CrossMark

Smooth strongly convex interpolation and exact worst-case performance of first-order methods

Adrien B. Taylor[1] · **Julien M. Hendrickx**[1] ·
François Glineur[1]

Abstract We show that the exact worst-case performance of fixed-step first-order methods for unconstrained optimization of smooth (possibly strongly) convex functions can be obtained by solving convex programs. Finding the worst-case performance of a black-box first-order method is formulated as an optimization problem over a set of smooth (strongly) convex functions and initial conditions. We develop closed-form necessary and sufficient conditions for smooth (strongly) convex interpolation, which provide a finite representation for those functions. This allows us to reformulate the worst-case performance estimation problem as an equivalent finite dimension-independent semidefinite optimization problem, whose exact solution can be recovered up to numerical precision. Optimal solutions to this performance estimation problem provide both worst-case performance bounds and explicit functions matching them, as our smooth (strongly) convex interpolation procedure is constructive. Our works build on those of Drori and Teboulle (Math Program 145(1–2):451–482, 2014) who introduced and solved relaxations of the performance estimation problem for smooth

✉ Adrien B. Taylor
   Adrien.Taylor@uclouvain.be

   Julien M. Hendrickx
   Julien.Hendrickx@uclouvain.be

   François Glineur
   Francois.Glineur@uclouvain.be

[1] ICTEAM Institute/CORE, Université catholique de Louvain, B-1348 Louvain-la-Neuve, Belgium

convex functions. We apply our approach to different fixed-step first-order methods with several performance criteria, including objective function accuracy and gradient norm. We conjecture several numerically supported worst-case bounds on the performance of the fixed-step gradient, fast gradient and optimized gradient methods, both in the smooth convex and the smooth strongly convex cases, and deduce tight estimates of the optimal step size for the gradient method.

Keywords Smooth convex minimization · Smooth convex interpolation · First-order methods · Worst-case analysis · Rates of convergence · Semidefinite programming

Mathematics Subject Classification 90C25 · 90C30 · 90C60 · 68Q25 · 90C22

1 Introduction to performance estimation

Consider the standard unconstrained minimization problem

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

where f is a smooth convex function, possibly strongly convex. First-order black-box methods, which only rely on the computation of f and its gradient at a sequence of iterates, can be designed to solve this type of problem iteratively. A central question is then to estimate the accuracy of solutions computed by such a method. More precisely, given a class of problems and a first-order method, one wishes to establish the worst-case accuracy of solutions that can obtained after applying a given number of iterations, i.e., the performance of the method on the given class of problems.

Many first-order algorithms have been proposed in the literature for smooth convex or smooth strongly convex functions, for which one usually provides a theoretical *upper bound* on the worst-case accuracy after a number of iterations (see e.g., [21] or [6, Chap. 6] for recent overviews). However, many analyses focus on the asymptotic rate of convergence of these bounds, rather than trying to compute exact numerical values. Similarly, *lower bounds* on the performance of first-order black-box methods on given classes of problems can be found in the literature (see e.g., the seminal [19]), again often with a focus on asymptotic rates of convergence. In many situations, the asymptotic rate of convergence of the best available methods match those lower bounds.

Nevertheless, the *exact* numerical value of the worst-case performance of a given method is usually unknown. This is because upper bounds are not assessed precisely, i.e., are known only up to a (possibly unspecified) constant. Another reason is that lower bounds for specific methods are not very frequently developed, and that general lower bounds (valid for all methods) can be quite weak for specific methods, especially if those methods do not feature the best possible asymptotic rate of convergence. Finally, even if exact numerical values are known for both lower and upper bounds, and share the same (optimal) asymptotic rate of convergence, a significant gap between the numerical values of those lower and upper bounds can subsist. If one cares about the worst-case efficiency of a first-order method in practice, this gap can translate into a very large uncertainty on the concrete behavior of a method.

This work is not concerned with asymptotic rates of convergence. It will focus on the computation of the exact worst-case performance of a given first-order black-box method, on a given class of functions, after a given number of iterations. We prove that this question can be formulated and solved exactly as a (finite-dimensional) convex optimization problem, with the following attractive features:

– Our formulation is a semidefinite optimization problem whose dimension is proportional to the square of the number of iterations of the method to be analyzed.
– Any dual feasible solution of our formulation provides an upper bound on the worst-case performance. This solution can be easily converted into a standard proof establishing a bound on the performance (i.e., a series of valid inequalities).
– Any primal feasible solution of our formulation provides a lower bound on the worst case performance. This solution can be easily converted into a concrete function on which the method exhibits the corresponding performance.
– Hence our formulation is exact, i.e., its optimal value provides the exact worst-case performance.

Our formulation covers both smooth convex functions and smooth strongly convex functions in a unified fashion. It covers a very large class of first-order methods which includes the majority of standard methods for smooth unconstrained convex optimization. It can be applied to a variety of performance measures, such as objective function accuracy, gradient norm, or distance to an optimal solution.

1.1 Formal definition

Our goal is to express the worst-case performance of an optimization algorithm as the solution of an optimization problem. This approach was pioneered by Drori and Teboulle [10], who called it a performance estimation problem (PEP). We now provide a formal definition for this problem.

We consider unconstrained minimization problems involving a given class of objective functions, and only treat first-order black-box methods. This means that the method can only gather information about the objective function using an oracle $\mathcal{O}_f$, which returns first-order information about specific points, i.e., $\mathcal{O}_f(x) = \{f(x), \nabla f(x)\}$. Formally, the first N iterates generated by a first-order black-box method $\mathcal{M}$ (which correspond to N calls of the oracle), starting from an initial point x_0, can be described with

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

In order to measure the performance of a given method $\mathcal{M}$ on a specific function f with a specific starting point, we introduce a performance criterion $\mathcal{P}$ to be minimized, that will only depend on the function f and the sequence of the iterates $\{x_0, x_1, \ldots, x_N\}$

generated by the method. Since we are in a black-box setting, we require that the criterion can be computed from the output of the oracle $\mathcal{O}_f$, which has only access to the iterates as well as to an additional point x_*, defined to be any minimizer of function f (the latter being necessary if the criterion has to compare iterates to an optimal solution).

Examples of this performance criterion $\mathcal{P}(\mathcal{O}_f, x_0, \cdots, x_N, x_*)$ include the objective function accuracy $f(x_N) - f(x_*)$, the norm of the gradient $\|\nabla f(x_N)\|$, or the distance to an optimal solution $\|x_N - x_*\|$ (see also Sect. 4.3 for an example of criterion that does not only depend on the last iterate).

Finally, we consider a given class $\mathcal{F}$ of smooth convex or smooth strongly convex functions, over which we wish to estimate the worst-case performance of a method after N iterations. Note that the dimension of functions belonging to class $\mathcal{F}$ is left unspecified; in particular we will allow class $\mathcal{F}$ to contain functions with varying dimensions, in order to obtain dimension-free results.

As methods try to minimize the performance criterion, their worst-case performance is obtained by maximizing $\mathcal{P}$ over functions in $\mathcal{F}$, which can be written as

$$w(\mathcal{F}, R, \mathcal{M}, N, \mathcal{P}) = \sup_{f, x_0, \ldots, x_N, x_*} \mathcal{P}(\mathcal{O}_f, x_0, \ldots, x_N, x_*) \qquad \text{(PEP)}$$

$$\text{such that } f \in \mathcal{F}$$
$$x^* \text{ is optimal for } f,$$
$$x_1, \ldots, x_N \text{ is generated from } x_0 \text{ by method } \mathcal{M} \text{ with } \mathcal{O}_f,$$
$$\|x_0 - x_*\|_2 \leq R.$$

Parameter R was introduced to bound the distance between the initial point x_0 and the optimal solution x_*. Indeed, it is well-known that in most situations, performance of a first-order method cannot be sensibly assessed without such a constraint (see also the discussion of Sect. 3.5).

1.2 Finite-dimensional reformulation using interpolation

Because it involves an unknown function f as a variable, problem (PEP) is infinite-dimensional. Nevertheless, using the black-box property of the method (and of the performance criterion), we will show that a completely equivalent finite-dimensional problem can readily be formulated by restricting the variable f to the knowledge of the output of its oracle $\mathcal{O}_f$ on the iterates $\{x_0, x_1, \ldots, x_N\}$ and x_*. Indeed, denoting the output of the oracle at each iterate x_i by $\mathcal{O}_f(x_i) = \{f_i, g_i\}$, method $\mathcal{M}$ defined by (1) can be equivalently rewritten as

$$\begin{aligned} x_1 &= \mathcal{M}_1(x_0, f_0, g_0), \\ x_2 &= \mathcal{M}_2(x_0, f_0, g_0, f_1, g_1), \\ &\vdots \\ x_N &= \mathcal{M}_N(x_0, f_0, g_0, \ldots, f_{N-1}, g_{N-1}). \end{aligned} \qquad (2)$$

Now, defining a set $I = \{0, 1, 2, \ldots, N, *\}$ for the indices of the iterates, we can reformulate (PEP) into a problem involving only the iterates $\{x_i\}_{i \in I}$, their function values $\{f_i\}_{i \in I}$ and their gradients $\{g_i\}_{i \in I}$ as (using equivalence between optimality of x_* and constraint $g_* = 0$, as our problem is unconstrained)

$$w^f(\mathcal{F}, R, \mathcal{M}, N, \mathcal{P}) = \sup_{\{x_i, g_i, f_i\}_{i \in I}} \mathcal{P}\left(\{x_i, g_i, f_i\}_{i \in I}\right), \qquad \text{(f-PEP)}$$

such that there exists $f \in \mathcal{F}$ such that $\mathcal{O}_f(x_i) = \{f_i, g_i\} \, \forall i \in I$,

$$g_* = 0,$$

$x_1, \ldots, x_N$ is generated from x_0 by method $\mathcal{M}$

with $\{f_i, g_i\}_{i \in \{0, \ldots, N-1\}}$,

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

The crucial part of this reformulation is the first constraint, which can be understood as requiring that the set of variables $\{x_i, g_i, f_i\}_{i \in I}$ can be *interpolated* by a function belonging to the class $\mathcal{F}$. This optimization problem is strictly equivalent to the original (PEP) in terms of optimal value, since every solution to (f-PEP) can be interpolated by a solution of (PEP) and, reciprocally, every solution of (PEP) can be discretized to provide a solution to (f-PEP). From that it is clear that $w(\mathcal{F}, R, \mathcal{M}, N, \mathcal{P}) = w^f(\mathcal{F}, R, \mathcal{M}, N, \mathcal{P})$.

1.3 Paper organization and main contributions

We focus on the class of smooth (strongly) convex functions. Therefore an exact formulation of problem (PEP) as (f-PEP) will require a set of necessary and sufficient conditions for the existence of a smooth strongly convex interpolating function, which is the main result obtained in Sect. 2. This set of conditions, which is of independent interest, was previously only known for general nonsmooth convex functions. Our approach is fully constructive, as we also exhibit a procedure to interpolate a smooth (strongly) convex function from a set of points with their associated gradients and function values, when such an interpolating function exists.

In Sect. 3, we show how the resulting finite-dimensional (f-PEP) problem can be reformulated exactly into a (convex) semidefinite optimization problem, which provides the first tractable and provably exact formulation of the performance estimation problem. We allow consideration of both smooth convex and smooth strongly convex functions, as well as a large class of performance criteria.

Section 4 then tests our approach numerically on several standard first-order methods, including the constant-step gradient method, the fast gradient method and the optimized gradient method from [14]. We are able to confirm several bounds appearing previously in [10], and to conjecture several new worst-case performance bounds, including bounds for strongly convex functions, and bounds on the gradient norm (either for the final iterate, or the smallest norm among all iterates). Another byproduct of our results is a tight estimate of the optimal step size for the gradient method on smooth convex and smooth strongly convex functions.

1.4 Prior work

Drori and Teboulle [10] were first to consider the notion of a performance estimation problem. They focus exclusively on the case of smooth convex functions equipped with the performance criterion $f(x_N) - f_*$, and introduce the idea of reducing (PEP) to a finite-dimensional problem involving only the iterates x_i, their gradients g_i and function values f_i, along with an optimal point x_* and optimal value f_*. They treat several standard first-order algorithms, namely, the standard fixed-step gradient algorithm, the heavy-ball method and the accelerated gradient method [20]. In their approach, (PEP) is expressed as a non-convex quadratic matrix program [2], which is then relaxed and dualized. The resulting convex problem is then used to provide bounds on the worst-case performance (and, in some cases, is solved analytically). As will be shown later in this paper (see Sect. 4), because of the use of a relaxation and the dualization of a non-convex problem, these bounds are in general not tight, although they are in many special cases.

A section in [10] is also devoted to the optimization of the coefficients of a fixed-step first-order black-box method. More precisely, a numerical optimization solver is used to identify a method performing best according to their relaxation of the performance estimation problem. This approach is taken further in [14], which provides an analytical description of this optimized method. Again we stress that, due to the non-tightness of the relaxation in general, these optimized methods are not guaranteed to have the best possible performances.

Another computational approach for the analysis and design of first-order algorithms is proposed in [16], in which optimization procedures are regarded as dynamical systems. Integral quadratic constraints (IQC), which are usually used to obtain stability guarantees on complicated dynamical systems, are adapted in order to obtain sufficient conditions for the convergence of optimization algorithms. This methodology is able to establish iteration-independent linear rates of convergence by solving a single small semidefinite program. However those bounds, valid for any number of iterations, are in general not tight, i.e., more conservative than ours and those of [10] when used to estimate worst-case performance after a given finite number of iterations (see Sect. 4.1.2 for an example). In addition, while this methodology is well-suited for studying the linear convergence rates of algorithms for smooth strongly convex optimization, it fails to recover the exact sublinear rates in the non-strongly convex case.

2 Smooth strongly convex interpolation

This section develops a necessary and sufficient condition for the existence of a smooth strongly convex function interpolating through a given set of data triples $\{x_i, g_i, f_i\}_{i \in I}$, i.e., deciding whether there exists a smooth strongly convex function f such that $f(x_i) = f_i$ and $g_i \in \partial f(x_i)$ for all $i \in I$.

This result generalizes the well-known set of conditions guaranteeing the existence of a convex, possibly nonsmooth interpolating function (see Theorem 1 in Sect. 2.3). It is the main technical ingredient of our exact convex reformulation of performance estimation problems.

