École polytechnique de Louvain

Performance estimation of the gradient method with fixed arbitrary step sizes

Author: **Antoine Daccache**
Supervisors: **Pr François Glineur, Pr Julien Hendrickx**
Reader: **Pr Yurii Nesterov**
Academic year 2018–2019
Master [120] in Mathematical Engineering

Performance estimation of the gradient method with fixed arbitrary step sizes

Daccache Antoine

August 18, 2019

Abstract

Based on the recently developed performance estimation framework ([DT13, Tay17]) and the associated tool Pesto toolbox [THG17a], we attempt to estimate the exact performance of the classical gradient method with fixed but arbitrary step sizes. We provide a new conjecture on the exact worst-case bound for two fixed but arbitrary step sizes of the gradient method for smooth convex and strongly convex functions. We also provide a partial conjecture for three steps in the case of smooth convex functions. Following these conjectures, we provide an estimation on the shape of the worst-case bound for N variable step sizes.

Based on those exact worst-case bounds, we also estimate the optimal step sizes for two and three arbitrary fixed steps and compare to the case of two and three constant steps, from which we infer a trend for N steps.

We also focus on obtaining proofs for these bounds based on the interpolation inequalities introduced in [Tay17], we introduce the concepts of minimal proof and local minimal proofs, corresponding to the minimal set of inequalities needed to provide a proof for all and some step sizes respectively, as well as a procedure to obtain an estimation of these optimal proofs. Finally, we apply this procedure to the case of two steps and we provide elements of the proof of the conjecture previously identified.

Acknowledgements

I would like to start by thanking my supervisors professors François Glineur and Julien Hendrickx for the opportunity to work on this subject and professor Glineur for his time all year and his support and insights that where very important to undertake this work.
Big thanks to my friends for their support specially Guillaume for his feedback on the report and Brieuc for the inspiring discussions and Benedicte for her moral support.
Finally I would like to thank my father for his feedback on the language in the report and my mother for the good mood at home.

Contents

Chapter 1

Introduction

Nowadays, first-order methods are regaining researcher's attention due to the relatively inexpensive computational cost per iteration. Although they do not usually provide high accuracy, they are suited for large scale problems, for example in machine learning where the least sophisticated optimization algorithms turns out to perform well as we can read in [BB08] foer example with the stochastic gradient descent. These methods are even more suited to convex optimization, where we have some theoretical guarantees on the existence of a minimum. When you need to choose a first-order method, information about its worst-case performance is a good criteria to consider. For example information on the accuracy of the final iterate after N iterations of the method, or the corresponding objective function accuracy are one of the performance criteria you could be interested in. It is not possible to focus on one specific problem, hence in general for each method we provide an upper bound on its performance also called the worst-case bound for a class of problems.

In the classic literature, you can usually find rates of convergence, but recently a new formulation of the problem of finding the *exact* worst-case bound on some performance criterion as a semi-definite program made it possible to find exact closed form of the worst-case bound for first-order methods in the context of convex optimization ([DT13] and [Tay17]). The process of finding worst-case bounds is even automated thanks to the PESTO toolbox [THG17a], a Matlab toolbox developed by A.Taylor, F.Glineur and J.Hendickx.

In this work we decided to use this toolbox on the simplest first-order method, the gradient method (GM). This method, tracing back to Cauchy [dC47], is the first and simplest first order method that you encounter when you study optimization. It has the property of being memory-less, meaning that you do not need to store information used in the previous iterations. Most extensions of this method require some storage if so to say, in order to obtain a better performance such as with accelerated methods [Nes98]. We can find in the literature for example in [DT13] and [Tay17] some conjectures on the exact worst-case bound on the objective function accuracy of this method with fixed constant step sizes, meaning that all the step sizes are chosen beforehand (fixed i.e. with no line search for example) and will have the same value (constant). In this work we are interested in the exact worst-case bound on the objective function accuracy of the GM with fixed variable step sizes. So the step sizes are still fixed beforehand but they can be different from one iteration to another. Our first goal is to develop a methodology or a procedure, based on the PESTO toolbox [THG17a], in order to provide a conjecture on the exact worst-case bound. The second goal would be to exploit the toolbox in order to get some intuitions for writing an explicit proof. Concretely, the thesis will be organized in three parts:

- In the first part, we start in Chapter 2 by presenting a review of the essential concepts that will be needed, essentially the concept of convex interpolation based on ([Tay17],[THG17b]). In Chapter 3 we review the SDP formulation of the problem of finding the worst-case bound described in [Tay17]. Then we introduce the procedure that we will follow to conjecture an exact worst-case bound and to prove it.

- In the second part, we detail in Chapter 4 the identification of the exact worst-case bound on the objective function accuracy and show the difficulties and the limitations of the approach we followed. Then, in light of the results we obtained, we compare with the case of constant step sizes.

- In the third and final part, we provide in Chapter 5 some concepts in order to provide proofs for those exact worst-case bounds and we apply them to provide elements of proofs for two arbitrarily fixed steps of GM.

Chapter 2

Convex interpolation

The main purpose of this introductory chapter is to define the concept of convex interpolation, that will be a corner stone for the remainder of this work. We will start with a first section that will introduce the two functional classes of interest in this work as well as some results in convex analysis, then the second and main section will present the key result on convex interpolation that will be used in the next chapters.
We will restrict this work to $\mathbb{R}^d$ and use the standard inner product $\langle x, y\rangle = x^T y$ and the induced norm, classical Euclidean norm $\|x\|_2 = \sqrt{x^T x}$. The results stated afterwords in this chapter, are general to any real vector space. As reference for this chapter chapters 2 and 3 of [Tay17] and [THG17b].

2.1 Elements of convex analysis

We start by recalling one common possible definition of a convex function. We will use the notation $\bar{\mathbb{R}} = \mathbb{R} \cup \{+\infty\}$.

Definition 2.1. *A function $f : \mathbb{R}^d \to \bar{\mathbb{R}}$ will be considered convex if for all $x, y \in \mathbb{R}^d$ and for any $\lambda \in [0.1]$ we have:*

$$f(\lambda x + (1-\lambda)y) \leq \lambda f(x) + (1-\lambda) f(y)$$

We can also define a convex function by requiring that the epigraph $epi f = \{(x,t) \in \mathbb{R}^d \times \mathbb{R} \,|\, f(x) \leq t\}$, to be a convex set. We are interested in closed proper convex functions, meaning that their epigraph is a non-empty closed set.

Notations 2.1. *We denote by $\mathcal{F}_{0,\infty}$ the class of closed proper convex functions.*

We remind also the defintion of subgradient and subdifferentials.

Definition 2.2. *Consider $g \in \mathbb{R}^d$ and $f : \mathbb{R}^d \to \bar{\mathbb{R}}$. Then g will be considered a subgradient of f at $x \in \mathbb{R}^d$ if it satisfies for all $y \in \mathbb{R}^d$:*

$$f(y) \geq f(x) + \langle g, y - x\rangle$$

At any point x, the set of all subgradients, called subdifferential of f at x and denoted by $\partial f(x)$. In the case of differentiable functions, which will be considered starting from the next chapter, the subdifferential at each x reduces to a singleton containing the gradient. So the subgradient is actually a replacement of the gradient in absence of differentiability. We can use subdifferentials to characterize convex functions, as we can see in [Theorem 3.1.13 Nes98], a function will be considered convex if and only if at any x on the interior of its domain, $\partial f(x)$ is not empty and bounded. At this stage, we can show one main grantee provided by working in the context of convex optimization, which states that every local optimum is global for convex functions. This is a direct consequence of the optimality condition that we state bellow without a proof. For one possible reference for the proof and the statement [Theorem 3.1.15 Nes98] and in our main reference the optimality condition is presented in Theorem 2.15 along with the proof [Chapter 2 Tay17, p.22].

Theorem 2.1. *Let f be a proper and convex function, then $f(x_*) = \min_{x \in Domf} f(x)$ if and only if $0 \in \partial f(x_*)$.*

So resumed in simple word and in the context of differentiable functions, for a convex function is it is sufficient to find x such that $\nabla f(x) = 0$ to find a global optimum.
Now that we have defined convex functions in general, we can go on and define the notion of smoothness. As explained in [Chapter 1, section 1.1.2 Nes98], in optimization it is common to add some conditions on the magnitude of the derivatives of the function if it is differentiable functions in order to get some properties of the minimization algorithms like rate of convergence for example. In the case of non differentiable functions, we impose such conditions on the subgradients. Usually it is done by requiring the function to be smooth and here is a defintion from our main reference [Definition 2.26 Tay17, p.27] for smoothness of closed proper convex functions.

Definition 2.3. *Consider $f \in \mathcal{F}_{0,\infty}$ and a constant $L \in \bar{\mathbb{R}}^+$. We say that f is L-smooth if it satisfies:*

$$\frac{1}{L} \|g_1 - g_2\|_2 \leq \|x_1 - x_2\|_2 \tag{2.1}$$

for all x_1, x_2 in $\mathbb{R}^d$ and $g_1 \in \partial f(x_1)$ and $g_2 \in \partial f(x_2)$.

When L is finite, condition (2.1), simply reduces to the Lipschitz condition on the gradient as explained in [Tay17], that can be written as follows:

$$\frac{1}{L} \|\nabla f(x_1) - \nabla f(x_2)\|_2 \leq \|x_1 - x_2\|_2$$

So this definition, generalizes the Lipschitz condition to include non differentiable functions when $L = \infty$, hence the usage of the subgradients. We could have shown the standard Lipschitz condition only, since in this work we are interested in differentiable functions only, but we chose to go for this defintion in order to follow the main reference and also because it is suited for the discussion about Legendre-Fenchel conjugation of smooth convex functions.

Now, we will introduce a stronger version of convexity, strong convexity. The first definition we will provide is a general definition, that works with or without differentiability.

Definition 2.4. *Consider a closed proper convex function f in $\mathcal{F}_{0,\infty}(\mathbb{R}^d)$ and a constant $\mu \in \mathbb{R}^+$. We say that f is μ-strongly convex if it satisfies for any x and y in $\mathbb{R}^d$ and any $\lambda \in [0,1]$:*

$$f(\lambda x + (1-\lambda)y) \leq \lambda f(x) + (1-\lambda)f(y) - \lambda(1-\lambda)\frac{\mu}{2}\|x-y\|_2^2$$

Notice that when we $\mu = 0$ the function will just be convex as the previous Definition reduces to Definition 2.1.

Notations 2.2. *Consider $L \in \bar{\mathbb{R}}^+$ and $\mu \in \mathbb{R}^+$.*
The class of L-smooth closed proper convex functions will be denoted by $\mathcal{F}_{0,L}$.
The class of L-smooth μ-strongly convex closed proper functions will be denoted by $\mathcal{F}_{\mu,L}(\mathbb{R}^d)$

Their are many other equivalent ways of defining or viewing strongly convex functions. One possible view is the following and corresponds to [Theorem 2.32 Tay17, p.29].

Theorem 2.2. *Consider $f \in \mathcal{F}_{0,\infty}(\mathbb{R}^d)$. The function $f \in \mathcal{F}_{\mu,\infty}(\mathbb{R}^d)$ if and only if*

$$f(y) \geq f(x) + \langle g, y-x \rangle + \frac{\mu}{2}\|y-x\|_2^2$$

for all $x, y \in \mathbb{R}^d$ and $g \in \mathbb{R}^d$ such that $g \in \partial f(x)$

It is interesting to notice that in chapter 2 of [Nes98] the version of the previous Theorem for differentiable functions is used as a definition [Definition 2.1.2 Nes98], and Defintion 2.4 is showed after [Theorem 2.1.8 Nes98]. Again we chose this presentation in order to follow our main reference.
Now we will state two results that are equivalent views of strongly convex functions and will be used later in this chapter. They correspond to Theorem 2.31 and Theorem 2.33 in [Tay17], adapted to the case of $\mathbb{R}^d$. The first one actually is a well known equivalent characterization of strongly convex functions:

Theorem 2.3. *$f \in \mathcal{F}_{\mu,\infty}(\mathbb{R}^d)$ if and only if $f(x) - \frac{\mu}{2}\|x\|_2^2$ is in $\mathcal{F}_{0,\infty}$*

The second theorem is an extension of 2.3 to include the case of smooth strongly convex functions:

Theorem 2.4. *Consider a function $f \in \mathcal{F}_{0,\infty}(\mathbb{R}^d)$. Then we have the following equivalence:*

$$f \in \mathcal{F}_{\mu,L}(\mathbb{R}^d) \Leftrightarrow f(x) - \frac{\mu}{2}\|x\|_2^2 \in \mathcal{F}_{0,L-\mu}(\mathbb{R}^d)$$

We recall also without going in details the notion of Legendre- Fenchel conjugate a concept well explained in [Section 2,4 Tay17, p.22]. It is basically a transformation that will allow to deal with properties concerning differentiability easier. It is a generalization of the Legendre transform that can be seen as the one switching coordinates for f in $\mathbb{R}^d$ to differentials of f^* and coordinates of f^* to differentials of f. The generalization, Fenchel-Legendre conjugation presented in the work of Werner Fenchel, is defined formally as follows in the case of $\mathbb{R}^d$:

Definition 2.5. *Let $f : \mathbb{R}^d \to \mathbb{R}\cup\{\infty\}$ be a function, its Legendre-Fenchel conjugate $f^* : \mathbb{R}^d \to \mathbb{R}\cup\{\infty\}$ is defined as follows:*

$$f^*(s) = \sup_{x\in\mathbb{R}^d} \langle s, x\rangle - f(x)$$

Also denote by $f^{**} : \mathbb{R}^d \to \mathbb{R}\cup\{\infty\}$ the biconjugate function:

$$f^{**}(x) = \sup_{s\in\mathbb{R}^d} \langle s, x\rangle - f^*(s)$$

Now we state the following theorems that will also be used in the next section, which corresponds to a version of Theorem 2.21 and 2.34 in [Tay17] for $\mathbb{R}^d$. First we start with the following theorem illustrating how the coordinates of a function f will become sub-gradients of f^* and sub-gradients of f^* coordinates of f.

Theorem 2.5. *Consider $f \in \mathcal{F}_{0,\infty}(\mathbb{R}^d)$ and its Lgendre-Fenchel conjugate $f^* \in \mathcal{F}_{0,\infty}(\mathbb{R}^d)$. The following propositions are equivalent for any $x \in \mathbb{R}^d$ and $x^* \in \mathbb{R}^d$:*

1. $f(x) + f^*(x^*) = \langle x^*, x\rangle$
2. $x^* \in \partial f(x)$
3. $x \in \partial f^*(x^*)$

Then we present a link between smoothness and strong convexity via the conjugate function:

Theorem 2.6. *Consider a function $f \in \mathcal{F}_{0,\infty}(\mathbb{R}^d)$. We have the following equivalence:*

$$f \in \mathcal{F}_{0,L}(\mathbb{R}^d) \Leftrightarrow f^* \in \mathcal{F}_{1/L,\infty}(\mathbb{R}^d)$$

Now that we have presented all the basics we need, we will present in the next section the main result on convex interpolation.

2.2 Interpolation in $\mathcal{F}_{\mu,L}$

Suppose first that you have a set of points $S = \{(x_i, f_i, g_i)\}_{i\in I}$ where I is a set of indices, so that if you have N points. You want to interpolate this set by a function in $\mathcal{F}_{\mu,L}(\mathbb{R}^d)$. We start by the following definition:

Definition 2.6. *A set $S = \{(x_i, f_i, g_i)\}_{i\in I}$ with x_i, g_i in $\mathbb{R}^d$ and f_i in $\mathbb{R}$ for all $i \in I$, is $\mathcal{F}_{\mu,L}(\mathbb{R}^d)$-interpolable if and only if there exist a function $F \in \mathcal{F}_{\mu,L}(\mathbb{R}^d)$ such that for all $i \in I$ we have $F(x_i) = fi$ and $g_i \in \partial F(x_i)$.*

Hence we want to find at least one L-smooth μ-strongly convex function that satisfies this condition. As explained in [Section 3.2 Tay17, p 45] the problem is not that easy and a naive approach consisting on imposing simply a discrete version of the main inequalities of the class is not always good as we can see in this original example in $\mathbb{R}$:

Example 2.1. *Consider a set of three points, $S = \{(x_i, f_i, g_i)\}$ such that $I = \{1,2,3\}$, that we want to interpolate by a smooth convex function $F \in \mathbb{F}_{0,L}(\mathbb{R})$. The points are given by:*

$$(x_1, f_1, g_1) = \left(-1, \frac{1}{2}, -\frac{1}{2}\right) \quad (x_2, f_2, g_2) = \left(0, 0, \frac{1}{2}\right) \quad (x_3, f_3, g_3) = \left(1, \frac{1}{2}, \frac{1}{2}\right)$$

One naive approach is to simply impose the following conditions:

$$\begin{aligned} f_i &\geq f_j + g_j(x_i - x_j) \quad \forall i, j \in I \\ |g_i - g_j| &\leq L\,|x_i - x_j| \quad \forall i, j \in I \end{aligned} \tag{2.2}$$

Which is equivalent to simply imposing the inequalities that should be verified by any function in $F \in \mathcal{F}_{0,L}(\mathbb{R})$:

$$\begin{aligned} f(x) &\geq f(y) + f'(y)(y - z) \quad \forall x, y \in \mathbb{R} \\ |f'(x) - f'(y)| &\leq L\,|x - y| \quad \forall x, y \in \mathbb{R} \end{aligned}$$

The set of conditions 2.2 is satisfied for S with $L = 1$. But we can't find any smooth convex function that interpolates this set, because convexity implies non differentiability at x_2 as we can see in the figure 2.1.
But if we are interested $\mathcal{F}_{0,\infty}(\mathbb{R}^d)$ interpolation then in this case we can simply consider the convex function $F(x) = \frac{|x|}{2}$ which will interpolate and it turns out that the condition is sufficient and necessary.

$$f_i \geq f_j + g_j(x_i - x_j) \quad \forall i, j \in I$$

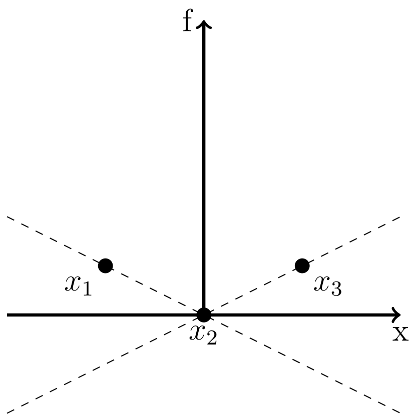


Figure 2.1: Interpolation of the set in 2.1

In this simple example we have illustrated an important and well known result about $\mathcal{F}_{0,\infty}(\mathbb{R}^d)$ which is stated in the following theorem:

Theorem 2.7. *The set $S = \{(x_i, f_i, g_i)\}_{i \in I}$ is $\mathcal{F}_{0,\infty}(\mathbb{R}^d)$-interpolable if and only if the following holds for all $i, j \in I$:*

$$f_i \geq f_j + g_j^T(x_i - x_j)$$

For a more general version of this theorem we refer to [Theorem 3.4 Tay17] which is also a reference for more examples that motivates the necessity of special conditions for smooth strongly convex interpolation.
Let us now go back to our main problem the one of smooth strongly convex interpolation. We are going to see how using all the results we stated in the first section, we will be able to prove the following theorem which is the main result of this section and a specific version of [Thorem 3.8 Tay17, p.51] for $\mathbb{R}^d$.

Theorem 2.8. *The set $S = \{(x_i, f_i, g_i)\}_{i\in I}$ is $\mathcal{F}_{\mu,L}(\mathbb{R}^d)$-interpolable if and only if the following inequalities holds for every $(i,j) \in I^2$*

$$f_i \geq f_j + g_j^T(x_i - x_j) + \frac{1}{2-\mu/L}\Big(\frac{1}{L}\|g_i - g_j\|_2^2 + \mu\|x_i - x_j\|_2^2 - 2\frac{\mu}{L}(g_i - g_j)^T(x_j - x_i)\Big) \tag{2.3}$$

For the exact proof of this theorem we refer to [Tay17]. As for this chapter we will only provide the ideas of the proof:

Proof. (Idea of the proof) Use theorem 2.4 to show the following (with $\mu \leq L$):

S is $\mathcal{F}_{\mu,L}(\mathbb{R}^d)-$interpolable $\Leftrightarrow$ $\{(x_i, f_i - \frac{\mu}{2}\|x_i\|_2^2, g_i - \mu x_i)\}_{i\in I}$ is $\mathcal{F}_{0,L-\mu}-$interpolable.

Then we show that it is also equivalent, by using theorem 2.6 and the first proposition in 2.5 with $x = x_i$ and $g_i = x^*$.

$$\{(g_i - \mu x_i, g_i^T x_i - f_i - \frac{\mu}{2}\|x_i\|_2^2, x_i)\}_{i\in I} \text{ is } \mathcal{F}_{1/(L-\mu),\infty}(\mathbb{R}^d)\text{-interpolable}$$

Finally use theorem 2.3 in order to prove that the preceding is also equivalent to:

$$\{(g_i - \mu x_i, \frac{L}{L-\mu}g_i^T x_i - f_i - \frac{\mu L}{2(L-\mu)}\|x_i\|_2^2 - \frac{\|g_i\|_2^2}{2(L-\mu)}, \frac{Lx_i}{L-\mu} - \frac{g_i}{L-\mu})\}_{i\in I}$$
is $\mathcal{F}_{0,\infty}$-interpolable

By theorem 2.6 and the convention $1/\infty = 0$, we can prove the following that the preceding is also equivalent to:

$$\{(\frac{Lx_i}{L-\mu} - \frac{g_i}{L-\mu}, \frac{\mu g_i^T x_i}{L-\mu} + f_i - \frac{\mu L\|x_i\|_2^2}{2(L-\mu)} - \frac{\|g_i\|_2^2}{2(L-\mu)}, g_i - \mu x_i)\}_{i\in I} \tag{2.4}$$
is $\mathcal{F}_{0,\infty}$-interpolable

Then we can show by 2.7 that 2.4 is equivalent to S being $\mathcal{F}_{\mu,L}(\mathbb{R}^d)$-interpolable and leads to the inequality of theorem 2.8. □

Then the following corollary provides a version of theorem 2.8 to the case of smooth convex interpolation. It is deduced by simply setting the strong convexity parameter μ to zero.

Corollary 2.9. *The set $S = \{(x_i, f_i, g_i)\}$ is $\mathcal{F}_{0,L}(\mathbb{R}^d)$-interpolable if and only if the following inequalities holds for all $(i, j) \in I^2$*

$$f_i \geq f_j + g_j^T(x_i - x_j) + \frac{1}{2L}\|g_i - g_j\|_2^2 \tag{2.5}$$

Notice that for $L = \infty$ we find theorem 2.7, and in the case of finite L we have the classical inequality for smooth convex functions:

$$f(y) \geq f(x) + \nabla f(x)^T(y - x) + \frac{1}{2L}\|\nabla f(y) - \nabla f(x)\|_2^2$$

Before we conclude this chapter we will give this small example to illustrate corollary 2.9.

Example 2.2. *If we go back to example 2.1 we can see that for the pair of indices $(2, 1)$ equation 2.5 becomes:*

$$0 \geq \frac{1}{2L}$$

And since $L \geq 0$ the set is not $\mathcal{F}_{0,L}(\mathbb{R})$-interpolable.
If we change the set S as follows:

$$(x_1, f_1, g_1) = \left(-1, \frac{1}{2}, -1\right) \quad (x_2, f_2, g_2) = (0, 0, 0) \quad (x_3, f_3, g_3) = \left(1, \frac{1}{2}, 1\right)$$

By corollary 2.9 the set will be $\mathcal{F}_{0,1}(\mathbb{R})$-interpolable. In fact the function interpolating will simply be $F(x) = \frac{x^2}{2}$

Conclusion

In this chapter some notations were introduced and we provided the main results for convex interpolation, Theorem 2.8 for necessary and sufficient conditions for a set to be interpolable by a smooth strongly convex function and its Corollary 2.9 for the case of L-smooth convex functions.
In the next chapter we will enter the main subject of the thesis by formulating the problem we are trying to solve properly.

Chapter 3

Performance estimation problem

In this thesis we will consider the usual unconstrained minimization problem:

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

The function f can be smooth convex or strongly convex, hence our problem P is an unconstrained convex optimization problem. For these problems it is common to use iterative methods that relies on first order approximation, first order methods. We assume that the method $\mathcal{M}$, does not have access to complete information on function f, hence it is provided with a first order oracle $\mathcal{O}_f$ that will provide $\mathcal{M}$ with values of the function and the gradient at a sequence of iterates. So the method will be considered a black box method. One important information about a method is its accuracy for a certain class of problems. Obviously, the exact accuracy may vary from a problem to another in the same class, for that reason we provide a so called *worst-case bound* on the accuracy which corresponds to the performance of the method on the worst problem of the class.
Many possible performance criteria can be considered, among them there is the objective function accuracy, norm of the gradient and distance to an optimal solution. We follow the notations used in [Tay17], and denote by $\mathcal{P}$ the performance criteria we will consider. the performance estimation problem in this context can be restated as follows: consider an unconstrained minimization problem P of a function f in the class of smooth convex or smooth strongly convex functions. Consider a first order black-box method $\mathcal{M}$ which is given oracle access to f $\mathcal{O}_f(x) = \{f(x), \nabla f(x)\}$. We want to find the worst-case bound on a certain performance criteria $\mathcal{P}$ in order to measure the performance of the method. In the literature (for example in [Nes98]), we can find for most of the first order methods an upper bound on the global worst-case. But a new vision by Drori and Teboulle [DT13] of the problem paved the way to compute the *exact* worst-case bound. Then the process of computing the worst-case bounds was pushed a step further by A.Taylor, F.Glineur and J.Hendrickx who created the pesto-toolbox [THG17a].
In this chapter we will provide a short review of this relatively new approach, and show the formulation of the performance estimation problem as a semidefinite program. Our problem will be a specific case of the general one, where we choose the classical gradient method (GM) as first order method to be analyzed using this approach. This section is based on chapter 4 of the PhD thesis dissertation [Tay17], which is the main reference for more details.

3.1 General formulation of the problem

As stated earlier we want to find an exact worst-case bound on some performance criterion. Denote by w the worst-case bound on a certain criterion $\mathcal{P}$.
As for upper bounds on w, in order to determine an exact bound, we need some assumptions on the starting point x_0. Suppose that we are given access to an optimal point x_*, then define the positive parameter R as an upper bound on the distance between our starting x_0 and the optimum such that we have:

$$\|x_0 - x_*\|_2 \leq R$$

The worst-case bound will thus depend on the functional class we consider $\mathcal{F}$, R and obviously on the method we choose $\mathcal{M}$ as well as the number of iterations denoted by N. We can obtain w by maximizing the criterion $\mathcal{P}$ on the class of function considered $\mathcal{F}$, note that this formulation is very general and at this stage no need to use specific class or method.

$$\begin{aligned} w(R, \mathcal{M}, \mathcal{F}, N, \mathcal{P}) = \max_{f, x_0, x_1, \cdots, x_N, x_*} & \mathcal{P}(\mathcal{O}_f, x_0, \cdots, x_N, x_*) \\ \text{such that} \quad & f \in \mathcal{F} \\ & x_* \text{ is optimal for } f \\ & \{x_1, \cdots, x_N\} \text{ generated from } x_0 \text{ by } \mathcal{M} \text{ with } \mathcal{O}_f \\ & \|x_0 - x_*\| \leq R \end{aligned} \tag{PEP}$$

Since one of the variables is a function $f \in \mathcal{F}$ the problem is infinite dimensional. So the first trick is to use the black-box assumption from which we have the following relationship between the iterates:

$$\begin{aligned} x_1 &= \mathcal{M}_1(x_0, \mathcal{O}_f(x_0)) \\ x_2 &= \mathcal{M}_2(x_0, \mathcal{O}_f(x_0), \mathcal{O}_f(x_1)) \\ &\vdots \\ x_N &= \mathcal{M}_N(x_0, \mathcal{O}_f(x_0), \mathcal{O}_f(x_1), \cdots, \mathcal{O}_f(x_{N-1})) \end{aligned} \tag{3.1}$$

Hence we can reformulate PEP by replacing variable f by the sequence of iterates $\{x_i\}$, the values of the function and the gradient at each iterate returned by the oracle $\{f_i\}$ and $\{g_i\}$ and we add (x_*, f_*, g_*) such that $g_* = 0$ from optimality condition. Using these variables we can reformulate (PEP) as the following finite optimization problem:

$$\begin{aligned} w^f(R, \mathcal{M}, \mathcal{F}, N, \mathcal{P}) = \max_{\{x_i, f_i, g_i\}_{i \in \{0, \cdots, N, *\}}} & \mathcal{P}(\{x_i, f_i, g_i\}_{i \in \{0, \cdots, N, *\}}) \\ \text{such that} \quad & \exists f \in \mathcal{F} \text{ s.t. } \mathcal{O}_f(x_i) = \{f_i, g_i\} \ \forall i \in \{0, \cdots, N, *\} \\ & \{x_1, \cdots, x_N\} \text{ generated from } x_0 \text{ by } \mathcal{M} \text{ using } \{f_i, g_i\}_{i \in \{0, \cdots, N-1\}} \\ & \|x_0 - x_*\|_2 \leq R \\ & g_* = 0 \end{aligned} \tag{finite-PEP}$$

Consider next a solution of (finite-PEP) $\{x_i, f_i, g_i\}$ with $i \in \{0, \cdots, N, *\}$ and a solution $f \in \mathcal{F}$ of (PEP) then we have the following relationship between the two of them

$$\{x_i, f_i, g_i\}_{i\in\{0,\cdots,N,*\}} \underset{\text{discretized by}}{\overset{\text{interpolated by}}{\rightleftarrows}} f$$

From this relation we can deduce that $w^f = w$ and both problems are equivalent. Now that we have defined the general formulation of the problem, we will consider instances of finite-PEP and adapt it to obtain the semidefinite program we will use in this thesis.

