

Provably Faster Gradient Descent via Long Steps

Benjamin Grimmer*

Abstract

This work establishes new convergence guarantees for gradient descent in smooth convex optimization via a computer-assisted analysis technique. Our theory allows nonconstant stepsize policies with frequent long steps potentially violating descent by analyzing the overall effect of many iterations at once rather than the typical one-iteration inductions used in most first-order method analyses. We show that long steps, which may increase the objective value in the short term, lead to provably faster convergence in the long term. A conjecture towards proving a faster $O(1/T \log T)$ rate for gradient descent is also motivated along with simple numerical validation.

1 Introduction

This work proposes a new analysis technique for gradient descent, establishing a path toward provably better convergence guarantees for smooth, convex optimization than is possible with existing approaches based on constant stepsizes. Instead, our theory allows for nonconstant stepsize policies, periodically taking larger steps that may violate the monotone decrease in objective value typically needed by analysis. In fact, contrary to the common intuition, we show periodic long steps, which may increase the objective value in the short term, provably speed up convergence in the long term, with increasingly large gains as longer and longer steps are periodically included. This bears a similarity to accelerated momentum methods, which also depart from ensuring a monotone objective decrease at every iteration.

Establishing this requires a proof technique capable of analyzing the overall effect of many iterations at once rather than the typical (naive) one-iteration inductions used in most first-order method analyses. Our proofs are based on the Performance Estimation Problem (PEP) ideas of [1–3], which cast computing/bounding the worst-case problem instance of a given algorithm as a Semidefinite Program (SDP). We show that the existence of a feasible solution to a related SDP proves a descent guarantee after applying a corresponding pattern of nonconstant stepsizes, from which faster convergence guarantees follow. Our technique is very similar to that first proposed by Altschuler’s Master’s thesis [4, Chapter 8] which established repeating stepsize patterns of length two or three with faster contractions towards the minimizer for smooth, strongly convex minimization. Most existing PEP literature uses computer-solves to guide the search for tighter convergence proofs [3, 5–13] and inform the development of new, provably faster algorithms [14–18]. Here computer outputs are directly used to constitute the proof, but are large (up to tens of thousands of rational numbers) and so may provide less guidance or intuition.

We consider gradient descent with a sequence of (normalized) stepsizes $h = (h_0, h_1, h_2, \dots)$ applied to minimize a convex function $f: \mathbb{R}^n \rightarrow \mathbb{R}$ with L -Lipschitz gradient by iterating

$$x_{k+1} = x_k - \frac{h_k}{L} \nabla f(x_k) \tag{1.1}$$

*Johns Hopkins University, Department of Applied Mathematics and Statistics, grimmer@jhu.edu

given an initialization $x_0 \in \mathbb{R}^n$. We assume throughout that a minimizer x_* of f exists and its level sets are bounded $D = \sup\{\|x - x_*\|_2 \mid f(x) \leq f(x_0)\}$ ¹. The classic convergence guarantee [19] for gradient descent is that with constant stepsizes $h = (1, 1, 1, \dots)$, every $T > 0$ has $f(x_T) - f(x_*) \leq LD^2/2(T + 1)$. Using performance estimation techniques, a tight refinement of this bound was given by [1, Theorem 3.1] (and elementary proof, avoiding the use of PEP, was given by [20]): every $T > 0$ has

$$f(x_T) - f(x_*) \leq \frac{LD^2}{4T + 2}. \quad (1.2)$$

By tight, we mean a matching problem instance exists that attains the above inequality. Using stepsizes between one and two, this rate can be improved by another factor of two (see [2, 20]). Stepsizes beyond length two have very little prior theory as one can no longer guarantee a decrease in objective value at each iteration.

Here we provide an analysis technique capable of handling nonconstant stepsizes, periodically longer than one can guarantee descent for, finding increasingly fast convergence (in terms of constants) follows. For example, consider gradient descent alternating stepsizes $h = (2.9, 1.5, 2.9, 1.5, \dots)$. Such a scheme is beyond the reach of traditional descent-based analysis as one may fear the stepsizes of 2.9 can increase the function value individually more than the 1.5 is guaranteed to decrease it. Regardless, we show this “long step” converges with

$$f(x_T) - f(x_*) \leq \frac{LD^2}{2.2 \times T} + O(1/T^2)$$

for every even $T > 0$. See our Theorem 3.2 characterizing many such alternating stepsize methods. The $+O(1/T^2)$ terms throughout this work are only used to suppress two universal constants, namely above, we show there exist constants \bar{s} and C such that all even $T > 2\bar{s}$ have bound $LD^2/(2.2 \times T - C)$.

Using longer cycles, we derive further performance gains. For example, we show a carefully selected stepsize pattern of length 127 periodically taking stepsizes of 370.0 converges at a rate of $LD^2/(5.8346303 \times T)$. Generally, given a stepsize pattern $h = (h_0, \dots, h_{t-1}) \in \mathbb{R}^t$, we consider the gradient descent method repeatedly applying the pattern of stepsizes

$$x_{k+1} = x_k - \frac{h_{(k \bmod t)}}{L} \nabla f(x_k). \quad (1.3)$$

In Theorem 2.1, we give a convergence guarantee for any *straightforward* stepsize pattern h of

$$f(x_T) - f(x_*) \leq \frac{LD^2}{\text{avg}(h)T} + O(1/T^2) \quad (1.4)$$

where $\text{avg}(h) = \frac{1}{t} \sum_{i=0}^{t-1} h_i$. See Section 2 for the formal introduction of this straightforwardness property. Hence the design of provably faster nonconstant stepsize gradient descent methods amounts to seeking straightforward stepsize patterns with large average stepsize values. Certifying a given pattern is straightforward can be done via semidefinite programming (see our Theorem 3.1). So the convergence analysis of such nonconstant stepsizes methods is a natural candidate for computer assistance.

Table 1 shows straightforward stepsize patterns with increasingly fast convergence guarantees, each proven using a computer-generated, exact-arithmetic semidefinite programming solution certificate. Future works identifying longer straightforward patterns and other tractable families

¹This assumption can likely be relaxed but eases our development herein.

Pattern Length	“Straightforward” Stepsize Pattern h (longest stepsize marked in bold)	Convergence Rate ($+O(1/T^2)$ omitted)
$t = 2$	$(\mathbf{3} - \eta, 1.5)$ for any $\eta \in (0, 3)$	$\frac{LD^2}{(2.25 - \eta/2) \times T}$
$t = 3$	$(1.5, \mathbf{4.9}, 1.5)$	$\frac{2.63333\dots \times T}{LD^2}$
$t = 7$	$(1.5, 2.2, 1.5, \mathbf{12.0}, 1.5, 2.2, 1.5)$	$\frac{3.1999999 \times T}{LD^2}$
$t = 15$	$(1.4, 2.0, 1.4, 4.5, 1.4, 2.0, 1.4, \mathbf{29.7}, 1.4, 2.0, 1.4, 4.5, 1.4, 2.0, 1.4)$	$\frac{3.8599999 \times T}{LD^2}$
$t = 31$	$(1.4, 2.0, 1.4, 3.9, 1.4, 2.0, 1.4, 8.2, 1.4, 2.0, 1.4, 3.9, 1.4, 2.0, 1.4, \mathbf{72.3}, 1.4, 2.0, 1.4, 3.9, 1.4, 2.0, 1.4, 8.2, 1.4, 2.0, 1.4, 3.9, 1.4, 2.0, 1.4)$	$\frac{LD^2}{4.6032258 \times T}$
$t = 63$	$(1.4, 2.0, 1.4, 3.9, 1.4, 2.0, 1.4, 7.2, 1.4, 2.0, 1.4, 3.9, 1.4, 2.0, 1.4, 14.2, 1.4, 2.0, 1.4, 3.9, 1.4, 2.0, 1.4, 7.2, 1.4, 2.0, 1.4, 3.9, 1.4, 2.0, 1.4, \mathbf{164.0}, 1.4, 2.0, 1.4, 3.9, 1.4, 2.0, 1.4, 7.2, 1.4, 2.0, 1.4, 3.9, 1.4, 2.0, 1.4, 14.2, 1.4, 2.0, 1.4, 3.9, 1.4, 2.0, 1.4, 7.2, 1.4, 2.0, 1.4, 3.9, 1.4, 2.0, 1.4)$	$\frac{LD^2}{5.2253968 \times T}$
$t = 127$	$(1.4, 2.0, 1.4, 3.9, 1.4, 2.0, 1.4, 7.2, 1.4, 2.0, 1.4, 3.9, 1.4, 2.0, 1.4, 12.6, 1.4, 2.0, 1.4, 3.9, 1.4, 2.0, 1.4, 7.2, 1.4, 2.0, 1.4, 3.9, 1.4, 2.0, 1.4, 23.5, 1.4, 2.0, 1.4, 3.9, 1.4, 2.0, 1.4, 7.2, 1.4, 2.0, 1.4, 3.9, 1.4, 2.0, 1.4, 12.6, 1.4, 2.0, 1.4, 3.9, 1.4, 2.0, 1.4, 7.2, 1.4, 2.0, 1.4, 3.9, 1.4, 2.0, 1.4, \mathbf{370.0}, 1.4, 2.0, 1.4, 3.9, 1.4, 2.0, 1.4, 7.2, 1.4, 2.0, 1.4, 3.9, 1.4, 2.0, 1.4, 12.6, 1.4, 2.0, 1.4, 3.9, 1.4, 2.0, 1.4, 7.2, 1.4, 2.0, 1.4, 3.9, 1.4, 2.0, 1.4, 23.5, 1.4, 2.0, 1.4, 3.9, 1.4, 2.0, 1.4, 7.2, 1.4, 2.0, 1.4, 3.9, 1.4, 2.0, 1.4, 12.6, 1.4, 2.0, 1.4, 3.9, 1.4, 2.0, 1.4, 7.2, 1.4, 2.0, 1.4, 3.9, 1.4, 2.0, 1.4)$	$\frac{LD^2}{5.8346303 \times T}$

Table 1: Improved convergence guarantees for Gradient Descent with stepsizes cycling through a “straightforward” pattern. Each convergence bound is proven by producing a certificate of feasibility for a related SDP, which is sufficient by our Theorems 2.1 and 3.1. Coefficients for $t \geq 7$ guarantees are slightly smaller than the ideal $\text{avg}(h)$ due to rounding to produce an exact arithmetic certificate.

of nonconstant, periodically long stepsize policies will surely be able to improve on this work’s particular guarantees.

