Optimizing the Efficiency of First-Order Methods for Decreasing the Gradient of Smooth Convex Functions

Donghwan Kim[1] · **Jeffrey A. Fessler**[2]

Date of current version: October 28, 2020

Abstract This paper optimizes the step coefficients of first-order methods for smooth convex minimization in terms of the worst-case convergence bound (*i.e.*, efficiency) of the decrease in the gradient norm. This work is based on the performance estimation problem approach. The worst-case gradient bound of the resulting method is optimal up to a constant for large-dimensional smooth convex minimization problems, under the initial bounded condition on the cost function value. This paper then illustrates that the proposed method has a computationally efficient form that is similar to the optimized gradient method.

Keywords First-order methods · Gradient methods · Smooth convex minimization · Worst-case performance analysis

Mathematics Subject Classification (2010) 90C25 · 90C30 · 90C60 · 68Q25 · 49M25 · 90C22

1 Introduction

Large-dimensional optimization problems arise in various modern applications of signal processing, machine learning, control, communication, and many other areas. First-order methods are widely used for solving such large-scale problems as their iterations involve only function/gradient calculations and simple vector operations. However, they can require many iterations to achieve the given accuracy level. Therefore, developing efficient first-order methods has received great interest, which is the main motivation of this paper. In particular, this paper targets the decrease in the gradient for smooth convex minimization, under the initial bounded condition on the cost function value. This paper uses the performance estimation problem (PEP) in [1] and constructs a new method called OGM-G.

Among first-order methods for smooth convex minimization, Nesterov's fast gradient method (FGM) [2, 3] has been used widely because its worst-case *cost function* inaccuracy bound (*i.e.*, the cost function efficiency) is optimal up to a constant, under the initial bounded *distance* condition [3, 4]. Recently, the optimized gradient method (OGM) [5] (that was numerically first identified in [1] using PEP) has been found to exactly achieve the optimal worst-case rate of decreasing the smooth convex cost functions [6], leaving no room for improvement in the worst-case. On the other hand, first-order methods that decrease the *gradient* at an optimal rate in [4] are yet unknown, even up to a constant. The proposed OGM-G method has such an optimal rate under the initial bounded *function* condition. After the initial version of this paper was posted online [7], a simple method using OGM-G was constructed in [8] that also has an optimal rate under the initial bounded *distance* condition.

Gradient rate analysis is useful both in theory (*e.g.*, for a dual approach [9] and a matrix scaling problem [10]) and in practice (*e.g.*, can be used as a stopping criterion). In addition, unlike smooth convex minimization, a worst-case *gradient* inaccuracy and an initial bounded *function* condition are standard choices for analyzing gradient methods for smooth *nonconvex* minimization [11]. Therefore, this work can provide a step towards better understanding the convergence behavior of gradient methods for nonconvex minimization.

[1] Department of Mathematical Sciences, KAIST, Republic of Korea
[2] Department of Electrical Engineering and Computer Science, University of Michigan, USA
E-mail: donghwankim@kaist.ac.kr, fessler@umich.edu

There is recent interest in developing accelerated methods for decreasing the gradient (in convex minimization) [9, 10, 12–14]. The best known worst-case gradient rate is achieved by FGM with a regularization technique in [9] that is optimal up to a logarithmic factor. However, a practical limitation of that method is that it requires knowledge of a bound on a value such as the distance between the initial and optimal points. In [14] we used PEP to derive efficient first-order methods that do not need knowledge of such unavailable values. However, the methods in [14] are far from achieving the optimal rate (not even up to a logarithmic factor), due to strict relaxations introduced to PEP in [14]. The methods in [9, 13–16] also achieve a similar nonoptimal rate. Thus, there is still room to improve the worst-case gradient convergence bound of the first-order methods for smooth convex minimization.

This paper optimizes the step coefficients of first-order methods in terms of the worst-case gradient decrease using PEP [1, 17], yielding OGM-G. The new analysis avoids the (unnecessary) strict relaxations on PEP in [14]. This paper then shows that OGM-G has an equivalent efficient form that is similar to OGM, and thus has an inexpensive per-iteration computational complexity. OGM-G attains the optimal bound of the worst-case gradient norm up to a constant under the initial bounded *function* condition [4]. On the way, this paper also provides a new exact worst-case gradient bound for the gradient method (GM).