3.2 Formulation for smooth strongly convex functions

Let us consider now an instance of (finite-PEP) where the functional class $\mathcal{F}$ is the class of smooth convex functions defined on $\mathbb{R}^d$. The functional class $\mathcal{F}_{\mu,L}(\mathbb{R}^d)$ is invariant with respect to additive shifts in the functional values and translation in the domain (see [chapter 2 section 1 Nes98] for the special case of smooth convex functions) and the same holds for first order-methods, since they rely on the derivative. So without loss of generality we will consider that $x_* = f_* = 0$, an assumption that will be made later to simplify the proofs in chapter 5.
Now let us take a closer look at the interpolation constraint which is now requiring $\{x_i, f_i, g_i\}_{i\in\{0,\cdots,N,*\}}$ to be $\mathcal{F}_{\mu,L}(\mathbb{R}^d)$-interpolable. Recall from theorem 2.8 that this expression is equivalent to imposing that the set of interpolation inequalities holds for every $(i, j) \in \{0, \cdots, N, *\}^2$.

$$I(i,j) : f(x_i) - f(x_j) + g_i^T(x_j - x_i) + \frac{1}{2L}\|g_i - g_j\|_2^2 \leq 0 \tag{3.2}$$

Rewriting the interpolation inequalities as in (3.2) and denoting them by *I(i,j)* we can write the instance of (finite-PEP) as follows.

$$\begin{aligned} w^d(R,\mathcal{M},N,\mathcal{P}) = & \max_{\{x_i,f_i,g_i\}_{i\in\{0,\cdots,N,*\}}\in(\mathbb{R}\times\mathbb{R}^d\times\mathbb{R})^{N+2}} \mathcal{P}(\{x_i, f_i, g_i\}_{i\in\{0,\cdots,N,*\}}) \\ \text{such that} \quad & I(i,j) \text{ holds for every } \quad (i,j) \in \{0,\cdots,N,*\}^2 \\ & \{x_1,\cdots,x_N\} \text{ generated from } \; x_0 \text{ by } \; \mathcal{M} \text{ using } \; \{f_i, g_i\}_{i\in\{0,\cdots,N-1\}} \\ & \{x_*, f_*, g_*\} = \{\mathbf{0}, 0, \mathbf{0}\} \\ & \|x_0\|_2 \leq R \end{aligned} \tag{d-PEP}$$

Note that once again we chose the same notations as in [chapter 4 Tay17] since at the end we are referring to the same problem. Of course, since we have a special instance of (finite-PEP) as we mentioned earlier, we have $w^d(R, \mathcal{M}, N, \mathcal{P}) = w(R, \mathcal{M}, \mathcal{F}_{\mu,L}(\mathbb{R}^d), N, \mathcal{P})$. Because of the interpolation inequalities, and specifically the terms like $g_i^T(x_j - x_i)$, we are far from having a convex problem, and in order to do so we will start adding assumptions on the first-order method $\mathcal{M}$.
We require the method to be with **fixed step sizes**, a strategy where we fix in advance the step sizes or, the scalar coefficients that multiplies the gradient in the

iterations and we can rewrite 3.1 as follows (see[Definition 4.1 Tay17, p.75] for formal definition):

$$x_i = x_0 - \sum_{k=0}^{i-1} h_{i,k} g_k \quad \forall i \in \{1, \cdots N\} \tag{3.3}$$

The method will now be represented by the matrix $\mathbf{H} = \{h_{i,k}\}_{1 \leq i \leq N, 0 \leq k \leq N-1}$ with $h_{i,k} = 0$ if $i \geq k$. Then finally define vector P as follows:

$$P = \begin{bmatrix} g_0, g_1, \cdots, g_N, x_0 \end{bmatrix}$$

We can define a gram matrix, $G = P^T P \in \mathbb{S}^{N+2}$. In order to determine the exact expressions of the entries of G let us take a simple example:

Example 1. In the simple case of $N = 1$ we can see that matrix $G \in \mathbb{S}^3$, is the matrix of the dot products in $\mathbb{R}^d$ between g_i and x_0.

$$G = \begin{bmatrix} g_0^T g_0 & g_0^T g_1 & g_0^T x_0 \\ g_1^T g_0 & g_1^T g_1 & g_1^T x_0 \\ x_0^T g_0 & x_0^T g_1 & x_0^T x_0 \end{bmatrix}$$

Hence we can see that the entries of G are given by the following relation:

$$\begin{cases} G_{i,j} & = & g_i^T g_j & 0 \leq i, j \leq N \\ G_{N+1,j} & = & x_0^T g_j & 0 \leq j \leq N \\ G_{i,N+1} & = & g_i^T x_0 & 0 \leq i \leq N \\ G_{N+1,N+1} & = & x_0^T x_0 & \end{cases}$$

We can rewrite 3.3 using P. Let us first check how we can do it in our previous simple example:

Example 1 (Continued)**.** For one iteration relation 3.3 reduces simply to $x_1 = x_0 - h_{1,0} g_0$. If we define the following vector $h_1 \in \mathbb{R}^3$ such that:

$$h_1^T = \begin{bmatrix} -h_{1,0} & 0 & 1 \end{bmatrix}$$

We can rewrite $x_1 = P h_1$

In a similar fashion, for N iterations and all $i \in \{1, \cdot, N\}$ we can write $x_i = P h_i$ and we will define $h_i \in \mathbb{R}^{N+2}$ as follows:

$$h_i^T = \begin{bmatrix} -h_{i,0} & -h_{i,1} & \cdots & -h_{i,i-1} & 0 & \cdots & 0 & 1 \end{bmatrix}$$

as for x_* we will simply take h_*, a vector of zeros. Then as explained in [Tay17], we can simplify the constrains of d-PEP as follows, where will denote the vector of the canonical basis of $\mathbb{R}^{N+2}$ by $u_i = e_{i+1}$ and u_* vector of zeros.

1. **Interpolation constraint**: We start by rewriting the inequalities for each i and j in $\{0, \cdots, N, *\}$, using these relations:

$$g_i^T x_j = u_j^T G h_i \quad \|x_i - x_j\|_2^2 = (h_i - h_j)^T G (h_i - h_j) \quad \|g_i - g_j\|_2^2 = (u_i - u_j)^T G (u_i - u_j)$$

Then replace all but the terms f_i and f_j by the trace of the product of G and the matrix A_{ij} defined in [section 4.2.3 Tay17, p.76] as follows:

$$2A_{ij} = \frac{L}{L-\mu}(u_j(h_i-h_j)^T + (h_i-h_j)u_j^T) + \frac{1}{2(L-\mu)}(u_i-u_j)(u_i-u_j)^T \\ + \frac{\mu}{L-\mu}(u_i(h_j-h_i)^T + (h_j-h_i)u_i^T)\frac{L\mu}{L-\mu}(h_i-h_j)(h_i-h_j)^T$$

Then for each (i,j) we can rewrite the interpolation conditions as follows:

$$f_i \geq f_j + Tr(GA_{ij})$$

2. **Distance to optimal** consider the square of the distance and $x_* = 0$ we get:

$$\|x_0\|_2^2 = x_0^T x_0 = u_{N+1}^T G u_{N+1} = Tr(\underbrace{u_{N+1}u_{N+1}^T}_{A_R} G) = Tr(GA_R) \leq R^2$$

Now, in order to have equivalence between the two problems, we should add the constrained stating that the matrix G should be semidefinite positive. In that way the point x_0 and g_i of any solution of d-PEP can be expressed into a Gram matrix of rank d since all the vectors are in $\mathbb{R}^d$ and every Gram matrix solution of our new problem can give us the values of x_0 and the gradients using matrix decomposition. We may also impose that the rank of G should be at most d.
The only thing left now is to express the objective function. Let us take an example of some common performance criterion $\mathcal{P}$.

Example 3.1. *suppose we perform N iterations of a constant step size first order method and suppose $f_* = x_* = 0$.*

1. *If we choose $\mathcal{P} = f(x_N) - f(x_*) = f_N - f_*$ the objective function accuracy. Then we can rewrite $\mathcal{P}$ as follows:*

$$\mathcal{P} = \begin{bmatrix} 0 & \cdots & 0 & 1 \end{bmatrix} f = b^T f$$

2. *If we choose $\mathcal{P} = \|x_N - x_*\|_2^2$ distance to optimal solution. If we define the matrix $C = h_N h_N^T \in \mathbb{S}^{N+2}$ we can write $\mathcal{P} = Tr(CG)$.*

From this example observe that we were able to write the objective function as sum of the entries of f and G. Combining everything we get the following semidefinite program presented in [therorem 4.2 Tay17, p.77]. We will only provide the version with $N < d-2$ where the size of $G \in \mathbb{S}^{N+2}$ is smaller then d so the rank is also smaller then d and to simplify the notations we will define the set $I = \{0, \cdots, N\}$.

$$\begin{aligned} w_{\mu,L}^{sdp}(R,H,N,b,C) = & \max_{f\in\mathbb{R}^{N+1},\, G\in\mathbb{S}^{N+2}} b^T f + Tr(CG) \\ \text{such that} \quad & f_j - f_i + Tr(GAij) \leq 0 \qquad i,j \in I \\ & Tr(GA_R) - R^2 \leq 0 \\ & G \succeq 0. \end{aligned} \tag{sdp-PEP}$$

Now let us derive the Lagrangian dual of (sdp-PEP). We start by defining the dual variables $\lambda_{i,j} \geq 0$ for $i,j \in I$ associated with the constraint related to the interpolation inequalities and $\tau \geq 0$ associated with the constraint on the distance between x_0 and x_*. Then we can write the Lagrangian function, where we recall that on the space of matrix $\langle A, B\rangle = Tr(A^T B)$

$$\mathcal{L}(\lambda_{i,j}, \tau, G, f) = b^T f + \langle C, G\rangle - \sum_{i,j\in I} \lambda_{i,j}(f_j - f_i + \langle G, A_{i,j}\rangle) - \tau(\langle G, A_R\rangle - R^2)$$

Then we can deduce the Lgrangian dual function g as follows:

$$\begin{aligned} g(\lambda_{i,j}, \tau) &= \max_{f, G\succeq 0} \mathcal{L}(\lambda_{i,j}, \tau, G, f) \\ &= \max_{f, G\succeq 0} \tau R^2 - \left\langle (\tau A_R - C + \sum_{i,j\in I} \lambda_{i,j} A_{i,j}), G \right\rangle \\ &\quad + (b - \sum_{i,j\in I} \lambda_{i,j}(u_j - u_i))^T f \end{aligned}$$

Then we can rewrite the Lagrangian dual problem as $min g(\lambda_{i,j}, \tau)$ and we get:

$$\begin{aligned} \min_{\lambda_{ij}, \tau} \tau R^2 \quad \text{such that} \quad \tau A_R - C + \sum_{i,j\in I} \lambda_{ij} A_{ij} &\succeq 0 \\ b - \sum_{i,j\in I} \lambda_{ij}(u_j - u_i) &= 0 \\ \lambda_{ij} &\geq 0 \quad i,j \in I \\ \tau &\geq 0 \end{aligned} \tag{d-sdpPEP}$$

The problem (d-sdpPEP) will be very useful as stated in the main reference of this section and as we will see later in the next chapters, because of [Theorem 4.7 Tay17, p.80] we have grantee on the existence of an explicit proof for any worst-case function.
Notice also that in [DT13, Problem G] is a special case of (sdp-PEP), where the method is the gradient method and the class of functions is the class of smooth convex functions ($\mu = 0$), because they do not consider all the interpolation inequalities and the dual they propose is a restricted version of d-sdpPEP where we can consider that all the $\lambda_{i,j}$ associated with the unconsidered interpolation inequalities are fixed to zero, Specifiaclly as we ca read in [Tay17], they fix $\lambda_{i,j} = 0$ for $i+1 \neq j$ or $i \neq 0$. This formulation enabled the development of a new procedure to study first order method for a specific class of problems and first order methods. Since sdp-PEP is a semi definite program, we can solve it numerically and the answers will give us an exact worst- case bound for the chosen performance criteria. The methodology has been made accessible by the pesto-toolbox [THG17a] which is a matlab toolbox freely available on github [1] based on YALMIP [LÖ4] modeling language and an SDP solver like SeDumi [Stu99] or Mosek [Mos10]. For more detailed explanation on how to use the toolbox you can check the user-guide available with the toolbox. Note also that in this work we only showed the formulation for unconstrained problems, but the work has been extended to include other framework (see [Chapter 5 Tay17]). A general methodology can be deduced to use the toolbox:

[1] https://github.com/AdrienTaylor/Performance-Estimation-Toolbox

- **Step1** Choose the class of functions smooth strongly convex (provide μ and L) or smooth convex (provide L) you want to study as well as the class of problems.
- **Step2** Choose the first order black box method with fixed step sizes you want to use.
- **Step3** Choose the performance criteria you want to consider.
- **Step4** Provide the number of iterations N and solve the corresponding sdp program to obtain the exact worst-case bound. The dual solution will help you provide an explicit poof of the exact bound you found.

This friendly user toolbox can thus be tailored for a wide range of convex optimization problems and first order methods. In this thesis we will see how using the toolbox we can guess and prove exact worst-case bounds on the classical gradient method with variable fixed step sizes.

3.3 PEP for the gradient method

In this section, we will choose the simplest first order black box method, the *Gradient Method* with fixed normalized step sizes that will be denoted by GM.

We assume that we want to use this classical method to solve P where the function f to be minimized is as mentioned earlier smooth convex or smooth strongly convex. The scheme of GM is presented here after, where N denotes as usual the number of iterations.

$$\begin{aligned} &\text{Initialization:} \quad x_0 \in \mathbb{R}^d. \\ &\text{For } i = 1, \cdots, N \\ &\qquad x_i = x_{i-1} - \frac{h_i}{L}\nabla f(x_{i-1}) \end{aligned} \tag{GM}$$

Obviously the scalars h_i represents the step sizes and since they are divided by L they are normalized and we will note $\gamma_i = \frac{h_i}{L}$ with $i \in \{1, \cdots, N\}$. We can easily rewrite the scheme as in equation 3.3 by fixing $h_{i,k} = \gamma_{k+1}$. Hence, we can denote the method by the lower triangular matrix H^{GM} as follows:

$$H^{GM} = \begin{bmatrix} \gamma_1 & 0 & \cdots & 0 \\ \gamma_1 & \gamma_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ \gamma_1 & \gamma_2 & \cdots & \gamma_N \end{bmatrix} \tag{3.4}$$

We want to study the objective function accuracy after N iterations of GM with fixed variable step sizes, so we can rewrite $\mathcal{P}$ as in example 3.1 and denote the vector b by b^{obj}.

In the literature, we have some estimates on the worst-case accuracy of GM when applied to unconstrained minimization of smooth convex and strongly convex functions see for example [Section 2.1.5 Nes98]. As we mentioned earlier, in their work [DT13], Drori and Teboulle studied the exact worst-case bound on the objective

function accuracy in the case of smooth convex functions and for constant step sizes, which was also done and extended in [Section 4.3.1 Tay17, p.86] using the methodology described in the previous section.
Our goal is to study the performance of GM with fixed variable step sizes, and generalize the conjectures already found for the case of constant step sizes. Let us clarify once and for all the difference between fixed constant step sizes and fixed variable step sizes.

- **Constant fixed step sizes**, where all the steps are **the same** and fixed beforehand.
- **Variable fixed step sizes**, where we fix in advance the step sizes **arbitrarily** for each iteration.

The difficulty relies on the fact that the toolbox will return numerical values for specific values of the parameters and it is up to the user to determine a closed form expression of the worst-case bound.
We defined a possible approach for finding the bound and proving it, which will be the strategy adopted in this work:

- **Step1**:Start by fixing the parameters needed to solve sdp-PEP. We choose first the case of smooth convex ($\mu = 0$) or strongly convex ($\mu > 0$) and provide the value of the lipschitz constant L. Then provide the bound on the distance to the optimum R and the number of iterations N. Finally choose a number of experiments or the number of variations (v) of steps you want to consider and define a set $V = \{1, \cdots, v\}$. For each $j \in V$ define H_j^{GM}.
- **Step2**: Solve sdp-PEP for all $j \in V$ and store the outputs in a vector W. Store also the values of all the dual variables corresponding to the interpolation inequalities ($\lambda_{i,j}$) in d-sdpPEP in a vector Λ.
- **Step3**: From vector W try to guess a function b_N^{μ} such that for any H^{GM} we have:
$$b^{N,\mu}(R, L, H^{GM}) = w_{\mu,L}^{sdp}(R, H^{GM}, N, b_{obj}, 0)$$
- **Step4**: From the dual variables gathered in Λ, try to provide a general proof for the function $b^{N,\mu}$ found in step 3 using valid inequalities.

We provide an overview of the procedure in GM-Proc.

$$\begin{array}{l}
(\mathbf{1})\ \text{Initialization: fix} \quad \mu, L, R, N \text{ and } \{H_i^{GM}\}_{i\in V} \\
(\mathbf{2})\ \text{For} \quad i \in V \\
\qquad w_i = w_{\mu,L}^{sdp}(R, H_i^{GM}, N, b_{obj}, 0) \\
\qquad \Lambda_i = \{\lambda_{ij}\} \\
(\mathbf{3})\ \text{From } W = \{w_i\}_{i\in V} \text{ guess } b^{N,\mu} = w_{\mu,L}^{sdp}(R, H^{GM}, N, b_{obj}, 0). \\
(\mathbf{4})\ \text{From } \Lambda = \{\Lambda_i\}_{i\in V} \text{ provide a proof for } b^{N,\mu}.
\end{array} \tag{GM-Proc}$$

Notice that in the data gathering step or step two of GM-Proc the problems will be solved for $R = 1$ and $L = 1$ and different values of step sizes. This approach is justified by the homogeneity of the optimal values of sdp-PEP with respect to L

and R presented in [section 4.2.5 Tay17] and for the case of the worst-case bound on the objective function accuracy we have the following relationship:

$$w^{sdp}_{\mu,L}(R, H/L, N, f(x_N) - f_*) = LR^2 w^{sdp}_{\mu,1}(1, H, N, f(x_N) - f_*) \tag{3.5}$$

Where the matrix H is the representation of the method before normalizing the steps (dividing the h_i by L) and is linked to H^{GM}, that we defined in (3.4), by the following relationship:

$$H = LH^{GM} = \begin{bmatrix} h_1 & 0 & \cdots & 0 \\ h_1 & h_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ h_1 & h_2 & \cdots & h_N \end{bmatrix}$$

So that in step three we will find the exact worst-case bound on the objective function accuracy $b^{N,\mu}$ such that :

$$b^{N,\mu}(h_1, \cdots, h_N) = w^{sdp}_{\mu,1}(1, H, N, f(x_N) - f_*) \tag{3.6}$$

Then we simply multiply by LR^2 to get an exact worst-case bound for any L and R. This is a direct conclusion from (3.5).
Using this procedure we obtain for a specific number of iterations (N) an expression of the exact worst-case bound that you can prove using the dual variables. Then the idea is to apply GM-Proc for different number of steps in order to guess a proof for any number of iterations of GM.
The process of guessing the formulas, step three in GM-Proc can be done using some tricks like polynomial or rational interpolation, but the intuition based on smaller values of N and the special case of constant step sizes is the best approach based, on my experience in this work.
As for the final steps for the proof, it is not that simple to determine a proof for general N specially when N becomes large enough. A possible approach is to try a proof without guessing a closed form for the variables lambda but the algebra becomes difficult as we will see in the final chapter of this thesis.
Maybe one possible improvement for the toolbox is to add new features that makes steps three and four easier. This is completely possible and we will do it a little bit for the fourth step.

We will also add that we will need to confront our guesses with numerical values, and to this aim we take a precision of 10^{-6} as a tolerance based on which we will accept the guess numerically or not.

Summary

In this chapter the first main results are the formulation perfomance estimation problem PEP into a finite version finite-PEP using the black box property of the method $\mathcal{M}$.

$$\boxed{\text{PEP}} \xrightarrow[\text{property}]{\text{Black Box}} \boxed{\text{finite-PEP}}$$

Then taking one specific class of problems which is the class of smooth strongly convex functions $\mathcal{F}_{\mu,L}(\mathbb{R}^d)$ and using convex interpolation we can reformulate the problem as d-PEP. Finally using a Gram matrix and restricting to fixed step sizes first order methods we get sdp-PEP.

$$\boxed{\text{finite-PEP}} \xrightarrow[\text{Convex interpolation}]{\mathcal{F}_{\mu,L}(\mathbb{R}^d)} \boxed{\text{d-PEP}} \xrightarrow[\text{Gram matrix}]{\text{Fixed step sizes}} \boxed{\text{sdp-PEP}}$$

After that we were able to formulate the performance estimation of the classical gradient method with variable step sizes and defined the procedure that we will use in the next chapters GM-Proc. In the next chapter we will mainly focus on **stage 3** of (GM-Proc) and in the final chapter we will work a little bit on **stage 4**.

Chapter 4

Identification of worst-case bounds

Introduction

Performance of the gradient method has been analyzed in the case of fixed step and we can find in the [Conjecture 1 section 3.2 DT13] the following conjecture for smooth convex functions:

Conjecture 4.1. *Any sequence of iterates $\{x_i\}$ generated by the gradient method GM with constant normalized step size $0 \leq h \leq 2$ on a smooth convex function $f \in \mathcal{F}_{0,L}(\mathbb{R}^d)$ satisfies*

$$f(x_N) - f_* \leq \frac{LR^2}{2} \max\left(\frac{1}{2Nh+1}, (1-h)^{2N}\right)$$

This conjecture has been verified in [Tay17] numerically by solving sdp-PEP and generalized for strongly convex functions:

Conjecture 4.2. *[Conjecture 4.10 Tay17, p.91] Any sequence of iterates $\{x_i\}$ generated by the gradient method GM with constant normalized step sizes $0 \leq h \leq 2$ on a smooth strongly convex function $f \in \mathcal{F}_{\mu,L}(\mathbb{R}^d)$ satisfies*

$$f(x_N) - f_* \leq \frac{LR^2}{2} \max\left(\frac{\kappa}{(\kappa-1)+(1-\kappa h)^{-2N}}, (1-h)^{2N}\right)$$

Where $\kappa = \frac{\mu}{L}$ is the inverse of the condition number.

In this chapter we will study the convergence of the gradient method GM with **variable** normalized step sizes. We will start by applying GM-Proc for two then three steps of GM in order to determine the exact worst-case bound. Then based on our conjectures and observations we will deduce some conclusion on the exact worst-case bound for N steps, for both smooth convex and strongly convex functions.

All the conjectures and results relies on an extensive usage of Pesto-toolbox [THG17a] and the numerical resolutions of sdp-PEP are done using SeDumi [Stu99].

4.1 One step of GM

From the two conjectures we can deduce directly the bound for one step of GM in the case of smooth convex functions ($b^{1,0}$) and strongly convex functions ($b^{1,\mu}$) by setting N to one and we get:

$$b^{1,0}(h_1) = \frac{LR^2}{2} \max\left(\frac{1}{1+2h_1}, (1-h_1)^2\right) = \frac{LR^2}{2} \max(b_1^{1,0}, b_2^{1,0}) \tag{4.1}$$

$$b^{1,\mu}(h_1) = \frac{LR^2}{2} \max\left(\frac{\kappa}{(\kappa-1)+(1-\kappa h_1)^{-2}}, (1-h_1)^2\right) = \frac{LR^2}{2} \max(b_1^{1,\mu}, b_2^{1,\mu}) \tag{4.2}$$

One possible proof for this exact worst-case bound on the objective function accuracy can be found in appendix A of chapter 4 in [Tay17].

For one step $b^{1,\mu}$ is the maximum of two simple functions. The values of $h_1 \in [0, 2]$ such that $max(b_1^{1,\mu}, b_2^{1,\mu}) = b_1^{1,\mu}$ will be considered "small" and when h_1 is small we will be in the small step sizes regime. Similarly, h_1 will be considered "big" when $\max(b_1^{1,\mu}, b_2^{1,\mu}) = b_2^{1,\mu}$ and we will be in those cases in the big step sizes regime. In the following example we illustrate these notations for the case of smooth convex functions.

Example 4.1. *In the case of one step of GM and smooth convex functions we have for $L = R = 1$:*

$$b_1^{1,0} = \frac{1}{1+2h_1} \qquad b_2^{1,0} = (1-h_1)^2 \qquad b^{1,0} = \frac{1}{2} max(b_1^{1,0}, b_2^{1,0})$$

In figure 4.1 we represented the functions $b_1^{1,0}$, $b_2^{1,0}$, their maximum and the optimal step size, optimal in the sense that it achieves the lowest worst-case denoted by $h_{opt}(1)$. As we can see graphically $h_{opt}(1)$ is on the intersection of the two functions and solving $b_1^{1,0} = b_2^{1,0}$ we get $h_{opt}(1) = 1.5$.

So for one step when $h_1 \leq h_{opt}(1)$ we will be in the small step sizes regime because $max(b_1^{1,0}, b_2^{1,0}) = b_1^{1,0}$ and for $h_1 \geq h_{opt}(1)$ we are in the big step sizes regime.

Now that we have analyzed one step, we have initialized our iterative procedure and we can study two steps.

4.2 Two steps of GM

For smooth convex and strongly convex functions we started by constituting a data set by solving the corresponding performance estimation problem (sdp-PEP) for different length of step sizes (h_1 and h_2) and different values of strong convexity parameter (μ) in the case of smooth strongly convex functions. We fixed $N = 2$ since we want to guess the worst-case bound for two steps.
As we mentioned in the previous chapter,the process of finding worst-case bounds is heavily based on our intuition from one step of GM, discussed in the previous section and on conjecture 4.1 and 4.2 for special cases where $h_1 = h_2$, which constituted a sanity check. We start by highlighting some preliminary observations that can be deduced from one iteration and eventually from the conjectures for N constant step sizes.

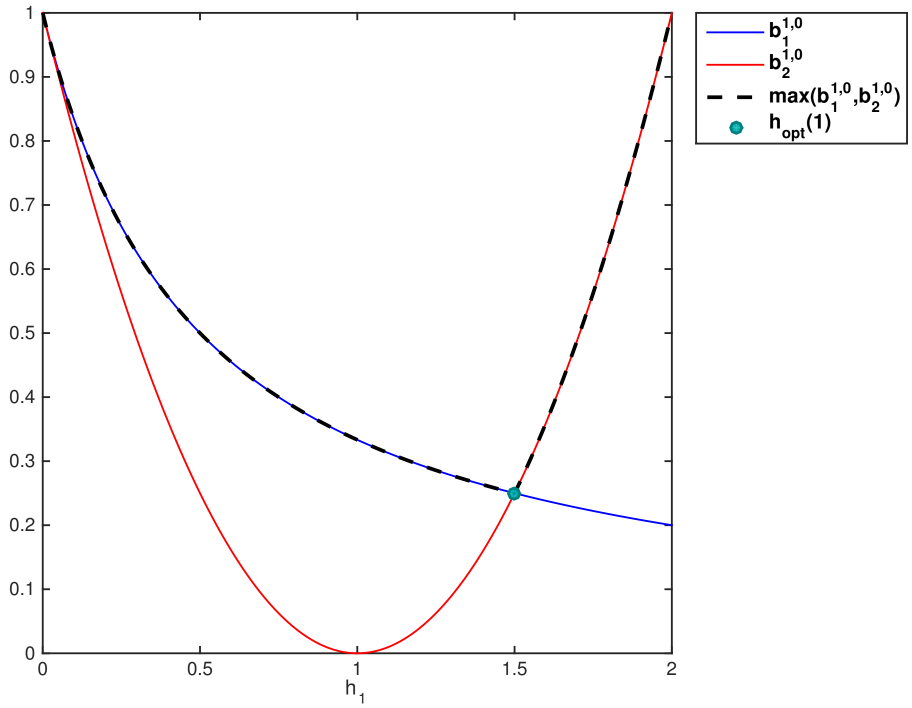


Figure 4.1: Plot of the functions $b_1^{1,0}$, $b_2^{1,0}$ their maximum and the optimal step size for one step of GM $h_{opt}(1)$

Preliminary assumptions:

Just by looking at the shape of the functions (4.1) and (4.2) involved in the bound for one step, and encouraged by the conjectures for N steps of GM we can make the following assumption on the worst-case bound we are trying to find $b^{2,\mu}$.

Assumption 4.1. *The exact worst-case bound on the objective function accuracy for two steps of GM $b^{2,\mu}$ has the following shape:*

$$b^{2,\mu}(h_1, h_2) = \frac{1}{2}\max(\{b_i\}_{i\in J})$$

Where J is a set of indices and b_i are simple functions (rational or polynomial) of h_1 and h_2.

From example 4.1, it seems like we obtained two regimes and $J = \{1, 2\}$ because we could choose h_1 as "small" or "big". In the case of constant step sizes we have the maximum of two functions as we can see in conjectures 4.1 and 4.2 and we have one choice for the step length h either small or big. From this discussion we can make the following assumption on the number of functions involved in the maximum and the set of indices J for two steps of GM.

Assumption 4.2. *For two variable steps of GM we will have **at most** four types of combination of step sizes h_1, h_2 which are:*

$$(S, S) \qquad (B, B) \qquad (B, S) \qquad (S, B)$$

Where S stands for small and B for big. So we will possibly have $J = \{1, 2, 3, 4\}$ and $b^{2,\mu} = \frac{1}{2}max(b_1^{2,\mu}, b_2^{2,\mu}, b_3^{2,\mu}, b_4^{2,\mu})$

At this stage it is hard to say without additional proofs if their will be exactly four regimes, but we are sure that we will have at least two.
Furthermore, notice that as κ or μ tends to zero the bound in conjecture 4.2 converges to the one in conjecture 4.1. This may not be trivial so let us prove the following for all N and h.

$$\lim_{\kappa\to 0}\frac{\kappa}{(\kappa-1)+(1-\kappa h)^{-2N}}=\frac{1}{1+2Nh} \tag{4.3}$$

Proof. We start by introducing the following the variable q defined as follows:

$$q=\frac{1}{(1-\kappa h)^2}$$

Clearly when κ tends to zero q tends to one.

$$\begin{aligned}\left(\frac{\kappa}{(\kappa-1)+(1-\kappa h)^{-2N}}\right)^{-1} &= \frac{\kappa-1+q^N}{\kappa}\\ &=1+\frac{1}{\kappa}(q^N-1)\\ &=1+\frac{1}{\kappa}\left(\frac{q^N-1}{q-1}\right)(q-1)\\ &=1+\frac{1}{\kappa}\left(\frac{1}{q}\sum_{i=1}^{N}q^i\right)(q-1)\\ &=1+\frac{1}{\kappa}\left(\frac{1}{q}\sum_{i=1}^{N}q^i\right)\left(\frac{2\kappa h-\kappa^2h^2}{(1-\kappa h)^2}\right)\\ &=1+\left(\sum_{i=1}^{N}q^i\right)(2h-\kappa h^2)\end{aligned}$$

Now as κ tends to zero $\sum_{i=1}^{N}q^i$ converges to N such that we have :

$$\lim_{\kappa\to 0}\left(\frac{\kappa}{(\kappa-1)+(1-\kappa h)^{-2N}}\right)^{-1}=(1+2Nh)$$

From which we can conclude 4.3. □

Now similarly we can prove, that for one step of GM we have the following:

$$\lim_{\kappa\to 0}b_i^{1,\mu}=b_i^{1,0}\qquad\forall i\in I$$

Notice also that the function $(1-h_1)^2$ is common for both convex and strongly convex functions. The same thing holds in the case of constant step sizes, so for the function $(1-h)^{2N}$. Based on these observations, we may assume that this convergence will hold for two steps.

