Mathematical Programming (2025) 213:1105–1118
https://doi.org/10.1007/s10107-024-02164-2

SHORT COMMUNICATION

Series A

Acceleration by stepsize hedging: Silver Stepsize Schedule for smooth convex optimization

Jason M. Altschuler[1] · **Pablo A. Parrilo**[2]

Received: 5 April 2024 / Accepted: 29 September 2024 / Published online: 25 November 2024
© The Author(s) 2024

Abstract
We provide a concise, self-contained proof that the Silver Stepsize Schedule proposed in our companion paper directly applies to smooth (non-strongly) convex optimization. Specifically, we show that with these stepsizes, gradient descent computes an ε-minimizer in $O(\varepsilon^{-\log_\rho 2}) = O(\varepsilon^{-0.7864})$ iterations, where $\rho = 1+\sqrt{2}$ is the silver ratio. This is intermediate between the textbook unaccelerated rate $O(\varepsilon^{-1})$ and the accelerated rate $O(\varepsilon^{-1/2})$ due to Nesterov in 1983. The Silver Stepsize Schedule is a simple explicit fractal: the i-th stepsize is $1+\rho^{\nu(i)-1}$ where $\nu(i)$ is the 2-adic valuation of i. The design and analysis are conceptually identical to the strongly convex setting in our companion paper, but simplify remarkably in this specific setting.

Keywords Convex optimization · Acceleration · Stepsize hedging · Silver Stepsize Schedule · Multi-step descent · Recursive gluing

Mathematics Subject Classification 90C25

1 Introduction

We revisit the classical problem of smooth convex optimization: solve $\min_{x\in\mathbb{R}^d} f(x)$ where f is convex and M-smooth (i.e., its gradient is M-Lipschitz). A celebrated result is that with a prudent choice of stepsizes $\{\alpha_t\}$, the gradient descent algorithm (GD)

✉ Jason M. Altschuler
alts@upenn.edu

✉ Pablo A. Parrilo
parrilo@mit.edu

[1] Department of Statistics and Data Science, University of Pennsylvania, Philadelphia, USA

[2] Laboratory for Information and Decision Systems (LIDS), Massachusetts Institute of Technology, Cambridge, USA

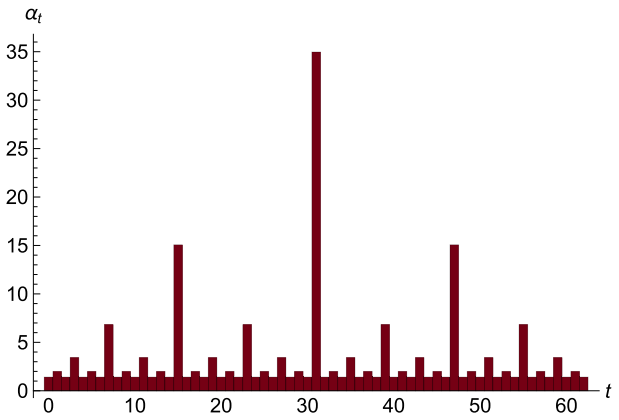


Fig. 1 Silver Stepsize schedule $\{\alpha_0, \alpha_1, \alpha_2, \ldots\}$. See (6) for the definition. Only the first $n = 63$ values are shown (i.e., $k = 8$). The fractal-like stepsizes are non-monotonic and have increasingly large "spikes" $\alpha_{2^k-1} = 1 + \rho^{k-1}$

$$x_{t+1} = x_t - \frac{\alpha_t}{M}\nabla f(x_t) \tag{1}$$

solves such a convex optimization problem to arbitrary accuracy from any initialization x_0. How quickly does GD converge? The mainstream approach (see e.g., the textbooks [1–6] among many others) is to use a constant stepsize schedule $\alpha_t \equiv \bar{\alpha} \in (0, 2)$ since this ensures

$$f(x_n) - f_* \leqslant \frac{cM\|x_0 - x_*\|^2}{n}, \tag{2}$$

where x_* denotes any minimizer of f, $f_* := f(x_*)$ denotes the corresponding minimal value, and c is a small constant, e.g., $c = \frac{1}{4}$ for $\bar{\alpha} = 1$ [7].

The main question posed in the companion paper [8] was: can we accelerate the convergence of GD without changing the algorithm—just by judiciously choosing the stepsizes? That paper investigated this question in the setting of smooth, strongly convex optimization. Here we continue to investigate this question, now in the setting of smooth (non-strongly) convex optimization. Note that this is markedly different from classical approaches to acceleration—starting from Nesterov's seminal result of 1983 [9], those approaches modify the basic GD algorithm by adding momentum, internal dynamics, or other additional building blocks beyond just changing the stepsizes. For this reason, we do not discuss that line of work in detail, and instead refer to [10] for a recent survey of this mainstream approach to acceleration, and to [8] for a full discussion of the relations between these approaches.

1.1 Contribution

This paper provides a concise, self-contained proof that the Silver Stepsize Schedule proposed in the companion paper [8] directly applies to smooth (non-strongly) convex optimization. This leads to faster convergence rates of GD for smooth convex optimization, as already pointed out in [8, Sect. 1.1.4].

In this setting, the Silver Stepsize Schedule is particularly simple. For any integer $n = 2^k - 1$, we recursively construct the schedule $h^{(2n+1)}$ of length $2n+1$ from the schedule $h^{(n)}$ of length n via

$$h^{(2n+1)} := [h^{(n)},\ 1 + \rho^{k-1},\ h^{(n)}]\,, \tag{3}$$

where $\rho := 1 + \sqrt{2}$ denotes the silver ratio and $h_1 := [\sqrt{2}]$. This results in the simple pattern $[\sqrt{2},\ 2,\ \sqrt{2},\ 1 + \sqrt{2}, \ldots]$ as depicted in Fig. 1. This schedule is exactly the Silver Stepsize Schedule from [8] in the limit that the strong convexity parameter vanishes (see Remark 1 for details), and bears similarities to those in [11–13]; see Sect. 1.2.

