Small errors in random zeroth-order optimization are imaginary

Wouter Jongeneel[†] Man-Chung Yue[‡] Daniel Kuhn[†]

[†]Risk Analytics and Optimization Chair, École Polytechnique Fédérale de Lausanne, {wouter.jongeneel,daniel.kuhn}@epfl.ch
[‡]Musketeers Foundation Institute of Data Science and Department of Industrial and Manufacturing Systems Engineering, The University of Hong Kong, mcyue@hku.hk

December 15, 2023

Abstract

Most zeroth-order optimization algorithms mimic a first-order algorithm but replace the gradient of the objective function with some gradient estimator that can be computed from a small number of function evaluations. This estimator is constructed randomly, and its expectation matches the gradient of a smooth approximation of the objective function whose quality improves as the underlying smoothing parameter δ is reduced. Gradient estimators requiring a smaller number of function evaluations are preferable from a computational point of view. While estimators based on a single function evaluation can be obtained by use of the divergence theorem from vector calculus, their variance explodes as δ tends to 0. Estimators based on multiple function evaluations, on the other hand, suffer from numerical cancellation when δ tends to 0. To combat both effects simultaneously, we extend the objective function to the complex domain and construct a gradient estimator that evaluates the objective at a complex point whose coordinates have small imaginary parts of the order δ. As this estimator requires only one function evaluation, it is immune to cancellation. In addition, its variance remains bounded as δ tends to 0. We prove that zeroth-order algorithms that use our estimator offer the same theoretical convergence guarantees as the state-of-the-art methods. Numerical experiments suggest, however, that they often converge faster in practice.

Keywords—zeroth-order optimization, derivative-free optimization, complex-step derivative.
AMS Subject Classification (2020)—65D25 · 65G50 · 65K05 · 65Y04 · 65Y20 · 90C56.

1 Introduction

We study optimization problems of the form

$$\underset{x\in\mathcal{X}}{\text{minimize}} \quad f(x), \tag{1.1}$$

where $f : \mathcal{D} \to \mathbb{R}$ is a real analytic and thus smooth objective function defined on an open set $\mathcal{D} \subseteq \mathbb{R}^n$, and $\mathcal{X} \subseteq \mathcal{D}$ is a non-empty closed feasible set. Throughout the paper we assume that problem (1.1) admits a global minimizer $x^\star$ and that the objective function f can only be accessed through a deterministic zeroth-order oracle, which outputs function

evaluations at prescribed test points. Under this premise, we aim to develop optimization algorithms that generate a (potentially randomized) sequence of iterates $x_1, x_2, \ldots, x_K \in \mathcal{X}$ approximating $x^\star$. As they only have access to a zeroth-order oracle, these algorithms fall under the umbrella of *zeroth-order optimization*, *derivative-free optimization* or, more broadly, *black-box optimization*, see, *e.g.,* [AH17]. As we will explain below and in contrast to all prior work on zeroth-order optimization, we will assume that our zeroth-order oracle also accepts *complex* inputs beyond $\mathcal{D}$.

Zeroth-order optimization algorithms are needed when problem (1.1) cannot be addressed with first- or higher-order methods. This is the case when there is no simple closed-form expression for f and its partial derivatives or when evaluating the gradient of f is expensive. In simulation-based optimization, for example, the function f can be evaluated via offline or online simulation methods, but its gradient is commonly inaccessible. Zeroth-order optimization algorithms can also be used for addressing minimax, bandit or reinforcement learning problems, and they lend themselves for hyperparameter tuning in supervised learning [Spa05; CSV09; NS17]. As they can only access function values, zeroth-order optimization methods are inevitably somewhat crude. This simplicity is both a curse and a blessing. On the one hand, it has a detrimental impact on the algorithms' ability to converge to local minima, on the other hand—and this requires further formalization [Sch22], it may enable zeroth-order methods to escape from saddle points and thus makes them attractive for non-convex optimization.

Zeroth-order optimization algorithms can be categorized into direct search methods, model-based methods and random search methods [LMW19]. Direct search methods evaluate the objective function at a set of trial points without the goal of approximating the gradient. A representative example of a direct search method is the popular Nelder–Mead algorithm [NM65]. Model-based methods use zeroth-order information acquired in previous iterations to calibrate a C^r-smooth model for some $r \in \mathbb{Z}_{\geq 0}$ that approximates the black-box function f locally around the current iterate and then construct the next iterate via r^{th}-order optimization methods. These approaches typically attain a higher accuracy than the direct and random search methods, and they have the additional advantage that function evaluations can be re-used. In general, however, they require at least $O(n)$ function evaluations in each iteration to construct a well-defined local model [Ber+21]. Examples of commonly used models include polynomial models, interpolation models and regression models [LMW19]. In contrast to model-based methods, random search methods estimate the gradients of f at the iterates directly from finitely many function evaluations and use the resulting estimators as surrogates for the actual gradients in a first-order algorithm. More precisely, random search methods typically approximate f by a smooth function f_δ that is close to f for small δ and construct an unbiased estimator $g_\delta(x)$ for $\nabla f_\delta(x)$ by sampling f at test points in the vicinity of x [FKM04; NS17]. For many popular approximations f_δ there exists $p \geq 1$ such that $\|\nabla f_\delta(x) - \nabla f(x)\| \leq O(\delta^p)$. In analogy to the model-based methods, $g_\delta(x)$ can thus be used as a surrogate for the actual gradient in a first-order algorithm. A striking advantage of these random search methods over model-based methods is that the computation of $g_\delta(x)$ requires only $O(1)$ function evaluations, yet at the expense of weaker approximation guarantees [Liu+20; Ber+21; Sch22]. In principle, the approximation quality of the surrogate gradients (and therefore also the convergence rate of the first-order method at hand) can be improved by reducing the smoothing parameter δ. As $g_\delta(x)$ is often reminiscent of a difference quotient with increment δ, however, its evaluation is plagued by numerical cancellation. This means that if δ drops below a certain threshold, innocent round-off errors in the evaluations of f have a dramatic impact on the evaluations of g_δ. Hence, the actual numerical performance of a random search zeroth-order algorithm may

fall significantly short of its theoretical performance [Shi+22], however, the awareness for this phenomenon seems to be lacking.

Inspired by techniques for numerically differentiating analytic functions, we propose here a new smoothed approximation f_δ as well as a corresponding stochastic gradient estimator g_δ that can be evaluated rapidly and faithfully for arbitrarily small values of δ without suffering from cancellation effects. Integrating the new estimator into the gradient-descent-type algorithm

$$x_{k+1} \leftarrow x_k - \mu_k \cdot g_{\delta_k}(x_k) \tag{1.2}$$

with adaptive stepsize μ_k and smoothing parameter δ_k gives rise to new randomized zeroth-order algorithms. The performance of such algorithms is measured by the decay rate of the regret $R_K = \frac{1}{K}\sum_{k=1}^{K} \mathbb{E}\left[f(x_k) - f(x^\star)\right]$ as K grows. Here, $x^\star$ is a global minimizer, and the expectation $\mathbb{E}[\cdot]$ is taken with respect to the randomness introduced by the algorithm. Note that if f is convex, then Jensen's inequality ensures that the expected suboptimality gap (or expected optimization error) of the *averaged* iterate $\bar{x}_K = \frac{1}{K}\sum_{k=1}^{K} x_k$ satisfies $\mathbb{E}\left[f(\bar{x}_K) - f(x^\star)\right] \leq R_K$. The main goal of this paper is to understand how R_K scales with the total number K of iterations and with critical problem parameters such as the dimension of x or Lipschitz moduli of f. Whenever possible (*e.g.,* when f is strongly convex), we also analyze the expected suboptimality gap $\mathbb{E}[f(x_K) - f(x^\star)]$ of the last iterate x_K. The scaling behavior of R_K with respect to K reflects the algorithm's *convergence rate*. We will show that algorithms of the form (1.2) equipped with the new gradient estimator offer provable convergence rates, are numerically stable, and empirically outperform algorithms that exploit existing smoothed approximations both in terms of accuracy and runtime.

Notation We reserve the symbol $i = \sqrt{-1}$ for the imaginary unit. The real and imaginary parts of a complex number $z = a + ib$ for $a, b \in \mathbb{R}$ are denoted by $\Re(z) = a$ and $\Im(z) = b$. In addition, V_n stands for the volume of the unit ball $\mathbb{B}^n = \{x \in \mathbb{R}^n : \|x\|_2 \leq 1\}$, and S_{n-1} stands for the surface area of the unit sphere $\mathbb{S}^{n-1} = \{x \in \mathbb{R}^n : \|x\|_2 = 1\}$. The family of all r times continuously differentiable real-valued functions on an open set $\mathcal{D} \subseteq \mathbb{R}^n$ is denoted by $C^r(\mathcal{D})$, and the family of all real analytic functions on $\mathcal{D}$ is denoted by $C^\omega(\mathcal{D})$.

1.1 Related work

Given a deterministic zeroth-order oracle, one could address problem (1.1) with a gradient-descent algorithm that approximates the gradient of f with a vector of coordinate-wise finite differences [KW52; KY03; Spa05; Ber+21]. The corresponding finite-difference methods for zeroth-order optimization are reminiscent of inexact gradient methods [d'A08; DGN14]. Maybe surprisingly, there is merit in using *stochastic* gradient estimates even if a *deterministic* zeroth-order oracle is available [NS17]. The randomness not only helps to penetrate previously unexplored parts of the feasible set but also simplifies the convergence analysis. Specifically, if f is convex, then it is often easy to show that $f(x_k)$ converges *in expectation* to the global minimum $f(x^\star)$ [NS17].

Zeroth-order optimization algorithms that mimic gradient descent algorithms can be categorized by the number of oracle calls needed for a single evaluation of the gradient estimator. The most efficient algorithms of this kind make do with one single oracle call. Arguably the first treatise on zeroth-order optimization with a random single-point gradient estimator appeared in [NY83, § 9.3], where the objective function $f(x)$ is approximated by the smoothed function $f_\delta(x) = V_n^{-1} \int_{\mathbb{B}^n} f(x + \delta y)\,\mathrm{d}y$, and the degree of smoothing is controlled by the parameter $\delta > 0$. By leveraging the dominated convergence theorem and the classical divergence theorem, one can then derive the following integral representation

for the gradient of $f_\delta(x)$,

$$\nabla f_\delta(x) = \tfrac{n}{\delta} \int_{\mathbb{S}^{n-1}} f(x+\delta y) y \, \sigma(\mathrm{d}y),$$

where σ represents the uniform distribution on the unit sphere $\mathbb{S}^{n-1}$ (see also the proof of Proposition 3.3). Hence, the gradient of the smoothed function f_δ admits the unbiased stochastic estimator

$$g_\delta(x) = \tfrac{n}{\delta} f(x+\delta y) y \quad \text{with} \quad y \sim \sigma, \tag{1.3}$$

which can be accessed with merely a single function evaluation. Stochastic gradient estimators of this kind have been used as surrogate gradients in gradient descent algorithms, for example, in the context of bandit problems [FKM04]. However, as already pointed out in [NY83], the variance of the gradient estimator (1.3) is of the order $O(n^2/\delta^2)$ for small δ even if the function f is constant. This is inconvenient because a smaller δ reduces the bias of f_δ *vis-à-vis* f. To improve this bias-variance trade-off, it has been proposed to subtract from $g_\delta(x)$ the control variate $n\delta^{-1} f(x) y$, which has a vanishing mean but is strongly correlated with $g_\delta(x)$ and therefore leads to a variance reduction [ADX10; NS17]. The resulting unbiased stochastic gradient is representable as

$$g'_\delta(x) = \tfrac{n}{\delta} \left(f(x+\delta y) - f(x)\right) y \quad \text{with} \quad y \sim \sigma, \tag{1.4}$$

which is reminiscent of a directional derivative and can be accessed via two function evaluations. Now, under mild conditions on f, the variance of $g'_\delta(x)$ remains bounded as δ tends to 0. If we aim to solve problem (1.1) to an arbitrary precision, however, the smoothing parameter δ needs to be made arbitrarily small, in which case $f(x+\delta y)$ and $f(x)$ become numerically indistinguishable. Subtractive cancellation therefore makes it impossible to evaluate estimators of the form (1.4) to an arbitrarily high precision. This phenomenon is exacerbated when the function evaluations are noisy, which commonly happens in simulation-based optimization [Lia+16]. Generalized stochastic gradient estimators requiring multiple function evaluations are discussed in [HL14], and in [Duc+15; LLZ21] various optimality properties of zeroth-order schemes with multi-point gradient estimators are discussed.

Stochastic gradient estimators akin to (1.4) with u following a Gaussian instead of a uniform distribution are studied in [NS17]. The corresponding stochastic gradient descent algorithms may converge as fast as $O(n/K)$ if f is convex and has a Lipschitz continuous gradient, but they are typically $O(n)$ times slower than their deterministic counterparts. Convergence can be accelerated by leveraging central finite-difference schemes or by adding random perturbations to the gradient estimators [Duc+15; Gas+17; Sha17]. Local convergence results for nonconvex optimization problems are investigated in [GL13], and second-order algorithms similar to (1.2), which use a Stein identity to estimate the Hessian matrix, are envisioned in [BG22]. Lower bounds on the convergence rate of algorithm (1.2) are established in [Aga+09; JNR12; Sha13].

Another stream of related research investigates zeroth-order optimization methods that have only access to a *stochastic* zeroth-order oracle, which returns function evaluations contaminated by noise. The performance of these methods critically depends on the smoothness properties of f. Indeed, the higher its degree of smoothness, the more terms in the Taylor series of f can be effectively averaged out [PT90]. Improved convergence results for zeroth-order optimization methods under convexity assumptions are derived in [BP16; APT20; NG22]. When function evaluations are noisy, the smoothing parameter δ controls a bias-variance tradeoff. Indeed, reducing δ reduces the bias introduced by smoothing f, while increasing δ reduces the variance of the gradient estimator induced by the noisy oracle, which scales as $1/\delta$ for small δ. The variance can be further reduced by mini-batching [Ji+19].

The impact of exact line search methods and adaptive stepsize selection schemes is discussed in [SMG13; BCS21]. Better stepsize rules are available if f displays a latent low-dimensional structure [Gol+20].

Generalized zeroth-order methods for optimizing functions defined on Riemannian manifolds are proposed in [LBM22], and algorithms that have only access to a *comparison oracle*, which is less informative than a zeroth-order oracle, are investigated in [Cai+22].

For comprehensive surveys of zeroth-order optimization and derivative-free optimization we refer to [LMW19; Liu+20]. Abstract zeroth-order methods for convex optimization are studied in [Hu+16]. The minimax regret bounds derived in this work reveal the importance of having control over the randomness of the zeroth-order oracle. Accordingly, most existing methods rely on the assumption that the noise distribution is light-tailed. In contrast, if the zeroth-order oracle is affected by adversarial noise, then optimization is easily obstructed [SV15, Thm 3.1].

1.2 Contributions Most existing zeroth-order schemes approximate the gradient of f in a way that makes them susceptible to numerical instability. For example, if $f \in C^1(\mathbb{R})$ is Lipschitz continuous with Lipschitz constant L, then, in theory, the finite-difference approximation $(f(x+\delta) - f(x))/\delta$ converges to $\partial_x f(x)$ as $\delta > 0$ tends to zero. In practice, however, f can only be evaluated to within machine precision, which means that $f(x+\delta)$ and $f(x)$ become indistinguishable for sufficiently small δ. More precisely, as f is Lipschitz continuous, we have $|f(x+\delta) - f(x)| \leq L \cdot |\delta|$, and thus cancellation errors are prone to occur when $L \cdot |\delta|$ approaches machine precision [Ove01, § 11]. Other gradient estimators that are based on multiple function evaluations or that involve interpolation schemes suffer from similar cancellation errors. Nevertheless, the convergence guarantees of the corresponding zeroth-order methods require that the smoothing parameter δ must be driven to zero. For example, [APT20, Thm. 3.1] establishes regret bounds under the assumption that the smoothing parameter of a multi-point estimator scales as $\delta_k = O(1/\sqrt{k})$.

The randomized gradient estimator (1.3) avoids cancellation errors because it requires only one single function evaluation—an attractive feature that has, to the best of our knowledge, gone largely unnoticed to date. However, as pointed out earlier, the variance of this estimator diverges as δ decays, which leads to suboptimal convergence rates. In this paper we propose a numerically stable gradient estimator that enables competitive convergence rates and is immune to cancellation errors. More precisely, we will use complex arithmetic to construct a one-point estimator akin to (1.3) that offers similar approximation and convergence guarantees as state-of-the-art two-point estimators. Maybe surprisingly, we will see that computing this new estimator is not significantly more expensive than evaluating (1.3). Our results critically rely on the assumption that the objective function f is real analytic on its domain $\mathcal{D}$. Recall that f is real analytic if it locally coincides with its multivariate Taylor series. We emphasize that real analyticity does not imply β^{th}-order smoothness for some $\beta \in \mathbb{Z}_{>0}$ in the sense of [BP16, § 1.1], which means that f is almost surely $\beta - 1$ times differentiable and that the $(\beta-1)^{\text{th}}$-order term of its Taylor series is globally Lipschitz continuous. We will recall that f can be extended to a complex analytic function $f : \Omega \to \mathbb{C}$ defined on some open set $\Omega \subseteq \mathbb{C}^n$ that covers $\mathcal{D} \subseteq \mathbb{R}^n$. By slight abuse of notation, this extension is also denoted by f. Given an oracle that evaluates f at any query point in Ω, we will devise new zeroth-order methods that combine the superior convergence rates and low variances of multi-point schemes reported in [Duc+15; LLZ21] with the numerical robustness of single-point approaches.

We now use $R = \|x_1 - x^\star\|_2$ to denote the distance from the initial iterate x_1 to a minimizer $x^\star$ and $F = f(x_1) - f(x^\star)$ to denote the suboptimality of x_1. Assuming that

the objective function f is real analytic and has an L-Lipschitz continuous gradient, we will devise zeroth-order methods that offer the following convergence guarantees. If (1.1) represents a (constrained or unconstrained) convex optimization problem with $x^\star \in \operatorname{int}(\mathcal{X})$, then our algorithm's regret decays as $O(nLR^2/K)$ with the iteration counter K. If, in addition, f is τ-strongly convex for some $\tau > 0$, then the expected suboptimality decays at the linear rate $O((1 - \tau/(4nL))^K LR^2)$. If (1.1) represents a non-convex optimization problem, finally, we establish local convergence to a stationary point and prove that $\min_{k\in[K]} \mathbb{E}[\|\nabla f(x_k)\|_2^2] \leq O(nLF/K)$. All of these convergence rates are qualitatively equivalent to the respective rates reported in [NS17, Thm. 8], and they are sharper than the rates provided in [APT20, § 3] in the noise-free limit. The latter rely on higher-order smoothness properties of f but do not require f to be analytic. The key difference to all existing methods is that we can drive the smoothing parameter to 0, *e.g.*, as $\delta_k = \delta/k$, without risking numerical instability.