The initial bounded condition on the distance between initial and optimal points is a standard assumption, whereas the initial condition on the cost function value of interest in this paper is less popular. However, sometimes a constant for the latter bounded condition is known, while a constant for the former condition is either not known or difficult to compute, making the latter condition more useful. In addition, there are cases where the latter initial bounded function condition holds, but the former condition does not. One such example is an unregularized logistic regression of an overparameterized model for separable datasets [18, 19], which does not have any finite minimizer. Therefore, this paper's analysis under the initial bounded condition has value for such cases.

Section 2 reviews a smooth convex problem and first-order methods. Section 3 reviews the efficiency of first-order methods and its lower bound. Section 4 studies the PEP approach [1] and provides relaxations for analyzing the worst-case gradient decrease. Section 5 uses the relaxed PEP to provide the exact worst-case gradient bound for GM. Section 6 optimizes the step coefficients of the first-order methods using the relaxed PEP, and develops an efficient first-order method named OGM-G under the initial function condition. Section 7 concludes the paper.

2 Problems and Methods

2.1 Smooth Convex Problems

We are interested in efficiently solving the following smooth and convex minimization problem:

$$f_* := \inf_{\boldsymbol{x}\in\mathbb{R}^d} f(\boldsymbol{x}), \tag{M}$$

where we assume that the function $f : \mathbb{R}^d \to \mathbb{R}$ is a convex function of the type $C_L^{1,1}(\mathbb{R}^d)$, *i.e.*, its gradient $\nabla f(\boldsymbol{x})$ is Lipschitz continuous:

$$||\nabla f(\boldsymbol{x}) - \nabla f(\boldsymbol{y})|| \le L||\boldsymbol{x}-\boldsymbol{y}||, \quad \forall \boldsymbol{x},\boldsymbol{y} \in \mathbb{R}^d \tag{1}$$

with a Lipschitz constant $L > 0$, where $||\cdot||$ denotes the standard Euclidean norm.

Definition 2.1 The class of smooth convex functions satisfying the two above conditions is denoted by $\mathscr{F}_L(\mathbb{R}^d)$.

Definition 2.2 The optimal set of f is defined by

$$X_*(f) := \arg\min_{\boldsymbol{x}\in\mathbb{R}^d} f(\boldsymbol{x}) = \{\boldsymbol{x} \in \mathbb{R}^d \;:\; f(\boldsymbol{x}) = f_*\}. \tag{2}$$

We further assume one of the following two initial conditions, where the latter is especially useful when (M) does not have a finite minimizer, *i.e.*, $X_*(f) = \emptyset$.

Assumption 1 (IFC) *The set $X_*(f)$ is nonempty, and an initial point $\boldsymbol{x}_0$ satisfies*

$$f(\boldsymbol{x}_0) - f_* \le \frac{1}{2}LR^2 \quad \textit{for a constant } R > 0. \tag{IFC}$$

Assumption 2 (IFC′) *An initial point $\boldsymbol{x}_0$ and the Nth iterate $\boldsymbol{x}_N$ of a given method satisfy*

$$f(\boldsymbol{x}_0) - f(\boldsymbol{x}_N) \le \frac{1}{2}LR_N^2 \quad \textit{for a constant } R_N > 0. \tag{IFC′}$$

Note that $f(\boldsymbol{x}_0) - f(\boldsymbol{x}_N) \le f(\boldsymbol{x}_0) - f_*$ for any $\boldsymbol{x}_N$.

2.2 **First-Order Methods**

To solve a large-dimensional problem (M), we consider first-order methods that iteratively gain first-order information, *i.e.*, values of the cost function f and its gradient ∇f at any given point in $\mathbb{R}^d$. The computational effort for acquiring those values depends mildly on the problem dimension. We are interested in developing a first-order method that efficiently generates a point $\boldsymbol{x}_N$ after N iterations (starting from an initial point $\boldsymbol{x}_0$) that minimizes the worst-case absolute gradient inaccuracy under the initial function condition (IFC).