We show that these stepsizes yield an improved convergence rate (2) where $\frac{c}{n}$ is replaced by

$$r_k := \frac{1}{1 + \sqrt{4\rho^{2k} - 3}} \leqslant \frac{1}{2\rho^{\log_2 n}} = \frac{1}{2n^{\log_2 \rho}} \approx \frac{1}{2n^{1.2716}}\,. \tag{4}$$

This bound gives the correct asymptotic scaling of r_k since the inequality is asymptotically tight.

Theorem 1 (Main result) *For any horizon $n = 2^k - 1$, any dimension d, any M-smooth convex function $f : \mathbb{R}^d \to \mathbb{R}$, and any initialization $x_0 \in \mathbb{R}^d$,*

$$f(x_n) - f_* \leqslant r_k\, M \|x_0 - x_*\|^2\,, \tag{5}$$

where x_* denotes any minimizer of f, and x_n denotes the output of n steps of GD using the Silver Stepsize Schedule. In particular, in order to achieve error $f(x_n) - f_* \leqslant \varepsilon$, it suffices to run GD for some number of iterations n, where

$$n \leqslant \left(\frac{M\|x_0 - x_*\|^2}{2\varepsilon}\right)^{\log_\rho 2} \approx \left(\frac{M\|x_0 - x_*\|^2}{2\varepsilon}\right)^{0.7864}\,.$$

We make several remarks. (1) This rate $n^{-\log_2 \rho} \approx n^{-1.2716}$ is intermediate between the textbook unaccelerated rate n^{-1} and the accelerated rate n^{-2} due to Nesterov in 1983 [9]; see Fig. 2 for a visualization. (2) We conjecture that, among all possible stepsize schedules, the exponent $\log_2 \rho$ in the rate is exactly optimal, and the multiplicative constant $\frac{1}{2}$ in the rightmost-expression of (4) is within a modest constant factor of optimality. (3) For the Silver Stepsize Schedule, Theorem 1 shows that r_k is an upper bound on its convergence rate; this is essentially tight since there exist

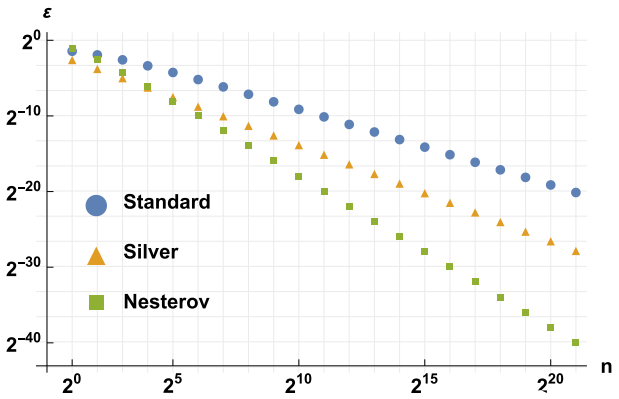


Fig. 2 Upper bound on the optimality gap $f(x_n) - f_*$, as a function of the number of iterations n, for $M\|x_0 - x_*\|^2 = 1$ (for other values, simply rescale everything by $M\|x_0 - x_*\|^2$). The three plots correspond to the rates $O(n^{-1})$ for the standard constant stepsize $\frac{1}{M}$ [3, Corollary 2.1.2], $O(n^{-\log_2 \rho}) \approx O(n^{-1.2716})$ for the Silver Stepsize Schedule (Theorem 1), and $O(n^{-2})$ for Nesterov acceleration [3, Theorem 2.2.2]

problem instances[1] for which the Silver Stepsize Schedule has rate $\frac{1}{4\rho^k - 2} \approx \frac{1}{4n^{1.2716}}$, which is within a factor of two of $r_k \approx \frac{1}{2n^{1.2716}}$. (4) The Silver Stepsize Schedule is independent of the horizon n, see Sect. 2. (5) The result holds for the final iterate x_n, which is more desirable than the averaged iterate—the standard alternative in optimization results. (6) Due to standard black-box reductions[2], this accelerated rate for smooth convex optimization immediately implies an analogously accelerated rate for smooth strongly-convex optimization by repeating, for large enough n, the n-step Silver Stepsize Schedule (3) for the present (non-strongly) convex setting. This asymptotically recovers the convergence rate of [8, Theorem 1.1], albeit with slightly worse constants non-asymptotically.

Corollary 2 *There exists a stepsize schedule such that for any dimension d, any m-strongly and M-smooth convex function $f : \mathbb{R}^d \to \mathbb{R}$, any initialization $x_0 \in \mathbb{R}^d$, and any precision $\varepsilon > 0$, then $\|x_n - x_*\|^2 \leqslant \varepsilon \|x_0 - x_*\|^2$ after*

$$n = O\left(\kappa^{\log_\rho 2} \log \frac{1}{\varepsilon}\right) = O\left(\kappa^{0.7864} \log \frac{1}{\varepsilon}\right)$$

iterations, where $\kappa := M/m$ denotes the condition number.

[1] Details: for any stepsize schedule, there is a 1-smooth convex (Huber loss) function f and initialization $x_0 \neq x_*$ such that $f(x_n) - f_* = (M\|x_0 - x_*\|^2)/(4\sum_{t=0}^{n-1} \alpha_t + 2)$. This fact is well-known and is, e.g., a straightforward extension of [7, Theorem 3.2]. For the Silver Stepsize Schedule, $\sum_{t=0}^{n-1} \alpha_t = \rho^k - 1$ by Lemma 4.