The analysis of such nonconstant, long stepsize gradient descent methods has eluded the literature, with only a few exceptions. In 1953, Young [21] showed optimal, accelerated convergence is possible for gradient descent when minimizing a smooth, strongly convex quadratic function by using a careful nonconstant selection of h_i . Namely, Young set $h_0 \dots h_{T-1}$ as one over the roots of the T -degree Chebyshev polynomial². Few works have shown faster convergence from long steps beyond quadratics. Two notable works have done so for smooth, strongly convex minimization:

Oymak [23] showed substantial speed-ups for strongly convex functions with special bimodal structured Hessians. Closely in line with this work’s reasoning, several recent Master’s and doctoral theses have addressed the optimal design and analysis of one, two, or three steps of gradient descent. The theses of Daccache [24] and Eloi [25] provide exhaustive characterizations of gradient descent’s objective gap after two or three steps for smooth convex optimization. Their results do not provide a mechanism to be applied inductively, so no convergence rates follow from repeatedly applying the studied patterns. For strongly convex optimization, Altschuler [4, Chapter 8] used PEP techniques to derive the optimal patterns of length $t = 1, 2, 3$ for contracting either the distance to optimal or the gradient’s norm. Inductively applying their improved contractions, one arrives at the same moral takeaway advanced by this work: SDP-based analysis can prove faster convergence follows from periodically taking longer steps, potentially violating descent. Importantly, their length two and three patterns are optimal, meaning their patterns give the best possible contraction factor. Note their patterns differ from those presented here, only being optimal for strongly convex problems. *The primary contribution in this work is identifying a tractable analysis technique for general smooth, convex optimization and showing increasing performance gains for increasingly large $t > 3$, continuing in this seventy-year-old direction.*

The search for long, straightforward stepsize patterns h is hard; the set of all straightforward patterns is nonconvex, making local searches often unfruitful. Our patterns of length $t = 2^m - 1$ in Table 1 were created by repeating the pattern for $t = 2^{m-1} - 1$ twice with a new long step added in between and (by hand) shrinking the long steps in the length $2^{m-1} - 1$ subpatterns. This recursive pattern has strong similarities to the cyclic and fractal Chebyshev patterns for quadratic minimization considered by [26–28], although we make no provable connections. This doubling procedure consistently increased $\text{avg}(h)$ by ≈ 0.6 . We conjecture the following.

Conjecture 1.1. *For any $t \in \mathbb{N}$, there exists a straightforward stepsize pattern $h \in \mathbb{R}^t$ with $\text{avg}(h) = \Omega(\log(t))$.*

If true, this would likely yield convergence rates on the order of $O(1/(T \log(T)))$, strictly improving on the classic $O(1/T)$ guarantee. If such long patterns exist, one natural question is how close to the optimal $O(1/T^2)$ rate attained by momentum methods can be achieved by gradient descent with long steps. The numerics of [13, Figure 2] suggested a $O(1/T^{1.178})$ rate may be possible. The theory of Lee and Wright [29] showed asymptotic $o(1/T)$ convergence for constant stepsize gradient descent (although no improved guarantees in finite-time were given), which also motivates the possibility for improved finite-time convergence guarantees.

For strongly convex optimization (or, more generally, any problem satisfying a Hölder growth bound), the classic convergence rates for constant stepsize gradient descent are known to improve. We show the same improvements occur for any straightforward stepsize pattern (see Theorem 2.2) with an additional gain of $\text{avg}(h)$. We validate that the convergence speed-ups of Table 1 actually occur on randomly generated least squares problems in Figure 1, seeing gains proportional to $\text{avg}(h)$.

²A nice summary of this is given by the recent blog post [22].

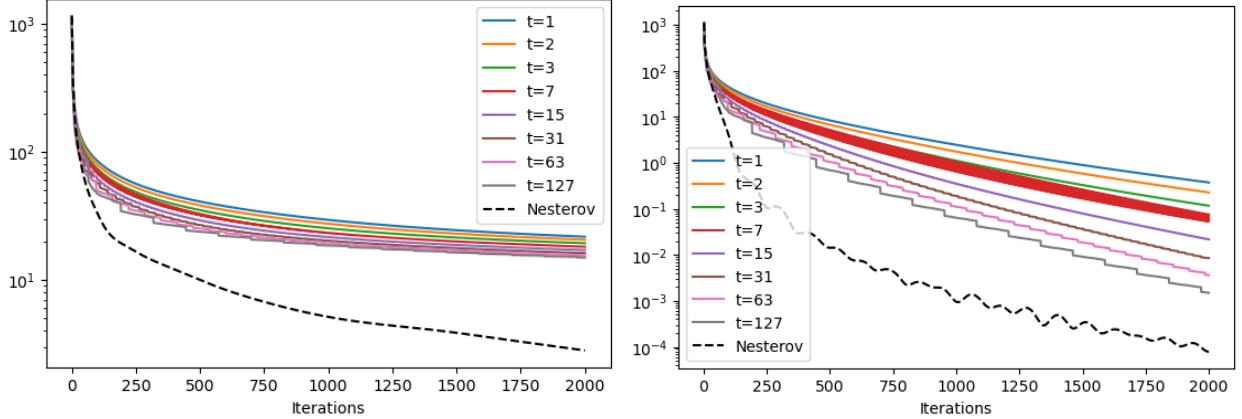


Figure 1: Least squares problems minimizing $\|Ax - b\|_2^2$ (left) and $\|Ax - b\|_2^2 + \|x\|_2^2$ (right) with i.i.d. normal entries in $A \in \mathbb{R}^{n \times n}$ and $b \in \mathbb{R}^n$ for $n = 4000$. Objective gaps are plotted over $T = 2000$ iterations with $h = (1)$ and with each pattern from Table 1. Note this second objective is substantially more strongly convex, so its faster linear convergence is expected. For comparison, Nesterov’s accelerated method (without modification to utilize strong convexity) is shown.

These plots also showcase that descent is not ensured within the execution of a straightforward pattern as $t = 7$ rapidly oscillates within each pattern while converging overall. Note, gradient descent is not an optimal first-order method as accelerated methods can attain a faster $O(1/T^2)$ rate. A basic implementation of Nesterov’s accelerated method is included in our numerics to showcase the degree of progress in closing this gap able to be accomplished via our longer steps.

Cyclic, periodically long stepsizes have been used in neural network training schedules [30–32]. Our results only apply to convex, deterministic problems. New techniques handling nonconvexity and stochasticity would need to be developed to describe the effect of long steps in such settings of machine learning.

Outline. In the remainder of this section, we informally sketch how our proof technique proceeds. Then Section 2 formally introduces our notion of straightforward stepsize patterns, showing that any such pattern has a guarantee of the form (1.4). Section 3 shows the existence of a solution to a certain semidefinite program implies straightforwardness. Section 4 concludes by outlining several future directions of interest enabled by and hopefully able to improve on this work.

1.1 Sketch of Proof Technique - Reducing proving eventual descent to an SDP

Our analysis works by guaranteeing a sufficient decrease is achieved after applying the whole pattern $h = (h_0, h_1, \dots, h_{t-1})$ of t steps (but not necessarily descending at any of the intermediate iterates). Our notion of straightforward stepsize patterns aims to ensure that for all $\delta > 0$ small enough, if $f(x_0) - f(x_\star) \leq \delta$, then x_t will always attain a descent of at least

$$f(x_t) - f(x_\star) \leq \delta - \frac{\sum_{i=0}^{t-1} h_i}{LD^2} \delta^2. \quad (1.5)$$

For constant stepsizes equal to one, this amounts to $f(x_0 - \nabla f(x_0)/L) \leq f(x_0) - \delta^2/LD^2$, a classic descent result that holds for all L -smooth convex f .

To prove a descent lemma like (1.5), one can take a direct combination of several known inequalities. This is a well-known approach, equivalent to providing dual solutions to the dual of the

associated performance estimation problem. We are given the equalities $x_{k+1} = x_k - (h_k/L)\nabla f(x_k)$ and $\nabla f(x_\star) = 0$ and as inequalities an initial distance bound

$$\|x_0 - x_\star\|_2^2 \leq D^2 ,$$

an initial objective gap bound

$$f(x_0) - f(x_\star) \leq \delta ,$$

and for any x_i and x_j with $i, j \in \{\star, 0, 1, 2, \dots, t\}$, convexity and smoothness imply [33, (2.1.10)]

$$f(x_i) \geq f(x_j) + \nabla f(x_j)^T(x_i - x_j) + \frac{1}{2L}\|\nabla f(x_i) - \nabla f(x_j)\|_2^2 .$$

For any nonnegative multipliers $v(\delta), w(\delta), \lambda_{i,j}(\delta) \geq 0$ (parameterized by the initial objective gap δ), one can combine these inequalities to conclude that on every L -smooth convex function, gradient descent (1.3) with stepsizes h satisfies

$$\begin{aligned} & v(\delta)(\|x_0 - x_\star\|_2^2 - D^2) \\ & + w(\delta)(f(x_0) - f(x_\star) - \delta) \\ & + \sum_{i,j \in \{\star, 0, \dots, t\}} \lambda_{i,j}(\delta) \left(f(x_j) + \nabla f(x_j)^T(x_i - x_j) + \frac{1}{2L}\|\nabla f(x_i) - \nabla f(x_j)\|_2^2 - f(x_i) \right) \leq 0 . \end{aligned} \tag{1.6}$$

Our proof then proceeds by showing carefully selected functions $w(\delta), v(\delta), \lambda(\delta)$ reduce this inequality to guaranteeing (1.5). We find to prove our guarantees in Table 1, it suffices to set $v(\delta) = \frac{\sum_{i=0}^{t-1} h_i}{LD^4}\delta^2$, $w(\delta) = 1 - \frac{2\sum_{i=0}^{t-1} h_i}{LD^2}\delta$, and use a linear function $\lambda(\delta) = \lambda + \delta\gamma$. For this choice of $v(\delta)$ and $w(\delta)$, our Theorem 3.1 shows (1.6) implies (1.5) if (λ, γ) is feasible to a certain SDP. Hence, proving a t iteration descent lemma of the form (1.5) can be done by semidefinite programming. Our Theorem 2.1 then completes the argument by showing such a periodic descent guarantee implies the guarantee (1.4).

2 Straightforward Stepsize Patterns

To analyze a given stepsize pattern, we first aim to understand its worst-case problem instance. We do so through the ‘‘Performance-Estimation Problem’’ (PEP) framework of [1]. Given a pattern h and bounds on smoothness L , initial distance to optimal D , and initial objective gap δ , the worst final objective gap able to be produced by one application of the stepsize pattern is given by

$$p_{L,D}(\delta) := \begin{cases} \max_{x_0, x_\star, f} & f(x_t) - f(x_\star) \\ \text{s.t.} & f \text{ is convex, } L\text{-smooth} \\ & \|x_0 - x_\star\|_2 \leq D \\ & f(x_0) - f(x_\star) \leq \delta \\ & \nabla f(x_\star) = 0 \\ & x_{k+1} = x_k - \frac{h_k}{L}\nabla f(x_k) \quad \forall k = 0, \dots, t-1 . \end{cases} \tag{2.1}$$

Note this problem is infinite-dimensional as it optimizes over functions $f: \mathbb{R}^n \rightarrow \mathbb{R}$ and vectors $x_0, x_\star \in \mathbb{R}^n$ for some given problem dimension n . The key insight of Drori and Teboulle [1] is that such infinite-dimensional problems can be relaxed to a finite-dimensional semidefinite program. Subsequently, Taylor et al. [2] showed this relaxation can be made an exact reformulation. We formalize and utilize these insights in Section 3. We note one particular nice property here: $p_{L,D}$ is concave.