Definition 2.3 The gradient efficiency is defined as the worst-case absolute gradient inaccuracy

$$\sup_{f \in \mathscr{F}_L(\mathbb{R}^d)} ||\nabla f(\boldsymbol{x}_N)||^2. \tag{3}$$

For simplicity in Sects. 4, 5 and 6 that use the PEP approach (as in [1]), we consider the following *fixed-step* first-order methods (FSFOM):

$$\boldsymbol{x}_{i+1} = \boldsymbol{x}_i - \frac{1}{L}\sum_{k=0}^{i} h_{i+1,k} \nabla f(\boldsymbol{x}_k) \quad i = 0,\ldots,N-1, \tag{4}$$

where $\boldsymbol{h} := \{h_{i+1,k}\} \in \mathbb{R}^{N(N+1)/2}$ is a tuple of fixed-step coefficients that do not depend on f, $\boldsymbol{x}_0$ and R (or R_N). This FSFOM class includes (fixed-step) GM (*i.e.*, $h_{i+1,k} = 0$ for $k < i$), (fixed-step) FGM [2,3] (see [1]), OGM [5], and the proposed OGM-G, but excludes line-search approaches, such as a backtracking version of FGM in [20] and an exact line-search version of OGM in [21].

3 Efficiency of First-Order Methods

This paper seeks to improve the *efficiency* of first-order methods, where the efficiency consists of the following two parts: the computational effort for selecting a search point (*e.g.*, computing $\boldsymbol{x}_{i+1}$ in (4) given $\boldsymbol{x}_i$ and $\{\nabla f(\boldsymbol{x}_k)\}_{k=0}^{i}$), and the number of evaluations of the cost function value and gradient at each given search point to reach a given accuracy. This paper considers both parts of the efficiency, particularly focusing on the latter part, as also detailed in this section. Regarding the former aspect of the efficiency, we later show that the proposed method has an efficient form, similar to (fixed-step) FGM and OGM, requiring computational effort comparable to that of a (fixed-step) GM.

An efficiency estimate of an optimization method is defined by the worst-case absolute inaccuracy. One popular choice of the worst-case absolute inaccuracy is the worst-case absolute cost function inaccuracy.

Definition 3.1 The cost function efficiency is defined as the worst-case absolute cost function inaccuracy

$$\sup_{f \in \mathscr{F}_L(\mathbb{R}^d)} f(\boldsymbol{x}_N) - f_*. \tag{5}$$

When analyzing the cost function efficiency, we usually consider the following initial condition.

Assumption 3 (IDC) *The set $X_*(f)$ is nonempty, and an initial point $\boldsymbol{x}_0$ satisfies*

$$||\boldsymbol{x}_0 - \boldsymbol{x}_*|| \le \overline{R} \quad \textit{for a constant } \overline{R} > 0, \tag{IDC}$$

for some $\boldsymbol{x}_* \in X_*(f)$.

Under (IDC), GM has an $O(1/N)$ cost function efficiency (5) [3], and this rate was improved to $O(1/N^2)$ rate by FGM [2,3]. This efficiency was further optimized by OGM [1,5], which was shown to exactly achieve the optimal efficiency in [6].

Compared to the worst-case *cost function* inaccuracy (5), the worst-case absolute *gradient* inaccuracy (3) has received less attention [4, 9, 17, 22]. For the initial *distance* condition (IDC), GM has an $O(1/N^2)$ gradient efficiency [9], while FGM with a regularization technique [9] that requires the knowledge of (practically unavailable) $\overline{R}$ achieves $O(1/N^4)$ up to a logarithmic factor. This is the best known rate, where the rate $O(1/N^4)$ is the optimal gradient efficiency for given (IDC) [4]. On the other hand, the papers [9, 13–16] studied first-order methods that do not require knowing $\overline{R}$ and that have $O(1/N^3)$ gradient efficiency, but none of them (including [9]) have the optimal efficiency (even up to a constant).

On the other hand, gradient efficiency with the initial *function* condition (IFC) has received even less attention [4,22]. It is known to have $O(1/N^2)$ optimal efficiency [4]. Section 5 provides the exact $O(1/N)$ rate of GM, which was studied numerically for the more general nonsmooth composite convex problems in [22]. The paper [12] discusses that FGM with a regularization technique [9] with (IFC) also achieves the optimal worst-case gradient rate $O(1/N^2)$ up to a logarithmic factor. This is the best previously known rate, and this paper provides a better rate.

In short, none of the existing first-order methods achieve the optimal rate for the gradient inaccuracy even up to a constant, and thus this paper focuses on optimizing the gradient efficiency of first-order methods for smooth convex minimization with (IFC) and (IFC′). Table 1 summarizes the efficiency of first-order methods and illustrates that the proposed OGM-G attains the optimal worst-case gradient rate $O(1/N^2)$ with (IFC) and (IFC′).