[2] Details: m-strong convexity implies $\frac{m}{2}\|x_n - x_*\|^2 \leqslant f(x_n) - f_*$, so by Theorem 1, $\|x_n - x_*\|^2 \leqslant 2\kappa r_k \|x_0 - x_*\|^2$ for any $n = 2^k - 1$. In particular, for a suitably large $k = O(\log_\rho \kappa)$, we have that $r_k \leqslant 1/(4\kappa)$, and therefore the distance to the optimum contracts by at least a factor of $1/2$ every n steps of the Silver Stepsize Schedule. Repeating this n-step schedule implies the convergence rate in Corollary 2. Notice that the resulting schedule is periodic of period $n = 2^k - 1 \asymp \kappa^{\log_\rho 2}$, similar to how the Silver Stepsize Schedule designed for the strongly convex setting [8] is approximately periodic of period $\kappa^{\log_\rho 2}$.

More broadly, this alternative mechanism for accelerated optimization opens up many directions for future work, such as robustness and generality to other settings and other algorithms; see the discussion in [8, Sect. 6]. We believe that the paradigm of stepsize hedging will be useful for the design and analysis of optimization algorithms in many contexts, and that the present setting of gradient descent is just scratching the surface of this algorithmic opportunity.

1.2 Related work

Several time-varying stepsize schedules have been considered, e.g., Armijo–Goldstein rules, Polyak-type schedules, Barzilai–Borwein-type schedules, etc. However, until recently, no convergence analyses improved over the textbook unaccelerated rate except in the special case of minimizing convex quadratics. Here we discuss the recent line of work on accelerated GD via time-varying stepsizes. For brevity, we refer to Paper I [8] for a more complete discussion of adjacent bodies of literature.

Beginning with Altschuler's 2018 MS thesis [14], a line of work designed time-varying stepsize schedules to achieve faster convergence rates for (non-quadratic) convex optimization. The thesis [14] showed that time-varying stepsizes can lead to improvements over the textbook unaccelerated GD rate—by an optimal asymptotic factor in the separable setting by using random stepsizes, and by a constant factor in the strongly convex setting by giving optimal stepsizes for $n = 2, 3$. The latter result was based on fitting numerical patterns observed from the SDP-based techniques of [7, 15] which compute the worst-case convergence rate for a *given* stepsize schedule. A key difficulty in the general non-separable setting is that the *search* for optimal stepsizes is non-convex and computationally difficult for larger horizons n.

Along similar lines, the Master's theses of Daccache [16] in 2019 and Eloi [17] in 2022 used these SDP-based techniques to compute optimal stepsizes for smooth, convex optimization for $n = 2, 3$ iterations. They also computed optimal stepsizes for smooth, strongly convex optimization for $n = 2, 3$ specifically for decreasing the objective value ([14] gave optimal schedules for minimizing distance to the optimum).

In 2022, Das Gupta et al. [11] combined Branch & Bound techniques with these SDP-based techniques to develop algorithms that perform this search for optimal stepsize schedules numerically, and as an example computed good approximate schedules in the convex setting for larger values of n up to 50. They observed a fit of roughly $O(n^{-1.178})$ and suggested from this that the asymptotic rate may be faster than the textbook rate $O(n^{-1})$.

In July 2023, Grimmer [12] showed how to prove asymptotic rates for the (non-strongly) convex setting by periodically cycling through finite schedules. While in the strongly convex setting composing progress from different cycles just amounts to multiplying contraction rates, in the non-strongly convex setting this can be more subtle depending on the approach.[3] To deal with this, he introduced the notion of straightforward stepsize patterns and obtained a constant-factor improvement over the

[3] Cf., our recursive gluing approach, which is a simple way of composing progress that unifies the convex (in Sect. 3) and strongly convex settings (in Paper I). This is why our analysis is so compact and completely bypasses the machinery of straightforward patterns. See Remark 2.

textbook unaccelerated rate by cycling through approximate schedules of length $n = 127$. From these numerics, he conjectured that further optimizing stepsizes might lead to an asymptotic rate of $O((n \log n)^{-1})$, a milder improvement than that conjectured by [11].

In September 2023, two concurrent lines of work appeared. One line of work was the two companion papers of the present authors—Altschuler and Parrilo [8] and the present paper. The other was the paper Grimmer et al. [13]. Altschuler and Parrilo [8] proposed the Silver Stepsize Schedule of arbitrary size to prove asymptotic acceleration for the strongly convex setting. This improved the dependence on the condition number κ in the textbook unaccelerated rate $O(\kappa)$ to $O(\kappa^{\log_\rho 2}) \approx O(\kappa^{0.7864})$. (All these bounds scale in the precision as $\log 1/\varepsilon$.) We conjectured and provided partial evidence that these rates are optimal among all possible stepsize schedules. This result was achieved by introducing the technique of recursive gluing to establish multi-step descent, which we make use of here. The other paper, Grimmer et al. [13], utilized a certain non-periodic sequence of increasingly large stepsizes and built upon the "straightforwardness" machinery in [12], to prove the rate $O(n^{-1.0245})$ for the convex setting in v1, later improved to $O(n^{-1.0564})$ in v2.

The results in [8] were stated for the strongly convex setting. As pointed out throughout that paper, this is not a restriction because on the one hand those results immediately imply analogously accelerated rates $O(n^{-\log_2 \rho}) \approx O(n^{-1.2716})$ for the convex setting via a standard black-box reduction; and on the other hand, this reduction can be bypassed by re-doing the analysis for the convex setting since all the core conceptual ideas extend directly [8, Sect. 1.1.4]. The present paper provides these details.

1.3 Notation

Indexing. Throughout, the horizon is $n = 2^k - 1$. This correspondence between $n \in \{1, 3, 7, \dots\}$ and $k \in \{1, 2, 3, \dots\}$ allows re-indexing in a way that simplifies notation for the recursion.