Lemma 2.1. For any $L, D > 0$, $p_{L,D}$ is concave.

Proof. Consider any $\delta^{(1)}, \delta^{(2)} \in \mathbb{R}$ and $\lambda \in (0, 1)$. Denote feasible solutions to the problem defining $p_{L,D}(\delta^{(i)})$ for each $i = 1, 2$ by $f^{(i)}, x_0^{(i)}, x_\star^{(i)}$. Note if no feasible solution exists for either of these problems, one vacuously has concavity as

$$\lambda p_{L,D}(\delta^{(1)}) + (1 - \lambda)p_{L,D}(\delta^{(2)}) = -\infty \leq p_{L,D}(\lambda\delta^{(1)} + (1 - \lambda)\delta^{(2)}) .$$

Note without loss of generality, $x_\star^{(i)} = 0$ and $f^{(i)}(x_\star^{(i)}) = 0$. Then one can easily check $\lambda f^{(1)} + (1 - \lambda)f^{(2)}, \lambda x_0^{(1)} + (1 - \lambda)x_0^{(2)}, \lambda x_\star^{(1)} + (1 - \lambda)x_\star^{(2)}$ is feasible to the problem defining $p_{L,D}(\lambda\delta^{(1)} + (1 - \lambda)\delta^{(2)})$. Consequently, taking the supremum over choices of $f^{(i)}, x_0^{(i)}$ gives concavity as

$$\lambda p_{L,D}(\delta^{(1)}) + (1 - \lambda)p_{L,D}(\delta^{(2)}) \leq p_{L,D}(\lambda\delta^{(1)} + (1 - \lambda)\delta^{(2)}) .$$

□

Generally, the worst-case functions f attaining (2.1) can be nontrivial. To find a tractable family of stepsizes for analysis, we focus on ones where this worst-case behavior is no worse than a simple one-dimensional setting: Consider the one-dimensional (nonsmooth) convex function linearly decreasing from x_0 to x_\star and constant thereafter. That is, given L, D, δ , consider the problem instance $f(x) = \max\{\delta x/D, 0\}$, $x_0 = D$, and $x_\star = 0$. Provided δ is small enough (i.e., $\delta \leq LD^2 / \sum_{i=0}^{t-1} h_i$), the gradient descent iteration $x_{k+1} = x_k - \frac{h_k}{L} \nabla f(x_k)$ has

$$\delta_t = \delta_0 - \frac{\sum_{i=0}^{t-1} h_i}{LD^2} \delta_0^2 \tag{2.2}$$

where $\delta_k = f(x_k) - f(x_\star)$ denotes the objective gap. In this example, gradient descent spends all t iterations moving straight forward in a line along the slope. So the descent achieved is just controlled by the objective's slope δ_0/D (squared) and the total length of steps taken $\sum_{i=0}^{t-1} h_i/L$.

We say that a stepsize pattern h is straightforward if its worst-case behavior (over all smooth functions, as defined in (2.1)) is no worse than this one-dimensional piecewise linear setting for δ small enough. Formally, we say that a stepsize pattern h of length t is *straightforward* if for some $\Delta \in (0, 1/2]$,

$$p_{L,D}(\delta) \leq \delta - \frac{\sum_{i=0}^{t-1} h_i}{LD^2} \delta^2 \quad \forall \delta \in [0, LD^2 \Delta]$$

for any $L, D > 0$. Our analysis of gradient descent with long steps then proceeds by:

- (i) *Certifying h is straightforward (able to be computer automated)* see Theorem 3.1,
- (ii) *Solving a resulting recurrence, showing $\delta_T \leq \frac{LD^2}{\text{avg}(h)T} + O(1/T^2)$* see Theorem 2.1.

The second step is simpler, so we address it first. Additionally, we show the same factor of guarantee improvement $\text{avg}(h)$ carries over to more structured domains like strongly convex optimization.

2.1 Convergence Guarantees for Straightforward Stepsize Patterns

Note straightforwardness provides no guarantees on the intermediate objective values at iterations $k = 1, 2, \dots, t - 1$. We show in our theorem below descent at every t th iteration. To allow for some numerical flexibility, we say a pattern is ϵ -*straightforward* if for some $\Delta \in (0, 1/2]$, all $\delta \in [0, LD^2 \Delta]$ have value function bounded by $p_{L,D}(\delta) \leq \delta - \frac{\sum_{i=0}^{t-1} (h_i - \epsilon)}{LD^2} \delta^2$.

Theorem 2.1. Consider any L -smooth, convex f . If $h = (h_0, \dots, h_{t-1})$ is ϵ -straightforward with parameter $\Delta \in (0, 1/2]$, then gradient descent (1.3) has, for any $s \in \mathbb{N}$, after $T = st$ gradient steps

$$f(x_T) - f(x_\star) \leq \begin{cases} (1 - \sum_{i=0}^{t-1} (h_i - \epsilon)\Delta)^s (f(x_0) - f(x_\star)) & \text{if } s \leq \bar{s} \\ \frac{LD^2}{(\text{avg}(h) - \epsilon)(T - \bar{s}t) + \frac{1}{\Delta}} & \text{if } s > \bar{s} \end{cases} \quad (2.3)$$

where $\bar{s} = \left\lceil \frac{\log\left(\frac{f(x_0) - f(x_\star)}{LD^2\Delta}\right)}{\log(1 - \sum_{i=0}^{t-1} (h_i - \epsilon)\Delta)} \right\rceil$. In particular, suppressing lower-order terms, this guarantee is

$$f(x_T) - f(x_\star) \leq \frac{LD^2}{(\text{avg}(h) - \epsilon)T} + O(1/T^2).$$

Proof. For any $\epsilon \geq 0$, we begin by showing ϵ -straightforwardness implies that for any $s \in \mathbb{N}$, the recurrence relation

$$\delta_{(s+1)t} \leq \begin{cases} \delta_{st} - \frac{\sum_{i=0}^{t-1} (h_i - \epsilon)\delta_{st}^2}{LD^2} & \text{if } \delta_{st} \leq LD^2\Delta \\ \left(1 - \sum_{i=0}^{t-1} (h_i - \epsilon)\Delta\right) \delta_{st} & \text{if } \delta_{st} > LD^2\Delta. \end{cases} \quad (2.4)$$

Consider first $s = 0$. The first case of (2.4) follows from the definition of straightforwardness as $\delta_t \leq p_{L,D}(\delta_0) \leq \delta_0 - \frac{\sum_{i=0}^{t-1} (h_i - \epsilon)\delta_0^2}{LD^2}$. For the second case of (2.4), first note that $p_{L,D}(\cdot)$ is concave by Lemma 2.1. Then since $p_{L,D}(0) = 0$ and $p_{L,D}(LD^2\Delta) \leq LD^2\left(\Delta - \sum_{i=0}^{t-1} (h_i - \epsilon)\Delta^2\right)$, every $\delta_0 > LD^2\Delta$ must have

$$\delta_t \leq p_{L,D}(\delta_0) \leq \frac{\delta_0}{LD^2\Delta} p_{L,D}(LD^2\Delta) \leq \delta_0 \left(1 - \sum_{i=0}^{t-1} (h_i - \epsilon)\Delta\right).$$

As a result, applying the sequence of steps from the straightforward pattern h from x_0 yields $\delta_t \leq \delta_0$. Hence $\|x_t - x_\star\| \leq D$ since x_t lies in the initial level set $\{x \mid f(x) \leq f(x_0)\}$. Thus the above reasoning can apply inductively, giving the claimed recurrence.

Next, we show this recurrence implies the claimed convergence guarantee (2.3). Suppose first, δ_{st} is larger than $LD^2\Delta$. Then applying the pattern h contracts the objective gap³, inductively giving $\delta_{st} \leq (1 - \sum_{i=0}^{t-1} (h_i - \epsilon)\Delta)^s \delta_0$. After at most \bar{s} executions of the stepsize pattern, one must have $\delta_{st} \leq LD^2\Delta$. Afterward, for any $s > \bar{s}$, the objective gap decreases by at least

$$\delta_{(s+1)t} \leq \delta_{st} - \frac{\sum_{i=0}^{t-1} (h_i - \epsilon)\delta_{st}^2}{LD^2}.$$

Solving this recurrence with the initial condition $\delta_{\bar{s}t} \leq LD^2\Delta$ gives

$$\delta_{st} \leq \frac{LD^2}{\sum_{i=0}^{t-1} (h_i - \epsilon)(s - \bar{s}) + \frac{1}{\Delta}}.$$

□

³Interestingly, this contraction factor is independent of L, D . As a result, problem conditioning plays a minimal role in this initial phase of convergence. Instead, only h and the associated straightforwardness parameter Δ matter.

2.2 Faster Convergence for Straightforward Patterns given Growth Bounds

A function f is μ -strongly convex if $f - \frac{\mu}{2}\|\cdot\|_2^2$ is convex. This condition is well-known to lead to linear convergence for most first-order methods in smooth optimization. More generally, faster convergence occurs whenever f satisfies a Hölder growth or error bound condition. Here we consider settings where all $x \in \mathbb{R}^n$ within the level set $f(x) \leq f(x_0)$ satisfy

$$f(x) - f(x_*) \geq \frac{\mu}{q}\|x - x_*\|_2^q. \quad (2.5)$$

Strong convexity implies this condition with $q = 2$ and leads gradient descent with constant $h = (1, 1, \dots)$ to converge at a rate of $O((1 - \mu/L)^T)$. When $q > 2$, improved sublinear guarantees of $O((L/\mu^{2/q}T)^{q/(q-2)})$ follow. Below, we show that any straightforward stepsize pattern enjoys the same convergence improvements, gaining a similar factor of $\text{avg}(h)$.

