Worst-case convergence analysis of relatively inexact gradient descent on smooth convex functions

Pierre Vernimmen[a,1,*], François Glineur[a,b]

[a] *UCLouvain, ICTEAM (INMA), Avenue Georges Lemaître 4, Louvain-la-Neuve, 1348, Belgium*
[b] *UCLouvain, CORE, Voie du Roman Pays 34, Louvain-la-Neuve, 1348, Belgium*

Abstract

We consider the classical gradient descent algorithm with constant stepsizes, where some error is introduced in the computation of each gradient. More specifically, we assume some relative bound on the inexactness in the sense that the norm of the difference between the true gradient and its approximate value is bounded by a certain fraction of the gradient norm. This paper presents a worst-case convergence analysis of this so-called relatively inexact gradient descent on smooth convex functions, using the Performance Estimation Problem (PEP) framework. We first derive the exact worst-case behavior of the method after one step. Then we study the case of several steps and provide computable upper and lower bounds using the PEP framework. Finally, we discuss the optimal choice of constant stepsize according to the obtained worst-case convergence rates.

Keywords: Gradient Descent, Inexact Gradient, Smooth Convex Optimization, Performance Estimation, Robustness

*Corresponding author. Email: pierre.vernimmen@uclouvain.be
[1]The first author is funded by the UCLouvain *Fonds spéciaux de Recherche* (FSR).

Contents

1. Introduction

1.1. Problem Setup

We consider the standard optimization problem of unconstrained minimization of a differentiable function:

$$\min_{x \in \mathbb{R}^d} f(x),$$

where $f : \mathbb{R}^d \to \mathbb{R}$ is assumed to be convex and L-smooth, that is

$$f(y) \geq f(x) + \langle \nabla f(x), y - x \rangle, \quad \forall x, y \in \mathbb{R}^d,$$

and

$$\|\nabla f(x) - \nabla f(y)\| \leq L\|x - y\|, \quad \forall x, y \in \mathbb{R}^d$$

(i.e. its gradient is L-Lipschitz). We assume that f admits at least one minimizer, denoted by $x^\star \in \arg\min f$. The goal of this paper is to analyze

first-order methods when encountering inexactness, and in particular Inexact Gradient Descent, for solving this problem.

Remark. The step-size in gradient-based methods typically depends on the smoothness constant L. In many applications, this constant can be computed exactly (for example when the Hessian is available), while in others it can be estimated or bounded (see e.g. [1]). Throughout this paper, we assume that an admissible value of the smoothness constant L is available.

1.2. First-order methods, relative inexactness and worst-case analysis

First-order methods and their practical relevance. First-order optimization methods, and in particular gradient-based techniques, are widely used in modern large-scale optimization. Their low per-iteration complexity and favorable convergence guarantees make them especially well-suited to machine learning and signal processing applications [2]. Among these methods, the gradient descent with a constant stepsize stands out due to its simplicity, robustness, and effectiveness, in particular for smooth convex problems [3].

Motivations for inexact gradient computations. In many practical situations, exact gradient computations are not available or too costly. For example, inexact gradients naturally arise when computations are performed with limited precision, as in floating point arithmetic, or when the gradient is computed approximately, e.g., through sampling, simulations, or inner optimization loops. These sources of inexactness can significantly affect the performance of gradient-based methods and are thus central to both theoretical analysis and practical algorithm design.

Inexactness may also be imposed intentionally to save computational resources, for example, by quantizing gradients or using less expensive gradient surrogates. In large-scale machine learning systems or distributed environments, robustness to such approximations becomes crucial, and understanding their impact on convergence is essential for reliable algorithmic implementation. This is especially true in modern hardware environments, where low-precision computations are often preferred to accelerate training or inference (see e.g. the discussion in [4]).

Different models of inexactness. Several models of inexactness have been studied in the literature. A common approach assumes that the approximate gradient differs from the true one by an error with bounded norm, leading

to an *absolute* error model, as studied, for instance, in [5]. In [6], another notion of inexact gradient is developed, based on the maximal error incurred by the corresponding quadratic upper bound. In these frameworks, the norm of the additive error is independent of the magnitude of the gradient.

More recently, a *relative* model of inexactness has attracted attention, in which the norm of the gradient error is bounded proportionally to the true gradient norm. Mathematically, let d_k be the approximate gradient at x_k, it is then said to be a δ-relatively inexact gradient if $\|d_k - \nabla f(x_k)\| \leq \delta \|\nabla f(x_k)\|$, where $\delta \in [0, 1)$ is a parameter controlling inexactness. This model is particularly relevant when the gradient norm itself indicates proximity to optimality: more accurate gradients are naturally required near the solution. It also includes, as special cases, errors arising from quantization or low-precision computation, as studied in [7] in the context of mixed-integer linear programming. The relative error model thus often offers a more flexible and realistic description of inexactness in practical applications.

Previous work. This notion of relative inexactness is used in [8, 9] in the context of convergence analysis, where the Performance Estimation Problem (PEP) framework [10, 11] is used to provide tight convergence guarantees for the inexact gradient descent on L-smooth, μ-strongly convex functions. For example, [9, Theorem 5.3] establishes a tight linear convergence rate for gradient descent with constant stepsize h and relative inexactness $\delta < \frac{2\mu}{L+\mu}$. This analysis is however restricted to the strongly convex setting, and the results become ineffective as $\mu \to 0$, for the smooth convex case.

Subsequent works, such as [12], [13], [14], and [15] extend the study of relative inexactness to several first-order and accelerated methods using Lyapunov-based analyses. In particular, [12] shows that the relatively inexact gradient descent applied to smooth convex functions converges to a minimizer for sufficiently small constant stepsizes. While the approaches from the above papers cover more algorithmic variants, they typically yield conservative and sometimes loose upper bounds (both for rates and allowed stepsizes), which may limit their usefulness.

Other related research with relative errors includes probabilistic relative error models [16], momentum-based schemes interpreted as inexact gradient descent [17], proximal methods under relative gradient errors [18], robustness to noise in distributed settings [19], and heuristic stopping criteria under gradient noise [20]. Complementary to these, [21] analyzes error-feedback schemes with compressed gradients, providing tight worst-case guarantees.

1.3. Contributions

Our main contribution is to extend the PEP-based analysis of [9] beyond the strongly convex case to smooth convex functions, providing some tight convergence guarantees for relatively inexact gradient descent in this broader setting. This journal article is an extension of our conference paper [22], which initiated the study of relatively inexact gradient descent in the convex setting using the PEP methodology, but left the analysis incomplete. Numerical experiments are presented in a separate companion paper [4], by the same authors.

Relatively inexact gradient descent. More specifically, in this paper, we will provide an improved, and, in some cases, tight convergence analysis of the following relatively inexact gradient descent:

Algorithm 1 Relatively Inexact Gradient Descent with constant (normalized) stepsize h

1: Given an L-smooth convex function $f(\cdot)$, a starting iterate x_0, a stepsize h and an inexactness level $\delta \in [0, 1)$
2: $k \leftarrow 0$
3: **while** $k <$ max_iterations **do**
4: Let d_k be an approximate gradient at x_k with δ relative inexactness, i.e. such that

$$\|d_k - \nabla f(x_k)\| \leq \delta \|\nabla f(x_k)\| \tag{1}$$

5: Compute the next iterate as $x_{k+1} = x_k - \frac{h}{L} d_k$
6: $k \leftarrow k + 1$
7: **end while**

Our analysis demonstrates that Algorithm 1 enjoys guaranteed convergence towards a minimizer of the function for any level of relative inexactness $\delta \in [0, 1)$, provided the stepsize is well chosen. This result is significant, as it shows that gradient descent can be robust to large relative inexactness levels. Specifically, we show that the method converges for any stepsize $h \in [0, h_{\max}]$, with $h_{\max} = \frac{2}{1+\delta}$, and is no longer convergent for $h > h_{\max}$. This tight bound on acceptable stepsizes improves the previously known result from [12].

Type of convergence analysis. In the previously studied smooth strongly convex setting, linear convergence rates were established on the objective accuracy, the gradient norm, and the distance to the solution (see Theorems

5.1 and 5.4 in [9]). However, in the smooth convex case, it is well known that linear convergence cannot hold. Instead, one has to resort to studying convergence rates based on two distinct criteria, characterizing respectively the starting iterate and the last iterate. The perhaps most classical type of rate in the smooth convex setting bounds the objective accuracy in terms of the initial distance to the solution, see e.g. [3, Corollary 2.1.2].
However, in this paper, we will prove results of the following type:

$$\tfrac{1}{L}\|\nabla f(x_N)\|^2 \leq \sigma_N\big(f(x_0) - f(x_*)\big). \tag{2}$$

i.e., we bound the (squared) gradient norm in terms of the initial objective accuracy. Constant σ_N describes the convergence rate after N iterations. This type of rate in the gradient norm has also been studied in the exact case, see e.g. [23], and also potentially allows generalization to nonconvex settings [24].

Main theorem. In the exact case ($\delta = 0$), it is well-known that the convergence rate σ_N for smooth convex functions has order $\mathcal{O}(\frac{1}{N})$. More precisely, the following tight rate is proved in [24, Theorem 2.1] (see the beginning of Section 2 for a complete proof for one iteration):

Theorem 1.1 ([24], Theorem 2.1)**.** *Algorithm 1 applied to a convex L-smooth function f with a constant stepsize $h \in [0, 2]$ and an exact gradient ($\delta = 0$), started from iterate x_0, generates iterates satisfying*

$$\tfrac{1}{L}\|\nabla f(x_N)\|^2 \leq \max\left\{\frac{1}{Nh + \frac{1}{2}}, 2(1-h)^{2N}\right\}(f(x_0) - f(x_*))$$

The main theoretical result in this paper is Theorem 3.1, which extends the above to situations where the gradient is relatively inexact.

Theorem 3.1. *Algorithm 1 applied to a convex L-smooth function f with an inexact gradient with relative inexactness $\delta \in (0, 1)$, started from iterate x_0 with a constant stepsize $h \in [0, \frac{2}{1+\delta}]$, generates iterates satisfying*

$$\tfrac{1}{L}\min_{k\in\{1,\cdots,N\}}\|\nabla f(x_k)\|^2 \leq \tilde{C}_N(h,\delta)(f(x_0) - f(x_*)),$$

where $\tilde{C}_N(h,\delta)$ is given by the continuous function

$$\tilde{C}_N(h,\delta) = \begin{cases} \frac{1}{Nh(1-\delta)+\frac{1}{2}}, & \text{if } h \in \left[0, \frac{3}{2(1+\delta)}\right), \\ \frac{2\tilde{\lambda}}{N(h\tilde{\lambda}^2+2(h-1)\tilde{\lambda}+h-1)+\tilde{\lambda}}, & \text{if } h \in \left[\frac{3}{2(1+\delta)}, \frac{3\delta+2-\sqrt{4-3\delta^2}}{2\delta(\delta+1)}\right], \\ \frac{2}{N((1-h(1+\delta))^{-2})-(N-1)}, & \text{if } h \in \left(\frac{3\delta+2-\sqrt{4-3\delta^2}}{2\delta(\delta+1)}, \frac{2}{1+\delta}\right], \end{cases}$$

and $\tilde{\lambda}$ is the solution of a cubic equation (see Definition 2.2).

Convergence is hence governed by three distinct regimes depending on the stepsize, compared to only two regimes in the exact case. We also discuss situations where this rate is tight, which includes the case of a single step for any stepsize (see in Section 2, where both analytical and numerical arguments are used to establish tightness), and some subset of the three stepsize regimes for any number of iterations (see Section 3).