Rescaling. For simplicity, we normalize $M = 1$, i.e., $\|\nabla f(x) - \nabla f(y)\| \leqslant \|x - y\|$ for all x, y. This is without loss of generality since if h is M-smooth and convex, then $f := h/M$ is 1-smooth and convex, thus the result we establish $f_n - f_* \leqslant r_k \|x_0 - x_*\|^2$ implies $h_n - h_* \leqslant r_k M \|x_0 - x_*\|^2$. Note that running GD on f simply amounts to rescaling the Silver Stepsize Schedules for h by $1/M$. As such, we slightly abuse notation throughout by referring to the α_t simply as stepsizes rather than normalized stepsizes.

2 Silver Stepsize Schedule

The Silver Stepsize Schedule is defined recursively in (3). Here we mention an equivalent direct expression and make several remarks. Let $\nu(t)$ denote the 2-adic valuation of t, i.e., the largest non-negative integer i such that 2^i divides t. For example, $\nu(1) = 0$, $\nu(2) = 1$, $\nu(3) = 0$, $\nu(4) = 2$, etc.

Definition 3 (Silver Stepsize Schedule for smooth convex optimization) For $t \in \{0, 1, 2, \dots\}$, the t-th stepsize of the Silver Stepsize Schedule is

$$\alpha_t := 1 + \rho^{\nu(t+1)-1}. \tag{6}$$

This schedule is non-monotonic, fractal-like, and has increasingly large spikes that grow exponentially (by a factor of ρ) yet become exponentially less frequent (by a factor of 2). See Fig. 1. It can also be easily implemented[4] in any computer language.

Remark 1 (Limit of Silver Stepsize Schedules in the strongly convex case) The schedule (6) is simply the Silver Stepsize Schedule for smooth *strongly*-convex optimization in the companion paper [8], in the limit as the strong convexity parameter tends to 0. The stepsizes simplify in the limit: they increase by a factor of ρ, shorter schedules are prefixes of longer schedules, neither a_1 nor any of the b_n sequence in [8] is needed, and they are non-periodic (hence why there is no rate saturation here).

We also record a simple expression for the sum of the first $n = 2^k - 1$ Silver Stepsizes.

Lemma 4 *(Sum of Silver Stepsizes)* $\sum_{t=0}^{n-1} \alpha_t = \rho^k - 1$ *for any* $k \in \mathbb{N}$.

Proof The base case $k = 1$ is trivial. The inductive step follows from the recursion (3) and the silver ratio property $\rho^2 = 2\rho + 1$. □

3 Recursive gluing

Here we prove that the Silver Stepsize Schedule has convergence rate r_k for smooth convex optimization. The analysis closely mirrors the strongly convex setting in the companion paper [8]: we prove the advantage of time-varying stepsizes via multi-step descent rather than iterating the greedy 1-step bound, show multi-step descent by exploiting long-range consistency conditions along the GD trajectory, certify multi-step descent via recursive gluing, and recursively glue by combining the same three components in the same way. In the interest of brevity, we refer to [8] for a detailed discussion of all these concepts.

Briefly, the idea behind multi-step descent is that it is essential to capture how different iterations affect other iterations' progress. We do this by exploiting long-range consistency conditions between the iterates along GD's trajectory, as encoded by the co-coercivities

$$Q_{ij} := 2(f_i - f_j) + 2\langle g_j, x_j - x_i\rangle - \|g_j - g_i\|^2. \tag{7}$$

The significance of these co-coercivities is that the constraints $\{Q_{ij} \geqslant 0\}_{i \neq j \in \{0,1,\dots,n,*\}}$ are necessary and sufficient for the existence of a 1-smooth convex function f satisfying $f_i = f(x_i)$ and $g_i = \nabla f(x_i)$ for each $i \in \{0, 1, \dots, n, *\}$ [15]. In other

[4] For instance, in Python the command `[1+rho**((k & -k).bit_length()-2) for k in range(1,64)]` generates the first 63 steps of the Silver Schedule shown in Fig. 1.

words, the co-coercivity conditions $\{Q_{ij} \geqslant 0\}_{i \neq j \in \{0,1,\ldots,n,*\}}$ generate all possible long-range consistency constraints on the objective function f. Or, said another way, the co-coercivity conditions generate all possible valid inequalities from which one can prove convergence rates for GD. For a further discussion, see [8, Sect. 2.2].

Concretely, to prove Theorem 1, we exhibit non-negative multipliers λ_{ij} satisfying

$$\sum_{i \neq j \in \{0,\ldots,n,*\}} \lambda_{ij} Q_{ij} = \|x_0 - x_*\|^2 - \|x_n - c_k g_n - x_*\|^2 + \frac{f_* - f_n}{r_k}, \tag{8}$$

where we use the shorthand $c_k := \frac{1}{2r_k}$. Since $\sum_{ij} \lambda_{ij} Q_{ij} \geqslant 0$ for any 1-smooth convex function, and since $\|x_n - c_k g_n - x_*\|^2 \geqslant 0$ trivially as it is a square, this immediately implies the desired rate

$$f_n - f_* \leqslant r_k \|x_0 - x_*\|^2 . \tag{9}$$

Remark 2 (The importance of a composable formulation) We emphasize that while there are several alternative formulations to (8) that imply a rate like (9) after dropping terms, our formulation (8) is "canonical" because it composes well under recurrence (Theorem 6). This enables a short and simple proof. Another reason this formulation is particularly nice is because the base case $n = 0$ amounts to a re-writing of the definition of Q_{*0}, which is an improvement over the standard bound $f_0 - f_* \leqslant \frac{1}{2}\|x_0 - x_*\|^2$. In fact, this improvement is precisely what makes recursive gluing work so seamlessly.

Example 1 ($n = 0$) For $n = 0$, the identity (8) is

$$Q_{*0} = \|x_0 - x_*\|^2 - \|x_0 - g_0 - x_*\|^2 + 2(f_* - f_0) .$$

Here, $r_0 = \frac{1}{2}$, $c_0 = 1$, and the only non-zero multiplier is $\lambda_{*0} = 1$.

