} \|g_k\|^2}.$$

We can convert to a bound on the average iterate:

$$\hat{x}_n = \frac{1}{\sum_{k}^{n} d_k} \sum_{k=0}^{n} d_k x_k,$$

via Jensen's inequality. Using $\|g_k\|^2 \leq G^2$ and simplifying gives the result. ■

Appendix C. Non-asymptotic analysis

Lemma 12 *Consider a sequence $d_0, \ldots d_{N+1}$, where for each k, $d_{k+1} = d_k$ or $d_{k+1} \geq 2d_k$. Let $r \leq \log_2(d_N/d_0)$ be the number of steps for which $d_{k+1} \geq 2d_k$. Then*

$$\min_{n \leq N} \frac{d_{n+1}}{\sum_{k=0}^{n} d_k} \leq \frac{\log_2(d_{N+1}/d_0)}{N+1}.$$

Proof We proceed by an inductive argument on r. In the base case, if $r = 0$ then the result follows immediately:

$$\min_{n \leq N} \frac{d_{n+1}}{\sum_{k=0}^{n} d_k} = \frac{d_{N+1}}{\sum_{k=0}^{N} d_k} = \frac{d_{N+1}}{(N+1)d_{N+1}} = \frac{1}{N+1} = \frac{\log_2(d_{N+1}/d_0)}{N+1}.$$

So assume that $r > 0$. First we show that no induction is needed if for all, and we may take $n = N$, if

$$d_k = d_{n+1}, \quad \text{for } k \geq \left\lfloor n + 1 - \frac{(n+1)}{\log_2(d_{n+1}/d_0)} \right\rfloor.$$

Since, in that case we have:

$$\begin{aligned}\sum_{k=0}^{n} d_k &\geq \sum_{k=\lfloor n+1-(n+1)/\log_2(d_{n+1}/d_0)\rfloor}^{n} d_k = \left(n + 1 - \left\lfloor n + 1 - \frac{(n+1)}{\log_2(d_n/d_0)} \right\rfloor\right) d_{n+1} \\ &\geq \frac{(n+1)\, d_{n+1}}{\log_2(d_{n+1}/d_0)}.\end{aligned}$$

Therefore:

$$\frac{d_{n+1}}{\sum_{k=0}^{n} d_k} \leq \frac{d_{n+1}}{(n+1)d_{n+1}} \log_2(d_{n+1}/d_0) = \frac{\log(d_{n+1}/d_0)}{n+1}.$$

So, instead suppose that there is at least one increase within the range $k \geq \left\lfloor N + 1 - \frac{(N+1)}{\log_2(d_{N+1}/d_0)} \right\rfloor + 1$. Note that +1 is due to the fact that the above case includes the edge case where an increase occurs exactly at the beginning of the interval. That implies that there are at most $r - 1$ increases in the range $k \leq \left\lfloor N + 1 - \frac{(N+1)}{\log_2(d_{N+1}/d_0)} \right\rfloor$, therefore, we can apply induction, assuming by induction that:

$$\min_{n \leq n'} \frac{d_{n+1}}{\sum_{k=0}^{n} d_k} \leq \frac{\log_2(d_{n'+1}/d_0)}{n'+1}, \quad \text{for } n' = \left\lfloor N + 1 - \frac{(N+1)}{\log_2(d_{N+1}/d_0)} \right\rfloor.$$

Under this inductive hypothesis assumption, we note that:

$$\begin{aligned}\frac{\log_2(d_{n'+1}/d_0)}{n'+1} &\leq \frac{1}{\left\lfloor N + 1 - \frac{(N+1)}{\log_2(d_{N+1}/d_0)} \right\rfloor + 1} \log_2(d_{n'+1}/d_0) \\ &\leq \frac{1}{N - \frac{(N+1)}{\log_2(d_{N+1}/d_0)} + 1} \log_2(d_{n'+1}/d_0) \\ &= \frac{\log_2(d_{N+1}/d_0)}{(N+1)\left(\log_2(d_{N+1}/d_0) - 1\right)} \log_2(d_{n'+1}/d_0) \\ &= \frac{\log_2(d_{N+1}/d_0)}{(N+1)} \cdot \frac{\log_2(d_{n'+1}/d_0)}{\log(d_{N+1}/d_0) - 1} \\ &\leq \frac{\log_2(d_{N+1}/d_0)}{(N+1)}\end{aligned}$$

the last inequality follows from $d_{k+1} \geq 2d_k$, as it implies that:

$$\log_2(d_{n'+1}/d_0) \leq \log(d_{N+1}/d_0) - 1.$$

Putting it all together, we have that:

$$\min_{n \leq N} \frac{d_{n+1}}{\sum_{k=0}^{n} d_k} \leq \left[\frac{d_{n+1}}{\sum_{k=0}^{n} d_k}\right]_{n=N-\frac{(N+1)}{\log_2(d_N/d_0)}} \leq \frac{\log_2(d_{N+1}/d_0)}{N+1}.$$