Theorem 2.2. *Consider any L -smooth, convex objective f satisfying (2.5). If $h = (h_0, \dots, h_{t-1})$ is ϵ -straightforward with parameter $\Delta \in (0, 1/2]$, then gradient descent (1.3) has, for any $T = st$*

$$f(x_T) - f(x_*) \leq \left(1 - \min\left\{\Delta, \frac{\mu}{2L}\right\} \sum_{i=0}^{t-1} (h_i - \epsilon)\right)^s (f(x_0) - f(x_*))$$

when $q = 2$ and when $q > 2$ and $s > \bar{s} := \left\lceil \frac{\log\left(\frac{f(x_0) - f(x_*)\mu^{2/(q-2)}}{q^{2/(q-2)}(L\Delta)^{q/(q-2)}}\right)}{\log(1 - \sum_{i=0}^{t-1} (h_i - \epsilon)\Delta)} \right\rceil$ has

$$f(x_T) - f(x_*) \leq q \left(\frac{L}{\mu^{2/q}(q-2)(\text{avg}(h) - \epsilon)((s - \bar{s})t)} \right)^{q/(q-2)}.$$

Proof. Let $D_k = \sup\{\|x - x_*\| \mid f(x) \leq f(x_k)\}$ denote the size of each level set visited by gradient descent. The growth bound (2.5) ensures $D_k \leq (\frac{q}{\mu}\delta_k)^{1/q}$. Then the recurrence (2.4) implies

$$\delta_{(s+1)t} \leq \begin{cases} \delta_{st} - \frac{\mu^{2/q} \sum_{i=0}^{t-1} (h_i - \epsilon)}{Lq^{2/q}} \delta_{st}^{2-2/q} & \text{if } \delta_{st} \leq LD_{st}^2\Delta \\ \left(1 - \sum_{i=0}^{t-1} (h_i - \epsilon)\Delta\right) \delta_{st} & \text{if } \delta_{st} > LD_{st}^2\Delta. \end{cases}$$

One can bound this recurrence relation by the maximum of the two cases. If $q = 2$, both cases give a contraction and so

$$\delta_{st} \leq \left(1 - \min\left\{\Delta, \frac{\mu}{2L}\right\} \sum_{i=0}^{t-1} (h_i - \epsilon)\right)^s \delta_0.$$

If $q > 2$, one has

$$\delta_{(s+1)t} \leq \max\left\{\delta_{st} - \frac{\mu^{2/q} \sum_{i=0}^{t-1} (h_i - \epsilon)}{Lq^{2/q}} \delta_{st}^{2-2/q}, \left(1 - \sum_{i=0}^{t-1} (h_i - \epsilon)\Delta\right) \delta_{st}\right\}.$$

Solving this recurrence gives linear convergence until the objective gap is less than $\left(\frac{q^{2/q}L\Delta}{\mu^{2/q}}\right)^{\frac{q}{q-2}}$ is reached, which must occur by iteration \bar{s} , and then afterwards gives a sublinear convergence rate (for example, see [34, Lemma A.1] for this calculation), giving the claim. \square

Note that the above bound for the case of $q > 2$ is independent of D . Asymptotically, this is reasonable as better bounds on distance to optimal are eventually available, namely $(\frac{q}{\mu}\delta_k)^{1/q}$. One could give a bound depending on $D = D_0$ early on by using the stronger bound $D_k \leq \min\{(\frac{q}{\mu}\delta_k)^{1/q}, D_0\}$. However, once the gap is small enough for the first term to dominate, the bound will again be independent of D_0 .

3 Certificates of Straightforwardness

All that remains is to show how one can certify the straightforwardness of a stepsize pattern. We do this in two steps. First, Section 3.1 shows that $p_{L,D}(\delta)$ is upper bounded by an SDP minimization problem using previously developed PEP techniques. Hence straightforwardness is implied by showing the SDP corresponding to each $\delta \in [0, LD^2\Delta]$ has a sufficiently good feasible solution. Second, Section 3.2 shows that another semidefinite programming feasibility problem can certify that such an interval of solutions exists.

3.1 The Worst-Case Value Function is Upper Bounded by an SDP

We first reformulate the infinite-dimensional problem (2.1) as a finite-dimensional nonconvex quadratically constrained quadratic problem (see (3.1)), relax that formulation into an SDP (see (3.2)), and then upper bound the SDP by its dual problem (see (3.3)). This process was been developed by [1–3] and is carried out in our particular setting below.

Step 1: A QCQP reformulation. First, as proposed by Drori and Teboulle [1], one can discretize the infinite-dimensional problem defining $p_{L,D}(\delta)$ over all possible objective values f_k and gradients g_k at the points x_k with $k \in I_t^* := \{\star, 0, 1, \dots, t\}$ as done below. Using the interpolation theorem of Taylor et al. [2], this is an exact reformulation rather than a relaxation, giving

$$p_{L,D}(\delta) = \begin{cases} \max_{x_0, f, g} & f_t - f_\star \\ \text{s.t.} & f_i \geq f_j + g_j^T(x_i - x_j) + \frac{1}{2L}\|g_i - g_j\|_2^2 \quad \forall i, j \in I_t^* \\ & \|x_0 - x_\star\|_2^2 \leq D^2 \\ & f_0 - f_\star \leq \delta \\ & x_\star = 0, f_\star = 0, g_\star = 0 \\ & x_{i+1} = x_i - \frac{h_i}{L}g_i \quad \forall i = 0, \dots, t-1 \end{cases} \quad (3.1)$$

where, without loss of generality, we have fixed $x_\star = 0, f_\star = 0, g_\star = 0$.

Step 2: An SDP relaxation. Second, one can relax the nonconvex problem (3.1) to the following SDP as done in [2,3,15]. We follow the particular notational choices of [13], which recently considered globally, numerically optimizing gradient descent’s stepsizes with fixed total iterations, defining

$$\begin{aligned} H &:= [x_0 \mid g_0 \mid g_1 \mid \dots \mid g_t] \in \mathbb{R}^{d \times (t+2)}, \\ G &:= H^T H \in \mathbb{S}_+^{t+2}, \\ F &:= [f_0 \mid f_1 \mid \dots \mid f_t] \in \mathbb{R}^{1 \times (t+1)}, \end{aligned}$$

with the following notation for selecting columns and elements of H and F :

$$\begin{aligned} \mathbf{g}_\star &:= 0 \in \mathbb{R}^{t+2}, \quad \mathbf{g}_i := e_{i+2} \in \mathbb{R}^{t+2}, \quad i \in [0 : t] \\ \mathbf{x}_0 &:= e_1 \in \mathbb{R}^{t+2}, \quad \mathbf{x}_\star := 0 \in \mathbb{R}^{t+2}, \\ \mathbf{x}_i &:= \mathbf{x}_0 - \frac{1}{L} \sum_{j=0}^{i-1} h_j \mathbf{g}_j \in \mathbb{R}^{t+2}, \quad i \in [1 : t] \\ \mathbf{f}_\star &:= 0 \in \mathbb{R}^{t+1}, \quad \mathbf{f}_i := e_{i+1} \in \mathbb{R}^{t+1}, \quad i \in [0 : t]. \end{aligned}$$

This notation ensures $x_i = H\mathbf{x}_i$, $g_i = H\mathbf{g}_i$, and $f_i = F\mathbf{f}_i$. Furthermore, for $i, j \in I_t^*$, define

$$\begin{aligned} A_{i,j}(h) &:= \mathbf{g}_j \odot (\mathbf{x}_i - \mathbf{x}_j) \in \mathbb{S}^{t+2}, \\ B_{i,j}(h) &:= (\mathbf{x}_i - \mathbf{x}_j) \odot (\mathbf{x}_i - \mathbf{x}_j) \in \mathbb{S}_+^{t+2}, \\ C_{i,j} &:= (\mathbf{g}_i - \mathbf{g}_j) \odot (\mathbf{g}_i - \mathbf{g}_j) \in \mathbb{S}_+^{t+2}, \\ a_{i,j} &:= \mathbf{f}_j - \mathbf{f}_i \in \mathbb{R}^{t+1} \end{aligned}$$

where $x \odot y = \frac{1}{2}(xy^T + yx^T)$ denotes the symmetric outer product. This notation is defined so that $g_j^T(x_i - x_j) = \text{Tr}GA_{i,j}(h)$, $\|x_i - x_j\|_2^2 = \text{Tr}GB_{i,j}(h)$, and $\|g_i - g_j\|_2^2 = \text{Tr}GC_{i,j}$ for any $i, j \in I_t^*$. Then the QCQP formulation (3.1) can be relaxed to

$$p_{L,D}(\delta) \leq \begin{cases} \max_{F,G} & F\mathbf{f}_t \\ \text{s.t.} & Fa_{i,j} + \text{Tr}GA_{i,j}(h) + \frac{1}{2L}\text{Tr}GC_{i,j} \leq 0, \quad \forall i, j \in I_t^* : i \neq j \\ & -G \preceq 0 \\ & \text{Tr}GB_{0,\star} \leq D^2 \\ & F\mathbf{f}_0 \leq \delta. \end{cases} \quad (3.2)$$

Equality holds above if one adds the constraint that $\text{rank } G \leq n$ where n is the dimension of problem instances considered when defining $p_{L,D}$. This constraint is vacuously true if the considered problem dimension n exceeds $t+2$. Consequently, the QCQP (3.1) and the SDP (3.2) are equivalent, provided one allows sufficiently high-dimensional objectives, but may differ if one restricts to finding the worst-case over lower-dimensional problem instances. However, this equality is not needed for our analysis, so we make no such assumption.

Step 3: The upper bounding dual SDP. Third, note the maximization SDP (3.2) is bounded above by its dual minimization SDP by weak duality, giving

$$p_{L,D}(\delta) \leq \begin{cases} \min_{\lambda,v,w,Z} & D^2v + \delta w \\ \text{s.t.} & \sum_{i,j \in I_t^* : i \neq j} \lambda_{i,j} a_{i,j} = a_{\star,t} - wa_{\star,0} \\ & vB_{0,\star} + \sum_{i,j \in I_t^* : i \neq j} \lambda_{i,j} \left(A_{i,j}(h) + \frac{1}{2L}C_{i,j} \right) = Z \\ & Z \succeq 0 \\ & v, w \geq 0, \lambda_{i,j} \geq 0, \quad \forall i, j \in I_t^* : i \neq j. \end{cases} \quad (3.3)$$

Although it is not needed for our analysis, equality holds here as well (i.e., strong duality holds) due to [2, Theorem 6].

3.2 An SDP Feasibility Certificate that implies Straightforwardness

The preceding bound (3.3) establishes that $\epsilon \geq 0$ -straightforwardness holds if for some $\Delta \in (0, 1/2]$, every $\delta \in [0, LD^2\Delta]$ has a corresponding dual feasible solution with objective at most $\delta - \frac{\sum_{i=0}^{t-1} (h_i - \epsilon)}{LD^2} \delta^2$. We claim that it suffices to fix $L = 1, D = 1$ without loss of generality. For any L -smooth f with $\|x_0 - x_\star\| \leq D$, this follows by instead considering minimizing $\tilde{f}(\tilde{x}) = \frac{1}{LD^2}f(D\tilde{x})$. One can easily verify \tilde{f} is 1-smooth, has $\|\tilde{x}_0 - \tilde{x}_\star\| \leq 1$ for $\tilde{x}_0 = Dx_0, \tilde{x}_\star = Dx_\star$, and gradient descent $\tilde{x}_{k+1} = \tilde{x}_k - h_k \nabla \tilde{f}(\tilde{x}_k)$ produces exactly the iterates of $x_{k+1} = x_k - h_k/L \nabla f(x_k)$ rescaled by D . Hence $p_{L,D}(\delta) = LD^2 p_{1,1}(\delta/LD^2)$.