Example 2 ($n = 1$) For $n = 1$, the identity (8) is

$$\sum_{i \neq j \in \{0,1,*\}} \lambda_{ij} Q_{ij} = \|x_0 - x_*\|^2 - \|x_1 - c_1 g_1 - x_*\|^2 + \frac{f_* - f_1}{r_1} .$$

Here, $x_1 = x_0 - \sqrt{2} g_0$, $r_1 = \frac{1}{1+\sqrt{4\rho^2 - 3}} \approx 0.1816$, $c_1 = \frac{1}{2r_1} \approx 2.7535$, and

$$\begin{bmatrix} \lambda_{00} & \lambda_{01} & \lambda_{0*} \\ \lambda_{10} & \lambda_{11} & \lambda_{1*} \\ \lambda_{*0} & \lambda_{*1} & \lambda_{**} \end{bmatrix} = \begin{bmatrix} 0 & 1+\sqrt{2} & 0 \\ 1 & 0 & \sqrt{2} \\ \sqrt{2} & \frac{1}{2r_1} & 0 \end{bmatrix} .$$

For larger horizons n, we construct λ_{ij} via the recursive gluing technique of [8]. See Fig. 3 for an illustration. In order to state this recursive gluing result, it is convenient to first define a certain sparsity pattern for the multipliers; this is satisfied by construction and simplifies the recursive proof (isolated in Lemma 7).

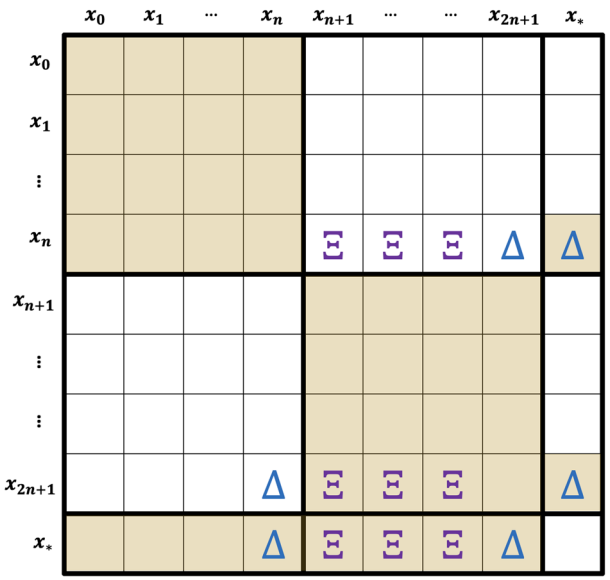


Fig. 3 Components of the recursively glued certificate in Theorem 6, illustrated here for combining two copies of the $n = 3$ certificate (shaded) to create the $2n + 1 = 7$ certificate. The structure of this recursive gluing is identical to the strongly convex setting from the companion paper [8], modulo re-indexing for horizons of the form $n = 2^k - 1$ rather than 2^k

Definition 5 ($*$-sparsity property) A collection of multipliers $\{\sigma_{ij}\}_{i,j\in\{0,\ldots,n,*\}}$ satisfies the $*$-sparsity property if $\sigma_{i*} = 0$ for all $i < n$.

Theorem 6 *(Recursive gluing) Let $n = 2^k - 1$. Suppose $\{\sigma_{ij}\}_{i,j\in\{0,\ldots,n,*\}}$ are non-negative, satisfy $*$-sparsity, and certify the n-step rate, i.e.,*

$$\sum_{i\neq j\in\{0,\ldots,n,*\}} \sigma_{ij} Q_{ij} = \|x_0 - x_*\|^2 - \|x_n - c_k g_n - x_*\|^2 + \frac{f_* - f_n}{r_k}. \tag{10}$$

Then there exists $\{\lambda_{ij}\}_{i,j\in\{0,\ldots,2n+1,*\}}$ that are non-negative, satisfy $*$-sparsity, and certify the $2n + 1$-step rate, i.e.,

$$\sum_{i\neq j\in\{0,\ldots,2n+1,*\}} \lambda_{ij} Q_{ij} = \|x_0 - x_*\|^2 - \|x_{2n+1} - c_{k+1} g_{2n+1} - x_*\|^2 + \frac{f_* - f_{2n+1}}{r_{k+1}}. \tag{11}$$

Moreover, this certificate is explicitly given by

$$\lambda_{ij} = \underbrace{\Theta_{ij}}_{\text{gluing}} + \underbrace{\Xi_{ij}}_{\text{rank-one correction}} + \underbrace{\Delta_{ij}}_{\text{sparse correction}}. \tag{12}$$

The "gluing component" Θ is defined as

$$\Theta_{ij} := \underbrace{\sigma_{i,j} \cdot \mathbb{1}_{i,j\in\{0,\ldots,n,*\}}}_{\text{recurrence for first } n \text{ steps}} + \underbrace{(1+2\rho)\,\sigma_{i-n-1,j-n-1} \cdot \mathbb{1}_{i,j\in\{n+1,\ldots,2n+1,*\}}}_{\text{recurrence for last } n \text{ steps}}. \tag{13}$$

The "rank-one correction" Ξ is zero except the entries $\{\Xi_{ij}\}_{i\in\{n,2n+1,*\},\ j\in\{n+1,\ldots,2n\}}$ which are

$$\begin{bmatrix} \Xi_{n,n+1} & \Xi_{n,n+2} & \cdots & \Xi_{n,2n} \\ \Xi_{2n+1,n+1} & \Xi_{2n+1,n+2} & \cdots & \Xi_{2n+1,2n} \\ \Xi_{*,n+1} & \Xi_{*,n+2} & \cdots & \Xi_{*,2n} \end{bmatrix} := \rho \begin{bmatrix} 1 \\ 1 \\ -2 \end{bmatrix} \begin{bmatrix} \alpha_{n+1} & \alpha_{n+2} & \cdots & \alpha_{2n} \end{bmatrix}. \tag{14}$$