As highlighted in the recent survey article [LMW19], an important open question in zeroth-order optimization is whether single-point estimators enable equally fast convergence rates as multi-point estimators. The desire to reap the benefits of multi-point estimators at the computational cost of using single-point estimators has inspired multi-point estimators with memory, which only require a single new function evaluation per call [Zha+22]. However, this endeavor has not yet led to algorithms that improve upon the theoretical and empirical performance of the state-of-the-art methods in [NS17]. Filtering techniques inspired by ideas from extremum seeking control can be leveraged to improve the convergence rates obtained in [Zha+22] to $O(n/K^{2/3})$ [CTL22]. However, this rate is still inferior to the ones reported in [NS17]. To our best knowledge, we propose here the first single-point zeroth-order algorithm that enjoys the same convergence rates as the multi-point methods in [NS17] but often outperforms these methods in experiments. The price we pay for these benefits is the assumption that there exists a zeroth-order oracle accepting complex queries. This assumption is restrictive as it rules out oracles that depend on performing a physical experiment or timing a computational run etc. Nevertheless, as we will see in Section 7, the approach can excel in the context of simulation-based optimization.

Numerical experiments built around standard test problems as well as a model predictive control (MPC) problem corroborate our theoretical results and demonstrate the practical efficiency of the proposed algorithms. Although cancellation effects are caused by rounding to machine precision, which is nowadays of the order 10^{-16}, our single-point gradient estimator improves both the accuracy as well as the speed of zeroth-order algorithms already when ε-optimal solutions with $\varepsilon \gg 10^{-16}$ are sought.

Structure Section 2 reviews basic tools from multivariate complex analysis and introduces the complex-step method from numerical differentiation. Section 3 then combines smoothing techniques with complex arithmetic to construct a new single-point gradient estimator, and Sections 4-6 analyze the favorable convergence rates of zeroth-order optimization methods equipped with the new gradient estimator in the context of convex, strongly convex and non-convex optimization, respectively. Section 7 reports on numerical experiments, and Section 8 concludes.

2 Preliminaries

Before presenting our main results, we review some tools that may not usually belong to the standard repertoire of researchers in optimization. Specifically, Section 2.1 reviews the relevant basics of multivariate complex analysis, Section 2.2 introduces the complex-step

approach, which uses complex arithmetic for computing highly precise and numerically stable approximations of derivatives based on a single function evaluation, and Section 2.4 provides a survey of inequalities that will be needed for the analysis of the algorithms proposed in this paper.

2.1 Multivariate complex analysis For any multi-index $\alpha \in \mathbb{Z}^n_{\geq 0}$ and vector $x \in \mathbb{R}^n$, we use x^α as a shorthand for the monomial $x_1^{\alpha_1} \cdots x_n^{\alpha_n}$, and we denote the degree of x^α by $|\alpha| = \sum_{i=1}^n \alpha_i$. The factorial of α is defined as $\alpha! = \prod_{i=1}^n \alpha_i!$, and ∂_x^α stands for the higher-order partial derivative $\partial_{x_1}^{\alpha_1} \cdots \partial_{x_n}^{\alpha_n}$. Multi-index notation facilitates a formal definition of real analytic functions.

Definition 2.1 (Real analytic function)**.** *The function $f : \mathcal{D} \to \mathbb{R}$ is real analytic on $\mathcal{D} \subseteq \mathbb{R}^n$, denoted $f \in C^\omega(\mathcal{D})$, if for every $x' \in \mathcal{D}$ there exist $f_\alpha \in \mathbb{R}$, $\alpha \in \mathbb{Z}^n_{\geq 0}$, and an open set $U \subseteq \mathcal{D}$ containing x' such that*

$$f(x) = \sum_{\alpha \in \mathbb{Z}^n_{\geq 0}} f_\alpha \cdot (x - x')^\alpha \quad \forall x \in U. \tag{2.1}$$

Whenever we write that a series has a finite value, we mean that it converges absolutely, that is, it converges when the summands of the series are replaced by their absolute values. In this case any ordering of the summands results in the same value.

One can show that any real analytic function is infinitely differentiable and that the coefficients of its power series are given by $f_\alpha = \frac{1}{\alpha!} \partial_x^\alpha f(x')$ for every $\alpha \in \mathbb{Z}^n_{\geq 0}$. This implies that the power series is unique and coincides with the multivariate Taylor series of f around x' [KP02, § 2.2]. We will now recall that every real analytic function admits a complex analytic extension.

Definition 2.2 (Complex analytic function)**.** *The function $f : \Omega \to \mathbb{C}$ is complex analytic on $\Omega \subseteq \mathbb{C}^n$, denoted $f \in H(\Omega)$, if for every $z' \in \Omega$ there exist $f_\alpha \in \mathbb{C}$, $\alpha \in \mathbb{Z}^n_{\geq 0}$, and an open set $U \subseteq \Omega$ containing z' such that*

$$f(z) = \sum_{\alpha \in \mathbb{Z}^n_{\geq 0}} f_\alpha \cdot (z - z')^\alpha \quad \forall z \in U. \tag{2.2}$$

Complex analytic functions are intimately related to holomorphic functions.

Definition 2.3 (Holomorphic function)**.** *The function $f : \Omega \to \mathbb{C}$ is holomorphic on an open set $\Omega \subseteq \mathbb{C}^n$ if the complex partial derivatives $\partial_{z_j} f$, $j = 1, \ldots, n$, exist and are finite at every $z \in \Omega$.*

The requirement that Ω be open is essential, and f may fail to be holomorphic on a neighborhood of a point z even if it is complex differentiable at z. For example, the Cauchy-Riemann equations reviewed below imply that $f(z) = |z|^3$ is complex differentiable at $z = 0$ but fails to be complex differentiable on any neighborhood of 0. Holomorphic functions are in fact infinitely often differentiable [Leb20, Prop. 1.1.3]. Moreover, a function is holomorphic if and only if it is complex analytic [Leb20, Thm. 1.2.1].

It is common to identify any complex vector $z \in \mathbb{C}^n$ with two real vectors $x, y \in \mathbb{R}^n$ through $z = x + iy$. Similarly, we may identify any complex function $f : \mathbb{C}^n \to \mathbb{C}$ with two real functions $u : \mathbb{R}^n \to \mathbb{R}$ and $v : \mathbb{R}^n \to \mathbb{R}$ through the relation $f(x+iy) = u(x, y)+iv(x, y)$. Clearly, u and v inherit the differentiability properties of f and vice versa. In particular, one can show that if f is holomorphic, then the partial derivatives of u and v exist and satisfy the multivariate Cauchy-Riemann equations.

Theorem 2.4 (Multivariate Cauchy-Riemann equations)**.** *If $f(x+iy) = u(x,y) + iv(x,y)$ is a holomorphic function on an open set $\Omega \subseteq \mathbb{C}^n$, then the multivariate Cauchy-Riemann equations*

$$\partial_{x_j} u(x,y) = \partial_{y_j} v(x,y) \quad \textit{and} \quad -\partial_{x_j} v(x,y) = \partial_{y_j} u(x,y) \quad \forall j = 1,\dots,n \tag{2.3}$$

hold for all $x, y \in \mathbb{R}^n$ with $x + iy \in \Omega$.

Theorem 2.4 is a standard result in complex analysis; see, *e.g.*, [Rud87, Thm. 11.2] or [Kra00]. Nevertheless, we provide here a short proof to keep this paper self-contained.

Proof of Theorem 2.4. We use e_j to denote the j^{th} standard basis vector in $\mathbb{R}^n$. By the definition of the complex partial derivative, for any $z \in \Omega$ we have

$$\partial_{z_j} f(z) = \lim_{\delta \in \mathbb{C},\, \delta \to 0} \tfrac{1}{\delta}(f(z + \delta e_j) - f(z)),$$

where the limit exists and is independent of how $\delta \in \mathbb{C}$ converges to 0 because f is holomorphic on Ω. In particular, δ may converge to 0 along the real or the imaginary axis without affecting the result. Using our conventions that $z = x+iy \in \Omega$ and $f(x+iy) = u(x,y)+iv(x,y)$, we thus have

$$\begin{aligned}
\partial_{x_j}(u(x,y) + iv(x,y)) &= \lim_{\delta \in \mathbb{R},\, \delta \to 0} \frac{f((x+\delta e_j)+iy) - f(x+iy)}{\delta} \\
&= \lim_{\delta \in \mathbb{R},\, \delta \to 0} \frac{f(x+i(y+\delta e_j)) - f(x+iy)}{i\delta} \\
&= \tfrac{1}{i}\partial_{y_j}(u(x,y) + iv(x,y))
\end{aligned}$$

for all $x, y \in \mathbb{R}^n$, where the second equality holds because both limits are equal to $\partial_{z_j} f(z)$. Matching the real and imaginary parts of the above equations yields (2.3). □

Under additional assumptions one can further show that the Cauchy-Riemann equations imply that f is holomorphic [GM78]. However, this reverse implication will not be needed in this paper. The following lemma based on [Kra00, § 2.3] establishes that any real analytic function defined on an open set $\mathcal{D} \subseteq \mathbb{R}^n$ admits a complex analytic extension defined on an open set $\Omega \subseteq \mathbb{C}^n$ that covers $\mathcal{D}$.

Lemma 2.5 (Complex analytic extensions)**.** *If $f \in C^\omega(\mathcal{D})$, then there exists an open set $\Omega \subseteq \mathbb{C}^n$ and a complex analytic function $g \in H(\Omega)$ such that $\mathcal{D} \subseteq \Omega$ and $f(x) = g(x)$ for every $x \in \mathcal{D}$, with $\mathcal{D}$ understood as embedded in $\mathbb{C}^n$.*

Proof. Select any $x' \in \mathcal{D}$. As $f \in C^\omega(\mathcal{D})$, there exists a neighborhood $U \subseteq \mathcal{D}$ of x' such that f admits a power series representation of the form (2.2) on U. Also, as U is open, there exists $x \in U$ with $r_j = |x_j - x'_j| > 0$ for every $j = 1,\dots,n$. By Abel's lemma [Kra00, Prop. 2.3.4], the power series (2.2) extended to $\mathbb{C}^n$ is thus guaranteed to converge on the open polydisc $\Delta(x') = \{z \in \mathbb{C}^n : |z_j - x'_j| < r_j \,\forall j = 1,\dots,n\}$. This reasoning implies that f extends locally around x' to a complex analytic function, which we henceforth denote as $g_{x'}$. It remains to be shown that the local extensions corresponding to different reference points $x' \in \mathcal{D}$ are consistent. To this end, select any $x', x'' \in \mathcal{D}$ such that the polydiscs $\Delta(x')$ and $\Delta(x'')$ overlap. We need to prove that $g_{x'}$ and $g_{x''}$ coincide on the open convex set $\Delta = \Delta(x') \cap \Delta(x'')$, which has a non-empty intersection with $\mathbb{R}^n$. For ease of exposition, we will equivalently prove that the holomorphic function $h = g_{x'} - g_{x''}$

vanishes on Δ. We first notice that h vanishes on $\Delta \cap \mathbb{R}^n$ because $g_{x'}$ and $g_{x''}$ are constructed to coincide with f on $\Delta \cap \mathbb{R}^n$. This implies that $\partial_{z_j} h = \partial_{x_j} h = 0$ on $\Delta \cap \mathbb{R}^n$, where the first equality follows from standard arguments familiar from the proof of Theorem 2.4. As any partial derivative of a holomorphic function is also holomorphic, one can use induction to show that all higher-order partial derivatives of h must vanish on $\Delta \cap \mathbb{R}^n$. Hence, the Taylor series of h around any reference point in $\Delta \cap \mathbb{R}^n$ vanishes, too. We have thus shown that $h = 0$ on an open subset of Δ. By standard results in complex analysis, this implies that h vanishes throughout Δ; see, *e.g.*, [Leb20, Thm. 1.2.2]. In summary, this reasoning confirms that all local complex analytic extensions $g_{x'}$, $x' \in \mathcal{D}$, of f are consistent and thus coincide with a complex analytic function g defined on the open set $\Omega = \cup_{x' \in \mathcal{D}} \Delta(x')$. This observation completes the proof. □

Lemma 2.5 implies that the complex extension of a real analytic function is unique. For example, $f(x) = \log(x)$ is real analytic on the positive real line. Representing $z = re^{i\theta} \in \mathbb{C}$ in polar form with $r \geq 0$ and $\theta \in (-\pi, \pi]$, the complex logarithm has countably many branches, that is, $\log(z)$ can be defined as $g_k(z) = \log(r) + i(\theta + 2\pi k)$ for any $k \in \mathbb{Z}$. However, only the branch g_0 corresponding to $k = 0$ matches f on the positive reals. We will henceforth use the same symbol f to denote both the given real analytic function as well as its unique complex analytic extension g. We now explicitly derive the complex analytic extensions of a few simple univariate functions.

Example 2.6 (Complex analytic extensions)**.** *The unique complex analytic extension of $f(x) = e^x$ is the entire function $g(z) = g(x + iy) = e^x(\cos(y) + i\sin(y))$. Similarly, the unique complex analytic extension of the even polynomial $f(x) = x^{2p}$ with $p \in \mathbb{Z}_{\geq 0}$ is the entire function*

$$g(z) = g(x + iy) = \textstyle\sum_{k=0}^{p} (-1)^k \binom{2p}{2k} y^{2k} x^{2(p-k)} + i \sum_{k=0}^{p-1} (-1)^k \binom{2p}{2k+1} y^{2k+1} x^{2(p-k)-1}.$$

Finally, the unique solution $f(x)$ to the Lyapunov equation $f(x) = x^2 f(x) + 1$ parametrized by $x \in \mathbb{R}$ is real analytic on $\mathbb{R} \setminus \{1\}$. It admits the extension

$$g(z) = g(x + iy) = \frac{1 - x^2 + y^2 - 2ixy}{(1 - x^2 + y^2)^2 + 4x^2 y^2},$$

which is analytic throughout $\mathbb{C} \setminus \{(1, 0)\}$.

The next example shows that the domain Ω of the complex analytic extension is not always representable as $\mathbb{R}^n + i \cdot (-\bar{\delta}, \bar{\delta})^n$ for some $\bar{\delta} > 0$ even if $\mathcal{D} = \mathbb{R}^n$.

Example 2.7 (Non-trivial extension)**.** *Consider the function $f(x) = \sum_{k=1}^{\infty} 2^{-k}(1 + k^2(x - k)^2)^{-1} \in C^\omega(\mathbb{R})$, which admits a unique complex analytic extension with domain $\Omega = \mathbb{C} \setminus \{k + ik^{-1} : k \in \mathbb{Z}_{>0}\}$. In addition, f can be extended to a meromorphic function on $\mathbb{C}$ with countably many poles $k + ik^{-1}$, $k \in \mathbb{Z}_{>0}$. As these poles approach $\mathbb{R}$ arbitrarily closely, however, Ω cannot contain any strip of the form $\mathbb{R} \times i \cdot (-\bar{\delta}, \bar{\delta})$.*

To avoid technical discussions of limited practical impact, we will from now on restrict attention to functions $f \in C^\omega(\mathcal{D})$ that admit a complex analytic extension to $\Omega = \mathcal{D} \times i \cdot (-\bar{\delta}, \bar{\delta})^n$ for some $\bar{\delta} > 0$. One can show that such an extension always exists if $f \in C^\omega(\mathbb{R}^n)$ and $\mathcal{D}$ is bounded or if f is *entire*, that is, if f has a globally convergent power series representation. The latter condition is restrictive, however, because it rules out simple functions such as $f(x) = 1/(1+x^2)$. Provided there is no risk of confusion, we will sometimes call a real analytic function $f \in C^\omega$ and its complex analytic extension simply an *analytic* function.

2.2 Complex-step approximation The *finite-difference* method [Vui+23, Ch. 3] is arguably the most straightforward approach to numerical differentiation. It simply approximates the derivative of any sufficiently smooth function $f \in C^2(\mathbb{R})$ by a difference quotient. For example, the forward-difference method uses the approximation

$$\partial_x f(x) = \tfrac{1}{\delta}(f(x+\delta) - f(x)) + O(\delta). \tag{2.4}$$

The continuity of the second derivative of f allows for a precise formula for the $O(\delta)$ remainder term. However, as explained earlier, the finite difference method suffers from cancellation errors when δ becomes small. The *complex-step* approximation proposed in [LM67; ST98] and further refined in [MSA03; ASM15; Abr+18] leverages ideas from complex analysis to approximate the derivative of any real analytic function $f \in C^\omega(\mathbb{R})$ on the basis of one single function evaluation only, thereby offering an elegant remedy for numerical cancellation. Denoting by u and v as usual the real and imaginary parts of the unique complex analytic extension of f, which exists thanks to Lemma 2.5, we observe that $\partial_x f(x)$ equals

$$\partial_x u(x,0) = \partial_y v(x,0) = \lim_{\delta \downarrow 0} \tfrac{1}{\delta}(v(x,\delta) - v(x,0)) = \lim_{\delta \downarrow 0} \tfrac{1}{\delta} v(x,\delta) = \lim_{\delta \downarrow 0} \tfrac{1}{\delta} \Im\big(f(x+i\delta)\big),$$

where the first and the fourth equalities hold because $f(x)$ must be a real number, which implies that $v(x,0) = 0$, while the second equality follows from the Cauchy-Riemann equations. The derivative $\partial_x f(x)$ can thus be approximated by the fraction $\Im(f(x+i\delta))/\delta$, which requires merely a single function evaluation. To estimate the approximation error, we consider the Taylor expansion

$$f(x+i\delta) = f(x) + \partial_x f(x) i\delta - \tfrac{1}{2}\partial_x^2 f(x)\delta^2 - \tfrac{1}{6}\partial_x^3 f(x) i\delta^3 + O(\delta^4) \tag{2.5}$$

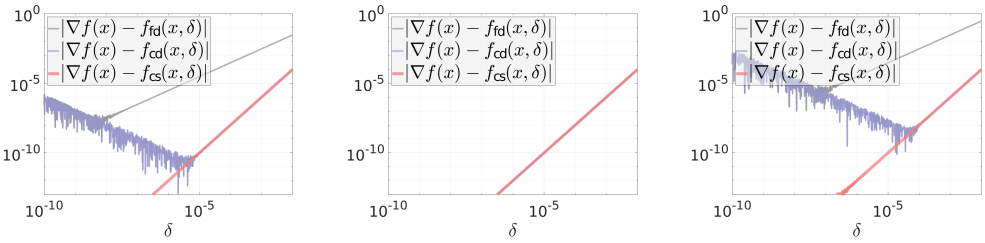
of the unique complex analytic extension of f, which exists thanks to Lemma 2.5. Separating the real and imaginary parts of (2.5) then yields

$$f(x) = \Re(f(x+i\delta)) + O(\delta^2) \quad \text{and} \quad \partial_x f(x) = \tfrac{1}{\delta}\Im\big(f(x+i\delta)\big) + O(\delta^2).$$

This reasoning shows that a single *complex* function evaluation $f(x+i\delta)$ is sufficient to approximate both $f(x)$ as well as $\partial_x f(x)$ without the risk of running into numerical instability caused by cancellation effects. In addition, the respective approximation errors scale quadratically with δ and are thus one order of magnitude smaller than the error incurred by (2.4). Note also that the complex-step approximation recovers the derivatives of quadratic functions *exactly* irrespective of the choice of δ. For example, if $f(x) = x^2$, then $\Im(f(x+i\delta))/\delta = 2x = \partial_x f(x)$. This insight suggests that the approximation is numerically robust for locally quadratic functions.