1.4. Performance Estimation

Formulation. The tight convergence rates described in this paper were obtained using the Performance Estimation Problem (PEP) methodology, first introduced in [10], which aims at computing the worst-case convergence rate of some optimization method as the solution of an optimization problem itself. More precisely, the convergence rate σ_N after N iterations of a given algorithm $\mathcal{A}$ over a given class of objective functions $\mathcal{F}$ can be shown to be equal to the optimal value of the following Performance Estimation Problem

$$\begin{aligned} \sigma_N = \max_{\{x_i,g_i,f_i\}_{i\in I}} \quad & \|g_N\|^2 \\ & x_1,\dots,x_N \text{ are generated from } x_0 \text{ by algorithm } \mathcal{A}, \\ & \text{there exists an interpolating function } f\in\mathcal{F} \text{ such} \\ & \text{that } f(x_i)=f_i \text{ and } \nabla f(x_i)=g_i \text{ for all } i\in I, \\ & g_*=0, \text{ and } f_0-f_*\leq 1. \end{aligned}$$

(with set $I=\{0,1,\cdots,N,*\}$), which can be formulated exactly and solved as a tractable semidefinite optimization problem, via the use of a Gram matrix lifting [11].

The condition that guarantees that the iterates, gradients, and function values contained in variables $\{x_i,g_i,f_i\}_{i\in I}$ match some interpolating function $f\in\mathcal{F}$ must be made explicit. For the class of L-smooth convex functions studied in this paper, it was shown in [11] that this is equivalent to requiring the following inequality, called *interpolation condition*

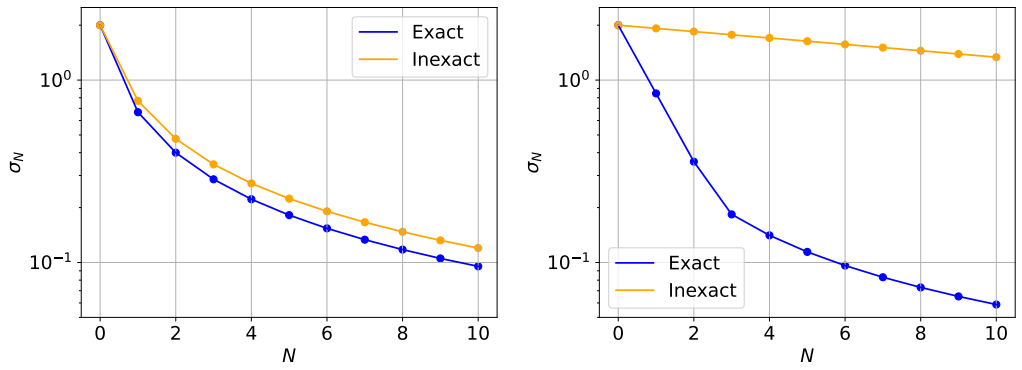
$$Q_{ij}: \quad f_i \geq f_j + \langle g_j, x_i - x_j\rangle + \tfrac{1}{2L}\|g_j-g_i\|^2 \qquad \text{(INT)}$$

to hold for every pair of indices $i,j\in I$.

To analyze algorithms that use relatively inexact gradients, such as Algorithm 1, additional variables d_i are introduced and the defining inequality (1) is added to the PEP, rewritten as the convex quadratic constraint $\|d_i-g_i\|^2\leq\delta^2\|g_i\|^2$ (see [25] for the idea of using inexact steps in a PEP).

The solution to the resulting semidefinite optimization problem can be obtained numerically using a standard solver such as MOSEK [26], used for all numerical results in this work. Several toolboxes are available to assist practitioners in deriving the formulation for a large variety of classes of algorithms and functions, such as PESTO [27] and PEPit [28].

Example. To illustrate the Performance Estimation technique and give an example of its relevance to analyze inexact methods, we plot in Figure 1.1 the tight numerical convergence rate obtained from solving the PEP for Algorithm 1 on 1-smooth convex functions, showing that the behavior of a first-order method can be very different when relying on an inexact gradient (here we test $\delta = 0.2$). On the left, the behavior of gradient descent with constant stepsize $h = 1$ is barely modified. On the right, the same gradient descent with constant stepsize $h = 1.65$, which improves the convergence rate in the exact case, is seen to deteriorate it significantly in the inexact case.



(a) $h = 1$ (b) $h = 1.65$

Figure 1.1: Convergence rate σ_N of exact and inexact ($\delta = 0.2$) gradient descent after N iterations.

Deriving proofs. Solving the PEP not only yields a tight worst-case performance guarantee, but also provides a way to derive a proof of those convergence bounds [11]. Those proofs are a byproduct of standard convex optimization duality. More concretely, all PEP convergence bounds can be obtained by summing the interpolation and inexactness inequalities arising in the formulation with suitable multipliers, which are given by optimal dual variables.

In this work, we use the PEP framework not only to compute tight worst-case rates numerically, but also as an exploration tool to gain insight about the structure of valid convergence proofs. In several situations, we can analyze the numerical optimal value and solution (both primal and dual) of the PEP for several values of N (number of iterations), h (stepsize), and δ (inexactness level), identify patterns, and conjecture candidate analytical expressions for them. These serve as a basis to derive rigorous mathematical proofs for lower and upper bounds on the convergence rates. This two-step approach, based on first gaining insights from numerical values and then producing a fully analytical argument, is classic within the PEP community, and allowed us to derive all convergence rates and proofs presented in this paper.

1.5. Paper organization

In Section 2 we derive a tight worst-case rate after one step, i.e., find an expression for σ_1 that cannot be improved, as it is exactly matched by some function f. An explicit analytical expression for such a function f is also provided for nearly all values of the stepsize.

Section 3 studies the case where several steps of the methods are performed, and provides computable upper and lower bounds using the PEP framework. While performing a tight analysis becomes intractable beyond a few steps due to the symbolic complexity of the resulting equations, the associated semidefinite programs can be explicitly constructed and solved for any number of iterations, allowing to obtain tight numerical performance bounds (albeit with a computational cost that grows with N).

We also provide principled recommendations for the optimal choice of a constant stepsize according to the worst-case rate: as opposed to the exact case analyzed in [11], the optimal stepsize does not seem to be very sensitive to the number of iterations. This reflects a key difference induced by the relative inexactness model and highlights the importance of adapting algorithmic parameters to account for inexactness. Section 4 presents some conclusions. All figures from this paper, as well as the associated code, are openly available on `https://github.com/Verpierre/Inexact_gradient_descent`.

2. One step of relatively inexact gradient descent

2.1. Convergence rate in the exact case ($\delta = 0$)

Before diving into the analysis of convergence rates in the relatively inexact setting, we first derive a comprehensive proof of one step of exact Gradient Descent ($N = 1$ in Theorem 1.1). This result is proved in [24], but we present an alternate proof here, whose structure will also be used in the inexact case.

Theorem 2.1 (Convergence rate of one step of exact gradient descent on smooth convex functions)**.** *One step of Algorithm 1 applied to a convex L-smooth function f with an exact gradient ($\delta = 0$), started from iterate x_0 with a constant stepsize $h \in [0, 2]$, generates an iterate x_1 satisfying*

$$\tfrac{1}{L}\|\nabla f(x_1)\|^2 \leq \max\left\{\frac{1}{h}, \frac{2}{(1-h)^{-2}-1}\right\}(f(x_0) - f(x_1))$$

and

$$\tfrac{1}{L}\|\nabla f(x_1)\|^2 \leq \max\left(\frac{1}{h+\frac{1}{2}}, 2(1-h)^2\right)(f(x_0) - f(x_*))$$

Proof. We assume without loss of generality that the smoothness constant is $L = 1$, because of a standard homogeneity argument [11, Section 3.5]. A single step of the method computes $x_1 = x_0 - hg_0$. Using interpolation inequalities (INT) derived from smoothness and convexity, we obtain

$$\begin{aligned} Q_{01} : f_0 - f_1 &\geq \frac{g_0^2}{2} + \frac{g_1^2}{2} + (h-1)g_0 g_1 \\ Q_{10} : f_1 - f_0 &\geq \frac{1-2h}{2} g_0^2 + \frac{g_1^2}{2} - g_0 g_1, \end{aligned}$$

where the definition of x_1 is used to express everything in terms of gradients.

We now form a nonnegative linear combination of these inequalities using the multipliers

$$\begin{aligned} \lambda_{01} &= \lambda + 1, \\ \lambda_{10} &= \lambda. \end{aligned}$$

where λ is a nonnegative parameter that will be chosen later. The combination $\lambda_{01}Q_{01} + \lambda_{10}Q_{10}$ yields a quadratic inequality of the form

$$f_0 - f_1 \geq \frac{1}{2}\begin{pmatrix} g_0 \\ g_1 \end{pmatrix}^\top \begin{pmatrix} 2(1-h)\lambda + 1 & (h-2)\lambda - 1 + h \\ (h-2)\lambda - 1 + h & 2\lambda + 1 \end{pmatrix}\begin{pmatrix} g_0 \\ g_1 \end{pmatrix}$$

(this specific parametrization with $\lambda \geq 0$ actually ensures that the left-hand side is equal to $f_0 - f_1$). We aim to lower-bound $f_0 - f_1$ by a multiple of $\|g_1\|^2$, i.e., obtain $f_0 - f_1 \geq \rho\|g_1\|^2$ for some value of ρ, as large as possible. To this end, we decompose the above matrix as

$$f_0 - f_1 \geq \frac{1}{2}\begin{pmatrix} g_0 \\ g_1 \end{pmatrix}\left[\underbrace{\begin{pmatrix} 2(1-h)\lambda + 1 & (h-2)\lambda - 1 + h \\ (h-2)\lambda - 1 + h & 2\lambda + 1 - 2\rho \end{pmatrix}}_{A} + \begin{pmatrix} 0 & 0 \\ 0 & 2\rho \end{pmatrix}\right]\begin{pmatrix} g_0 & g_1 \end{pmatrix}.$$

When matrix A above is positive semidefinite, inequality $f_0 - f_1 \geq \rho\|g_1\|^2$ follows. To find the best possible rate ρ, we aim to maximize it under the constraint that A is positive semidefinite.

Matrix A is positive semidefinite if and only if all its principal minors are non-negative. This yields the following optimization problem

$$\begin{aligned} \max_{\lambda,\rho \geq 0} \quad & \rho \\ \text{s.t.} \quad & A_{0,0} = 2(1-h)\lambda + 1 \geq 0 \\ & \det(A) = 2\rho(2(h-1)\lambda - 1) + h(-h(1+\lambda)^2 + 4\lambda + 2) \geq 0 \end{aligned}$$

which will, in principle, compute the best possible convergence rate. We now distinguish three cases based on the value of the stepsize h.

Case 1: $h = 1$**.** As $\lambda \geq 0$, the first constraint is trivially satisfied, and the second one simplifies to

$$2\rho \leq -\lambda^2 + 2\lambda + 1.$$

The right-hand side is a quadratic in λ, maximized for $\lambda_* = 1$, yielding

$$\rho_* = \frac{-1^2 + 2 \cdot 1 + 1}{2} = 1.$$

Case 2: $h < 1$**.** The first constraint becomes

$$\lambda \geq \frac{1}{2(h-1)}, \tag{3}$$

while the second one can be rewritten as

$$\rho \leq \frac{h(h(1+\lambda)^2 - 4\lambda - 2)}{2(2(h-1)\lambda - 1)}.$$

We define

$$g(\lambda) := \frac{h(h(1+\lambda)^2 - 4\lambda - 2)}{2(2(h-1)\lambda - 1)},$$

so that the second constraint is $\rho \leq g(\lambda)$. To maximize ρ, we thus maximize $g(\lambda)$ subject to the constraint (3). Computing the first-order derivative of $g(\lambda)$ yields the first-order optimality condition.

$$\frac{dg(\lambda)}{d\lambda} = \frac{2h^2(\lambda - 1)\left((h-1)\lambda + h - 2\right)}{(2(h-1)\lambda - 1)^2} = 0$$

whose roots are $\lambda = \frac{2-h}{h-1}$ and $\lambda = 1$. Studying the sign of $\frac{dg(\lambda)}{d\lambda}$ tells us that $g(\lambda)$ is maximized for $\lambda = 1$, which satisfies (3), as $h < 1$. It thus leads to $\lambda_* = 1$ and $\rho_* = g(\lambda_*) = h$. We can note that the rate found in *Case 1* also matches this expression.