The "sparse correction" Δ is zero except the entries $\{\Delta_{ij}\}_{i\neq j\in\{n,2n+1,*\}}$ which are

$$\begin{bmatrix} \Delta_{n,n} & \Delta_{n,2n+1} & \Delta_{n,*} \\ \Delta_{2n+1,n} & \Delta_{2n+1,2n+1} & \Delta_{2n+1,*} \\ \Delta_{*,n} & \Delta_{*,2n+1} & \Delta_{*,*} \end{bmatrix} := \begin{bmatrix} 0 & \rho & 1-\rho^k \\ \rho^k & 0 & 2\rho - \sqrt{2}\rho^{k+1} \\ 1+\rho^{k-1} - \frac{1}{2r_k} & \frac{1}{2r_{k+1}} - \frac{1+2\rho}{2r_k} & 0 \end{bmatrix}. \tag{15}$$

We remark that using this recursion, one can also write the n-step certificate directly. The value of each multiplier depends on the binary expansion of its indices.

This recursive gluing immediately implies the main result of the paper, Theorem 1.

Proof of Theorem 1 It suffices to prove (8); we prove this by induction. The base case $n = 0$ is Example 1. (Note that the induction gives the case $n = 1$ in Example 2.) The inductive step is Theorem 6. □

The rest of the section proves Theorem 6. We first isolate two helper lemmas in Sect. 3.1, and then we combine them to prove the result in Sect. 3.2. For notational simplicity, henceforth we assume $x_* = 0$; this is without loss of generality after translating.

3.1 Helper lemmas

Here we provide three helper lemmas for the proof of Theorem 6. The first lemma explicitly computes all multipliers involving x_* for the n-step certificate.

Lemma 7 (Multipliers involving x_*) *$\sigma_{n*} = \rho^k - 1$, $\sigma_{*n} = \frac{1}{2r_k}$, and $\sigma_{*t} = \alpha_t$ for $t \in \{0, \ldots, n-1\}$.*

Proof Expand both sides of the identity (10) using $*$-sparsity, the definition of the co-coercivities, and the definition of GD, i.e., $x_t = x_0 - \sum_{s=0}^{t-1} \alpha_s g_s$. Matching coefficients for the $\langle x_0, g_t \rangle$ terms yields the claimed formulas for σ_{*t}. Matching coefficients for the f_* term gives the identity $\sigma_{n*} = \sum_{j=0}^{n} \sigma_{*j} - \frac{1}{2r_k}$. This equals $\rho^k - 1$ by Lemma 4.

For example, in more detail, matching coefficients for the f_* term in (10) yields the identity $2(\sum_{j=0}^{n} \sigma_{*j} - \sum_{i=0}^{n} \sigma_{i*}) = \frac{1}{r_k}$ since by definition of the co-coercivities,

the only terms $\sigma_{ij} Q_{ij}$ that contribute to the coefficient of f_* are those for which either i or j is $*$. By $*$-sparsity, $\sigma_{i*} = 0$ for all $i < n$. Dropping those terms and re-arranging yields the aforementioned identity $\sigma_{n*} = \sum_{j=0}^{n} \sigma_{*j} - \frac{1}{2r_k}$. Solving for the other expressions is similar, as described above. □

The next two lemmas are more substantial. These help us verify (11)—which consists of a linear form in all $2n + 3$ function values and a quadratic form in all $2n + 3$ gradients and iterates. Naïvely verifying such an identity requires checking $\Theta(n)$ coefficients for the linear form and $\Theta(n^2)$ coefficients for the quadratic form. The following two lemmas show that due to the recursive construction of the Silver Stepsize Schedule and the gluing, these forms only affect the indices $n, 2n + 1, *$. This reduces verifying the rate certificate (11) to checking only $\Theta(1)$ coefficients, as detailed below in Sect. 3.2. Below, for shorthand, let $F_{ij} := 2(f_i - f_j)$ and $P_{ij} := 2\langle g_j, x_j - x_i\rangle - \|g_i - g_j\|^2$ denote the linear and quadratic components of Q_{ij}, respectively. For bookkeeping purposes, we use 3-dimensional vectors and 4×4 matrices to denote the coefficients of these forms. The proofs of these lemmas are deferred to the appendix for brevity.

Lemma 8 *(Succinct linear forms) Let* $u := [f_n, f_{2n+1}, f_*]^T$*, and let* $e, s, \ell \in \mathbb{R}^3$ *be the vectors defined in Appendix A.1.*

- *Gluing error:* $\frac{f_* - f_{2n+1}}{r_{k+1}} - \sum_{ij} \Theta_{ij} F_{ij} = \langle e, u\rangle$.
- *Sparse correction:* $\sum_{ij} \Delta_{ij} F_{ij} = \langle s, u\rangle$.
- *Rank-one correction:* $\sum_{ij} \Xi_{ij} F_{ij} = \langle \ell, u\rangle$.

Lemma 9 *(Succinct quadratic forms) Let* $v := [x_n, g_n, x_{2n+1}, g_{2n+1}]^T$*, and let* E*,* S*,* L *be the* 4×4 *matrices defined in Appendix A.2.*

- *Gluing error:* $\|x_0\|^2 - \|x_{2n+1} - c_{k+1} g_{2n+1}\|^2 - \sum_{ij} \Theta_{ij} P_{ij} = \langle E, vv^T\rangle$.
- *Sparse correction:* $\sum_{ij} \Delta_{ij} P_{ij} = \langle S, vv^T\rangle$.
- *Rank-one correction:* $\sum_{ij} \Xi_{ij} P_{ij} = \langle L, vv^T\rangle$.