The error of the complex-step approximation can be further reduced to $O(\delta^4)$ by enriching it with a finite-difference method [ASM15; HS23]. However, the resulting scheme requires multiple function evaluations and is thus again prone to cancellation errors. Unless time is expensive, the standard complex-step approximation therefore remains preferable. The complex-step approximation can also be generalized to handle matrix functions [AMH10] or to approximate higher-order derivatives [LRD12]. Its ramifications for automatic differentiation (AD) are discussed in [MSA03]. We return to AD below.

Being immune to cancellation effects, the complex-step approach offers approximations of almost arbitrary precision. For example, software by the UK's National Physical Laboratory is reported to use smoothing parameters as small as $\delta = 10^{-100}$ [CH04, p. 44]. The complex-step approach also emerges in various other domains. For example, it is successfully used in airfoil design [GWX17]. However, its potential for applications in optimization



(a) $x = -1$ **(b)** $x = 0$ **(c)** $x = 10$

Figure 2.1: Comparison of the gradient estimators of Example 2.8 at different test points.

has not yet been fully exploited. Coordinate-wise complex-step approximations with noisy function evaluations show promising performance in line search experiments [NS18] but come without a rigorous convergence analysis. In addition, the complex-step approach is used to approximate the gradients and Hessians in deterministic Newton algorithms for blackbox optimization models [HS23]. The potential of leveraging complex arithmetic in mathematical optimization is also mentioned in [SW18; BBN19]. In this paper we use the complex-step method to construct an estimator akin to (1.3) and provide a full regret analysis. Our approach is most closely related to the recent works [WS21; WZS21], which integrate the complex-step and simultaneous perturbation stochastic approximations [Spa+92] into a gradient-descent algorithm and offer a rigorous asymptotic convergence theory. In contrast, we will derive convergence *rates* for a variety of zeroth-order optimization problems.

In optimization, the ability to certify that the gradient of an objective function is sufficiently small (*i.e.*, smaller than a prescribed tolerance) is crucial to detect local optima. The following example shows that, with the exception of the complex-step approach, standard numerical schemes to approximate gradients fail to offer such certificates—at least when a high precision is required.

Example 2.8 (Numerical stability of gradient estimators)**.** *To showcase the power of the complex-step method and to expose the numerical difficulties encountered by finite-difference methods, we approximate the derivative of* $f(x) = x^3$ *at* $x \in \{-1, 0, 10\}$ *via a forward-difference (*fd*), central-difference (*cd*) and complex-step (*cs*) method, that is, for small values of* δ *we compare* $f_{\mathsf{fd}}(x,\delta) = \frac{1}{\delta}(f(x+\delta) - f(x))$, $f_{\mathsf{cd}}(x,\delta) = \frac{1}{2\delta}(f(x+\delta) - f(x-\delta))$ *and* $f_{\mathsf{cs}}(x,\delta) = \frac{1}{\delta}\Im(f(x+i\delta))$. *Figure 2.1 visualizes the absolute approximation errors as a function of* δ. *We observe that* f_{cd} *and* f_{cs} *offer the same approximation quality and incur an error of* $O(\delta^2)$ *for all sufficiently large values of* δ. *However, only the complex-step approximation reaches machine precision (*$\approx 10^{-16}$*), whereas both finite-difference methods deteriorate below* $\delta \approx 10^{-6}$ *due to subtractive cancellation errors. Note that for* $x = 0$ *all errors are equal to* δ^2 *because* $f(0) = 0$. *As most existing zeroth-order optimization methods use finite-difference-based gradient estimators, we conclude that there is room for numerical improvements by leveraging complex arithmetic.*

2.3 Automatic differentiation

The complex-step approach is closely related to *automatic differentiation* (AD) [GW08; Ell09]. AD decomposes the evaluation of f into a partially ordered set of elementary operations and evaluates its derivative recursively using the rules of differentiation such as the chain and product rules etc. Much like the complex-step approach, AD differs both from symbolic differentiation (which requires an algebraic representation of

f and is computationally inefficient) as well as numerical differentiation (which suffers from round-off errors and cancellation). Moreover, while the complex step approach evaluates f at *complex* numbers of the form $a + ib$ with $a, b \in \mathbb{R}$ and an abstract imaginary unit i satisfying $i^2 = -1$, forward-mode AD evaluates f at *dual* numbers of the form $a + b\varepsilon$ with $a, b \in \mathbb{R}$ and $\varepsilon \neq 0$ an abstract number satisfying $\varepsilon^2 = 0$. The arithmetics of dual numbers imply that $f(x+\varepsilon) = f(x)+\varepsilon\partial_x f(x)$ whenever f is real analytic, and hence one can compute both $f(x)$ as well as $\partial_x f(x)$ in one forward pass. Intuitively, ε should thus be interpreted as a nilpotent infinitesimal unit. While the set of complex numbers forms a *field*, the set of dual numbers only forms a *ring* (in fact, it forms the quotient ring $\mathbb{R}[\varepsilon]/\varepsilon^2$, which fails to be a field because multiplicative inverses and hence terms such as $\varepsilon^2/\varepsilon$ and $\sqrt{\varepsilon^2}$ are not defined). The assumption that $\varepsilon \neq 0$ and $\varepsilon^2 = 0$ require us to give up the law of the excluded middle [Bel08] and thus also the axiom of choice [Bau17].

The complex-step approach is computationally cheap, can *approximate* the derivative of an analytic function f at extremely high accuracy levels (see Figure 2.1) and even remains applicable when function evaluations are noisy [MW14]. AD is computationally more expensive than the complex-step approach [MSA01] but usually finds the *exact* derivative of f. Whether or not AD will succeed, however, depends on the representation of f; see, *e.g.*, [Hüc+23] for a discussion of possible pitfalls. That is, AD must be able to evaluate f at all dual numbers of the form $x+\varepsilon$. The following example inspired by [Ber92] illustrates why this is restrictive. Consider the function $f \in C^\omega(\mathbb{R})$ defined through $f(x) = \operatorname{sinc}(x) = \sin(x)/x$ for all $x \in \mathbb{R} \setminus \{0\}$ with $f(0) = 1$. This function is entire and has a unique global maximum of 1 at $x = 0$. Nevertheless, AD breaks down at $x = 0$ because $1/\varepsilon$ is not defined. For instance, the deep learning toolbox in MATLAB as well as the state-of-the-art AD tools in Julia [Bez+17] (*e.g.*, `ForwardDiff.jl` [RLP16], `Zygote.jl` [Inn18] and `Enzyme.jl` [MC20]) or in Python (*e.g.*, `JAX` [Bra+18]) evaluate $\partial_x f(0)$ to `NaN`, whereas the complex-step method provides a close approximation of the correct value $\partial_x f(0) = 0$. We remark that the derivative of $f(x) = \operatorname{sinc}(x)$ is hard-coded in Julia.[1] Hence, for AD to succeed the representation of f is critical, that is, f must be defined as $f(x) = \operatorname{sinc}(x)$ instead of $f(x) = \sin(x)/x$. A simpler, yet contrived, example is $f(x) = x/x$, for which AD evaluates $\partial_x f(0)$ to `NaN`, too. Note also that AD fails to compute the derivative of $f(x) = \exp((\sqrt{x^2})^2)$ at 0 even though there is no division by 0. All of these problems emerge because the dual numbers form only a ring instead of a field. As pointed out in [Ber92], these theoretical deficiencies of AD could be remedied by working with the *Levi-Civita field*, whose members generalize the dual numbers and are representable as $\sum_{q\in\mathbb{Q}} a_q \varepsilon^q$ with $a_q \in \mathbb{R}$ for all $q \in \mathbb{Q}$. Unfortunately, the members of the Levi-Civita field do not admit a finite representation in general and are therefore difficult to handle computationally.

2.4 Lipschitz inequalities In order to be able to design reasonable zeroth-order optimization algorithms, we need to impose some regularity on the objective function f. This is usually done by requiring f to display certain Lipschitz continuity properties. Following [Nes03], for any integers $p, k \geq 0$ with $p \leq k$, we thus use $C_L^{k,p}(\mathcal{D})$ to denote the family of all k times continuously differentiable functions on $\mathcal{D}$ whose p^{th} derivative is Lipschitz continuous with Lipschitz constant $L \geq 0$. Similarly, we use $C_L^{\omega,p}(\mathcal{D})$ to denote the family of all analytic functions in $C_L^{p,p}(\mathcal{D})$.

For example, if $f \in C_{L_1}^{1,1}(\mathcal{D})$, then f has a Lipschitz continuous gradient, that is,

$$\|\nabla f(x) - \nabla f(y)\|_2 \leq L_1 \|x - y\|_2 \quad \forall x, y \in \mathcal{D}. \tag{2.6}$$

[1] https://github.com/JuliaDiff/DiffRules.jl/blob/9030629bbea6b25851789af5f236f35c9009b1f6/src/rules.jl

By [NS17, Eq. (6)], this condition is equivalent to the inequality

$$|f(y) - f(x) - \langle \nabla f(x), y - x\rangle| \leq \tfrac{1}{2} L_1 \|x - y\|_2^2 \quad \forall x, y \in \mathcal{D}. \tag{2.7}$$

If $f \in C^{1,1}_{L_1}(\mathcal{D})$ is also convex then, the Lipschitz condition (2.6) is also equivalent to

$$f(y) \geq f(x) + \langle \nabla f(x), y - x\rangle + \tfrac{1}{2L_1} \|\nabla f(y) - \nabla f(x)\|_2^2 \quad \forall x, y \in \mathcal{D}, \tag{2.8}$$

see, *e.g.,* [Nes03]. In particular, if x is a local minimizer of f with $\nabla f(x) = 0$, then the estimate (2.8) simplifies to $2L_1 \left(f(y) - f(x)\right) \geq \|\nabla f(y)\|_2^2$ for all $y \in \mathcal{D}$.

If $f \in C^{2,2}_{L_2}(\mathcal{D})$, then f has a Lipschitz continuous Hessian, *i.e.*,

$$\|\nabla^2 f(x) - \nabla^2 f(y)\|_2 \leq L_2 \|x - y\|_2 \quad \forall x, y \in \mathcal{D}. \tag{2.9}$$

By [Nes03, Lem. 1.2.4], this condition is equivalent to the inequality

$$|f(y) - f(x) - \langle \nabla f(x), y - x\rangle - \tfrac{1}{2}\langle \nabla^2 f(x)(y - x), y - x\rangle| \leq \tfrac{1}{6} L_2 \|x - y\|_2^3 \quad \forall x, y \in \mathcal{D}. \tag{2.10}$$

More generally, any $f \in C^{p,p}_{L_p}(\mathcal{D})$ has a Lipschitz continuous p^{th} derivative. Recalling the definitions of higher-order partial derivatives and multi-indices, this requirement can be expressed as

$$|\textstyle\sum_{|\alpha|=p} \partial_x^\alpha f(x) \cdot u^\alpha - \sum_{|\alpha|=p} \partial_x^\alpha f(y) \cdot u^\alpha| \leq L_p \|x - y\|_2 \quad \forall x, y \in \mathcal{D},\ u \in \mathbb{S}^{n-1}.$$

It is often referred to as a $(p+1)^{\text{th}}$-order smoothness condition [BP16, § 1.1] as it implies that any $f \in C^{p+1,p}_{L_p}(\mathcal{D}) \subseteq C^{p,p}_{L_p}(\mathcal{D})$ has a bounded $(p+1)^{\text{th}}$ derivative, that is,

$$|\textstyle\sum_{|\alpha|=p+1} \partial_x^\alpha f(x) \cdot u^\alpha| = |\partial_t^{p+1} f(x + tu)|_{t=0}| \leq L_p \quad \forall x \in \mathcal{D},\ u \in \mathbb{S}^{n-1}. \tag{2.11}$$

3 A smoothed complex-step approximation

We now use ideas from [NY83; NS17] to construct a new gradient estimator, which can be viewed as a complex-step generalization of the estimators proposed in [NY83; FKM04]. Our construction is based on the following assumption, which we assume to hold throughout the rest of the paper.

Assumption 3.1 (Analytic extension)**.** *The function $f : \mathcal{D} \to \mathbb{R}$ of problem (1.1) admits an analytic extension to the strip $\mathcal{D} \times i \cdot (-\bar{\delta}, \bar{\delta})^n$ for some $\bar{\delta} \in (0, 1)$.*

Recall from Lemma 2.5 that f admits an analytic extension to some open set $\Omega \subseteq \mathbb{C}^n$ covering $\mathcal{D}$ whenever $f \in C^\omega(\mathcal{D})$. However, unless f is entire or $\mathcal{D}$ is bounded, Ω may not contain a strip of the form envisaged in Assumption 3.1. Hence, this assumption is *not* automatically satisfied for any real analytic function $f \in C^\omega(\mathcal{D})$. The requirement $\bar{\delta} \in (0, 1)$ is unrestrictive and has the convenient consequence that $\delta^p \leq \delta^{p-1}$ for any $\delta \in (0, \bar{\delta})$ and $p \in \mathbb{Z}_{\geq 0}$. All subsequent results are based on a smoothed complex-step approximation f_δ of f, which is defined through

$$f_\delta(x) = V_n^{-1} \textstyle\int_{\mathbb{B}^n} \Re\big(f(x + i\delta y)\big) \mathrm{d}y. \tag{3.1}$$

Here, the radius $\delta \in (0, \bar{\delta})$ of the ball used for averaging represents a tuneable smoothing parameter. Given prior structural knowledge about f, one could replace $\mathbb{B}^n$ with a different compact set [HL14; Jon21]. We emphasize that the integral in (3.1) is well-defined whenever $\delta \in (0, \bar{\delta})$, which ensures that f has no singularities in the integration domain. Next, we address the approximation quality of f_δ.

Proposition 3.2 (Approximation quality of f_δ)**.** *If $f \in C^{\omega,1}_{L_1}(\mathcal{D})$ satisfies Assumption 3.1, then for f_δ defined as in (3.1) and for any fixed $x \in \mathcal{D}$ and $\kappa \in (0,1)$ there exists $C_\kappa \geq 0$ with*

$$|f_\delta(x) - f(x)| \leq \tfrac{1}{2} L_1 \delta^2 + C_\kappa \delta^4 \quad \forall \delta \in (0, \kappa\bar{\delta}].$$

Proof. By the definition of f_δ in (3.1), we have $|f_\delta(x) - f(x)| \leq V_n^{-1} \int_{\mathbb{B}^n} |\Re\big(f(x + i\delta y)\big) - f(x)| \mathrm{d}y$. The Taylor series of $f(x + i\delta y)$ around x then yields

$$\begin{aligned} \Re\left(f(x + i\delta y)\right) - f(x) &= \textstyle\sum_{k=0}^{\infty} \frac{(-1)^k \delta^{2k}}{(2k)!} \sum_{|\alpha|=2k} \partial_x^\alpha f(x) y^\alpha - f(x) \\ &= -\tfrac{1}{2}\delta^2 \langle \nabla^2 f(x) y, y\rangle + \delta^4 R(y, \delta), \end{aligned}$$

where the real-valued remainder term $R(y,\delta)$ is continuous in $y \in \mathbb{B}^n$ and $\delta \in [0, \bar{\delta})$. Substituting the last expression into the above estimate and using (2.11), we obtain

$$|f_\delta(x) - f(x)| \leq V_n^{-1} \textstyle\int_{\mathbb{B}^n} \frac{1}{2}\delta^2 L_1 + \delta^4 |R(y,\delta)| \mathrm{d}y \leq \frac{1}{2}\delta^2 L_1 + C_\kappa \delta^4 \quad \forall \delta \in (0, \kappa\bar{\delta}],$$

where the non-negative constant $C_\kappa = \max_{y \in \mathbb{B}^n} \max_{\delta \in [0, \kappa\bar{\delta}]} |R(y,\delta)|$ is finite due to continuity of $R(y,\delta)$ and compactness of $\mathbb{B}^n$ and $[0, \kappa\bar{\delta}]$. Hence, the claim follows. □

Note that if f is affine, then $f_\delta = f$. Note also that $[0, \kappa\bar{\delta}]$ is a compact subset of the set $[0, \bar{\delta})$ on which $R(y,\delta)$ is continuous in δ and that $R(y,\delta)$ may be unbounded on $[0, \bar{\delta})$. The following proposition provides an integral representation for the gradient of f_δ. It extends [NY83, § 9.3] and [FKM04, Lem. 1] to the realm of complex arithmetic.

Proposition 3.3 (Gradient of the smoothed complex-step function)**.** *If $f \in C^\omega(\mathcal{D})$ satisfies Assumption 3.1, then f_δ defined as in (3.1) is differentiable, and we have*

$$\nabla f_\delta(x) = \tfrac{n}{\delta} \mathbb{E}_{y \sim \sigma} \left[\Im\left(f(x + i\delta y)\right) y\right] \quad \forall x \in \mathcal{D},\ \delta \in (0, \bar{\delta}), \tag{3.2}$$

where σ denotes the uniform distribution on $\mathbb{S}^{n-1}$.