Case 3: $h > 1$**.** In this case, the first constraint becomes

$$\lambda \leq \frac{1}{2(h-1)} \tag{4}$$

which is the same constraint as in the previous case, but with the inequality reversed. The second constraint is the same as in the previous case, with the same function $g(\lambda)$.

When $h \in (1, \frac{3}{2}]$, we observe that $\lambda = 1$ satisfies the constraint (4), so the maximum is attained at $\lambda_* = 1$, yielding $\rho_* = h$.

When $h \in [\frac{3}{2}, 2]$, the maximum of $g(\lambda)$ shifts to $\lambda = \frac{2-h}{h-1}$, which still satisfies the constraint (4). Plugging this into g gives

$$\rho_* = \frac{(2-h)h}{2(h-1)^2} = \frac{1}{2}\left((1-h)^{-2} - 1\right).$$

Combining all three cases, the optimal multiplier is

$$\lambda_* = \begin{cases} 1 & \text{if } h \leq \frac{3}{2}, \\ \frac{2-h}{h-1} & \text{if } h \in [\frac{3}{2}, 2], \end{cases}$$

leading to the following convergence rate, combining two regimes

$$f_0 - f_1 \geq \min\left(h, \frac{1}{2}((1-h)^{-2} - 1)\right) \|g_1\|^2$$

which is equivalent to the first part of the theorem. Then, as the function f is 1-smooth and convex, we can write the following

$$f(x) - f(x_*) \geq \frac{1}{2}\|\nabla f(x)\|^2,$$

which comes from the interpolation inequality 1 written between x and a minimizer x^* (where $\nabla f(x_*) = 0$ holds). This inequality with $x = x_1$ gives $f_1 - f_* \geq \frac{1}{2}\|g_1\|^2$, and adding it to the previsously obtained inequality leads to

$$f_0 - f_* \geq \min\left(h + \tfrac{1}{2}, \tfrac{1}{2}(1-h)^{-2}\right)\|g_1\|^2,$$

which is equivalent to the second bound we wanted to establish. □

This result highlights the well-structured behavior of exact gradient descent, where two distinct regimes naturally emerge depending on the stepsize h. Furthermore, this worst-case convergence rate is *tight*, meaning that it is achieved by an explicit function, and therefore cannot be improved; see [24] for more details and results in the exact case.

2.2. Convergence rate in the inexact case ($\delta \in (0, 1)$)

Building on the intuition and structure of the proof in the exact case, we now turn to the relatively inexact setting. Before presenting the result, we introduce a key quantity that will play a central role in the convergence rate and its proof.

Definition 2.2. *The quantity $\tilde{\lambda}$ is defined as the largest real root of the following cubic equation in λ, depending on both h and δ*

$$\left[(\delta^2-1)h^2+2h\right]\lambda^3+\left[2(\delta^2-1)h^2+5h-4\right]\lambda^2+\left[(\delta^2-1)h^2+4h-4\right]\lambda+[h-1]=0.$$

This quantity, which acts as a multiplier in the proof of the next theorem, is illustrated in Figure 2.2. We now state the main result for a single iteration of relatively inexact gradient descent.

Theorem 2.3 (Convergence rate of one step of inexact gradient descent on smooth convex functions)**.** *One step of Algorithm 1 applied to a convex L-smooth function f with an inexact gradient with relative inexactness $\delta \in$*

$(0,1)$, *started from iterate* x_0 *with a constant stepsize* $h \in [0, \frac{2}{1+\delta}]$, *generates an iterate* x_1 *satisfying*

$$\frac{1}{L}\|\nabla f(x_1)\|^2 \leq C(h,\delta)(f(x_0) - f(x_1)),$$

where $C(h,\delta)$ *is given by the continuous function*

$$C(h,\delta) = \begin{cases} \frac{1}{h(1-\delta)} & \text{if } h \in \left[0, \frac{3}{2(1+\delta)}\right) \quad \textit{(left regime)} \\ \frac{2\tilde{\lambda}}{h\tilde{\lambda}^2 + 2(h-1)\tilde{\lambda} + h - 1} & \text{if } h \in \left[\frac{3}{2(1+\delta)}, \frac{3\delta+2-\sqrt{4-3\delta^2}}{2\delta(\delta+1)}\right] \quad \textit{(intermediate regime)} \\ \frac{2}{(1-h(1+\delta))^{-2} - 1} & \text{if } h \in \left(\frac{3\delta+2-\sqrt{4-3\delta^2}}{2\delta(\delta+1)}, \frac{2}{1+\delta}\right] \quad \textit{(right regime)} \end{cases}$$

and

$$\frac{1}{L}\|\nabla f(x_1)\|^2 \leq \tilde{C}(h,\delta)(f(x_0) - f(x_*)),$$

where $\tilde{C}(h,\delta)$ *is given by the continuous function*

$$\tilde{C}(h,\delta) = \begin{cases} \frac{1}{h(1-\delta)+\frac{1}{2}}, & \text{if } h \in \left[0, \frac{3}{2(1+\delta)}\right) \quad \textit{(left regime)} \\ \frac{2\tilde{\lambda}}{h\tilde{\lambda}^2 + (2h-1)\tilde{\lambda} + h - 1}, & \text{if } h \in \left[\frac{3}{2(1+\delta)}, \frac{3\delta+2-\sqrt{4-3\delta^2}}{2\delta(\delta+1)}\right] \quad \textit{(intermediate regime)} \\ 2(1-h(1+\delta))^2, & \text{if } h \in \left(\frac{3\delta+2-\sqrt{4-3\delta^2}}{2\delta(\delta+1)}, \frac{2}{1+\delta}\right] \quad \textit{(right regime)} \end{cases}$$

Proof. Recall that d_0 denotes the inexact gradient used for the first iteration. We assume again, without loss of generality, that the smoothness constant is $L = 1$. The standard homogeneity argument valid for exact gradient descent also applies to relatively inexact gradient descent, i.e., when the computed direction d_0 satisfies $\|d_0 - \nabla f(x_0)\| \leq \delta\|\nabla f(x_0)\|$, since this condition is preserved under a scaling of function f. We define

$$\tilde{D} \triangleq \frac{d_0 - g_o}{\delta},$$

so that the inequality defining the relatively inexact gradient (1) becomes

$$\|\tilde{D}\|^2 \leq \|g_0\|^2,$$

We write interpolation inequalities Q_{01} and Q_{10}, now involving the inexact

search direction, and the above condition on inexactness as

$$\begin{aligned} Q_{01} : f_0 - f_1 &\geq h g_1 d_0 + \frac{g_1^2}{2} + \frac{g_0^2}{2} - g_0 g_1 \\ &= h\delta g_1 \tilde{D} + (h-1) g_0 g_1 + \frac{g_1^2}{2} + \frac{g_0^2}{2} \\ Q_{10} : f_1 - f_0 &\geq -h g_0 d_0 + \frac{g_1^2}{2} + \frac{g_0^2}{2} - g_0 g_1 \\ &= -h\delta g_0 \tilde{D} + \left(\frac{1}{2} - h\right) g_0^2 + \frac{g_1^2}{2} - g_0 g_1 \\ Q_{inexact} : 0 &\geq \tilde{D}^2 - g_0^2 \end{aligned}$$

We choose the following non-negative multipliers, where b is a nonnegative parameter

$$\begin{aligned} \lambda_{01} &= \lambda + 1 \\ \lambda_{10} &= \lambda \\ \lambda_{inexact} &= b \end{aligned}$$

The nonnegative combination $(\lambda+1)Q_{01} + \lambda Q_{10} + bQ_{\text{inexact}}$ yields

$$f_0 - f_1 \geq \frac{1}{2} \begin{pmatrix} g_0 \\ g_1 \\ \tilde{D} \end{pmatrix} \left[A + \begin{pmatrix} 0 & 0 & 0 \\ 0 & 2\rho & 0 \\ 0 & 0 & 0 \end{pmatrix} \right] \begin{pmatrix} g_0 & g_1 & \tilde{D} \end{pmatrix},$$

with

$$A = \begin{pmatrix} 2(1-h)\lambda - 2b + 1 & (h-2)\lambda - 1 + h & -h\delta\lambda \\ (h-2)\lambda - 1 + h & 2\lambda + 1 - 2\rho & h\delta(\lambda+1) \\ -h\delta\lambda & h\delta(\lambda+1) & 2b \end{pmatrix}$$

where, similarly to the proof of the exact case, we introduce ρ to denote the coefficient of $\|g_1\|^2$ in the rate. We decompose the matrix A using the Schur complement [29] with respect to the bottom right element. This leads to the sum $A = A_1 + A_2$, where

$$A_1 = \begin{pmatrix} -2b - 2\lambda(h-1) + 1 - \frac{\delta^2 h^2 \lambda^2}{2b} & h + \lambda(h-2) - 1 + \frac{\delta^2 h^2 \lambda(\lambda+1)}{2b} & 0 \\ h + \lambda(h-2) - 1 + \frac{\delta^2 h^2 \lambda(\lambda+1)}{2b} & -2\rho + 2\lambda + 1 - \frac{\delta^2 h^2 (\lambda+1)^2}{2b} & 0 \\ 0 & 0 & 0 \end{pmatrix},$$

$$A_2 = \begin{pmatrix} \frac{\delta^2 h^2 \lambda^2}{2b} & -\frac{\delta^2 h^2 \lambda(\lambda+1)}{2b} & -h\delta\lambda \\ -\frac{\delta^2 h^2 \lambda(\lambda+1)}{2b} & \frac{\delta^2 h^2 (\lambda+1)^2}{2b} & h\delta(\lambda+1) \\ -h\delta\lambda & h\delta(\lambda+1) & 2b \end{pmatrix}.$$

Matrix A_2 is rank-1 and positive semidefinite matrix and can be written as

$$A_2 = \begin{pmatrix} -\frac{h\delta\lambda}{\sqrt{2b}} \\ \frac{h\delta(\lambda+1)}{\sqrt{2b}} \\ \sqrt{2b} \end{pmatrix} \begin{pmatrix} -\frac{h\delta\lambda}{\sqrt{2b}} & \frac{h\delta(\lambda+1)}{\sqrt{2b}} & \sqrt{2b} \end{pmatrix},$$

Hence, matrix A will be positive semidefinite as soon as this is the case for matrix A_1, and maximizing the worst-case bound corresponds to maximizing ρ under the constraint that $A_1 \succeq 0$. It is easy to show that $A_1 \succeq 0$ is equivalent to one of the following

- $A_{1_{0,0}} > 0$ and $\det(A_1) \geq 0$ (Cases 1 and 2 below).
- $A_{1_{0,0}} = 0$, $A_{1_{1,0}} = A_{1_{0,1}} = 0$, and $A_{1_{1,1}} \geq 0$ (Case 3 below).

The first two cases lead to straightforward but lengthy computations, which can be carried out symbolically using the Python library `SymPy` [30]. The corresponding code is available at: `https://github.com/Verpierre/Inexact_gradient_descent`. Some further symbolic simplifications were performed with the software Wolfram Mathematica [31].

We now focus on the first two cases, for which the feasibility domain of b and the corresponding upper bound on ρ read

$$\begin{aligned} \max_{\lambda,b,\rho} \quad & \rho \\ \text{s.t.} \quad & b > \frac{1}{4} + \frac{\lambda}{2}(1-h) - \frac{\sqrt{(2\lambda(h(1-\delta)-1)-1)\,(2\lambda(h(1+\delta)-1)-1)}}{4} \quad (c_{1b}) \\ & b < \frac{1}{4} + \frac{\lambda}{2}(1-h) + \frac{\sqrt{(2\lambda(h(1-\delta)-1)-1)\,(2\lambda(h(1+\delta)-1)-1)}}{4} \quad (c_{1b}) \\ & \rho \leq \frac{b^2(8\lambda+4) + b\left(-2h\left(\left(\delta^2-1\right)h(\lambda+1)^2 + 4\lambda + 2\right)\right) + \delta^2h^2(2\lambda+1)}{8b^2 + b(8(h-1)\lambda - 4) + 2\delta^2h^2\lambda^2} \quad (c_2) \end{aligned} \tag{5}$$

Considering the natural extension of the solutions of the exact case (see the proof of Theorem 2.1) leads to the following cases.