2.1 Definitions and problem statement

We start by defining the functional class of interest, using the standard point of view from convex analysis—we refer to classic books [1,12,23,24] for details. Given two parameters μ and L satisfying $0 \leq \mu < L \leq +\infty$, we consider proper closed convex functions (i.e., whose epigraphs are non-empty closed convex sets) satisfying both a smoothness condition (depending on the parameter L, which is the Lispchitz constant of the gradient) and a strong convexity condition (depending on the parameter μ). We explicitly allow the case $L = +\infty$ to include nonsmooth functions, while μ on the other hand is always assumed to be finite. In the rest of this paper, we use the conventions $1/+\infty = 0$ and $+\infty - \mu = +\infty$ to deal with the case $L = +\infty$.

Definition 1 *(L-smooth μ-strongly convex functions)* Consider a proper and closed convex function $f : \mathbb{R}^d \to \mathbb{R} \cup \{+\infty\}$, and constants $\mu \in \mathbb{R}^+$, $L \in \mathbb{R}^+ \cup \{+\infty\}$ with $\mu < L$. We say that function f is L-smooth μ-strongly convex (which we denote by $f \in \mathcal{F}_{\mu,L}(\mathbb{R}^d)$) if and only if the following two conditions are satisfied:

(a) inequality $\frac{1}{L}\|g_1 - g_2\|_2 \leq \|x_1 - x_2\|_2$ holds for all pairs $x_1, x_2 \in \mathbb{R}^d$ and corresponding subgradients $g_1, g_2 \in \mathbb{R}^d$ (i.e., such that $g_1 \in \partial f(x_1)$ and $g_2 \in \partial f(x_2)$);
(b) function $f(x) - \frac{\mu}{2}\|x\|_2^2$ is convex.

This definition is not entirely standard, as it involves subgradients and allows the constant L to be equal to $+\infty$. In the case of a finite L, condition (a) immediately implies uniqueness of the subgradient at each point, hence differentiability of the function, and we recover the well-known Lipschitz condition on the gradient of a smooth function. On the other hand, when $L = \infty$, condition (a) becomes vacuous, and the function can be non-differentiable. Condition (b) can easily be seen to be algebraically equivalent to the standard definition of strong convexity (and the case $\mu = 0$ corresponds to a convex but not strongly convex function). The class of proper closed convex functions simply corresponds to $\mathcal{F}_{0,\infty}$. The case $L = \mu$ can be safely discarded, as it only involves simple quadratic functions whose minimization is trivial. The reason for this slightly non-standard definition of $\mathcal{F}_{\mu,L}$ and the possibility of choosing $L = +\infty$ will become clear later, when dealing with Legendre–Fenchel conjugation.

As explained above and in the introduction, our approach to express the original infinite-dimensional (PEP) in a finite-dimensional fashion relies on an interpolating condition for smooth strongly convex functions. This directly motivates the following definition.

Definition 2 *($\mathcal{F}_{\mu,L}$-interpolation)* Let I be an index set, and consider the set of triples $S = \{(x_i, g_i, f_i)\}_{i \in I}$ where $x_i, g_i \in \mathbb{R}^d$ and $f_i \in \mathbb{R}$ for all $i \in I$. Set S is $\mathcal{F}_{\mu,L}$-*interpolable* if and only if there exists a function $f \in \mathcal{F}_{\mu,L}(\mathbb{R}^d)$ such that we have both $g_i \in \partial f(x_i)$ and $f(x_i) = f_i$ for all $i \in I$.

2.2 Necessity and sufficiency of conditions for smooth convex interpolation

Our goal is to identify a set of necessary of sufficient conditions involving the set of data triples and characterizing the existence of an interpolating function. Finding necessary
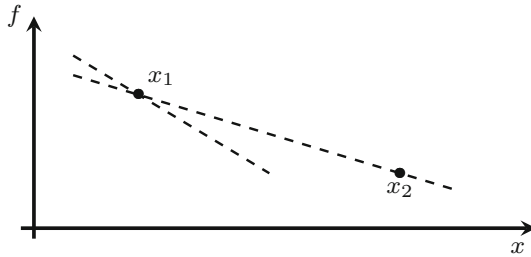
Fig. 1 Example $(x_1, g_1, f_1) = (-1, -2, 1)$ and $(x_2, g_2, f_2) = (0, -1, 0)$ for $I = \{1, 2\}$ and $d = 1$. This set satifies conditions (C1f) but cannot be interpolated by a smooth convex function: the convexity requirement forces the interpolating convex function to lie entirely above its linear under-approximations, which lead to an unavoidable non-differentiability at x_1

conditions is relatively easy: starting from any set of necessary conditions that holds on the whole domain of a smooth strongly convex function, one can simply restrict this set to those conditions involving only points x_i with $i \in I$ (i.e., to discretize it). For example, it is well-known that the class of L-smooth convex functions $\mathcal{F}_{0,L}(\mathbb{R}^d)$ is characterized by the pair of inequalities

$$f(y) \geq f(z) + \nabla f(z)^\top (y - z), \qquad \forall\, y, z \in \mathbb{R}^d, \tag{C1}$$

$$||\nabla f(y) - \nabla f(z)||_2 \leq L||y - z||_2, \qquad \forall\, y, z \in \mathbb{R}^d.$$

Therefore, specializing those conditions for $y = x_i$ and $z = x_j$ with $i, j \in I$ leads to the following set of inequalities, which is *necessary* for the existence of an interpolating function in $\mathcal{F}_{0,L}$:

$$f_i \geq f_j + g_j^\top (x_i - x_j), \qquad \forall i, j \in I, \tag{C1f}$$

$$||g_i - g_j||_2 \leq L||x_i - x_j||_2, \qquad \forall i, j \in I.$$

Now, perhaps surprisingly, it turns out that this latter set of conditions is *not sufficient* to guarantee $\mathcal{F}_{0,L}$-interpolability, despite the fact that the originating conditions (C1) *are sufficient* to guarantee that $f \in \mathcal{F}_{0,L}(\mathbb{R}^d)$. In order to see that, consider the following example with $I = \{1, 2\}$ and $d = 1$:

$$(x_1, g_1, f_1) = (-1, -2, 1) \quad \text{and} \quad (x_2, g_2, f_2) = (0, -1, 0).$$

This pair can clearly not be interpolated by a function in $\mathcal{F}_{0,L}(\mathbb{R}^1)$ for any L, as there is an unavoidable non-differentiability at x_1 as illustrated on Fig. 1. However, it satisfies Conditions (C1f) with $L = 1$, which is therefore not sufficient to guarantee smooth convex interpolation.

Similarly, we can carry out the same exercise for the following conditions, also well-known to be equivalent to inclusion on $\mathcal{F}_{0,L}(\mathbb{R}^d)$ when imposed on the whole space:

$$f_i \geq f_j + g_j^\top (x_i - x_j), \qquad \forall\, i, j \in I, \tag{C2f}$$

$$f_i \leq f_j + g_j^\top (x_i - x_j) + \frac{L}{2}||x_i - x_j||_2^2, \qquad \forall\, i, j \in I.$$

With an appropriate use of an additional dimension ($d = 2$), one can readily observe that some information may be hidden to this pair of inequalities. Consider the example

$$(x_1, g_1, f_1) = \left(\begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 1 \\ 0 \end{pmatrix}, 0 \right) \quad \text{and} \quad (x_2, g_2, f_2) = \left(\begin{pmatrix} 1 \\ 0 \end{pmatrix}, \begin{pmatrix} 1 \\ 1 \end{pmatrix}, 1 \right),$$

from which no smooth convex interpolation can be made (again, unavoidable non-differentiability at both x_1 and x_2). However, both Conditions (C1f) and (C2f) are satisfied with $L = 1$.

Those examples illustrate the weakness of a naive approach that consists in discretizing standard necessary and sufficient conditions defined on the whole space. If those discretized conditions were used in a performance estimation problem over a given class of functions $\mathcal{F}_{\mu,L}$, they would implicitly allow the performance of functions that do not belong to the class $\mathcal{F}_{\mu,L}$ to be taken into account. This would correspond to a relaxation of the original performance estimation problem, and would only lead to upper bounds on the worst-case performance. To conclude this section, note that any set of necessary and sufficient conditions for smooth convex interpolability must be the discretization of some necessary and sufficient conditions on the whole domain, whereas the previous examples precisely show that the converse is not true.

In the next subsections, we follow a more principled approach in order to tackle the $\mathcal{F}_{\mu,L}$-interpolation problem. We start with a special case of convex interpolation, that of proper convex functions without smoothness or strong convexity requirement (i.e., the class $\mathcal{F}_{0,\infty}$), for which a solution is well-known.

2.3 Convex interpolation

In order to build interpolation conditions for the class of smooth strongly convex functions, we begin by constructing interpolation conditions for the simpler class of convex functions $\mathcal{F}_{0,\infty}(\mathbb{R}^d)$. As this result will be one of building blocks of the smooth strongly convex interpolation procedure, a simple constructive proof of this theorem is provided.

Theorem 1 *(Convex interpolation) The set* $\{(x_i, g_i, f_i)\}_{i \in I}$ *is* $\mathcal{F}_{0,\infty}$-*interpolable if and only if*

$$f_i \geq f_j + g_j^\top (x_i - x_j) \quad \forall i, j \in I. \tag{3}$$

Proof (Necessity.) Assume there exists a convex function $f : \mathbb{R}^d \to \mathbb{R}$ such that $f_i = f(x_i)$ and $g_i \in \partial f(x_i) \ \forall i \in I$. The definition of a subgradient then immediately implies that

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

(Sufficiency). Define the following piecewise-linear convex function

$$f(x) = \max_{j \in I} \left\{ f_j + g_j^\top (x - x_j) \right\}.$$

Since f is the pointwise maximum of a finite number of affine functions, its epigraph is a non-empty polyhedron, and hence f is convex, closed and proper. In addition, $f(x_i) = f_i$ holds by construction. Indeed, we first see that

$$
\begin{aligned}
f_i &= f_i + g_i^\top (x_i - x_i), \\
&\leq \max_{j \in I} \left\{ f_j + g_j^\top (x_i - x_j) \right\} = f(x_i).
\end{aligned}
$$

Therefore, we have $f_i \leq f(x_i)$. In addition to this, we have

$$
\begin{aligned}
f(x_i) &= \max_{j \in I} \left\{ f_j + g_j^\top (x_i - x_j) \right\}, \\
&\leq f_i \quad \text{using Condition (3) for each } j,
\end{aligned}
$$

which allows to conclude that $f(x_i) = f_i$. The construction also implies that $g_i \in \partial f(x_i)$, i.e.,

$$
\begin{aligned}
f(x) &= \max_{j \in I} \left\{ f_j + g_j^\top (x - x_j) \right\} \quad \forall x \in \mathbb{R}^d, \\
&\geq f_i + g_i^\top (x - x_i) \quad \forall i \in I, x \in \mathbb{R}^d, \\
&\geq f(x_i) + g_i^\top (x - x_i) \quad \forall i \in I, x \in \mathbb{R}^d.
\end{aligned}
$$

$\square$

One should note that the effective domain of f is dom $f = \mathbb{R}^d$—that is, the function takes finite values for all $x \in \mathbb{R}^d$. This is of course not the only way of reconstructing a valid f. For example, we could choose

$$
f(x) = \begin{cases} \max_j \left\{ f_j + g_j^\top (x - x_j) \right\} & \text{if } x \in \text{conv} \left(\{x_i\}_{i \in I} \right), \\ +\infty & \text{otherwise.} \end{cases}
$$

Remark 1 Our interpolation problem is an extension of the classical finite convex integration problem, which is concerned with the recovery of a convex function from a set of points x_i associated with a subgradient g_i (i.e., function values are not specified). Finite convex integration is treated in details in [15] (only in the convex case $\mu = 0$ and $L = +\infty$). It is the finite version of the continuous convex integrability problem, which is treated in [23].

A direct necessary and sufficient condition for deciding whether a set is convex integrable is to require the existence of function values f_i for which the set $\{(x_i, g_i, f_i)\}_{i \in I}$ is convex interpolable. It is however also possible to derive a set of inequalities that does not involve unknown function values f_i, using the so-called cyclic monotonicity conditions. More precisely, consider for every sequence $\{i_1, i_2, \ldots, i_N\}$ of distinct indices in I, and the corresponding cyclic sequence of successive pairs $\{(i_1, i_2), (i_2, i_3), \ldots, (i_{N-1}, i_N), (i_N, i_1)\}$. Summing inequality (3) over each pair of indices in this cyclic sequence produces a necessary inequality that does

not involve function values f_i. Moreover, the set of all such inequalities, obtained from all possible sequences of distinct indices, is necessary and sufficient for convex integration, see e.g. [15,23]. Note that this condition features a much larger number of inequalities.

2.4 Conjugation and minimal curvature subtraction

In this section, we review some concepts and results needed for our generalization of convex interpolation to smooth convex interpolation. We begin with the concept of *conjugation* operation. This operation is a key element in our approach, since it provides a way to reduce the general smooth strongly convex interpolation problem to a simpler convex interpolation problem.

Definition 3 Given a proper function $f : \mathbb{R}^d \to \mathbb{R} \cup \{+\infty\}$, the (Legendre–Fenchel) conjugate $f^* : \mathbb{R}^d \to \mathbb{R} \cup \{+\infty\}$ of f is defined as:

$$f^*(y) = \sup_{x \in \mathbb{R}^d} y^\top x - f(x).$$

Conjugate functions enjoy numerous useful properties. Among other, they are always closed and convex. In fact, conjugation realizes a one-to-one correspondence in the set of proper closed convex functions (i.e., an involution), see for example Theorem 12.2 of [23]. That is, for any $f \in \mathcal{F}_{0,\infty}(\mathbb{R}^d)$ we have $f^* \in \mathcal{F}_{0,\infty}(\mathbb{R}^d)$ and $f^{**} = f$.

Among the useful properties of conjugate functions, we note that for any function $f \in \mathcal{F}_{0,\infty}(\mathbb{R}^d)$, conjugation can be seen as an operation reversing the roles of the coordinates and the subgradients: any subgradient (resp. coordinate) in one space becomes a coordinate (resp. subgradient) in the second space. In other words, for any function $f \in \mathcal{F}_{0,\infty}(\mathbb{R}^d)$ and its conjugate f^*, it is equivalent to require that x and g satisfy condition $g \in \partial f(x)$, condition $x \in \partial f^*(g)$ or condition $f(x) + f^*(g) = g^\top x$. This can be obtained using first-order optimality conditions on the definition of conjugate function; we refer to Theorem 23.5 from [23] for more details. The next theorem emphasizes the effect of this link for the class of smooth convex functions.