3.2 Proof of recursive gluing (Theorem 6)

Non-negativity. We verify $\lambda_{ij} \geqslant 0$ for all $i \neq j \in \{0, \ldots, 2n + 1, *\}$. For nearly all entries, this is obvious since λ_{ij} is constructed by adding and multiplying non-negative numbers. For the remaining entries where either Ξ or Δ is negative, compute the corresponding entry of λ by summing the corrections and using Lemma 7. This gives $\lambda_{n*} = 0$, $\lambda_{2n+1,*} = \rho^{k+1} - 1$, $\lambda_{*,n} = \rho^{k-1} + 1$, $\lambda_{*,2n+1} = \frac{1}{2r_{k+1}}$, and $\lambda_{*,t} = \alpha_t$ for all $t \in \{n + 1, \ldots, 2n\}$. All these entries are clearly non-negative.

Rate certificate. The identity (11) has two components: a linear form in the function values and a quadratic form in the iterates and gradients. For the linear form, it suffices to verify $e - s - \ell = 0$ by Lemma 8, where $e, s, \ell \in \mathbb{R}^3$ are the vectors defined in Appendix A.1. This is obvious by inspection. For the quadratic form, it suffices to verify $E - S - L = 0$, where $E, S, L \in \mathbb{R}^{4\times 4}$ are the matrices in Lemma 9 defined in Appendix A.2. By plugging in the explicit values for r_k and Δ, this is straightforward to check by hand. For brevity, we provide a simple Mathematica script that verifies these identities at the URL [18].

Acknowledgements We thank the anonymous reviewers for helpful comments, especially for suggesting we elaborate on implications for the strongly convex setting (Corollary 2) and the near-optimality of the Silver Convergence Rate (Footnote 1).

Funding Open Access funding provided by the MIT Libraries.

Declarations

Conflict of interest The authors have no financial or non-financial Conflict of interest.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit http://creativecommons.org/licenses/by/4.0/.

Appendix A Deferred proof details

A.1 Succinct linear forms (Lemma 8)

The vectors e, s, ℓ in Lemma 8 are defined as follows:

$$e := \begin{bmatrix} \frac{1}{r_k} \\ -\frac{1}{r_{k+1}} + \frac{1+2\rho}{r_k} \\ \frac{1}{r_{k+1}} - \frac{2+2\rho}{r_k} \end{bmatrix}, \qquad s := \begin{bmatrix} \frac{1}{r_k} - 2\rho(\rho^k - 1) \\ -\frac{1}{r_{k+1}} + \frac{1+2\rho}{r_k} - 2\rho(\rho^k - 1) \\ \frac{1}{r_{k+1}} - \frac{2+2\rho}{r_k} + 4\rho(\rho^k - 1) \end{bmatrix},$$

$$\ell := 2\rho(\rho^k - 1) \begin{bmatrix} 1 \\ 1 \\ -2 \end{bmatrix}.$$

Proof of Lemma 8 Expand the co-coercivities using the definition (7) and collect terms corresponding to f_n, f_{2n+1}, f_*. To obtained the simplified forms above, simplify the sparse correction using the Pell recurrence $\rho^{k+1} = 2\rho^k + \rho^{k-1}$, simplify the rank-one correction using Lemma 4, and simplify the gluing component using the identities $2\sum_{j\neq *}(\sigma_{*j} - \sigma_{j*}) = \frac{1}{r_k}$ and $2\sum_{j\neq n}(\sigma_{nj} - \sigma_{jn}) = -\frac{1}{r_k}$ which respectively follow from matching the f_* and f_n coefficients on both sides of (10). For example, the first entry of e is the coefficient of f_n in $\sum_{ij}\Theta_{ij}F_{ij}$, namely $2\sum_{j\neq n}(\sigma_{jn} - \sigma_{nj}) = \frac{1}{r_k}$. □

A.2 Succinct quadratic forms (Lemma 9)

The matrices in Lemma 9 are defined as follows. For brevity, we write $\sim$ for the entries below the diagonal since these coefficient matrices are symmetric.

$$E := \begin{bmatrix} -2\rho & (1+2\rho)(1+\rho^{k-1}) - \frac{1}{2r_k} & 0 & 0 \\ \sim & \frac{1}{4r_k^2} - (1+2\rho)(1+\rho^{k-1})^2 & 0 & 0 \\ \sim & \sim & 2\rho & \frac{1}{2r_{k+1}} - \frac{1+2\rho}{2r_k} \\ \sim & \sim & \sim & \frac{1+2\rho}{4r_k^2} - \frac{1}{4r_{k+1}^2} \end{bmatrix},$$

$$S := \begin{bmatrix} 0 & \Delta_{2n+1,n} + \Delta_{*,n} & 0 & -\Delta_{n,2n+1} \\ \sim & -\left(\Delta_{n,*} + \Delta_{*,n} + \Delta_{2n+1,n} + \Delta_{n,2n+1}\right) & -\Delta_{2n+1,n} & \Delta_{n,2n+1} + \Delta_{2n+1,n} \\ \sim & \sim & 0 & \Delta_{n,2n+1} + \Delta_{*,2n+1} \\ \sim & \sim & \sim & -\left(\Delta_{n,2n+1} + \Delta_{2n+1,n} + \Delta_{2n+1,*} + \Delta_{*,2n+1}\right) \end{bmatrix},$$

$$L := \rho \begin{bmatrix} -2 & 2+\rho^{k-1} & 0 & 1 \\ \sim & -1-\rho^k-2\rho^{k-1} & \rho^{k-1} & -1-\rho^{k-1} \\ \sim & \sim & 2 & -1 \\ \sim & \sim & \sim & 1-\rho^k \end{bmatrix}.$$