Case 1: $\lambda = 1$**.** The problem becomes

$$
\begin{aligned}
\max_{b,\rho} \quad & \rho \\
\text{s.t.} \quad & b > \frac{3}{4} - \frac{h}{2} - \frac{\sqrt{4(1-\delta^2)h^2 - 12h + 9}}{4} \quad (c_{11a}) \\
& b < \frac{3}{4} - \frac{h}{2} + \frac{\sqrt{4(1-\delta^2)h^2 - 12h + 9}}{4} \quad (c_{11b}) \\
& \rho \leq \frac{12b^2 + b\left(-2h\left(4\left(\delta^2-1\right)h + 6\right)\right) + 3\delta^2h^2}{8b^2 + b(8h-12) + 2\delta^2h^2} \quad (c_{12})
\end{aligned}
$$

We denote the objective function as $\rho(b)$. As it is maximized, the problem is equivalent to

$$
\begin{aligned}
\max_{b} \quad & \rho(b) = \frac{12b^2 + b\left(-2h\left(4\left(\delta^2-1\right)h + 6\right)\right) + 3\delta^2h^2}{8b^2 + b(8h-12) + 2\delta^2h^2} \\
\text{s.t.} \quad & b > \frac{3}{4} - \frac{h}{2} - \frac{\sqrt{4(1-\delta^2)h^2 - 12h + 9}}{4} \quad (c_{11a}) \\
& b < \frac{3}{4} - \frac{h}{2} + \frac{\sqrt{4(1-\delta^2)h^2 - 12h + 9}}{4} \quad (c_{11b})
\end{aligned}
$$

The first-order optimality condition for the objective function is

$$
\frac{d\rho(b)}{db} = \frac{(2b-\delta h)(2b+\delta h)(2\delta h - 2h + 3)(2\delta h + 2h - 3)}{\left(4b^2 + 4bh - 6b + \delta^2h^2\right)^2} = 0
$$

whose maximum is reached when $b_* = \frac{h\delta}{2}$. We now check for which values of h this value lies inside the admissible domain from the constraints c_{11a} and c_{11b}. Solving both irrational inequations in (c_{11}) leads to the following condition on h for b_* to be admissible

$$
h < \frac{3}{2(1+\delta)}.
$$

Thus, the solution $b_* = \frac{h\delta}{2}$ holds as soon as $h < \frac{3}{2(1+\delta)}$. Furthermore, replacing b by its optimal value in the expression of the rate $\rho(b)$ leads to

$$
\begin{aligned}
\lambda &= 1 \\
b &= \frac{h\delta}{2} \\
\rho &= h(1-\delta),
\end{aligned}
$$

when $h < \frac{3}{2(1+\delta)}$. Note that extending the multiplier λ from the exact case leads to a rate that is very close to the exact one: it is scaled by a factor $1-\delta$, and equal to the exact case when $\delta = 0$).

Case 2: $\lambda = \frac{2-h(1+\delta)}{h(1+\delta)-1}$. This second case is an extension of the exact one (see the proof of Theorem 2.1), where $h \leftarrow h(1+\delta)$. In this case, problem (5) can be rewritten as

$$
\begin{aligned}
\max_{b,\rho} \quad & \rho \\
\text{s.t.} \quad & b > \frac{1}{4} + \frac{(1-h)\left(-h\left(\delta+1\right)+2\right)}{2(h\left(\delta+1\right)-1)} - \frac{1}{4}\sqrt{\left(-\frac{\left(-h\left(\delta+1\right)+2\right)\left(2\delta h-2h+2\right)}{h\left(\delta+1\right)-1}-1\right)\left(\frac{\left(-h\left(\delta+1\right)+2\right)\left(2\delta h+2h-2\right)}{h\left(\delta+1\right)-1}-1\right)} \quad (c_{21a}) \\
& b < \frac{1}{4} + \frac{(1-h)\left(-h\left(\delta+1\right)+2\right)}{2(h\left(\delta+1\right)-1)} + \frac{1}{4}\sqrt{\left(-\frac{\left(-h\left(\delta+1\right)+2\right)\left(2\delta h-2h+2\right)}{h\left(\delta+1\right)-1}-1\right)\left(\frac{\left(-h\left(\delta+1\right)+2\right)\left(2\delta h+2h-2\right)}{h\left(\delta+1\right)-1}-1\right)} \quad (c_{21b}) \\
& \rho \leq -\frac{b^2(4(\delta h+h-3)(\delta h+h-1))+b(-2h((\delta+1)h(-\delta+2(\delta+1)h-7)+6))+\delta^2h^2(\delta h+h-3)(\delta h+h-1)}{2\left(b^2\left(4(\delta h+h-1)^2\right)+b\left(-2(\delta h+h-1)\left(2(\delta+1)h^2-(\delta+5)h+3\right)\right)+\delta^2h^2(\delta h+h-2)^2\right)}
\end{aligned}
$$

which is equivalent to

$$
\begin{aligned}
\max_{b} \quad & \rho(b) = -\frac{b^2(4(\delta h+h-3)(\delta h+h-1))+b(-2h((\delta+1)h(-\delta+2(\delta+1)h-7)+6))+\delta^2h^2(\delta h+h-3)(\delta h+h-1)}{2\left(b^2\left(4(\delta h+h-1)^2\right)+b\left(-2(\delta h+h-1)\left(2(\delta+1)h^2-(\delta+5)h+3\right)\right)+\delta^2h^2(\delta h+h-2)^2\right)} \\
\text{s.t.} \quad & b \geq \frac{1}{4} + \frac{(1-h)\left(-h\left(\delta+1\right)+2\right)}{2(h\left(\delta+1\right)-1)} - \frac{1}{4}\sqrt{\left(-\frac{\left(-h\left(\delta+1\right)+2\right)\left(2\delta h-2h+2\right)}{h\left(\delta+1\right)-1}-1\right)\left(\frac{\left(-h\left(\delta+1\right)+2\right)\left(2\delta h+2h-2\right)}{h\left(\delta+1\right)-1}-1\right)}. \\
& b \leq \frac{1}{4} + \frac{(1-h)\left(-h\left(\delta+1\right)+2\right)}{2(h\left(\delta+1\right)-1)} + \frac{1}{4}\sqrt{\left(-\frac{\left(-h\left(\delta+1\right)+2\right)\left(2\delta h-2h+2\right)}{h\left(\delta+1\right)-1}-1\right)\left(\frac{\left(-h\left(\delta+1\right)+2\right)\left(2\delta h+2h-2\right)}{h\left(\delta+1\right)-1}-1\right)}
\end{aligned}
$$

The first-order optimality condition on the objective function yields

$$
\frac{d\rho(b)}{db} = \frac{(2(\delta+1)h-3)(2b(\delta h+h-1)-\delta h)(\delta h(2h(\delta(\delta h+h-3)-1)+3)-2b(2h-3)(\delta h+h-1))}{\left(b^2\left(4(\delta h+h-1)^2\right)+b\left(-2(\delta h+h-1)\left(2(\delta+1)h^2-(\delta+5)h+3\right)\right)+\delta^2h^2(\delta h+h-2)^2\right)^2} = 0
$$

and this rate is maximized for $b_* = \frac{h\delta}{2(h(1+\delta)-1)}$. Constraints (c_{21a}) and (c_{21b}) are satisfied as soon as $h > \frac{3\delta+2-\sqrt{4-3\delta^2}}{2\delta(\delta+1)}$. Replacing b with its optimal value leads to

$$
\begin{aligned}
\lambda &= \frac{2-h(1+\delta)}{h(1+\delta)-1} \\
b &= \frac{h\delta}{2(h(1+\delta)-1)} \\
\rho &= \frac{h(1+\delta)(2-h(1+\delta))}{2(h(1+\delta)-1)^2},
\end{aligned}
$$

for $h > \frac{3\delta+2-\sqrt{4-3\delta^2}}{2\delta(\delta+1)}$. In this case also, we can see a natural evolution of the rate from the exact case, and we can, in particular, recover it when $\delta = 0$.

Case 3: Tight constraint on b. We now handle the remaining regime, corresponding to stepsizes ($h \in [\frac{3}{2(1+\delta)}, \frac{3\delta+2-\sqrt{4-3\delta^2}}{2\delta(1+\delta)}]$), where the constraint $A_{1_{0,0}} = 0$ is saturated. This leads to the following equality involving the variables b, λ, and h

$$\begin{aligned} -2b^2 + (2\lambda(1-h)+1)\, b - \frac{\delta^2 h^2 \lambda^2}{2} &= 0 \\ b &\geq 0 \end{aligned} \tag{6}$$

Under this condition, matrix A_1 reduces to the form

$$A_1 = \begin{pmatrix} 0 & A_{1_{0,1}} & 0 \\ A_{1_{1,0}} & A_{1_{1,1}} & 0 \\ 0 & 0 & 0 \end{pmatrix},$$

where

$$\begin{aligned} A_{1_{1,0}} &= A_{1_{0,1}} = h + \lambda(h-2) - 1 + \frac{\delta^2 h^2 \lambda(\lambda+1)}{2b}, \\ A_{1_{1,1}} &= -2\rho + 2\lambda + 1 - \frac{\delta^2 h^2 (\lambda+1)^2}{2b}. \end{aligned}$$

The matrix is positive semidefinite if and only if $A_{1_{1,0}} = A_{1_{0,1}} = 0$ and $A_{1_{1,1}} \geq 0$. These conditions yield

$$\begin{aligned} (h + \lambda\,(h-2) - 1) + \frac{\delta^2 h^2 \lambda\,(\lambda+1)}{2b} &= 0 \\ \rho \leq \lambda + 0.5 - \frac{\delta^2 h^2 (\lambda+1)^2}{4b} & \end{aligned} \tag{7}$$

The first equation in (7) can be rewritten as

$$-\frac{\delta^2 h^2 \lambda(\lambda+1)}{4b} = \frac{1}{2}(h + \lambda(h-2) - 1)$$

We can now substitute this expression into the inequality on C, which is tight at optimality (i.e., the inequality becomes an equality). This

gives

$$\begin{aligned}\rho &= \lambda + 0.5 - \frac{\delta^2 h^2 (\lambda+1)^2}{4b} \\ &= \lambda + 0.5 - \frac{\delta^2 h^2}{4b}\lambda(\lambda+1) - \frac{\delta^2 h^2}{4b}\frac{\lambda(\lambda+1)}{\lambda} \\ &= \lambda + 0.5 - \frac{1}{2}\left(h - 1 + \lambda(h-2)\right)\left(1 + \frac{1}{\lambda}\right) \\ &= h - 1 + \frac{h}{2}\lambda + \frac{h-1}{2}\frac{1}{\lambda}\end{aligned}$$

Now, using the first equation of (7) in the quadratic relation (6), we get

$$\begin{aligned}-2b^2 + (2\lambda(1-h)+1)b &= \frac{\delta^2 h^2}{2}\lambda^2 \\ &= -b\left[(h-2)\lambda + 1 - \frac{1}{\lambda+1}\right],\end{aligned}$$

hence, since $b \neq 0$, we find

$$\begin{aligned}2b &= (2\lambda(1-h)+1) + (h-2)\lambda + 1 - \frac{1}{\lambda+1} \\ &= 2 - h\lambda - \frac{1}{\lambda+1}\end{aligned}$$