Theorem 2 *Consider a function $f \in \mathcal{F}_{0,\infty}(\mathbb{R}^d)$. We have $f \in \mathcal{F}_{0,L}(\mathbb{R}^d)$ if and only if $f^* \in \mathcal{F}_{1/L,\infty}(\mathbb{R}^d)$.*

Theorem 2 is basically Proposition 12.60 of [24] in the case $L < +\infty$ and reduces to Theorem 12.2 of [23] in the case $L = +\infty$. This also explains why we need to include the case $L = +\infty$ in our interpolation problem: this is so that we can include the conjugates of smooth but non-strongly convex functions in $\mathcal{F}_{0,L}(\mathbb{R}^d)$.

The next lemma gives a simple way of expressing smooth strongly convex functions in terms of smooth functions.

Theorem 3 *Consider a function $f \in \mathcal{F}_{0,\infty}(\mathbb{R}^d)$. We have $f \in \mathcal{F}_{\mu,L}(\mathbb{R}^d)$ if and only if $f(x) - \frac{\mu}{2}\|x\|_2^2 \in \mathcal{F}_{0,L-\mu}(\mathbb{R}^d)$.*

This theorem holds true by Definition 1 when $L = +\infty$. The case $L < +\infty$ can be found in the proof of Theorem 2.1.11 in [21].

2.5 Necessary and sufficient conditions for smooth strongly convex interpolation

We now focus on transforming the smooth strongly convex interpolation problem into a convex interpolation problem. In order to do so, we mainly use the two previously defined operations: conjugation (using Theorem 2) and minimal curvature subtraction (using Theorem 3). The reasoning is the following:

(i) Reformulate the $\mathcal{F}_{\mu,L}$ interpolation problem into a $\mathcal{F}_{0,L-\mu}$ interpolation problem using minimal curvature subtraction.
(ii) Write the $\mathcal{F}_{0,L-\mu}$ interpolation problem into a $\mathcal{F}_{1/(L-\mu),\infty}$ interpolation problem using Legendre–Fenchel conjugation.
(iii) Transform the $\mathcal{F}_{1/(L-\mu),\infty}$ interpolation problem into a $\mathcal{F}_{0,\infty}$ interpolation problem using again minimal curvature subtraction.

The effect of minimal curvature subtraction on our interpolation problem, used in steps (i) and (iii), is described by the following Lemma.

Lemma 1 *Consider a finite set $\{(x_i, g_i, f_i)\}_{i \in I}$ with $x_i, g_i \in \mathbb{R}^d$ and $f_i \in \mathbb{R}$. The following propositions are equivalent for any constants $0 \leq \mu < L \leq +\infty$:*

(a) $\{(x_i, g_i, f_i)\}_{i \in I}$ *is $\mathcal{F}_{\mu,L}$-interpolable,*
(b) $\left\{\left(x_i, g_i - \mu x_i, f_i - \frac{\mu}{2}\|x_i\|_2^2\right)\right\}_{i \in I}$ *is $\mathcal{F}_{0,L-\mu}$-interpolable.*

Proof [$(a) \Rightarrow (b)$] It follows from Theorem 3 that if there exists $f \in \mathcal{F}_{\mu,L}(\mathbb{R}^d)$ interpolating the set, then $h(x) = f(x) - \frac{\mu}{2}\|x\|_2^2$ exists and satisfies $h \in \mathcal{F}_{0,L-\mu}(\mathbb{R}^d)$ and $\forall i \in I$:

$$h(x_i) = f_i - \frac{\mu}{2}\|x_i\|_2^2, \quad g_i - \mu x_i \in \partial h(x_i).$$

Hence, the function $h \in \mathcal{F}_{0,L-\mu}(\mathbb{R}^d)$ interpolates the set $\{(x_i, g_i - \mu x_i, f_i - \frac{\mu}{2}\|x_i\|_2^2)\}_{i \in I}$.

[$(a) \Leftarrow (b)$] If such a $h \in \mathcal{F}_{0,L-\mu}(\mathbb{R}^d)$ exists and satisfies the interpolation conditions (b), then one can reconstruct a function $f(x) = h(x) + \frac{\mu}{2}\|x\|_2^2$, $f \in \mathcal{F}_{\mu,L}(\mathbb{R}^d)$ which interpolates the set $\{(x_i, g_i, f_i)\}_{i \in I}$. $\qquad\square$

The effect of conjugation in step (ii) of the reduction procedure is precisely described in the following lemma.

Lemma 2 *Consider a finite set $\{(x_i, g_i, f_i)\}_{i \in I}$ with $x_i, g_i \in \mathbb{R}^d$ and $f_i \in \mathbb{R}$. The following propositions are equivalent for any constant $0 < L \leq +\infty$:*

(a) $\{(x_i, g_i, f_i)\}_{i \in I}$ *is $\mathcal{F}_{0,L}$-interpolable,*
(b) $\left\{\left(g_i, x_i, x_i^\top g_i - f_i\right)\right\}_{i \in I}$ *is $\mathcal{F}_{1/L,\infty}$-interpolable.*

Proof [$(a) \Rightarrow (b)$] It follows from Theorem 2 that if there exists $f \in \mathcal{F}_{0,L}(\mathbb{R}^d)$ then f^* exists and satisfies $f^* \in \mathcal{F}_{1/L,\infty}(\mathbb{R}^d)$. In addition to that, if both $f \in \mathcal{F}_{0,L}(\mathbb{R}^d)$ and f^* exists, then they satisfy $\forall i \in I$ the three conditions (e.g., Theorem 23.5 of [23]):

$$f(x_i) + f^*(g_i) = g_i^\top x_i, \quad g_i \in \partial f(x_i), \quad x_i \in \partial f^*(g_i).$$

[$(b) \Rightarrow (a)$] If a function $f^* \in \mathcal{F}_{1/L,\infty}(\mathbb{R}^d)$ exists and satisfies the interpolation conditions (b), then the conjugate f^{**} (which is convex, proper and closed by construction) satisfies $f^{**} \in \mathcal{F}_{0,L}(\mathbb{R}^d)$ by Theorem 2, as well as the interpolation conditions (e.g., Theorem 23.5 of [23]) $\forall i \in I$:

$$f^{**}(x_i) + f^*(g_i) = g_i^\top x_i, \quad g_i \in \partial f^{**}(x_i), \quad x_i \in \partial f^*(g_i).$$

We obtain the desired result by choosing $f = f^{**}$. $\qquad\square$

We are now properly armed in order to define all interpolation equivalences. Let us now use steps (i), (ii) and (iii) to prove the main theorem of this section.

Theorem 4 ($\mathcal{F}_{\mu,L}$-*interpolability*) *Set* $\{(x_i, g_i, f_i)\}_{i \in I}$ *is* $\mathcal{F}_{\mu,L}$-*interpolable if and only if the following set of conditions holds for every pair of indices* $i \in I$ *and* $j \in I$

$$
\begin{aligned}
f_i &- f_j - g_j^\top(x_i - x_j) \\
&\geq \frac{1}{2(1 - \mu/L)} \left(\frac{1}{L}\|g_i - g_j\|_2^2 + \mu\|x_i - x_j\|_2^2 - 2\frac{\mu}{L}(g_j - g_i)^\top(x_j - x_i) \right).
\end{aligned}
\tag{4}
$$

Proof We begin by showing the equivalence between the following propositions:

(a) $\{(x_i, g_i, f_i)\}_{i \in I}$ is $\mathcal{F}_{\mu,L}$-interpolable,
(b) $\left\{ \left(x_i, g_i - \mu x_i, f_i - \frac{\mu}{2}\|x_i\|_2^2 \right) \right\}_{i \in I}$ is $\mathcal{F}_{0,L-\mu}$-interpolable,
(c) $\left\{ \left(g_i - \mu x_i, x_i, x_i^\top g_i - f_i - \frac{\mu}{2}\|x_i\|_2^2 \right) \right\}_{i \in I}$ is $\mathcal{F}_{1/(L-\mu),\infty}$-interpolable,
(d) $\left\{ \left(g_i - \mu x_i, \frac{Lx_i}{L-\mu} - \frac{g_i}{L-\mu}, \frac{Lx_i^\top g_i}{L-\mu} - f_i - \frac{L\mu\|x_i\|_2^2}{2(L-\mu)} - \frac{\|g_i\|_2^2}{2(L-\mu)} \right) \right\}_{i \in I}$
   is $\mathcal{F}_{0,\infty}$-interpolable,
(e) $\left\{ \left(\frac{Lx_i}{L-\mu} - \frac{g_i}{L-\mu}, g_i - \mu x_i, \frac{\mu x_i^\top g_i}{L-\mu} + f_i - \frac{L\mu\|x_i\|_2^2}{2(L-\mu)} - \frac{\|g_i\|_2^2}{2(L-\mu)} \right) \right\}_{i \in I}$
   is $\mathcal{F}_{0,\infty}$-interpolable.

Both $(a) \Leftrightarrow (b)$ and $(c) \Leftrightarrow (d)$ are direct applications of Lemma 1, whereas both $(b) \Leftrightarrow (c)$ and $(d) \Leftrightarrow (e)$ are direct applications of Lemma 2. Theorem 4 follows from equivalence between propositions (a) and (e) applied to the necessary and sufficient conditions for convex interpolation of Theorem 1. Finally, it is straightforward to check that condition (e) reduces to the statement of the theorem. $\qquad\square$

Remark 2 Note that one can also easily construct an interpolating function $f(x)$ for the original set of points from Theorem 4(a). It follows from Theorem 1 that a possible interpolating function for the set $\left\{ \left(\tilde{x}_i, \tilde{g}_i, \tilde{f}_i \right) \right\}_{i \in I}$ of Theorem 4(c) is given by

$$
h(\tilde{x}) = \max_i \left\{ \tilde{f}_i + \tilde{g}_i^\top(\tilde{x} - \tilde{x}_i) + \frac{1}{2(L - \mu)}\|\tilde{x} - \tilde{x}_i\|_2^2 \right\} = \max_i \ h_i(\tilde{x}).
$$

This can be conjugated into an interpolating function $h^*(x)$ of the set given by Theorem 4(b). Using Theorem 16.5 from [23], this can equivalently be written in the form
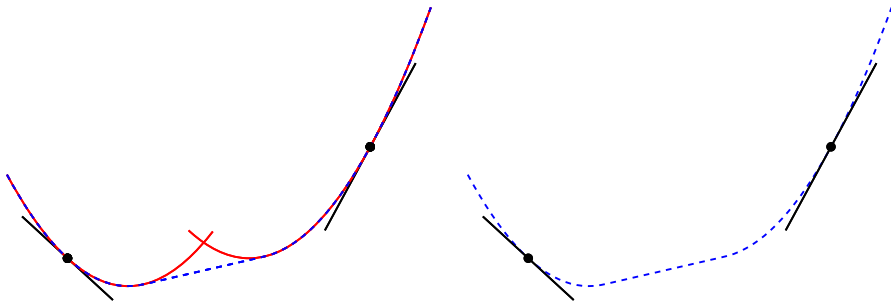
Fig. 2 Example of an interpolating function; the data triples to be interpolated by a 1-smooth convex function are $(x_1, g_1, f_1) = (2, 2, 3)$ and $(x_2, g_2, f_2) = (-3, -1, 1)$. Figure shows the upper-bounding quadratic functions $h_i^*(x)$ (*red, left*), the interpolating function $f(x) = \mathrm{conv}\left(h_i^*(x)\right)$ (*dashed blue*) and the gradients (*black tangents*) (color figure online)

$$h^*(x) = \mathrm{conv}\left(h_i^*(x)\right),$$

where the conv operation takes the convex hull of the epigraphs of the h_i^*'s. Hence an interpolating function for the original set $\{(x_i, g_i, f_i)\}_{i \in I}$ is given by

$$f(x) = \mathrm{conv}\left(h_i^*(x)\right) + \frac{\mu}{2}\|x\|_2^2.$$

We provide an example of such an interpolating function on Fig. 2.

It is straightforward to establish the equivalent interpolation conditions for both the smooth but non-strongly convex case ($\mu = 0$) and the nonsmooth strongly convex case ($L = +\infty$). In the first case—given by Corollary 1—we find the discrete version of the well-known inequality characterizing L-smooth convex functions, which turns out to be necessary and sufficient:

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

Corollary 1 *The finite set $\{(x_i, g_i, f_i)\}_{i \in I}$ is $\mathcal{F}_{0,L}$-interpolable if and only if*

$$f_i \geq f_j + g_j^\top(x_i - x_j) + \frac{1}{2L}\|g_i - g_j\|_2^2, \qquad \forall i, j \in I.$$

Nonsmooth strongly convex interpolation conditions are given in Corollary 2, which corresponds to the well-known inequality characterizing the subgradients of strongly convex functions.

Corollary 2 *The finite set $\{(x_i, g_i, f_i)\}_{i \in I}$ is $\mathcal{F}_{\mu,\infty}$-interpolable if and only if*

$$f_i \geq f_j + g_j^\top(x_i - x_j) + \frac{\mu}{2}\|x_i - x_j\|_2^2, \qquad \forall i, j \in I.$$

Remark 3 Following the spirit of Remark 1, we note that Theorem 4 can readily be extended to handle the finite (and continuous) integration problems for L-smooth μ-strongly convex functions (i.e., interpolation without function values). Indeed, summing inequality (4) from Theorem 4 over each pair in the cyclic sequence $\{(i_1, i_2), (i_2, i_3), \ldots, (i_{N-1}, i_N), (i_N, i_1)\}$ produces a necessary inequality that does not involve function values f_i. Moreover one can show that the set of those inequalities for all possible sequences is necessary and sufficient for finite convex integration of L-smooth μ-strongly convex functions, generalizing the standard cyclic monotonicity conditions. As an illustration, note that the following inequality

$$(g_i - g_j)^\top (x_i - x_j) \geq \frac{1}{1 + \mu/L} \left(\frac{1}{L} \|g_i - g_j\|_2^2 + \mu \|x_i - x_j\|_2^2 \right),$$

that is standard in the analysis of gradient methods on smooth strongly convex functions (see e.g., Theorem 2.1.12 of [21]), corresponds to cycles of length 2 (i.e., cyclic sequences $\{(i, j), (j, i)\}$). The set of all such inequalities is necessary but not sufficient, as it omits longer cycles.

3 A convex formulation for performance estimation

As explained in the introduction, our performance estimation problem can now be expressed in terms of the iterates and optimal point $\{x_i, g_i, f_i\}_{i \in \{0, \ldots, N, *\}}$ only, using the interpolation conditions given by Theorem 4.