■

Theorem 13 *Consider Algorithm 1 run for n steps, if we return the point $\hat{x}_t = \frac{1}{\sum_k^t d_k}\sum_{k=0}^t d_k x_k$ where t is chosen to be:*

$$t = \arg\min_{t \leq n} \frac{d_{t+1}}{\sum_{k=0}^t d_k},$$

Then:

$$f(\hat{x}_t) - f_* \leq 11\frac{\log_2(D/d_0)}{n+1} D\sqrt{\sum_{k=0}^t \|g_k\|^2}.$$

Proof Consider the bound from Theorem 10:

$$\begin{aligned}\frac{1}{\sum_{k=0}^n d_k}\sum_{k=0}^n d_k\left(f(x_k) - f_*\right) \leq \frac{d_{n+1}}{\sum_{k=0}^n d_k} 8D\sqrt{\sum_{k=0}^n \|g_k\|^2} \\ + \frac{1}{\sum_{k=0}^n d_k}\left[\frac{2D\sqrt{\sum_{k=0}^n \frac{\gamma_k}{2} d_k^2 \|g_k\|^2}}{\sqrt{\gamma_{n+1}}}\right] \\ + \frac{1}{\sum_{k=0}^n d_k}\sum_{k=0}^n \frac{\gamma_k}{2} d_k^2 \|g_k\|^2.\end{aligned}$$

For the middle term, we can simplify via:

$$\sqrt{\sum_{k=0}^n \frac{\gamma_k}{2} d_k^2 \|g_k\|^2} \leq \sqrt{d_{n+1}^2 \sum_{k=0}^n \frac{\gamma_k}{2}\|g_k\|^2} = d_{n+1}\sqrt{\sum_{k=0}^n \frac{\gamma_k}{2}\|g_k\|^2}.$$

For the third term:

$$\frac{1}{\sum_{k=0}^n d_k}\sum_{k=0}^n \frac{\gamma_k}{2} d_k^2 \|g_k\|^2 \leq \frac{D d_{n+1}}{\sum_{k=0}^n d_k}\sum_{k=0}^n \frac{\gamma_k}{2}\|g_k\|^2.$$

So we have:

$$f(\hat{x}_n) - f_* \leq \frac{d_{n+1} D}{\sum_{k=0}^n d_k}\left[8\sqrt{\sum_{k=0}^n \|g_k\|^2} + \frac{\sqrt{2\sum_{k=0}^n \gamma_k \|g_k\|^2}}{\sqrt{\gamma_{n+1}}} + \frac{1}{2}\sum_{k=0}^n \gamma_k \|g_k\|^2\right].$$

Now using Lemma 12, we can return the point $\hat{x}_t$ and at time t at which $\min_{n \leq N} \frac{d_{n+1}}{\sum_{k=0}^n d_k}$ is smallest, ensuring that:

$$\frac{d_{t+1}}{\sum_{k=0}^t d_k} = \min_{n \leq N} \frac{d_{n+1}}{\sum_{k=0}^n d_k} \leq \frac{\log_2(d_{N+1}/d_0)}{N+1},$$

Giving us an upper bound:

$$f(\hat{x}_t) - f_* \leq \frac{\log_2(d_{N+1}/d_0)}{N+1} D\left[8\sqrt{\sum_{k=0}^t \|g_k\|^2} + \frac{\sqrt{2\sum_{k=0}^t \gamma_k \|g_k\|^2}}{\sqrt{\gamma_{n+1}}} + \frac{1}{2}\sum_{k=0}^t \gamma_k \|g_k\|^2\right].$$

We can further simplify using Proposition 9:

$$f(\hat{x}_t) - f_* \leq \frac{\log_2(d_{N+1}/d_0)}{N+1} D \left[8\sqrt{\sum_{k=0}^{t} \|g_k\|^2} + \sqrt{2\sum_{k=0}^{t} \|g_k\|^2} + \sqrt{\sum_{k=0}^{t} \|g_k\|^2} \right],$$

$$f(\hat{x}_t) - f_* \leq 11 \frac{\log_2(D/d_0)}{N+1} D \sqrt{\sum_{k=0}^{t} \|g_k\|^2}.$$

■