Thus, we obtain closed-form expressions for b and C as functions of λ

$$\begin{aligned}\rho(\lambda) &= h - 1 + \frac{h}{2}\lambda + \frac{h-1}{2}\frac{1}{\lambda} \\ b(\lambda) &= 1 - \frac{h}{2}\lambda - \frac{1}{2(1+\lambda)}\end{aligned}$$

In addition, the constraint from (7) can now be written entirely in terms of λ

$$0 = \left(h - 1 + \lambda(h-2)\right) + \frac{\delta^2 h^2 \lambda(\lambda+1)}{2b(\lambda)}$$

Substituting the expression of $b(\lambda)$ into this equation yields a cubic equation in λ, which defines $\tilde{\lambda}$ as in Definition 2.2. The convergence

rate is maximized for $\lambda_* = \tilde{\lambda}$, the largest real root of this equation. Ultimately, we obtain the following

$$\tilde{\lambda} \text{ as defined in Definition 2.2}$$
$$\rho = h - 1 + \frac{h}{2}\tilde{\lambda} + \frac{h-1}{2}\frac{1}{\tilde{\lambda}} = \frac{h\tilde{\lambda}^2 + 2(h-1)\tilde{\lambda} + h - 1}{2\tilde{\lambda}}$$
$$b = 1 - \frac{h}{2}\tilde{\lambda} - \frac{1}{2(1+\tilde{\lambda})}$$

To conclude, we take $C(h, \delta) = \frac{1}{\rho}$ and discussing the value of λ_* as a function of h and δ, which leads to the first bound in the Theorem, involving $f(x_0) - f(x_1)$ in the right-hand side. The second bound with $f(x_0) - f(x_*)$ is then obtained with the argument used at the end of the proof of Theorem 2.1. □

This convergence rate, depending on the stepsize h according to three distinct regimes, can be visualized in Figure 2.1 for different values of δ, while the evolution of the multiplier λ with h is shown in Figure 2.2.

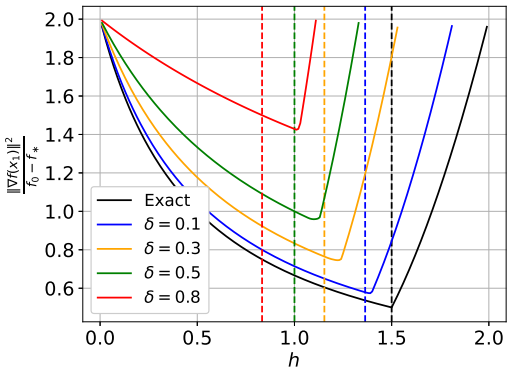
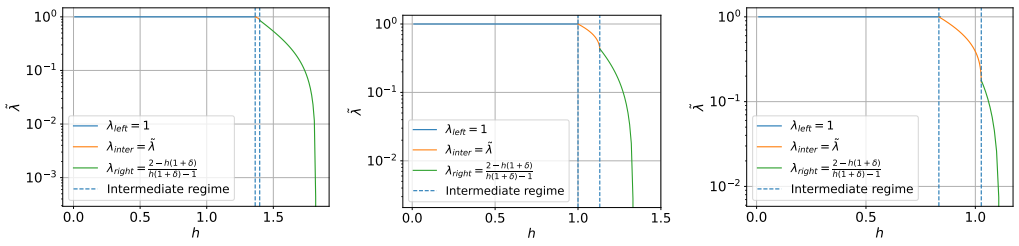


Figure 2.1: Convergence rate of one step of exact and inexact gradient descents with several levels of inexactness δ, depending on the stepsize h. Vertical dotted lines mark the particular stepsize $h = \frac{3}{2(1+\delta)}$



(a) $\delta = 0.1$ (b) $\delta = 0.5$ (c) $\delta = 0.8$

Figure 2.2: Multiplier $\lambda(h)$ from the proof of Theorem 2.3 for different values of δ along the three regimes. This shows that the proof is *continuous* over all step sizes $\in [0, \frac{2}{1+\delta}]$.

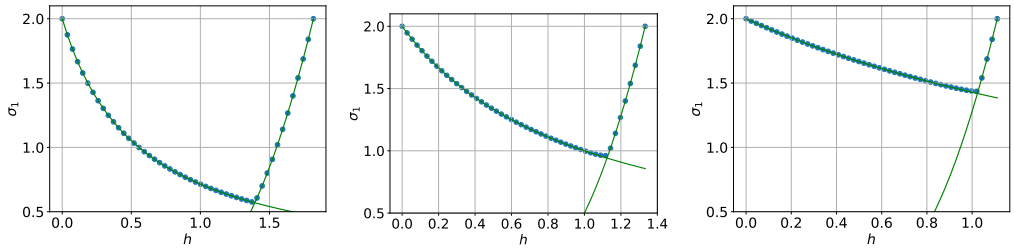
Compared to the exact case, where two stepsize regimes naturally arise (short-step and long-step), an interesting phenomenon occurs in the inexact setting: the emergence of a third distinct regime. We refer to the first regime, $h \in \left[0, \frac{3}{2(1+\delta)}\right)$, as the left or short-step regime. It is followed by the intermediate regime, $h \in \left[\frac{3}{2(1+\delta)}, \frac{3\delta+2-\sqrt{4-3\delta^2}}{2\delta(\delta+1)}\right]$, and finally the right or long-step regime for $h \in [\frac{3\delta+2-\sqrt{4-3\delta^2}}{2\delta(\delta+1)}, \frac{2}{1+\delta}]$.

The short-step and long-step regimes can be seen as natural extensions of the exact case. However, the intermediate regime is entirely new and arises uniquely from the presence of inexactness in the gradient. It does not correspond to any continuous extension of the behavior seen in the exact setting. Instead, it appears as a separate solution branch resulting from the structure of the positivity constraints in the semidefinite programming formulation. Interestingly, numerical experiments show that a similar tripartite regime structure also appears in settings with absolute inexactness, though a thorough analysis of that case falls outside the scope of this paper.

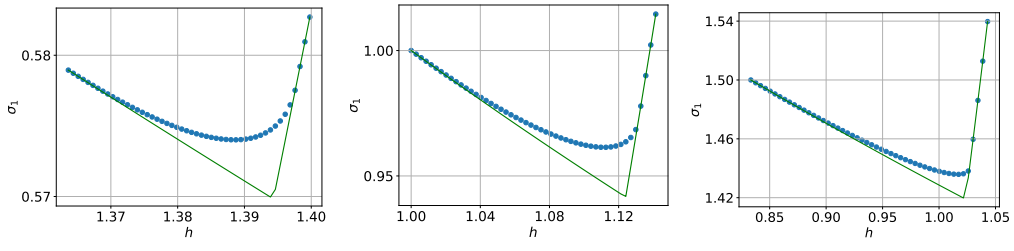
An important point is that this intermediate regime, while narrow, contains the optimal stepsize, i.e., the one that minimizes the worst-case convergence rate. However, characterizing this optimal value analytically is challenging: it results from solving the third-order polynomial equation from Definition 2.2 with coefficients depending on both the stepsize h and the inexactness parameter δ. These equations are generally not solvable in closed form, even with symbolic tools, making a precise analytical expression for the optimal rate intractable.

We observe this behavior clearly in Figure 2.3, which shows the convergence rate curves for three different values of δ. The zoom-in highlights the intermediate regime, revealing a sharp minimum in the rate curve. This

confirms that choosing a stepsize within the intermediate regime yields the best performance. Moreover, a stepsize chosen too large (i.e., beyond the end of the intermediate regime) leads to a rapid deterioration of the convergence rate. This illustrates a form of asymmetry: it's safer to err on the side of smaller stepsizes, as overly aggressive steps can significantly degrade performance. This behavior is similar to what is observed in the exact case [11, 32].



(a) $\delta = 0.1$, $h \in \left[0, \frac{2}{1+\delta}\right]$ (b) $\delta = 0.5$, $h \in \left[0, \frac{2}{1+\delta}\right]$ (c) $\delta = 0.8$, $h \in \left[0, \frac{2}{1+\delta}\right]$



(d) $\delta = 0.1$, $h \in \left[\frac{3}{2(1+\delta)}, \frac{3\delta+2-\sqrt{4-3\delta^2}}{2\delta(\delta+1)}\right]$ (e) $\delta = 0.5$, $h \in \left[\frac{3}{2(1+\delta)}, \frac{3\delta+2-\sqrt{4-3\delta^2}}{2\delta(\delta+1)}\right]$ (f) $\delta = 0.8$, $h \in \left[\frac{3}{2(1+\delta)}, \frac{3\delta+2-\sqrt{4-3\delta^2}}{2\delta(\delta+1)}\right]$

Figure 2.3: (top) Convergence rate (blue dots) for different levels of inexactness for all stepsizes $h \in \left[0, \frac{2}{1+\delta}\right]$, and (bottom) zooming on the intermediate regime. Solid lines are the natural extension of the exact case and correspond to the other regimes.

We emphasize another key structural difference between the exact and relatively inexact settings: the region of guaranteed convergence for the normalized stepsize h shrinks as the inexactness increases. In the exact case ($\delta = 0$), convergence is ensured for all $h < 2$. However, under relative inexactness, convergence is guaranteed only when $h < \frac{2}{1+\delta}$. This dependence on δ reflects a fundamental limitation introduced by the errors: the maximal admissible stepsize necessarily decreases with increasing inexactness. This qualitative behavior, specific to the relatively inexact setting, highlights the

importance of adapting the stepsize to the inexactness level. In particular, the factor $\frac{1}{1+\delta} \leq 1$ appears repeatedly in the analysis, and has motivated an empirical strategy of shortening stepsizes by this factor in [4], with encouraging results. A broader discussion on stepsize selection over multiple iterations is provided after Figure 3.1 in the next section.

Quantitatively speaking, our analysis also significantly improves upon the result of [12] (Theorems 2.4 and 2.5), where convergence is only established for $h < h_{\max} = 2\left(\frac{1-\delta}{1+\delta}\right)^{3/2}$. Our maximal stepsize $h_{\max} = \frac{2}{1+\delta}$ allows for substantially larger values of h, especially when δ is large. This improvement is illustrated in Figure 2.4, where the two bounds are compared. It confirms that our theoretical bound is not only tight but also more permissive than previous results, allowing for faster convergence in practice without compromising stability.

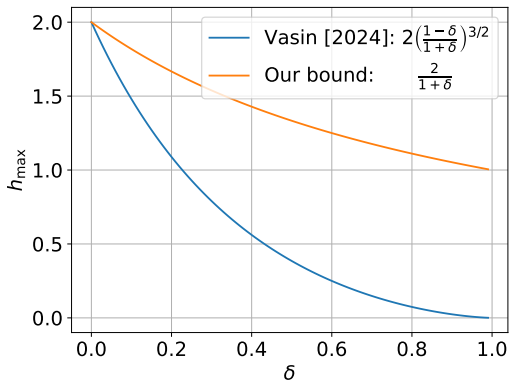


Figure 2.4: Comparison of our $h_{\max}$ for which convergence is guaranteed with the one from [12].

2.3. Tightness of the rate in the left and right regimes

After establishing worst-case convergence rates for the inexact gradient descent in Theorem 2.3, we now show that these rates cannot be improved. For the left and right regimes, we provide an explicit function and a specific inexact direction for which the behavior of the method exactly matches the theoretical bound. In the intermediate regime, while we could not identify a similar explicit worst-case instance, we conjecture that the rate is also tight based on strong numerical evidence. We formalize this in the following result:

Theorem 2.4 (Tightness of left and right regimes)**.** *The left and right rates in Theorem 2.3 are tight. Moreover, they are attained by univariate worst-case functions: a Huber function for the left regime and a quadratic function for the right regime.*

Proof. We prove the tightness for the second rate of Theorem 2.3 (the one with $f(x_0) - f(x_*)$), but the first one can be proved in a similar way (and with the same function).