Proof of Lemma 9 Gluing component. Recall that $h^{(2n+1)} = [h^{(n)}, \alpha_n, h^{(n)}]$ by the recursive construction (3) of the Silver Stepsize Schedule. Thus, since σ certifies the n-step rate, we have

$$\sum_{i \neq j \in \{0,\ldots,n,*\}} \sigma_{ij} P_{ij} = \|x_0\|^2 - \|x_n - c_k g_n\|^2,$$

$$\sum_{i \neq j \in \{n+1,\ldots,2n,*\}} \sigma_{i-n-1,j-n-1} P_{ij} = \|x_{n+1}\|^2 - \|x_{2n+1} - c_k g_{2n+1}\|^2.$$

By definition, $\sum_{ij} \Theta_{ij} P_{ij}$ is the former plus $(1+2\rho)$ times the latter. Thus the gluing error $\|x_0\|^2 - \|x_{2n+1} - c_{k+1} g_{2n+1}\|^2 - \sum_{ij} \Theta_{ij} P_{ij}$ is equal to

$(1+2\rho)\|x_{2n+1} - c_k g_{2n+1}\|^2 - \|x_{2n+1} - c_{k+1} g_{2n+1}\|^2 + \|x_n - c_k g_n\|^2 - (1+2\rho)\|x_{n+1}\|^2$.

The key point is that after expanding $x_{n+1} = x_n - \alpha_n g_n$, this is a quadratic form in only the 4 variables x_n, g_n, x_{2n+1}, g_{2n+1}. Collecting terms, substituting $c_k = \frac{1}{2r_k}$, and using the definition of the Silver Stepsize $\alpha_n = 1 + \rho^{k-1}$ yields $\langle E, vv^T \rangle$, as desired.

Sparse component. Expand the definition of the co-coercivity and collect terms. Note also that this equals the matrix S in the companion paper [8], in the limit that the strong convexity tends to 0.

Low-rank component. Expanding the co-coercivities, using the identity $x_{2n+1} - x_{n+1} = -\sum_{t=n+1}^{2n} \alpha_t g_t$ by definition of GD, and using the identity $\sum_{t=n+1}^{2n} \alpha_t = \sum_{t=0}^{n-1} \alpha_t = \rho^k - 1$ from Lemma 4,

$$\sum_{t=n+1}^{2n}\left(\Xi_{n,t}P_{n,t}-\Xi_{*,t}P_{*,t}\right)=\rho\sum_{t=n+1}^{2n}\alpha_t\left(2\langle g_t,g_n-x_n\rangle-\|g_n\|^2\right)$$
$$=2\rho\langle x_{2n+1}-x_{n+1},x_n-g_n\rangle-\rho(\rho^k-1)\|g_n\|^2.$$

An identical calculation yields

$$\sum_{t=n+1}^{2n}\left(\Xi_{2n+1,t}P_{2n+1,t}-\Xi_{*,t}P_{*,t}\right)=2\rho\langle x_{2n+1}-x_{n+1},x_{2n+1}-g_{2n+1}\rangle$$
$$s-\rho(\rho^k-1)\|g_{2n+1}\|^2.$$

Now sum these displays, use the definition of GD to expand $x_{n+1}=x_n-\alpha_n g_n$, and use the definition of the Silver Stepsize $\alpha_n=1+\rho^{k-1}$. This gives $\langle L,vv^T\rangle$, as desired. □

References

1. Bertsekas, D.P.: Nonlinear programming. Athena Scientific, Belmont, MA (1999)
2. Nocedal, J., Wright, S.J.: Numerical optimization. Springer, New York, NY (1999)
3. Nesterov, Y.: Introductory lectures on convex optimization: a basic course, vol. 87. Springer, Boston, MA (1998)
4. Bubeck, S.: Convex optimization: algorithms and complexity. Found. Trends Mach. Learn. **8**(3–4), 231–357 (2015)
5. Bertsekas, D.P., Nedić, A., Ozdaglar, A.E.: Convex analysis and optimization, p. 534. Athena Scientific, Belmont, MA (2003)
6. Lan, G.: First-order and stochastic optimization methods for machine learning, vol. 1. Springer, New York, NY (2020)
7. Drori, Y., Teboulle, M.: Performance of first-order methods for smooth convex minimization: a novel approach. Math. Program. **145**(1–2), 451–482 (2014)
8. Altschuler, J.M., Parrilo, P.A.: Acceleration by stepsize hedging I: multi-step descent and the Silver Stepsize Schedule. arXiv preprint arXiv:2309.07879 (2023)
9. Nesterov, Y.E.: A method of solving a convex programming problem with convergence rate $O(1/k^2)$. Soviet Math. Dokl. **27**(2), 372–376 (1983)
10. d'Aspremont, A., Scieur, D., Taylor, A.: Acceleration methods. Found. Trends Optim. **5**(1–2), 1–245 (2021)
11. Das Gupta, S., Van Parys, B.P., Ryu, E.K.: Branch-and-bound performance estimation programming: a unified methodology for constructing optimal optimization methods. Math. Program. **204**, 1–73 (2023)
12. Grimmer, B.: Provably faster gradient descent via long steps. arXiv preprint arXiv:2307.06324 (2023)
13. Grimmer, B., Shu, K., Wang, A.L.: Accelerated gradient descent via long steps. arXiv preprint arXiv:2309.09961 (2023)
14. Altschuler, J.M.: Greed, hedging, and acceleration in convex optimization. Master's thesis, Massachusetts Institute of Technology (2018)
15. Taylor, A.B., Hendrickx, J.M., Glineur, F.: Smooth strongly convex interpolation and exact worst-case performance of first-order methods. Math. Program. **161**(1–2), 307–345 (2017)
16. Daccache, A.: Performance estimation of the gradient method with fixed arbitrary step sizes. PhD thesis, Master's thesis, Université Catholique de Louvain (2019)
17. Eloi, D.: Worst-case functions for the gradient method with fixed variable step sizes. PhD thesis, Master's thesis, Université Catholique de Louvain (2022)
18. Acceleration by Stepsize Hedging – verification of identities computer algebra script. https://jasonaltschuler.github.io/AccelerationByStepsizeHedging

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.