Proof. Any function $g \in C^1(\mathbb{R}^n)$ and vector $w \in \mathbb{R}^n$ define a vector field $v(y) = g(y) \cdot w$. The divergence theorem [Lee13, Thm. 16.32] then implies that

$$\begin{aligned} \textstyle\int_{\mathbb{B}^n} \langle w, \nabla g(y)\rangle \,\mathrm{d}y = \int_{\mathbb{B}^n} \operatorname{div}(v(y)) \,\mathrm{d}y &= S_{n-1} \textstyle\int_{\mathbb{S}^{n-1}} \langle v(y), y\rangle \,\sigma(\mathrm{d}y) \\ &= S_{n-1} \textstyle\int_{\mathbb{S}^{n-1}} g(y) \langle w, y\rangle \,\sigma(\mathrm{d}y), \end{aligned}$$

where the scaling factor S_{n-1} accounts for the fact that the uniform distribution σ is normalized on $\mathbb{S}^{n-1}$. Note also that the outward-pointing unit normal vector of $\mathbb{S}^{n-1}$ at any point $y \in \mathbb{S}^{n-1}$ is exactly y itself. As the above equation holds for all vectors $w \in \mathbb{R}^n$ and as both the leftmost and rightmost expressions are linear in w, their gradients must coincide. This reasoning implies that

$$\textstyle\int_{\mathbb{B}^n} \nabla g(y) \,\mathrm{d}y = S_{n-1} \int_{\mathbb{S}^{n-1}} g(y) y \,\sigma(\mathrm{d}y). \tag{3.3}$$

We are now ready to prove (3.2) by generalizing tools developed in [NY83; FKM04] to the complex domain. Specifically, by the definition of f_δ in (3.1) we have

$$\begin{aligned} \nabla f_\delta(x) = V_n^{-1} \textstyle\int_{\mathbb{B}^n} \nabla_x \Re\left(f(x + i\delta y)\right) \mathrm{d}y &= (V_n \delta)^{-1} \textstyle\int_{\mathbb{B}^n} \nabla_y \Im\left(f(x + i\delta y)\right) \mathrm{d}y \\ &= S_{n-1} (V_n \delta)^{-1} \textstyle\int_{\mathbb{S}^{n-1}} \Im\left(f(x + i\delta y)\right) y \,\sigma(\mathrm{d}y) \\ &= S_{n-1} (V_n \delta)^{-1} \mathbb{E}_{y \sim \sigma} \left[\Im\left(f(x + i\delta y)\right) y\right], \end{aligned}$$

where the interchange of the gradient and the integral in the first equality is permitted by the dominated convergence theorem, which applies because $\mathbb{B}^n$ is compact and because any continuously differentiable function on a compact set is Lipschitz continuous. The second equality is a direct consequence of the Cauchy-Riemann equations, and the third equality, finally, holds thanks to the generalized Achimedean principle (3.3) with pressure function $g(y) = \Im\left(f(x + i\delta y)\right)$. We finally observe that the volume of the unit ball and the surface of the unit sphere satisfy $V_n = \int_{\mathbb{B}^n} \mathrm{d}y = S_{n-1} \int_0^r r^{n-1} \mathrm{d}r = S_{n-1}/n \implies S_{n-1}/V_n = n$. Thus, the claim follows. □

Proposition 3.3 reveals that ∇f_δ admits the unbiased single-point estimator

$$g_\delta(x) = \tfrac{n}{\delta} \Im\left(f(x + i\delta y)\right) y \quad \text{with} \quad y \sim \sigma. \tag{3.4}$$

Now we show that $\nabla f_\delta(x)$ approximates $\nabla f(x)$ arbitrarily well as δ drops to 0.

Proposition 3.4 (Approximation quality of ∇f_δ)**.** *If $f \in C^{\omega,2}_{L_2}(\mathcal{D})$ satisfies Assumption 3.1, then for f_δ defined as in (3.1) and for any fixed $x \in \mathcal{D}$ and $\kappa \in (0,1)$ there exists $C_\kappa \geq 0$ with*

$$\|\nabla f_\delta(x) - \nabla f(x)\|_2 \leq \tfrac{1}{6} n L_2 \delta^2 + n C_\kappa \delta^4 \quad \forall \delta \in (0, \kappa\bar{\delta}]. \tag{3.5}$$

Proof. If we denote as usual by I_n the identity matrix in $\mathbb{R}^n$, then the covariance matrix of the uniform distribution σ on the unit sphere $\mathbb{S}^{n-1}$ can be expressed as

$$\int_{\mathbb{S}^{n-1}} y y^\mathsf{T} \sigma(\mathrm{d}y) = \int_{\mathbb{S}^{n-1}} \|y\|_2^2 \, \sigma(\mathrm{d}y) \cdot \tfrac{1}{n} I_n = \tfrac{1}{n} I_n, \tag{3.6}$$

where the two equalities hold because the sought covariance matrix must be isotropic and because $\|y\|_2 = 1$ for all $y \in \mathbb{S}^{n-1}$, respectively. Thus, the gradient of f can be represented as $\nabla f(x) = n \int_{\mathbb{S}^{n-1}} \langle \nabla f(x), y \rangle y \, \sigma(\mathrm{d}y)$. Together with Proposition 3.3, this yields the estimate

$$\begin{aligned} \|\nabla f_\delta(x) - \nabla f(x)\|_2 &= \tfrac{n}{\delta} \left\| \int_{\mathbb{S}^{n-1}} \Im\left(f(x + i\delta y)\right) y - \delta \langle \nabla f(x), y \rangle y \, \sigma(\mathrm{d}y) \right\|_2 \\ &\leq \tfrac{n}{\delta} \int_{\mathbb{S}^{n-1}} \left| \Im\left(f(x + i\delta y)\right) - \delta \langle \nabla f(x), y \rangle \right| \|y\|_2 \, \sigma(\mathrm{d}y). \end{aligned}$$

By using the Taylor series representation of $f(x + i\delta y)$ around x, we find

$$\begin{aligned} \Im\left(f(x + i\delta y)\right) - \delta \langle \nabla f(x), y \rangle &= \sum_{k=0}^\infty \tfrac{(-1)^k \delta^{2k+1}}{(2k+1)!} \sum_{|\alpha|=2k+1} \partial_x^\alpha f(x) y^\alpha - \delta \langle \nabla f(x), y \rangle \\ &= \sum_{k=1}^\infty \tfrac{(-1)^k \delta^{2k+1}}{(2k+1)!} \sum_{|\alpha|=2k+1} \partial_x^\alpha f(x) y^\alpha \\ &= -\tfrac{1}{6} \delta^3 \sum_{|\alpha|=3} \partial_x^\alpha f(x) y^\alpha + \delta^5 R(y, \delta), \end{aligned}$$

where the real-valued remainder term $R(y, \delta)$ is continuous in $y \in \mathbb{B}^n$ and $\delta \in [0, \bar{\delta})$. Substituting the last expression into the above and using (2.11), we obtain

$$\begin{aligned} \|\nabla f_\delta(x) - \nabla f(x)\|_2 &\leq \tfrac{n}{\delta} \int_{\mathbb{S}^{n-1}} \left(\tfrac{1}{6} \delta^3 L_2 + \delta^5 \left| R(y, \delta) \right| \right) \|y\|_2 \, \sigma(\mathrm{d}y) \\ &\leq \tfrac{1}{6} \delta^2 n L_2 + n C_\kappa \delta^4 \quad \forall \delta \in (0, \kappa\bar{\delta}], \end{aligned}$$

where the non-negative constant $C_\kappa = \max_{y \in \mathbb{S}^{n-1}} \max_{\delta \in [0, \kappa\bar{\delta}]} |R(y, \delta)|$ is again finite due to the continuity of $R(y, \delta)$ and the compactness of $\mathbb{S}^{n-1}$ and $[0, \kappa\bar{\delta}]$. □

Proposition 3.4 implies that the *single-point* estimator (3.4) incurs only errors of the order $O(\delta^2)$ on average. Equally small errors were attained in [NS17] for $f \in C^{2,2}_{L_2}$ by using Gaussian smoothing and a *multi-point* estimator. Unfortunately, the latter is susceptible to cancellation effects. Proposition 3.4 also implies that $\lim_{\delta \downarrow 0} \nabla f_\delta(x) = \nabla f(x)$. In addition, one readily verifies that if f is quadratic (that is, if $L_2 = 0$), then $\nabla f_\delta(x) = \nabla f(x)$ for all $x \in \mathcal{D}$ and $\delta \in (0, \bar{\delta})$. The single-point estimator $g_\delta(x)$ introduced in (3.4) is unbiased by construction. In addition, as for the multi-point estimator proposed in [NS17], the second moment of $g_\delta(x)$ admits a convenient bound.

Corollary 3.5 (Second moment of $g_\delta(x)$)**.** *If $f \in C^{\omega,2}_{L_2}(\mathcal{D})$ satisfies Assumption 3.1, then for g_δ as in (3.4) and for any fixed $x \in \mathcal{D}$ and $\kappa \in (0,1)$ we have*

$$\begin{aligned}\mathbb{E}_{y\sim\sigma}\left[\|g_\delta(x)\|_2^2\right] \leq n^2(\tfrac{1}{6}L_2\delta^2 + C_\kappa\delta^4)^2 + n\|\nabla f(x)\|_2^2 \\ + 2n^2\left(\tfrac{1}{6}L_2\delta^2 + C_\kappa\delta^4\right)\|\nabla f(x)\|_2,\end{aligned} \tag{3.7}$$

where $C_\kappa \geq 0$ is the same constant as in Proposition 3.4.

Proof. Using the definition of g_δ and the fact that $\|y\|_2 = 1 \; \forall\, y \in \mathbb{S}^{n-1}$, we find

$$\mathbb{E}_{y\sim\sigma}\left[\|g_\delta(x)\|_2^2\right] = \tfrac{n^2}{\delta^2}\mathbb{E}_{y\sim\sigma}[(\Im\left(f(x+i\delta y)\right))^2]. \tag{3.8}$$

By essentially the same arguments as in the proof of Proposition 3.4, we further have

$$\begin{aligned}|\Im\left(f(x+i\delta y)\right)| &= |\Im\left(f(x+i\delta y)\right) - \langle\nabla f(x), \delta y\rangle + \langle\nabla f(x), \delta y\rangle| \\ &\leq \left|\tfrac{1}{6}\delta^3 L_2 + \delta^5 C_\kappa\right| + \left|\langle\nabla f(x), \delta y\rangle\right|.\end{aligned}$$

Squaring the above and applying the Cauchy-Schwarz inequality yields

$$\begin{aligned}|\Im\left(f(x+i\delta y)\right)|^2 \leq \left(\tfrac{1}{6}\delta^3 L_2 + \delta^5 C_\kappa\right)^2 + \langle\nabla f(x), \delta y\rangle^2 \\ + 2\delta\left(\tfrac{1}{6}\delta^3 L_2 + \delta^5 C_\kappa\right)\|\nabla f(x)\|_2\|y\|_2.\end{aligned}$$

The claim then follows from substituting the above into (3.8) and using (3.6). □

In analogy to Proposition 3.4, one readily verifies that if f is quadratic (*i.e.*, if $L_2 = 0$), then the right hand side of (3.7) vanishes. Under a third-order smoothness condition, there exist multi-point estimators that satisfy a bound akin to (3.7) [NS17, Thm. 4.3].

Unlike the smooth approximations proposed in [NS17], the smoothed complex-step approximation f_δ does frequently *not* belong to the same function class as f. For example, even though the Lorentzian function $f(x) = 1/(1+x^2)$ has a Lipschitz continuous gradient with $L_1 = 2$, the Lipschitz modulus of its approximation f_δ strictly exceeds 2 for some values of δ close to 1 because f has two poles at i and $-i$. Similarly, f_δ does not necessarily inherit convexity from f.

Example 3.6 (Loss of convexity)**.** *If $f \in C^\omega(\mathbb{R})$ is entire, then it has a globally convergent power series representation with real coefficients. Consequently, f satisfies*

$$\Re(f(x+i\delta y)) = \textstyle\sum_{k=0}^\infty (-1)^k \frac{f^{(2k)}(x)}{(2k)!}(\delta y)^{2k}.$$

In the special case when $f(x) = x^2$, the complex-step approximation $\Re(f(x+i\delta y)) = x^2 - (\delta y)^2$ inherits convexity from f regardless of the choice of $\delta > 0$ and $y \in \mathbb{R}$. Thus, f_δ is also convex because convexity is preserved by integration. However, if $f(x) = x^4$, then we

Algorithm 1 Imaginary zeroth-order optimization

Input: initial iterate $x_1 \in \mathcal{X}$, stepsizes $\{\mu_k\}_{k\in\mathbb{Z}_{\geq 0}}$, smoothing parameters $\{\delta_k\}_{k\in\mathbb{Z}_{\geq 0}}$
for $k = 1, 2, \ldots, K-1$ **do**
  sample $y_k \sim \sigma$
  set $g_{\delta_k}(x_k) = \frac{n}{\delta_k}\Im\left(f(x_k + i\delta_k y_k)\right) y_k$
  set $x_{k+1} = \Pi_{\mathcal{X}}\left(x_k - \mu_k\, g_{\delta_k}(x_k)\right)$
end for
Output: last iterate x_K and averaged iterate $\bar{x}_K = \frac{1}{K}\sum_{k=1}^{K} x_k$

find $\Re(f(x+i\delta y)) = x^4 - 6x^2(\delta y)^2 + (\delta y)^4$*, which fails to be convex in* x *for any* $\delta > 0$ *and* $y \neq 0$*. In this case,* f_δ *remains non-convex despite the smoothing. Finally, if* f *is strongly convex* (e.g.*, if* $f(x) = x^2 + x^4$*), then one readily verifies that* $\Re(f(x+i\delta y))$ *is convex in* x *provided that* δ *is sufficiently small.*

If f_δ inherited convexity from f, one could simply incorporate the estimator (3.4) into the algorithms studied in [NS17, § 5], and the corresponding convergence analysis would carry over with minor modifications. As the smoothed complex-step approximation may destroy convexity, however, a different machinery is needed here.

4 Convex optimization

We now study the convergence properties of zeroth-order algorithms for solving problem (1.1) under the assumption that f is a convex function on $\mathcal{D}$ and $\mathcal{X}$ is a non-empty closed convex subset of $\mathcal{D}$. Our methods mimic existing algorithms developed in [NS17] but use the single-point estimator g_δ defined in (3.4) instead of a multi-point estimator that may suffer from cancellation effects. Our method is described in Algorithm 1, where $\Pi_{\mathcal{X}} : \mathcal{D} \to \mathcal{X}$ denotes the Euclidean projection onto $\mathcal{X}$. Note that $\Pi_{\mathcal{X}}$ reduces to the identity operator if $\mathcal{X} = \mathcal{D}$.

In the remainder we will assume that the iterates $\{x_k\}_{k\in\mathbb{Z}_{>0}}$ generated by Algorithm 1 as well as all samples $\{y_k\}_{k\in\mathbb{Z}_{>0}}$ and the corresponding gradient estimators $\{g_{\delta_k}(x_k)\}_{k\in\mathbb{Z}_{>0}}$ represent random objects on an abstract filtered probability space $(\Omega, \mathcal{F}, \{\mathcal{F}_k\}_{k\in\mathbb{Z}_{>0}}, \mathbb{P})$, where $\mathcal{F}_k$ denotes the σ-algebra generated by the independent and identically distributed samples $y_1, \ldots, y_{k-1}$. Therefore, x_k is $\mathcal{F}_k$-measurable. In the following, we use $\mathbb{E}[\cdot]$ to denote the expectation operator with respect to $\mathbb{P}$.

Theorem 4.1 (Convergence rate of Algorithm 1 for convex optimization)**.** *Suppose that* f *is a convex, real analytic function satisfying Assumption 3.1 as well as the Lipschitz conditions (2.6) and (2.9) with* $L_1 > 0$ *and* $L_2 \geq 0$*. Also assume that* $\mathcal{X}$ *is non-empty, closed and convex and that there exists* $x^\star \in \mathcal{X}$ *with* $\nabla f(x^\star) = 0$*. Denote by* $\{x_k\}_{k\in\mathbb{Z}_{>0}}$ *the iterates generated by Algorithm 1 with constant stepsize* $\mu_k = \mu = 1/(2nL_1)$ *and adaptive smoothing parameter* $\delta_k \in (0, \kappa\bar{\delta}]$ *for all* $k \in \mathbb{Z}_{>0}$*, where* $\kappa \in (0,1)$*, and define* $R = \|x_1 - x^\star\|_2$*. Then, the following hold for all* $K \in \mathbb{Z}_{>0}$*.*

(i) There is a constant $C_1 \geq 0$ *such that*

$$\begin{aligned}\mathbb{E}\left[f(\bar{x}_K) - f(x^\star)\right] \leq{}& \tfrac{1}{\mu K}R^2 + \tfrac{1}{K}C_1 nR\textstyle\sum_{k=1}^{K}\delta_k^2 + \tfrac{1}{K}\mu C_1^2 n^2 (\sum_{k=1}^{K}\delta_k^2)^2 \\ &+ \tfrac{1}{K}\mu C_1 C_2 n^2 (\textstyle\sum_{k=1}^{K}\delta_k^2)(\sum_{k=1}^{K}\delta_k^4)^{\frac{1}{2}} + \tfrac{1}{K}\mu C_2^2 n^2 \textstyle\sum_{k=1}^{K}\delta_k^4.\end{aligned}$$

(ii) If $\delta_k = \delta$ *for all* $k \in \mathbb{Z}_{>0}$*, then we have*

$$\mathbb{E}\left[f(\bar{x}_K) - f(x^\star)\right] \leq \tfrac{1}{K}2nL_1R^2 + C_1 nR\delta^2 + \tfrac{1}{L_1}(1+\sqrt{K})^2 C_1^2 n\delta^4.$$

(iii) If $\delta_k = \delta/k$ for all $k \in \mathbb{Z}_{>0}$, then there is a constant $C_2 \geq 0$ such that

$$\mathbb{E}\left[f(\bar{x}_K) - f(x^\star)\right] \leq \tfrac{n}{K}\left(\sqrt{2L_1}R + C_2\delta^2\right)^2.$$

Under the assumptions of Theorem 4.1, problem (1.1) is convex and $x^\star$ represents a global minimizer. Note, however, that $\mathcal{X}$ may not contain any $x^\star$ with $\nabla f(x^\star) = 0$ even if $\mathcal{X}$ is compact. This is usually the case if the global minimum of (1.1) is attained at the boundary of $\mathcal{X}$. If $x^\star$ is not unique, one should set $R = \|x_1 - P^\star(x_1)\|_2$ for the bounds not to be trivial, with $P^\star(x_1) = \operatorname{argmin}_{x^\star} \|x^\star - x_1\|_2^2$, which is well-defined since f is convex, real analytic. Explicit formulas for C_1 and C_2 in terms of κ, L_2 etc. are derived in the proof of Theorem 4.1.