Left regime $h \leq \frac{3}{2(1+\delta)}$ We consider a univariate Huber-like function, which is 1-smooth and convex

$$f(x) = \begin{cases} \frac{1}{\sqrt{h(1-\delta)+\frac{1}{2}}}|x| - \frac{1}{2h(1-\delta)+1} & \text{, when } |x| \geq \frac{1}{\sqrt{h(1-\delta)+\frac{1}{2}}} \\ \frac{x^2}{2} & \text{, when } |x| < \frac{1}{\sqrt{h(1-\delta)+\frac{1}{2}}} \end{cases}$$

which admits a unique minimizer for $x = 0$ and $f(0) = 0$.

Assuming that $x_0 \geq \frac{1}{\sqrt{h(1-\delta)+0.5}}$, we have $\nabla f(x_0) = \frac{1}{\sqrt{h(1-\delta)+\frac{1}{2}}}$. Choosing $d_0 = (1-\delta)\nabla f(x_0)$, which clearly satisfies the definition of relative δ-inexactness, iterate x_1 is

$$x_1 = x_0 - \frac{h(1-\delta)}{\sqrt{h(1-\delta)+\frac{1}{2}}}.$$

Now choosing x_0 such that $x_1 = \frac{1}{\sqrt{h(1-\delta)+\frac{1}{2}}}$, i.e. $x_0 = \frac{h(1-\delta)}{\sqrt{h(1-\delta)+\frac{1}{2}}} + \frac{1}{\sqrt{h(1-\delta)+\frac{1}{2}}}$, it is straightforward to check that $f(x_0) - f(x_*) = 1$ and we have shown that

$$\begin{aligned} \|g_1\|^2 &= \frac{1}{h(1-\delta)+\frac{1}{2}} \\ &= \frac{f(x_0) - f(x_*)}{h(1-\delta)+\frac{1}{2}}. \end{aligned}$$

Since this matches the convergence rate from Theorem 2.3 exactly, this proves tightness on the left regime.

Right regime $h \geq \frac{3\delta+2-\sqrt{4-3\delta^2}}{2\delta(\delta+1)}$ We now turn to the following univariate quadratic function

$$f(x) = \frac{x^2}{2},$$

whose minimizer is $x = 0$, with corresponding function value $f(0) = 0$. Its gradient is $\nabla f(x) = x$. Furthermore choosing $d_0 = (1+\delta)\nabla f(x_0) = (1+\delta)x_0$, satisfying the definition of relative δ-inexactness leads to iterate x_1 such that

$$\begin{aligned} x_1 &= x_0 - hd_0 \\ &= x_0 - hx_0(1+\delta) \\ &= x_0(1 - h(1+\delta)) \end{aligned}$$

To get $f(x_0) - f(x_*) = f(x_0) = 1$, we thus need $x_0^2 = 2$. Iterate x_1 can thus be computed straightforwardly, and we get

$$\begin{aligned} \|g_1\|^2 = x_1^2 &= x_0^2(1 - h(1+\delta))^2 \\ &= 2(1 - h(1+\delta))^2 \\ &= 2(1 - h(1+\delta))^2(f(x_0) - f(x_*)) \end{aligned}$$

again matching the theoretical bound in Theorem 2.3. Tightness is proven.

□

The above result closely mirrors what occurs in the exact gradient descent setting, where the worst-case for the left and right regimes also happens respectively for a Huber and a quadratic function (see e.g. [24]). Interestingly, the worst-case inexact directions satisfy a very specific structure for those left and right regimes,

$$d_0 = (1 \pm \delta)\nabla f(x_0).$$

This observation leads to the following remark, which offers an intuitive interpretation of the worst-case behavior.

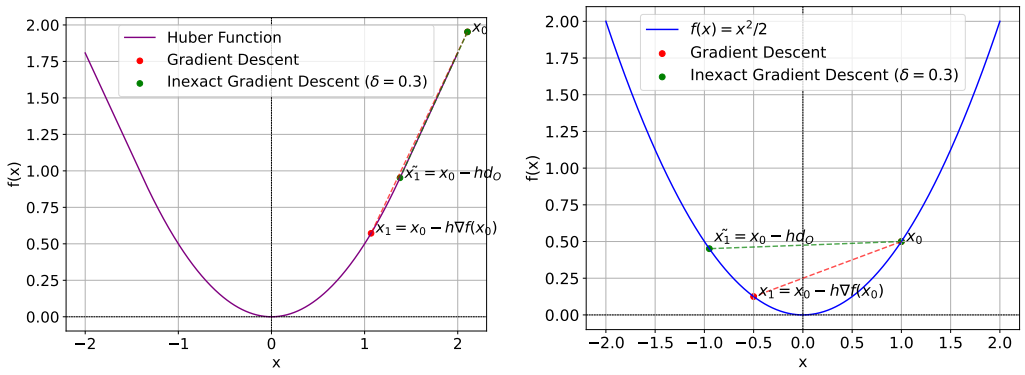
Remark 2.5. *When the inexact direction takes the form $d_0 = (1\pm\delta)\nabla f(x_0)$, the update step of inexact gradient descent can be reinterpreted as an exact gradient step with a miscalibrated stepsize*

$$\begin{aligned} x_1 &= x_0 - hd_0 \\ &= x_0 - h_{\mathit{eff}}\nabla f(x_0), \quad \textit{where } h_{\mathit{eff}} = h(1 \pm \delta). \end{aligned}$$

In other words, the inexactness effectively scales the stepsize, leading to either a too conservative or too aggressive update.

This provides a useful perspective: in the worst-case scenarios, relative inexactness does not merely distort the direction of descent, it acts as a perturbation of the stepsize itself. As such, the worst possible errors under a relative inexactness model can be interpreted as poor stepsize tuning, either undershooting or overshooting the optimal update.

This phenomenon is illustrated in Figure 2.5, which displays the behavior of exact and inexact gradient descent on the 1D worst-case functions identified in Theorem 2.4.



(a) Left (undershoot) regime ($h = 0.75$)

(b) Right (overshoot) regime ($h = 1.5$)

Figure 2.5: Exact and inexact ($\delta = 0.3$) gradient descent on worst-case functions for left and right regimes.

From these plots, we observe a clear pattern: in the left regime, where the stepsize is already small, the worst-case effect of relative inexactness is to make the step even shorter—further slowing down convergence. Conversely, in the right regime, where the stepsize is large, the worst-case is to amplify it even more—leading to larger overshooting.

This interpretation links naturally to earlier insights on gradient descent sensitivity to stepsize, such as those discussed around Figure 3 in [11]. Moreover, similar patterns of behavior seem to emerge when the method is run over multiple iterations, suggesting that the link between relative inexactness and stepsize miscalibration persists across time and may be a core feature of such methods.

2.4. Tightness of the rate in the intermediate regime

We now focus on the intermediate regime, where the situation is more subtle. Indeed, no univariate function seems able to match the bound. As we explain below, this appears to be a structural limitation: achieving the worst-case rate requires geometric flexibility that only arises in dimension two or higher. This leads us to formulate the following conjecture:

Conjecture 2.6 (Tightness of intermediate regime)**.** *The intermediate regime in Theorem 2.3 is tight. Furthermore, the corresponding worst-case function is bivariate.*

This conjecture is supported by strong numerical evidence obtained by solving the PEP. More specifically, for any $\delta \in (0, 1)$, we compute the numerical tight worst-case rate over a fine grid of stepsizes h belonging to the intermediate regime, and observe that all of them perfectly match the bound of Theorem 2.3. In addition, we observe that a specific two-dimensional construction achieves a *worse* convergence rate than any univariate candidate. This suggests that the true worst-case behavior in this regime fundamentally requires two dimensions.

To make this insight precise, assume for contradiction that the worst case is realized by a univariate function. Then, without loss of generality, we may take

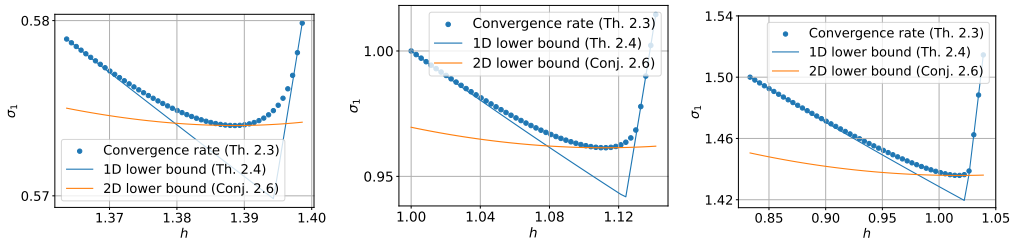
$$\nabla f(x_0) = \begin{pmatrix} a \\ 0 \end{pmatrix}, \quad d_0 = \begin{pmatrix} \tilde{a} \\ 0 \end{pmatrix},$$

and saturate the relative inexactness condition, leading to $\|d_0 - \nabla f(x_0)\| = \delta\|\nabla f(x_0)\|$, so that $\tilde{a} \in [(1-\delta)a, (1+\delta)a]$. Now, consider instead the bivariate construction

$$\nabla f(x_0) = \begin{pmatrix} a \\ 0 \end{pmatrix}, \quad d_0 = \begin{pmatrix} (1-\delta^2)a \\ \delta\sqrt{1-\delta^2}a \end{pmatrix},$$

which also saturates the inexactness condition (1), and enforces that the error $d_0 - \nabla f(x_0)$ is orthogonal to the gradient, i.e. $\langle d_0 - \nabla f(x_0), \nabla f(x_0)\rangle = 0$. Orthogonality is possible due to some rotational freedom that is not available in one dimension. We then maximize the final gradient norm $\|\nabla f(x_1)\|^2$ over all admissible values of $\nabla f(x_1) = (u, v)^\top$. Numerical resolution of the PEP confirms that letting both u and v vary leads to a strictly larger worst-case value than in the univariate case (where necessarily $v = 0$).

This phenomenon is illustrated in Figure 2.6, which compares the theoretical rate with the worst-case rates achieved by 1D and 2D constructions[2]. We observe that the proposed 2D construction matches the theoretical bound specifically at the optimal stepsize, leading to a numerical proof that the bound is tight in that case, whereas the *worse* 1D constructions fall short. This provides strong support for the second part of the conjecture.



(a) $\delta = 0.1$ (b) $\delta = 0.5$ (c) $\delta = 0.8$

Figure 2.6: Convergence rate of intermediate regime from Theorem 2.3 versus 1D and 2D worst-case constructions for different values of δ.

The key insight here is geometric: in the intermediate regime, the worst-case error vector $d_0 - \nabla f(x_0)$ is no longer colinear with the true gradient. Since the relative inexactness condition (1) constrains this error to lie within a ball of radius $\delta\|\nabla f(x_0)\|$, the worst case occurs when the error lies on the boundary of the ball. This gives rise to a rotational degree of freedom that allows the direction error to "turn" around the gradient, leading to greater degradation in performance. This phenomenon is illustrated in Figure 2.7.

[2]Solving a PEP in fixed dimension requires a non-convex formulation, see e.g. [33].

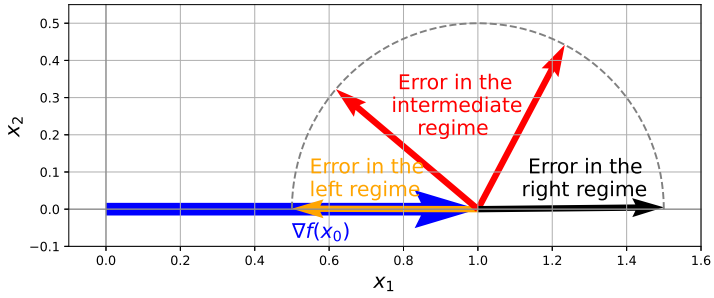


Figure 2.7: Worst-case direction error $d_0 - \nabla f(x_0)$ rotating around the gradient in the intermediate regime (example with normalized gradient and $\delta = 0.5$).