Assumption 4.3. *Consider two variable step sizes of GM. Then we have the following link between the exact worst-case on the objective function accuracy for smooth convex functions and smooth strongly convex functions we have for all $i\in J$*

$$\lim_{\mu\to 0}b_i^{2,\mu}=b_i^{2,0}\quad\forall(h_1,h_2)\in[0,2]$$

This last assumption (4.3) suggest also that their will be the same set J, or the same number of regimes for $\mu = 0$ and for any μ. So guessing the worst-case bound for smooth convex functions, will simplify the process of guessing the bound for smooth strongly convex functions as we will see later in this section. Armed with all these assumptions we will see how we can deduce the exact worst-case bound for two steps of GM in step three of GM-Proc.

4.2.1 Conjecture on smooth convex functions

The data set is generated for two hundred one values of h_1 and h_2 chosen between zero and two such that:

$$(h_1, h_2) \in \{i/100\}^2_{i=0,\cdots,200}$$

From assumption 4.1 we stated by multiplying the data set by two and by doing so we are just focused on guessing $\max(b_i)$ or $2b^{2,0}$. The data set will be denoted by *NumSdpPep*. In table 4.1 you can find a sample from *NumSdpPep* where we chose eight values for h_1 and h_2. We will use this sample data as a working example to illustrate the procedure we followed to guess the worst-case bound. Of course in practice we worked on the whole *NumSdpPep* directly.

h_1 \ h_2	0	0.5	0.8	1	1.2	1.5	1.8	2
0	1.0000	0.5000	0.3846	0.3333	0.2941	0.2500	0.6400	1.0000
0.5	0.5000	0.3333	0.2778	0.2500	0.2273	0.2000	0.2844	0.4444
0.8	0.3846	0.2778	0.2381	0.2174	0.2000	0.1786	0.1975	0.3086
1	0.3333	0.2500	0.2174	0.2000	0.1852	0.1667	0.1600	0.2500
1.2	0.2941	0.2273	0.2000	0.1852	0.1724	0.1562	0.1429	0.2066
1.5	0.2500	0.2000	0.1786	0.1667	0.1562	0.1429	0.1600	0.2500
1.8	0.6400	0.3200	0.2462	0.2133	0.1882	0.1600	0.4096	0.6400
2	1.0000	0.5000	0.3846	0.3333	0.2941	0.2500	0.6400	1.0000

Table 4.1: Sample of Numerical values returned by solving sdp-PEP for h_1 and h_2 in $\{0\,0.5\,0.8\,1\,1.2\,1.5 1.8\,2\}$

The strategy will be to start by guessing one of the simple functions b_i and to compute compute 201×201 matrix representing the value of b_i for each h_1 and h_2 considered. Then we compute the matrix of the maximum of the functions guessed until now and compare it to *NumData*. If the six first decimals are identical then we will consider that we have a fit. One good way of representing this is to create a matrix of Errors denoted by E and defined as follows:

$$E = |max(b_i) - NumSdpPep| \quad \text{such that} \quad E_{i,j} = 0 \text{ if } E_{i,j} \leq 1e-6 \tag{4.4}$$

By assumption 4.2 we may expect to have four regimes hence the maximum of four simple functions. In order to get a first glimpse we decided to observe the contour plot of *NumSdpPep*. In figure 4.2, we can see this contour plot and it seems like, even tho we do not have any idea of what the regimes looks like, it is safe to assume that their will be four regimes. Even though we do not know until now, when h_1 and h_2 will be considered small (S) or big (B) since the functions involved in the maximum are unknown, from conjecture 4.1 we can see that in the

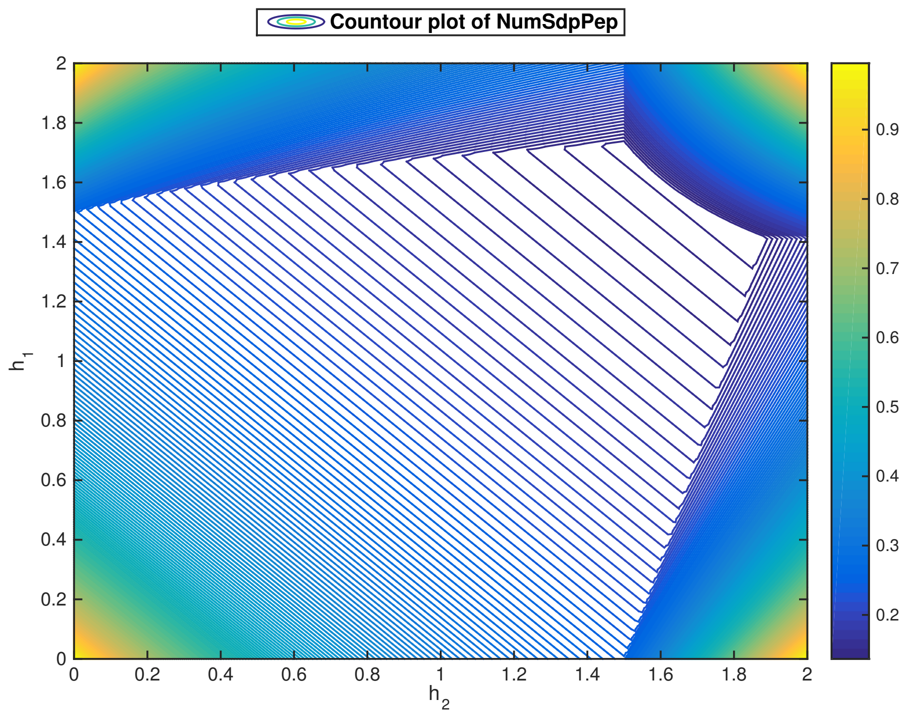


Figure 4.2: Contour plot of the numerical data in $NumSdpPep$ for the 201 values of h_1 and h_2 between zero and two.

case where $h_1 = h_2 = h$ we have only two possible regimes (S, S) and (B, B) and we will have the maximum of two functions. For (S, S) and constant step sizes it seems like we took the function $b_1^{1,0}$ and replaced h_1 by $h_1 + h_2 = 2h$. So it seems like the function for (S, S) that will generalize the one for constant step sizes denoted by $b_1^{2,0}$ will have the following shape:

$$\boxed{b_1^{2,0}(h_1, h_2) = \frac{1}{1 + 2h_1 + 2h_2}} \tag{4.5}$$

As for (B, B), also looking at the case of two constant step sizes given by conjecture 4.1, it seems like we can obtain the function for the second regime by multiplying $b_2^{1,0}(h)$ by $b_2^{1,0}(h)$. So the function generalizing for two variable step sizes in the (B, B) regime seems to be given by $b_2^{2,0}(h_1, h_2) = b_2^{1,0}(h_1)b_2^{1,0}(h_2)$. So we conjecture the following shape for b_2:

$$\boxed{b_2^{2,0}(h_1, h_2) = (1 - h_1)^2(1 - h_2)^2} \tag{4.6}$$

Now, as explained earlier we computed the maximum of the two functions we have until know $max(b_1^{2,0}, b_2^{2,0})$ for each of the 201 values of h_1 and h_2 that we stored in a matrix. we computed the matrix of error E as defined in 4.4 and figure 4.3a presents a spy plot of the matrix E which we found convenient for showing the number of problems H that did not match our current guess numerically. From 4.3a we can clearly see that these two function are not enough. We thus have a new evidence in favor of our assumption on the number of regimes (assumption 4.2) as we have clearly eliminated two regions, (S, S) and (B, B) and their is two regions left.

In table 4.2 we have showed the data fitted by the function $b_1^{2,0}$ and $b_2^{2,0}$ respectively in blue and red. We have also replaced them by their relative error, which as we can see is smaller then $1e-06$. As for the non fitted data we kept them as in table 4.1. Looking at the remaining non fitted data in table 4.2, we can clearly see the two remaining regimes. On the upper right corner we can see the (S, B) regime and on the lower left corner the (B, S) regime.

h_1 \ h_2	0	0.5	0.8	1	1.2	1.5	1.8	2
0	1e-09	1e-10	1e-08	3e-09	7e-10	3e-10	9e-10	2e-08
0.5	2e-10	5e-09	2e-10	3e-10	1e-10	2e-10	0.2844	0.4444
0.8	3e-09	1e-10	9e-11	2e-09	1e-10	1e-09	0.1975	0.3086
1	2e-10	7e-11	4e-10	1e-10	1e-10	5e-10	0.1600	0.2500
1.2	1e-10	4e-11	1e-10	2e-09	2e-10	1e-09	1e-09	0.2066
1.5	4e-09	1e-10	2e-10	2e-10	1e-10	1e-10	1e-10	1e-10
1.8	3e-08	0.3200	0.2462	0.2133	0.1882	1e-07	1e-10	1e-10
2	3e-08	0.5000	0.3846	0.3333	0.2941	1e-08	5e-10	1e-08

Table 4.2: Error between the Sample of $NumSdpPep$ presented in table 4.1 and $max(b_1^{2,0}, b_2^{2,0})$. The data fitted by $b_1^{2,0}$ is represented in blue and the data fitted by $b_2^{2,0}$ is represented in red and replaced by their relative error.

The first two functions $b_1^{2,0}$ and $b_2^{2,0}$ seemed to be deduced directly from the preliminary observations on one step of GM and conjecture 4.1. For the two others the task will be more difficult since the constant step sizes case is not included in the two regimes that are left. The only information that could help is the one step regime. Let us see how we can exploit it. Observe first that for $h_1 = 0$ we have:

$$max(b_1^{2,0}(0, h_2), b_2^{2,0}(0, h_2)) = max(b_1^{1,0}(h2), b_2^{1,0}(h_2))$$

So it seems like we get the bound for one step. One similar observation can be done in the case where $h_2 = 0$. This implies the following on the functions $b_3^{2,0}$ representing (B, S) regime and $b_4^{2,0}$ representing (S, B) regime:

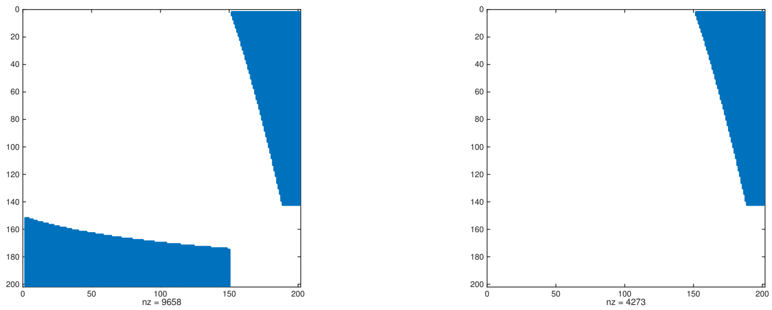
$$b_3^{2,0}(h_1, 0) = (1 - h_1)^2 \qquad b_4^{2,0}(0, h_2) = (1 - h_2)^2$$

The next thing we did is to isolate the data in $NumSdpPep$ that did not match the function $\max(b_1^{2,0}, b_2^{2,0})$ in order to take a closer look. In table 4.5 we have showed the samples of the data corresponding to (B,S) to the left and the samples corresponding to (S,B) to the right. This is a sample of what we obtained when separating them in $NumSdpPep$.

The next thing we did was multiplying the extracted data by $(1 - h_1)^2$ for (B, S) region and by $(1 - h_2)^2$ for the (S, B) region. We observed that we lose dependence on h_1 and h_2, since for (B, S) $((S, B))$ for all h_1 (resp. h_2) the values obtained are the same, at least when comparing the first six decimals, and only vary with h_2 (resp. h_1).

From this observation we can deduce the following:

$$b_3^{2,0}(h_1, h_2) = \theta(h_2)(1 - h_1)^2 \qquad b_4^{2,0}(h_1, h_2) = \alpha(h_1)(1 - h_2)^2 \tag{4.7}$$



(a) $\left|\max(b_1^{2,0}, b_2^{2,0}) - NumSdpPep\right|$ (b) $\left|\max(b_1^{2,0}, b_2^{2,0}, b_3^{2,0}) - NumSdpPep\right|$

Figure 4.3: Spy plot of the matrix of error, between our guesses and numerical data in $NumSdpPep$ after guessing $b_1^{2,0}$ and $b_2^{2,0}$ to the left and after adding $b_3^{2,0}$ to the right.

Table 4.3: Region (B,S)

$h_1 \backslash h_2$	0.5	0.8	1	1.2
1.8	0.3200	0.2462	0.2133	0.1882
2	0.5000	0.3846	0.3333	0.2941

Table 4.4: Region (S,B)

$h_1 \backslash h_2$	1.8	2
0.5	0.2844	0.4444
0.8	0.1975	0.3086
1	0.1600	0.2500
1.2	-	0.2066

Table 4.5: The regions (B,S) on the left and (S,B) on the right isolated from the sample of NumSdpPep presented in table 4.1 using table 4.2.

$h_1 \backslash h_2$	0.5	0.8	1	1.2
1.8	0.499999999	0.384615	0.333333	0.294117
2	0.499999999	0.384615	0.333333	0.294117

Table 4.6: Sample of $NumSdpPep$ presented in the table to the left of 4.5 corresponding to the (B, S) data that we divided by $(1 - h_1)^2$.

$h_1 \backslash h_2$	1.8	2
0.5	0.444444	0.444444
0.8	0.308641	0.308641
1	0.249999	0.249999
1.2	-	0.206611

Table 4.7: Sample of $NumSdpPep$ presented in the table to the right of 4.5 corresponding to the (S, B) data that we divided by $(1 - h_2)^2$.

Where θ and α are two univariate functions. Just by observing $NumSdpPep$ or the sample in table 4.1 or in 4.2, we can deduce that $\alpha \neq \theta$ so that the values of the worst-case bound is not the same when you switch h_1 and h_2. For example consider the exact worst-case bound $b^{2,\mu}(1.8, 0.5) = \frac{0.32}{2} \neq \frac{0.2844}{2}$ as we can see in table

4.2. At this stage we have exploited all the possible information. Now intuitively, we expected that one of the two functions (α or θ) should be equal to $b_1^{1,0}$. We first checked if it is the case for function $b_3^{2,0}$ in the (B, S) regime and we tested numerically $\theta = b_1^{1,0}(h_2)$ so that we have the following shape for $b_3^{2,0}$:

$$\boxed{b_3^{2,0}(h_1, h_2) = \frac{1}{1 + 2h_2}(1 - h_1)^2} \tag{4.8}$$

As usual we computed the error E as defined in 4.4 between the remaining numerical data and the maximum of our current guesses $\max(b_1^{2,0}, b_2^{2,0}, b_3^{2,0})$. The spy plot for the error E is presented in figure 4.3b. Here is an update of the numerical data fitted by $\max(b_1^{2,0}, b_2^{2,0}, b_3^{2,0})$ in our sample in table 4.8 where we updated table 4.2 by replacing the data fitted by $b_3^{2,0}$ by the relative error and highlighting it in green.

h_1 \ h_2	0	0.5	0.8	1	1.2	1.5	1.8	2
0	1e-09	1e-10	1e-08	3e-09	7e-10	3e-10	9e-10	2e-08
0.5	2e-10	5e-09	2e-10	3e-10	1e-10	2e-10	0.2844	0.4444
0.8	3e-09	1e-10	9e-11	2e-09	1e-10	1e-09	0.1975	0.3086
1	2e-10	7e-11	4e-10	1e-10	1e-10	5e-10	0.1600	0.2500
1.2	1e-10	4e-11	1e-10	2e-09	2e-10	1e-09	1e-09	0.2066
1.5	4e-09	1e-10	2e-10	2e-10	1e-10	1e-10	1e-10	1e-10
1.8	3e-08	3e-10	1e-10	1e-09	3e-09	1e-07	1e-10	1e-10
2	3e-08	1e-10	1e-10	5e-10	1e-09	1e-08	5e-10	1e-08

Table 4.8: Error between the Sample of *NumSdpPep* presented in table 4.1 and $max(b_1^{2,0}, b_2^{2,0}, b_3^{2,0})$. The data fitted by $b_1^{2,0}$ is represented in blue, the data fitted by $b_2^{2,0}$ is represented in red and the one fitted by $b_3^{2,0}$ is represented in green and replaced by their relative error when there is a fit.

So we got three functions and there is still one left and we know that $\alpha(h_1) \neq \frac{1}{2h_1+1}$ as explained earlier, so we went on and tried other functions that may possibly fit.
The advantage is that we can fix h_2 (for example $h_2 = 2$) which will simplify the guess. Let us introduce the column matrix called $NumSB$ that is equal to the column of $NumSdpPep$ where $h_2 = 2$ and h_1 is small, which corresponds to $h_1 \in [0, 1.41]$ or the first one hundred forty one rows of the column. In our sample, K would correspond to the last column in table 4.7. One idea was to guess $1/\alpha(h_1)$ because when plotting the inverse of $NumSB$ we noticed that it looks like it can be fitted by a simple degree two polynomial in h_1. In figure 4.4 you can find this plot as well as our first guess, the simplest polynomial of degree two in h_1 we can think of $(1 + h_1)^2$. We also added the plot of $\frac{1}{b_1^{1,0}(h_1)} = 1 + 2h_1$, to illustrate the fact that $\alpha \neq b_1^{1,0}(h_1)$. So our simple degree two polynomial seems to fit with $NumSB$ as we can see on the plot, and this was quite surprising. In fact until now the shapes of the simple functions we found are "familiar", which means that we have already seen such functions.

This numerical experiment encouraged us to replace $\alpha(h_1)$ by $(1 - h_1)^{-2}$ in 4.7 so that we obtain the following guess for the expression for $b_4^{2,0}$.

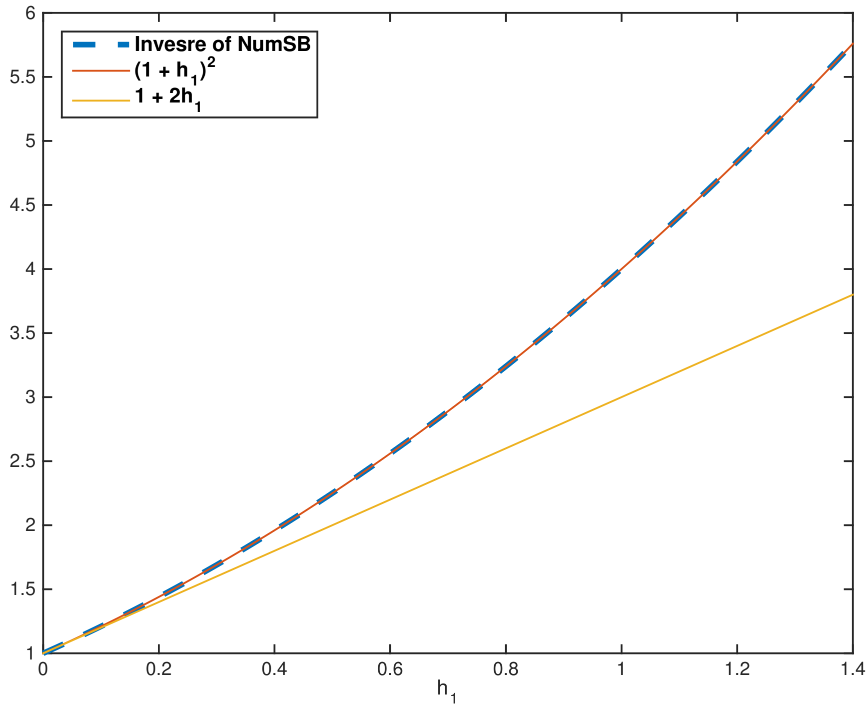


Figure 4.4: Plot comparing the inverse of $NumSB$ corresponding to $h_2 = 2$ and $h_1 \in [0, 1.41]$ extracted from $NumSdpPep$ and the functions $(1+h_1)^2$ and $1+2h_1$.

$$b_4^{2,0}(h_1,h_2) = \frac{1}{(1+h_1)^2}(1-h_2)^2 \tag{4.9}$$

Now let us compute, as we have done untill now after each guess, the matrix of error E between the maximum of our current guesses in this case $max(b_1^{2,0}, b_2^{2,0}, b_3^{2,0}, b_4^{2,0})$ and $NumSdpPep$.As usual, we provide a spy plot in figure 4.5 of the error matrix to see what is left. And it seems like all but two of the data fitted the problem.

The two points remaining in the spy plot of figure 4.5 corresponds to values of h_1 and h_2 where $NumSdpPep$ did not fit the maximum of the four functions we have guessed until now. Here are some details about these two problems of GM:

$$\left|2b^{2,0} - NumSdpPep\right| = 0.1335710^{-5} \quad \text{for} \quad (h_1,h_2) = (0.29, 1.63) \tag{4.10}$$
$$\left|2b^{2,0} - NumSdpPep\right| = 0.1063910^{-5} \quad \text{for} \quad (h_1,h_2) = (0.74, 1.95) \tag{4.11}$$

Looking closer see that we may rewrite the error as being $1.33e-06$ and $1.06e-06$ which is actually larger then 10^{-6} but if we speak in terms of magnitude we have an error of magnitude 10^{-6}. To double check, we resolved sdp-PEP, independently from $NumSdpPep$, and this time we asked specifically the usage of SeDumi and did not use the default command for solving. We did not have a problem and the

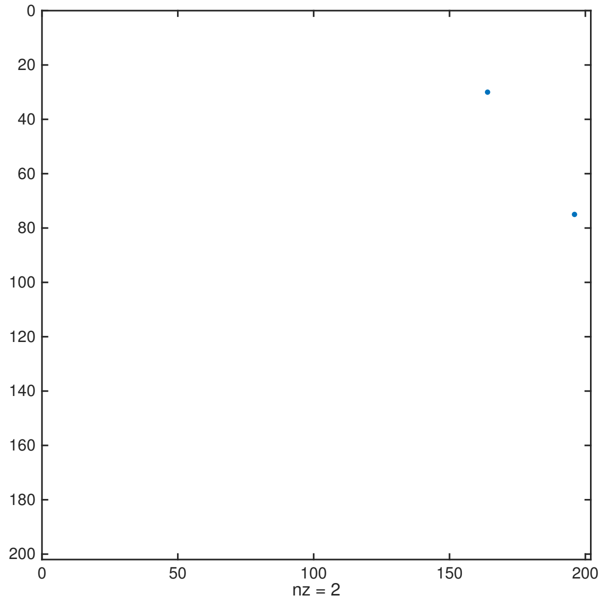


Figure 4.5: Spy plot of the matrix of error, between our guesses and numerical data in $NumSdpPep$ after adding $b_4^{2,0}$.

function $b_4^{2,\mu}$ fitted these two points.

$$\left| b_4^{2,\mu}(0.29, 1.63) - w_{0,1}^{sdp}(1, H(0.29, 1.63), 2, f(x_2) - f_*) \right| = 7.04e - 11$$
$$\left| b_4^{2,\mu}(0.74, 1.95) - w_{0,1}^{sdp}(1, H(0.29, 1.63), 2, f(x_2) - f_*) \right| = 5.56e - 11$$

Hence we can consider our guess for $b_4^{2,0}$ in equation 4.9 valid.

Now let us see what is happening in our sample data. In table 4.9 we have updated the version of the sample set in table 4.8 by replacing the data fitted by the function $b_4^{2,0}$ by the relative error and highlighting them in cyan color in case of fit. So as we can see in the numerical sample all the data we considered is fitted and now we have enough to provide a conjecture on the exact worst-case bound on the objective function accuracy of two steps of GM for smooth convex functions.

Based on our numerical experiment we can deduce $b^{2,0}$ by replacing the functions b_i by their respective expressions in the assumed form in assumption 4.1 and we get by 3.6:

$$b^{2,0} = w_{0,1}^{sdp}(1, H, 2, f(x_2) - f_*) = \frac{1}{2} max(b_1^{2,0}, b_2^{2,0}, b_3^{2,0}, b_4^{2,0})$$

Now as we mentioned earlier in the introduction, using relation 3.5 we can deduce the closed form expression of the exact worst worst-case bound for any L and R by multiplying $b^{2,0}$ by LR^2 and we get :

$$w_{0,L}^{sdp}(R, H/L, 2, f(x_2) - f_*) = LR^2 w_{0,1}^{sdp}(1, H, 2, f(x_2) - f_*) = LR^2 b^{2,0} \tag{4.12}$$

Now that we have $w_{0,L}^{sdp}(R, H/L, 2, f(x_2) - f_*)$ we can finally provide our conjecture for a tight worst-case bound for two steps of GM.

h_1 \ h_2	0	0.5	0.8	1	1.2	1.5	1.8	2
0	1e-09	1e-10	1e-08	3e-09	7e-10	3e-10	9e-10	2e-08
0.5	2e-10	5e-09	2e-10	3e-10	1e-10	2e-10	4e-09	2e-10
0.8	3e-09	1e-10	9e-11	2e-09	1e-10	1e-09	1e-10	1e-08
1	2e-10	7e-11	4e-10	1e-10	1e-10	5e-10	1e-10	1e-08
1.2	1e-10	4e-11	1e-10	2e-09	2e-10	1e-09	1e-09	2e-10
1.5	4e-09	1e-10	2e-10	2e-10	1e-10	1e-10	1e-10	1e-10
1.8	3e-08	3e-10	1e-10	1e-09	3e-09	1e-07	1e-10	1e-10
2	3e-08	1e-10	1e-10	5e-10	1e-09	1e-08	5e-10	1e-08

Table 4.9: Error between the Sample of *NumSdpPep* presented in table 4.1 and $max(b_1^{2,0}, b_2^{2,0}, b_3^{2,0}, b_4^{2,0})$. The data fitted by $b_1^{2,0}$ is represented in blue, the data fitted by $b_2^{2,0}$ is represented in red, the one fitted by $b_3^{2,0}$ is represented in green and the one fitted by $b_4^{2,0}$ is represented in cyan and replaced by their relative error when there is a fit.

Conjecture 4.3. *Any sequence of itterates $\{x_1, x_2\}$ generated by two steps of the gradient method with variable normilzed step sizes $(h_1, h_2) \in [0,2] \times [0,2]$ on a smooth convex function $f \in \mathcal{F}_{0,L}(\mathbb{R}^d)$ satisfies:*

$$f(x_2) - f_* \leq \frac{LR^2}{2} \max\left(\frac{1}{1+2h_1+2h_2}, \frac{1}{(1+h_1)^2}(1-h_2)^2, \right.$$
$$\left. \frac{1}{1+2h_2}(1-h_1)^2, (1-h_1)^2(1-h_2)^2\right) \quad (4.13)$$

Now that we have the expressions of the four functions $b_i^{2,0}$, we start by representing the regions where each b_i is maximal and the corresponding regime. In figure 4.6 we have represented an (X-Y) view of the 3D-plot of the four functions for $(h_1, h_2) \in [0,2] \times [0,2]$. This is one possible way of representing the four regimes. The domain where $b_1^{2,0}$ is maximal is represented in dark blue and this corresponds to the (S,S) regime. Similarly, the (B,B) regime corresponds to the region where $b_2^{2,0}$ is maximal and is represented in red, (B,S) regime is the domain where $b_3^{2,0}$ is maximal and is represented in green and finally we have the domain where $b_4^{2,0}$ is maximal hence the (S,B) regime represented in cyan. We can also notice by looking at the plot in figure 4.6, that the (S,S) region is the largest and seems to cover many problems for h_1 and h_2 between zero and two. This is also observed in our sample set in table 4.9. Now at this stage let us formally say when we will consider h_1 and or h_2 small or big.

Notations 4.1. *For two steps of GM, the value of the first step size h_1 will be considered small if this choice makes us fall in the (S,S) regime or (S,B) which means formally that:*

$$max(b_1^{2,0}, b_2^{2,0}, b_3^{2,0}, b_4^{2,0}) = b_1^{2,0} \quad or \quad b_4^{2,0}$$

Graphically it means that we are in the blue region of figure 4.9 or the cyan region.