As our class of functions $\mathcal{F}_{\mu,L}(\mathbb{R}^d)$ and the first-order methods we study are invariant with respect to both an additive shift in the function values and a translation in their domain, we can assume without loss of generality that $x_* = 0$ and $f_* = 0$, which will simplify our derivations. We can also assume $g_* = 0$, from the optimality conditions of unconstrained optimization. The problem can now be stated in its finite-dimensional formulation:

$$w_{\mu,L}^{(d)}(R, \mathcal{M}, N, \mathcal{P}) = \sup_{\{x_i, g_i, f_i\}_{i \in I} \in \left(\mathbb{R}^d \times \mathbb{R}^d \times \mathbb{R} \right)^{N+2}} \mathcal{P}\left(\{x_i, g_i, f_i\}_{i \in I} \right), \quad \text{(d-PEP)}$$

such that $\{x_i, g_i, f_i\}_{i \in I}$ is $\mathcal{F}_{\mu,L}$ − interpolable,

$x_1, \ldots, x_N$ is generated from x_0 by method $\mathcal{M}$ with (2),

$\{x_*, g_*, f_*\} = \{0^d, 0^d, 0\}$ and $\|x_0 - x_*\|_2 \leq R.$

Problem (d-PEP) is an instance of (f-PEP) where the function class $\mathcal{F}$ is chosen to be $\mathcal{F}_{\mu,L}(\mathbb{R}^d)$, the set of d-dimensional L-smooth μ-strongly convex functions, hence we have $w(\mathcal{F}_{\mu,L}(\mathbb{R}^d), R, \mathcal{M}, N, \mathcal{P}) = w_{\mu,L}^{(d)}(R, \mathcal{M}, N, \mathcal{P})$. Interestingly, in most situations of interest, quantity $w_{\mu,L}^{(d)}(R, \mathcal{M}, N, \mathcal{P})$ is monotonically increasing with d, as a higher dimensional function can usually mimic a lower dimensional one (see Theorem 5 and subsequent results for a discussion on finite convergence of this sequence).

Finally, note that problem (d-PEP) is not convex, as it involves several non-convex quadratic constraints (e.g., $g_j^\top x_i$ terms in the interpolation conditions). In the next

section, we show how (d-PEP) can be cast as a convex semidefinite program [26] when dealing with a certain class of first-order black-box methods, those based on fixed steps.

3.1 Fixed-step first-order methods

We hereby restrict ourselves to the class of fixed-step first-order methods, where each iterate is obtained by adding a series of gradients steps with fixed step sizes to the starting point x_0.

Definition 4 A method $\mathcal{M}$ is called a *fixed-step method* if its iterates are computed according to

$$x_i = x_0 - \sum_{k=0}^{i-1} h_{i,k} g_k.$$

with fixed scalar coefficients $h_{i,k}$.

A fixed-step method performing N steps is completely defined by the lower triangular $N \times N$ matrix $H = \{h_{i,k}\}_{1 \le i \le N, 0 \le k \le N-1}$ (where $h_{i,k}$ is defined to be zero if $k \ge i$). Note that many classical methods such as the gradient method with constant step size (GM) and the fast gradient method (FGM) are included in this class of algorithms (see the details in Sect. 4).

3.2 A convex reformulation using a Gram matrix

In order to obtain a convex formulation for (d-PEP), we introduce a Gram matrix to describe the iterates and their gradients. Denoting

$$P = [g_0 \ g_1 \ \cdots \ g_N \ x_0]$$

we define the symmetric $(N + 2) \times (N + 2)$ Gram matrix $G = P^\top P \in \mathbb{S}^{N+2}$, which is equivalent to

$$G = \{G_{i,j}\}_{0 \le i,j \le N} \quad \text{with} \quad \begin{cases} G_{i,j} = g_i^\top g_j & \text{for any } 0 \le i, j \le N, \\ G_{N+1,j} = x_0^\top g_j & \text{for any } 0 \le j \le N, \\ G_{i,N+1} = g_i^\top x_0 & \text{for any } 0 \le i \le N, \\ G_{N+1,N+1} = x_0^\top x_0 \end{cases}$$

(note that the size of this matrix does not depend on the dimension of iterate x_0 and gradients g_i).

The constraints in problem (d-PEP) can now be entirely formulated in terms of the entries of the Gram matrix G along with the function values f_i. Indeed all iterates apart from x_0 can be substituted out of the formulation using Definition 4 of a fixed-step method, and the resulting formulation only involves function values f_i and inner products between x_0 and all gradients g_i.

Note that the initial iterate x_0 and successive gradients g_i of any solution to problem (d-PEP) can be transformed into a symmetric and positive semidefinite Gram matrix G. Moreover, since vectors x_0 and g_i belong to $\mathbb{R}^d$, matrix G has rank at most d. In the other direction, it is easy to see that any symmetric and positive semidefinite Gram matrix G of rank at most d can be converted back (using Cholesky decomposition for example) into $N + 2$ vectors $x_0 \in \mathbb{R}^d$ and $g_i \in \mathbb{R}^d$ which describe the initial iterate and successive gradients of a d-dimensional function (this transformation is however not unique). From those observations we can anticipate that an equivalent formulation of (d-PEP) will rely on imposing that G is symmetric and positive semidefinite, which is a convex constraint and will naturally lead to a semidefinite program.

3.3 Exact worst-case performance of fixed-step first-order methods as a semidefinite program

For notational convenience, we define vectors $h_i \in \mathbb{R}^{N+2}$ for any i between 0 and N and $h_* \in \mathbb{R}^{N+2}$ as follows

$$h_i^\top = [-h_{i,0} \ -h_{i,1} \ \cdots \ -h_{i,i-1} \ 0 \ \cdots \ 0 \ 1], \quad h_*^\top = [0 \ \cdots \ 0],$$

so that we have $x_i = Ph_i$. In order to lighten the notations we also define $u_i = e_{i+1} \in \mathbb{R}^{N+2}$, the canonical basis vectors, and u_* the vector of zeros. Using those notations, we rewrite the interpolation constraints (4) coming from Theorem 4 in the following form:

$$f_i \geq f_j + \frac{L}{L-\mu}(u_j^\top Gh_i - u_j^\top Gh_j) + \frac{1}{2(L-\mu)}(u_i - u_j)^\top G(u_i - u_j)$$

$$+ \frac{\mu}{L-\mu}(u_i^\top Gh_j - u_i^\top Gh_i) + \frac{L\mu}{2(L-\mu)}(h_i - h_j)^\top G(h_i - h_j), \quad \text{for all } i,j \in I.$$

We can equivalently formulate all constraints using the trace operator, and add the distance constraint $\|x_0 - x_*\|_2 \leq R$ on the starting point as well as the positive semidefiniteness constraint for G. Defining matrices A_{ij} and A_R in the following way

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

$$+ \frac{\mu}{L-\mu}\left(u_i(h_j - h_i)^\top + (h_j - h_i)u_i^\top\right) + \frac{L\mu}{L-\mu}(h_i - h_j)(h_i - h_j)^\top, \quad \text{for all } i,j \in I,$$

$$A_R = u_{N+1}u_{N+1}^\top.$$

we obtain the following compact formulation for the feasible region that is *linear* in its variables $f \in \mathbb{R}^{N+1}$ and $G \in \mathbb{S}^{N+2}$

$$f_j - f_i + \text{Tr}\left(GA_{ij}\right) \leq 0, \qquad \text{for all } i, j \in I,$$

$$\text{Tr}\left(GA_R\right) - R^2 \leq 0,$$

$$G \succeq 0.$$

From the discussion at the end of the previous section, it is easy to see that any d-dimensional function f and starting point $x_0 \in \mathbb{R}^d$ produce a feasible solution (f, G) where matrix G has rank at most d. On the other hand, any feasible solution (f, G) where G has rank at most d can be interpolated into a d-dimensional function $f \in \mathcal{F}_{\mu,L}(\mathbb{R}^d)$ and a starting point $x_0 \in \mathbb{R}^d$. Indeed, matrix $G = P^\top P$ provides $x_0 \in \mathbb{R}^d$ and $N + 1$ successive gradients $g_i \in \mathbb{R}^d$, while the other iterates x_i derive from the definition of the method. Our interpolating conditions ensure that a function compatible with these data triples $\{x_i, g_i, f_i\}_{i \in I}$ exists.

Considering finally the performance criterion $\mathcal{P}$, we observe that any concave semidefinite-representable function in G and f leads to a worst-case estimation problem that can be cast as a convex semidefinite optimization problem (see e.g., [5]), and that such a criteria does not depend on the dimension d of the functions. In particular, linear functions of the entries of f and G are suitable. Classical performance criteria such as $f(x_N) - f_*$, $\|\nabla f(x_N)\|_2^2$ and $\|x_N - x_*\|_2^2$ are indeed covered by this formulation. We focus below on the case of a linear performance criterion, but note that other criteria can be useful (see for example a concave piecewise linear criteria used in Sect. 4.3).

We can now state the main result of this paper.

Theorem 5 *Consider the class $\mathcal{F}_{\mu,L}(\mathbb{R}^d)$ of L-smooth μ-strongly convex functions with $0 \le \mu < L \le \infty$, a fixed-step first-order method that computes N iterates according to matrix $H \in \mathbb{R}^{N \times N}$, and a performance criterion $\mathcal{P}_{b,C}(f, G) = b^\top f +$ $\mathrm{Tr}\,(CG)$ that depends linearly on the function values at those iterates and quadratically on the iterates and their gradients ($b \in \mathbb{R}^{N+1}$ and $C \in \mathbb{S}^{N+2}$).*

If $N \le d - 2$, the worst-case performance after N iterations of method H applied to some function in $\mathcal{F}_{\mu,L}(\mathbb{R}^d)$ is equal to the optimal value of the following semidefinite program

$$w_{\mu,L}^{sdp}(R, H, N, b, C) = \sup_{G \in \mathbb{S}^{N+2}, f \in \mathbb{R}^{N+1}} b^\top f + \mathrm{Tr}\,(CG) \qquad \text{(sdp-PEP)}$$
$$\text{such that } f_j - f_i + \mathrm{Tr}\,(GA_{ij}) \le 0, \quad i, j \in I,$$
$$\mathrm{Tr}\,(GA_R) - R^2 \le 0,$$
$$G \succeq 0,$$

with matrices A_{ij} and A_R as defined above. In others words,

$$w_{\mu,L}^{sdp}(R, H, N, b, C) = w_{\mu,L}^{(d)}(R, \mathcal{M}, N, \mathcal{P}_{b,C}) \text{ for any } d \ge N + 2.$$

Proof We have already shown the two-way correspondence between functions in $\mathcal{F}_{\mu,L}(\mathbb{R}^d)$ and feasible solutions of this problem where matrix G has rank at most d. Since matrix G has size $N + 2$, it has rank at most $N + 2$, which establishes that this semidefinite program is a correct formulation of the performance estimation problem when $d \ge N + 2$.   □

The optimal value $w_{\mu,L}^{sdp}(R, H, N, b, C)$ of (sdp-PEP) is not necessarily finite or attained at some feasible point. However, when L is finite, any continuous performance

criterion $\mathcal{P}$ will force the optimal value to be attained and finite, because the constraints on variables f and G force the domain to be compact.

Proposition 1 *Under the assumptions of Theorem 5, the optimum of (sdp-PEP) is attained and finite when $L < \infty$.*

Proof To show that the solution of (sdp-PEP) is attained and finite, it suffices to prove that its feasible region is compact (since the objective is continuous). We first prove that the iterates of method H applied to any function in $\mathcal{F}_{\mu,L}(\mathbb{R}^d)$ with $L < \infty$ are bounded, as well as their gradients.

Note that the Lipschitz condition on the gradients (C1f) with $j = *$ shows that if iterate x_i is bounded, gradient g_i is also bounded. We proceed by recurrence. We start with the fact that x_0 is bounded, using the assumption that $x_* = 0$ and constraint $\|x_0 - x_*\|_2 \leq R$. This implies that g_0 is bounded, hence that x_1 is bounded using Definition 4 of a fixed-step method. This implies in turn that g_1 is bounded, then that x_2 is bounded, and so on until we have shown that all iterates and gradients are bounded.

Condition (C2f) with $j = *$ then implies that function values f_i are bounded. Therefore all entries in variables f and G are bounded which, combined with closedness of the feasible region, establishes the claim.   □

Remark 4 When $L = +\infty$ (recall the conventions $1/+\infty = 0$ and $+\infty - \mu = +\infty$ used in this paper), the feasible region may be unbounded and it is possible to design feasible functions which drive standard performance criteria arbitrarily away from 0. Nevertheless, performance estimation on such nonsmooth functions could still be tackled after introduction of another appropriate Lipschitz condition on the class of functions, such as $||g_i||_2 \leq L$. We leave this as a topic for further research and, in the rest of this text, restrict ourselves to the smooth case $L < +\infty$.

Our formulation (sdp-PEP) is dimension-independent, and computes the exact worst-case performance of a first-order method with N steps as long as the class of functions of interest contains functions of dimension at least $N + 2$. This corresponds to the so-called large-scale optimization setting, which is usually assumed when analyzing the worst-case of first-order methods. Function classes with smaller dimensions can also be handled with the addition of a (non-convex) rank constraint on G. We obtain the following result, whose proof is straightforward.

Proposition 2 *Consider the setting of Theorem 5. If $d < N + 2$, the worst-case performance after N iterations of method H applied to some function in $\mathcal{F}_{\mu,L}(\mathbb{R}^d)$ is equal to the optimal value of the semidefinite program (sdp-PEP) under the additional constraint* rank $G \leq d$.

Note that this also establishes that the sequence $\{w_{\mu,L}^{(d)}(R, \mathcal{M}, N, \mathcal{P}_{b,C})\}_{d=1,2,\cdots}$ is monotonically increasing, and that it converges for a finite value of d.

Corollary 3 *The worst-case performance after N steps of a fixed-step method on a L-smooth (μ-strongly) convex function is achieved by an $N+2$-dimensional function.*

Finally, note that, when applied to the gradient method in the non-strongly convex case ($\mu = 0$), problem (sdp-PEP) is equivalent to one of the formulations proposed in [10], more specifically to their problem (G). Theorem 5 establishes that this relaxation is in fact exact.

3.4 A dual semidefinite program to generate upper bounds

In general, it is not easy to find an analytical optimal solution to (sdp-PEP). Hence, we are also interested in a generic and easier way of obtaining analytical upper bounds on the performance of a given algorithm. A classical way of doing so is to work with the Lagrangian dual of (sdp-PEP):

$$\inf_{\lambda_{ij},\tau} \tau R^2 \text{ such that } \tau A_R - C + \sum_{i,j\in I} \lambda_{ij} A_{ij} \succeq 0, \qquad \text{(d-sdp-PEP)}$$

$$b - \sum_{i,j\in I} \lambda_{ij}(u_j - u_i) = 0,$$

$$\lambda_{ij} \geq 0, \qquad i,j \in I,$$

$$\tau \geq 0,$$

whose feasible solutions will provide theoretical upper bounds on the worst-case behavior of every fixed-step first-order method (using weak duality). Note that the final dual formulation used in [10], which deals with the case $\mu = 0$, can be recovered by taking $\lambda_{ij} = 0$ for $i + 1 \neq j$ or $i \neq *$ in our dual, i.e., it is a restriction of (d-sdp-PEP) with a potentially larger optimal value.