Moreover, numerical evidence from the PEP experiments suggests that the orthogonality condition discussed above is met *precisely at the optimal (best) stepsize*, i.e.,

$$\langle d_0 - \nabla f(x_0), \nabla f(x_0) \rangle = 0 \quad \text{at the best stepsize}$$

meaning that the worst-case error vector is orthogonal to the true gradient only when the method is tuned to its best stepsize. This condition is satisfied by the bivariate construction above and seems to be a consistent feature observed in our numerical worst-case PEP solutions. However, due to the algebraic complexity of the problem, we have not derived an explicit analytical worst-case function in this regime. Investigating whether this orthogonality condition always corresponds to the best stepsize, and how it depends on δ, is an interesting direction for future work.

3. Generalization to several steps

Extending our analysis to multiple iterations reveals new insights, but also new challenges. While the exact convergence rate for a single step of relatively inexact gradient descent already required solving a non-trivial cubic equation, the situation becomes even more complicated when analyzing the behavior over several steps, especially in the intermediate regime of stepsizes. An exact characterization of the worst-case rate for an arbitrary number of iterations is unlikely to be obtainable analytically. However, we can derive useful and informative *upper and lower bounds* that capture the behavior of the method over several iterations.

3.1. Upper bound on the rate for several steps

Theorem 3.1 (Upper bound on the convergence rate for N steps of inexact gradient descent on smooth convex functions)**.** *Algorithm 1 applied to a convex L-smooth function f with an inexact gradient with relative inexactness $\delta \in (0,1)$, started from iterate x_0 with a constant stepsize $h \in [0, \frac{2}{1+\delta}]$, generates iterates satisfying*

$$\frac{1}{L} \min_{k \in \{1,\cdots,N\}} \|\nabla f(x_k)\|^2 \leq \tilde{C}_N(h,\delta)(f(x_0) - f(x_*)),$$

where $\tilde{C}_N(h,\delta)$ is given by the continuous function

$$\tilde{C}_N(h,\delta) = \begin{cases} \frac{1}{Nh(1-\delta)+\frac{1}{2}}, & \text{if } h \in \left[0, \frac{3}{2(1+\delta)}\right) \quad \text{(left regime)} \\ \frac{2\tilde{\lambda}}{N(h\tilde{\lambda}^2+2(h-1)\tilde{\lambda}+h-1)+\tilde{\lambda}}, & \text{if } h \in \left[\frac{3}{2(1+\delta)}, \frac{3\delta+2-\sqrt{4-3\delta^2}}{2\delta(\delta+1)}\right] \quad \text{(intermediate regime)} \\ \frac{2}{N((1-h(1+\delta))^{-2})-(N-1)}, & \text{if } h \in \left(\frac{3\delta+2-\sqrt{4-3\delta^2}}{2\delta(\delta+1)}, \frac{2}{1+\delta}\right] \quad \text{(right regime)} \end{cases}$$

Proof. The first rate from Theorem 2.3 can be generalized in a very straightforward manner. We use it at each iteration to write

$$\frac{1}{L}\|\nabla f(x_{k+1})\|^2 \leq C(h,\delta)(f(x_k) - f(x_{k+1})),$$

with $C(h,\delta)$ defined in the aforementioned theorem, and summing it from $k = 0$ to $N-1$ leads to

$$\begin{aligned} \sum_{k=0}^{N-1} \|\nabla f(x_{k+1})\|^2 &\leq \sum_{k=0}^{N-1} C(h,\delta)(f(x_k) - f(x_{k+1})) \\ &= C(h,\delta)(f(x_0) - f(x_N)) \end{aligned}$$

Note that the sum of N squared gradient norms appearing in the left-hand side is always greater than or equal to N times its minimum term, i.e. we have $\sum_{k=0}^{N-1} \|\nabla f(x_k)\|^2 \geq N \min_{k \in \{1,\cdots,N\}} \|\nabla f(x_k)\|^2$. Following the argument from the end of the proof of Theorem 2.1 allows to replace $f(x_N)$ and $C_N(h,\delta)$ by $f(x_*)$ and $\tilde{C}_N(h,\delta)$, and yields the stated result. □

This upper bound can be visualized in Figure 3.1 for different inexactness levels δ and different numbers of steps N, together with the lower bound from the next subsection and the exact value of the rate computed numerically by a PEP.

3.2. Lower bound on the rate for several steps

We now provide lower bounds on the convergence rate of inexact gradient descent, using explicit smooth convex functions whose convergence rates after N steps are close to the upper bound derived above.

Theorem 3.2 (Lower bound on the convergence rate for N steps of inexact gradient descent on smooth convex functions)**.** *When applying any number of steps $N \in \mathbb{N}$ of Algorithm 1 with an inexact gradient with relative inexactness $\delta \in (0,1)$ with a constant stepsize $h \in [0, \frac{2}{1+\delta}]$, there exists an L-smooth convex function f and an initial point x_0 such that iterates satisfy*

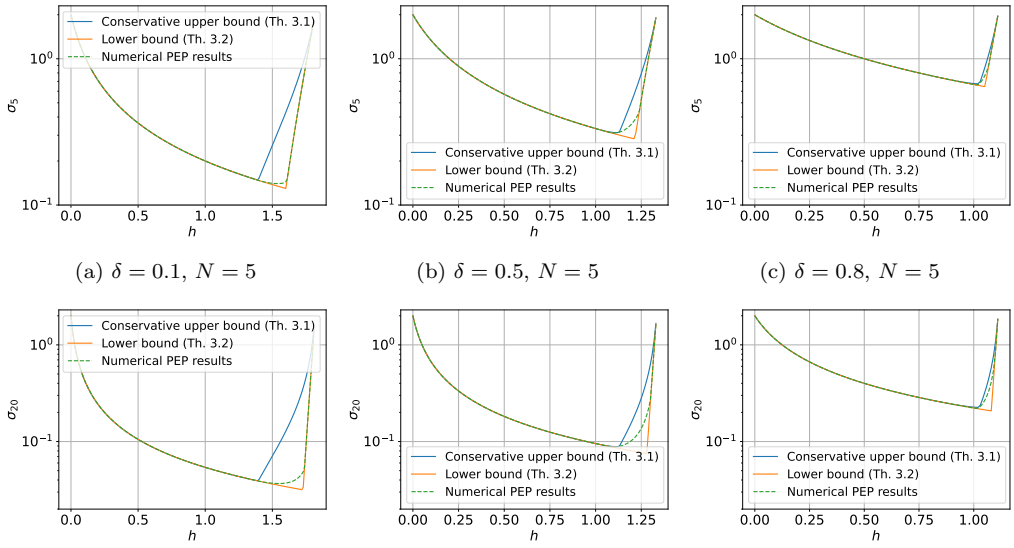
$$\max\left\{\frac{1}{Nh(1-\delta)+\frac{1}{2}}, 2(1-h(1+\delta))^{-2N}\right\}(f(x_0)-f(x_*)) \leq \frac{1}{L}\|\nabla f(x_N)\|^2$$

Proof. Consider the following two 1-smooth convex functions

$$f_1(x) = \begin{cases} \frac{1}{\sqrt{Nh(1-\delta)+\frac{1}{2}}}|x| - \frac{1}{2Nh(1-\delta)+\frac{1}{2}} & \text{, when } |x| \geq \frac{1}{\sqrt{Nh(1-\delta)+\frac{1}{2}}} \\ \frac{x^2}{2} & \text{, when } |x| < \frac{1}{\sqrt{Nh(1-\delta)+\frac{1}{2}}} \end{cases}$$

and $f_2(x) = \frac{x^2}{2}$, i.e. a Huber and a quadratic function. The stated result then follows using an argument similar to that in Theorem 2.4, adapted to the N-steps case. □

The upper and lower bounds provided by Theorems 3.1 and 3.2 are illustrated in Figure 3.1 for several values of inexactness level δ and numbers of iterations N. The PEP-based numerical worst-case rates are plotted alongside the analytical bounds.



(a) $\delta = 0.1$, $N = 5$ (b) $\delta = 0.5$, $N = 5$ (c) $\delta = 0.8$, $N = 5$

(d) $\delta = 0.1$, $N = 20$ (e) $\delta = 0.5$, $N = 20$ (f) $\delta = 0.8$, $N = 20$

Figure 3.1: Lower and upper bounds and exact value of the convergence rate for different inexactness levels δ and different number of steps N.

In the left regime, upper and lower bounds coincide exactly, thereby establishing the tightness of the convergence rate in this case. In contrast, the bounds do not match in the other two regimes; however, they still offer valuable insights into the worst-case behavior of inexact gradient descent when multiple iterations are performed. In particular, we can make an important remark that partially tightens the convergence rate in a subinterval of the right regime.

Remark 3.3. *In the right stepsize regime* $h \in \left]\frac{3\delta+2-\sqrt{4-3\delta^2}}{2\delta(\delta+1)}, \frac{2}{1+\delta}\right[$, *the upper bound*

$$\frac{2}{N((1-h(1+\delta))^{-2}) - (N-1)}$$

is valid. Moreover, it can be shown that a sharper upper bound, matching the lower bound

$$2(1-h(1+\delta))^{-2N},$$

also holds for all $h \geq \tilde{h}(N)$*, for some threshold* $\tilde{h}(N)$ *belonging to the right regime. This leads to a tight convergence rate for several steps in the subinterval* $[\tilde{h}(N), \frac{2}{1+\delta})$*. However, the threshold* $\tilde{h}$ *is defined implicitly as the root*

of a polynomial equation whose degree increases with N, making it difficult to express in closed form.

As an illustration, Figure 3.2 shows how the numerically determined value of $\tilde{h}(N)$ evolves with N for a few values of δ. The Python code used for these experiments, based on the **SymPy** *package but used here purely for numerical purposes, is available online.*

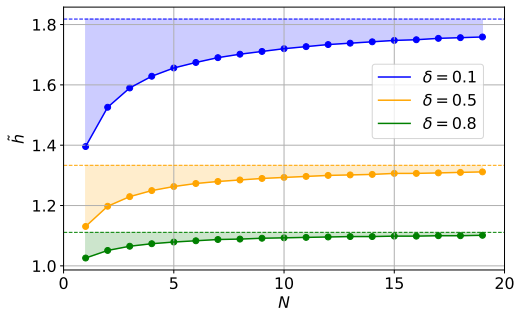


Figure 3.2: Threshold $\tilde{h}(N)$ as a function of N for different values of δ (dots). Dashed lines indicate the maximum stepsize $2/(1+\delta)$, and the shaded regions highlight the interval between $\tilde{h}$ and this threshold.

3.3. Approximation of the optimal stepsize

We now investigate the optimal stepsize $h_{\text{opt}}(N, \delta)$, defined as the value of h that minimizes the (numerical) exact PEP worst-case bound for a given δ and N. Figure 3.3 plots the numerical value of $h_{\text{opt}}(N, \delta)$ (solid lines), and the end of the one-step intermediate regime in the inexact case (dashed lines, see Theorem 2.3).

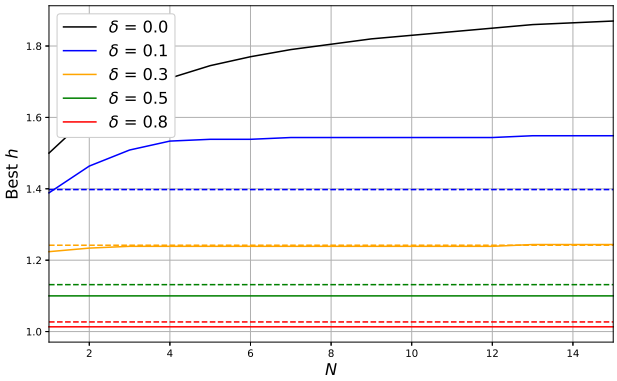


Figure 3.3: Numerical h_{opt} depending on the number of iterations N for different levels of inexactness δ (solid lines) and corresponding end of 1-step intermediate regimes (dashed lines).