Remark 3.1 After the initial version of this paper was posted online [7], the paper [8, Remark 2.1] constructed a simple method using OGM-G that achieves $O(1/N^4)$ under the initial distance condition (IDC). The method runs an accelerated method such as Nesterov's FGM and OGM for the first half of the iterations and then runs OGM-G for the rest. That approach (built upon the proposed OGM-G) further closes the open problem of developing an optimal method for decreasing the gradient, under the initial distance condition (IDC).

Table 1 Summary of the efficiency of first-order methods discussed in Sect. 3 [3,4,9,12,22]; The rates of the proposed OGM-G and a method in [8, Remark 2.1] using OGM-G (see Remark 3.1) are also presented. $\widetilde{O}(\cdot)$ is a big-O notation that ignores a logarithmic factor.

Efficiency	Initial cond.	GM rate	Best known rate		OGM-G rate	[8] rate	Optimal rate
			w/o R or $\overline{R}$	w/ R or $\overline{R}$			
Cost func. (5)	(IDC)	$O(1/N)$	$O(1/N^2)$		·	·	$O(1/N^2)$
Gradient (3)	(IDC)	$O(1/N^2)$	$O(1/N^3)$	$\widetilde{O}(1/N^4)$	·	$O(1/N^4)$	$O(1/N^4)$
	(IFC)	$O(1/N)$	$O(1/N)$	$\widetilde{O}(1/N^2)$	$O(1/N^2)$	·	$O(1/N^2)$
	(IFC′)	$O(1/N)$	·	·	$O(1/N^2)$	·	·

As Table 1 demonstrates, worst-case rates of any given method and optimal worst-case rates depend dramatically on the initial condition. In particular, the worst-case gradient rates for (IFC) tend to be slower than those for (IDC). At first glance, this situation might hinder one's interest on the initial function condition (IFC) studied in this paper. However, one should also consider the constants R and $\overline{R}$ for a fair comparison of the worst-case rates. In particular, consider a problem instance $(f, \boldsymbol{x}_0)$ where $f \in \mathscr{F}_L(\mathbb{R}^d)$ and $X_*(f) \neq \emptyset$. Then, choose R and $\overline{R}$ such that

$$f(\boldsymbol{x}_0) - f_* = \frac{1}{2}LR^2 \quad \text{and} \quad ||\boldsymbol{x}_0 - \boldsymbol{x}_*|| = \overline{R} \tag{6}$$

for some $\boldsymbol{x}_* \in X_*(f)$. Using the inequality $f(\boldsymbol{x}_0) - f_* \leq \frac{L}{2}||\boldsymbol{x}_0 - \boldsymbol{x}_*||^2$ due to the smoothness of f in (1), we have the relationship:

$$R \leq \overline{R}. \tag{7}$$

For any optimization method, including GM and OGM-G, the ratio $\overline{R}/R$ can be in the order of N or beyond, for a given N, and should not be neglected. Section 6.5 gives one such example.

4 Performance Estimation Problem (PEP) for the Worst-Case Gradient Decrease

This section studies PEP [1] and its relaxations for the worst-case gradient analysis under the condition (IFC).

4.1 **Exact PEP**

The papers [1, 17] suggest that for any given step coefficients $\boldsymbol{h} := \{h_{i,k}\}$ of a FSFOM, total number of iterations N, problem dimension d, and constants L, R, the exact worst-case gradient bound under (IFC) is given by

$$\mathscr{B}_{\text{P}}(\boldsymbol{h},N,d,L,R) := \max_{f\in\mathscr{F}_L(\mathbb{R}^d)} \max_{\boldsymbol{x}_0,\ldots,\boldsymbol{x}_N\in\mathbb{R}^d} \frac{1}{L^2R^2}||\nabla f(\boldsymbol{x}_N)||^2 \quad \text{s.t.} \begin{cases} \boldsymbol{x}_{i+1} = \boldsymbol{x}_i - \frac{1}{L}\sum_{k=0}^{i} h_{i+1,k}\nabla f(\boldsymbol{x}_k), \quad i=0,\ldots,N-1, \\ f(\boldsymbol{x}_0) - f_* \le \frac{1}{2}LR^2, \quad \boldsymbol{x}_* \in X_*(f) \neq \emptyset, \end{cases} \tag{P}$$

where $||\nabla f(\boldsymbol{x}_N)||^2$ is multiplied by $\frac{1}{L^2R^2}$ for convenience in later analysis. However, as noted in [1], it is intractable to solve (P) due to its infinite-dimensional function constraint. Thus the next section employs relaxations introduced in [1].