We restrict our search for dual certificates bounding $p_{1,1}(\delta)$ to a special case, which we numerically observed to hold approximately at the minimizers of (3.3): given δ , fix $v = \sum_{i=0}^{t-1} (h_i + \epsilon) \delta^2$ and

$w = 1 - 2 \sum_{i=0}^{t-1} h_i \delta$. Noting this fixed variable setting has $v + \delta w = \delta - \sum_{i=0}^{t-1} (h_i - \epsilon) \delta^2$, ϵ -straightforwardness follows if one can show feasible solutions with these fixed values exist.

Given δ and a selection of $\lambda \in \mathbb{R}^{(t+2) \times (t+2)}$ and fixing v as above, we define

$$Z_{h,\epsilon}(\lambda, \delta) := \sum_{i=0}^{t-1} (h_i + \epsilon) \delta^2 B_{0,\star} + \sum_{i,j \in I_t^*: i \neq j} \lambda_{i,j} \left(A_{i,j}(h) + \frac{1}{2} C_{i,j} \right). \quad (3.4)$$

Observe that $Z_{h,\epsilon}(\lambda, \delta)$ is nearly linear: the first entry has the only nonlinear behavior, depending quadratically on δ , with the rest depending only linearly on λ . Written in block form, we denote

$$Z_{h,\epsilon}(\lambda, \delta) =: \begin{bmatrix} \sum_{i=0}^{t-1} (h_i + \epsilon) \delta^2 & m_h(\lambda)^T \\ m_h(\lambda) & M_h(\lambda) \end{bmatrix} \quad (3.5)$$

where $m_h: \mathbb{R}^{(t+2) \times (t+2)} \rightarrow \mathbb{R}^{t+1}$ and $M_h: \mathbb{R}^{(t+2) \times (t+2)} \rightarrow \mathbb{R}^{(t+1) \times (t+1)}$ are linear functions. Certifying $p_{1,1}(\delta) \leq \delta - \sum_{i=0}^{t-1} (h_i - \epsilon) \delta^2$ for fixed δ then follows by showing the following spectral set is nonempty

$$\mathcal{R}_{h,\epsilon,\delta} = \left\{ \lambda \in \mathbb{R}^{(t+2) \times (t+2)} \mid \begin{array}{l} \sum_{i,j \in I_t^*: i \neq j} \lambda_{i,j} a_{i,j} = a_{\star,t} - \left(1 - 2 \sum_{i=0}^{t-1} h_i \delta\right) a_{\star,0} \\ \lambda \succeq 0 \\ Z_{h,\epsilon}(\lambda, \delta) \succeq 0 \end{array} \right\}.$$

Lemma 3.1. *A stepsize pattern $h \in \mathbb{R}^t$ is $\epsilon \geq 0$ -straightforward if for some $\Delta \in (0, 1/2]$, $\mathcal{R}_{h,\epsilon,\delta}$ is nonempty for all $\delta \in [0, \Delta]$. Straightforwardness of h is implied by each $\mathcal{R}_{h,0,\delta}$ being nonempty.*

This lemma alone does not directly enable the computation of a convergence-proof certificate. One would need certificates of feasibility for the infinitely many sets given by each $\delta \in [0, \Delta]$. The following theorem shows that the existence of such solutions can be certified via a single feasible solution to yet another semidefinite program.

Theorem 3.1. *A stepsize pattern $h \in \mathbb{R}^t$ is $\epsilon \geq 0$ -straightforward if for some $\Delta \in (0, 1/2]$, $\mathcal{S}_{h,\epsilon,\Delta}$ is nonempty where*

$$\mathcal{S}_{h,\epsilon,\Delta} = \left\{ (\lambda, \gamma) \in \mathbb{R}^{(t+2) \times (t+2)} \times \mathbb{R}^{(t+2) \times (t+2)} \mid \begin{array}{l} \sum_{i,j \in I_t^*: i \neq j} \lambda_{i,j} a_{i,j} = a_{\star,t} - a_{\star,0} \\ \sum_{i,j \in I_t^*: i \neq j} \gamma_{i,j} a_{i,j} = 2 \sum_{i=0}^{t-1} h_i a_{\star,0} \\ m_h(\lambda) = 0 \\ \lambda \succeq 0, \lambda + \Delta \gamma \succeq 0 \\ \begin{bmatrix} \sum_{i=0}^{t-1} (h_i + \epsilon) & m_h(\gamma)^T \\ m_h(\gamma) & M_h(\lambda) \end{bmatrix} \succeq 0 \\ \begin{bmatrix} \sum_{i=0}^{t-1} (h_i + \epsilon) & m_h(\gamma)^T \\ m_h(\gamma) & M_h(\lambda + \Delta \gamma) \end{bmatrix} \succeq 0 \end{array} \right\}.$$

Proof. Let $(\lambda, \gamma) \in \mathcal{S}_{h,\epsilon,\Delta}$. We prove this by showing $\lambda^{(\delta)} := \lambda + \delta \gamma \in \mathcal{R}_{h,\epsilon,\delta}$ for every $\delta \in [0, \Delta]$ by Lemma 3.1. This amounts to verifying the three conditions defining $\mathcal{R}_{h,\epsilon,\delta}$ for each $\lambda^{(\delta)}$.

First, we check $\sum_{i,j \in I_t^*: i \neq j} \lambda_{i,j}^{(\delta)} a_{i,j} = a_{\star,t} - \left(1 - 2 \sum_{i=0}^{t-1} h_i \delta\right) a_{\star,0}$. The first equality defining $\mathcal{S}_{h,\epsilon,\Delta}$ ensures this for $\lambda^{(0)} = \lambda$. Adding δ times the second equality defining $\mathcal{S}_{h,\epsilon,\Delta}$ establishes the equality for every $\lambda^{(\delta)}$ as $\sum_{i,j \in I_t^*: i \neq j} (\lambda_{i,j}^{(\delta)} + \delta \gamma_{i,j}) a_{i,j} = a_{\star,t} - \left(1 - 2 \sum_{i=0}^{t-1} h_i \delta\right) a_{\star,0}$. Second, we check nonnegativity $\lambda^{(\delta)} \succeq 0$. This follows by noting $\lambda^{(\delta)}$ is a convex combination of λ and $\lambda + \Delta \gamma$,

which are nonnegative by construction. Finally, we check the nonlinear (but nearly linear) condition $Z_{h,\epsilon}(\lambda^{(\delta)}, \delta) \succeq 0$. We consider the block-form (3.5) of this semidefinite inequality, seeking

$$Z_{h,\epsilon}(\lambda^{(\delta)}, \delta) = \begin{bmatrix} \sum_{i=0}^{t-1} (h_i + \epsilon) \delta^2 & m_h(\lambda^{(\delta)})^T \\ m_h(\lambda^{(\delta)}) & M_h(\lambda^{(\delta)}) \end{bmatrix} \succeq 0.$$

Since $m_h(\lambda) = 0$, using the linearity of m_h and M_h , the above can be expanded to equal

$$\begin{bmatrix} \sum_{i=0}^{t-1} (h_i + \epsilon) \delta^2 & \delta m_h(\gamma)^T \\ \delta m_h(\gamma) & M_h(\lambda) + \delta M_h(\gamma) \end{bmatrix} \succeq 0.$$

Rescaling the first row and column by $1/\delta$ gives an equivalent condition, which is now linear in δ ,

$$\begin{bmatrix} \sum_{i=0}^{t-1} (h_i + \epsilon) & m_h(\gamma)^T \\ m_h(\gamma) & M_h(\lambda) + \delta M_h(\gamma) \end{bmatrix} \succeq 0.$$

When $\delta = 0$ or Δ , this condition is explicitly ensured by the definition of $\mathcal{S}_{h,\epsilon,\Delta}$. Then the linearity and convexity of this condition imply it holds for all intermediate $\lambda^{(\delta)}$, completing the proof. \square

Note computing a member of $\mathcal{S}_{h,\epsilon,\Delta}$ does not correspond to solving a particular performance estimation problem. Instead, each member provides a “line segment” of solutions to a series of the performance estimation problems $p_{L,D}(\delta)$ for an interval of possible δ values.

3.3 Certificates of Straightforwardness Proving Guarantees in Table 1

To prove a given pattern h converges at rate $LD^2/\text{avg}(h)T$, we only need to show some $\mathcal{S}_{h,0,\Delta}$ is nonempty. The most natural path is to provide an exact member of this set. For all of the stepsizes in Table 1, it was relatively easy to find a feasible solution $\mathcal{S}_{h,0,\Delta}$ in floating point arithmetic via an interior point method. However, exactly identifying a member of $\mathcal{S}_{h,0,\Delta}$ from this can still be hard.

First, we prove the claimed guarantees for the $t = 2$ and $t = 3$ stepsize patterns of Table 1 by presenting exact members of $\mathcal{S}_{h,0,\Delta}$. Then, to handle larger values of t , we present a simple rounding approach able to produce members of $\mathcal{S}_{h,\epsilon,\Delta}$, often with ϵ around the accuracy of our SDP solves $\approx 10^{-9}$. This approach produced rational-valued certificates proving the rest of the claimed convergence guarantees in Table 1.

The exact rational arithmetic verifying the correctness of all certificates (λ, γ) was done in `Mathematica 13.0.1.0`. Note that the entries in these certificates for $t \geq 7$ are entirely computer-generated and lack real human insight. As an example for reference, the certificate for $t = 7$ is included in the appendix. Larger certificates are impractical to include here. For example, our $t = 127$ guarantee is certificate (λ, γ) has 32640 nonzero entries. Certificates for every pattern in Table 1 and exact verifying computations are available at github.com/bgrimmer/LongStepCertificates.