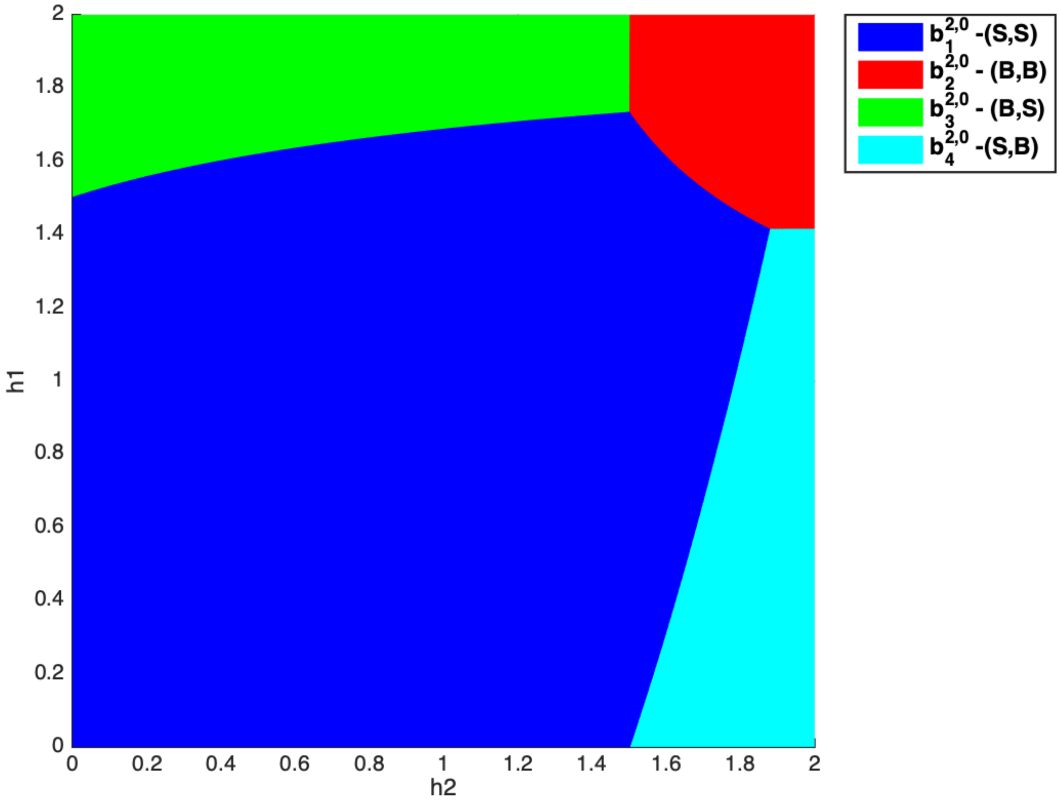


Figure 4.6: X-Y view of the 3D-plot of the functions $b_1^{2,0}$, $b_2^{2,0}$, $b_3^{2,0}$ and $b_4^{2,0}$ and the corresponding regime for $(h_1, h_2) \in [0, 2] \times [0, 2]$

h_1 will be considered big if we fall under (B, B) or (B, S) regime hence when $b_2^{2,0}$ or $b_3^{2,0}$ is maximal. For h_2 it is the same logic. So if h_2 is such that we fall under the (B, S) or (S, S) regime, it will be considered small and if we fall under (S, B) or (B, B) then it will be considered big.

It is interesting to see how these four regimes can be identified on the contour plot of the numerical data $NumSdpPep$ in figure 4.7 where we have represented two hundred levels. Notice the similarity with the the X-Y view in figure 4.6 and how, now that we have seen a representation of the regimes we can clearly identify them by looking at the contour plot of the numerical data. Now we can discuss the optimal step sizes or what we will call the optimal combination of steps. This corresponds to the values of h_1 and h_2 that achieves the lowest worst-case and we will denote such a combination by a vector $\mathbf{h_{opt}}(2)$. In example 4.1, we have seen that the optimal step length for one step of GM denoted by $h_{opt}(1)$ was the coordinate of the intersection of $b_1^{1,0}$ and $b_2^{1,0}$. Then referring to [Tay17], we can find a discussion on the optimal step length for N constant step sizes of GM denoted by $h_{opt}(N)$ which is also the coordinate of the intersection of the two functions in the maximum, which expressions can be found in conjecture 4.1.
For two constant steps we can thus find it by solving $b_1^{2,0}(h, h) = b_2^{2,0}(h, h)$ and solving numerically we found:

$$h_{opt}(2) \simeq 1.605829 \qquad b^{2,0}(h_{opt}(2), h_{opt}(2)) = 0.0673553 \tag{4.14}$$

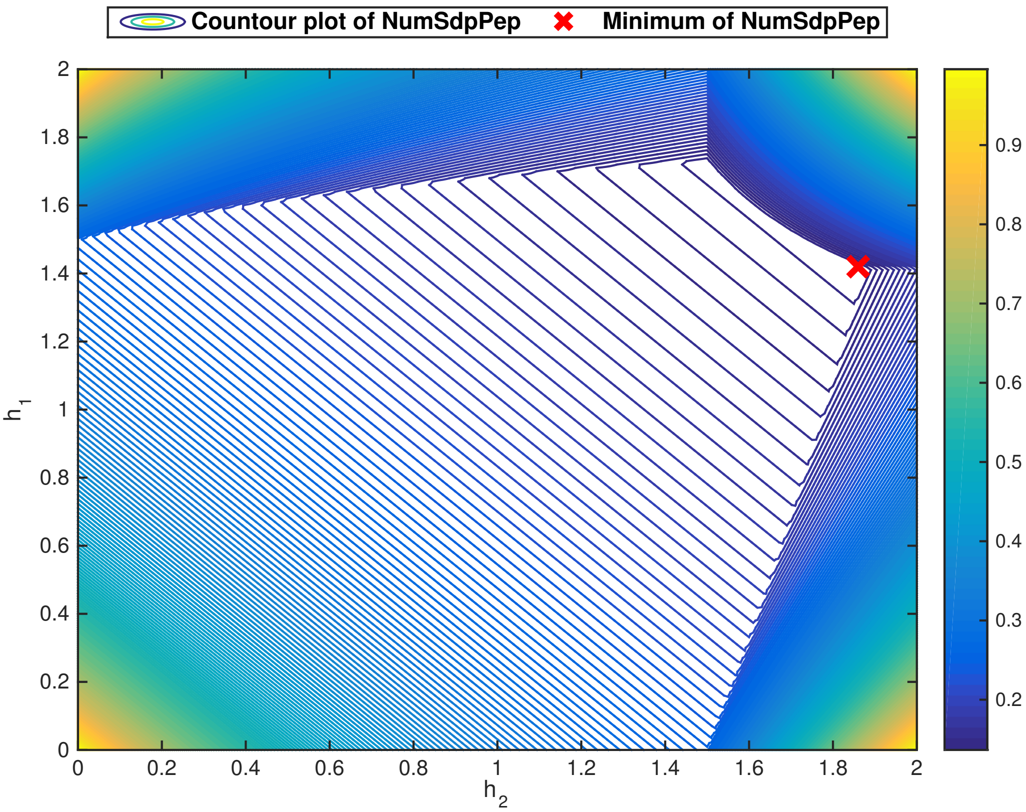


Figure 4.7: Contour plot of the numerical data in *NumSdpPep* and the location of the minimal value in the set for the 201 values of h_1 and h_2 between zero and two.

For two variable step sizes, we may expect to find this point on one of the intersections between the functions $b_i^{2,0}$. First we wanted to get a numerical intuition so we started by locating the minimal value in *NumSdpPep* which gave us an approximate location of the optimal combination of step. In figure 4.7 along with the contour levels we have marked by a red cross the minimal value of *NumSdpPep* corresponding to $h_1 = 1.42$ and $h_2 = 1.86$. Notice that it seems like we are at one of the intersections, specifically at the intersection between $b_1^{2,0}$, $b_2^{2,0}$ and $b_4^{2,0}$.
Now let us try to find the argument of the minimum (argmin) of $w_{0,L}^{sdp}(R, H/L, 2, f(x_2) - f_*)$. It is equivalent to finding the argmin of $b^{2,0}$ from 4.12. We started by a 3D-plot of $b^{2,0}$ represented in figure 4.8. On the same plot we have added some of the contour levels of $b^{2,0}$, to make a link with the contour levels of *NumSdpPep* represented in figure 4.7, as well as two points representing the intersections of $b_1^{2,0}$ $b_2^{2,0}$ and $b_4^{2,0}$ and $b_1^{2,0}$ $b_2^{2,0}$ and $b_3^{2,0}$ represented by a red cross and a black cross respectively.

We can find the coordinates of these points analytically or numerically, and we find the following:

$$b_1 \cap b_2 \cap b_3 \quad (h_1^{1,2,3}, h_2^{1,2,3}) = (\sqrt{3}, 1.5)$$
$$b_1 \cap b_2 \cap b_4 \quad (h_1^{1,2,4}, h_2^{1,2,4}) = (\sqrt{2}, 0.25(3 + \sqrt{9 + 8\sqrt{2}}))$$

Graphically, in figure 4.8 we can see that the minimum is located at one of the two points. From numerical observation, we can deduce that most probably the argmin

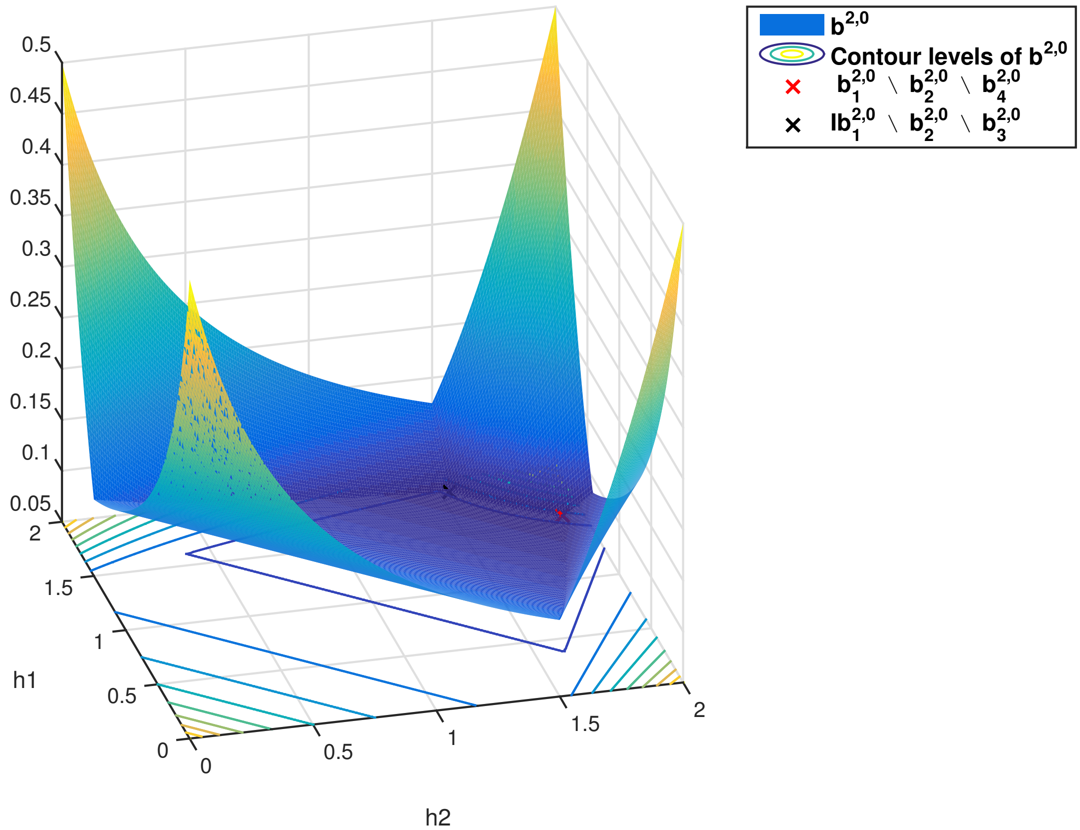


Figure 4.8: 3D-Plot surface plot of $b^{2,0}$ as well as some contour levels and the intersections of $b_1^{2,0}$ $b_2^{2,0}$ and $b_4^{2,0}$ and $b_1^{2,0}$ $b_2^{2,0}$ and $b_3^{2,0}$ which seems to be candidates for minimum of $b^{2,0}$.

is $(h_1^{1,2,4}, h_2^{1,2,4})$. Graphically it seems hard to judge so we computed the value of $b^{2,0}$ at those two points and compare them.

$$b^{2,0}(h_1^{1,2,3}, h_2^{1,2,3}) = 0.066987 \qquad > \qquad b^{2,0}(h_1^{1,2,4}, h_2^{1,2,4}) = 0.065945$$

So we can deduce the argmin of $b^{2,0}$ thus the optimal combination of steps $\mathbf{h_{opt}}(2)$ that achives the lowest worst cast of GM which is equal to $(h_1^{1,2,4}, h_2^{1,2,4})$.

$$\mathbf{h_{opt}(2)} = \left(\sqrt{2}, 0.25\left(3 + \sqrt{9 + 8\sqrt{2}}\right)\right) \simeq (1.41, 1.87) \tag{4.15}$$

Comparing with the optimal step length computed for two constant steps of GM presented in 4.14, notice that $b^{2,0}(h_{opt}(2), h_{opt}(2)$ is larger then $b^{2,0}(h_1^{1,2,4}, h_2^{1,2,4})$. Specifically we have:

$$\frac{\left|b^{2,0}(h_1^{1,2,4}, h_2^{1,2,4}) - b^{2,0}(h_{opt}(2), h_{opt}(2))\right|}{b^{2,0}(h_{opt}(2), h_{opt}(2)} = 0.0209$$

So that we decrease the optimal wost case value by 2%, which is not that large but still, it seems like the optimal choice for two steps of GM is not to the one corresponding to two constant steps but $\mathbf{h_{opt}}(2)$ in (4.15).

Finally we can make a final comment on the convexity of the new functions $b_i^{2,0}$ that we discovered. When we look at one step in example 4.1, you can notice that the two functions $b_1^{1,0}$ and $b_2^{1,0}$ are convex for al the values of h_1 considered. We may see that graphically in figure 4.1. But for two steps and it is not necessarily the case. Let us compute for example the Hessian of the function $b_2^{2,0}$.

$$\nabla^2 b_2^{2,0}(h_1, h_2) = \begin{pmatrix} 2(1-h_2)^2 & 4(1-h_1)(1-h_2) \\ 4(1-h_1)(1-h_2) & 2(1-h_1)^2 \end{pmatrix}$$

When computing the determinant you find $-12(1-h_1)^2(1-h_2)^2$ which is clearly negative, thus this function is not convex.

So to sum up a little bit, in this section we started by the first step of GM-Proc by fixing μ to zero and N=2 of course. Then we generated the data for 201 values of h_1 and h_2 between zero by solving sdp-PEP numerically and all the data was stored in a matrix called $NumSdpPep$. Then we started by observing the history which is the case of one step and the conjectures for N constant steps of GM. Based on these observations we made some assumptions and used Assumptions 4.1 and 4.2 to guide our intuition in the experiments that led to conjecture 4.3. One important thing is that the function $b_4^{2,0}$ was unexpected because it involves the term $\alpha(h_1) = (1+h_1)^{-2}$ that was unseen before and that, in opposition to what we may deduce from the case of one step, the functions $b_i^{2,0}$ are not convex.
Finally, by computing the optimal combination of steps we noticed a small decrease in the optimal worst-case with respect to the case of constant step sizes, suggesting that it may be better to go for two fixed but different step sizes of GM instead of constant step sizes.

4.2.2 Conjecture on smooth strongly convex functions

In the case of smooth strongly convex functions, we have three parameters and functions of three variables to guess. We can add to the history the bound for smooth convex functions conjectured and tested numerically in the previous section. This case corresponds to the "asymptotic" case since as we mention in assumption 4.3. Hence the procedure for guessing the bound for smooth strongly convex functions will be similar to the one for smooth convex functions. We started by solving sdp-PEP for forty one values of h_1 and h_2 between zero and two and fifty one values of strong convexity parameter μ between zero and 0.5. We did not generate a large data set simply because we have three variables. We decided to store the data in a matlab structure called cell that we will denote by $NumSdpPepMu$, where we associated to each value of μ a 41×41 matrix corresponding to the numerical values of $w_{\mu,1}^{sdp}(1, H, 2, f(x_2) - f_*)$ for different values of h_1 and h_2 considered.

It was natural from assumption 4.3to expect the following for the worst-case bound we are trying to find for smooth strongly convex functions:

- The bound is the maximum of four functions And there are four regimes in terms of h_1 and h_2 for each μ. We will associate the functions $b_i^{2,\mu}$ to the regimes as we did in the previous section. So we have:

$$b_1^{2,\mu} \leftrightharpoons (S,S) \quad b_2^{2,\mu} \leftrightharpoons (B,B) \quad b_3^{2,\mu} \leftrightharpoons (B,S) \quad b_4^{2,\mu} \leftrightharpoons (S,B) \tag{4.16}$$

- These functions converge to the one conjectured for smooth convex functions.

The easiest guess was the one of $b_2^{2,\mu}$. As we can see for one step the regime associated to the function $(1-h_1)^2$ which is maximal when h_1 is big, is common to $b^{1,0}$ (4.1) and $b^{1,\mu}$ (4.2). Furthermore we have the same observation for the term $(1-h)^{2N}$ in conjectures 4.2 and 4.1. So we make the assumption that $b_2^{2,\mu}$ is equal to $b_2^{2,0} = (1-h_1)^2(1-h_2)^2$ a function that we found previously in (4.6).
As for the (S,S) regime, notice first that in the case of constant step sizes ($h_1 = h_2 = h$) we have that:

$$\lim_{\kappa\to 0} \frac{\kappa}{(\kappa-1)+(1-\kappa h)^{-4}} = \frac{1}{1+4h}$$

Then for one step of GM and smooth strongly convex functions $b_1^{1,\mu} = s(1-\kappa h_1)$ where the function s is defined as follows:

$$s(x) = \frac{\kappa}{(\kappa-1)+x^{-2}}$$

So combining these two information we can try $b_1^{2,\mu}(h_1,h_2) = s((1-\kappa h_1)(1-\kappa h_2))$. Hence here are the first guesses for $b_1^{2,\mu}$ and $b_2^{2,\mu}$, where we remind the reader that $\kappa = \frac{\mu}{L}$ is the inverse of the condition number.

$$b_1^{2,\mu}(h_1,h_2) = \frac{\kappa}{(\kappa-1)+(1-\kappa h_1)^{-2}(1-\kappa h_2)^{-2}} \tag{4.17}$$

$$b_2^{2,\mu}(h_1,h_2) = (1-h_1)^2(1-h_2)^2 \tag{4.18}$$

Then we tested these functions for different values of μ and for each μ, we computed the matrix of error E as defined in 4.4 and got a fit with a tolerance of 10^{-6} for two out of four regions. For the (B,S) regime, we where inspired by the function $b_3^{2,0}$ (4.8). So we can keep the term $(1-h_1)^2$ as it is and replace $(1+2h_2)^{-1}$ by $s(1-\kappa h_2)$. Thus we obtain the following expression for $b_3^{2,\mu}$:

$$b_3^{2,\mu}(h_1,h_2) = \frac{\kappa}{(\kappa-1)+(1-\kappa h_2)^{-2}}(1-h_1)^2 \tag{4.19}$$

Again we tested numerically by computing the matrix of error E (4.4) between the maximum of the three functions currently guessed and $NumSdpPepMu$ and got a fit for most of the remaining data, that where not matched by $b_1^{2,\mu}$ and $b_2^{2,\mu}$.
Now for the (S,B) regime so function $b_4^{2,\mu}$, since $b_3^{2,\mu}$ can be written as the product of a function of h_1 and a function of h_2 similarly to the case of smooth convex functions, we started by assuming that

$$b_4^{2,\mu}(h_1,h_2) = \beta_\kappa(h_1)(1-h_2)^2$$

Where β_κ is an unknown function that converges to α, the function defined for $b_4^{2,0}$, when μ tends to zero. Here we relied on our intuition and really there is no other explanation, and we found:

$$\beta_\kappa(x) = \left(\frac{\kappa}{(\kappa-1)+(1-\kappa x)^{-1}}\right)^2 \tag{4.20}$$

Here is the proof of the convergence of β_κ to $\alpha(x) = (1+x)^{-2}$.

Proof. We start by rearranging the terms in $\frac{1}{\beta_\kappa(x)}$.

$$\begin{aligned}\frac{1}{\beta_\kappa(x)} &= \Big(\frac{\kappa - 1 + \frac{1}{1-\kappa x}}{\kappa}\Big)^2 \\ &= \Big(\frac{1 + x - \kappa x}{1-\kappa x}\Big)^2\end{aligned}$$

When μ tends to zero, κ does to, and we get:

$$\lim_{\mu\to 0} \frac{1}{\beta_\kappa(x)} = (1+x)^2$$

□

So now that we have guessed the function β_κ we provide the following guess for the function $b_4^{2,\mu}$:

$$b_4^\mu(h_1, h_2) = \Big(\frac{\kappa}{(\kappa-1)+(1-\kappa h_1)^{-1}}\Big)^2 (1-h_2)^2 \tag{4.21}$$

We tested numerically by computing as usual the error matrix E defined in 4.4 between the maximum of the current guessed functions $max(b_1^{2,\mu}, b_2^{2,\mu}, b_3^{2,\mu}, b_4^{2,\mu})$ and $NumSdpPepMu$ for different values of μ. We obtain a fit with a tolerance of 10^{-6} and this time we did not have any data that did not fit our assumption for all the values of μ we tested. Let us see what is happening to the error for the two points in which we had an error of $1.33e-06$ and $1.06e-06$ presented in 4.11. We have computed the relative error between $2b^{2,\mu} = max(b_1^{2,\mu}, b_2^{2,\mu}, b_3^{2,\mu}, b_4^{2,\mu})$ for some values of μ. The results are shown in table 4.10 and as you can see the error is smaller then our tolerance of $1e-06$, comforting us in the choice we made to ignore these two points for the case of smooth convex functions.

(h_1, h_2) \ μ	0.001	0.01	0.02	0.1
(0.29,1.63)	2e-09	2e-08	6e-07	1e-08
(0.74,1.95)	1e-08	1e-08	4e-08	6e-07

Table 4.10: Testing the numerical error between $2b^{2,\mu}$ and numerical values returned by solving sdp-PEP for the two problems where we observed a bad fit for the case where $\mu = 0$ presented in 4.11.

So the empirical results from the resolution of sdp-PEP seems to support the following conjecture on the worst-case bound of two steps of gradient method applied to smooth strongly convex function:

Conjecture 4.4. *Any sequence of iterates $\{x_1, x_2\}$ generated by two steps of gradient method with variable normalized step sizes $(h_1, h_2) \in [0,2] \times [0,2]$ on a smooth strongly convex function $f \in \mathcal{F}_{\mu,L}(\mathbb{R}^d)$*

$$f(x_2) - f_* \leq \frac{LR^2}{2} \max\Bigg(\frac{\kappa}{(\kappa-1) + (1-\kappa h_1)^{-2}(1-\kappa h_2)^{-2}}, \left(\frac{\kappa}{(\kappa-1)+(1-\kappa h_1)^{-1}}\right)^2 (1-h_2)^2, \frac{\kappa}{(\kappa-1)+(1-\kappa h_2)^{-2}}(1-h_1)^2, (1-h_1)^2(1-h_2)^2 \Bigg) \quad (4.22)$$

Where $\kappa = \frac{\mu}{L}$ the inverse of the condition number.

Notice also as we mentioned earlier that we can observe that since each of the functions $b_1^{2,\mu}$, $b_2^{2,\mu}$, $b_3^{2,\mu}$ and $b_4^{2,\mu}$ converges to $b_1^{2,0}$, $b_2^{2,0}$, $b_3^{2,0}$ and $b_4^{2,0}$ respectively we have the following result:

$$\lim_{\mu \to 0} b^{2,\mu}(h_1, h_2) = b^2(h_1, h_2) \ \ \forall (h_1, h_2) \in [0,2] \times [0,2]$$

Which seems to confirm assumption 4.3. The proof of this result is similar to the proof of 4.3 presented during our preliminary observations, or the proof for β_κ. In figure 4.9 we have plotted an (X-Y) view of the 3D-plot of the four functions for $\mu = 0.3$ and $(h_1, h_2) \in [0,2]^2$. We have highlighted regions where $b_1^{2,\mu}$ is maximal so the (S,S) region in dark blue, the one where $b_2^{2,\mu}$ is maximal ((B,B) regime) in red, $b_3^{2,\mu}$ in green and $b_4^{2,\mu}$ in light blue. The choice of colors is the same as in figure 4.6 to illustrate the similarity between the cases of smooth convex and smooth strongly convex. Observe that although we have four regimes also, the size of the regions differs specially (S,S).

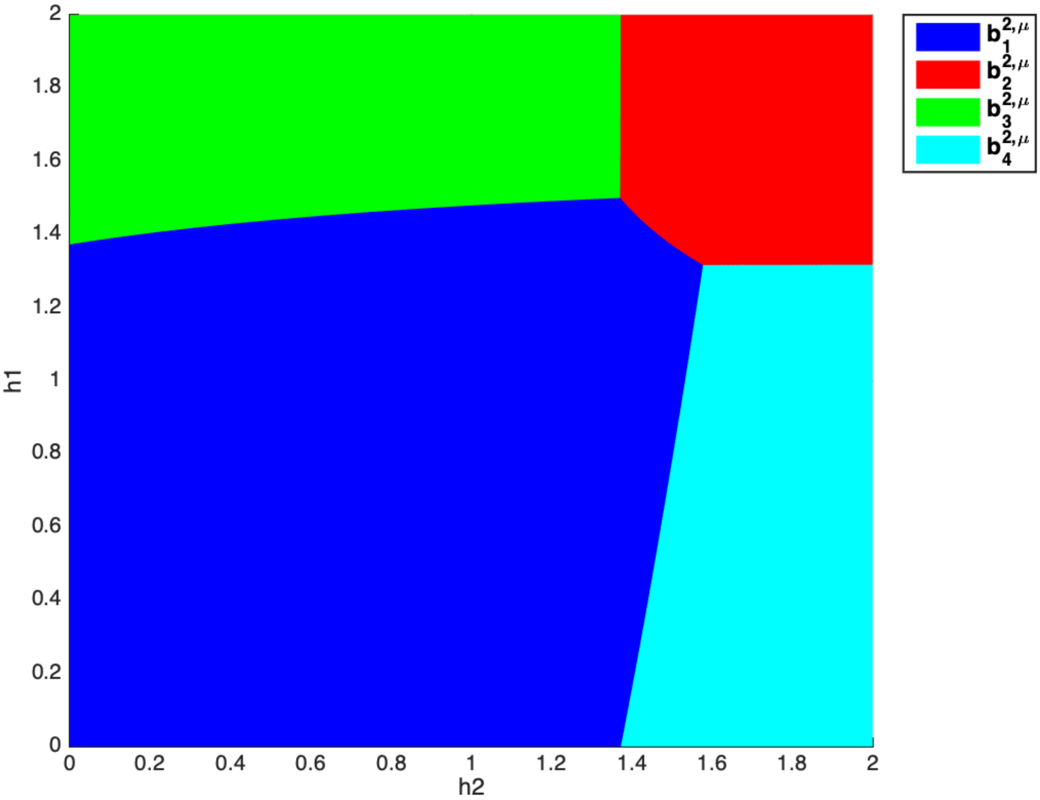


Figure 4.9: Plot of the bound in conjecture 4.4 for $(h_1, h_2) \in [0, 2] \times [0, 2]$ and $\mu = 0.3$

4.3 Three steps of GM

So far, no clear conclusions can be made regarding N variable step sizes of GM. Nevertheless, we gained some intuition and learned more about general worst-case bound on objective function accuracy. It seems like the bound is the maximum of functions and their number is growing like 2^N with N.
It seems also that the bound for the case of smooth strongly convex functions can be deduced from the one for smooth convex functions, the latter being easier to guess because it depends only on the normalized step sizes. This is true under the assumption that the functions $b_i^{N,\mu}$ seems to converge when μ tends to zero to their equivalent for smooth convex functions (assumption 4.3). In this section we decided to tackle the problem of finding bound for three steps of GM. We will restrict ourselves to the case of smooth convex functions.
We started by solving as usual, sdp-PEP for different values of h_1, h_2 and h_3 chosen between zero and two in the following set:

$$\{i/10\}_{i=0,1,\cdots,20}$$

and stored the data in a cell Matlab structure where for each h_1 we have presented the numerical solutions of sdp-PEP. This dataset will be denoted by $NumSdpPep3$. We also multiplied these solutions by two as we did for two steps and after that we started our tests. Eventually we predicted that the bound will be the maximum of eight functions, so the task was not easy but now, along with the special cases of

constant step sizes (conjecture 4.1) and the bound for one step discussed in example 4.1, we also have the bound found for two steps (conjecture 4.3).
We can consider that there is eight combinations of steps, corresponding to the choice small or big for h_1, h_2 and h_3. To each combination we want to associate a function b_i which will be maximal for the specific combination we are considering. We found it convenient here to change the notations for the functions, since now we are assuming, that for N steps their will 2^N regimes, each one mapped by a certain function.

Notations 4.2. *From now on in this chapter, for every function $b_i^{N,0}$ with $i \in J = \{1, \cdots, 2^N\}$ we will replace i by the regime or the region mapped by this function.*

Here are the first four formulas that we guessed, corresponding to the four combinations where h_1 is considered small.

$$b^{3,0}_{S,S,S}(h_1,h_2,h_3) = \frac{1}{1+2h_1+2h_2+2h_3} \tag{4.23}$$

$$b^{3,0}_{S,B,B}(h_1,h_2,h_3) = \Big(\frac{1}{(1+h_1)^2}\Big)(1-h_2)^2(1-h_3)^2 \tag{4.24}$$

$$b^{3,0}_{S,S,B}(h_1,h_2,h_3) = \Big(\frac{1}{(1+h_1+h_2)^2}\Big)(1-h_3)^2 \tag{4.25}$$

$$b^{3,0}_{S,B,S}(h_1,h_2,h_3) = \Big(\frac{1}{(1+h_1)^2}\Big)(1-h_2)^2\Big(\frac{1}{1+2h_3}\Big) \tag{4.26}$$

Before moving further, we will provide some explanation on how we found these functions. Inspired by the procedure two variable step sizes, the function $b_1^{3,0}$ is the generalization of (4.5). For the function $b_2^{3,0}$, corresponding to h_1 and h_2 small and h_3 big or (S,S,B) regime, we simply took (4.9) and replaced h_1 by h_1+h_2 and h_2 by h_3. As for b_3 and b_4, they where inspired by combining (4.8) and (4.9).

As usual, we computed the error matrix for each h_1 between the maximum of the functions guessed until now and the data in $NumSdpPep3$. We observed a fit of the maximum of these four functions for small values of h_1 so we can say that these formulas fits the numerical bounds with a precision of 10^{-6}. But at this stage this we do not have a complete picture yet, we can't be more specific about the values where h_1, h_2 or h_3 will be considered small or big.

In figure 4.10 we have plotted an X-Y view of the 3D-plot of $b^{3,0}_{S,S,S}$, $b^{3,0}_{S,B,B}$, $b^{3,0}_{S,S,B}$ and $b^{3,0}_{S,B,S}$ for $h_1 = 1$ and $(h_2,h_3) \in [0,2] \times [0,2]$.
We can make two observation in this plot. First if we compare the blue region that correponds to the function (S,S,S) to the (S,S) region in blue on the plot of figure 4.6, you can see that it is much bigger for three steps, a trend observed for other small values of h_1. Second, the intersection between these functions lies in the regions where $h_2 \geq 2$ and $h_3 \geq 2$, suggesting that optimal step sizes are values of h_2 and h_3 larger then two when $h_1 = 1$, more on that at the end of this section.