Proof of Theorem 4.1. For ease of notation, we define $r_k = \|x_k - x^\star\|_2$ for all $k \in \mathbb{Z}_{>0}$. We prove the theorem first under the simplifying assumption that $\mathcal{X} = \mathcal{D}$, which implies the projection onto $\mathcal{X}$ becomes obsolete, that is, $x_{k+1} = x_k - \mu_k \cdot g_{\delta_k}(x_k)$. Thus, we have

$$\begin{aligned}\mathbb{E}\left[r_{k+1}^2 \,\middle|\, \mathcal{F}_k\right] &= \mathbb{E}\left[r_k^2 - 2\mu_k\langle g_{\delta_k}(x_k), x_k - x^\star\rangle + \mu_k^2 \|g_{\delta_k}(x_k)\|_2^2 \,\middle|\, \mathcal{F}_k\right] \\ &= r_k^2 - 2\mu_k\langle \nabla f_{\delta_k}(x_k), x_k - x^\star\rangle + \mu_k^2\, \mathbb{E}\left[\|g_{\delta_k}(x_k)\|_2^2 \,\middle|\, \mathcal{F}_k\right],\end{aligned}$$

where the second equality follows from (3.2), the definition of $g_{\delta_k}(x_k)$ and the $\mathcal{F}_k$-measurability of x_k and r_k. The Cauchy-Schwartz inequality then implies that

$$\begin{aligned}&\mathbb{E}\left[r_{k+1}^2 \,\middle|\, \mathcal{F}_k\right] \\ &\leq r_k^2 - 2\mu_k\langle \nabla f(x_k), x_k - x^\star\rangle + 2\mu_k\|\nabla f_{\delta_k}(x_k) - \nabla f(x_k)\|_2\, r_k + \mu_k^2\, \mathbb{E}\left[\|g_{\delta_k}(x_k)\|_2^2 \,\middle|\, \mathcal{F}_k\right] \\ &\leq r_k^2 - 2\mu_k\left(f(x_k) - f(x^\star)\right) + 2\mu_k\left(\tfrac{1}{6}nL_2\delta_k^2 + nC_\kappa\delta_k^4\right) r_k \\ &\quad + \mu_k^2 n^2\left(\left(\tfrac{1}{6}L_2\delta_k^2 + C_\kappa\delta_k^4\right)^2 + \tfrac{1}{n}\|\nabla f(x_k)\|_2^2 + 2\left(\tfrac{1}{6}L_2\delta_k^2 + C_\kappa\delta_k^4\right)\|\nabla f(x_k)\|_2\right) \\ &\leq r_k^2 - 2\mu_k\left(f(x_k) - f(x^\star)\right) + 2n\mu_k\delta_k^2\left(\tfrac{1}{6}L_2 + C_\kappa\delta_k^2 + nL_1\mu_k\left(\tfrac{1}{6}L_2 + C_\kappa\delta_k^2\right)\right) r_k \\ &\quad + \mu_k^2 n^2\left(\delta_k^4\left(\tfrac{1}{6}L_2 + C_\kappa\delta_k^2\right)^2 + \tfrac{1}{n}2L_1(f(x_k) - f(x^\star))\right),\end{aligned}$$

where the second inequality exploits the convexity of f as well as Proposition 3.4 and Corollary 3.5, while the third inequality follows from the estimates (2.6) and (2.8), which imply that $\|\nabla f(x_k)\|_2 \leq L_1\|x_k - x^\star\|_2$ and $2L_1(f(x_k) - f(x^\star)) \geq \|\nabla f(x_k)\|_2^2$, respectively. To simplify notation, we now introduce the constant $C_1 = \frac{1}{2}L_2 + 3C_\kappa$, which upper bounds $\frac{1}{2}L_2 + 3C_\kappa\delta_k^2$ and $\frac{1}{6}L_2 + C_\kappa\delta_k^2$ for any $k \in \mathbb{Z}_{>0}$ because all smoothing parameters belong to the interval $[-1, 1]$. Recalling that the stepsize is constant and equal to $\mu = 1/(2nL_1)$, the above display equation thus simplifies to

$$\mathbb{E}\left[r_{k+1}^2 \,\middle|\, \mathcal{F}_k\right] \leq r_k^2 - \mu\left(f(x_k) - f(x^\star)\right) + n\mu\delta_k^2 C_1 r_k + \mu^2 n^2 C_1^2 \delta_k^4. \tag{4.1}$$

Taking unconditional expectations and rearranging terms then yields

$$\begin{aligned}\mathbb{E}\left[f(x_k) - f(x^\star)\right] &\leq \tfrac{1}{\mu}\left(\mathbb{E}\left[r_k^2\right] - \mathbb{E}\left[r_{k+1}^2\right]\right) + nC_1\delta_k^2\mathbb{E}\left[r_k\right] + \mu n^2 C_1^2\delta_k^4 \\ &\leq \tfrac{1}{\mu}\left(\mathbb{E}\left[r_k^2\right] - \mathbb{E}\left[r_{k+1}^2\right]\right) + nC_1\delta_k^2\sqrt{\mathbb{E}\left[r_k^2\right]} + \mu n^2 C_1^2\delta_k^4.\end{aligned}$$

Next, choose any $k' \in \mathbb{Z}_{>0}$ and sum the above inequalities over all $k \leq k' - 1$ to obtain

$$\textstyle\sum_{k=1}^{k'-1}\mathbb{E}\left[f(x_k) - f(x^\star)\right] \leq \tfrac{1}{\mu}\left(r_1^2 - \mathbb{E}\left[r_{k'}^2\right]\right) + C_1 n\sum_{k=1}^{k'-1}\delta_k^2\sqrt{\mathbb{E}\left[r_k^2\right]} + \mu C_1^2 n^2\sum_{k=1}^{k'-1}\delta_k^4. \tag{4.2}$$

Clearly, the inequality (4.2) remains valid if we lower bound its left hand side by 0 and upper bound its right hand side by increasing the upper limits of the two sums to k'. We then obtain $\mathbb{E}[r_{k'}^2] \leq r_1^2 + \mu C_1 n \sum_{k=1}^{k'} \delta_k^2 \sqrt{\mathbb{E}\,[r_k^2]} + \mu^2 C_1^2 n^2 \sum_{k=1}^{k'} \delta_k^4$. Setting $t_k = \sqrt{\mathbb{E}\,[r_k^2]}$ and $\nu_k = \mu C_1 n \delta_k^2$ for all $k \in \mathbb{Z}_{>0}$ and defining $T_{k'} = r_1^2 + \mu^2 C_1^2 n^2 \sum_{k=1}^{k'} \delta_k^4$ for all $k' \in \mathbb{Z}_{>0}$, we may use Lemma A.1 to conclude that

$$\begin{aligned}\sqrt{\mathbb{E}\,[r_{k'}^2]} &\leq \tfrac{1}{2}\mu C_1 n \textstyle\sum_{k=1}^{k'} \delta_k^2 + \Big(r_1^2 + \mu^2 C_1^2 n^2 \sum_{k=1}^{k'} \delta_k^4 + (\tfrac{1}{2}\mu C_1 n \textstyle\sum_{k=1}^{k'} \delta_k^2)^2\Big)^{\frac{1}{2}} \\ &\leq \mu C_1 n \textstyle\sum_{k=1}^{K} \delta_k^2 + r_1 + (\mu^2 C_1^2 n^2 \sum_{k=1}^{K} \delta_k^4)^{\frac{1}{2}} \quad \forall k \leq K,\end{aligned}$$

where the second inequality holds because $\sqrt{a+b+c} \leq \sqrt{a} + \sqrt{b} + \sqrt{c}$ for all $a, b, c \geq 0$ and because the sums increase when we increase their upper limits from k' to K. Next, consider the estimate (4.2) for $k' = K+1$, replace $\mathbb{E}[r_{K+1}^2]$ with its trivial lower bound 0 and replace $\sqrt{\mathbb{E}[r_k^2]}$ with the above upper bound for every $k \leq K$. Noting that $r_1 = R$ and dividing by K then yields

$$\begin{aligned}&\tfrac{1}{K}\textstyle\sum_{k=1}^{K} \mathbb{E}\,[f(x_k) - f(x^\star)] \\ &\leq \tfrac{1}{\mu K} R^2 + \tfrac{1}{K}\mu C_1^2 n^2 \textstyle\sum_{k=1}^{K} \delta_k^4 \\ &\quad + \tfrac{1}{K} C_1 n \textstyle\sum_{k=1}^{K} \delta_k^2 \Big(\mu C_1 n \sum_{k=1}^{K} \delta_k^2 + R + (\mu^2 C_1^2 n^2 \sum_{k=1}^{K} \delta_k^4)^{\frac{1}{2}}\Big) \\ &= \tfrac{1}{\mu K} R^2 + \tfrac{1}{K} C_1 n R \textstyle\sum_{k=1}^{K} \delta_k^2 + \tfrac{1}{K}\mu C_1^2 n^2 (\sum_{k=1}^{K} \delta_k^2)^2 \\ &\quad + \tfrac{1}{K}\mu C_1^2 n^2 (\textstyle\sum_{k=1}^{K} \delta_k^2)(\sum_{k=1}^{K} \delta_k^4)^{\frac{1}{2}} + \tfrac{1}{K}\mu C_1^2 n^2 \textstyle\sum_{k=1}^{K} \delta_k^4.\end{aligned}$$

As $\mathbb{E}[f(\bar{x}_K) - f(x^\star)] \leq \frac{1}{K}\sum_{k=1}^{K} \mathbb{E}[f(x_k) - f(x^\star)]$ by Jensen's inequality, assertion *(i)* thus follows. If $\delta_k = \delta \in (0, \kappa\delta]$ for all $k \in \mathbb{Z}_{>0}$, then assertion *(i)* implies that

$$\begin{aligned}\mathbb{E}\,[f(\bar{x}_K) - f(x^\star)] &\leq \tfrac{1}{\mu K} R^2 + C_1 n R \delta^2 + C_1^2 K \mu n^2 \delta^4 + C_1^2 \sqrt{K} \mu n^2 \delta^4 + C_1^2 \mu n^2 \delta^4 \\ &\leq \tfrac{1}{\mu K} R^2 + C_1 n R \delta^2 + (C_1\sqrt{K} + C_1)^2 \mu n^2 \delta^4 \\ &\leq \tfrac{1}{K} 2 n L_1 R^2 + C_1 n R \delta^2 + (C_1\sqrt{K} + C_1)^2 \tfrac{1}{L_1} n \delta^4 \\ &\leq \tfrac{1}{K} 2 n L_1 R^2 + C_1 n R \delta^2 + C_1^2 (1 + \sqrt{K})^2 \tfrac{1}{L_1} n \delta^4,\end{aligned}$$

where the last two inequalities exploit the assumption $\mu = 1/(2nL_1)$. Thus, assertion *(ii)* follows. Next, assume that $\delta_k = \delta/k$ for all $k \in \mathbb{Z}_{>0}$. In analogy to the proof of assertion *(ii)*, we combine assertion *(i)* with the standard zeta function inequalities (A.1) to conclude that

$$\begin{aligned}\mathbb{E}\,[f(\bar{x}_K) - f(x^\star)] \leq& \tfrac{1}{\mu K} R^2 + \tfrac{1}{6}\pi^2 C_1 \tfrac{1}{K} n R \delta^2 + \tfrac{1}{90}\pi^4 C_1^2 \tfrac{1}{K}\mu n^2 \delta^4 \\ &+ \tfrac{1}{36}\pi^4 C_1^2 \tfrac{1}{K}\mu n^2 \delta^4 + \tfrac{1}{6\sqrt{90}}\pi^4 C_1^2 \tfrac{1}{K}\mu n^2 \delta^4 \\ &\leq \tfrac{n}{K} 2 L_1 R^2 + \tfrac{n}{K} R \tfrac{1}{6}\pi^2 C_1 \delta^2 + \tfrac{n}{K} R \pi^4 (\tfrac{1}{6} C_1 + \tfrac{1}{\sqrt{90}} C_1)^2 \tfrac{1}{2L_1} \delta^4 \\ &\leq \tfrac{n}{K} (\sqrt{2L_1} R + C_2 \delta^2)^2,\end{aligned}$$

where $C_2 = \pi^2(C_1/3 + C_1/\sqrt{90})/\sqrt{2L_1}$. The third inequality holds because $\mu = 1/(2nL_1)$. Thus, assertion *(iii)* follows. This completes the proof for $\mathcal{X} = \mathcal{D}$.

In the last part of the proof we show that the three assertions remain valid when $\mathcal{X}$ is a non-empty closed convex subset of $\mathcal{D}$. Indeed, as the projection $\Pi_{\mathcal{X}}$ onto $\mathcal{X}$ is contractive, we have

$$r_{k+1}^2 = \|x_{k+1} - x^\star\|_2^2 = \|\Pi_{\mathcal{X}}(x_k - \mu_k g_{\delta_k}(x_k)) - \Pi_{\mathcal{X}}(x^\star)\|_2^2 \leq \|x_k - \mu_k g_{\delta_k}(x_k) - x^\star\|_2^2.$$

Thus, all arguments used above carry over trivially to situations where $\mathcal{X} \neq \mathcal{D}$. □

Theorem 4.1 *(iii)* shows that if δ_k decays as $O(1/k)$, then one needs $O\left(nL_1R^2/\epsilon\right)$ iterations to guarantee that $\mathbb{E}\left[f(\bar{x}_K) - f(x^\star)\right] \leq \epsilon$. This is the first-order complexity scaled by n [Nes03, § 2.1.5]. Theorem 4.1 can be extended to a larger class of convex optimization problems by relaxing the assumption of constant stepsizes [Jon21]. In particular, it can be extended to constrained optimization problems whose constraints are binding at optimality, in which case $\nabla f(x^\star) \neq 0$; see also Example 7.4 below.

5 Strongly convex optimization

We now extend the results from Section 4 to analytic objective functions f that are τ-strongly convex over their domain $\mathcal{D}$ for some $\tau > 0$, *i.e.*, we assume that $f(y) \geq f(x) + \langle \nabla f(x), y - x\rangle + \frac{1}{2}\tau\|y - x\|_2^2 \ \forall x, y \in \mathcal{D}$. If y is a stationary point with $\nabla f(y) = 0$, then τ-strong convexity ensures that

$$f(y) - f(x) \geq \tfrac{1}{2}\tau\|y - x\|_2^2 \quad \forall x \in \mathcal{D}, \tag{5.1}$$

which in turn implies via the Polyak-Łojasiewicz inequality $\|\nabla f(x)\|_2^2 \geq 2\tau(f(x) - f(y))$ for τ-strongly convex functions [Nes03, Eq. 2.1.19] that

$$\|\nabla f(x)\|_2 \geq \tau\|y - x\|_2. \tag{5.2}$$

Theorem 5.1 (Convergence rate of Algorithm 1 for strongly convex optimization)**.** *Suppose that all assumptions of Theorem 4.1 (iii) are satisfied and that f is τ-strongly convex for some $\tau > 0$. Then, there is a constant $C \geq 0$ such that the following inequality holds for all $K \in \mathbb{Z}_{>0}$.*

$$\mathbb{E}[f(x_K) - f(x^\star)] \leq \tfrac{1}{2}L_1\left(\delta^2 C + (1 - \tfrac{\tau}{4nL_1})^{K-1}\left(R^2 - \delta^2 C\right)\right)$$

An explicit formula for C in terms of n, L_1, L_2 and τ is derived in the proof.