4.2 **Relaxing PEP**

As suggested by [1, 17], to convert (P) into an equivalent finite-dimensional problem, we replace the infinite-dimensional constraint $f \in \mathscr{F}_L(\mathbb{R}^d)$ by a finite set of inequalities satisfied by $f \in \mathscr{F}_L(\mathbb{R}^d)$ [3, Theorem 2.1.5]:

$$\frac{1}{2L}||\nabla f(\boldsymbol{x}_i) - \nabla f(\boldsymbol{x}_j)||^2 \le f(\boldsymbol{x}_i) - f(\boldsymbol{x}_j) - \langle \nabla f(\boldsymbol{x}_j), \boldsymbol{x}_i - \boldsymbol{x}_j\rangle \tag{8}$$

on each pair (i,j) for $i,j = 0,\ldots,N,*$. For simplicity in the proofs, we further narrow down the set[1] of inequalities (8), specifically the pairs $\{(i-1,i) : i=1,\ldots,N\}$, $\{(N,i) : i=0,\ldots,N-1\}$ and $\{(N,*)\}$.[2] This relaxation leads to

$$\mathscr{B}_{\text{P1}}(\boldsymbol{h},N,d) := \max_{\substack{\boldsymbol{G}\in\mathbb{R}^{(N+1)\times d},\\ \boldsymbol{\delta}\in\mathbb{R}^{N+1}}} \operatorname{Tr}\left\{\boldsymbol{G}^\top \boldsymbol{u}_N\boldsymbol{u}_N^\top \boldsymbol{G}\right\} \quad \text{s.t.} \begin{cases} \operatorname{Tr}\left\{\boldsymbol{G}^\top \boldsymbol{A}_{i-1,i}(\boldsymbol{h})\boldsymbol{G}\right\} \le \delta_{i-1} - \delta_i, \quad i=1,\ldots,N, \\ \operatorname{Tr}\left\{\boldsymbol{G}^\top \boldsymbol{B}_{N,i}(\boldsymbol{h})\boldsymbol{G}\right\} \le \delta_N - \delta_i, \quad i=0,\ldots,N-1, \\ \operatorname{Tr}\left\{\boldsymbol{G}^\top \boldsymbol{C}_N\boldsymbol{G}\right\} \le \delta_N, \quad \delta_0 \le \frac{1}{2}, \end{cases} \tag{P1}$$

where we define

$$\begin{cases} \boldsymbol{g}_i := \frac{1}{LR}\nabla f(\boldsymbol{x}_i), \quad i=0,\ldots,N, \quad \boldsymbol{G} := [\boldsymbol{g}_0,\ldots,\boldsymbol{g}_N]^\top, \\ \delta_i := \frac{1}{LR^2}(f(\boldsymbol{x}_i) - f_*), \quad i=0,\ldots,N, \quad \boldsymbol{\delta} := [\delta_0,\ldots,\delta_N]^\top, \\ \boldsymbol{u}_i := [0,\ldots,0,\underbrace{1}_{(i+1)\text{th entry}},0,\ldots,0]^\top \in \mathbb{R}^{N+1}, \quad i=0,\ldots,N, \end{cases} \tag{9}$$

and

$$\begin{cases} \boldsymbol{A}_{i-1,i}(\boldsymbol{h}) := \frac{1}{2}(\boldsymbol{u}_{i-1} - \boldsymbol{u}_i)(\boldsymbol{u}_{i-1} - \boldsymbol{u}_i)^\top + \frac{1}{2}\sum_{k=0}^{i-1} h_{i,k}(\boldsymbol{u}_i\boldsymbol{u}_k^\top + \boldsymbol{u}_k\boldsymbol{u}_i^\top), \quad i=1,\ldots,N, \\ \boldsymbol{B}_{N,i}(\boldsymbol{h}) := \frac{1}{2}(\boldsymbol{u}_N - \boldsymbol{u}_i)(\boldsymbol{u}_N - \boldsymbol{u}_i)^\top - \frac{1}{2}\sum_{l=i+1}^{N}\sum_{k=0}^{l-1} h_{l,k}(\boldsymbol{u}_i\boldsymbol{u}_k^\top + \boldsymbol{u}_k\boldsymbol{u}_i^\top), \\ \qquad\qquad\qquad i=0,\ldots,N-1, \\ \boldsymbol{C}_N := \frac{1}{2}\boldsymbol{u}_N\boldsymbol{u}_N^\top. \end{cases} \tag{10}$$

[1] We found that the set of constraints in (P1) is sufficient for the exact worst-case gradient analysis of GM and OGM-G for (IFC), as illustrated in later sections. In other words, the resulting worst-case rates of GM and OGM-G in this paper are tight with our specific choice of the set of inequalities. Note that this relaxation choice in (P1) differs from the choice in [1, Problem (G′)].