The next theorem guarantees that no duality gap occurs between (sdp-PEP) and (d-sdp-PEP) under the technical assumption that $h_{i,i-1} \neq 0$ ($i \in \{1, \cdots, N\}$). This assumption is reasonable as it only implies that, at each iteration, the most recent gradient obtained from the oracle has to be used in the computation of the next iterate. The theorem will also guarantee the existence of a dual feasible point attaining the optimal value of the primal-dual pair of estimation problems (sdp-PEP) and (d-sdp-PEP), i.e., a tight upper bound on the worst-case performance of the considered method.

Theorem 6 *The optimal value of the dual problem* (d-sdp-PEP) *with* $0 \leq \mu < L < \infty$ *is attained and equal to* $w_{\mu,L}^{(sdp)}(R, H, N, b, c)$ *under the assumptions that* $h_{i,i-1} \neq 0$ *for all* $i \in \{1, \ldots, N\}$.

Proof We use the classical Slater condition [8] on the primal problem in order to guarantee a zero duality gap—that is, we show that (sdp-PEP) has a feasible point with $G \succ 0$. The reasoning is divided in two parts; we consider first the case $\mu = 0$ and $L = 2 + 2\cos(\pi/(N+2))$, and we generalize it to general $\mu < L$ afterwards. Consider the quadratic function $f(x) = \frac{1}{2}x^\top Q x$ with the following tridiagonal positive definite matrix Q

$$Q = \begin{pmatrix} 2 & 1 & 0 & 0 & \cdots & 0 \\ 1 & 2 & 1 & 0 & \cdots & 0 \\ 0 & 1 & 2 & 1 & \ddots & \vdots \\ \vdots & & \ddots & \ddots & \ddots & \end{pmatrix} \succ 0.$$

We show how to construct a full-rank G feasible for (sdp-PEP) using the values of the quadratic function f. In order to do so, we exhibit a full-rank matrix

$$P = [x_0 \ g_0 \ g_1 \cdots g_N]$$

corresponding to the application of a given method (with $h_{i,i-1} \neq 0$) to the quadratic function f. Indeed, choosing $x_0 = Re_1$, we can show that P is upper triangular with non-zero diagonal entries. Then we have

$$\begin{aligned} g_0 &= Qx_0 = 2e_1 + e_2, \\ x_1 &= x_0 - h_{1,0}g_0, \\ g_1 &= Qx_1 = g_0 - h_{1,0}Qg_0 = 2e_1 + e_2 - h_{1,0}(4e_1 + 4e_2 + e_3). \end{aligned}$$

Hence g_1 has a non-zero element associated with e_3 whereas the only non-zero elements of g_0 are associated with e_1 and e_2. Now, we assume that g_{i-1} has a non-zero element corresponding to e_{i+1} and zero elements corresponding to e_k for all $k > i+1$, while all previous gradients have zero components corresponding to e_k for all $k > i$. Then we have

$$g_i^\top e_{i+2} = x_i^\top Q e_{i+2} = x_i^\top (e_{i+1} + 2e_{i+2} + e_{i+3}),$$

with $x_i^\top e_{i+2} = x_i^\top e_{i+3} = 0$ and $x_i^\top e_{i+1} \neq 0$ because of the recurrence assumption and the iterative form of the algorithm:

$$x_i^\top e_{i+1} = \underbrace{x_0^\top e_{i+1}}_{=0} - \sum_{k=0}^{i-2} h_{i,k} \underbrace{g_k^\top e_{i+1}}_{=0} - h_{i,i-1} \underbrace{g_{i-1}^\top e_{i+1}}_{\neq 0},$$

$$x_i^\top e_{i+2} = \underbrace{x_0^\top e_{i+2}}_{=0} - \sum_{k=0}^{i-2} h_{i,k} \underbrace{g_k^\top e_{i+2}}_{=0} - h_{i,i-1} \underbrace{g_{i-1}^\top e_{i+2}}_{=0},$$

$$x_i^\top e_{i+3} = \underbrace{x_0^\top e_{i+3}}_{=0} - \sum_{k=0}^{i-2} h_{i,k} \underbrace{g_k^\top e_{i+3}}_{=0} - h_{i,i-1} \underbrace{g_{i-1}^\top e_{i+3}}_{=0}.$$

Hence, g_i has a non-zero element associated with e_{i+2}. We deduce that the following components are equal to zero by computing $g_i^\top e_{i+2+k}$ for $k > 0$:

$$g_i^\top e_{i+2+k} = x_i^\top Q e_{i+2+k} = x_i^\top (e_{i+1+k} + 2e_{i+2+k} + e_{i+3+k}),$$

which is zero because of the algorithmic structure of x_i, i.e.,

$$x_i^\top e_{i+1+k} = \underbrace{x_0^\top e_{i+1+k}}_{=0} - \sum_{k=0}^{i-2} h_{i,k} \underbrace{g_k^\top e_{i+1+k}}_{=0} - h_{i,i-1} \underbrace{g_{i-1}^\top e_{i+1+k}}_{=0}.$$

Hence, P is an upper triangular matrix with positive entries on the diagonal, and is therefore full-rank. In order to make this statement hold for general $\mu < L$, observe that the structure of the matrix is preserved using the operation (I_{N+2} is the identity matrix)

$$Q' = (Q - \lambda_{\min}(Q)I_{N+2}) \frac{(L - \mu)}{\lambda_{\max}(Q) - \lambda_{\min}(Q)} + \mu I_{N+2}.$$

The corresponding quadratic function is easily seen to be L-smooth and μ-strongly convex. Therefore, the interior of the domain of (sdp-PEP) is non-empty and Slater's condition applies for $\mu < L$, ensuring that no duality gap occurs and that the dual optimal value is attained. □

One can note that Theorem 6 guarantees the existence of a fully explicit proof (i.e., a combination of valid inequalities, or equivalently, a dual feasible solution) for any worst-case function (see the example at the end of this section).

3.5 Homogeneity of the optimal values with respect to L and R

We observe that, for most performance criteria, one can predict how the worst-case performance depends from parameters L and R, provided the fixed step sizes contained in H are scaled appropriately (i.e., inversely proportional to L). In the rest of this paper we will only consider such scaled (normalized) step sizes. Therefore, the corresponding performance estimation problems have only to be solved numerically in the case $R = 1$ and $L = 1$, from which a general bound valid for any L and R can be deduced.

More specifically, a classical reasoning involving appropriate scaling operations easily leads to the following homogeneity relations for the standard criteria $f(x_N) - f_*$, $\|\nabla f(x_N)\|_2$ and $\|x_N - x_*\|_2$:

$$w_{\mu,L}^{sdp}(R, H/L, N, f(x_N) - f_*) = LR^2 \, w_{\kappa,1}^{sdp}(1, H, N, f(x_N) - f_*),$$
$$w_{\mu,L}^{sdp}(R, H/L, N, \|\nabla f(x_N)\|_2^2) = LR \, w_{\kappa,1}^{sdp}(1, H, N, \|\nabla f(x_N)\|_2^2),$$
$$w_{\mu,L}^{sdp}(R, H/L, N, \|x_N - x_*\|_2^2) = R \, w_{\kappa,1}^{sdp}(1, H, N, \|x_N - x_*\|_2^2),$$

where $\kappa = \mu/L$ is the inverse condition number and H/L describes the fixed-step method obtained by dividing all step sizes $h_{i,j}$ by the Lipschitz constant L. Results in the rest of this paper implicitly rely on these relations.

3.6 A simple example

Consider the very simple case of a method performing a single gradient step using the non-standard step-size $\frac{3}{2L}$, i.e., $x_1 = x_0 - \frac{3}{2L} \nabla f(x_0)$ (this is actually conjectured to be the best possible step size for a single step, see Sect. 4.1.1). One wishes to estimate the worst-case objective function accuracy after taking that step, i.e., maximize $f(x_1) - f_*$, over all L-smooth convex functions. Solving the corresponding semidefinite formulation (sdp-PEP) with $\mu = 0$, $N = 1$, $H = \left(\frac{3}{2}\right)$ and $\mathcal{P}_{b,C}(f, G) = f_1$ provides the optimal value

$$w_{0,L}^{sdp}\left(R, \left(\tfrac{3}{2}\right), 1, \begin{pmatrix} 0 \\ 1 \end{pmatrix}, 0^{3\times3}\right) = \frac{LR^2}{8},$$

attained by the following optimal solution

$$f_0 = \frac{LR^2}{2}, \quad f_1 = \frac{LR^2}{8} \quad \text{and} \quad G = LR^2 \begin{pmatrix} L & -L/2 & 1 \\ -L/2 & L/4 & -1/2 \\ 1 & -1/2 & 1/L \end{pmatrix} \succeq 0.$$

This means that $f(x_1) - f_* \leq \frac{LR^2}{8}$ holds for any $f \in \mathcal{F}_{0,L}(\mathbb{R}^d)$ for any d and provided that $\|x_0 - x_*\| \leq R$. It is easy to check that function $f(x) = \frac{L}{2}x^2 \in \mathcal{F}_{0,L}(\mathbb{R})$ achieves this worst-case when started from $x_0 = R$. Indeed one can successively evaluate $f_0 = f(x_0) = \frac{LR^2}{2}$, $g_0 = \nabla f(x_0) = LR$, $x_1 = R - \frac{3}{2}R = -\frac{R}{2}$, $f_1 = f(x_1) = \frac{LR^2}{8}$ and $g_1 = -\frac{LR}{2}$. This function is one-dimensional since the optimal G has rank one (note that Corollary 3 only guaranteed the existence of a three-dimensional worst-case).

Solving the dual problem (d-sdp-PEP) leads to the same optimal value $\frac{LR^2}{8}$, attained by optimal multipliers $\lambda_{01} = \lambda_{*0} = \lambda_{*1} = \frac{1}{2} \geq 0$ and $\tau = \frac{L}{8}$. The corresponding dual slack matrix is

$$S = \frac{1}{2} \begin{pmatrix} 1/L & 1/L & -1/2 \\ 1/L & 1/L & -1/2 \\ -1/2 & -1/2 & L/4 \end{pmatrix} = \frac{L}{2} \begin{pmatrix} -1/L \\ -1/L \\ 1/2 \end{pmatrix} \left(-1/L \;\; -1/L \;\; 1/2\right) \succeq 0.$$

From this dual solution, a fully explicit proof of the worst-case performance can be derived, which can be checked independently without any knowledge about our approach. Indeed, linear equalities in the dual imply that the objective accuracy $f(x_1) - f_*$ can be written exactly as follows

$$\begin{aligned} &f(x_1) - f(x_*) \\ &= \frac{1}{2}\left(f(x_1) - f(x_0) + \nabla f(x_1)^\top (x_0 - x_1) + \frac{1}{2L}\|\nabla f(x_0) - \nabla f(x_1)\|_2^2 \right) \\ &\quad + \frac{1}{2}\left(f(x_0) - f(x_*) + \nabla f(x_0)^\top (x_* - x_0) + \frac{1}{2L}\|\nabla f(x_0) - \nabla f(x_*)\|_2^2 \right) \end{aligned}$$

$$+\frac{1}{2}\left(f(x_1) - f(x_*) + \nabla f(x_1)^\top (x_* - x_1) + \frac{1}{2L}\|\nabla f(x_1) - \nabla f(x_*)\|_2^2\right)$$

$$+\frac{L}{8}\|x_0 - x_*\|^2 - \frac{L}{2}\left\|\frac{1}{2}(x_0 - x_*) - \frac{\nabla f(x_0)}{L} - \frac{\nabla f(x_1)}{L}\right\|^2$$

(where for the last term we write the quadratic form Tr (SG) as a square, since S is rank-one). This equality, which is straightforward to check using $x_1 = x_0 - \frac{3}{2L}\nabla f(x_0)$ and $\nabla f(x_*) = 0$, immediately implies inequality $f(x_1) - f_* \leq \frac{L}{8}\|x_0 - x_*\|^2$, since the first three bracketed expressions are nonpositive because of inequalites from Corollary 1 valid for all functions in $\mathcal{F}_{0,L}$.

4 Numerical performance estimation of standard first-order algorithms

In this section we apply the convex PEP formulation to study convergence of the fixed-step gradient method (GM), the standard fast gradient method (FGM) and the optimized gradient method (OGM) proposed by [14]. We begin with the GM for smooth convex optimization, whose worst-case is conjectured in [10] to be attained on a simple one-dimensional function. Numerical experiments with our exact formulation confirm this conjecture. Further experiments on the worst-case complexity for different methods, problem classes and performance criteria lead to a series of conjectures based on worst-case functions possessing a similar shape. We conclude this section with the study of a nonlinear performance criteria corresponding to the smallest gradient norm among all iterates computed by the method.

All numerical results in this section were obtained on an Intel 3.5Ghz desktop computer using a combination of the YALMIP modeling environment in MATLAB [17], the MOSEK [18] and SeDuMi [25] semidefinite solvers and the VSDP (verified semidefinite programming) toolbox [11].

4.1 Gradient method

4.1.1 A conjecture on smooth convex functions by Drori and Teboulle [10]

Consider the classical fixed-step gradient method (GM) with constant step sizes applied to a smooth convex function in $\mathcal{F}_{0,L}(\mathbb{R}^d)$. Following the discussion in Sect. 3.5 we use normalized step sizes $\frac{h}{L}$, inversely proportional to L.

Gradient Method (GM)

Input: $f \in \mathcal{F}_{0,L}(\mathbb{R}^d)$, $x_0 \in \mathbb{R}^d$, $y_0 = x_0$.

For $i = 0 : N - 1$

$$x_{i+1} = x_i - \frac{h}{L}\nabla f(x_i)$$

The following conjecture on the convergence of the worst-case objective function values was made in [10].