Theorem 3.2. *For any $\eta \in (0, 3)$, the stepsize pattern $h = (3 - \eta, 1.5)$ is straightforward. Hence gradient descent (1.3) alternating between these two stepsizes has every even T satisfy*

$$f(x_T) - f(x_*) \leq \frac{LD^2}{(2.25 - \eta/2) \times T} + O(1/T^2).$$

Proof. For any $\eta \in (0, 3)$, consider the selection of (λ, γ) given by

$$\lambda = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{1}{2} & \frac{1}{2} \\ 0 & 0 & 0 & \frac{1}{2} \\ 0 & 0 & 0 & 0 \end{bmatrix}, \quad \gamma = \begin{bmatrix} 0 & 3 - \eta & \frac{6-\eta}{2} & \frac{6-\eta}{2} \\ 0 & 0 & \frac{-(6-\eta)}{2} & \frac{-(6-\eta)}{2} \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}.$$

It suffices to show for some $\Delta \in (0, 1/2]$, $(\lambda, \gamma) \in \mathcal{S}_{h,0,\Delta}$. One can easily verify the needed equalities and nonnegativities hold for all $0 \leq \Delta \leq 1/(6 - \eta)$. The first positive semidefiniteness condition of $(\lambda, \gamma) \in \mathcal{S}_{h,0,\Delta}$ amounts to checking every $\eta \in (0, 3)$ has

$$\begin{bmatrix} \sum_{i=0}^{t-1} (h_i + \epsilon) & m_h(\gamma)^T \\ m_h(\gamma) & M_h(\lambda) \end{bmatrix} = \begin{bmatrix} \frac{9-2\eta}{2} & \frac{-(3-\eta)}{2} & \frac{-(6-\eta)}{4} & \frac{-(6-\eta)}{4} \\ \frac{-(3-\eta)}{2} & \frac{1}{2} & \frac{2-\eta}{4} & \frac{2-\eta}{4} \\ \frac{-(6-\eta)}{4} & \frac{2-\eta}{4} & \frac{1}{2} & \frac{1}{2} \\ \frac{-(6-\eta)}{4} & \frac{2-\eta}{4} & \frac{1}{2} & \frac{1}{2} \end{bmatrix} \succeq 0.$$

Since this convex condition is linear in η , it suffices to check it at $\eta = 0$ and $\eta = 3$. Moreover, for any $\eta \in (0, 3)$, note this matrix has exactly two zero eigenvalues with associated eigenvectors spanning $(1/2, 1/2, 1, 0)$ and $(1/2, 1/2, 0, 1)$. The second positive semidefiniteness condition amounts to checking an update to this matrix of size Δ remains positive semidefinite, namely $\begin{bmatrix} \sum_{i=0}^{t-1} (h_i + \epsilon) & m_h(\gamma)^T \\ m_h(\gamma) & M_h(\lambda + \Delta\gamma) \end{bmatrix}$ which equals

$$\begin{bmatrix} \frac{9-2\eta}{2} & \frac{-(3-\eta)}{2} & \frac{-(6-\eta)}{4} & \frac{-(6-\eta)}{4} \\ \frac{-(3-\eta)}{2} & \frac{1}{2} & \frac{2-\eta}{4} & \frac{2-\eta}{4} \\ \frac{-(6-\eta)}{4} & \frac{2-\eta}{4} & \frac{1}{2} & \frac{1}{2} \\ \frac{-(6-\eta)}{4} & \frac{2-\eta}{4} & \frac{1}{2} & \frac{1}{2} \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & \frac{-3}{2} & \frac{6-\eta}{4} & \frac{6-\eta}{4} \\ 0 & \frac{6-\eta}{4} & 0 & 0 \\ 0 & \frac{6-\eta}{4} & 0 & 0 \end{bmatrix} \Delta$$

must be positive semidefinite. One can check this added matrix term is positive semidefinite on the subspace spanned by $(1/2, 1/2, 1, 0)$ and $(1/2, 1/2, 0, 1)$ (again by checking when $\eta = 0$ and $\eta = 3$ and then using convexity). As a result, positive semidefiniteness is maintained for Δ small enough. Exact arithmetic verifying all of these claims are given in the associated **Mathematica** notebook. Hence $(\lambda, \gamma) \in \mathcal{S}_{h,0,\Delta}$, proving the main claim by Theorem 3.1 and the claimed convergence guarantee by Theorem 2.1. \square

Theorem 3.3. *The stepsize pattern $h = (1.5, 4.9, 1.5)$ is straightforward. Hence gradient descent (1.3) cycling through these three stepsizes has every $T = 3s$ satisfy*

$$f(x_T) - f(x_*) \leq \frac{LD^2}{2.63333... \times T} + O(1/T^2).$$

Proof. This result is certified with $\Delta = 10^{-4}$ by the following exact values for $(\lambda, \gamma) \in \mathcal{S}_{h,0,\Delta}$ of

$$\lambda = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1.95 & 0.003 & 0.007 \\ 0 & 0.95 & 0 & 0.5 & 0.5 \\ 0 & 0.006 & 0 & 0 & 0.51 \\ 0 & 0.004 & 0 & 0.013 & 0 \end{bmatrix}, \quad \gamma = \begin{bmatrix} 0 & 0.005 & 7.825 & 3.9497 & 4.0203 \\ 0 & 0 & -5.24 & -10.555 & 0 \\ 0 & 0 & 0 & 7.9 & -5.315 \\ 0 & 0 & 0 & 0 & 1.2947 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}.$$

\square

Based on numerical exploration, we conjecture that patterns of the form $(3 - \eta, 1.5)$ are the longest straightforward patterns of length two and $(1.5, 5 - \eta, 1.5)$ are the longest of length three.

For larger settings $t \geq 7$, determining an exact member of $\mathcal{S}_{h,0,\Delta}$ proved difficult. So we resort to a fully automated construction of a convergence guarantee certificate by first numerically computing an approximate member of $\mathcal{S}_{h,0,\Delta}$ (via an interior point method) and then rounding to a nearby

rational-valued exact member of $\mathcal{S}_{h,\epsilon,\Delta}$ for some small ϵ . In light of our Theorem 2.1, such rounding only weakens the resulting guarantee’s coefficient from $\text{avg}(h)$ to $\text{avg}(h) - \epsilon$.

Computer Generation of Convergence Proof Certificates. Given h and Δ ,

- (i) Numerically compute some $(\tilde{\lambda}, \tilde{\gamma})$ approximately in $\mathcal{S}_{h,0,\Delta}$,
- (ii) Compute rational $(\hat{\lambda}, \hat{\gamma})$ near $(\tilde{\lambda}, \tilde{\gamma})$ exactly satisfying the three needed equalities,
- (iii) Check in exact arithmetic nonnegativity and positive definiteness of $M_h(\hat{\lambda})$ and $M_h(\hat{\lambda} + \Delta\hat{\gamma})$,
- (iv) If so, $(\hat{\lambda}, \hat{\gamma}) \in \mathcal{S}_{h,\epsilon,\Delta}$, certifying $\frac{LD^2}{(\text{avg}(h) - \epsilon)T} + O(1/T^2)$ convergence, for

$$\epsilon = \frac{\max\{m_h(\hat{\gamma})^T M_h(\hat{\lambda})^{-1} m_h(\hat{\gamma}), m_h(\hat{\gamma})^T M_h(\hat{\lambda} + \Delta\hat{\gamma})^{-1} m_h(\hat{\gamma})\}}{t} - \text{avg}(h) .$$

The above value of ϵ is the smallest value with $(\hat{\lambda}, \hat{\gamma}) \in \mathcal{S}_{h,\epsilon,\Delta}$, since by considering their Schur complements, the two needed positive semidefinite conditions hold if and only if

$$\sum_{i=0}^{t-1} (h_i + \epsilon) - m_h(\hat{\gamma})^T M_h(\hat{\lambda})^{-1} m_h(\hat{\gamma}) \geq 0 \quad \text{and} \quad \sum_{i=0}^{t-1} (h_i + \epsilon) - m_h(\hat{\gamma})^T M_h(\hat{\lambda} + \Delta\hat{\gamma})^{-1} m_h(\hat{\gamma}) \geq 0 .$$

Theorem 3.4. *The stepsize patterns of lengths $t \in \{7, 15, 31, 63, 127\}$ in Table 1 are all ϵ -straightforward for $\epsilon \in \{10^{-9}, 10^{-9}, 10^{-11}, 10^{-3}, 10^{-4}\}$ with $\Delta \in \{10^{-5}, 10^{-6}, 10^{-8}, 10^{-7}, 10^{-8}\}$. Hence the convergence guarantees claimed in Table 1 hold for each corresponding “long step” gradient descent method.*

Proof. Certificates $(\hat{\lambda}, \hat{\gamma}) \in \mathcal{S}_{h,\epsilon,\Delta}$ (produced via the above procedure) with exact arithmetic validation are available at github.com/bgrimmer/LongStepCertificates. \square

4 Future Directions

Building on existing performance estimation ideas, we have demonstrated an analysis technique capable of proving convergence of gradient descent using nonconstant, long stepsize patterns, which gain in strength as longer patterns are considered. This runs contrary to widely held intuitions regarding constant stepsize selections and the importance of monotone objective decreases. Instead, we show that long-run performance can improve by periodically taking (very) long steps that may increase the objective value in the short term. We accomplish this via computer-generated proof certificates bounding the effect of many steps collectively by providing solutions for a sequence of related performance estimation problems. We conclude by discussing a few possible future improvements on and shortcomings of this technique.

Possible Extension to More Families of Gradient Methods. Often, analysis techniques for gradient descent and its accelerated variants extend rather directly to constrained minimization or minimizing composite objectives $f(x) + r(x)$ by utilizing projections and proximal operators. PEP techniques naturally extend to these settings. Drori [35, Theorem 2.7 and 2.9] shows the tight convergence rate for projected gradient descent is strictly worse than the unconstrained setting, having rate $LD^2/4T$ instead of (1.2). A similar (small) worsening of the optimal rate also holds for accelerated proximal gradient methods [36]. This difficulty appears to extend to our straightforward analysis technique. Figure 2 shows the length two and three stepsize patterns that numerically satisfy a generalized notion of straightforwardness for gradient descent, projected gradient descent, and proximal gradient descent. The set of “straightforward” patterns for constrained is strictly smaller and composite yet strictly smaller. Consequently, strictly less aggressive stepsizes than Table 1 would be required for future projected or proximal extensions.