Now their are still four functions to be guessed and they all correspond to regimes where h_1 is big. Here are the first three we guessed using the same reasoning as for the combinations where h_1 was small hence directly inspired by the functions found

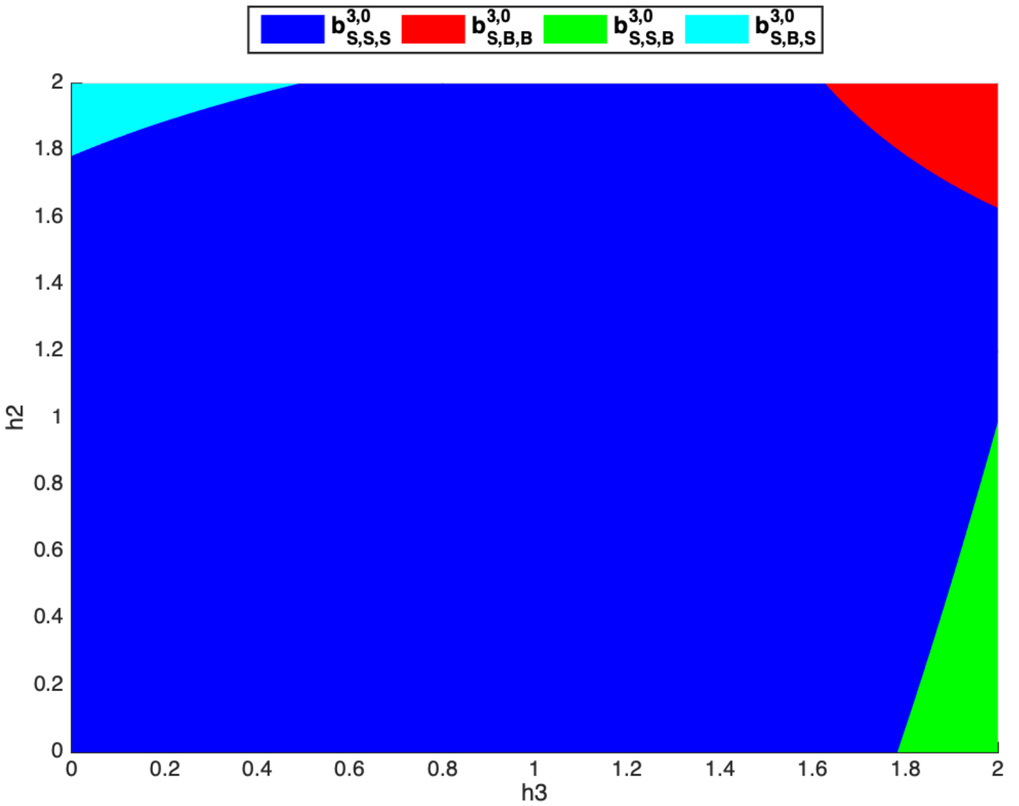


Figure 4.10: X-Y view of the 3D-plot of $b^{3,0}_{S,S,S}$, $b^{3,0}_{S,B,B}$, $b^{3,0}_{S,S,B}$ and $b^{3,0}_{S,B,S}$ for $h_1 = 1$.

for two steps.

$$b^{3,0}_{B,S,S}(h_1, h_2, h_3) = \left(\frac{1}{1 + 2h_2 + 2h_3}\right)(1 - h_1)^2 \tag{4.27}$$

$$b^{3,0}_{B,S,B}(h_1, h_2, h_3) = \left(\frac{1}{(1 + h_2)^2}\right)(1 - h_1)^2(1 - h_3)^2 \tag{4.28}$$

$$b^{3,0}_{B,B,B}(h_1, h_2, h_3) = (1 - h_1)^2(1 - h_2)^2(1 - h_3)^2 \tag{4.29}$$

In figure 4.11 we have presented a spy-Plot of the mismatches between $\max(b^{3,0}_{B,S,S}, b^{3,0}_{B,S,B}, b^{3,0}_{B,B,B})$ and numerical values of the bound with 10^{-6} accuracy for $h_1 = 2$.

As for the function $b^{3,0}_{B,B,S}$, the function should be valid for h_3 small ($h_3 \leq 1.4$ in our numerical experiments) so that we are in the eighth regime (B, B, S). We found it natural to test the following function, as it is the procedure we have followed until now.

$$b^{3,0}_{B,B,S}(h_1, h_2, h_3) = \frac{1}{(1 + 2h_3)}(1 - h_2)^2(1 - h_1)^2$$

But when tested numerically, the fit was much lower then what we expected. In fact if we consider again the case of $h_1 = 2$, the maximum value of the error between the numerical data and $b^{3,0}_{B,B,S}$ was 10^{-4} attained for $h_2 = 1.7$ and $h_3 = 1.1$. It seems like their is a new formula here or that we are looking in the wrong direction.

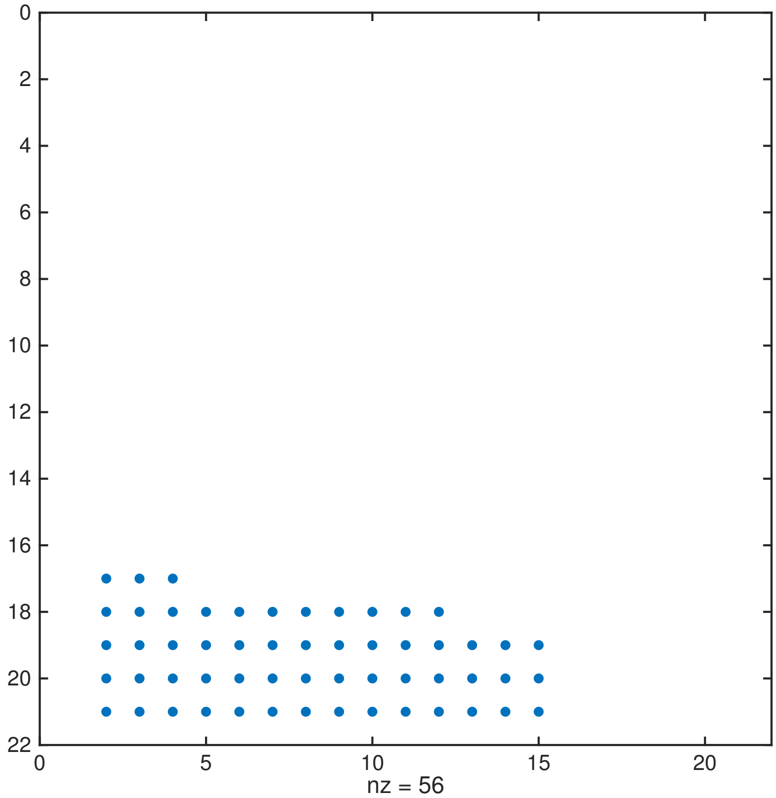


Figure 4.11: Spy plot of the matrix of error E between $\max(b^{3,0}_{B,S,S}, b^{3,0}_{B,S,B}, b^{3,0}_{B,B,B})$ and data from $NumSdpPep3$ where $h_1 = 2$. The dots corresponds to points where $\left|\max(b^{3,0}_{B,S,S}, b^{3,0}_{B,S,B}, b^{3,0}_{B,B,B}) - NumData\right| > 10^{-6}$

We are still hopping that this function can be written as:

$$b^{3,0}_{B,B,S}(h_1, h_2, h_3) = \gamma(h_3)(1 - h_2)^2(1 - h_1)^2$$

With our numerical experiments we were sure that the function γ is not equal to any function or expression we have discovered up to this point:

$$\gamma(h_3) \neq \frac{1}{1 + 2h_3} \qquad \gamma(h_3) \neq \frac{1}{(1 + h_3)^2}.$$

for $h_1 = h_2 = 2$ we have plotted in figure 4.12 the numerical data, extracted from $NumSdpPep3$ and the functions that we know until now for small step sizes $(1 + 2h_3)^{-1}$ and $(1 + h_3)^{-2}$ in order to emphasize that it is a new function.

So maybe γ is not only a function of h_3 but also a function of h_2 (this is not odd because we observed such functions for b_5) and/or h_1. In fact, when we divided the remaining data by $(1 - h_1)^2(1 - h_2)^2$, we did not loose the dependence in h_1 and h_2 as we observed in the case of $b^{2,0}_3$ 4.8 as we can see table 4.7 in the study of two steps for smooth convex functions. Then the best thing to do is to try some interpolation or maybe other technique in the hope of finding a **simple** function. Actually, because we did not have any clue until now we have tried an interpolation using matlab curve fitting app and for $h_1 = h_2 = 2$ we obtained the best match with

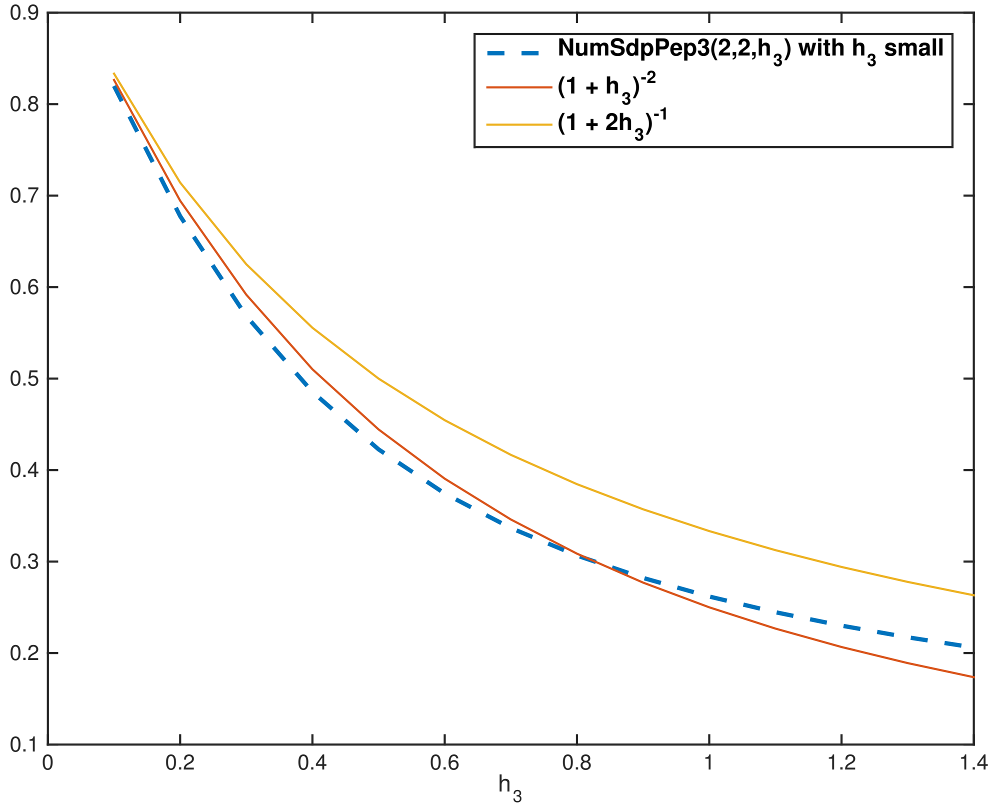


Figure 4.12: Plot comparing Numerical Data extracted from $NumSdpPep3$ for $h_1 = h_2 = 2$ and h_3 small, to the functions $(1+2h_3)^{-1}$ and $(1+h_3)^{-2}$.

the following rational function :

$$\gamma(h_3) = \frac{p_1 h_3^2 + p_2 h_3 + p_3}{h_3^3 + q_1 h_3^2 + q_2 h_3 + q_3} \tag{4.30}$$

But the values obtained for these coefficients was not easy to interpret, or clear enough to generalize for h_1 and h_2 different then two. We can say based on our numerical experiments that γ is actually a function of h_1, h_2 and h_3 and for it to make sense, at least to be coherent with what we have got until now, the function $b^{3,0}_{B,B,S}$ should verify the following:

$$b^{3,0}_{B,B,S}(h_1, h_2, 0) = (1-h_1)^2(1-h_2)^2$$

So one idea was to divide the remaining unmatched data in $NumSdpPep3$ corresponding to the (B, B, S) region by $(1-h_1)^2(1-h_2)^2$ and try to guess the function γ witch is a real multivariate function. Interpolation for multiple dimension may be one way to proceed, but is surely not easy specially when we are not sure of the shape of the function.
Although we were not able to find the exact worst-case bound for three steps of GM, we can have an idea about the optimal step size at least numerically. This is why we computed the argmin of $NumSdpPep3$ numerically and found that the minimum of the data set is attained for $h_1 = 1.4$, $h_2 = 2$ and $h_3 = 1.4$ considering

as we mentioned i the beginning that h_1, h_2 and h_3 are in $\{i/10\}_{i=1,\cdots,20}$. The value of hh_2 suggests that we are at the boundary of the domain of h_2 and we may found smaller values for the worst-case bound for values of h_2 larger then two.
In fact until, now we have limited our estimations to the cases where h_i are less then two because in the textbook study of the GM, we make such a consideration and simply to follow conjectures 4.1 and 4.2 for constant step sizes where they limit the step size to two. But going back to [DT13], we noticed that using some approximations, they where able to produce a first order method, where you fix at the beginning your step sizes to some optimal values. In section 5 of [DT13] you can see an example for $N = 5$ and we observe that some of the optimal values are larger then two.
So we generated a new data set but this time we considered h_1, h_2 and h_3 are chosen between 0 and **3** in the following set:

$$\{i/10\}_{i=0,\cdots,30}$$

Now we recomputed the bound numerically and we obtained the following numerical estimation of $\mathbf{h_{opt}}(3)$ denoted by $\mathbf{h_{opt}^{NUM}}(3)$

$$\mathbf{h_{opt}^{NUM}}(3) = (1.4, \mathbf{2.4}, 1.5) \qquad w_{0,1}^{sdp}(1, H(1.4, 2.4, 1.5), 3, f(x_3) - f_*) \approx 0.086206$$

So numerically we have an indication that indeed the optimal combinations of steps implies that one of the steps will be larger then two. Now looking closer we noticed that this value is located on the intersection between the functions $b^{3,0}_{S,S,S}$, $b^{3,0}_{S,B,B}$ and $b^{3,0}_{S,B,S}$ as we can see in the 3D plot in figure 4.13. Luckily we have guessed these functions already. Now when we tested numerically for $h_1 = 1.4$ we where surprised to see that their is one additional regime that did not match the functions guessed for small h_1, we suppose that it is the eight's function but we can say is that these values are larger then the one fitted by the four functions corresponding to h_1 small. Now it was interesting to compute the intersection of the three functions $b^{3,0}_{S,S,S}$, $b^{3,0}_{S,B,B}$ and $b^{3,0}_{S,B,S}$. This can be done using Wolfram Mathematica [Inc] and we get the following:

$$b^{3,0}_{S,S,S} \cap b^{3,0}_{S,B,B} \cap b^{3,0}_{S,B,S} = \left\{ (h_1, h_2, h_3) \mid h_3 = 1.5 \; h_1 = \frac{h_2^2 - 3}{2} \; 1 - h_2^2 \neq 0 \right\} \tag{4.31}$$

SO if we consider $h_1 = 1.4$ we find the following point:

$$(h_1, h_2, h_3) = \left(1.4, \sqrt{\frac{29}{5}}, 1.5 \right) \approx (1.4, 2.4083189, 1.5) \tag{4.32}$$

Notice how close we are from $\mathbf{h_{opt}^{NUM}}(3)$ and when we compute $w_{0,1}^{sdp}(1, H(1.4, 2.4083189, 1.5), 3, f(x_3) - f_*)$ we obtained a fit with actually the three functions. So that we can obtain a first approximation of the $\mathbf{h_{opt}}(3)$:

$$\widehat{\mathbf{h}}_{\mathbf{opt}}(3) = \left(1.4, \sqrt{\frac{29}{5}}, 1.5 \right) \approx (1.4, 2.4083189, 1.5) \tag{4.33}$$

$$b^{3,0}(\widehat{\mathbf{h}}_{\mathbf{opt}}(3)) = w_{0,1}^{sdp}(1, H(\widehat{\mathbf{h}}_{\mathbf{opt}}(3)), 3, f(x_3) - f_*) = 0.0430416 \tag{4.34}$$

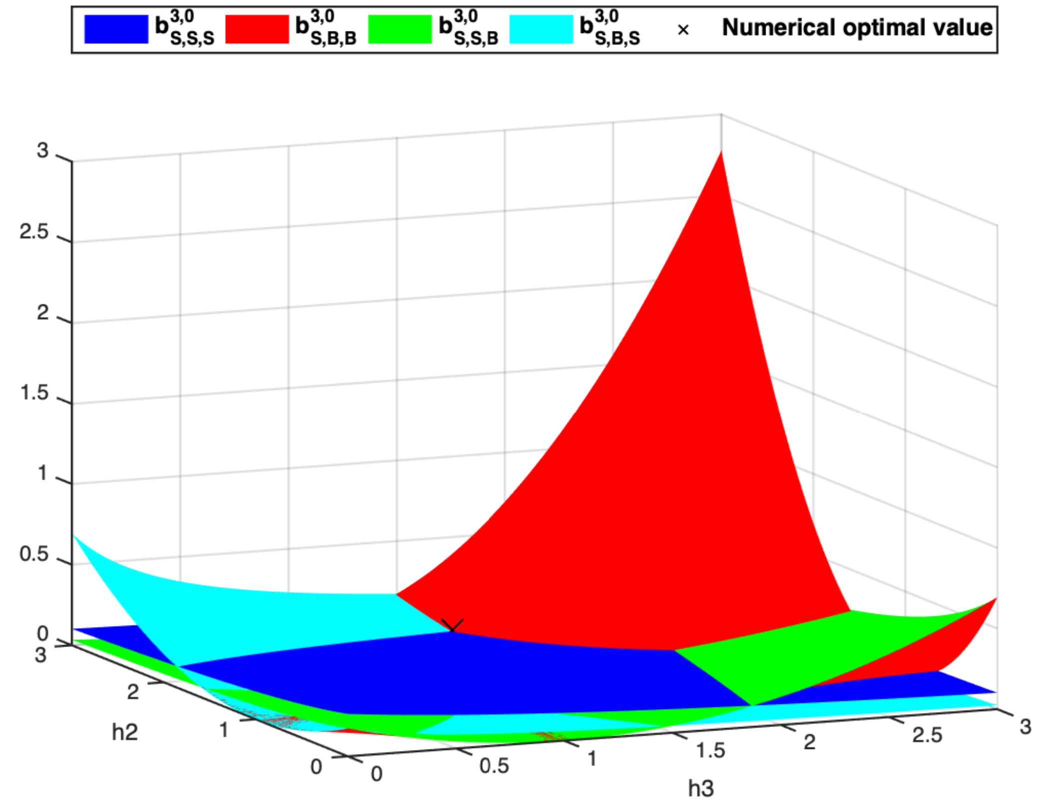


Figure 4.13: 3D plot of the functions $b^{3,0}_{S,S,S}$, $b^{3,0}_{S,B,B}$, $b^{3,0}_{S,B,S}$, $b^{3,0}_{S,S,B}$ and the minimum computed numerically, $w^{sdp}_{0,1}(1, H(1.4, 2.4, 1.5), 3, f(x_3) - f_*)$.

As we did in the previous section, we start by computing $h_{opt}(3)$, the optimal step size for the case of three constant step sizes by solving $(1 + 6h)^{-1} = (1 - h)^6$ and numerically we get :

$$h_{opt}(3) \approx 1.670332 \qquad b^{3,0}(h_{opt}(3), h_{opt}(3), h_{opt}(3)) \approx 0.0453638 \tag{4.35}$$

We can see that the approximation of the optimal worst-case bound for variable step sizes (4.34) is already smaller then the one for constant step sizes (4.35). Now if we compute the percentage of change or in this case decrease between $b^{3,0}(h_opt(3), h_{opt}(3), h_{opt}(3))$ and $b^{3,0}(\widehat{\mathbf{h}}_{\mathbf{opt}}(3))$ we get:

$$\frac{\left| b^{3,0}(\widehat{\mathbf{h}}_{\mathbf{opt}}(3)) - b^{3,0}(h_opt(3), h_{opt}(3), h_{opt}(3)) \right|}{b^{3,0}(h_opt(3), h_{opt}(3), h_{opt}(3))} \approx 0.051189 \quad \text{so} \quad 5\%$$

Hence, for three steps choosing $\mathbf{h_{opt}}(3)$ is better then choosing three constant steps equal to $h_{opt}(3)$. Now recall that for two steps, we had a decrease of 2% between $b^{2,0}(h_{opt}(2), h_{opt}(2))$ and $b^{2,0}(\mathbf{h_{opt}}(2))$. So it seems like the more N increase the more it is worth to go for variable step sizes instead of constant, but of course we need more tests for larger N in order to confirm.
So the worst-case bound for three steps of GM on a smooth convex function is still an open problem that may be the key to find a general bound for N variable steps.

4.4 N steps of GM

Even though we were not able to find the complete bound for three steps in the previous section, we were still able to find some conclusions on the case of N variable steps of GM. We start by the case of smooth convex functions then the case of strongly convex functions.

4.4.1 Smooth convex functions

Let us start with this conjecture on the shape of the worst-case bound on $f(x_N)-f_*$, the maximum of 2^N functions.

Conjecture 4.5. *Any sequence of iterates $\{x_1, \cdots, x_N\}$generated by N steps steps of gradient method with normalized step sizes $\mathbf{h} = (h_1, h_2, \cdots, h_N) \in [0,2]^N$ on a smooth convex function $f \in \mathcal{F}_{0,L}(\mathbb{R}^d)$ satisfies:*

$$f(x_N) - f_* \leq \frac{LR^2}{2} \max(b_1^N(\mathbf{h}), \cdots, b_{2^N}^N(\mathbf{h})) \tag{4.36}$$

2^N functions corresponding to the fact that each h_i we can be small or big, the domain for which the h_i will be considered small or big depends on N.

Based on our observations in the previous sections, we have the following guesses for some of the functions presented in equation4.36. The most obvious functions are the following mapping the $(S, S, \cdots, S)$ or S^N regime and B^N regime.

$$b_{S^N}^{N,0}(\mathbf{h}) = \frac{1}{1 + \sum_{i=1}^{N} 2h_i} \tag{4.37}$$

$$b_{B^N}^{N,0}(\mathbf{h}) = \prod_{i=1}^{N} (1-h_i)^2 \tag{4.38}$$

The function $b_{S^N}^{N,0}$ is valid for small values of h_i and $b_{B^N}^{N,0}$ for big values of h_i. We started by testing numerically the fit of these functions for some values of N and $\mathbf{h}$. The results are presented in table 4.11 and seems to support our guesses at least for small and big values of h_i.

Notice also that when $h_i = h$ for all i, in the case of constant step sizes, the bound we get is similar to the one presented in conjecture 4.1. Now if we start mixing steps, and suppose first that we choose $\mathbf{h}$ such that all the h_i are small for $i < N$ and h_N is big then then in this case the following function mapping the (S^{N-1}, B) regime seems to take the lead:

$$b_{S^{N-1},B}^{N,0}(\mathbf{h}) = \frac{1}{(1 + \sum_{i=1}^{N-1} h_i)^2} (1-h_N)^2 \tag{4.39}$$

So we tested numerically the maximum of those three functions specially for vectors $\mathbf{h}$ where $b_{S^{N-1},B}^{N,0}$ is bigger then the other two.

N	$\mathbf{h}$	Relative error
S6	$(0.2, 0.3, 0.4, 0.5, 1.5, 1)$	$2e-08$
B6	$(1.86, 1.7, 1.9, 2, 2, 1.8)$	$1e-08$
S25	linspace(0,1,25)	$2e-09$
B25	linspace(1.86,2,25)	$3e-10$
S50	linspace(0,1.5,50)	5e-11
B50	linspace(1.9,2,50)	2e-08

Table 4.11: Error between worst-case bound for the criterion $f(x_N) - f_*$ computed via $\frac{LR^2}{2}\max\left(b_{S^N}^{N,0} b_{B^N}^{N,0}\right)$ and numerically by solving (sdp-PEP), for different number of steps N and Small and Big values of h_i.Results obtained with SeDuMi [Sedum]

N	$\mathbf{h}$	Relative error
5	$(0.2, 0.9, 0.2, 0.3, 2)$	$1e-09$
15	(linspace(0,0.2,14), 2)	$2e-09$
50	(linspace(0,0.05,49), 2)	$3e-09$

Table 4.12: Error between worst-case bound for the criterion $f(x_N) - f_*$ computed via $\frac{LR^2}{2}\max\left(b_{S^N}^{N,0} b_{B^N}^{N,0}, b_{S^{N-1},B}^{N,0}\right)$ and numerically by solving (sdp-PEP), for different number of steps N, small values for all h_i such that $i < N$ and h_N Big.

At this stage one important observation has to be made. Remember when observing three steps for $h_1 = 1$ i.e. small values of h_1 that we observed in figure 4.10 that the the function $b_{S,S,S}^{3,0}$ equation 4.23 seems to be maximal for a wide range of h_2 and h_3 and we couldn't even observe the intersections by staying in $[0, 2]$, which led to the eventuality of optimal step sizes larger then two. It seems like, as N grows the domain where function $b_{S^N}^{N,0}$ dominates the other functions is growing larger, and is practically covering all the values between zero and two for each h_i. That is why we had trouble finding wide range where $b_{S^{N-1},B}^{N,0}$ dominates in table 4.12. That is why we conjectured (4.36) that the domain where h_i is considered small or big depends on the number of steps.
Let us choose the vector $\mathbf{h}$ of normalized step sizes as follows: the set S_1, B and S_2 and take a vector h_i such that the first $|S_1|$ terms are small then the next $|B|$ terms are large and the final $|S_2|$ are small and define the following function:

$$G^N(\mathbf{h}) = \left(\frac{1}{(1+\sum_{i\in S_1} h_i)^2}\right)\prod_{i\in B}(1-h_i)^2\left(\frac{1}{1+2\sum_{i\in S_2} h_i}\right) \tag{4.40}$$

This function resumes what we have done until now, and based on our analysis in the previous chapters. The functions $b_{S^N}^{N,0}$,$b_{B^N}^{N}$ and $b_{S^{N-1},B}^{N}$ that we found and tested earlier for some values of N, can be deduced from G^N. Here is how, from these function, we can reconstruct the bound for two and three steps totally and partially:

- **Two Steps GM** (N=2)
  Choose $(S_1, B, S_2) = (\emptyset, \emptyset, \{1, 2\})$ you get the function $b_1^{2,0}$. Then $(\emptyset, \{1, 2\}, \emptyset)$

you get the function $b_2^{2,0}$, $(\emptyset, \{1\}, \{2\})$ you get $b_3^{2,0}$ and $(\{1\}, \{2\}, \emptyset)$ you get $b_4^{2,0}$.

- **Three Steps GM** (N=3)
  Choose $(\emptyset, \emptyset, \{1,2,3\})$ you get $b^{3,0}_{S,S,S}$, $(\{1,2\}, \{3\}, \emptyset)$ you get $b^{3,0}_{S,B,B}$ then $\{1\}, \{2\}, \{3\}$ for $b^{3,0}_{S,S,B}$, $(\{1\}, \{2,3\}, \emptyset)$ for $b^{3,0}_{S,B,S}$, $(\{2,3\}, \{1\}, \emptyset)$ for $b^{3,0}_{B,S,S}$, $(\{2\}, \{1,3\}, \emptyset)$ for $b^{3,0}_{B,S,B}$ and finally $(\emptyset, \{1,2,3\}, \emptyset)$ for $b^{3,0}_{B,B,B}$. Then for $b^{3,0}_{B,B,S}$ if we follow the previous function we should obtain for $(\emptyset, \{1,2\}, \{3\})$:

$$b_8 = \frac{1}{(1+2h_3)}(1-h_2)^2(1-h_1)^2$$

  which was previously discarded by numerical experiment (see previous section) and plot in figure 4.12.

Hence from the case $N = 3$, we can see that the function G^N is not sufficient to conjecture a general bound for any number of variable step sizes N. Finding the function $b^{3,0}_{B,B,S}$ is necessary but may not be sufficient, because for larger values of N we may have some functions that can't be deduced from the one we already have.

Finally, as we mentioned earlier it seems based on the cases where $N = 2$ and $N = 3$ that the larger N grow the more it is optimal to go for variable step sizes.

4.4.2 Smooth Strongly convex functions

Our numerical experiment for two steps of GM and from conjectures 4.1 and 4.2, we can deduce this conjecture for the worst-case bound on the objective function accuracy.

Conjecture 4.6. *Any sequence of iterates $\{x_1, \cdots, x_N\}$ generated by N steps of gradient method with normalized step sizes $\mathbf{h} = (h_1, h_2, \cdots, h_N) \in [0,2]^N$ on a smooth strongly convex function $f \in \mathcal{F}_{\mu,L}(\mathbb{R}^d)$ satisfies:*

$$f(x_N) - f_* \leq \frac{LR^2}{2} \max(b_1^{N,\mu}(\mathbf{h}), \cdots, b_{2^N}^{N,\mu}(\mathbf{h})) \tag{4.41}$$

2^N functions corresponding to the fact that each h_i we can be small or big, the domain for which the h_i will be considered small or big depends on N and μ.

Here the functions dominating for normalized step sizes h_i small for all i mapping the small step sizes regime denoted by $b^{N,\mu}_{S^N}$ is given by:

$$b^{N,\mu}_{S^N}(\mathbf{h}) = \frac{\kappa}{(\kappa - 1) + \prod_{i=1}^{N}(1-\kappa h_i)^{-2}} \tag{4.42}$$

Then we tested numerically for $L = R = 1$ our function for some values of μ and small normalized step sizes. The results are presented in table 4.13.

N	μ	$\mathbf{h}$	Relative error
3	0.2	(0 1 1.2)	$3e-09$
6	0.2	(1 1.2 0.5 0.7 1.4 1.5)	$1e-09$
30	0.5	(linspace(0,1,15),linspace(0.8,1.5,15))	$5e-09$

Table 4.13: Error between worst-case bound for the criterion $f(x_N) - f_*$ computed via $\frac{1}{2}b^{N,\mu}_{S^N}$ and numerically by solving (sdp-PEP), for different number of steps N, small values of h_i for all i, and different values of μ

.