Conjecture 1 ([10], Conjecture 3.1.) Any sequence of iterates $\{x_i\}$ generated by the gradient method GM with constant normalized step size $0 \le h \le 2$ on a smooth convex function $f \in \mathcal{F}_{0,L}(\mathbb{R}^d)$ satisfies

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

A proof of the conjecture is provided in [10] for step sizes $0 \le h \le 1$, leaving the case $1 < h < 2$ open. We also recall that the upper bound in this conjecture cannot be improved, as it matches the performance of the GM on two specific one-dimensional functions. Indeed, define

$$f_1(x) = \begin{cases} \frac{LR}{2Nh+1}|x| - \frac{LR^2}{2(2Nh+1)^2} & \text{if } |x| \ge \frac{R}{2Nh+1}, \\ \frac{L}{2}x^2 & \text{else,} \end{cases}$$

$$f_2(x) = \frac{L}{2}x^2.$$

It is straightforward to check that the final objective value accuracy of GM on f_1 is equal to $\frac{LR^2}{2}\frac{1}{2Nh+1}$, and that it is equal to $\frac{LR^2}{2}(1-h)^{2N}$ on f_2. This means that the conjecture can be reformulated as saying that the worst-case behavior of the GM according to objective function accuracy is achieved by function f_1 or f_2, depending on which of the two is worst (which will depend only on the normalized step size h and number of iterations N).

Intuitively, the behavior of GM on piecewise affine-quadratic f_1 corresponds to a situation in which iterates slowly approach the optimal value without oscillating around it (i.e., no overshooting), whereas GM applied on purely quadratic f_2 generates a sequence oscillating around the optimal point. Those behaviors are illustrated on Fig. 3. We also note that iterates for f_1 stay on the affine piece of the function, and even never come close to the quadratic piece. Interestingly, the existence of a one-dimensional worst-case function with a simple affine-quadratic shape will also be observed for the other algorithms studied in this section, both in the smooth convex and in the smooth strongly convex cases.

Empirical results from the numerical resolution of (sdp-PEP) strongly support Conjecture 1. Indeed, when comparing its predictions with numerically computed worst-case bounds, we obtained a maximal relative error of magnitude 10^{-7} (all pairs of values $N \in \{1, 2, \ldots, 30\}$ and $h \in \{0.05, 0.10, \ldots, 1.95\}$ were tested). It is also worth pointing out that the Gram matrices computed numerically correspond to the one-dimensional worst-case functions f_1 and f_2 introduced above.

Before going into the details of other methods, we underline another observation coming from [10]: Conjecture 1 also suggests the existence of an optimal step size $h_{\text{opt}}(N)$ for the GM—optimal in the sense of achieving the lowest worst-case. That is, if you know in advance how many iterations of the GM you will perform, it suggests using a step size $h_{\text{opt}}(N)$ that is the unique minimizer of the right-hand side of the
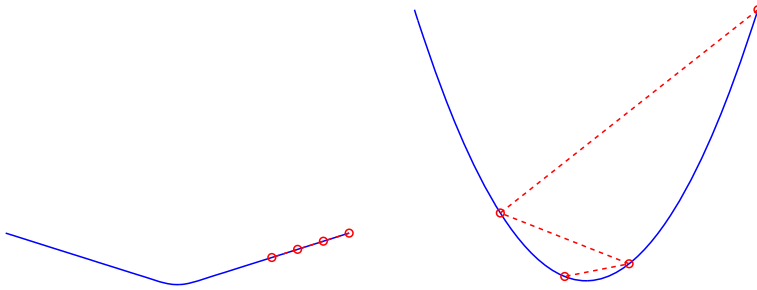
Fig. 3 Behavior of the gradient method on f_1 (*left*) and f_2 (*right*), for $L = R = 1$. We observe that GM does not overshoot the optimal solution on f_1, while it does so at each iteration on f_2

Conjecture 1 for a fixed value of N. It is obtained by solving[1] the following non-linear equation in h_{opt} (for which no closed-form solution seems to be available):

$$\frac{1}{2Nh_{\text{opt}} + 1} = (1 - h_{\text{opt}})^{2N}.$$

This optimal step size can be interpreted in terms of the trade-off between what we obtain on functions f_1 and f_2. On the one hand, we ensure that we are not going too slowly to the optimal point on f_1, and on the other hand we do not want to overshoot too much on f_2.

Assuming Conjecture 1 holds true, one can show that the optimal step size is an increasing function of N with $3/2 \leq h_{\text{opt}}(N) < 2$ and $h_{\text{opt}}(N) \rightarrow 2$ as $N \rightarrow \infty$. More precisely, working out the expression defining h_{opt} gives the following tight lower and upper estimates:

$$2 - \frac{\log 4N}{2N} \sim 1 + (1 + 4N)^{-1/(2N)} \leq h_{\text{opt}}(N)$$

$$\leq 1 + (1 + 2N)^{-1/(2N)} \sim 2 - \frac{\log 2N}{2N}. \tag{5}$$

It is interesting to compare the results from the relaxation (G') proposed for GM in [10] with ours, for values of the normalized step size h that are close to h_{opt}. Indeed, while the results of the two formulations are quite similar for most values of h, it turns out that those from [10] are significantly more conservative in the zone around h_{opt}, as presented in Table 1 for different values of N. This also formally establishes the fact that the formulation from [10] is a strict relaxation of the performance estimation problem.

These numerical results have been obtained with MOSEK, a standard semidefinite optimization solver. Despite convexity of the formulation, it might happen that the solution returned by such as solver is inaccurate, and in particular (slightly) infeasible.

---

[1] This equation possesses several solutions, but the optimum is the unique point where the two terms feature derivatives of opposite signs (a necessary and sufficient condition for the maximum of two convex functions of one variable). This point can easily be computed numerically with an appropriate bisection method.

Table 1 Gradient Method with $\mu = 0$, worst-case computed with relaxation from [10] and worst-case obtained by exact formulation (sdp-PEP) for the criterion $f(x_N) - f^*$. Error is measured relatively to the conjectured result. Results obtained with MOSEK [18]

N	h_{opt}	Conjecture 1	Value computed in [10]	Rel. error	Value from (sdp-PEP)	Rel. error
1	1.5000	$LR^2/8.00$	$LR^2/8.00$	0.00	$LR^2/8.00$	7e–09
2	1.6058	$LR^2/14.85$	$LR^2/14.54$	2e–02	$LR^2/14.85$	5e–09
5	1.7471	$LR^2/36.94$	$LR^2/32.57$	1e–01	$LR^2/36.94$	1e–08
10	1.8341	$LR^2/75.36$	$LR^2/59.80$	3e–01	$LR^2/75.36$	3e–08
20	1.8971	$LR^2/153.77$	$LR^2/109.58$	4e–01	$LR^2/153.77$	6e–08
30	1.9238	$LR^2/232.85$	$LR^2/156.23$	5e–01	$LR^2/232.85$	7e–08
40	1.9388	$LR^2/312.21$	$LR^2/201.10$	6e–01	$LR^2/312.21$	3e–08
50	1.9486	$LR^2/391.72$	$LR^2/244.70$	6e–01	$LR^2/391.72$	1e–07
100	1.9705	$LR^2/790.22$	$LR^2/451.72$	7e–01	$LR^2/790.22$	1e–07

Table 2 Gradient method with relative step size $h = 1.5$: numerical values from Conjecture 1 and relative error for the upper and lower limits of the guaranteed interval obtained numerically with VSDP [11] and SeDuMi [25]

N	1	2	5	10	15	20	30
Relative error (upper limit)	2e–09	7e–10	2e–09	1e–09	9e–10	1e–09	9e–10
Conjecture	1.2e–01	7.1e–02	3.1e–02	1.6e–02	1.1e–02	8.2e–03	5.5e–03
Relative error (lower limit)	2e–09	3e–09	9e–09	9e–08	2e–07	3e–07	9e–07

In that case, the objective value of the approximate primal (resp. dual) solution is no longer guaranteed to be a lower (resp. upper) bound on the exact optimal value, hence potentially negating the advantage of an exact convex formulation. For this reason, all numerical results reported in this section have been double checked with an interval arithmetic-based semidefinite optimization solver [11] that returns an interval that is guaranteed to contain the optimal value. These guaranteed bounds are reported in Table 2 for the case $h = 1.5$, which compares them with Conjecture 1.

We can observe that the use of a verified solver does not impact our conclusions about the validity of the conjecture. Moreover, this table is typical of what we observed for all conjectures in this section: all numerical results reported were validated[2], and in what follows we will no longer mention this verification explicitly.

Finally, we compare results obtained with Conjecture 1 with classical analytical bounds from the literature for the GM with unit normalized step size $h = 1$ (which is usually recommended, and sometimes called optimal). The best analytical bound we could find, e.g. in [7], states that

---

[2] Except for tests where validation encountered numerical difficulties, i.e for which VSDP returned no valid interval, which occurred more and more frequently as the value of the worst-case bound became closer to zero.

$$f(x_N) - f_* \leq \frac{LR^2}{2} \frac{1}{N+1}. \tag{6}$$

This analytical bound is asymptotically worse by a factor of 2 than the bound predicted by Conjecture 1 with $h = 1$. Similarly, one can investigate the effect of choosing the optimal normalized step size $h_{\mathrm{opt}}(N)$ instead of $h = 1$: Conjecture 1 then predicts another improvement by a factor of 2. These observations follow from the asymptotic (large N) behaviors of the different worst-case bounds on $f(x_N) - f_*$, which can easily be computed:

$$\text{Conjecture 1 with } h = 1 \longrightarrow \frac{LR^2}{2} \frac{1}{2N+1},$$

$$\text{Conjecture 1 with } h = h_{\mathrm{opt}}(N) \xrightarrow[N\to\infty]{} \frac{LR^2}{2} \frac{1}{4N+1}.$$

4.1.2 A generalized conjecture for strongly convex functions

In view of the encouraging results obtained for the GM in the smooth case, we now study the behavior of the GM on the class of strongly convex functions $\mathcal{F}_{\mu,L}(\mathbb{R}^d)$ using our formulation (sdp-PEP) with the same performance criterion, objective function accuracy. It turns out that the solution for every problem consisted again in a one-dimensional worst-case function (rank $G = 1$) of the same piecewise quadratic type. We therefore introduce the following general definitions for functions $f_{1,\tau}$ and f_2:

$$f_{1,\tau}(x) = \begin{cases} \frac{\mu}{2}x^2 + a_\tau |x| + b_\tau & \text{if } |x| \geq \tau \\ \frac{L}{2}x^2 & \text{else,} \end{cases}$$

$$f_2(x) = \frac{L}{2}x^2,$$

where scalars $a_\tau = (L - \mu)\tau$ and $b_\tau = -\left(\frac{L-\mu}{2}\right)\tau^2$ are chosen to ensure continuity of $f_{1,\tau}$ and its first derivative, and τ is a parameter that controls the radius of the central quadratic piece (with the largest curvature). Although the value of parameter τ could in principle be estimated from the numerical solutions of our problems, it turns out it can be computed analytically by maximizing the final objective value $f_{1,\tau}(x_N)$ (assuming that all iterates stay in the affine zone $|x| \geq \tau$), which then leads to

$$\tau = \frac{R\kappa}{(\kappa - 1) + (1 - \kappa h)^{-2N}} \tag{7}$$

where $\kappa = \frac{\mu}{L}$ is the inverse condition number of the problem class $f \in \mathcal{F}_{\mu,L}(\mathbb{R}^d)$. We are now able to extend Conjecture 1 to the GM applied to strongly convex functions.

Table 3 Maximum relative estimated differences between Conjecture 2 and corresponding numerical results obtained with SeDuMi [25]. The maximum is taken over all $N \in \{1, \dots, 30\}$ and $h \in \{0.05, \dots, 1.95\}$ for which the conjecture predicts a worst-case larger than 10^{-6}

$\kappa =$	0	.001	.005	.010	.015	0.1	0.2	0.5
Rel. error	6e–10	7e–10	4e–10	6e–10	8e–10	2e–07	9e–08	1e–06

Conjecture 2 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

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

As in the previous section, this conjecture states that the worst-case behavior of the GM according to objective function accuracy is achieved by function $f_{1,\tau}$ or f_2, depending on which of the two is worse. Proceeding now to its numerical validation, we first point out that our results are intrinsically limited to the accuracy that can be reached by numerical SDP solvers. For this reason, we only report on situations for which Conjecture 2 predicts a final accuracy larger than 10^{-6}, ensuring a few significant digits for the numerical results. The resulting estimated relative differences between Conjecture 2 and the numerical results obtained with (sdp-PEP) are given in Table 3, for different values of κ. We observe that the conjecture is very well supported by our numerical results, with a largest relative error around 10^{-6}, reached for the largest value of κ considered here. This is expected as GM tends to perform better as κ increases (i.e., final accuracy $f(x_N) - f_*$ approaches zero), which renders a precise comparison between numerical results and the conjecture more and more difficult.

We now investigate some consequences of our conjecture. First, we note that Conjecture 2 tends to Conjecture 1 as μ tends to zero. This is a consequence of the fact that τ tends to $\frac{R}{2Nh+1}$ as κ tends to zero (one can also check that function $f_{1,\tau}$ tends to function f_1 introduced earlier). Hence our formulation (sdp-PEP) closes an apparent gap between worst-case analyses of the smooth convex and the smooth strongly convex cases. Indeed, to the best of our knowledge, existing worst-case bounds for the smooth strongly convex case do not converge to the smooth case as $\mu \to 0$.

It is also interesting to compare our results to those obtained with the IQC methodology of [16]. If we only care about asymptotic linear rates of convergence, Conjecture 2 predicts

$$f(x_N) - f_* \leq \frac{LR^2}{2} \max\{\kappa \, \rho_1^{2N}, \rho_2^{2N}\} \quad \text{with } \rho_1 = |1 - \kappa h| \text{ and } \rho_2 = |1 - h|$$

(the first term in the max was obtained by neglecting $(\kappa - 1)$ in the denominator). On the other hand [16, Section 4.4] proves that the distance to the solution converges linearly according to

$$\|x_N - x_*\| \leq \rho^N \|x_0 - x_*\| \quad \text{with a factor } \rho = \max\{\rho_1, \rho_2\}$$

with the same values for ρ_1 and ρ_2. This matches our asymptotic rate up to a multiplicative constant.