We observe two distinct behaviors depending on the inexactness level δ:

- For small values of δ, the optimal stepsize increases with N, much like in the exact case ($\delta = 0$), in which it tends toward 2.
- For moderate to large values of δ, the optimal stepsize stabilizes, becoming nearly independent of N. This indicates a transition to a regime where increasing the number of steps does not justify taking larger stepsizes.

Figure 3.4 confirms that choosing as stepsize the right end of the intermediate regime, namely $h = \frac{3\delta+2-\sqrt{4-3\delta^2}}{2\delta(\delta+1)}$, yields a convergence rate extremely close to the optimal one (at h_{opt}) for a wide range of values of N, especially when $\delta \geq 0.2$. This simple formula, independent from the number of iterations N, is thus a highly effective and practical stepsize choice in the inexact regime.

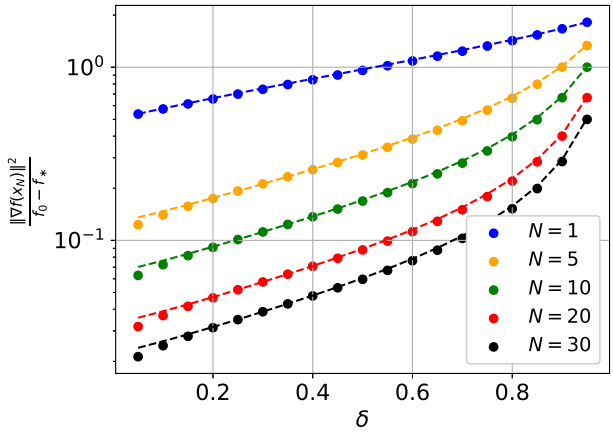


Figure 3.4: Numerical rate at h_{opt} (solid dots) and at its approximation by $\frac{3\delta+2-\sqrt{4-3\delta^2}}{2\delta(\delta+1)}$ (dashed lines) for different number of iterations.

This sharply contrasts with the exact case, where stepsize h must grow with N to maintain optimality. In the relatively inexact setting, large values of the stepsize can lead to excessive amplification of errors, which explains the stabilization of h_{opt} as N increases.

4. Conclusion

In this work, we conducted a detailed analysis of the constant stepsize gradient descent method under relative inexactness. Our contributions are twofold.

First, in Section 2, we established a tight theoretical characterization of the one-step behavior of relatively inexact gradient descent on smooth convex functions. We derived an explicit worst-case convergence rate σ_1, composed of three different stepsize regimes, and proved its tightness, either analytically or numerically. We observed that inexactness does not significantly deteriorate the convergence rate, but noted that smaller stepsizes should be used compared to the exact case. We also discussed that characterizing the optimal stepsize analytically is hard, due to difficult symbolic computations.

Secondly, in Section 3, we extended the analysis to multiple iterations, providing both upper and lower bounds on the worst-case convergence rate. This was achieved through a combination of analytical reasoning and computer-aided proofs using the Performance Estimation Problem (PEP) framework.

The provided lower and upper bounds were often close to each other, providing insight about the method's behavior when performing several iterations. We also observed that the best stepsize according to the obtained worst-case convergence rates is not very sensitive to the number of iterations, highlighting a significant difference with the exact case. We also gave a simple analytical approximation of the optimal stepsize that can deliver near-optimal performance across any number of iterations.

Beyond these results, our study highlights the intrinsic complexity of analyzing the exact worst-case behavior of relatively inexact methods. In particular, the intermediate regime involves solving a high-degree polynomial system, whose structure resists symbolic simplification. This suggests that the difficulty is not only technical but may reflect a deeper hardness of the underlying worst-case analysis. However, it is also possible that this hardness results from some choices made in the setup of our analysis (such as the performance criteria used to measure convergence). Nevertheless, such challenges underscore the importance of combining symbolic analysis with numerical and computer-assisted tools to gain insights into complex inexact optimization settings.

Looking ahead, several promising research directions emerge. First, our numerical results suggest that the worst-case error vector becomes orthogonal to the true gradient precisely when the stepsize is optimally tuned. Formalizing this observation and understanding how this orthogonality condition depends on the inexactness level δ remains an open theoretical challenge. Second, our analysis reveals a shift in optimal algorithm design under relative gradient noise in the smooth convex setting: taking many small steps is preferable to using large steps. Extending this insight to the stochastic case, where inexactness arises from sampling noise, is a natural and important next step. In particular, it would be valuable to investigate whether similar stepsize tuning principles apply when stochasticity and relative inexactness interact. Initial progress in this direction has been made recently by using the PEP methodology to analyze stochastic gradient descent without variance assumptions [34]. This analysis has been expanded even more recently in [35], but the full picture remains to be explored.

References

[1] M. Fazlyab, A. Robey, H. Hassani, M. Morari, G. Pappas, Efficient and accurate estimation of Lipschitz constants for deep neural networks, Advances in neural information processing systems 32 (2019).

[2] G. Lan, First-order and stochastic optimization methods for machine learning, Vol. 1, Springer, 2020.

[3] Y. Nesterov, Introductory lectures on convex optimization: A basic course, Vol. 87, Springer Science & Business Media, 2013.

[4] P. Vernimmen, F. Glineur, Empirical and computer-aided robustness analysis of long-step and accelerated methods in smooth convex optimization, arXiv preprint arXiv:2506.09730 (2025).

[5] A. d'Aspremont, Smooth optimization with approximate gradient, SIAM Journal on Optimization 19 (3) (2008) 1171–1183. doi:10.1137/060676386.

[6] O. Devolder, F. Glineur, Y. Nesterov, First-order methods of smooth convex optimization with inexact oracle, Mathematical Programming 146 (2014) 37–75.

[7] N.-C. Kempke, T. Koch, Low-precision first-order method-based fix-and-propagate heuristics for large-scale mixed-integer linear optimization, arXiv preprint arXiv:2503.10344 (2025).

[8] E. De Klerk, F. Glineur, A. B. Taylor, On the worst-case complexity of the gradient method with exact line search for smooth strongly convex functions, Optimization Letters 11 (2017) 1185–1199.

[9] E. De Klerk, F. Glineur, A. B. Taylor, Worst-case convergence analysis of inexact gradient and newton methods through semidefinite programming performance estimation, SIAM Journal on Optimization 30 (3) (2020) 2053–2082.

[10] Y. Drori, M. Teboulle, Performance of first-order methods for smooth convex minimization: a novel approach, Mathematical Programming 145 (1) (2014) 451–482.

[11] A. B. Taylor, J. M. Hendrickx, F. Glineur, Smooth strongly convex interpolation and exact worst-case performance of first-order methods, Mathematical Programming 161 (2017) 307–345.

[12] A. Vasin, Gradient directions and relative inexactness in optimization and machine learning, arXiv preprint arXiv:2407.00667 (2024).

[13] A. Vasin, A. Gasnikov, P. Dvurechensky, V. Spokoiny, Accelerated gradient methods with absolute and relative noise in the gradient, Optimization Methods and Software 38 (6) (2023) 1180–1229.

[14] N. Kornilov, E. Gorbunov, M. Alkousa, F. Stonyakin, P. Dvurechensky, A. Gasnikov, Intermediate gradient methods with relative inexactness, arXiv preprint arXiv:2310.00506 (2023).

[15] A. Vasin, V. Krivchenko, D. Kovalev, F. Stonyakin, N. Tupitsa, P. Dvurechensky, M. Alkousa, N. Kornilov, A. Gasnikov, On solving minimization and min-max problems by first-order methods with relative error in gradients, arXiv preprint arXiv:2503.06628 (2025).

[16] N. Hallak, K. Y. Levy, A study of first-order methods with a probabilistic relative-error gradient oracle, Preprint, optimization-online. URL `https://optimization-online.org/wp-content/uploads/2025/02/RAGM.pdf`

[17] P. D. Khanh, B. Mordukhovich, D. B. Tran, Convergence of first-order algorithms with momentum from the perspective of an inexact gradient descent method, arXiv preprint arXiv:2505.03050 (2025).

[18] Y. Bello-Cruz, M. L. Gonçalves, J. G. Melo, C. Mohr, A relative inexact proximal gradient method with an explicit linesearch, Journal of Optimization Theory and Applications 206 (1) (2025) 1–32.

[19] S. Wang, V. Y. Tan, Robust distributed gradient descent to corruption over noisy channels, in: 2024 IEEE International Symposium on Information Theory (ISIT), IEEE, 2024, pp. 2520–2525.

[20] A. Vasin, A. Gasnikov, V. Spokoiny, Stopping rules for accelerated gradient methods with additive noise in gradient (2021).

[21] D. B. Thomsen, A. Taylor, A. Dieuleveut, Tight analyses of first-order methods with error feedback, arXiv preprint arXiv:2506.05271 (2025).

[22] P. Vernimmen, F. Glineur, Convergence analysis of an inexact gradient method on smooth convex functions, in: Proceedings of the 32nd European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning, 2024, pp. 125–130.

[23] D. Kim, J. A. Fessler, Optimizing the efficiency of first-order methods for decreasing the gradient of smooth convex functions, Journal of optimization theory and applications 188 (1) (2021) 192–219.

[24] T. Rotaru, F. Glineur, P. Patrinos, Exact worst-case convergence rates of gradient descent: a complete analysis for all constant stepsizes over nonconvex and convex functions, arXiv preprint arXiv:2406.17506 (2024).

[25] A. B. Taylor, J. M. Hendrickx, F. Glineur, Exact worst-case performance of first-order methods for composite convex optimization, SIAM Journal on Optimization 27 (3) (2017) 1283–1313.

[26] MOSEK ApS, The MOSEK Python Fusion API manual. Version 11.0. (2025).
URL `https://docs.mosek.com/latest/pythonfusion/index.html`

[27] A. B. Taylor, J. M. Hendrickx, F. Glineur, Performance estimation toolbox (pesto): Automated worst-case analysis of first-order optimization methods, in: 2017 IEEE 56th Annual Conference on Decision and Control (CDC), IEEE, 2017, pp. 1278–1283.

[28] B. Goujaud, C. Moucer, F. Glineur, J. M. Hendrickx, A. B. Taylor, A. Dieuleveut, PEPit: computer-assisted worst-case analyses of first-order optimization methods in Python, Mathematical Programming Computation 16 (3) (2024) 337–367.

[29] F. Zhang, The Schur complement and its applications, Vol. 4, Springer Science & Business Media, 2006.

[30] A. Meurer, C. P. Smith, M. Paprocki, O. Čertík, S. B. Kirpichev, M. Rocklin, A. Kumar, S. Ivanov, J. K. Moore, S. Singh, T. Rathnayake, S. Vig, B. E. Granger, R. P. Muller, F. Bonazzi, H. Gupta, S. Vats, F. Johansson, F. Pedregosa, M. J. Curry, A. R. Terrel, v. Roučka, A. Saboo,

I. Fernando, S. Kulal, R. Cimrman, A. Scopatz, SymPy: symbolic computing in Python, PeerJ Computer Science 3 (2017) e103.

[31] Wolfram Research Inc., Mathematica, Version 14.2, champaign, IL, 2024.
URL `https://www.wolfram.com/mathematica`

[32] P. Vernimmen, Tight convergence analysis of exact and inexact gradient methods with constant and silver schedules, Master's thesis, UCLouvain (2024).

[33] S. Das Gupta, B. P. Van Parys, E. K. Ryu, Branch-and-bound performance estimation programming: A unified methodology for constructing optimal optimization methods, Mathematical Programming 204 (1) (2024) 567–639.

[34] D. Cortild, L. Ketels, J. Peypouquet, G. Garrigos, New tight bounds for SGD without variance assumption: A computer-aided lyapunov analysis, arXiv preprint arXiv:2505.17965 (2025).

[35] A. Rubbens, S. Colla, J. M. Hendrickx, Computer-aided analyses of stochastic first-order methods, via interpolation conditions for stochastic optimization, arXiv preprint arXiv:2507.05466 (2025).