So numerical tests seems to support this guess. Furthermore notice that for all small $\mathbf{h}$ and any number of step sizes N, we have:

$$\lim_{\mu \to 0} b^{N,\mu}_{S^N}(\mathbf{h}) = b^{N,0}_{S^N}(\mathbf{h})$$

Then as we can expect from the case of two steps of GM, we have the following functions for large normalized step sizes $\mathbf{h}$ (h_i large for all i):

$$b^{N,\mu}_{B^N}(\mathbf{h}) = \prod_{i=1}^{N}(1-h_i)^2 \tag{4.43}$$

And we have obviously for all large $\mathbf{h}$ and for a random number of step sizes:

$$\lim_{\mu \to 0} b^{N,\mu}_{B^N}(\mathbf{h}) = b^{N,0}_{B^N}(\mathbf{h})$$

Then we have tested numerically these functions for big normalized step sizes $\mathbf{h}$ for different number of step sizes and values of μ and $L = R = 1$. The results are presented in table 4.14.

Then if we take the $N-1$ first steps small then the last one big, the following

N	μ	$\mathbf{h}$	Relative error
3	0.2	(1.8 2 1.7)	$2e-10$
6	0.3	(1.7 1.8 1.9 2.0 1.8 1.76)	$3e-11$
30	0.1	(linspace(1.7,2,15), linspace(1.85,2,15))	$9e-10$

Table 4.14: Error between worst-case bound for the criterion $f(x_N) - f_*$ computed via $\frac{1}{2}\max(b^{N,\mu}_{S^N}, b^{N,\mu}_{B^N})$ and numerically by solving (sdp-PEP), for different number of steps N, big values of h_i for all i, and different values of μ

.

function seems to dominate the others:

$$b^{N,\mu}_{S^{N-1},B}(\mathbf{h}) = \left(\frac{\kappa}{(\kappa-1) + \prod_{i=1}^{N-1}(1-\kappa h_i)^{-1}}\right)^2 (1-h_N)^2 \tag{4.44}$$

N	μ	$\mathbf{h}$	Relative error
3	0.1	(0.1 0.2 1.9)	$2e-08$
6	0.3	(0.1 0.22 0.5 0.2 0.3 2)	$5e-10$
30	0.3	(linspace(0,0.7,15) linspace(0.6,1,14) 1.9)	$1e-07$

Table 4.15: Error between worst-case bound for the criterion $f(x_N) - f_*$ computed via $\frac{1}{2}\max(b^{N,\mu}_{S^N}, b^{N,\mu}_{B^N}, b^{N,\mu}_{S^{N-1},B})$ and numerically by solving (sdp-PEP), for different number of steps N, small values of h_i for all $i < N$, big values of h_N and different values of μ

.

We tested this function numerically for different values of N, μ, values of $\mathbf{h}$ for which $\max(b^{N,\mu}_{S^N}, b^{N,\mu}_{B^N}, b^{N,\mu}_{S^{N-1},B}) = b^{N,\mu}_{S^{n-1},B}$ and $L = R = 1$. The results are shown in table 4.15

Once again from these few experiments, similarly to the case of smooth convex functions, we observe that hat when N grows larger and $h_i \in [0, 2]$, the region where $b^{N,\mu}_1$ is the biggest of the three is growing larger, suggesting also optimal step sizes larger then two. Finally observe also that for all N and $\mathbf{h}$ we have:

$$\lim_{\mu \to 0} b^{N,\mu}_{S^{N-1},B}(\mathbf{h}) = b^{N,0}_{S^{N-1},B}(\mathbf{h})$$

Then here is a function that generalize the function G^N (equation (4.40)) to the case of smooth strongly convex functions:

$$G^{N,\mu}(\mathbf{h}) = \left(\frac{\kappa}{(\kappa-1) + \prod_{i \in S_1} (1-\kappa h_i)^{-1}} \right)^2 \prod_{i \in B} (1-h_i)^2 \left(\frac{\kappa}{(\kappa-1) + \prod_{i \in S_2} (1-\kappa h_i)^{-2}} \right) \tag{4.45}$$

Once again from this function we can generate the worst-case bound for two steps of GM on smooth strongly convex functions found previously. We guess that, as it was the case for smooth convex functions, these functions are not sufficient to determine the worst-case bound for N variable step sizes of GM. Note also that we have for all $\mathbf{h}$ and N:

$$\lim_{\mu \to 0} G^{N,\mu}(\mathbf{h}) = G^N(\mathbf{h})$$

This section illustrate how we can easily deduce bounds for smooth strongly convex functions just by using the bounds for smooth convex functions.

Summary

The problem of finding the worst-case bound on objective function accuracy for N variable step sizes of GM on smooth strongly convex and smooth convex functions is still an open problem.
Nevertheless our results enabled us to conjecture some interesting results.

- The worst-case bound seems to be the maximum of 2^N functions, for smooth convex and strongly convex functions. These functions as we have seen for two steps are not necessarily convex as one can expect when seeing the bounds in conjecture 4.1 and 4.2 or in the bounds for one step (4.1) and (4.2).
- It seems like the functions involved in the bound for strongly convex functions converges to their equivalent for smooth convex functions, based on the bounds we have until now.
- We have provided two functions G^N (4.40) and $G^{N,\mu}$ (4.45) that will partially help us to determine some of the functions for the worst-case bound for a new N. These two functions can be deduced directly from the cases we have studied until now, specifically the cases of two and three steps.
- For some values of N the optimal step sizes may involve some values of h_i larger then two. This was the case for three steps, and we discussed this in the final section during the small experiments conducted for N steps.
- When we compared the optimal performances of the GM obtained for fixed and different step sizes to the one obtained for fixed and constant step sizes, we noticed that the variable steps strategy produces smaller worst-case bounds for two and three steps. It seems like the decrease between the two optimal worst-case are increasing with N, suggesting that as N grows larger it is more interesting to use a fixed variable step sizes strategy instead of fixed constant one. But the observed decrease are not significant enough to constitute a strong numerical argument.

All of these conjectures where based on the conjectures we found in the literature for constant step sizes (conjectures 4.1 and 4.2) and the cases of two and three variable step sizes of GM. We are confident that closing the case of three steps will give us new information's. Nevertheless we can't exclude the possibility that for each N we will have at least one new function that is quite different from the previous ones. This was the case for our function $b_4^{2,0}$ (4.9) representing (S, B) regime for two steps of GM, and will most probably be the case of the function $b_{B,B,S}^{3,0}$ for three steps.
As for this master theses, we will focus in the remaining on providing proofs of some of the conjectures made in this chapter.

Chapter 5

Proofs for worst-case bounds

Introduction

In this chapter we will try to prove the results we conjectured in the previous chapter or at least provide some new information and concepts that will improve the methodology for the proofs. First of all we go back to a proof presented in [Theorem 4.14 Tay17, p.105] in order to reproduce it for smooth convex functions. Going through the details of this proof will give us some important insights, that will lead us to some new concepts that could help simplify the proofs. Finally, we will apply those concepts and methods to the case of two steps of GM, where we have guessed all the functions involved in the worst-case bound.

5.1 One step of GM

Here is the version of [Theorem 4.14 Tay17, p.105] for smooth convex functions that we will focus on in this first section.

Theorem 5.1. *Any iterate x_1 produced by the gradient method with normalized step sizes $0 \leq h \leq 2$ on a smooth convex function satisfies:*

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

which implies that the optimal step size for one step is $h_{opt} = 1.5$.

We already discussed this theorem in example 4.1, and recall that values of h smaller then h_{opt} will be considered small and we will be in the small step regime since $(1+2h)^{-1}$ is maximal and h will be considered big if it is grater then h_{opt} and the function $(1-h)^2$ will be maximal so that we are in the big regime.

The existing proof is based on the dual variables of the interpolation inequalities in the sdp formulation (sdp-PEP). The idea is to choose the correct value of these variables and sum up the inequalities multiplied by their respective variable. It seems like their is an infinity of proofs possible because of the non uniqueness of the dual variables. The toolbox returns values of the dual variables and the corresponding interpolation inequality presented in corollary 2.9 denoted by $I(i,j)$ where i refers to x_i and j refers to x_j and are written as in [referencer dans chap3]. We remind the reader of the meaning of the notation $I(i,j)$:

$$I(i,j) : f(x_i) - f(x_j) + g_i^T(x_j - x_i) + \frac{1}{2L} \|g_i - g_j\|_2^2 \leq 0 \tag{5.1}$$

Notice that for one step of GM we will get six inequalities as we can see in table 5.1, where we present the inequalities and the notations we will use for their dual variables.

Inequalities	I(0,*)	I(0,1)	I(1,*)	I(1,0)	I(*,0)	I(*,1)
Dual Variables	λ_0	λ_1	λ_2	λ_3	λ_4	λ_5

Table 5.1: Interpolation inequalities and corresponding dual variables for one step

We will start by illustrating that various proofs can be obtained, using the non-uniqueness of optimal dual values, for the particular case of $h = 1$. Then we will try to provide a generic proof and see how we have the transition between the small step sizes and large step sizes.

Case of $\mathbf{h = 1}$

In table 5.2, we are giving different values of the dual variables and four of the interpolation inequalities that will lead to a proof of the inequality in theorem 5.1 for $h = 1$:

$$f(x_1) - f(x_*) \leq \frac{L\|x_0 - x_*\|^2}{2}\left(\frac{1}{6}\right)$$

Version I in the proof is deduced directly from looking at the values of the dual variables obtained when we solved (sdp-PEP) for $N = 1$ and $R = L = 1$. As for version II, we took the values of the dual variables used [Appendix A Tay17, p.105] with $\mu = 0$ and $h = 1$. Since these two proofs are valid, this was the first indication of the non-uniqueness of the proof. When we look closer to the values of the dual variables in version I and II, we can see that they come from one parametric proof corresponding and one possible of writing this parametric proof is as we have done in version III by introducing this parameter ϵ. In fact for $\epsilon = 2/3$ we get version I obtained via the toolbox and for $\epsilon = 1$ we get Version II.

	Dual variables			
Inequalities	I(0,*)	I(0,1)	I(1,*)	I(1,0)
Version I	1/3	1/3	1/6	1/6
Version II	1/2	0	1/2	1/2
Version III	$\epsilon/2$	$1-\epsilon$	$1-\epsilon/2$	$1-\epsilon/2$

Table 5.2: Proofs for **h=1**

We will only present the proof for the parametric version, as the other two stems from this parametric proof.

Proof. For simplicity we will take $L = 1$ and $f(x_*) = g(x_*) = x_* = 0$. Summing up the inequalities as follows:

$$(\epsilon/2)I(0,*) + (1-\epsilon)I(0,1) + (1-\epsilon/2)I(1,*) + (1-\epsilon/2)I(1,0)$$

We get after simplification:

$$f(x_1) \leq \frac{\|x_0\|^2}{6} - \frac{1}{6}\left\|x_0 - 3(\frac{\epsilon}{2}g_0 + (1-\frac{\epsilon}{2})g_1)\right\|^2 - p(\epsilon)\|g_0 - g_1\|^2$$

With $p(\epsilon) = \frac{\epsilon}{2}\left(1 - \frac{3\epsilon}{4}\right)$. For any $\epsilon \in [0, 4/3]$ we get $p(\epsilon) \geq 0$ and the proof works. Hence we can get many proofs, and this is an illustration of this multiplicity of proofs. □

Now we shall try to generalize this argument and provide a parametric proof for any **small h**. Notice that **Version II** presented in table 5.2 is also valid for any small h.

Small step sizes

Inspired by the case for $h = 1$, we will try to prove the version of Theorem 5.1 for small step sizes hence we will try to prove the inequality:

$$f(x_1) - f(x_*) \leq \frac{L\|x_0 - x_*\|_2^2}{2}\frac{1}{2h+1}$$

In the following proof we will use the notations introduced in table 5.1 for the interpolation inequalities and the associated dual variables.

Proof. Once again for simplicity we will consider $L = 1$ and $f(x_*) = g_* = x_* = 0$. We start by considering that for small step sizes $\lambda_4 = \lambda_5 = 0$ and the choice will be discussed later. Then we will simply sum up the remaining inequalities multiplied by their respective dual variables:

$$\lambda_0 I(0,*) + \lambda_1 I(0,1) + \lambda_2 I(1,*) + \lambda_3 I(1,0)$$

Using the relationship $x_1 = x_0 - hg_0$ we simplify and we get the following:

$$\begin{aligned}
(\lambda_0 + \lambda_1 - \lambda_3)f(x_0) + (\lambda_2 + \lambda_3 - \lambda_1)f(x_1)& \\
-(\lambda_0 g_0 + \lambda_2 g_1)^T(x_0)& \\
+(h\lambda_2 + (1-h)\lambda_3 - \lambda_1)g_0^T g_1& \\
+(\frac{\lambda_0}{2} + \lambda_1(\frac{1}{2} - h) + \frac{\lambda_3}{2})\|g_0\|_2^2& \\
+\frac{1}{2}(\lambda_1 + \lambda_2 + \lambda_3)\|g_1\|_2^2 &\leq 0
\end{aligned}$$

At this stage we should impose the following conditions on the lambdas:

$$\boxed{\begin{aligned}\lambda_3 &= \lambda_0 + \lambda_1 \\ \lambda_2 &= 1 - \lambda_0\end{aligned}} \tag{5.2}$$

By doing so we will get $\lambda_0 + \lambda_1 - \lambda_3 = 0$ and $\lambda_2 + \lambda_3 - \lambda_1 = 1$. Then adding and retrieving $\|x_0\|_2^2 \frac{1}{2(2h+1)}$ we get :

$$f(x_1) - \frac{\|x_0\|_2^2}{2(2h+1)} \leq -\frac{\|x_0\|_2^2}{2(2h+1)} - \Big(\lambda_0 + (1-h)\lambda_1\Big)\|g_0\|_2^2 - \Big(\frac{1}{2} + \lambda_1\Big)\|g_1\|_2^2 + \Big(\lambda_0 g_0 + (1-\lambda_0)g_1\Big)^T x_0 - \Big(h + (h-1)\lambda_1 - \lambda_1 - \lambda_0\Big) g_0^T g_1$$

Then we notice that :

$$\Big(\lambda_0 g_0 - (1-\lambda_0)g_1\Big)^T x_0 = \frac{\|x_0\|_2^2}{2(2h+1)} + \Big(\frac{2h+1}{2}\Big)\|\lambda_0 g_0 - (1-\lambda_0)g_1\|_2^2 - Qf_1$$

Where Qf_1 is the following semi definitive quadratic form:

$$Qf_1 = \frac{1}{2(2h+1)}\|x_0 - (2h+1)(\lambda_0 g_0 - (1-\lambda_0)g_1)\|_2^2$$

Plugging everything in and regrouping terms we get:

$$\begin{aligned} f(x_1) - \frac{\|x_0\|_2^2}{2(2h+1)} \leq &- \underbrace{\Big(-\Big(\frac{2h+1}{2}\Big)\lambda_0^2 + (1-h)\lambda_1 + \lambda_0\Big)}_{A}\|g_0\|_2^2 \\ &- \underbrace{\Big(-\Big(\frac{2h+1}{2}\Big)\lambda_0^2 + (2h+1)\lambda_0 - h + \lambda_1\Big)}_{B}\|g1\|_2^2 \\ &- \underbrace{\Big((2h+1)\lambda_0^2 - (2h+1)\lambda_0 + h + (h-1)\lambda_1 - \lambda_1 - \lambda_0\Big)}_{-(A+B)} g_0^T g_1 - Qf_1 \end{aligned}$$

Hence we get:

$$f(x_1) - \frac{\|x_0\|_2^2}{2(2h+1)} \leq \underbrace{(A+B)g_0^T g_1 - A\|g_0\|_2^2 - B\|g_1\|_2^2}_{Qf_2} - Qf_1$$

In what follows we should try to reduce Qf_2. To do so one simple approach is to find α and β such that we can simplify the terms in $\|g_0\|_2^2$ and $\|g_1\|_2^2$ by rewriting $g_0^T g_1$ as follows:

$$g_0^T g_1 = \frac{\alpha}{2\beta}\|g_0\|_2^2 + \frac{\beta}{2\alpha}\|g_1\|_2^2 - \frac{1}{2\alpha\beta}\|\alpha g_0 - \beta g_1\|_2^2$$

We should impose that $(\alpha, \beta) \neq 0$ and the following holds:

$$\begin{cases} (A+B)\alpha - 2A\beta = 0 \\ -2B\alpha + (A+B)\beta = 0 \end{cases}$$

The previous linear system in (α, β) should have an infinity of solutions hence we need that the following holds:

$$\begin{vmatrix} A+B & -2A \\ -2B & A+B \end{vmatrix} = (A-B)^2 = 0$$

Hence from the previous equation we get a relationship between λ_0 and λ_1:

$$\boxed{\lambda_1 = 1 - 2\lambda_0}$$

Under these conditions we can simplify our inequality and we get:

$$f(x_1) \leq \frac{\|x_0\|_2^2}{2(2h+1)} - A(\lambda_0)\,\|g_0 - g_1\|_2^2 - Qf_1$$

Last but not least we find the range of λ_0 by imposing that $A(\lambda_0) \leq 0$. Notice that $A(\lambda_0)$ is a simple degree two polynomial in λ_0 with discriminant $\Delta = 3 - 2h$. We start by checking values of h for which the discriminant is positive. In fact, if it is not the case, A will take the sign of $-(2h+1)/2$ which is negative for $h \geq 0$. So we get the following condition on h:

$$\boxed{h \leq \frac{3}{2} = h_{opt}}$$

Hence for any h in the small step size regimes i.e. $h \in [0, \frac{3}{2}]$ we get the following range for λ_0:

$$\lambda_0 \in \Big[\underbrace{\max(\frac{(2h-1) - \sqrt{3-2h}}{2h+1}, 0)}_{LB}, \underbrace{\frac{(2h-1) + \sqrt{3-2h}}{2h+1}}_{UB}\Big] = \mathcal{V}_s$$

Then recall that all the lambdas are dual variable hence they should be positive. Checking values of λ_0 for which it is the case we obtain the condition $\lambda_0 \leq 1/2$. Then in the plot of figure 5.1, you can see that the upper bound UB is always greater then 0.5 for all $h \leq 1.5$, so that we replace UB by 0.5. Hence the final set of valid choices for λ_0 is given by:

$$\boxed{\lambda_0 \in \Big[\max\Big(\frac{(2h-1) - \sqrt{3-2h}}{2h+1}, 0\Big), \frac{1}{2}\Big] = \mathcal{V}_s} \tag{5.3}$$

□

As a summary we have proved that for all $h \in [0, 1.5]$ we can prove the inequality of theorem 5.1 with any value of the dual variable λ_0 in the interval $\mathcal{V}_s$ defined in (5.3). The conditions presented in table 5.3 gives you the value of the three remaining dual variables.

Finally, notice that for $\lambda_0 = \frac{1}{2} \in \mathcal{V}_s$ for any $h \in [0, 1.5]$, we get **Version II** which works also for any small h as we mentioned earlier. Then it is interesting to notice the limit case for $h = 3/2$ interval $\mathcal{V}_s$ reduces to $\{1/2\}$ as you can see graphically in figure 5.1.

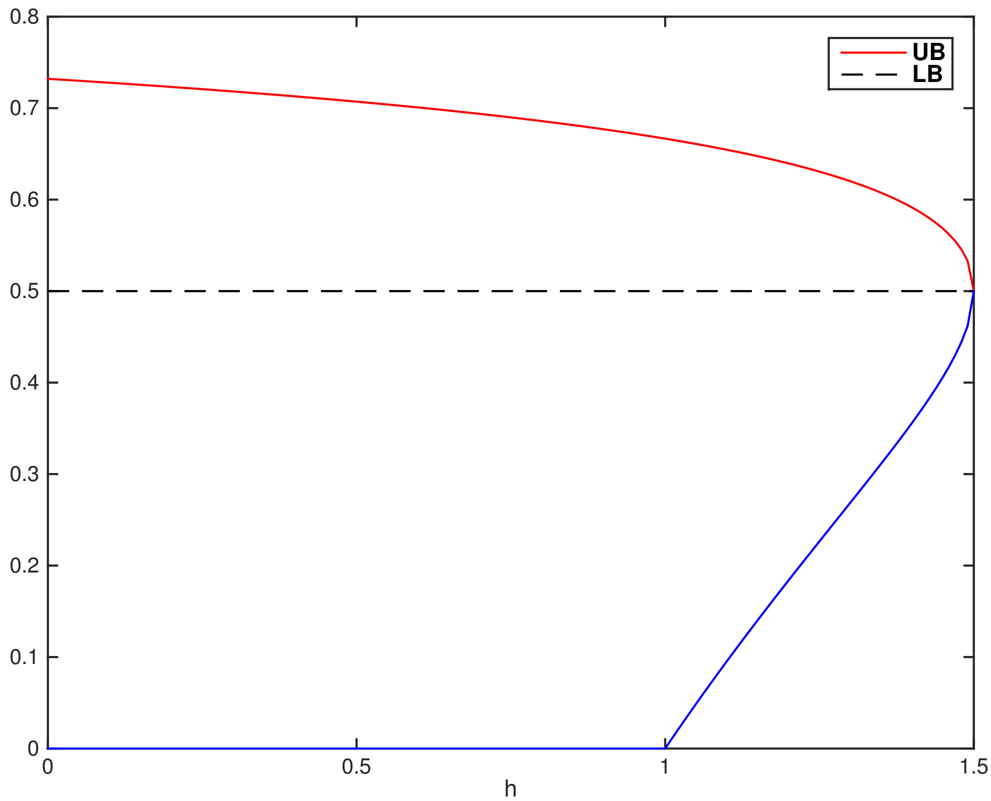


Figure 5.1: Plots of the upper bound UB and the lower bound LB of the interval $\mathcal{V}_s$.

Inequalities	Dual Variabels
I(0,*)	λ_0
I(0,1)	$\lambda_1 = 1 - 2\lambda_0$
I(1,*)	$\lambda_2 = 1 - \lambda_0$
I(1,0)	$\lambda_3 = 1 - \lambda_0$
I(*,0)	$\lambda_4 = 0$
I(*,1)	$\lambda_5 = 0$

Table 5.3: Small step sizes proof

Large step sizes

Inspired by the parametric proof we provided for small step sizes, we will try to show that for large step sizes their will be an infinity of proofs for the version of theorem 5.1 for large step sizes hence we will prove the following inequality :

$$f(x_1) - f(x_*) \leq \frac{L \left\| x_0 - x_* \right\|_2^2}{2} (h-1)^2$$

As previously, we started by looking at the dual values considered in the proof in [Appendix Chapter 4 Tay17, p.105-106], where setting μ to zero you get the values presented in **Version I** of table 5.4. Then we also tried guess the values for the dual variables returned by the toolbox, but this time we where not able to guess them easily as is seems like they are functions of h. Nevertheless they enables us to consider a proof with $\lambda_1 = \lambda_5 = 0$. The parametric proof we will consider is the one denoted by **Version II** in table 5.4.

	Dual variables			
Inequalities	I(0,*)	I(1,*)	I(1,0)	I(*,0)
Version I	h-1	2-h	h-1	0
Version II	λ_0	λ_2	λ_3	λ_4

Table 5.4: Proofs for Large step sizes

Proof. Again for simplicity we will consider $L = 1$ and $f(x_*) = g_* = x_* = 0$. We will consider $\lambda_1 = \lambda_5 = 0$ and this choice will be explained later similarly to the choice of $\lambda_4 = \lambda_5 = 0$ in the proof of small step sizes. Then we will simply sum up the remaining inequalities multiplied by their respective dual variable:

$$\lambda_4 I(*,0) + \lambda_0 I(0,*) + \lambda_2 I(1,*) + \lambda_3 I(1,0)$$

After simplification we get:

$$\begin{aligned} &(\lambda_0 - \lambda_4 - \lambda_3) f(x_0) + (\lambda_2 + \lambda_3) f(x_1) \\ &+\frac{1}{2}(\lambda_4 + \lambda_0 + \lambda_3) \|g_0\|_2^2 + \frac{1}{2}(\lambda_2 + \lambda_3) \|g_1\|_2^2 \\ &-(\lambda_0 g_0 + \lambda_2 g_1)^T x_0 + (h\lambda_2 + h\lambda_3 - \lambda_3) g_0^T g_1 \leq 0 \end{aligned}$$

Then we will impose:

$$\boxed{\begin{aligned} \lambda_4 &= \lambda_0 + \lambda_2 - 1 \\ \lambda_3 &= 1 - \lambda_2 \end{aligned}} \tag{5.4}$$

By doing so we will get $\lambda_0 - \lambda_4 - \lambda_3 = 0$ and $\lambda_2 + \lambda_3 = 1$ plugging everything in the previous inequality and adding and retrieving $\frac{\|x_0\|_2^2 (h-1)^2}{2}$ we get :

$$\begin{aligned} f(x_1) - \frac{\|x_0\|_2^2}{2} X^2 \leq &- \frac{\|x_0\|_2^2}{2} X^2 - \lambda_0 \|g_0\|_2^2 - \frac{1}{2} \|g_1\|_2^2 \\ &+ (\lambda_0 g_0 + \lambda_2 g_1)^T x_0 - (h\lambda_2 + h\lambda_3 - \lambda_3) g_0^T g_1 \end{aligned}$$

Where $X = h - 1$. Then we notice that:

$$(\lambda_0 g_0 + \lambda_2 g_1)^T x_0 = \frac{\|x_0\|_2^2}{2} X^2 + \frac{1}{2X^2} \|\lambda_0 g_0 + \lambda_2 g_1\|_2^2 - \underbrace{\frac{X^2}{2} \left\| x_0 - \frac{1}{X^2}(\lambda_0 g_0 + \lambda_2 g_1) \right\|_2^2}_{Qf_3}$$

Plugging everything in the inequalities and regrouping terms we get:

$$\begin{aligned} f(x_1) - \frac{\|x_0\|_2^2}{2}X^2 &\leq \underbrace{\left(\frac{\lambda_0^2}{2X^2} - \lambda_0\right)}_{q_1(\lambda_0)} \|g_0\|_2^2 + \underbrace{\left(\frac{\lambda_2^2}{2X^2} - \frac{1}{2}\right)}_{q_2(\lambda_2)} \|g_1\|_2^2 \\ &+ \underbrace{\left(\frac{\lambda_0\lambda_2}{X^2} - (\lambda_2 + X)\right)}_{q_3(\lambda_0,\lambda_2)} g_0^T g_1 - Qf_3 \\ &\leq Qf_4 - Qf_3 \end{aligned}$$

We start by defining vector $\mathbf{g}$ and matrix $\mathbf{Q}$ as follows:

$$\mathbf{g} = \begin{pmatrix} g_0 \\ g_1 \end{pmatrix} \qquad\qquad \mathbf{Q} = \begin{pmatrix} q_1(\lambda_0) & \frac{q_3(\lambda_0,\lambda_2)}{2} \\ \frac{q_3(\lambda_0,\lambda_2)}{2} & q_2(\lambda_2) \end{pmatrix}$$

Then $Qf_4 = \mathbf{g}^T\mathbf{Q}\mathbf{g}$ and we need it to be semi positive definite. Hence, we need to make sure that $Det(\mathbf{Q}) \geq 0$ and $trace(\mathbf{Q}) \leq 0$ and we get the following:

$$\begin{cases} -\left(\frac{\lambda_0}{2X} - \frac{(\lambda_2+X)}{2}\right)^2 & \geq & 0 \\ q_1(\lambda_0) + q_2(\lambda_2) & \leq & 0 \end{cases}$$

Hence after solving this system we get a relation between λ_2 and λ_0 as well as the domain of validity of λ_0:

$$\boxed{\begin{aligned} &\lambda_2 = \frac{\lambda_0}{(h-1)} - (h-1) \\ &\lambda_0 \in \left[0, 2(h-1)^2\right] \end{aligned}}$$

In this case the sign of the lambdas does not depend only on λ_0. In order to have all dual variables positive, λ_0 should verify:

$$\begin{cases} \lambda_0 & \leq & 2(h-1)^2 & (\lambda_0 \geq 0) \\ \lambda_0 & \geq & (h-1)^2 & (\lambda_2 \geq 0) \\ \lambda_0 & \leq & h(h-1) & (\lambda_3 \geq 0) \\ \lambda_0 & \geq & h-1 & (\lambda_4 \geq 0) \end{cases}$$

for $0 < h < 1$ ($h \neq 1$ is a trivial condition), we cannot find any λ_0 positive verifying these conditions because $(h-1) \leq 0$ and $h(h-1) \leq 0$ hence we can't verify :

$$h - 1 \leq \lambda_0 \leq h(h-1)$$

Then for $1 < h < 1.5$ we have that $2(h-1)^2 \leq h-1$ hence we can't verify:

$$h - 1 \leq \lambda_0 \leq 2(h-1)^2$$

But for $h \geq 3/2$ all of these conditions are satisfied and we get the final domain of validity of λ_0

$$\boxed{\lambda_0 \in \left[(h-1), 2(h-1)^2\right] = \mathcal{V}_b} \tag{5.5}$$

hence under these conditions we can simplify we get:

$$f(x_1) \leq \frac{\|x_0\|_2^2}{2}(h-1)^2 + \underbrace{Qf_4}_{s.d.n} - \underbrace{Qf_3}_{s.d.p} \leq \frac{\|x_0\|_2^2}{2}(h-1)^2$$

where s.d.n and s.d.p. stands respectively for semi definite negative and semi definite positive. □

Notice that for $\lambda_0 = h - 1 \in \mathcal{V}_b$ we get Version I of the proof for large step sizes presented in table 5.4. Furthermore, for $h = h_{opt}$ the interval $\mathcal{V}_b$ reduces to $\{0.5\}$ and we get the same values of the dual variables we used in the proof for small step sizes.

Summary

We now have a generic proof for the Large and big step sizes and we can see that both proof are disjoint, in the sense that the inequality for $0 \leq h \leq 1.5$ holds when the other does not and reciprocally. Table 5.5 bellow summarizes all the choices available for dual variables for which we can prove theorem 5.1.