Proof of Theorem 5.1. As in the proof of Theorem 4.1, we set $C_1 = 3(\frac{1}{6}L_2 + C_\kappa)$ and $r_k = \|x_k - x^\star\|_2$ for all $k \in \mathbb{Z}_{>0}$, and we initially assume that $\mathcal{X} = \mathcal{D}$. Combining the estimate (4.1) from the proof of Theorem 4.1 with the strong convexity condition (5.1) yields $\mathbb{E}\left[r_{k+1}^2 | \mathcal{F}_k\right] \leq \left(1 - \frac{\mu\tau}{2}\right) r_k^2 + \mu C_1 n\delta_k^2 r_k + \mu^2 C_1^2 n^2 \delta_k^4$. By taking unconditional expectations and applying Jensen's inequality, we then find

$$\begin{aligned}
\mathbb{E}[r_{k+1}^2] &\leq \left(1 - \tfrac{\mu\tau}{2}\right)\mathbb{E}[r_k^2] + \mu C_1 n\delta_k^2\sqrt{\mathbb{E}[r_k^2]} + \mu^2 C_1^2 n^2\delta_k^4 && \text{(5.3a)}\\
&\leq \mathbb{E}[r_k^2] + \mu C_1 n\delta_k^2\sqrt{\mathbb{E}[r_k^2]} + \mu^2 C_1^2 n^2\delta_k^4. && \text{(5.3b)}
\end{aligned}$$

Next, choose any $k' \in \mathbb{Z}_{>0}$ and sum the above inequalities over all $k \leq k' - 1$ to obtain

$$\begin{aligned}
\mathbb{E}[r_{k'}^2] &\leq r_1^2 + \mu C_1 n\textstyle\sum_{k=1}^{k'-1}\delta_k^2\sqrt{\mathbb{E}[r_k^2]} + \mu^2 C_1^2 n^2\sum_{k=1}^{k'-1}\delta_k^4\\
&\leq r_1^2 + \mu C_1 n\textstyle\sum_{k=1}^{k'}\delta_k^2\sqrt{\mathbb{E}[r_k^2]} + \mu^2 C_1^2 n^2\sum_{k=1}^{k'}\delta_k^4.
\end{aligned}$$

By using the same reasoning as in the proof of Theorem 4.1, the last bound implies

$$\sqrt{\mathbb{E}\left[r_{k'}^2\right]} \leq \mu C_1 n\textstyle\sum_{k=1}^{k'}\delta_k^2 + r_1 + (\mu^2 C_1^2 n^2\sum_{k=1}^{k'}\delta_k^4)^{\frac{1}{2}}.$$

Substituting this inequality into (5.3a) for $k = k'$ and noting that $r_1 = R$ yields

$$\begin{aligned}
\mathbb{E}[r_{k'+1}^2] \leq \left(1 - \tfrac{\mu\tau}{2}\right)\mathbb{E}[r_{k'}^2] &+ \mu^2 C_1^2 n^2\delta_{k'}^4\\
&+ \mu C_1 n\delta_{k'}^2\left(\mu C_1 n\textstyle\sum_{k=1}^{k'}\delta_k^2 + R + (\mu^2 C_1^2 n^2\sum_{k=1}^{k'}\delta_k^4)^{\frac{1}{2}}\right).
\end{aligned}$$

As $\delta_k = \delta/k$ for all $k \in \mathbb{Z}_{>0}$ and as the constant stepsize satisfies $\mu = 1/(2nL_1)$, we may then use the standard zeta function inequalities (A.1) to obtain

$$\begin{aligned}\mathbb{E}[r_{k'+1}^2] \leq & (1 - \tfrac{\tau}{4nL_1})\mathbb{E}[r_{k'}^2] + C_1^2 \tfrac{\delta^4}{4L_1^2(k')^4} + C_1^2 \tfrac{\pi^2\delta^4}{24L_1^2(k')^2} + C_1 R \tfrac{\delta^2}{2L_1(k')^2} + C_1^2 \tfrac{\pi^2\delta^4}{4\sqrt{90}L_1^2(k')^2} \\ \leq & (1 - \tfrac{\tau}{4nL_1})\mathbb{E}[r_{k'}^2] + C_1 R \tfrac{\delta^2}{L_1} + 3C_1^2 \tfrac{\delta^4}{L_1^2},\end{aligned}$$

where the last inequality follows from the elementary bounds $\frac{1}{2(k')^2} < 1$, $\frac{1}{4(k')^4} < 1$, $\frac{\pi^2}{24(k')^2} < 1$ and $\pi^2/(4\sqrt{90}(k')^2) < 1$. As $|\delta| < 1$, we may set $C = \frac{4n}{\tau}(C_1 R + 3C_1^2/L_1)$ to obtain

$$\mathbb{E}[r_{k'+1}^2] \leq (1 - \tfrac{\tau}{4nL_1})\mathbb{E}[r_{k'}^2] + \tfrac{\tau}{4nL_1}\delta^2 C.$$

Taken together, the Lipschitz inequality (2.6) and the strong convexity inequality (5.2) imply that $\tau \leq L_1$, which in turn ensures that $\tau/(4nL_1) < 1$. Hence, the above inequality implies $\mathbb{E}[r_{k'+1}^2] - \delta^2 C \leq (1 - \frac{\tau}{4nL_1})\left(\mathbb{E}([r_{k'}^2] - \delta^2 C\right)$. As this estimate holds for all $k' < K$, we may finally conclude that

$$\mathbb{E}[r_K^2] - \delta^2 C \leq (1 - \tfrac{\tau}{4nL_1})\left(\mathbb{E}[r_{K-1}^2] - \delta^2 C\right) \leq \cdots \leq (1 - \tfrac{\tau}{4nL_1})^{K-1}(R - \delta^2 C).$$

The claim then follows by combining this inequality with the estimate $\mathbb{E}[f(x_K) - f(x^\star)] \leq \frac{1}{2}L_1\mathbb{E}[r_K^2]$, which follows from the Lipschitz condition (2.7). This completes the proof for $\mathcal{X} = \mathcal{D}$. To show that the claim remains valid when $\mathcal{X}$ is a non-empty closed convex subset of $\mathcal{D}$, we may proceed as in the proof of Theorem 4.1. Details are omitted for brevity. □

By Theorem 5.1 and the construction of C, we can enforce $\mathbb{E}[f(x_K) - f(x^\star)] \leq \epsilon$ for a given tolerance $\epsilon > 0$ by selecting a sufficiently small smoothing parameter $\delta \leq O(\sqrt{\epsilon\tau/(nL_1^2)})$ and by running Algorithm 1 over $O(nL_1/\tau \log(L_1 R^2/\epsilon))$ iterations.

6 Non-convex optimization

We now extend the convergence guarantees for Algorithm 1 to unconstrained non-convex optimization problems. Our proof strategy differs from the one in [NS17] as the smoothed objective function f_δ does not necessarily admit a Lipschitz continuous gradient. In this setting, convergence can still be guaranteed if the initial iterate x_1 is sufficiently close to some global minimizer $x^\star$.

Theorem 6.1 (Convergence rate of Algorithm 1 for nonconvex optimization)**.** *Suppose that all assumptions of Theorem 4.1 (iii) hold, but assume that f may be non-convex, $\mathcal{X} = \mathcal{D}$ and $\mu_k = \mu = 1/(nL_1)$ for all $k \in \mathbb{Z}_{>0}$. Define $F = f(x_1) - f(x^\star)$, where $x^\star$ is a global minimizer of problem (1.1). If $\|\nabla f(x_1)\|_2^2 \leq 2nL_1 F$, then there is a constant $C \geq 0$ such that for all $K \in \mathbb{Z}_{>0}$ we have*

$$\tfrac{1}{K}\textstyle\sum_{k=1}^K \mathbb{E}\left[\|\nabla f(x_k)\|_2^2\right] \leq \tfrac{n}{K}\left(2L_1 F + \delta^2 C\right).$$

The dependence of C on n, L_1, L_2 and F can be derived from the proof of Theorem 6.1.

Proof of Theorem 6.1. As $\mathcal{X} = \mathcal{D}$, the iterates of Algorithm 1 satisfy $x_{k+1} = x_k - \mu_k \cdot g_{\delta_k}(x_k)$. In addition, as f has a Lipschitz continuous gradient, the Lipschitz inequality (2.7) implies that

$$\begin{aligned}f(x_{k+1}) \leq & f(x_k) - \mu_k\langle\nabla f(x_k), g_{\delta_k}(x_k)\rangle + \tfrac{1}{2}\mu_k^2 L_1\|g_{\delta_k}(x_k)\|_2^2 \\ = & f(x_k) - \mu_k\|\nabla f(x_k)\|_2^2 - \mu_k\langle\nabla f(x_k), g_{\delta_k}(x_k) - \nabla f(x_k)\rangle + \tfrac{1}{2}\mu_k^2 L_1\|g_{\delta_k}(x_k)\|_2^2.\end{aligned}$$

Taking conditional expectations on both sides of this expression, recalling that g_{δ_k} is an unbiased estimator for ∇f_{δ_k} conditional on $\mathcal{F}_k$ and applying the Cauchy-Schwarz inequality then yields

$$\begin{aligned}\mathbb{E}\left[f(x_{k+1})|\mathcal{F}_k\right] \leq & f(x_k) - \mu_k\|\nabla f(x_k)\|_2^2 \\ & + \mu_k\|\nabla f(x_k)\|_2\|\nabla f_{\delta_k}(x_k) - \nabla f(x_k)\|_2 + \tfrac{1}{2}\mu_k^2 L_1 \mathbb{E}[\|g_{\delta_k}(x_k)\|_2^2|\mathcal{F}_k].\end{aligned}$$

Defining $C_0 = \frac{1}{6}L_2 + C_\kappa$, we may use the estimates (3.5) and (3.7) to obtain

$$\begin{aligned}\mathbb{E}\left[f(x_{k+1})|\mathcal{F}_k\right] \leq & f(x_k) - \mu_k\|\nabla f(x_k)\|_2^2 + \mu_k C_0 n\delta_k^2\|\nabla f(x_k)\|_2 \\ & + \tfrac{1}{2}\mu_k^2 L_1\left(n\|\nabla f(x_k)\|_2^2 + C_0^2 n^2\delta_k^4 + 2C_0 n^2\delta_k^2\|\nabla f(x)\|_2\right) \\ = & f(x_k) - \tfrac{1}{2nL_1}\|\nabla f(x_k)\|_2^2 + \tfrac{1}{2L_1}C_0^2\delta_k^4 + \tfrac{2}{L_1}C_0\delta_k^2\|\nabla f(x_k)\|_2,\end{aligned}$$

where the equality holds because the stepsize is constant and equal to $\mu_k = 1/(nL_1)$. By taking unconditional expectations, applying Jensen's inequality and rearranging terms, we then find

$$\begin{aligned}\mathbb{E}[\|\nabla f(x_k)\|_2]^2 &\leq \mathbb{E}[\|\nabla f(x_k)\|_2^2] \\ &\leq 2nL_1\mathbb{E}[f(x_k) - f(x_{k+1})] + 4nC_0\delta_k^2\mathbb{E}\left[\|\nabla f(x_k)\|_2\right] + nC_0^2\delta_k^4.\end{aligned} \tag{6.1}$$

Next, choose any $k' \in \mathbb{Z}_{>0}$ and sum the left- and rightmost terms in (6.1) over all $k \leq k'$ to obtain

$$\begin{aligned}&\mathbb{E}[\|\nabla f(x_{k'})\|_2]^2 \\ &\leq \textstyle\sum_{k=1}^{k'}\mathbb{E}[\|\nabla f(x_k)\|_2]^2 \\ &\leq 2nL_1\mathbb{E}[f(x_1) - f(x_{k'+1})] + 4nC_0\textstyle\sum_{k=1}^{k'}\delta_k^2\mathbb{E}[\|\nabla f(x_k)\|_2] + nC_0^2\sum_{k=1}^{k'}\delta_k^4 \\ &\leq 2nL_1F + 4nC_0\textstyle\sum_{k=1}^{k'}\delta_k^2\mathbb{E}[\|\nabla f(x_k)\|_2] + nC_0^2\sum_{k=1}^{k'}\delta_k^4,\end{aligned}$$

where the third inequality holds because $x^\star$ is a global minimizer of problem (1.1), which implies $\mathbb{E}[f(x_1) - f(x_{k'+1})] = \mathbb{E}[f(x_1) - f(x^\star)] + \mathbb{E}[f(x^\star) - f(x_{k'+1})] \leq F$. Setting $t_k = \mathbb{E}[\|\nabla f(x_k)\|_2]$ and $\nu_k = 4nC_0\delta_k^2$ for all $k \in \mathbb{Z}_{>0}$, and defining $T_{k'} = 2nL_1F + nC_0^2\sum_{k=1}^{k'}\delta_k^4$ for all $k' \in \mathbb{Z}_{>0}$, we may then use Lemma A.1, which applies because $\|\nabla f(x_1)\|_2^2 \leq 2nL_1F$, to find

$$\mathbb{E}[\|\nabla f(x_{k'})\|_2] \leq 2nC_0\textstyle\sum_{k=1}^{k'}\delta_k^2 + \left(2nL_1F + nC_0^2\sum_{k=1}^{k'}\delta_k^4 + (2nC_0\sum_{k=1}^{k'}\delta_k^2)^2\right)^{\frac{1}{2}}.$$

As $\delta_k = \delta/k$ for all $k \in \mathbb{Z}_{>0}$, the standard zeta function inequalities (A.1) imply that

$$\begin{aligned}\mathbb{E}[\|\nabla f(x_{k'})\|_2] &\leq nC_0\delta^2\tfrac{\pi^2}{3} + \left(2nL_1F + nC_0^2\delta^4\tfrac{\pi^4}{90} + n^2C_0^2\delta^4\tfrac{\pi^4}{9}\right)^{\frac{1}{2}} \\ &\leq \sqrt{2nL_1F} + nC_0\delta^2\left(\tfrac{2\pi^2}{3} + \tfrac{\pi^2}{\sqrt{90}}\right),\end{aligned}$$

where the second inequality holds because $\sqrt{a+b+c} \leq \sqrt{a} + \sqrt{b} + \sqrt{c}$ for all $a, b, c \geq 0$ and because $\sqrt{n} \leq n$ for all $n \in \mathbb{Z}_{\geq 0}$. Averaging the second inequality in (6.1) across all $k \leq K$ and using the above upper bound on $\mathbb{E}[\|\nabla f(x_k)\|_2]$ for each $k \leq K$ finally yields

$$\begin{aligned}\tfrac{1}{K}\textstyle\sum_{k=1}^{K}\mathbb{E}\left[\|\nabla f(x_k)\|_2^2\right] \leq & \tfrac{n}{K}\Big[2L_1F + 4C_0\textstyle\sum_{k=1}^{K}\delta_k^2\left(\sqrt{2nL_1F} + nC_0\delta^2\left(\tfrac{2\pi^2}{3} + \tfrac{\pi^2}{\sqrt{90}}\right)\right) \\ & + C_0^2\textstyle\sum_{k=1}^{K}\delta_k^4\Big].\end{aligned}$$

Applying the zeta function inequalities (A.1) once again and recalling that $\delta_k^2 \leq 1$ for all $k \in \mathbb{Z}_{>0}$, it is then easy to construct a constant $C \geq 0$ such that $\frac{1}{K}\sum_{k=1}^{K} \mathbb{E}[\|\nabla f(x_k)\|_2^2] \leq \frac{n}{K}(2L_1F + \delta^2 C)$. □

By Theorem 6.1, we can enforce $\frac{1}{K}\sum_{k=1}^{K} \mathbb{E}[\|\nabla f(x_k)\|_2^2] \leq \epsilon$ for a given $\epsilon > 0$ by selecting a smoothing parameter $\delta \leq O(\sqrt{K\epsilon/n})$ and by running Algorithm 1 over $O(nL_1F/\epsilon)$ iterations.

7 Numerical experiments

We will now assess the empirical performance of different variants of Algorithm 1 equipped with different gradient estimators on standard test problems. Specifically, we will compare the proposed complex-step estimator g_{cs} defined in (3.4) against the forward-difference estimator

$$g_{\mathsf{fd}}(x,\delta) = \tfrac{1}{\delta}(f(x+\delta y) - f(x))y \quad \text{with} \quad y \sim \mathcal{N}(0, I_n)$$

and the central-difference estimator

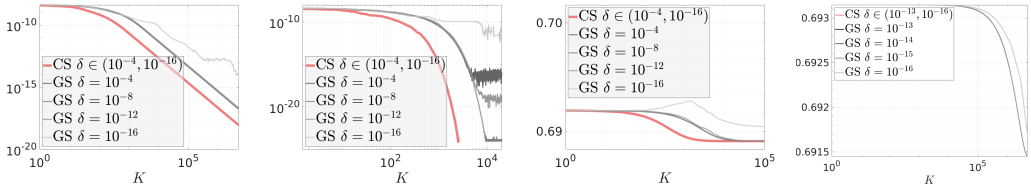
$$g_{\mathsf{cd}}(x,\delta) = \tfrac{1}{2\delta}(f(x+\delta y) - f(x-\delta y))y \quad \text{with} \quad y \sim \mathcal{N}(0, I_n),$$

both of which rely on Gaussian smoothing [NS17, Eq. (30)]; see also Example 2.8. As pointed out in the introduction, the single-point estimator (1.3) displays a higher variance and thus leads to slow convergence. Therefore, we exclude it from the numerical experiments. When using g_{fd} or g_{cd}, we set the stepsize of Algorithm 1 to $\mu_k = 1/(4(n+4)L_1)$ as recommended in [NS17, Eq. (55)]. When using g_{cs}, on the other hand, we select the stepsize in view of the structural properties of the given objective function f in accordance with Theorems 4.1, 5.1 and 6.1. The initial iterate x_1 is always set to 0 unless stated otherwise. All experiments are performed in MATLAB on a x86_64 machine with a 4 GHz CPU and 16 GB RAM, using double precision, that is, machine precision is $2^{-52} \approx 2.2204 \cdot 10^{-16}$.

From Sections 4–6 we know that Algorithm 1 with g_{cs} is guaranteed to find stationary points of a wide range of convex and non-convex optimization problems provided that its stepsize is inversely proportional to the Lipschitz constant L_1 of the gradient of f. Implementing Algorithm 1 in practice thus requires knowledge of L_1. Unfortunately, the Lipschitz modulus of f is typically unknown in the context of zeroth-order optimization, and the results of Sections 4–6 indicate that increasing L_1 increases the number of iterations and decreases the smoothing parameter δ needed to attain a desired suboptimality gap ϵ. These insights are consistent with classical results in zeroth-order optimization based on multi-point gradient estimators such as g_{fd} or g_{cd} (*cf.* [NS17]). As the complex-step method proposed in this paper remains numerically stable for almost arbitrarily small smoothing parameters δ, it may thus be preferable to classical methods when L_1 is overestimated.

The experiments will show that if δ is sufficiently large for multi-point methods to be applicable, then our complex-step method converges equally fast or faster than the multi-point methods, which obey the theoretical convergence rates reported in [NS17]. A theoretical explanation for the better empirical *transient* convergence behavior of the complex-step method is left for future work.

7.1 Unconstrained convex optimization



(a) Suboptimality gap $f(\bar{x}_K) - f^\star$ for (7.1).

(b) Suboptimality gap $f(x_K) - f^\star$ for (7.1).

(c) Cost $f(\bar{x}_K)$ for the log loss function (7.2).

(d) Cost $f(x_K)$ for the log loss function (7.2).

Figure 7.1: Comparison of the single-point complex-step estimator g_{cs} (CS) against the multi-point Gaussian smoothing estimator g_{fd} (GS) on two objective functions.

Example 7.1 (Quadratic test function)**.** *Assume that $\mathcal{X} = \mathbb{R}^n$ and f is an ill-conditioned version of what Nesterov calls the 'worst function in the world' [Nes03, § 2.1.2], that is, assume that*

$$f(x) = L\left(\tfrac{1}{2}\left[(x^{(1)})^2 + \textstyle\sum_{j=1}^{n-1}(x^{(j+1)} - x^{(j)})^2 + (x^{(n)})^2\right] - x^{(1)}\right), \tag{7.1}$$

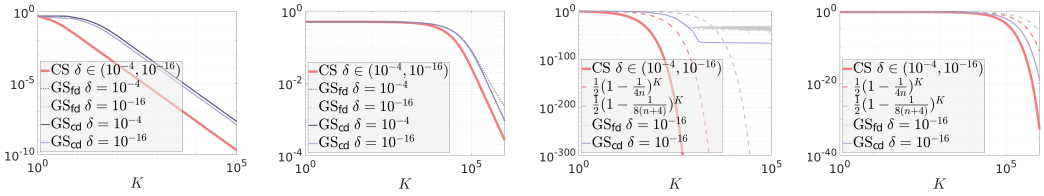
where $n = 5$, $L = 10^{-8}$, and $x^{(j)}$ denotes the j^{th} component of x for any $j \leq n$. One can show that ∇f has Lipschitz modulus $L_1 = 4L$ and that the unique global minimizer $x^\star$ of f has coordinates $(x^\star)^{(j)} = 1 - j/(n+1)$. In this case, the theoretical convergence guarantees of Algorithm 1 are independent of whether g_{cs} or g_{fd} is used. However, starting from $x_1 = 0$ and gradually reducing the smoothing parameter δ towards machine precision exposes the advantages of the one-point estimator g_{cs} over the multi-point estimator g_{fd}. Figures 7.1a and 7.1b visualize the suboptimality gap of $\bar{x}_K$ and x_K as a function of K along a single sample trajectory, respectively. Note that especially the performance of x_K is significantly better when g_{cs} is used. One might argue that g_{fd} leads to a higher suboptimality gap than g_{cs} because of its inferior approximation quality; see, e.g., Figure 2.1. We shed more light on this conjecture in Example 7.3, where we compare g_{cs} against g_{cd}.