As for Conjecture 1, our new Conjecture 2 suggests optimal step sizes $h_{\mathrm{opt}}(N, \kappa)$, which can be obtained by solving the equation (for $0 < \kappa < 1$)

$$\frac{\kappa}{(\kappa - 1) + (1 - \kappa h_{\mathrm{opt}})^{-2N}} = (1 - h_{\mathrm{opt}})^{2N} \tag{8}$$

(note that one recovers the previous equation for h_{opt} when μ tends to zero). For a given N, as κ increases from 0 to 1, those optimal step sizes decrease from $h_{\mathrm{opt}}(N, 0)$ (optimal step size in the smooth case) to $h_{\mathrm{opt}}(N, 1) = 1$ (the latter being expected since it can only correspond to the case of function f_2 in the original (PEP), for which the GM with $h = 1$ converges in one iteration). For a given κ, we find that $h_{\mathrm{opt}}(N, \kappa)$ increases as N increases, as in the smooth convex case, according to the following lower and upper estimates

$$1 + \left(\frac{\kappa - 1}{\kappa} + \frac{1}{\kappa} \left(\frac{1 + \kappa}{1 - \kappa} \right)^{2N} \right)^{-\frac{1}{2N}} \leq h_{\mathrm{opt}}(N, \kappa)$$
$$\leq \min \left\{ 1 + \left(\frac{(\kappa - 1)}{\kappa} + \frac{1}{\kappa} (1 - \kappa)^{-2N} \right)^{-\frac{1}{2N}}, \frac{2}{1 + \kappa} \right\} \tag{9}$$

which both tend to $\frac{2}{1+\kappa}$ as N increases (the first term appearing in the min of the upper bound tends to $2 - \kappa$, which is always greater than $\frac{2}{1+\kappa}$). This limiting normalized step size $\frac{2}{1+\kappa}$ corresponds to step size $\frac{2}{L+\mu}$ that is often recommended for the GM, and sometimes called optimal.

We now illustrate the improvements provided by Conjecture 2 with respect to the classical analytical worst-case bound found in the literature. When using normalized step size $h = \frac{2}{1+\kappa}$, iterates from GM applied to functions in $\mathcal{F}_{\mu,L}(\mathbb{R}^d)$ are known to satisfy (see [21] for example)

$$f(x_N) - f_* \leq \frac{LR^2}{2} \left(\frac{1 - \kappa}{1 + \kappa} \right)^{2N}. \tag{10}$$

On the other hand, as the number of steps N tends to infinity, the true worst-case predicted by Conjecture 2 for the same step size asymptotically tends to $\frac{LR^2}{2} \left(\frac{1-\kappa}{1+\kappa} \right)^{2N}$, which is exactly the same as (10). Indeed, one can check that this rate is equal to the second term appearing in the max of Conjecture 2, while the first term tends to $\frac{LR^2}{2} \kappa \left(\frac{1-\kappa}{1+\kappa} \right)^{2N}$ which is always smaller.

One can however do better using the optimal step size h_{opt}. Since it is not closed-form, we use the following approximate expression obtained after solving a suitable approximation of Eq. (8)

$$\tilde{h}_{\mathrm{opt}}(N) = \frac{1 + \kappa^{\frac{1}{2N}}}{1 + \kappa^{1 + \frac{1}{2N}}}$$

(note that $\tilde{h}_{\mathrm{opt}}(N)$ tends to $\frac{2}{1+\kappa}$ as N grows), and find that Conjecture 2 predicts a worst-case bound tending to

$$\frac{LR^2}{2} \kappa^{\frac{1}{1+\kappa}} \left(\frac{1-\kappa}{1+\kappa}\right)^{2N}$$

which improves the asymptotic rate by a factor $\left(\frac{1}{\kappa}\right)^{\frac{1}{1+\kappa}}$ (which can be shown to lie between $\frac{3}{4}\frac{1}{\kappa}$ and $\frac{1}{\kappa}$).

4.1.3 A conjecture on the gradient norm

We now consider a different performance criterion, given by the norm of the gradient computed at the last iterate. Numerical experiments with our formulation suggest that results similar to those presented in the previous sections can be obtained both in the smooth convex and smooth strongly convex cases, based again on one-dimensional piecewise quadratic worst-case functions. Using the same definition for functions $f_{1,\tau}$ and f_2, and choosing now the parameter τ according to

$$\tau = \frac{R\kappa}{(\kappa - 1) + (1 - \kappa h)^{-N}}, \tag{11}$$

we propose the following conjecture.

Conjecture 3 Any sequence of iterates $\{x_i\}$ generated by the gradient method GM with constant normalized step sizes $0 \le h \le 2$ on a smooth strongly convex function $f \in \mathcal{F}_{\mu,L}(\mathbb{R}^d)$ satisfies

$$\|\nabla f(x_N)\|_2 \le LR \max\left(\frac{\kappa}{(\kappa - 1) + (1 - \kappa h)^{-N}}, |1 - h|^N\right).$$

As for Conjecture 2, we limit our numerical validation to the cases where the worst-case values predicted by the Conjecture are larger than 10^{-6}; the largest relative error is about 10^{-7}.

We note that, as κ tends to zero (i.e., the smooth case), Conjecture 3 tends to

$$\|\nabla f(x_N)\|_2 \le LR \max\left(\frac{1}{Nh + 1}, |1 - h|^N\right).$$

From that, we see that the optimal step size $h_{\text{opt}}^{\nabla}(N, 0)$ for the GM is again an increasing function of N with $\sqrt{2} \le h_{\text{opt}}^{\nabla}(N, 0) < 2$ and $h_{\text{opt}}^{\nabla}(N, 0) \to 2$ as $N \to \infty$. In the strongly convex case $\kappa > 0$, the optimal step size is a decreasing function of κ and satisfies $h_{\text{opt}}^{\nabla}(N, \kappa) \to 1$ as $\kappa \to 1$. As in the previous case, $h_{\text{opt}}^{\nabla}(N, \kappa)$ is bounded above by $\frac{2}{1+\kappa}$, which we can confirm with the following lower and upper bounds on h_{opt}^{∇}:

$$1 + \left(\frac{\kappa - 1}{\kappa} + \left(\frac{1+\kappa}{1-\kappa}\right)^N\right)^{-1/N} \le h_{\text{opt}}^{\nabla}(N, \kappa)$$

$$\le \min\left\{1 + \left(\frac{\kappa - 1}{\kappa} + \frac{1}{\kappa}(1 - \kappa)^{-N}\right)^{-1/N}, \frac{2}{1+\kappa}\right\}.$$

In the smooth case, those bounds reduce to the simpler expression

$$2 - \frac{\log 2N}{N} \sim 1 + (1 + 2N)^{-1/N} \le h_{\text{opt}}^{\nabla} \le 1 + (1 + N)^{-1/N} \sim 2 - \frac{\log N}{N}.$$

We now compare with a standard analytical worst-case bound. The iterates of the GM method with normalized step size $\frac{2}{1+\kappa}$ are known to satisfy

$$\|x_N - x_*\|_2 \le R \left(\frac{1-\kappa}{1+\kappa} \right)^N \quad \text{and} \quad \|\nabla f(x_N)\|_2 \le L\|x_N - x_*\|_2 \le LR \left(\frac{1-\kappa}{1+\kappa} \right)^N \tag{12}$$

(see for example [21] for the left inequality, and use the L-Lipschitz property of the gradient along with $\nabla f(x_*) = 0$ to derive the right inequality). The latter estimate is tight according to Conjecture 3. Using the following approximate optimal step size

$$\tilde{h}_{\text{opt}}^{\nabla}(N) = \frac{1 + \kappa^{\frac{1}{N}}}{1 + \kappa^{1+\frac{1}{N}}}$$

(which tends to $\frac{2}{1+\kappa}$ as N grows) can be shown to improve the conjectured asymptotic rate by the same factor $\kappa^{-\frac{1}{1+\kappa}}$ as in the previous section.

4.2 Fast gradient method and optimized gradient method

In this section we assess the performance in the smooth convex case ($\mu = 0$) of two accelerated first-order methods: the so-called fast gradient method (FGM) due to Nesterov [20], and an optimized gradient method (OGM) recently proposed by Kim and Fessler [14].

Fast Gradient Method (FGM)

Input: $f \in \mathcal{F}_{0,L}(\mathbb{R}^d)$, $x_0 \in \mathbb{R}^d$, $y_0 = x_0, \theta_0 = 1$.

For $i = 0 : N - 1$

$$y_{i+1} = x_i - \frac{1}{L}\nabla f(x_i)$$

$$\theta_{i+1} = \frac{1 + \sqrt{4\theta_i^2 + 1}}{2}$$

$$x_{i+1} = y_{i+1} + \frac{\theta_i - 1}{\theta_{i+1}}(y_{i+1} - y_i)$$

> **Optimized Gradient Method (OGM)**
>
> Input: $f \in \mathcal{F}_{0,L}(\mathbb{R}^d)$, $x_0 \in \mathbb{R}^d$, $y_0 = x_0$, $\theta_0 = 1$.
> For $i = 0 : N - 1$
>
> $$y_{i+1} = x_i - \frac{1}{L}\nabla f(x_i)$$
>
> $$\theta_{i+1} = \begin{cases} \frac{1+\sqrt{4\theta_i^2+1}}{2}, & i \leq N-2 \\ \frac{1+\sqrt{8\theta_i^2+1}}{2}, & i = N-1 \end{cases}$$
>
> $$x_{i+1} = y_{i+1} + \frac{\theta_i - 1}{\theta_{i+1}}(y_{i+1} - y_i) + \frac{\theta_i}{\theta_{i+1}}(y_{i+1} - x_i)$$

Both of these algorithms are defined in terms of two sequences: $\{y_i\}_i$ is a primary sequence, and $\{x_i\}_i$ is a secondary sequence, where the gradient is evaluated. We first show that both of these algorithms can be expressed as fixed-step first-order methods, which we defined as

$$x_i = x_0 - \sum_{k=0}^{i-1} h_{i,k}\nabla f(x_k) \quad \text{(for } L = 1\text{)}.$$

One way to proceed is to focus on the secondary sequence $\{x_i\}_i$ and substitute the y_i's in the algorithm formulation. For FGM, we have

$$x_{i+1} = x_i - \frac{g_i}{L} + \frac{\theta_i - 1}{\theta_{i+1}}\left(x_i - x_{i-1} - \frac{g_i}{L} + \frac{g_{i-1}}{L}\right),$$

$$= x_i + \frac{\theta_i - 1}{\theta_{i+1}}(x_i - x_{i-1}) - \left(\frac{\theta_i - 1}{\theta_{i+1}} + 1\right)\frac{g_i}{L} + \frac{\theta_i - 1}{\theta_{i+1}}\frac{g_{i-1}}{L},$$

which allows to obtain the step sizes relative to x_0 by recurrence:

$$h_{i+1,k} = \begin{cases} h_{i,k} + \frac{\theta_i-1}{\theta_{i+1}}\left(h_{i,k} - h_{i-1,k}\right) & \text{if } k \leq i-2, \\ h_{i,k} + \frac{\theta_i-1}{\theta_{i+1}}(h_{i,k} - 1) & \text{if } k = i-1, \\ \frac{\theta_i-1}{\theta_{i+1}} + 1 & \text{if } k = i, \end{cases}$$

with initial conditions $h_{1,0} = 1$, $h_{1,k} = 0$ if $k < 0$ and $h_{0,k} = 0$ for all k. Similarly, we have for OGM

$$h_{i+1,k} = \begin{cases} h_{i,k} + \frac{\theta_i-1}{\theta_{i+1}}\left(h_{i,k} - h_{i-1,k}\right) & \text{if } k \leq i-2, \\ h_{i,k} + \frac{\theta_i-1}{\theta_{i+1}}(h_{i,k} - 1) & \text{if } k = i-1, \\ \frac{2\theta_i-1}{\theta_{i+1}} + 1 & \text{if } k = i, \end{cases}$$

with the same initial conditions. This approach will provide estimates for the last secondary iterate x_N. If an estimate for last primary iterate y_N is needed, one just has to replace the expression of x_N by y_N, which is done by using the following alternative coefficients for the last step:

$$h_{N,k} = \begin{cases} h_{N-1,k} & \text{if } k \leq N-2, \\ 1 & \text{if } k = N-1, \end{cases}$$

for both FGM and OGM.

Again, our numerical experiments strongly suggest the same assumption about the shape of the worst-case functions, i.e., one-dimensional and piecewise quadratic (with iterates staying in the affine zone of $f_{1,\tau}$). Using this property, we are able to compute the following values of τ achieving the worst-case final objective accuracy, which surprisingly hold for both the classical FGM and the more recent OGM (a coincidence for which we can offer no explanation)

$$\tau_1 = \frac{R}{2 \sum_{k=0}^{N-2} h_{N-1,k} + 3} \quad \text{for the primary sequence,}$$

$$\tau_2 = \frac{R}{2 \sum_{k=0}^{N-1} h_{N,k} + 1} \quad \text{for the secondary sequence.}$$

Our numerical results suggest the following two conjectures (validations for both conjectures were performed for values of $N \in \{1, \ldots, 100\}$ and displayed a relative error less than 10^{-4}).

Conjecture 4 Any (primary) sequence of iterates $\{y_i\}$ generated by the fast gradient method FGM (resp. optimized gradient method OGM) on a smooth convex function $f \in \mathcal{F}_{0,L}(\mathbb{R}^d)$ satisfies

$$f(y_N) - f_* \leq f_{1,\tau_1}(y_{1,N}) = \frac{LR^2}{2} \frac{1}{2 \sum_{k=0}^{N-2} h_{N-1,k} + 3},$$

where $y_{1,N}$ is the final (primary) iterate computed by FGM (resp. OGM) applied to f_{1,τ_1} starting from $x_0 = R$, and quantities $h_{N-1,k}$ are the fixed coefficients of the last step of FGM (resp. OGM).

Conjecture 5 Any (secondary) sequence of iterates $\{x_i\}$ generated by the fast gradient method FGM (resp. optimized gradient method OGM) on a smooth convex function $f \in \mathcal{F}_{0,L}(\mathbb{R}^d)$ satisfies

$$f(x_N) - f_* \leq f_{1,\tau_2}(x_{1,N}) = \frac{LR^2}{2} \frac{1}{2 \sum_{k=0}^{N-1} h_{N,k} + 1},$$

where $x_{1,N}$ is the final (secondary) iterate computed by FGM (resp. OGM) applied to f_{1,τ_2} starting from $x_0 = R$, and quantities $h_{N,k}$ are the fixed coefficients of the last step of FGM (resp. OGM).

The worst-case bounds in these two conjectures involve the normalized step sizes of the FGM and OGM methods. It turns out these can be computed in closed form for OGM (see also [13]), and give ($N \geq 1$)

$$f(y_N) - f_* \leq \frac{LR^2}{4\theta_{N-1}^2 + 2} \leq \frac{LR^2}{2} \frac{2}{(N+1)^2 + 2} \quad \text{and}$$

$$f(x_N) - f_* \leq \frac{LR^2}{2\theta_N^2} \leq \frac{LR^2}{2} \frac{2}{(N+1)(N+1+\sqrt{2})}$$

(where the inequalities rely on the bounds $\theta_{N-1}^2 \geq \frac{(N+1)^2}{4}$ and $\theta_N^2 \geq \frac{(N+1)(N+1+\sqrt{2})}{2}$). We were not able to obtain similar closed-form bounds for the FGM.