	Dual Variables		
Inequalities	$0 \leq h \leq 1.5$	h=1.5	$1.5 \leq h \leq 2$
I(0,*)	$\lambda_0 \in \mathcal{V}_s$	$\mathcal{V}_s = \{0.5\} = \mathcal{V}_b$	$\lambda_0 \in \mathcal{V}_b$
I(0,1)	$1 - 2\lambda_0$	0	0
I(1,*)	$1 - \lambda_0$	0.5	$\frac{\lambda_0}{(h-1)} - (h-1)$
I(1,0)	$1 - \lambda_0$	0.5	$h - \frac{\lambda_0}{(h-1)}$
I(*,0)	0	0	$\lambda_0 + \frac{\lambda_0}{(h-1)} - h$
I(*,1)	0	0	0

Table 5.5: Possible values of the dual variable λ_0 associated with the inequality $I(0,*)$ and the corresponding proof of the exact worst-case bound for one step of GM on smooth convex functions.

We have thus showed that there is an infinity of proofs we can get but they all lead to the same conclusion. Notice that the proof can be adapted to any values of L and R we need to replace h by $\gamma = \frac{h}{L}$ for the values of the lambdas in the case of large step sizes.

To end this section, let's go back on the choices we made in the previous proofs, $\lambda_4 = \lambda_5 = 0$ for small step sizes and $\lambda_1 = \lambda_5 = 0$ for large step sizes. These choices were made because of observations in the toolbox. But we can eventually consider another proof where these dual variables are non zero and this will work perfectly fine, ignoring the difficulty of the algebra in the proofs. Maybe for one step of GM it is still possible to handle a proof that uses the six inequalities but for more then one step, the task is much more complicated.

Based on this argument an idea of minimal proof arises, minimal in the sense that it uses the smallest number of interpolations inequalities. In order to determine this minimal number of inequalities, one idea is to use the Pesto-toolbox [THG17a] as explained in the next section.

5.2 Minimal proofs

Generally for any N steps of GM there may be an infinity of proofs. It is actually the case for one step as we showed explicitely in the previous section, and for two steps as our numerical experiments showed. Among this large number of proofs, we may want to choose the most tractable proof as discussed in the summary of the previous section. First let us start by defining Ω the set of interpolation inequalities defined in corollary 2.9 and we write them as in 5.1. Then consider now a variation of sdp-PEP that we will denote by $\omega_{\mu,L}^{sdp}(R, H, N, b, C, S)$ where we have added a set $S \subset \Omega$. So this new problem is a relaxation of sdp-PEP where we only use a subset of the interpolation inequalities.

We can represent a proof of an exact worst-case bound by a set of interpolation inequalities $\mathcal{S} \subset \Omega$. In the next definition we will formally define the concept of minimal proof.

Definition 5.1. *Consider N step of a fixed step first order method and $b^{N,\mu}$ the exact worst-case bound on the objective function accuracy for a function $f \in \mathcal{F}_{\mu,L}(\mathbb{R}^d)$. A proof $\mathcal{S}$ will be considered a **minimal proof** for $b^{N,\mu}$ if for all step sizes $\mathbf{H}$, we cannot prove the exact worst-case bound with less then $|\mathcal{S}|$ inequalities.*

Sometimes, a weaker notion of minimal proof can be useful. The main restriction in the previous definition is the fact that the proof should hold for all combination of step sizes. So in the next definition we will relax the definition and consider a subset of combination of step sizes.

Definition 5.2. *Consider N steps of a fixed step sizes first order method with an exact worst-case bound denoted by $b^{N,\mu}$ on the objective function accuracy for a function $f \in \mathcal{F}_{\mu,L}(\mathbb{R}^d)$. We say that a proof $\mathcal{S}_l \subset \Omega$ is a local minimal proof for $b^{N,\mu}$ if there exists at least one combination of step sizes $\mathcal{V}$, such that $\mathcal{S}_l$ is a minimal proof for the worst-case bound for all the combination in $\mathcal{V}$.*

The subset of combination of step sizes $\mathcal{V}$ can contain only one H, but of course practically it is interesting to have as much as possible.

Example 5.1. *Consider one step of GM. As we can check in table 5.5,the minimal proof for all $h \in [0, 2]$ is given by:*

$$\mathcal{S} = \{I(0, *), I(1, *), I(1, 0)\}$$

Also looking at table 5.5 one possible local minimal proof is:

$$\mathcal{S}_l = \{I(1, *), I(0, 1), I(1, 0)\}$$

Notice that this local minimal proof corresponds to the choice of $\lambda_0 = 0$ referring to table 5.5. It is actually valid if you look at figure 5.1, for $0 \leq h \leq 1$.

Until now we do not have a new method to prove formally that a set of inequalities is indeed minimal. The only way we have until now to prove it is to see how we can reduce a proof with more then the inequalities needed to the minimal proof. The trick is to exploit the toolbox to estimate those proofs, and if after writing them down it works then no need for more. In order to determine one possible numerical procedure to estimate the minimal proof, we define the concept of essential inequalities in the next section.

5.2.1 Essential inequalities.

We will define another concept that will help us classify the interpolation inequalities.

Definition 5.3. *Consider N steps of a fixed steps first order black box method and the performance criterion $\mathcal{P}$ defined by b and C. An interpolation inequality $I(i,j) \in \Omega$ will be considered essential if and only if there exist at least one* $\mathbf{H}$ *such that we have:*

$$\omega_{\mu,L}^{sdp}(R, H, N, b, C, \Omega \setminus \{I(i,j)\}) \neq w_{\mu,L}^{sdp}(R, H, N, b, C)$$

The following proposition linking the minimal proof to the essential inequalities, holds true.

Proposition 5.2. *Consider N steps of a fixed steps first order black box method. Let $\mathcal{S}$ be a minimal proof and $\mathcal{E}$ be the set of essential inequalities for all possible step sizes. Then the following holds true:*

$$\mathcal{S} = \mathcal{E} \tag{5.6}$$

Proof. First consider $I(i,j)$ in $\mathcal{S}$ and not in $\mathcal{E}$. Then this inequality is not essential, and we can create a new set $\tilde{\mathcal{S}} = \mathcal{S} \setminus \{I(i,j)\}$ such that $|\tilde{\mathcal{S}}| \leq |\mathcal{S}|$ and we also have for all step sizes

$$\omega_{\mu,L}^{sdp}(R, H, N, b, C, \tilde{\mathcal{S}}) = w_{\mu,L}^{sdp}(R, H, N, b, C)$$

But since $\mathcal{S}$ is a minimal proof the previous statement cannot hold so $I(i,j) \in \mathcal{E}$. Now consider an interpolation inequality $I(i,j)$ in $\mathcal{E}$ and suppose it is not in $\mathcal{S}$. So there exists at least one combination of steps for which we can solve the problem using $\mathcal{S}$, so without $I(i,j)$ which is impossible since it is an essential inequality. □

Thanks to proposition 5.2, we know that finding the set of essential inequalities for all $\mathbf{H}$ will give us a minimal proof.
The next issue is to test these concepts in practice. So we went back to the Pesto-toolbox [THG17a] and we modified a little bit the function `GetInterp` in the `functionClass.m`. This function is the one that will add the interpolation

inequalities in the construction phase of sdp-PEP. So we decided to add the interpolation inequalities manually. The Idea is that we can check if an inequality is essential by removing it and seeing if there exist at least one $\mathbf{H}$ such that $\omega^{sdp}_{\mu,L}(R,H,N,b,C,\Omega\setminus\{I(i,j)\}) \neq w^{sdp}_{\mu,L}(R,H,N,b,C)$. So now we define a numerical experiment that will provide an estimation of the set of essential inequalities.
We start by defining a set of testing points that will be denoted by T. This set contains chosen $\mathbf{H}$ that will be used to check if an inequality is essential or not. Define $E=\emptyset$ and $Res=\Omega$. Then while Res in not empty, consider $I(i,j)$ in Res. If there exist $\mathbf{H}$ in T such that $\omega^{sdp}_{\mu,L}(R,H,N,b,C,\Omega\setminus\{I(i,j)\}) \neq w^{sdp}_{\mu,L}(R,H,N,b,C)$, then you can update E as $E=E\cup\{I(i,j)\}$. Next remove the $I(i,j)$ from Res and go on.

$$
\begin{array}{l}
(\mathbf{0})\ E=\emptyset\ ,\ Res=\Omega \\
(\mathbf{1})\ \text{While } Res\neq\emptyset \\
\quad\quad \text{Choose } I(i,j)\in Res. \\
\quad\quad \text{For all} \mathbf{H}\in T: \\
\quad\quad\quad\quad \text{If } \omega^{sdp}_{\mu,L}(R,H,N,b,C,\Omega\setminus\{I(i,j)\}) \neq w^{sdp}_{\mu,L}(R,H,N,b,C) \\
\quad\quad\quad\quad\quad\quad E=E\cup\{I(i,j)\} \\
\quad\quad Res=Res\setminus\{I(i,j)\} \\
(\mathbf{2})\ \text{Return } E
\end{array}
\tag{E-Proc}
$$

The resulting set E is such that $E\subset\mathcal{E}$, thus we provide an estimation of the real essential set. This is actually trivial since we are just testing the inequalities on $\mathbf{H}$ in T and not all $\mathbf{H}$. So it is important here to notice that choosing the testing points is really important. One should try to include as much points as possible, specially points where we may have problems. Let us quickly provide an example of this procedure for one step of GM:

Example 5.2. *For one step of GM let us apply the procedure E-Proc.*
<u>Step 0</u> we define T by choosing three small steps smaller then $h_{opt}(1)=1.5$, the optimal step size and three large step between 1.5 and 2. So we have :

$$
\begin{aligned}
T &= \{0.5,1,1.3,1.5,1.6,1.8,2\} \\
Res &= \Omega = \{I(i,j)\mid i,j\in\{0,1,*\}\} \\
E &= \emptyset
\end{aligned}
$$

<u>Step 1</u> Here we will summarize the results of the investigations in table 5.6 where for each inequality, we mention "E" if it is indeed essential and we show one value of $h\in T$ for which we verify the definition. And for the non essential inequalities, we just mention "NE" for not essential ***with respect to our testing set*** *T.*

When we tested the inequality $I(0,*)$ for $h=0.5$ we did not verify the definition of essential so if our sample T did not contain the appropriate values, $I(0,*)$ would have been considered not essential. For one step it may be easy to choose the appropriate data set, but for more it is not trivial. Notice how we rely on the analysis we made for one step in example 4.1 to choose the testing points in a way to cover all the possible regimes and their intersections.
<u>Step 2</u> $E=\{I(0,1),I(,*),I(1,0)\}$

Inequalities	$I(*,0)$	$I(*,1)$	$I(0,1)$	$I(0,*)$	$I(1,*)$	$I(1,0)$
Result	NE	NE	NE	E	E	E
Certificate	-	-	-	h=1.5	h=1.3	h=2

Table 5.6: Step two of (E-Proc) applied to the case of one step of GM with smooth convex functions

Defining the notion of essential inequalities helped us gain an easy numerical procedure to estimate the minimal proofs numerically by Proposition 5.2. In fact the set E returned by (E-Proc) can also be considered as an estimation of the minimal proof.

An observation deduced from the case of one step, will provide us with a conjecture on the initial candidates for essential inequalities. For this conjecture, we will restrict to the case of smooth convex functions ($\mu = 0$)since we only proved in details the bound for one step of GM and smooth convex functions. Let us start with an example that will introduce the intuition behind this conjecture.

Example 5.3. *In the case of one step, consider the interpolation inequalities $I(0,*)$ and $I(1,*)$ with $x_* = f(x_*) = 0$.*

$$I(0,*) : f(x_0) - \boxed{g_0^T x_0} + \frac{1}{2L}\|g_0\|_2^2$$
$$I(1,*) : f(x_1) - g_1^T x_1 + \frac{1}{2L}\|g_1\|_2^2 = f(x_1) - \boxed{g_1^T x_0} + g_0^T g_1 + \frac{1}{2L}\|g_1\|_2^2$$

As discussed in example 5.2, we now know that these two inequalities are essential. One possible explanation of the fact that the two inequalities $I(0,*)$ and $I(1,*)$ are essential for one step can be obtained by going back to the details in the proof. The next step after combining the inequalities, and because of relations (5.2) *and* (5.4) *we combine the inequalities multiplied by the corresponding dual variable. The next step in the proof is adding and subtracting the term $\frac{\|x_0\|^2 C(h_1)}{2}$ where $C(h_1)$ can be either $(1+2h_1)^{-1}$ or $(1-h_1)^2$. Then we reach the following:*

$$f(x_1) - \frac{\|x_0\|^2 C(h_1)}{2} \leq \underbrace{\frac{\|x_0\|^2 C(h_1)}{2} + (\lambda_{0,*}g_0 + \lambda_{1,*}g_1)^T x_0}_{T_1(x_0,g_0,g_1)} + T_2(g_0,g_1)$$

Now at this stage we will simplify $\frac{\|x_0\|^2 C(h_1)}{2}$ by completing the squares. So here the intuition is that if one of the two interpolation inequalities is removed, the proof won't work. We do not know why and how or if the true reason is linked to this fact or not but numerically it seems to be confirmed as we have seen previously.

Here is the conjecture based on the observation in example 5.3.

Conjecture 5.3. *Consider N steps of GM and suppose you want to prove the bound $b^{N,0}$ on the objective function accuracy for the class of smooth convex functions $\mathcal{F}_{0,L}(\mathbb{R}^d)$. Now among all the inequalities in Ω (the set of all the intepolation inequalities defined in 2.8 and written as in 5.1), those who contains terms like $\lambda_{i,j} g_i^T x_0$ will be considered essential.*

For the local minimal proofs, although we can't exclude the possibility of them containing non essential inequalities, as we can see in example 5.1 where $I(0,1)$ is not essential, we may find some local minimal proofs by applying a sort of branch and bound on the set of essential inequalities. Here is a step by step description of the procedure:

- Step 0 start with $\mathcal{S}_l = E_0$ where $E_0 = \{I(i,j) \mid I(i,j) \text{ contains terms in} \lambda_{i,j} g_i^T x_0\}$. Define also a set of testing points, As we did in (E-Proc), T that will be used to check the following:

$$\omega_{\mu,L}^{sdp}(R,H,N,b,C,\mathcal{S}_l) = w_{\mu,L}^{sdp}(R,H,N,b,C) \tag{5.7}$$

- Step 1 Check if condition (5.7) is verified for some $H \in T$ and with $\mathcal{S}_l = E_0$. If it is the case stop and proceed to step three.

- Step 2 While $\mathcal{S}_l \neq \mathcal{S}*$, where $\mathcal{S}^*$ stands for the minimal proof, choose $I(i,j) \in \mathcal{E} \setminus \mathcal{E}_l$ and update $\mathcal{S}_l = \mathcal{S}_l \cup \{I(i,j)\}$
If there exists at least one $H \in T$ such that you verify (5.7), then by definition $\mathcal{S}_l$ can be considered as a local minimal, so stop and proceed to step 3. Else, continue.

- Step 3 Return $\mathcal{S}_l$. If $\mathcal{S}_l = \mathcal{S}^*$, then your testing set T is not good enough. Else, you have found a local minimal proof that worked for at least one H and return this H.

Once again we insist on the importance of the testing points, in fact by definition, finding at least one H is sufficient to have a local minimal proof, but it is more interesting to find more then one. The advantage of searching for local minimal proofs among the essential inequalities, is that you may find some proofs that can be combined together in the minimal proof valid for all H. Here the example of one step is not rich enough to illustrate. We will show an example, in the next section where we will apply the concepts and the procedures we defined in this chapter to the case of two steps of GM.

5.2.2 Flow interpretation.

In this section, we will show one way of visualizing a proof in general, not necessarily a minimal proof. We will represent a proof as a directed graph such that the nodes are the iterates and the arrays are the interpolation inequalities. The idea is that a proof is equivalent to choosing arrays and choosing the values of the dual variables associated is like defining a flow in the graph. The intuition that encouraged to use such a representation originate from the proof of one step of GM, like practically all the concepts in this chapter. First we will start by defining formally the graph associated with a proof. We restrict to the gradient method for this definition.

Definition 5.4. *(Graphical representation of proofs) Consider N steps of gradient method with fixed step sizes where the exact worst-case bound on the objective function accuracy for functions in $\mathcal{F}_{\mu,L}(\mathbb{R}^d)$ is given by $b^{N,\mu}$. Suppose also that $\mathcal{S}$ is a proof for $b^{N,\mu}$. We can represent $\mathcal{S}$ in a directed graph where the nodes are the $N+2$ points $\{x_0, \cdots, x_N, x_*\}$ and the the edges represents the interpolation inequalities in $\mathcal{S}$ of type $I(i,j)$ oriented from x_i to x_j. Then to each node we associate a flow equal to $\lambda_{i,j}$.*

We will illustrate this for one step as usual in the next example.

Example 5.4. *In figure 5.2 you can find the graphical representation of the minimal proof for all $h \in [0,2]$ given by:*

$$\mathcal{S} = \mathcal{E} = \{I(0,*), I(1,*), I(1,0)\}$$

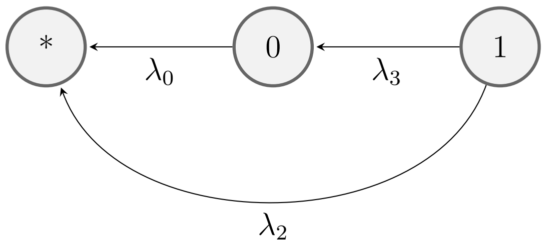


Figure 5.2: Minimal proof for one step of GM.

The flows in the graph are not chosen randomly. If you look at (5.2) and (5.4), you can see that they should verify some specific relations so that we obtain $f(x_N) - f(x_*)$ after we combine the interpolation inequalities. This statement is made more explicit in the next proposition:

Proposition 5.4. *Consider the graph G representing a proof $\mathcal{S}$ of an exact worst-case bound for N steps of GM. for each edge we denote by $\lambda_{i,j}$ the dual variable associated to the inequality $I(i,j) \in \mathcal{S}$ for all $i, j \in I = \{0, \cdots, N, *\}$ and $i \neq j$. We can show that, in order to combine the inequalities in $\mathcal{S}$ and prove the bound, the following relation between the $\lambda_{i,j}$ must hold:*

$$\sum_{j \in I \setminus \{i\}} (\lambda_{i,j} - \lambda_{j,i}) = 0 \quad \forall i \neq \{*, N\} \tag{5.8}$$

$$\sum_{j \in I \setminus \{N\}} (\lambda_{N,j} - \lambda_{j,N}) = 1 \tag{5.9}$$

And for the the node corresponding to x_* simply denoted by $*$, we can deduce from the two previous relations that:

$$\sum_{i \in I \setminus \{*\}} (\lambda_{*,j} - \lambda_{j,*}) = -1$$

Proof. We consider $\mathcal{S} = \Omega$ in this proof for simplicity, but the same reasoning holds for $\mathcal{S} \subset \Omega$ where you simply replace the $\lambda_{i,j}$ of an inequality that is not present in $\mathcal{S}$ by zero. Suppose you want to prove the exact worst-case bound you conjectured for N steps of GM for functions in $\mathcal{F}_{\mu,L}(\mathbb{R}^d)$ denoted by $b^{N,\mu}$. So basically you should prove that :

$$f(x_N) - f(x_*) \leq b^{N,\mu} \tag{5.10}$$

Now assume you take the inequalities $I(i,j)$ in S written as in 5.1, and sum them up after multiplying each of these inequalities by the corresponding dual variable $\lambda_{i,j}$. Then let us denote the left hand side of the inequality you get by rhs where we put all the terms that do not include $f(x_i)$ with $i \in I = \{0, \cdots, N, *\}$. Consider now any x_i notice that in $I(i,j)$ you will have $\lambda_{i,j} f(x_i)$ and in $I(j,i)$ you will have $(-)\lambda_{j,i} f(x_i)$ for all j, so that we can write:

$$\left(\sum_{j\in I j\neq N} \lambda_{N,j} - \sum_{j\in I, j\neq N} \lambda_{j,N}\right) f(x_N) + \sum_{i\in I\setminus\{*,N\}} \left(\sum_{j\in I\setminus\{i\}} \lambda_{i,j} - \sum_{j\in I\setminus\{i\}} \lambda_{j,i}\right) f(x_i)$$
$$+ \left(\sum_{j\in I\setminus\{*\}} \lambda_{*,j} - \sum_{j\in I\setminus\{*\}} \lambda_{j,*}\right) f(x_*) \leq rhs$$

Now since we want to have $f(x_N) - f(x_*)$, we directly deduce that the following should hold for us to obtain the desired right hand side in equation 5.10 thus proving the first part of proposition 5.4.

$$\sum_{j\in I j\neq N} \lambda_{N,j} - \sum_{j\in I, j\neq N} \lambda_{j,N} = 1$$
$$\sum_{j\in I\setminus\{i\}} \lambda_{i,j} - \sum_{j\in I\setminus\{i\}} \lambda_{j,i} = 0 \quad \forall\, i \neq \{N, *\}$$

Now we will show that the relation for $f(x_*)$ will hold automatically from the other two relations. Summing up the two previous relations you get:

$$\sum_{i\in I\setminus\{*\}} \sum_{j\in I\setminus\{i\}} (\lambda_{i,j} - \lambda_{j,i}) = 1$$
$$\sum_{i\in I\setminus\{*\}} (\lambda_{i,*} - \lambda_{*,i}) + \underbrace{\sum_{i\in I\setminus\{*\}} \sum_{j\in I\setminus\{i,*\}} (\lambda_{i,j} - \lambda_{j,i})}_{Sum} = 1$$

Let us now detail sum and show that it is equal to zero.

$$\sum_{j\in I\setminus\{0\}} \lambda_{0,j} - \lambda_{j,0} = \lambda_{0,1} - \lambda_{1,0} + \cdots + \lambda_{0,N-1} - \lambda_{N-1,0} + \lambda_{0,N} - \lambda_{N,0}$$
$$+ \sum_{j\in I\setminus\{1\}} \lambda_{1,j} - \lambda_{1,0} = \lambda_{1,0} - \lambda_{0,1} + \cdots + \lambda_{1,N-1} - \lambda_{N-1,1} + \lambda_{1,N} - \lambda_{N,1}$$
$$\vdots \quad = \quad \vdots$$
$$+ \sum_{j\in I\setminus\{N-1\}} \lambda_{N-1,j} - \lambda_{j,N-1} = \lambda_{N-1,0} - \lambda_{0,N-1} + \lambda_{N-1,1} - \lambda_{1,N-1} + \cdots + \lambda_{N-1,N} - \lambda_{N,N-1}$$
$$+ \sum_{j\in I\setminus\{N\}} \lambda_{N,j} - \lambda_{j,N} = \lambda_{N,0} - \lambda_{0,N} + \lambda_{N,1} - \lambda_{1,N} + \cdots + \lambda_{N,N-1} - \lambda_{N-1,N}$$

$$Sum = 0$$

Then we can deduce the following :

$$\sum_{i \in I \setminus \{*\}} (\lambda_{i,*} - \lambda_{*,i}) = -\left(\sum_{i \in I \setminus \{*\}} (\lambda_{*,i} - \lambda_{i,*}) \right) = 1 \tag{5.11}$$

Thus you have the right hand in 5.10 and you can proceed with the proof. □

Proposition 5.4 will turn out to be useful. In fact relations 5.8 and 5.9 together will constitute a linear system hence we can link those variables together and express the pivot variables of this linear system as linear combination of the free variables so that if we guess the free variables, numerically for example, we can deduce the others directly.

In the next section we will see how we can use these concepts and ideas in order to provide the highlights of a proof for conjecture 4.3.

5.3 Two steps of GM

In light of the preceding section, we will highlight the search for a proof for our conjectured worst-case bound for two fixed but arbitrary step sizes of GM, namely conjecture 4.3. In this case we will have twelve inequalities and defining $I = \{0, 1, 2, *\}$ we can define the set of all inequalities Ω.

$$\Omega = \{I(i,j) \mid i, j \in I \text{ and } i \neq j\} \tag{5.12}$$

Each $I(i,j)$ is written as in 5.1 and the dual variable associated with it will be denoted by $\lambda_{i,j}$ following the notations of the previous section.
We start by obtaining a numerical estimation or approximation of the set of essential inequalities by applying procedure (E-Proc). With this set we obtain an approximation of the minimal proof. Then we branch and bound on the set of essential inequalities to find some local minimal proofs, and we show how we can combine them to obtain the minimum proof for all h_1 and h_2 in $[0, 2]$.

Essential Inequalities and minimal proof

We started by applying (E-Proc) in order to find the local minimal proofs. We constituted a testing set containing values of (h_1, h_2) such that we cover all the regimes, namely $(S, S), (B, B), (S, B)$ and (B, S) interior and boundary. Applying this procedure we obtain E an approximation of the set of essential inequalities $\mathcal{E}$. By proposition 5.2 we have thus obtained $\mathcal{S}$, an approximation of the minimal proof $\mathcal{S}^*$.

$$E = \{I(0,*), I(0,1), I(1,*), I(1,0), I(2,*), I(2,0), I(2,1)\} = \mathcal{S} \approx \mathcal{E} = \mathcal{S}^* \tag{5.13}$$

In figure 5.3, you can find the graphical representation of this approximate minimal proof according to the definition 5.4:

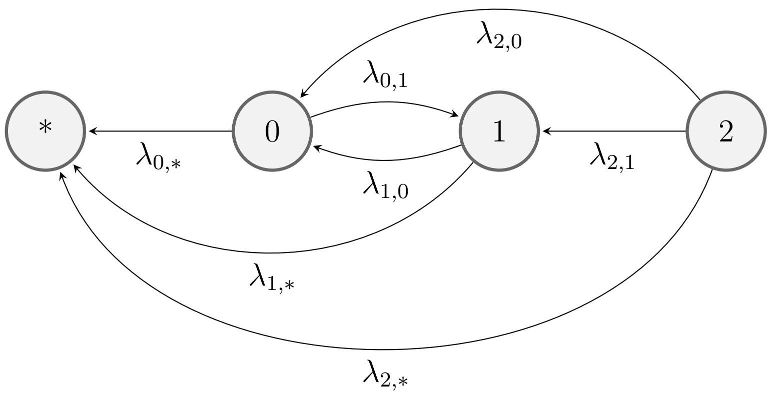


Figure 5.3: Estimation of the minimal proof for two steps of GM between zero and two.

Furthermore by proposition 5.4 we have the following relations:

$$\lambda_{0,*} + \lambda_{0,1} = \lambda_{1,0} + \lambda_{2,0} \tag{5.14}$$
$$\lambda_{1,0} + \lambda_{1,*} = \lambda_{0,1} + \lambda_{2,1} \tag{5.15}$$
$$\lambda_{2,*} + \lambda_{2,0} + \lambda_{2,1} = 1 \tag{5.16}$$

So if we guess the values of four of the dual variables we can guess the remaining. Of course we may try to deduce the values as we did for one step but here the algebra is not easy, and it is worth spending some time to guess the values of the dual variables numerically instead of directly going through the proof.
The fact that the algebra becomes quite hard in the proof in the way we decided to write them (i.e. by combining interpolation inequalities weighted by their dual variables then combing squares) is the first limiting factor that motivates the need to guess the maximum number of dual variables numerically.
Another strategy is to try some minimal proofs and seeing how we can combine them to obtain the minimal proof.

Local minimal proofs

When it comes to local minimal proofs, here we can find some proofs with less then seven inequalities, as we where able to do when we applied the branch and bound procedure discussed in section 5.2.1. Again we used practically the same data set we used for essential inequalities. But we repeated the procedure in order to find a significant number of problems H that can be proved, so more then one.
We were first inspired by one proof that can be found in [DT13] where they proved their conjecture for N constant step sizes h less then one. For two constant steps sizes we have the following local minimal proof $\mathcal{S}_l^{ss}$ valid in $\mathcal{V}^{ss} = \{h_1 = h_2 = h \,|\, 0 \leq h \leq 1\}$:

The proof is local, because it only covers $h \leq 1$. We started by checking that these values can be found numerically. So first from proposition 5.4 we had the

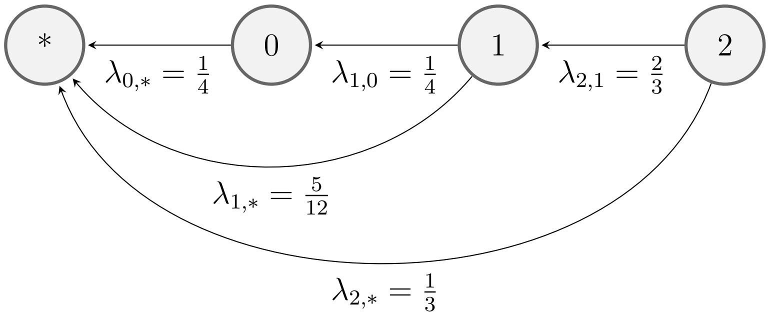


Figure 5.4: Local minimal proof for two constant steps of GM.

following:

$$\lambda_{0,*} = \lambda_{1,0} \tag{5.17}$$
$$\lambda_{1,*} + \lambda_{1,0} = \lambda_{2,1} \tag{5.18}$$
$$\lambda_{2,1} + \lambda_{2,*} = 1 \tag{5.19}$$

We solved for different values of h in $\mathcal{V}^{ss}$ and guessed $\lambda_{0,*}$ and $\lambda_{2,1}$. Then using the relationships we deduced the remaining dual variables and we obtained the same values.

Notice that all the interpolation inequalities are essential, as we can see in (5.13). Then we have tried to see how far can we extend this proof with variable steps. So we started by guessing the values of $\lambda_{0,*}$ and $\lambda_{2,1}$ numerically for arbitrary values of h_1 and h_2 in $[0, 1]$ and we guessed the following:

$$\lambda_{0,*} = \frac{h_1}{2(h_1 + h_2)} \tag{5.20}$$
$$\lambda_{2,1} = \frac{h_2 + h_1}{2h_2 + h_1} \tag{5.21}$$

So with $\mathcal{S}_l^{ss}$ we can prove the following:

Theorem 5.5. *consider a sequence $\{x_1, x_2\}$ generated by two step sizes of the classical gradient method on a smooth convex function f in $\mathcal{F}_{0,L}(\mathbb{R}^d)$ then the exact worst-case bound on the objective function accuracy is given by for all $(h_1, h_2) \in \mathcal{V} = [0, 1]^2$*

$$f(x_2) - f(x_*) \leq \frac{LR^2}{2} \frac{1}{1 + 2h_1 + 2h_2}$$

Now we will provide a proof of this theorem.