Example 7.2 (Logistic regression)**.** *Assume that $\mathcal{X} = \mathbb{R}^n$ and f is the log loss function used to quantify the prediction loss in logistic regression. Specifically, set*

$$f(x) = \tfrac{1}{m}\textstyle\sum_{i=1}^{m} \log\left(1 + \exp\left(-v_i a_i^{\mathsf{T}} x\right)\right) \tag{7.2}$$

for $m = 100$ and $n = 2$, and assume that the features a_i and the labels v_i are sampled independently from the standard normal distribution on $\mathbb{R}^n$ and the uniform distribution on $\{-1, 1\}$, respectively. Denoting by $A \in \mathbb{R}^{m \times n}$ the matrix with rows a_i^{T} for all $i \leq m$, one readily verifies that $L_1 = \frac{1}{m}\|A\|_2$. We compare again the empirical convergence properties of Algorithm 1 equipped with g_{cs} or g_{fd}. Figures 7.1c and 7.1d visualize the objective function values of $\bar{x}_K$ and x_K as a function of K. We observe that the cancellation effects in the cost of $\bar{x}_K$ are mild even if g_{fd} is used and δ is small, whereas those in the cost of x_K are significantly more pronounced.

Example 7.3 (Dimension-dependence and strong convexity)**.** *Figure 7.1 not only confirms that the complex-step estimator g_{cs} is less susceptible to cancellation effects than the forward-difference estimator g_{fd}, but it also suggests that Algorithm 1 converges faster if g_{cs} is used instead of g_{fd}. In view of Example 2.8, we further expect that the central-difference estimator g_{cd} should lead to faster convergence than g_{fd}. To verify these conjectures numerically,*



(a) Suboptimality gap $f(\bar{x}_K) - f^\star$ for $n = 1$. **(b)** Suboptimality gap $f(\bar{x}_K) - f^\star$ for $n = 10^4$. **(c)** Suboptimality gap $f(x_K) - f^\star$ for $n = 1$. **(d)** Suboptimality gap $f(x_K) - f^\star$ for $n = 10^4$.

Figure 7.2: Comparison of the single-point complex-step estimator g_{cs} (CS) against multi-point Gaussian smoothing estimators g_{fd} ($\mathrm{GS_{fd}}$) and g_{cd} ($\mathrm{GS_{cd}}$) on $f(x) = \frac{1}{2}\|x\|_2^2$.

we now set $\mathcal{X} = \mathbb{R}^n$ and $f(x) = \frac{1}{2}\|x\|_2^2$. Figure 7.2 visualizes the suboptimality gap of $\bar{x}_K$ and x_K as a function of K for the three gradient estimators g_{cs}, g_{fd} and g_{cd} and for increasing dimensions $n \in \{1, 10{,}000\}$, starting from $x_1 = n^{-1/2}\mathbb{1}$. Cancellation effects prevail even in this simple example, yet they are mitigated in higher dimensions. We observe that $\bar{x}_K$ is less susceptible to cancellation effects but converges significantly slower than x_K. Even though the central-difference estimator g_{cd} does indeed provide a speed-up compared to the finite difference estimator g_{fd}, it is still dominated by the complex-step estimator g_{cs}. As f has a Lipschitz continuous gradient with $L_1 = 1$ and is strictly convex with $\tau = 1$, Theorem 5.1 ensures that for a negligible smoothing parameter δ and for an initial iterate with $\|x_1\|_2 = 1$ the suboptimality gap of x_K decays at least as fast as $\frac{1}{2}(1 - \frac{1}{4n})^K$. Figure 7.2 also visualizes this theoretical convergence rate and contrasts it with the rate $\frac{1}{2}(1 - \frac{1}{8(n+4)})^K$ for Algorithm 1 with g_{cd} [NS17, Eq. (57)].

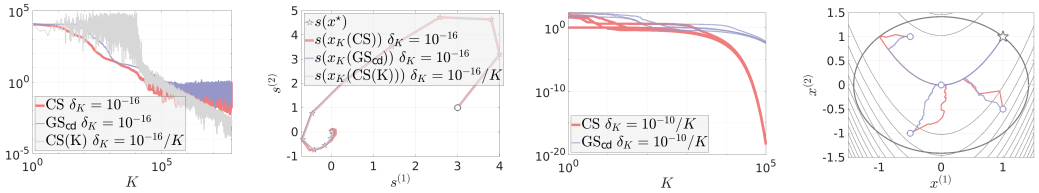
7.2 Constrained convex optimization The next example revolves around a constrained optimization problem grounded in control theory. We remark that the (unconstrained) infinite-horizon version of this problem could be addressed with the policy iteration scheme proposed in [Faz+18; Mal+19].

Example 7.4 (Policy iteration)**.** *We now address the MPC problem*

$$\begin{aligned} \underset{x=\{x_t\}_{t=0}^{T-1} \subseteq \mathbb{R}^{n_x}}{\text{minimize}} \quad & \textstyle\sum_{t=0}^{T-1} \langle Q s_t, s_t \rangle + \langle R x_t, x_t \rangle + \langle Q s_T, s_T \rangle \\ \text{subject to} \quad & s_{t+1} = A s_t + B x_t \quad \forall t = 0, \dots, T-1 \\ & \|x_t\|_\infty \leq 1 \quad \forall t = 0, \dots, T-1 \end{aligned} \tag{7.3}$$

with planning horizon $T \in \mathbb{Z}_{\geq}$ and initial state $s_0 \in \mathbb{R}^{n_s}$. Note that the dynamic constraints in (7.3) can be used to express the state trajectory $s = \{s_t\}_{t=1}^T$ as a linear function of the n-dimensional input trajectory $x = \{x_t\}_{t=0}^{T-1}$ with $n = T n_x$. We can thus eliminate s and express the objective function of (7.3) as a quadratic function $f(x)$ of the inputs x alone. Similarly, we can identify the feasible set of (7.3) with the compact hypercube $\mathcal{X} = \{x \in \mathbb{R}^n : \|x\|_\infty \leq 1\}$. Hence, the MPC problem (7.3) constitutes an instance of (1.1). We further assume that the cost matrices $Q \succeq 0$ and $R \succ 0$ are known, that the system matrices A and B are unknown, and that the costs of a given input trajectory x can be evaluated by simulation. This implies that f is unknown but admits a zeroth-order oracle. Throughout this experiment we set (A, B, Q, R) to the standard two-dimensional MPC instance[2] in Yalmip [Löf04], and

[2] https://yalmip.github.io/example/standardmpc/



(a) Suboptimality gap $f(x_K) - f^\star$ for Example 7.4.

(b) State trajectories generated by the control policies.

(c) Suboptimality gap $f(x_K) - f^\star$ for Example 7.5.

(d) Paths of iterates starting at four different initial points.

Figure 7.3: Comparison of the single-point complex-step estimator g_{cs} (CS for $\mu_k = \mu$ and CS(K) for $\mu_k = 1/k$) against the multi-point Gaussian smoothing estimator g_{cd} (GS$_{\mathsf{cd}}$) on the optimization problems of Examples 7.4 and 7.5.

we set $T = 15$ and $s_0 = (3, 1)$. We emphasize that the optimal solution of (1.1) may reside on the boundary of $\mathcal{X}$, and thus the theoretical guarantees of Sections 4 and 5 do not apply. Nevertheless, we will show that Algorithm 1 performs significantly better when the complex-step estimator g_{cs} is used instead of the central-difference estimator g_{cd}. We initialize the algorithm at the origin and upper bound the Lipschitz modulus of ∇f by $L_1 = 4 \cdot 10^4$. This crude bound is merely based on the operator norms of A and B. Figure 7.3a visualizes the suboptimality gap of x_K as a function of K. The oscillations in the suboptimality gap corresponding to g_{cs} emerge because the optimizer $x^\star$ of (7.3) resides on the boundary of $\mathcal{X}$. Note that Theorems 4.1 and 5.1 do not apply even though f is strongly convex. The reason is again that $\nabla f(x^\star) \neq 0$. Convergence results for optimization problems with boundary solutions have recently been obtained in [Jon21, Thm. 4.1] by allowing the stepsize μ_k to decay with k. These results even hold if we have only access to noisy function evaluations. We thus solve problem (7.3) once again with Algorithm 1 and the complex-step estimator g_{cs} but set $\mu_k = 1/k$ instead of $\mu_k = 1/(2nL_1)$ and $\delta_k = \delta = 10^{-16}$ instead of $\delta_k = \delta/k$. In this case, [Jon21, Thm. 4.1] guarantees the suboptimality gap to decay as $O(1/K)$. Figure 7.3a empirically validates this theoretical result. Note also that initial convergence is slower under a harmonically decaying stepsize. Figure 7.3b shows the state trajectories corresponding to differently computed inputs x_K for $K = 10^5$.

7.3 Non-convex optimization We finally apply our method to a classical non-convex test problem.

Example 7.5 (Rosenbrock function)**.** *Set $\mathcal{X} = \sqrt{2}\mathbb{B}^2$, and let f be the Rosenbrock function defined through $f(x) = (1 - x^{(1)})^2 + 100[x^{(2)} - (x^{(1)})^2]^2$. Then, problem (1.1) is uniquely solved by $x^\star = (1, 1)$, which coincides with the global minimizer of f over $\mathbb{R}^2$. We compare the complex-step estimator g_{cs} against g_{cd} but remark that the convergence behavior of Algorithm 1 does not change noticeably when g_{fd} is replaced with g_{cd}. We also set x_1 to one of four different points in $\mathcal{X}$ as visualized in Figure 7.3d. Figures 7.3c and 7.3d show the convergence of the suboptimality gap of x_K and the paths of iterates generated by Algorithm 1, respectively. Again, the complex-step estimator g_{cs} leads to significantly faster convergence. An additional acceleration can be achieved by decreasing δ below 10^{-10}. In this case, however, the Gaussian smoothing method breaks down.*

7.4 Outlook To close the paper, we discuss potential applications of our methods in the context of simulation-based optimization, where evaluating the objective function f requires the solution of an ordinary differential equation (ODE) or a partial differential equation (PDE). This section is illustrative only, and additional work is required to derive rigorous convergence guarantees. We start with an optimization problem involving an ODE, which—due to its chaotic nature—often serves as a benchmark problem in the dynamical systems literature; see, *e.g.*, [KBK22].

Example 7.6 (Lorenz system)**.** *The ODE*

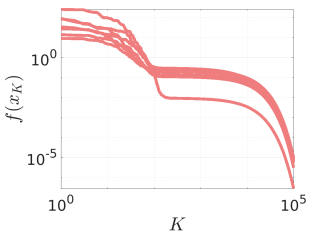
$$\frac{\mathrm{d}}{\mathrm{d}t}\begin{pmatrix}\ell_1(t)\\ \ell_2(t)\\ \ell_3(t)\end{pmatrix} = \begin{pmatrix}\sigma(\ell_2(t)-\ell_1(t))\\ \ell_1(t)(r-\ell_3(t))-\ell_2(t)\\ \ell_1(t)\ell_2(t)-b\ell_3(t)\end{pmatrix} \tag{7.4}$$

is commonly known as the Lorenz system [Str18, Ch. 9]. It was developed as a stylized model of atmospheric convection, with ℓ_1, ℓ_2 and ℓ_3 representing the rate of convection, the horizontal temperature variation and the vertical temperature variation, respectively. However, the Lorenz system also arises in the study of chemical reactions, population dynamics or electric circuits etc. In the following we denote by $\varphi^t(x)$ the time-t state of a Lorenz system with initial state $\ell(0) = x$. Given a potentially noisy measurement p of the state at time $t \geq 0$, a problem of practical interest is to estimate the initial state x that led to p. If x is known to belong to a closed set $\mathcal{X} \subseteq \mathbb{R}^3$, then it can conveniently be estimated by solving an instance of problem (1.1) with objective function $f(x) = \|p - \varphi^t(x)\|_2^2$ and feasible set $\mathcal{X}$. We expect this problem to be challenging because the Lorenz system is known to be chaotic. Thus, slight changes in the initial state have a dramatic impact on the future trajectory. Moreover, the objective function is not available in closed form but must be evaluated with a numerical ODE solver. As all commonly used ODE solvers map the initial state x to an approximation of $\varphi^t(x)$ by recursively applying analytic (in fact, polynomial) transformations, the resulting instance of problem (1.1) can be addressed with Algorithm 1. We remark that most out-of-the-box ODE solvers accept complex-valued initial conditions. Here we use MATLAB's `ode45` *routine.*

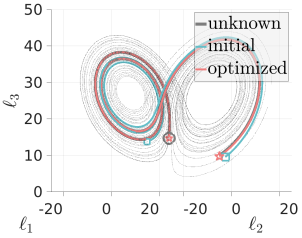
In the following we set the problem parameters to $\sigma = 10$, $r = 28$ and $b = 8/3$. In addition, we define $t = 2$ and $\mathcal{X} = \{x \in \mathbb{R}^3 : \|x - \ell(0)\|_2 \leq 2\}$, and we sample p from the normal distribution $\mathcal{N}(\varphi^2(\ell(0)), \epsilon^2 I_3)$, where $\ell(0) = (10, 10, 10)$ and $\epsilon = 10^{-3}$. Finally, we sample the initial iterate x_1 from the uniform distribution on the boundary of $\mathcal{X}$, use $L_1 = 1{,}000$ as a conservative estimate for the Lipschitz modulus of ∇f and set the smoothing parameter to $\delta = 10^{-10}$. By Theorem 6.1, Algorithm 1 converges to a stationary point $x^\star$ of the objective function with $\nabla f(x^\star) = 0$ provided that $\|\nabla f(x_1)\|_2$ is sufficiently small. See the discussion below Example 7.4 for the case $\nabla f(x^\star) \neq 0$. Figure 7.4a shows the decay of $f(x_K)$ with the total number K of iterations for 10 independent simulation runs. Figure 7.4b visualizes the corresponding state trajectories[3] $\{\varphi^t(x_K)\}_{t\in[0,2]}$ for $K = 10^5$. Only 6 of the 10 trajectories are shown for better visibility.

To close this section, we highlight that the complex-step method offers distinct benefits in the context of simulation-based optimization, where the objective function f can only be evaluated within prescribed error tolerances. In Example 7.6, for instance, the evaluation of f is corrupted by ODE integration errors. Unfortunately, such errors can have

[3] A video of the state evolution is available from https://wjongeneel.nl/Lorenz.gif.



(a) Decay of $f(x_K)$ with the total number K of iterations for 10 independent simulation runs.



(b) Trajectories of (7.4) starting from $\ell(0)$ (unknown), x_0 (initial) and x_K for $K = 10^5$ (optimized).

Figure 7.4: Estimating the initial state $\ell(0)$ of a Lorenz system (7.4) from a noisy measurement p of the state $\ell(2) = \varphi^2(\ell(0))$ (grey circle in 7.4b) at time 2. Even though the initial estimate x_0 is close to the optimized estimate x_K, $\varphi^2(x_0)$ is far from $\varphi^2(\ell(0))$.

a detrimental impact on classical finite-difference-based optimization schemes. Indeed, the central-difference estimator $\frac{1}{2\delta_k}(f(x_k+\delta_k y_k)-f(x_k-\delta_k y_k))y_k$ for $\nabla f(x_k)$ is useless for optimization unless the numerical errors in the evaluation of f are significantly smaller than δ_k. As δ_k must decay to 0 as k grows, so must the numerical tolerances. Otherwise, the ODE integration errors would dominate, which could be seen as another manifestation of catastrophic cancellation. Inexact evaluations of f can conveniently be modeled as outputs of a *noisy* zeroth-order oracle. While this paper was under review, it has been shown that convergence guarantees for Algorithm 1 can be obtained even if the complex zeroth-order oracle is affected by independently and identically distributed noise and even if the sequence of smoothing parameters $\{\delta_k\}_{k\in\mathbb{Z}_{>0}}$ is chosen *independently* of the noise statistics [Jon21]. This provides strong evidence that the complex-step approach may be able to overcome the practical obstructions outlined above that plague classical finite-difference schemes in simulation-based optimization. As integration errors are arguably not purely random and serially independent, however, further research is needed.

We highlight that complex-step derivatives are routinely used in PDE-constrained optimization. For example, they are used in a recent airfoil optimization package[4] developed in 2021. The underlying algorithm relies on *sequential quadratic programming* [NW06, Ch. 18] and assumes that the complex-step derivative equals the gradient. In contrast, our analysis provides a rigorous treatment of approximation errors.

Additional applications of the complex-step derivative are discussed in [MH13, § 3.2].

8 Conclusions and future work

The cancellation effects that plague all multi-point gradient estimators tend to have a detrimental effect on the numerical stability and the convergence behavior of zeroth-order algorithms. These numerical problems can sometimes be mitigated by replacing the terminal iterate x_K with the averaged iterate $\bar{x}_K = \frac{1}{K}\sum_{k=1}^{K} x_k$, at the cost of slower convergence. The single-point complex-step gradient estimator thus provides an attractive alternative to the classical gradient estimators because it leads to provably fast and numerically stable algorithms. As pointed out in [AMH10], smoothness is not a necessary condition for the

[4] https://mdolab-cmplxfoil.readthedocs-hosted.com.

applicability of the complex-step approximation, which suggests that the analyticity assumption used in this paper can perhaps be relaxed. Other promising research directions would be to extend our convergence guarantees to the class of weakly convex functions and to investigate multi-batch as well as online settings.

A Appendix

The proofs of our convergence results rely on the following lemma borrowed from [SRB11].