We now compare the numerical values obtained with Conjectures 4 and 5 with analytical bounds known for the FGM. We use for the primary sequence

$$f(y_N) - f_* \leq \frac{2LR^2}{(N+1)^2}, \tag{13}$$

which can be found in [4], and for the secondary sequence

$$f(x_N) - f_* \leq \frac{2LR^2}{(N+2)^2} \tag{14}$$

which was very recently derived in [14]. The comparison is displayed on Fig. 4. The asymptotic behaviors of both sequences are well captured by the analytical bounds (13)
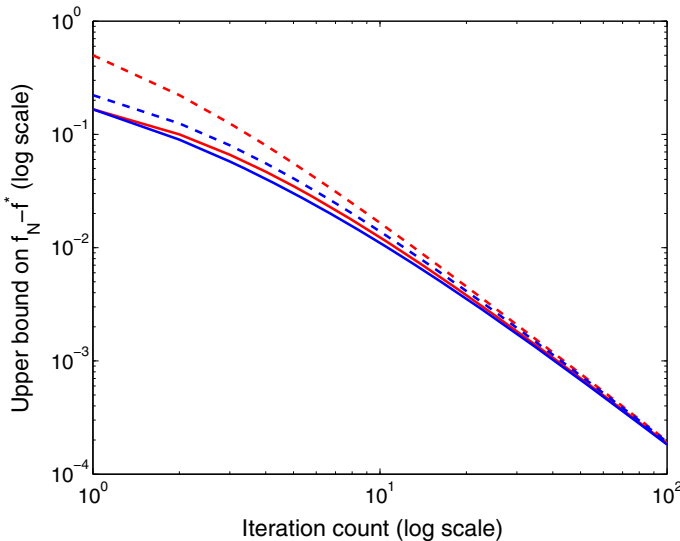

Fig. 4 Comparison of the worst-case performance of the FGM: analytical bound (13) (*dashed red*) versus Conjecture 4 (*red*) and analytical bound (14) (*dashed blue*) versus Conjecture 5 (*blue*) (color figure online)

and (14), but we observe that the estimation of the transient worst cases are improved by our conjectures: a factor approximately equal to 1.15 is gained for both sequences after 30 iterations.

Before going into the next section, we comment on the applicability of our results to monotone variants of first-order methods, i.e. methods which guarantee $f(y_{i+1}) \leq f(y_i)$. Consider for example FISTA [4], which is equivalent to FGM when applied to smooth unconstrained minimization. MFISTA [3], a monotone variant of FISTA, happens to generate a monotonically decreasing sequence $\{f(y_i)\}_i$ when applied to our worst-case function f_{1,τ_1} from $x_0 = R$. This means that the corresponding lower bound from Conjecture 4 also applies to MFISTA.

4.3 Estimation of the smallest gradient norm among all iterates

First-order methods are often used in dual approaches where, in addition to objective function accuracy, gradient norm plays an important role. Indeed, this quantity controls primal feasibility of the iterates (see e.g., [9]). Considering for example the accelerated FGM in the smooth case, we know from the previous section that the classical analytical bound on the worst-case accuracy for a function in $\mathcal{F}_{0,L}(\mathbb{R}^d)$ is given by $\frac{2LR^2}{(N+1)^2}$. From that bound, it is easy to obtain a similar bound on the last gradient norm, using Corollary 1:

$$\|\nabla f(y_N)\|_2 \leq \sqrt{2L(f(y_N) - f_*)} \leq \frac{2LR}{N+1}. \tag{15}$$

Observe that this asymptotic rate is significantly worse than that of the objective function accuracy, and not better than that of the gradient method GM (see Conjecture 3).

However, it is well-known that the norm of the gradient is not decreasing monotonically among iterates of the FGM. Hence, in this section, we will estimate the worst-case performance of FGM according to the smallest observed gradient norm among all iterates:

$$\min_{i \in \{0,...,N\}} \|\nabla f(y_i)\|_2.$$

In order to do so, only a slightly modified version of (sdp-PEP) is needed: this min-type objective function is representable using a new variable t for the objective and $N + 1$ additional linear inequalities $t \leq \|\nabla f(y_i)\|_2^2 \Leftrightarrow t \leq G_{i,i}$ for all $0 \leq i \leq N$. Note that the maximum is still attained since this concave piecewise linear objective function is continuous.

This criterion was suggested in [22], which proposes a variant of FGM that consists in performing $N/2$ steps of the standard FGM followed by $N/2$ steps of the GM with $h = 1$. It is then theoretically established that this variant of FGM, which we denote by MFGM, satisfies

$$\min_{i \in \{0,...,N\}} \|\nabla f(y_i)\|_2 \leq \frac{8LR}{N^{3/2}}, \tag{16}$$

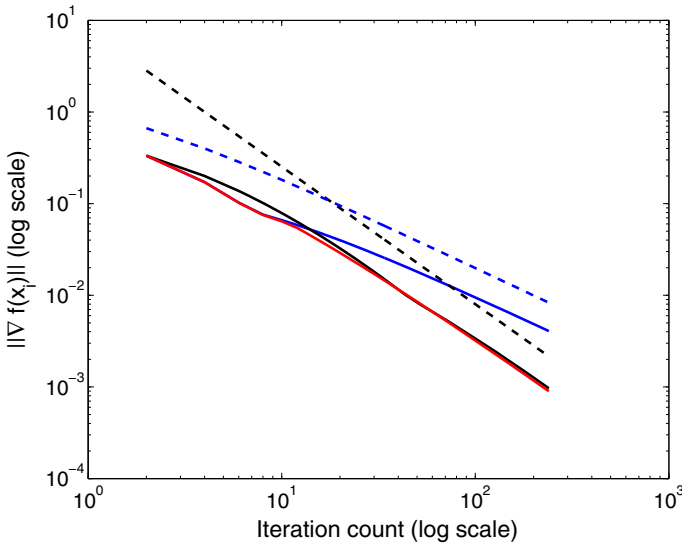an improvement compared to the rate of convergence of the gradient of the last iterate.

Fig. 5 Comparison of gradient norm convergence rates for the FGM and the MFGM from [22]. Theoretical guarantees are *dashed*. Analytical bound on FGM (15) in its last iterate (*dashed blue*); numerical worst-case for FGM at its last iterate (*blue*); numerical worst-case for FGM at its best iterate (*red*); analytical bound on MFGM (16) for the best iterate (*dashed black*); numerical worst-case for MFGM at its best iterate (*black*) (color figure online)

Table 4 FGM and MFGM: comparison between theoretical bounds and numerical results for the criteria $\|\nabla f(x_N)\|_2$(last) and $\min_i \|\nabla f(x_i)\|_2$ (best) Results obtained with [18]

N	FGM, analytic (15)	FGM, last	FGM, best	MFGM, analytic (16)	MFGM, best
2	$LR/1.50$	$LR/3.00$	$LR/3.00$	$LR/0.35$	$LR/3.00$
4	$LR/2.50$	$LR/5.84$	$LR/5.84$	$LR/1.00$	$LR/5.00$
10	$LR/5.50$	$LR/15.14$	$LR/15.62$	$LR/3.95$	$LR/12.66$
20	$LR/10.50$	$LR/25.08$	$LR/34.49$	$LR/11.18$	$LR/30.77$
30	$LR/15.50$	$LR/35.13$	$LR/58.50$	$LR/20.54$	$LR/55.38$
40	$LR/20.50$	$LR/45.19$	$LR/86.17$	$LR/31.62$	$LR/86.41$
50	$LR/25.50$	$LR/55.25$	$LR/117.08$	$LR/44.19$	$LR/119.63$
100	$LR/50.50$	$LR/105.49$	$LR/311.34$	$LR/125.00$	$LR/296.58$
200	$LR/100.50$	$LR/205.77$	$LR/850.59$	$LR/353.55$	$LR/791.87$

We now compare FGM with this modified variant MFGM using our performance estimation formulation. Fig. 5 compares the behaviors of those methods in both their last (for FGM) and best iterates, as well as the above analytic bounds (15) and (16). This experiment confirms that the gradient norm of the last iterate of FGM decreases according to the slower $\mathcal{O}(N^{-1})$ rate of (15). We also observe that both the MFGM and the original FGM achieve the same $\mathcal{O}(N^{-3/2})$ convergence rate for the smallest gradient norm, which was not known before for FGM. In addition, numerical results reported in Table 4 suggest that FGM performs slightly better than MFGM.

A regularization technique is also described in [22], featuring a $\mathcal{O}(N^{-2})$ convergence rate up to a logarithmic factor. A drawback of this approach is that it requires a bound on the distance to the optimal solution, and that the coefficients of the method explicitly depend on this bound. No fixed-step method achieving the same $\mathcal{O}(N^{-2})$ seems to be known.

5 Conclusion

The contribution of this paper is threefold: first, we present necessary and sufficient conditions for smooth strongly convex interpolation. Those conditions are derived by showing an explicit way of constructing the interpolating functions. Second, we show that the exact worst-case performance of any fixed-step first-order algorithm for smooth strongly convex unconstrained optimization can be formulated as a convex problem. In this context, our interpolation procedure also provides explicit functions achieving the worst-case bounds computed by our approach. Third, we test of our formulation numerically on a variety of functions classes, first-order methods and performance criteria, establishing on the way a series of conjectures on the corresponding worst-case behaviors. In particular, we suggest new tight estimates of the optimal step size for the fixed-step gradient method with constant step size, which depend on the number of iterations and the condition number.

Our performance estimation problem provide a generic tool to analyze fixed-step first-order methods. It allows computing both exact worst-case guarantees and functions reaching them, and provides a unified algorithmic analysis for smooth convex functions and smooth strongly convex functions.

The exact worst-case values provided by our approach require solving a convex semidefinite program whose size grows as the square of the number of iterations considered, which may become prohibitive when this number of iterations is large. This can be avoided using iteration-independent bounds, as proposed in [16], but at the cost of obtaining poorer worst-case guarantees.

Further improvements to our approach include an extension of (PEP) to more general methods, such as first-order methods equipped with line search, or first-order methods designed to work on a restricted convex feasible region (projected gradient). Another desirable feature would be the ability to optimize the step sizes of the method considered in (sdp-PEP), as was proposed in [10,14] for the relaxed version of (PEP).

Software Our semidefinite programming approach to performance estimation has been implemented with MATLAB code, which can be downloaded from http://perso.uclouvain.be/adrien.taylor. This routine features an easy-to-use interface, which allows the estimation of the worst-case performance of a given fixed-step first-order algorithm (to be chosen among a pre-defined list or to be specified by its coefficients) on a given class of functions, for a given performance criterion, after any number of steps.

References

1. Bauschke, H., Combettes, P.: Convex analysis and monotone operator theory in Hilbert spaces. Springer, Berlin (2011)

2. Beck, A.: Quadratic matrix programming. SIAM J. Optim. **17**(4), 1224–1238 (2007)
3. Beck, A., Teboulle, M.: Fast gradient-based algorithms for constrained total variation image denoising and deblurring problems. Image Process. IEEE Trans. **18**(11), 2419–2434 (2009)
4. Beck, A., Teboulle, M.: A fast iterative shrinkage-thresholding algorithm for linear inverse problems. SIAM J. Imag. Sci. **2**(1), 183–202 (2009)
5. Ben-Tal, A., Nemirovski, A.: Lectures on modern convex optimization: analysis, algorithms, and engineering applications, vol. 2. SIAM, Philadelphia (2001)
6. Bertsekas, D.P.: Convex optimization theory. Athena Scientific, Belmont (2009)
7. Bertsekas, D.P.: Convex optimization algorithms. Athena Scientific, Belmont (2015)
8. Boyd, S., Vandenberghe, L.: Convex optimization. Cambridge University Press, Cambridge (2004)
9. Devolder, O., Glineur, F., Nesterov, Y.: Double smoothing technique for large-scale linearly constrained convex optimization. SIAM J. Optim. **22**(2), 702–727 (2012)
10. Drori, Y., Teboulle, M.: Performance of first-order methods for smooth convex minimization: a novel approach. Math. Program. **145**(1–2), 451–482 (2014)
11. Härter, V., Jansson, C., Lange, M.: VSDP: A matlab toolbox for verified semidefinite-quadratic-linear programming. Technical report, Institute for Reliable Computing, Hamburg University of Technology (2012)
12. Hiriart-Urruty, J.B., Lemaréchal, C.: Convex analysis and minimization algorithms. Springer, Heidelberg (1996). (Two volumes—2nd printing)
13. Kim, D., Fessler, J.: On the convergence analysis of the optimized gradient methods. arXiv preprint arXiv:1510.08573 (2015)
14. Kim, D., Fessler, J.: Optimized first-order methods for smooth convex minimization. Math. Program. (2015). doi:10.1007/s10107-015-0949-3
15. Lambert, D., Crouzeix, J.P., Nguyen, V., Strodiot, J.J.: Finite convex integration. J. Convex Anal. **11**(1), 131–146 (2004)
16. Lessard, L., Recht, B., Packard, A.: Analysis and design of optimization algorithms via integral quadratic constraints. SIAM. J. Optim. **26**(1), 57–95 (2016)
17. Löfberg, J.: YALMIP: a toolbox for modeling and optimization in MATLAB. In: Proceedings of the CACSD Conference (2004)
18. Mosek, A.: The MOSEK optimization software, vol. 54. http://www.mosek.com (2010)
19. Nemirovsky, A., Yudin, D.: Problem complexity and method efficiency in optimization. Wiley-Interscience, New York (1983)
20. Nesterov, Y.: A method of solving a convex programming problem with convergence rate O($1/k^2$)). Sov. Math. Dokl. **27**, 372–376 (1983)
21. Nesterov, Y.: Introductory lectures on convex optimization: a basic course. Applied optimization. Kluwer Academic Publishers, Dordrecht (2004)
22. Nesterov, Y.: How to make the gradients small. Optima **88**, 10–11 (2012)
23. Rockafellar, R.T.: Convex Analysis. Princeton University Press, Princeton (1996)
24. Rockafellar, R.T., Wets, R.B.: Variational analysis. Springer, Berlin (1998)
25. Sturm, J.F.: Using SeDuMi 1.02, a MATLAB toolbox for optimization over symmetric cones. Optim. Methods Softw. **11–12**, 625–653 (1999)
26. Vandenberghe, L., Boyd, S.: Semidefinite programming. SIAM Rev. **38**, 49–95 (1994)