Proof. As usual, we will focus on a proof where $f(x_*) = x_* = 0$ and $L = R = 1$. Combining all the inequalities inequalities in $\mathcal{S}_l^{ss}$ multiplied by their respective dual variables we get the following:

$$\begin{aligned}(\lambda_{0,*}-\lambda_{1,0})f(x_0)+(\lambda_{1,0}+\lambda_{1,*}-\lambda_{2,1})f(x_1)+(\lambda_{2,1}+\lambda_{2,*})f(x_2)-(\lambda_{0,*}g_0+\lambda_{1,*}g_1+\lambda_{2,*}g_2)^T x_0 \\ + (\lambda_{1,0}(h_1-1)+\lambda_{1,*}h_1)g_0^T g_1 + (\lambda_{2,1}(h_2-1)+\lambda_{2,*}h_2)g_2^T g_1 + \lambda_{2,*}h_1 g_2^T g_0 \\ + (\frac{\lambda_{0,*}}{2}+\frac{\lambda_{1,0}}{2})\|g_0\|^2 + (\frac{\lambda_{1,0}}{2}+\frac{\lambda_{1,*}}{2}+\frac{\lambda_{2,1}}{2})\|g_1\|^2 + (\frac{\lambda_{2,1}}{2}+\frac{\lambda_{2,*}}{2})\|g_2\|^2 \leq 0\end{aligned}$$

Now thanks to the previous relationships between the dual variables (5.17, 5.18 and 5.19) and adding and subtracting the exact worst-case bound we are trying to prove we get the following:

$$\begin{aligned} f(x_2) - \frac{\|x_0\|^2}{2(1+2h_1+2h_2)} \leq & -\frac{\|x_0\|^2}{2(1+2h1_1+2h_2)} - \lambda_{0,*}\|g_0\|^2 - \lambda_{2,1}\|g_1\|^2 - \frac{1}{2}\|g_2\|^2 \\ & + (\lambda_{10*}g_0 + \lambda_{1,*}g_1 + \lambda_{2,*}g_2)^T x_0 \\ & - (\lambda_{2,1}h_1 - \lambda_{0,*})g_0^T g_1 - (h_2 - \lambda_{2,1})g_2^T g_1 \\ & - ((1-\lambda_{2,1})h_1)g_2^T g_0 \end{aligned} \tag{5.22}$$

Then we have the following:

$$\begin{aligned} (\lambda_{0,*}g_0 + \lambda_{1,*}g_1 + \lambda_{2,*}g_2)^T x_0 = & \frac{\|x_0\|^2}{2(1+2h_1+2h_2)} + \frac{1+2h_1+2h_2}{2}\|\lambda_{0,*}g_0 + \lambda_{1,*}g_1 + \lambda_{2,*}g_2\|^2 \\ & - \underbrace{\frac{1}{1+2h_1+2h_2}\|x_0 - (1+2h_1+2h_2)(\lambda_{0,*}g_0 + \lambda_{1,*}g_1 + \lambda_{2,*}g_2)\|}_{Qf_1} \end{aligned}$$

Plugging this relation and the expressions of $\lambda_{0,*}$ (5.20) and $\lambda_{2,1}$ (5.21) we can simplify 5.22.

$$\begin{aligned} f(x_2) - \frac{\|x_0\|^2}{2(1+2h_1+2h_2)} \leq & q_1(h_1,h_2)\|g_0\|^2 + q_2(h_1,h_2)\|g_1\|^2 + q_3(h_1,h_2)\|g_2\|^2 \\ & + q_4(h_1,h_2)g_0^T g_1 + q_5(h_1,h_2)g_2^T g_1 + q_6(h_1,h_2)g_2^T g_0 - Qf_1 \end{aligned}$$

Then we define the following vector and matrix:

$$\mathbf{g} = \begin{pmatrix} g_0 \\ g_1 \\ g_2 \end{pmatrix} \qquad \mathbf{Q} = \begin{pmatrix} q_1(h_1,h_2) & \frac{q_4(h_1,h_2)}{2} & \frac{q_6(h_1,h_2)}{2} \\ \frac{q_4(h_1,h_2)}{2} & q_2(h_1,h_2) & \frac{q_5(h_1,h_2)}{2} \\ \frac{q_6(h_1,h_2)}{2} & \frac{q_5(h_1,h_2)}{2} & q_3(h_1,h_2) \end{pmatrix}$$

Then we define the quadratic form $Qf_2 = \mathbf{g}^T\mathbf{Q}\mathbf{g}$. In order to to check the regions where this quadratic form is semi-definite negative, we used Wolfram Mathematica [Inc]. We computed the determinant and we obtained zero. Then the only choice we had is to check the eigenvalues of the matrix $\mathbb{Q}$. There are two non-zero eigenvalues, denoted by v_1 and v_2.

The first eigenvalue v_1 is negative for all h_1 and h_2 in $[0,2]$ hence in $\mathcal{V}$. As for v_2 it is not the case everywhere. Let us define the region of small step sizes (Blue region in figure 4.6). Recall that we have defined the function $b_1^{2,0}$ (4.5) as the one mapping to what we called the small step sizes regime and we can formulate this definition as follows:

$$SS = \{h_1, h_2 \in [0,2] \mid max(b_1^{2,0}, b_2^{2,0}, b_3^{2,0}, b_4^{2,0}) = b_1^{2,0}\}$$

Figure 5.5 represents a plot of the region where v_2 is negative so the one where Qf_2 is s.d.n in blue and SS in orange. We can see that for all $(h_1,h_2) \in \mathcal{V}$ the quadratic form Qf_2 is semi definite negative and since Qf_1 is s.d.p we can conclude the proof.

□

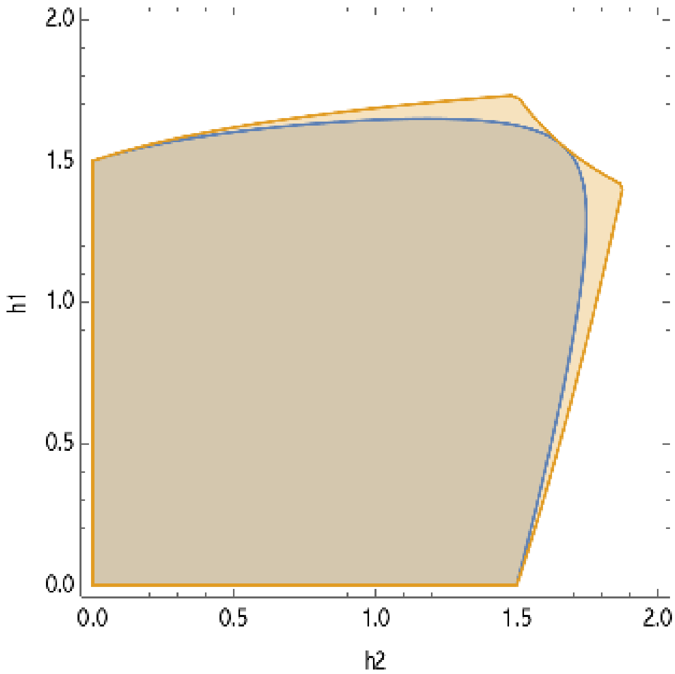


Figure 5.5: Region where Qf_2 s.d.n in BLUE and SS in Orange

In the previous figure, notice as we concluded from numerical experiments by solving $\omega_{0,L}^{sdp}(R, H^{GM}, 2, [0,0,1], 0, \mathcal{S}_l^{ss})$, that we can extend $\mathcal{V}$ to include values of h_1 and h_2 larger then one. But since we have problem on the boundary this local minimal proof cannot be used to prove the bound of conjecture 4.3.
We where also able to deduce a local minimal proof with the same inequalities but different values for the dual variable in the region "BB" (red region in figure 4.6) defined as follows based on the function $b_2^{2,0}$ defined in 4.6.

$$BB = \{h_1, h_2 \in [0,2] \mid max(b_1^{2,0}, b_2^{2,0}, b_3^{2,0}, b_4^{2,0}) = b_2^{2,0}\}$$

With some numerical testing we guessed the following for $\lambda_{0,*}$ and $\lambda_{2,1}$

$$\lambda_{0,*} = (h_1 - 1)(h_2 - 1)^2 \tag{5.23}$$
$$\lambda_{2,1} = (h_2 - 1) \tag{5.24}$$

Notice that their is still a problem on the boundary of the domains so this local minimal proof can't be extended or used to prove the general bound of conjecture 4.3. Repeating the same reasoning as in the proof of the previous theorem, we can get the region where the proof is valid, denoted by $\mathcal{V}^{bb}$, using Wolfram Mathematica [Inc]. This region is represented in blue on the plot of figure 5.6 where we also added the region BB in orange in order to illustrate the fact that this proof is not valid on the whole region BB.

Nevertheless, it is still interesting to notice the simplicity of these local minimal proofs. Now we anted to see if we can find local minimal proofs with six inequalities that will be helpful, in the sense that we can combine them to get the minimal proof. So we applied the branch and bound procedure and it we where able to find two local minimal proofs that seems to combine in order to obtain the minimal proof for

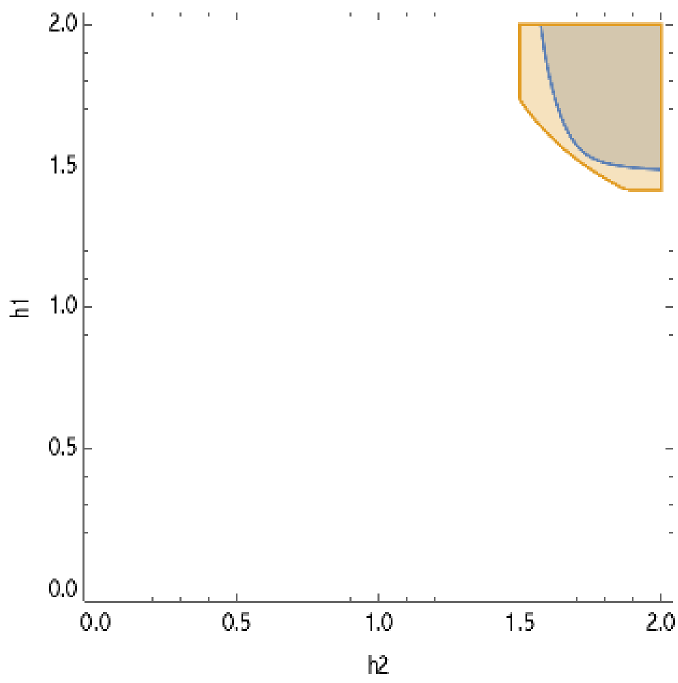


Figure 5.6: Region $\mathcal{V}^{bb}$ in blue and BB in orange

all h_1 and h_2 in $[0,2]$. Let us define those two local minimal proofs:

$$\mathcal{S}_{l,1} = \{I(0,*), I(1,*), I(1,0), I(2,*), I(2,1), I(0,1)\}$$
$$\mathcal{S}_{l,2} = \{I(0,*), I(1,*), I(1,0), I(2,*), I(2,1), I(2,0)\}$$

Notice for example that $\mathcal{S}_{l,1}$ could be used to prove the bound $b_4^{2,0}$ on the region (S, B) defined as follows:

$$SB = \{h_1, h_2 \in [0,2] \mid max(b_1^{2,0}, b_2^{2,0}, b_3^{2,0}, b_4^{2,0}) = b_4^{2,0}\}$$

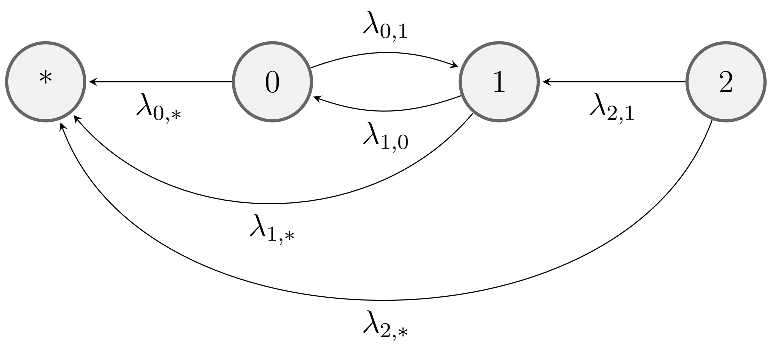


Figure 5.7: Graphical representation of the local minimal proof $\mathcal{S}_{l,1}$.

In figure 5.7 we have provided a graphical representation of $\mathcal{S}_{l,1}$ from which we can deduce the following according to proposition 5.4:

$$\lambda_{0,*} + \lambda_{0,1} = \lambda_{1,0}$$
$$\lambda_{1,0} + \lambda_{1,*} = \lambda_{2,1}$$
$$\lambda_{2,1} + \lambda_{2,*} = 1$$

Then if we can find the value of three of the dual variables we can deduce the remaining. One way of doing it is to try to guess them numerically, to this aim we use the stored values of the dual variables gathered during stage two of GM-Procand try to fit them. Using Matlab Plot-Fitting app we were able to deduce two of the variables:

$$\lambda_{1,0} = \frac{(h_2-1)^2}{(1+h_1)} \tag{5.25}$$

$$\lambda_{2,1} = \frac{(h_2-1)(h_1+h_2)}{h_2(h_1+1)} \tag{5.26}$$

We were not able to deduce a third variable, so if we are going to write the proof, then one of dual variables will be considered free and we will determine its value as we did for one step of GM.
This local minimal proof is also local of course otherwise the minimal proof wont be minimal, and if you test numerically (e.g. for $h_1 = 1.8$ and $h_2 = 1.5$) you will observe this.
Interestingly, when you run the minimal proof with seven inequalities for different values of $(h_1, h_2) \in SB$ you can observe that $\lambda_{2,0} = 0$ so the advantage of this proof is that it can be reused in the global proof.

Now the other local minimal proof of six inequalities $\mathcal{S}_{l,2}$ can be used to prove the bound $b_3^{2,0}$ in the region BS defined as follows:

$$BS = \{h_1, h_2 \in [0,2] \mid max(b_1^{2,0}, b_2^{2,0}, b_3^{2,0}, b_4^{2,0}) = b_3^{2,0}\}$$

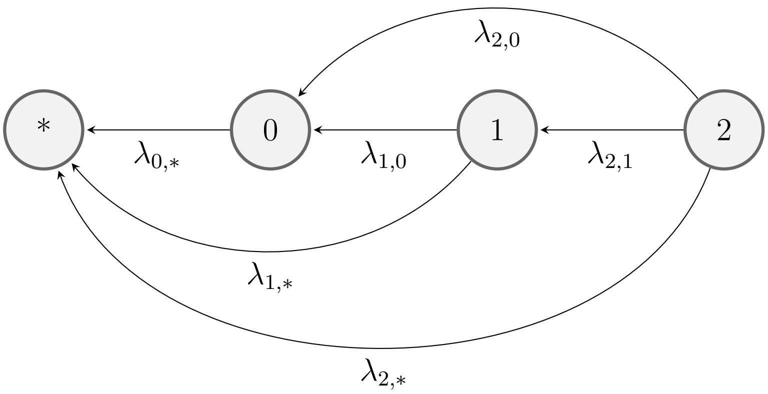


Figure 5.8: Graphical representation of the local minimal proof $\mathcal{S}_{l,2}$.

The graphical representation can be found in figure 5.8 from which applying proposition 5.4 we deduce the following relations:

$$\lambda_{0,*} = \lambda_{2,0} + \lambda_{1,0}$$
$$\lambda_{1,0} + \lambda_{1,*} = \lambda_{2,1}$$
$$\lambda_{2,1} + \lambda_{2,0} + \lambda_{2,*} = 1$$

Here we where only able to deduce $\lambda_{2,1}$ numerically and we obtained:

$$\lambda_{2,1} = \frac{1}{2} \tag{5.27}$$

So it is a bit harder then the other cases and we need to proceed with a proof with two unknown dual variables. Nevertheless when we test $\mathcal{S}_{l,2}$ for different values of h_1 and h_2 in BS, we find that $\lambda_{0,1} = 0$ this time which once again means that we can recycle this proof in the global minimal proof.

One interesting final numerical observation is that it seems like those two minimal proofs will cut the domain $[0, 2]^2$ in two as follows:

$$\begin{array}{cccc} \mathcal{S}_{l,2} & \text{is valid} & \text{for} & h_1 \geq h_2 \\ \mathcal{S}_{l,1} & \text{is valid} & \text{for} & h_1 < h_2 \end{array} \tag{5.28}$$

The main advantage is that using these two local minimal proofs that uses less inequalities, we can provide two proofs that can be combined to give a general proof. We may expect that analytically, we will be able to verify that the first one is correct for $h_1 \geq h_2$ or the left half plain, and the other is valid for $h_1 < h_2$ or the right half plain. Notice that in the plain, h_2 correspond to the X-Axis and h_1 to the Y-Axis. Sadly, we did not have time to investigate more, and of course there are high chances that this won't hold true. In fact the only way is to proceed with the full analytic proof, and check if this numerical observation holds true. As we mentioned earlier this is not an easy task, even though we have six inequalities, specially because we where not able to guess all the dual variables.

Summary

In this chapter, we have started by analyzing the proof of the worst-case bound on the objective function accuracy for one step of GM applied on a smooth convex function. This analysis led us to the conclusion that their an infinite number of possible proofs, and their is no reason that it won't be the case for N steps of GM. So among all these proofs what is the one that uses the least number of inequalities and that is valid for all combination of step sizes H ?
To answer this question, we introduced the concept of minimal proof, the smallest set of inequalities that you can use to prove the exact worst-case bound for all combinations of N fixed but arbitrary steps for any first order method with fixed step sizes and not just for GM. We also defined some local minimal proofs that are valid for at least one value of H, but of course one should try to include as much values of H as possible.
But how can we find such proofs ?
We where not able to find a tool or a concept that can guarantee that the proof is indeed minimal other then writing down a general proof and see how we can reduce it as we did for one step of GM. Nevertheless, by introducing the concept of essential inequalities, we where able to find a numerical procedure (E-Proc) that enabled us to estimate the minimal proof. We were also able to define a procedure, similar to branch and bound, that will enable us to search for local minimal proofs with essential inequalities. We also provided a flow interpretation of the proofs. So when you try to find a proof, you are simply finding a flow in a graph verifying certain conditions. This was formally defined in definition 5.4, and the relation that should be verified by the flow is the one presented in Proposition 5.4. The advantage of these relations, is that they can help you guess the dual variables numerically for example.
We ended this chapter by showing how we can apply these concepts in order to prove Conjecture 4.3. But their remain one limiting factor, the complexity of the algebra in the proof, which was the reason we did not have time to dig more in the analytic proof based on the numerical observations.

Chapter 6

Conclusion

6.1 Research Outcomes

Procedure definition. This research aimed at estimating the performance of the gradient method with fixed variable step sizes for the unconstrained minimization problem of smooth convex and strongly convex functions. To this aim, we started by defining a possible procedure based on the PESTO toolbox [THG17a]. It implies to solve the SDP program (sdp-PEP) for different step sizes chosen between zero and two for a fixed N and $L = R = 1$. Then, based on this data, we tried to guess the exact bounds by fitting the numerical data inspired by the worst-case bounds we already have for smaller number of steps and the Conjectures 4.1 and 4.2 for the case of constant step sizes. Then, we identified these conjectures, we considered a proof by combing the interpolation inequalities weighted by their corresponding dual variable,guessed numerically for example also by solving (sdp-PEP) for different step sizes.

Worst-case bounds identification. We started by testing the third part of our procedure consisting in identifying worst-case bounds. We were first able to deduce the exact worst-case bound on the objective function accuracy for two fixed but arbitrary step sizes of GM.
For the case of smooth convex functions, it turns out that it is the maximum of four functions as displayed in Conjecture 4.3 where each function represents a certain regime corresponding to one possible pairs (h_1, h_2) where h_1 and h_2 are chosen from $\{S, B\}$. Recall, that h_1 will be considered small (S), if we are in the case where $b_1^{2,0}$ or $b_4^{2,0}$ and big (B) otherwise and h_2 will be considered small (S) if we are in the regions where $b_1^{2,0}$ or $b_3^{2,0}$ is maximal. More details in Notation 4.1.
It is important to notice that the function mapping (S, B) (4.9) is not simply deduce by the one mapping (B, S) (4.8) by switching h_1 and h_2 and keeping the same function i.e. that the bound is not symmetric in h_1 and h_2.
As for smooth strongly convex functions, we were able to conjecture the exact worst-case bound on the objective function accuracy for two steps, by exploiting the convergence as μ tends to zero of the general bound for strongly convex functions to the one for smooth convex functions ($\mu = 0$).

One difficulty we encountered is linked to one of the functions for the smooth convex case that we denoted by $b_4^{2,0}$ (4.9) corresponding to the (S, B) regime. It was impossible to deduce it completely from what we had for one step, and it required a new intuition and interpolation. As we discovered in the case of three steps, these possibly new functions that cannot be deduced easily from smaller number of steps seemed like the limiting factor to our procedure, which relies heavily on the worst-case bounds we already have. Thus, we were not able to find the bound for three steps completely. We identified seven functions that seemed to cover seven out of eight regimes and we can clearly isolate the unfitted data corresponding to the eighth one or the (B, B, S) regime for h_1, h_2 and h_3 between zero and two.

By the cases we have studied, we could draw some conclusions on the worst-case bound for N steps. We conjectured that the shape of the worst-case bound would be the maximum of 2^N functions for both smooth convex and strongly convex functions, as you can see in Conjecture 4.36 and 4.41. and we provided two functions G^N and $G^{N,\mu}$ that can be used to generate some of the functions representing the 2^N regimes for any N. They are based the reasoning we followed for two and three steps, so clearly they cannot generate all the cases in the worst-case bound since we already know that the bound is incomplete for three steps.

Optimal step sizes. As for optimal step sizes, or the combination of step sizes that leads to the lowest worst-case, we were able to find this combination for two steps (4.15). It turns out that it corresponds to the intersection of the functions $b_1^{2,0}$, $b_2^{2,0}$ and $b_4^{2,0}$ representing the (S, S), (B, B) and (S, B) regimes respectively, that we were able to compute analytically. For three steps, investigating the location of the minimal worst-case numerically led us to a combination corresponding to $h_2 = 2$. This suggested that we should look for values of h_2 larger then two, something we did not consider until now. So we generated a new data set showing that the numerical minimum worst-case is indeed attained for $h_2 = 2.4 \geq 2$. Investigating the location of this numerical minimum, We deduced that the optimal combination of step sizes is located on the intersection of the (S, S, S), (S, B, B) and (S, B, S), which is no longer one single point as we can see in (4.31). We obtained an estimation of the real optimal step size given by (4.34) by setting $h_1 = 1.4$ in (4.31), which was the value of h_1 corresponding to the numerical minimum of the data set. When we compared the optimal values with variable steps and the one for constant steps, we observed a decrease by 2% and 5% for two and three steps respectively (for three steps we used the approximation). Notice that going for variable steps, is better for two steps and even more for three steps but still the decrease is not that significant. This may suggest that, as N grows larger, variable steps seem better then constant in terms of objective function accuracy, but we do not have a solid numerical argument.

Proofs discussion. Now that we obtained some conjectures on the exact worst-case bounds, we can proceed to the last step of our procedure, the proofs of worst-case bounds. As we mentioned, the main idea of the proof is to combine interpolation inequalities $I(i, j)$ multiplied by their corresponding dual variable (actually all valid proofs should be of that type). The value of the dual variable must depend on the regime we want to prove. For one step of GM, we can find in [Appendix A - Chapter 4 Tay17, p.105] a proof of the exact worst-case bound for the case of smooth strongly convex functions. In fact, it is only one possible proof among many others. We started the last chapter on the proofs, by providing a parametric proof for the exact worst-case bound on objective function accuracy for one step of GM in the case of smooth convex functions. As you can see in table 5.5, many choices of the dual variable λ_0 corresponding to the inequality $I(0, *)$ are possible, and each choice leads to one possible proof of the bound. It turns out that the proof in [Tay17], where we set $\mu = 0$, corresponds to one choice of λ_0 for small step sizes and large step sizes.

From the case of one step, we could predict the difficulty of the algebra in the proofs for more than one step, where you have more inequalities to consider. To tackle with this difficulty, we defined the notions of minimal proofs in Definition 5.1 and its weaker version, local minimal proofs in Definition 5.2. You can see them as a set of inequalities that are necessary and sufficient to prove an exact worst-case bound for all combination of step sizes in the case minimal proofs, and for some combination of step sizes for local minimal proofs. The motivation of these definitions is the fact that in order to simplify the algebra in the proofs, we need to use the minimal possible number of inequalities.
These two definitions are useful, but we need to find a way to find these sets for N steps. The most formal way is to write down a parametric proof with unnecessary inequalities and reduce it to find the minimal as we have done for one step. But, if you have a parametric proof, there is no need for these definitions. So once again we will use the toolbox to obtain an estimation of the minimal proof, because we can add or remove the inequalities in the construction of (sdp-PEP).

We started by introducing the concept of essential inequalities in Definition 5.3. They can be seen as inequalities that should be present in (sdp-PEP) for all H, otherwise the problem will be unbounded or you would obtain a different (larger) worst-case then the one you would have obtained if you had considered all the inequalities. In Proposition 5.2 we showed that the set of essential inequalities corresponds to the minimal proof. The advantage is that it is easy to estimate the set of essential inequalities numerically. So we introduced the procedure (E-Proc) that outputs an estimation of the set of essential inequalities, by testing the definition for each inequality on a testing set of combination of steps. The quality of the estimation is highly dependent on the testing set, so with an appropriate one you can get a good approximation. So we have a way of finding an estimation of the minimal proofs. This is good, but sometimes the minimal number of inequalities could still be large.

Although the local minimal proofs may contain some non essential inequalities, we can use some local minimal proofs that are easy to write and that combine to give the minimal proofs for all H. One way of being sure that it will be the case is to find local minimal proofs as a subset of essential inequalities. The procedure consists in adding the essential inequalities gradually, and finding values of H for which we can prove the worst-case bound using this local minimal proof. Of course the goal is to find the appropriate minimal proofs that could be combined in order to obtain the global one for all H, the minimal proof.

We provided also a graphical representation of the proofs, and the problem of finding the correct values of the dual variables can be compared to the problem of finding a flow in the graph verifying set of $N-1$ relations that we show in Proposition 5.4. These relations insures that when you write the proof, you will get a term in $f(x_N) - f(x_*)$ after combining the inequalities. Furthermore, they provide a linear system that will have infinitely many solutions so that if you guess some of the dual variables numerically for example, you can deduce the others.
We ended this thesis with an investigation of the proof for two steps of GM. We found that although the limitation due to the difficulty of the algebra is reduced with the concepts introduced and thanks to the toolbox, they are still present. In fact the minimal proof estimated still contains seven inequalities. As for the minimal proofs, we were able to find two local minimal proofs that combines to give the minimal proof but we did not have time to investigate more.

6.2 Further research

The exact worst-case for N fixed but arbitrary step sizes, is still an open question. Finding the remaining worst case regime, hence the complete exact worst-case bound for three fixed but arbitrary step sizes can be the starting point of a future research on the subject. The next step would be to check if the worst-case bounds for one, two and three steps are sufficient to conjecture a general bound for N steps. By obtaining such a conjecture, it could be important to check if for N steps of gradient method, the optimal step sizes strategy would consist in fixed but arbitrary step sizes, specially for a large number of iterations.
As for the proofs, the remaining question is to check by completing the proof for two steps for example, if it is worth to develop some algorithms that will automate the procedures of finding estimations of minimal proof (E-Proc) and local minimal proofs. They could then be added as options in the toolbox for possible future use in the performance analysis of possibly other first-order methods. In fact we only tested them on the gradient method, but the definitions seems to hold for any first order method with fixed step sizes strategy.

Bibliography

[BB08] Léon Bottou and Olivier Bousquet. The tradeoffs of large scale learning. In J.C. Platt, D. Koller, Y. Singer, and S. Roweis, editors, *Advances in Neural Information Processing Systems 20 (NIPS 2007)*, pages 161–168. NIPS Foundation (http://books.nips.cc), 2008.

[dC47] Augustin de Cauchy. Méthode générale pour la résolution des systèmes d'équations simultanées. In *Comptes Rendus de l'Académie des Sciences de Paris*, volume 25, pages p. 536–538, 1847.

[DT13] Yoel Drori and Marc Teboulle. Performance of first-order methods for smooth convex minimization: a novel approach. *Springer-Verlag Berlin Heidelberg and Mathematical Optimization Society 2013*, 2013.

[Inc] Wolfram Research, Inc. Mathematica, Version 12.0. Champaign, IL, 2019.

[LÖ4] J. Löfberg. YALMIP : A toolbox for modeling and optimization in MATLAB. In *Proceedings of the CACSD Conference*, 2004.

[Mos10] APS Mosek. The MOSEK optimization software. *Online at http://www.mosek.com*, 54, 2010.

[Nes98] *Inroductory lectures on convex optimastion : A basic course*, volume I. •, 1998.

[Stu99] J. F. Sturm. Using SeDuMi 1.02, a MATLAB toolbox for optimization over symmetric cones. *Optimization Methods and Software*, 11–12:625–653, 1999.

[Tay17] Adrien Taylor. *Convex interpolation and performance estimation of first-order methods for convex optimization.* PhD thesis, Univesrsité Catholique de Louvain, 2017.

[THG17a] Adrien Taylor, Julien Hendrickx, and François Glineur. Performance Estimation Toolbox (PESTO): automated worst-case analysis of first-order optimization methods. In *Proceedings of the 56th IEEE Conference on Decision and Control (CDC 2017)*, 2017.

[THG17b] Adrien B Taylor, Julien M Hendrickx, and François Glineur. Smooth strongly convex interpolation and exact worst-case performance of first-order methods. *Mathematical Programming*, 161(1-2):307–345, 2017.

UNIVERSITÉ CATHOLIQUE DE LOUVAIN
École polytechnique de Louvain
Rue Archimède, 1 bte L6.11.01, 1348 Louvain-la-Neuve, Belgique | **www.uclouvain.be/epl**