Lemma A.1 ([SRB11, Lem. 1])**.** *If* $\{t_k\}_{k\in\mathbb{Z}_{>0}}$ *and* $\{\nu_k\}_{k\in\mathbb{Z}_{>0}}$ *are two sequence of non-negative real numbers, while* $\{T_K\}_{K\in\mathbb{Z}_{>0}}$ *is a non-decreasing sequence of real numbers with* $T_1 \geq t_1^2$ *such that* $t_K^2 \leq T_K + \sum_{k=1}^K \nu_k t_k \ \forall k \in \mathbb{Z}_{>0}$*, then we have*

$$t_K \leq \tfrac{1}{2}\textstyle\sum_{k=1}^K \nu_k + \left(T_K + (\tfrac{1}{2}\sum_{k=1}^K \nu_k)^2\right)^{\frac{1}{2}} \quad \forall k \in \mathbb{Z}_{>0}.$$

In addition, several proofs in the main text make use of the inequalities

$$\textstyle\sum_{j=1}^J j^{-2} \leq \zeta(2) = \frac{1}{6}\pi^2 \quad \text{and} \quad \sum_{j=1}^J j^{-4} \leq \zeta(4) = \frac{1}{90}\pi^4 \quad \forall J \in \mathbb{Z}_{>0}, \tag{A.1}$$

which are obtained by truncating the series that defines the Riemann zeta function.

Acknowledgements WJ and DK are supported by the Swiss National Science Foundation under the NCCR Automation, grant agreement 51NF40_180545. MCY is supported by the Hong Kong Research Grants Council under the grant 25302420. WJ wishes to thank Prof. Arkadi Nemirovski for the supplied reference material, Prof. Timm Faulwasser for the pointer to the imaginary trick and Roland Schwan for fruitful discussions.

Bibliography

[Abr+18] R. Abreu, Z. Su, J. Kamm, and J. Gao. "On the accuracy of the Complex-Step-Finite-Difference method". *Journal of Computational and Applied Mathematics* 340 (2018), pp. 390–403.

[ADX10] A. Agarwal, O. Dekel, and L. Xiao. "Optimal algorithms for online convex optimization with multi-point bandit feedback". *Conference on Learning Theory*. 2010, pp. 28–40.

[Aga+09] A. Agarwal, M. J. Wainwright, P. Bartlett, and P. Ravikumar. "Information-theoretic lower bounds on the oracle complexity of convex optimization". *Neural Information Processing Systems*. 2009, pp. 1–9.

[AH17] C. Audet and W. Hare. *Derivative-Free and Blackbox Optimization*. Springer, 2017.

[AMH10] A. Al-Mohy and N. Higham. "The complex step approximation to the Fréchet derivative of a matrix function". *Numerical Algorithms* 53 (2010), pp. 133–148.

[APT20] A. Akhavan, M. Pontil, and A. Tsybakov. "Exploiting higher order smoothness in derivative-free optimization and continuous bandits". *Neural Information Processing Systems*. 2020, pp. 9017–9027.

[ASM15] R. Abreu, D. Stich, and J. Morales. "The complex-step-finite-difference method". *Geophysical Journal International* 202.1 (2015), pp. 72–93.

[Bau17] A. Bauer. "Five stages of accepting constructive mathematics". *Bulletin of the American Mathematical Society* 54.3 (2017), pp. 481–498.

[BBN19] A. S. Berahas, R. H. Byrd, and J. Nocedal. "Derivative-free optimization of noisy functions via quasi-Newton methods". *SIAM Journal on Optimization* 29.2 (2019), pp. 965–993.

[BCS21] A. S. Berahas, L. Cao, and K. Scheinberg. "Global convergence rate analysis of a generic line search algorithm with noise". *SIAM Journal on Optimization* 31.2 (2021), pp. 1489–1518.

[Bel08] J. L. Bell. *A Primer of Infinitesimal Analysis*. Cambridge University Press, 2008.

[Ber+21] A. S. Berahas, L. Cao, K. Choromanski, and K. Scheinberg. "A theoretical and empirical comparison of gradient approximations in derivative-free optimization". *Foundations of Computational Mathematics* (2021), pp. 1–54.

[Ber92] M. Berz. "Automatic differentiation as nonarchimedean analysis". *Computer Arithmetic and Enclosure Methods*. Ed. by L. Atanassova and J. Herzberger. Elsevier Science Publishers, 1992, pp. 439–450.

[Bez+17] J. Bezanson, A. Edelman, S. Karpinski, and V. B. Shah. "Julia: A Fresh Approach to Numerical Computing". *SIAM Review* 59.1 (2017), pp. 65–98.

[BG22] K. Balasubramanian and S. Ghadimi. "Zeroth-order nonconvex stochastic optimization: Handling constraints, high dimensionality, and saddle points". *Foundations of Computational Mathematics* 22.1 (2022), pp. 35–76.

[BP16] F. Bach and V. Perchet. "Highly-smooth zero-th order online optimization". *Conference on Learning Theory*. 2016, pp. 257–283.

[Bra+18] J. Bradbury, R. Frostig, P. Hawkins, M. J. Johnson, C. Leary, D. Maclaurin, G. Necula, A. Paszke, J. VanderPlas, S. Wanderman-Milne, and Q. Zhang. *JAX: Composable transformations of Python+NumPy programs*. Version 0.3.13. 2018. URL: http://github.com/google/jax.

[Cai+22] H. Cai, D. Mckenzie, W. Yin, and Z. Zhang. "A one-bit, comparison-based gradient estimator". *Applied and Computational Harmonic Analysis* 60 (2022), pp. 242–266.

[CH04] M. G. Cox and P. M. Harris. *Numerical Analysis for Algorithm Design in Metrology*. Software Support for Metrology Best Practice Guide No. 11. National Physical Laboratory, Teddington, 2004.

[CSV09] A. R. Conn, K. Scheinberg, and L. N. Vicente. *Introduction to Derivative-Free Optimization*. SIAM, 2009.

[CTL22] X. Chen, Y. Tang, and N. Li. "Improved single-point zeroth-order optimization using high-pass and low-pass filters". *International Conference on Machine Learning*. 2022, pp. 3603–3620.

[d'A08] A. d'Aspremont. "Smooth optimization with approximate gradient". *SIAM Journal on Optimization* 19.3 (2008), pp. 1171–1183.

[DGN14] O. Devolder, F. Glineur, and Y. Nesterov. "First-order methods of smooth convex optimization with inexact oracle". *Mathematical Programming* 146.1 (2014), pp. 37–75.

[Duc+15] J. C. Duchi, M. I. Jordan, M. J. Wainwright, and A. Wibisono. "Optimal rates for zero-order convex optimization: The power of two function evaluations". *IEEE Transactions on Information Theory* 61.5 (2015), pp. 2788–2806.

[Ell09] C. M. Elliott. "Beautiful differentiation". *ACM Sigplan Notices* 44.9 (2009), pp. 191–202.

[Faz+18] M. Fazel, R. Ge, S. Kakade, and M. Mesbahi. "Global convergence of policy gradient methods for the linear quadratic regulator". *International Conference on Machine Learning*. 2018, pp. 1467–1476.

[FKM04] A. Flaxman, A. T. Kalai, and H. B. McMahan. "Online convex optimization in the bandit setting: Gradient descent without a gradient". *arXiv preprint arXiv:0408007* (2004).

[Gas+17] A. V. Gasnikov, E. A. Krymova, A. A. Lagunovskaya, I. N. Usmanova, and F. A. Fedorenko. "Stochastic online optimization. Single-point and multi-point non-linear multi-armed bandits. Convex and strongly-convex case". *Automation and Remote Control* 78.2 (2017), pp. 224–234.

[GL13] S. Ghadimi and G. Lan. "Stochastic first-and zeroth-order methods for nonconvex stochastic programming". *SIAM Journal on Optimization* 23.4 (2013), pp. 2341–2368.

[GM78] J. D. Gray and S. A. Morris. "When is a function that satisfies the Cauchy-Riemann equations analytic?" *The American Mathematical Monthly* 85.4 (1978), pp. 246–256.

[Gol+20] D. Golovin, J. Karro, G. Kochanski, C. Lee, X. Song, and Q. Zhang. "Gradientless Descent: High-Dimensional Zeroth-Order Optimization". *International Conference on Learning Representations*. 2020.

[GW08] A. Griewank and A. Walther. *Evaluating derivatives: Principles and techniques of algorithmic differentiation*. SIAM, 2008.

[GWX17] Y. Gao, Y. Wu, and J. Xia. "Automatic differentiation based discrete adjoint method for aerodynamic design optimization on unstructured meshes". *Chinese Journal of Aeronautics* 30.2 (2017), pp. 611–627.

[HL14] E. Hazan and K. Levy. "Bandit convex optimization: Towards tight bounds". *Neural Information Processing Systems*. 2014, pp. 784–792.

[HS23] W. Hare and K. Srivastava. "A numerical study of applying complex-step gradient and Hessian approximations in blackbox optimization". *Pacific Journal of Optimization* 19.3 (2023), pp. 391–410.

[Hu+16] X. Hu, L. Prashanth, A. György, and C. Szepesvari. "(Bandit) convex optimization with biased noisy gradient oracles". *Artificial Intelligence and Statistics*. 2016, pp. 819–828.

[Hüc+23] J. Hückelheim, H. Menon, W. Moses, B. Christianson, P. Hovland, and L. Hascoët. "Understanding Automatic Differentiation Pitfalls". *arXiv preprint arXiv:2305.07546* (2023).

[Inn18] M. Innes. "Don't unroll adjoint: Differentiating SSA-form programs". *arXiv preprint arXiv:1810.07951* (2018).

[Ji+19] K. Ji, Z. Wang, Y. Zhou, and Y. Liang. "Improved zeroth-order variance reduced algorithms and analysis for nonconvex optimization". *International Conference on Machine Learning*. 2019, pp. 3100–3109.

[JNR12] K. G. Jamieson, R. Nowak, and B. Recht. "Query complexity of derivative-free optimization". *Advances in Neural Information Processing Systems*. 2012, pp. 2681–2689.

[Jon21] W. Jongeneel. "Imaginary Zeroth-Order Optimization". *arXiv preprint arXiv:2112.07488* (2021).

[KBK22] K. Kaheman, S. L. Brunton, and J. N. Kutz. "Automatic differentiation to simultaneously identify nonlinear dynamics and extract noise probability distributions from data". *Machine Learning: Science and Technology* 3.1 (2022), p. 015031.

[KP02] S. G. Krantz and H. R. Parks. *A Primer of Real Analytic Functions*. Birkhäuser, 2002.

[Kra00] S. G. Krantz. *Function Theory of Several Complex Variables*. AMS Chelsea Publishing, 2000.

[KW52] J. Kiefer and J. Wolfowitz. "Stochastic estimation of the maximum of a regression function". *The Annals of Mathematical Statistics* 23.3 (1952), pp. 462–466.

[KY03] H. Kushner and G. G. Yin. *Stochastic Approximation and Recursive Algorithms and Applications*. Springer, 2003.

[LBM22] J. Li, K. Balasubramanian, and S. Ma. "Stochastic zeroth-order Riemannian derivative estimation and optimization". *Mathematics of Operations Research* 48.2 (2022), pp. 1183–1211.

[Leb20] J. Lebl. "Tasty Bits of Several Complex Variables". https://www.jirka.org/scv/scv.pdf. 2020.

[Lee13] J. M. Lee. *Introduction to Smooth Manifolds*. Springer, 2013.

[Lia+16] X. Lian, H. Zhang, C.-J. Hsieh, Y. Huang, and J. Liu. "A comprehensive linear speedup analysis for asynchronous stochastic parallel optimization from zeroth-order to first-order". *Neural Information Processing Systems*. 2016, pp. 3062–3070.

[Liu+20] S. Liu, P. Y. Chen, B. Kailkhura, G. Zhang, A. O. Hero III, and P. K. Varshney. "A primer on zeroth-order optimization in signal processing and machine learning: Principals, recent advances, and applications". *IEEE Signal Processing Magazine* 37.5 (2020), pp. 43–54.

[LLZ21] H. Lam, H. Li, and X. Zhang. "Minimax efficient finite-difference stochastic gradient estimators using black-box function evaluations". *Operations Research Letters* 49.1 (2021), pp. 40–47.

[LM67] J. N. Lyness and C. B. Moler. "Numerical differentiation of analytic functions". *SIAM Journal on Numerical Analysis* 4.2 (1967), pp. 202–210.

[LMW19] J. Larson, M. Menickelly, and S. M. Wild. "Derivative-free optimization methods". *Acta Numerica* 28 (2019), pp. 287–404.

[Löf04] J. Löfberg. "YALMIP: A toolbox for modeling and optimization in MATLAB". *IEEE International Conference on Robotics and Automation*. 2004, pp. 284–289.

[LRD12] G. Lantoine, R. P. Russell, and T. Dargent. "Using multicomplex variables for automatic computation of high-order derivatives". *ACM Transactions on Mathematical Software* 38.3 (2012), pp. 1–21.

[Mal+19] D. Malik, A. Pananjady, K. Bhatia, K. Khamaru, P. Bartlett, and M. Wainwright. "Derivative-free methods for policy optimization: Guarantees for linear quadratic systems". *Artificial Intelligence and Statistics*. 2019, pp. 2916–2925.

[MC20] W. Moses and V. Churavy. "Instead of Rewriting Foreign Code for Machine Learning, Automatically Synthesize Fast Gradients". *Advances in Neural Information Processing Systems*. 2020, pp. 12472–12485.

[MH13] J. R. Martins and J. T. Hwang. "Review and unification of methods for computing derivatives of multidisciplinary computational models". *AIAA Journal* 51.11 (2013), pp. 2582–2599.

[MSA01] J. Martins, P. Sturdza, and J. Alonso. "The connection between the complex-step derivative approximation and algorithmic differentiation". *39th Aerospace Sciences Meeting and Exhibit*. 2001, pp. 1–11.

[MSA03] J. R. R. A. Martins, P. Sturdza, and J. J. Alonso. "The complex-step derivative approximation". *ACM Transactions on Mathematical Software* 29.3 (2003), pp. 245–262.

[MW14] J. J. Moré and S. M. Wild. "Do you trust derivatives or differences?" *Journal of Computational Physics* 273 (2014), pp. 268–277.

[Nes03] Y. Nesterov. *Introductory Lectures on Convex Optimization: A Basic Course*. Springer, 2003.

[NG22] V. Novitskii and A. Gasnikov. "Improved exploitation of higher order smoothness in derivative-free optimization". *Optimization Letters* 16.7 (2022), pp. 2059–2071.

[NM65] J. A. Nelder and R. Mead. "A simplex method for function minimization". *The Computer Journal* 7.4 (1965), pp. 308–313.

[NS17] Y. Nesterov and V. Spokoiny. "Random gradient-free minimization of convex functions". *Foundations of Computational Mathematics* 17.2 (2017), pp. 527–566.

[NS18] F. Nikolovski and I. Stojkovska. "Complex-step derivative approximation in noisy environment". *Journal of Computational and Applied Mathematics* 327 (2018), pp. 64–78.

[NW06] J. Nocedal and S. J. Wright. *Numerical Optimization*. Springer, 2006.

[NY83] A. S. Nemirovsky and D. B. Yudin. *Problem Complexity and Method Efficiency in Optimization*. Wiley, 1983.

[Ove01] M. L. Overton. *Numerical Computing with IEEE Floating Point Arithmetic*. SIAM, 2001.

[PT90] B. T. Polyak and A. B. Tsybakov. "Optimal order of accuracy of search algorithms in stochastic optimization". *Problemy Peredachi Informatsii* 26.2 (1990), pp. 45–53.

[RLP16] J. Revels, M. Lubin, and T. Papamarkou. "Forward-mode automatic differentiation in Julia". *arXiv preprint arXiv:1607.07892* (2016).

[Rud87] W. Rudin. *Real and Complex Analysis*. McGraw-Hill Education, 1987.

[Sch22] K. Scheinberg. "Finite Difference Gradient Approximation: To Randomize or Not?" *INFORMS Journal on Computing* 34.5 (2022), pp. 2384–2388.

[Sha13] O. Shamir. "On the complexity of bandit and derivative-free stochastic convex optimization". *Conference on Learning Theory*. 2013, pp. 3–24.

[Sha17] O. Shamir. "An optimal algorithm for bandit and zero-order convex optimization with two-point feedback". *Journal of Machine Learning Research* 18.1 (2017), pp. 1703–1713.

[Shi+22] H.-J. M. Shi, M. Q. Xuan, F. Oztoprak, and J. Nocedal. "On the numerical performance of derivative-free optimization methods based on finite-difference approximations". *Optimization Methods and Software* 38.2 (2022), pp. 289–311.

[SMG13] S. U. Stich, C. L. Müller, and B. Gärtner. "Optimization of convex functions with random pursuit". *SIAM Journal on Optimization* 23.2 (2013), pp. 1284–1309.

[Spa05] J. C. Spall. *Introduction to Stochastic Search and Optimization: Estimation, Simulation, and Control*. John Wiley & Sons, 2005.

[Spa+92] J. C. Spall et al. "Multivariate stochastic approximation using a simultaneous perturbation gradient approximation". *IEEE Transactions on Automatic Control* 37.3 (1992), pp. 332–341.

[SRB11] M. Schmidt, N. Roux, and F. Bach. "Convergence rates of inexact proximal-gradient methods for convex optimization". *Neural Information Processing Systems*. 2011, pp. 1458–1466.

[ST98] W. Squire and G. Trapp. "Using complex variables to estimate derivatives of real functions". *SIAM Review* 40.1 (1998), pp. 110–112.

[Str18] S. H. Strogatz. *Nonlinear Dynamics and Chaos: With Applications to Physics, Biology, Chemistry, and Engineering*. CRC Press, 2018.

[SV15] Y. Singer and J. Vondrák. "Information-theoretic lower bounds for convex optimization with erroneous oracles". *Neural Information Processing Systems*. 2015, pp. 3204–3212.

[SW18] J. A. Snyman and D. N. Wilke. *Practical Mathematical Optimization: Basic Optimization Theory and Gradient-Based Algorithms*. Springer, 2018.

[Vui+23] C. Vuik, F. J. Vermolen, M. B. van Gijzen, and M. J. Vuik. *Numerical Methods for Ordinary Differential Equations*. TU Delft Open, 2023.

[WS21] L. Wang and J. C. Spall. "Improved SPSA using complex variables with applications in optimal control problems". *American Control Conference*. 2021, pp. 3519–3524.

[WZS21] L. Wang, J. Zhu, and J. C. Spall. "Model-free optimal control using SPSA with complex variables". *Conference on Information Sciences and Systems*. 2021, pp. 1–5.

[Zha+22] Y. Zhang, Y. Zhou, K. Ji, and M. M. Zavlanos. "A New One-Point Residual-Feedback Oracle for Black-Box Learning and Control". *Automatica* 136 (2022).