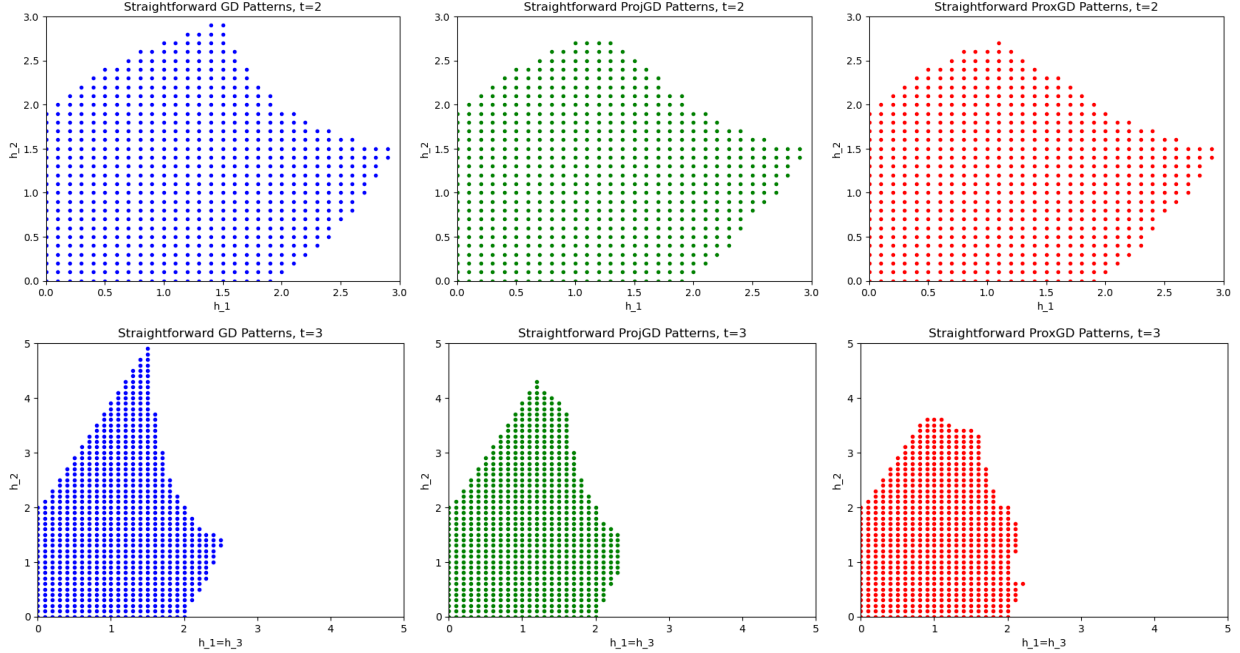


Figure 2: For unconstrained, constrained, and composite minimization, the stepsize patterns satisfying a natural generalization of straightforwardness are shown. Patterns (h_1, h_2) of length two and symmetric patterns (h_1, h_2, h_1) of length three were sampled at every 0.1 increment. Straightforwardness was approximated by solving a performance estimation problem determining if $f(x_t) - p_*$ (or $f(x_t) + r(x_t) - p_*$ for composite problems) is always at most $f(x_0) - p_* - \frac{\sum h_i}{LD^2} (f(x_0) - p_*)^2$ (or $f(x_0) + r(x_0) - p_* - \frac{\sum h_i}{LD^2} (f(x_0) + r(x_0) - p_*)^2$) given the initial gap was at most $\Delta = 10^{-4}$.

The PEP framework has previously been successfully employed in handling settings of inexact gradients [37] and relatively smooth optimization (via Bregman divergences) [7]. Determining the degree to which utilizing periodic long steps can improve the convergence of inexact, Bregman, and stochastic variations of gradient descent also provides interesting future directions.

Future Improvements in Algorithm Design. The search for long, straightforward stepsize patterns h is hard. The patterns presented in Table 1 resulted from substantial brute force searching. The task of maximizing $\text{avg}(h)$ subject to h being straightforward, although nonconvex, may be approachable using branch-and-bound techniques similar to those recently developed by Gupta et al. [13] and applied to a range of PEP parameter optimization problems. Such an approach may yield numerically, globally optimal h for fixed length t . This may also generate insights into the general form of the longest straightforward stepsize patterns for each fixed t .

One practical drawback of the method (1.3) is the requirement that one knows L . Since our analysis in Theorem 2.1 only relies on decreases in objective value after t steps, backtracking linesearch schemes or other adaptive ideas may be applicable. For example, one could consider an Armijo-type linesearching procedure, backtracking with an estimate of \tilde{L} seeking $f(x_t) \leq f(x_0) - \frac{\sum h_i}{\tilde{L}} \|\nabla f(x_0)\|^2$. Since convexity ensures $\|\nabla f(x_0)\|^2 \geq \delta_0^2/D^2$, this condition guarantees the descent required by straightforwardness. Analysis and practical development of such ideas are beyond our scope.

Future Improvements in Analysis Techniques. Future works may improve our analysis by considering other Lyapunov functions. The distance to optimal and norm of the gradient were both used in [4, Chapter 8]. Our proofs are only concerned with the eventual decrease of the

objective gap. The analysis of optimal accelerated and subgradient methods relies additionally on the decreasing distance to a minimizer or a decreasing combination thereof. Identifying better Lyapunovs and stepsize patterns guaranteed to eventually decrease them may lead to stronger guarantees.

Future Improvements in Computational Aspects. We observed that numerically computed primal optimal solutions to (3.2) were rank-one for all considered straightforward patterns. This corresponds to the worst-case objective function being essentially one-dimensional. This is in line with many prior PEP-based analyses finding one-dimensional Huber functions often attain the worst-case performance. This property was not used herein but could likely be leveraged to enable customized solvers for evaluating $p_{L,D}(\delta)$ and checking membership of $\mathcal{S}_{h,0,\Delta}$. Such improvements in tractability for SDPs with rank-one solutions have been studied widely [38–44] and may enable the search for longer, provably faster straightforward stepsize patterns than shown here.

As another avenue of improvement, note that any certificates produced by using floating point arithmetic followed by a rounding step (as done here) will likely lose a small ϵ amount in the guarantee. The use of an algebraic solver, like SPECTRA [45], could enable the automated production of exact certificates of straightforwardness as well as being able to certify when $\mathcal{S}_{h,0,\Delta}$ is empty.

Acknowledgements. The author thanks Jason Altschuler for conversations at the Simons Institute in Spring 2017 about the preliminary ideas subsequently developed in his excellent Master’s thesis work, Fanghua Chen for preliminary discussions based on numerical observations that motivated this work, and Adrien Taylor and Axel Böhm for providing useful feedback on the initial presentation of this work. The openly released Performance Estimation Problem Branch-and-Bound software of [13] was especially helpful in early explorations and building intuitions.

References

- [1] Yoel Drori and Marc Teboulle. Performance of first-order methods for smooth convex minimization: a novel approach. *Mathematical Programming*, 145:451–482, 2012.
- [2] Adrien Taylor, Julien Hendrickx, and François Glineur. Smooth strongly convex interpolation and exact worst-case performance of first-order methods. *Mathematical Programming*, 161:307–345, 2017.
- [3] Adrien Taylor, Julien Hendrickx, and François Glineur. Exact worst-case performance of first-order methods for composite convex optimization. *SIAM Journal on Optimization*, 27(3):1283–1313, 2017.
- [4] Jason Altschuler. *Greed, hedging, and acceleration in convex optimization*. PhD thesis, Massachusetts Institute of Technology, 2018.
- [5] Etienne de Klerk, François Glineur, and Adrien Taylor. On the worst-case complexity of the gradient method with exact line search for smooth strongly convex functions. *Optimization Letters*, 11:1185–1199, 2016.
- [6] Adrien Taylor, Bryan Van Scoy, and Laurent Lessard. Lyapunov functions for first-order methods: Tight automated convergence guarantees. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80, pages 4897–4906. PMLR, 10–15 Jul 2018.
- [7] Radu-Alexandru Dragomir, Adrien Taylor, Alexandre d’Aspremont, and Jérôme Bolte. Optimal complexity and certification of bregman first-order methods. *Mathematical Programming*, 194:41 – 83, 2019.
- [8] Adrien Taylor and Francis Bach. Stochastic first-order methods: non-asymptotic and computer-aided analyses via potential functions. *ArXiv*, abs/1902.00947, 2019.
- [9] Felix Lieder. On the convergence rate of the halpern-iteration. *Optimization Letters*, pages 1–14, 2020.

- [10] Ernest Ryu, Adrien Taylor, Carolina Bergeling, and Pontus Giselsson. Operator splitting performance estimation: Tight contraction factors and optimal parameter selection. *SIAM Journal on Optimization*, 30(3):2251–2271, 2020.
- [11] Guoyong Gu and Junfeng Yang. Tight sublinear convergence rate of the proximal point algorithm for maximal monotone inclusion problems. *SIAM Journal on Optimization*, 30(3):1905–1921, 2020.
- [12] Mathieu Barré, Adrien Taylor, and Francis Bach. Principled analyses and design of first-order methods with inexact proximal operators. *Mathematical Programming*, 2022.
- [13] Shuvomoy Das Gupta, Bart P.G. Van Parys, and Ernest Ryu. Branch-and-bound performance estimation programming: A unified methodology for constructing optimal optimization methods. *Mathematical Programming*, 2023.
- [14] Donghwan Kim and Jeffrey A. Fessler. Optimized first-order methods for smooth convex minimization. *Math. Program.*, 159(1–2):81–107, sep 2016.
- [15] Yoel Drori and Adrien Taylor. Efficient first-order methods for convex minimization: a constructive approach. *Mathematical Programming*, 184:183 – 220, 2018.
- [16] Donghwan Kim and Jeffrey A. Fessler. Optimizing the efficiency of first-order methods for decreasing the gradient of smooth convex functions. *J. Optim. Theory Appl.*, 188(1):192–219, jan 2021.
- [17] Donghwan Kim. Accelerated proximal point method for maximally monotone operators. *Math. Program.*, 190(1–2):57–87, nov 2021.
- [18] Adrien Taylor and Yoel Drori. An optimal gradient method for smooth strongly convex minimization. *Math. Program.*, 199(1–2):557–594, jun 2022.
- [19] Dimitri P. Bertsekas. Convex optimization algorithms. 2015.
- [20] Marc Teboulle and Yakov Vaisbourd. An elementary approach to tight worst case complexity analysis of gradient based methods. *Math. Program.*, 201(1–2):63–96, oct 2022.
- [21] David Young. On richardson’s method for solving linear systems with positive definite matrices. *Journal of Mathematics and Physics*, 32(1-4):243–255, 1953.
- [22] Fabian Pedregosa. Acceleration without momentum, 2021.
- [23] Samet Oymak. Provable super-convergence with a large cyclical learning rate. *IEEE Signal Process. Lett.*, 28:1645–1649, 2021.
- [24] Antoine Daccache. Performance estimation of the gradient method with fixed arbitrary step sizes. Master’s thesis, Université Catholique de Louvain, 2019.
- [25] Diego Eloi. Worst-case functions for the gradient method with fixed variable step sizes. Master’s thesis, Université Catholique de Louvain, 2022.
- [26] V.I. Lebedev and S.A. Finogenov. Ordering of the iterative parameters in the cyclical chebyshev iterative method. *USSR Computational Mathematics and Mathematical Physics*, 11(2):155–170, 1971.
- [27] Naman Agarwal, Surbhi Goel, and Cyril Zhang. Acceleration via fractal learning rate schedules. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139, pages 87–99, 2021.
- [28] Baptiste Goujaud, Damien Scieur, Aymeric Dieuleveut, Adrien B. Taylor, and Fabian Pedregosa. Super-acceleration with cyclical step-sizes. In *Proceedings of The 25th International Conference on Artificial Intelligence and Statistics*, pages 3028–3065, 2022.
- [29] Ching-Pei Lee and Stephen Wright. First-order algorithms converge faster than $o(1/k)$ on convex problems. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97, pages 3754–3762, 2019.
- [30] Ilya Loshchilov and Frank Hutter. Sgdr: Stochastic gradient descent with restarts. *ArXiv*, abs/1608.03983, 2016.

- [31] Leslie N. Smith. Cyclical learning rates for training neural networks. *2017 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 464–472, 2015.
- [32] Leslie N. Smith and Nicholay Topin. Super-convergence: very fast training of neural networks using large learning rates. In *Defense + Commercial Sensing*, 2017.
- [33] Yurii Nesterov. *Introductory Lectures on Convex Optimization: A Basic Course*. Springer Publishing Company, Incorporated, 1 edition, 2014.
- [34] Mateo Díaz and Benjamin Grimmer. Optimal convergence rates for the proximal bundle method. *SIAM Journal on Optimization*, 33(2):424–454, 2023.
- [35] Yoel Drori. *Contributions to the complexity analysis of optimization algorithms*. PhD thesis, Tel-Aviv University, 2014.
- [36] Uijeong Jang, Shuvomoy Das Gupta, and Ernest K. Ryu. Computer-assisted design of accelerated composite optimization methods: Optista, 2023.
- [37] Etienne De Klerk, François Glineur, and Adrien B. Taylor. Worst-case convergence analysis of inexact gradient and newton methods through semidefinite programming performance estimation. *SIAM Journal on Optimization*, 30(3):2053–2082, 2020.
- [38] Stefania Bellavia, Jacek Gondzio, and Margherita Porcelli. A relaxed interior point method for low-rank semidefinite programming problems with applications to matrix completion. *Journal of Scientific Computing*, 89, 2019.
- [39] Lijun Ding, Alp Yurtsever, Volkan Cevher, Joel A. Tropp, and Madeleine Udell. An optimal-storage approach to semidefinite programming using approximate complementarity. *SIAM Journal on Optimization*, 31(4):2695–2725, 2021.
- [40] Alp Yurtsever, Joel A. Tropp, Olivier Fercoq, Madeleine Udell, and Volkan Cevher. Scalable semidefinite programming. *SIAM Journal on Mathematics of Data Science*, 3(1):171–200, 2021.
- [41] Mario Souto, Joaquim D. Garcia, and Álvaro Veiga. Exploiting low-rank structure in semidefinite programming by approximate operator splitting. *Optimization*, 71(1):117–144, 2022.
- [42] Alex L. Wang, Yunlei Lu, and Fatma Kiliç-Karzan. Implicit regularity and linear convergence rates for the generalized trust-region subproblem. *SIAM Journal on Optimization*, 33(2):1250–1278, 2023.
- [43] Lijun Ding and Benjamin Grimmer. Revisiting spectral bundle methods: Primal-dual (sub)linear convergence rates. *SIAM Journal on Optimization*, 33(2):1305–1332, 2023.
- [44] Alex L. Wang and Fatma Kilinc-Karzan. Accelerated first-order methods for a class of semidefinite programs, 2022.
- [45] Didier Henrion, Simone Naldi, and Mohab Safey El Din. Spectra – a maple library for solving linear matrix inequalities in exact arithmetic. *Optimization Methods and Software*, 34:62 – 78, 2016.

A A Computer-Generated Straightforwardness Certificate with $\Delta = 10^{-5}$ and $\epsilon = 10^{-9}$ for $h = (1.5, 2.2, 1.5, 12.0, 1.5, 2.2, 1.5)$

Below is a certificate $(\hat{\lambda}, \hat{\gamma}) \in \mathcal{S}_{h, \epsilon, \Delta}$, completely computer generated, proving a $LD^2/(3.1999999 \times T)$ rate for the pattern of length $t = 7$ in Table 1. Given the length of these 9×9 matrices, we display their first five and last four columns separately below. Exact calculations verifying the feasibility of

these values are given in the associated publicly posted [Mathematica notebook](#).

$$\hat{\lambda}_{1:5} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 8837407518919583 & 5370688140802311 & 960254226721649 \\ 0 & 2191522964335457 & 4503599627370496 & 2305843009213693952 & 144115188075855872 \\ 0 & 2251799813685248 & 0 & 411829053855273 & 2688409565283275 \\ 0 & 6721678331720401 & 4407991053556385 & 2251799813685248 & 4503599627370496 \\ 0 & 2305843009213693952 & 18014398509481984 & 0 & 5801123626984329 \\ 0 & 1313681189860411 & 2695784755734549 & 8068866010524833 & 1125899906842624 \\ 0 & 1152921504606846976 & 2251799813685248 & 2251799813685248 & 0 \\ 0 & 1137203495150241 & 4688926225212825 & 4572972758977007 & 1182579142099203 \\ 0 & 4611686018427387904 & 9223372036854775808 & 4611686018427387904 & 1152921504606846976 \\ 0 & 5789750435206701 & 4956986009057139 & 5204247088958345 & 5375927246703545 \\ 0 & 18446744073709551616 & 9223372036854775808 & 4611686018427387904 & 9223372036854775808 \\ 0 & 7965115934238233 & 6653418691702949 & 4849791056907609 & 7288685819438951 \\ 0 & 36893488147419103232 & 9223372036854775808 & 4611686018427387904 & 9223372036854775808 \\ 0 & 1977810093374139 & 1222316311137735 & 6194623724653895 & 1504051934577545 \\ 0 & 9223372036854775808 & 1152921504606846976 & 4611686018427387904 & 1152921504606846976 \end{bmatrix},$$

$$\hat{\lambda}_{6:9} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 4697224493034383 & 8368394272075953 & 6128278626086993 & 80543723501136599 \\ 9223372036854775808 & 2305843009213693952 & 9223372036854775808 & 36893488147419103232 \\ 7733370871946025 & 5388133457257119 & 8119439550484479 & 19797186857495837 \\ 4611686018427387904 & 2305843009213693952 & 4611686018427387904 & 9223372036854775808 \\ 2740682573240027 & 529033377740781 & 4877381506142205 & 1248592962009775 \\ 576460752303423488 & 1152921504606846976 & 1152921504606846976 & 2305843009213693952 \\ 856652459407733 & 472316692024194 & 5356440120675597 & 339319020784307309 \\ 72057594037927936 & 18014398509481984 & 18014398509481984 & 1152921504606846976 \\ 0 & 772844649199827 & 2338413531710477 & 13381432557675475 \\ 6929007831194361 & 4503599627370496 & 288230376151711744 & 2305843009213693952 \\ 288230376151711744 & 0 & 8173541145090013 & 3837009581398546887 \\ 3923691798295005 & 1250438084247729 & 36028797018963968 & 18446744073709551616 \\ 144115188075855872 & 288230376151711744 & 0 & 18985683012247614283 \\ 13037057806369 & 7367859533233689 & 2876396710542189 & 36893488147419103232 \\ 2251799813685248 & 576460752303423488 & 288230376151711744 & 0 \end{bmatrix},$$

$$\hat{\gamma}_{1:5} = \begin{bmatrix} 0 & 1445047782665419 & 4403953050470099 & 12063929807726837 & 3158567943322891003 \\ 0 & 360287970189639680 & 36028797018963968 & 144115188075855872 & 144115188075855872 \\ 0 & 0 & -4876214136104831 & -8833525210676647 & -2173342829362743 \\ 0 & -3452744755754301 & 140737488355328 & -1125899906842624 & -281474976710656 \\ 0 & 70368744177664 & 0 & -6337493011708677 & -6175482448055731 \\ 0 & 5865189115672077 & -7694464236006067 & 36028797018963968 & 2251799813685248 \\ 0 & 36028797018963968 & 281474976710656 & -170650240960227 & 5761085953121451 \\ 0 & 8990085085489805 & 8180724221663923 & -5824324289872487 & -144115188075855872 \\ 0 & 562949953421312 & 9007199254740992 & -70368744177664 & 0 \\ 0 & 814636383991245 & 500238223044879 & -5824324289872487 & 5420042187723199 \\ 0 & 2251799813685248 & 9007199254740992 & -4503599627370496 & -2251799813685248 \\ 0 & 3576868018334213 & 3095322656598895 & -4741216870166437 & -8742107678118927 \\ 0 & 18014398509481984 & -18014398509481984 & -4503599627370496 & 18014398509481984 \\ 0 & 7467009600885335 & 7502387060304591 & 2299019529082251 & 5226016391129105 \\ 0 & 72057594037927936 & 18014398509481984 & -2251799813685248 & 4503599627370496 \\ 0 & 5171454268783955 & 4976089659881855 & 849946842273963 & 4041378073126239 \\ 0 & 18014398509481984 & 4503599627370496 & -1125899906842624 & 2251799813685248 \end{bmatrix},$$

$$\hat{\gamma}_{6:9} = \begin{bmatrix} 842771278878770475 & 1763107326300397405 & 490223898427757151 & 246036905476920105 \\ 288230376151711744 & 288230376151711744 & 72057594037927936 & 36028797018963968 \\ 7575374264747013 & -847735720154307 & -3644178695304033 & -2065817857485759 \\ -1125899906842624 & -140737488355328 & -562949953421312 & -281474976710656 \\ 256866650877453 & 2501054473773415 & 1008218866207467 & 6366578400890641 \\ 562949953421312 & -1125899906842624 & -281474976710656 & 2251799813685248 \\ 4819157742253121 & 8130105639853323 & 2019419188776927 & 2348470109478957 \\ 2251799813685248 & 18014398509481984 & 1125899906842624 & 281474976710656 \\ 451884763914337 & 7742823693376695 & 2908076698229235 & 6004115926843261 \\ 281474976710656 & 18014398509481984 & 2251799813685248 & 1125899906842624 \\ 0 & 3545908575737889 & 3082103085688847 & 3631847684938235 \\ -6392823462111415 & -288230376151711744 & 18014398509481984 & 1125899906842624 \\ -72057594037927936 & 0 & 226196870745667 & 5233938836038739 \\ -3454102047914875 & -2760923356849777 & 9007199254740992 & -2251799813685248 \\ -9007199254740992 & -36028797018963968 & 0 & -664739622472857 \\ -1957421956488181 & -7174137438907087 & -4503200630060789 & -2251799813685248 \\ 4503599627370496 & -4503599627370496 & 36028797018963968 & 0 \end{bmatrix}.$$