[2] The inequality (8) for the pair $\{(N,*)\}$ simplifies to $\frac{1}{2L}||\nabla f(\boldsymbol{x}_N)||^2 \le f(\boldsymbol{x}_N) - f_*$ under the condition $X_*(f) \neq \emptyset$. Such inequality is not used under the assumption (IFC′) in Corollaries 5.1 and 6.1.

As in [17], we further relax (P1) by introducing the Gram matrix $\boldsymbol{Z} := \boldsymbol{G}\boldsymbol{G}^\top$ as

$$\mathscr{B}_{\text{P2}}(\boldsymbol{h},N,d) := \max_{\substack{\boldsymbol{Z}\in\mathbb{S}_+^{N+1},\\ \boldsymbol{\delta}\in\mathbb{R}^{N+1}}} \operatorname{Tr}\left\{\boldsymbol{u}_N\boldsymbol{u}_N^\top\boldsymbol{Z}\right\} \tag{P2}$$

$$\text{s.t.} \begin{cases} \operatorname{Tr}\{\boldsymbol{A}_{i-1,i}(\boldsymbol{h})\boldsymbol{Z}\} \le \delta_{i-1}-\delta_i, & i=1,\dots,N,\\ \operatorname{Tr}\{\boldsymbol{B}_{N,i}(\boldsymbol{h})\boldsymbol{Z}\} \le \delta_N-\delta_i, & i=0,\dots,N-1,\\ \operatorname{Tr}\{\boldsymbol{C}_N\boldsymbol{Z}\} \le \delta_N, \quad \delta_0 \le \frac{1}{2}. \end{cases}$$

This problem has the following Lagrangian dual:

$$\mathscr{B}_{\text{D}}(\boldsymbol{h},N) := \min_{(\boldsymbol{a},\boldsymbol{b},c,e)\in\mathbb{R}_+^{2N+2}} \frac{1}{2}e \tag{D}$$

$$\text{s.t.} \begin{cases} \boldsymbol{S}(\boldsymbol{h},\boldsymbol{a},\boldsymbol{b},c) \succeq \boldsymbol{0}, \quad -a_1+b_0+e=0, \quad a_N-\sum_{i=0}^{N-1} b_i - c = 0,\\ a_i - a_{i+1} + b_i = 0, \quad i=1,\dots,N-1. \end{cases}$$

where

$$\begin{aligned} \boldsymbol{S}(\boldsymbol{h},\boldsymbol{a},\boldsymbol{b},c) &:= \sum_{i=1}^N a_i\boldsymbol{A}_{i-1,i}(\boldsymbol{h}) + \sum_{i=0}^{N-1} b_i\boldsymbol{B}_{N,i}(\boldsymbol{h}) + c\boldsymbol{C}_N(\boldsymbol{h}) - \boldsymbol{u}_N\boldsymbol{u}_N^\top \\ &= \frac{1}{2}\sum_{i=1}^N a_i(\boldsymbol{u}_{i-1}-\boldsymbol{u}_i)(\boldsymbol{u}_{i-1}-\boldsymbol{u}_i)^\top + \frac{1}{2}\sum_{i=0}^{N-1} b_i(\boldsymbol{u}_N-\boldsymbol{u}_i)(\boldsymbol{u}_N-\boldsymbol{u}_i)^\top + \frac{1}{2}(c-2)\boldsymbol{u}_N\boldsymbol{u}_N^\top \\ &\quad + \frac{1}{2}\sum_{i=1}^N\sum_{k=0}^{i-1} a_i h_{i,k}(\boldsymbol{u}_i\boldsymbol{u}_k^\top + \boldsymbol{u}_k\boldsymbol{u}_i^\top) - \frac{1}{2}\sum_{i=0}^{N-1}\sum_{k=0}^{N-1}\left(b_i \sum_{l=\max\left\{\substack{i+1,\\ k+1}\right\}}^{N} h_{l,k}\right)(\boldsymbol{u}_i\boldsymbol{u}_k^\top + \boldsymbol{u}_k\boldsymbol{u}_i^\top). \end{aligned} \tag{11}$$

For given $\boldsymbol{h}$ and N, a semidefinite programming (SDP) problem (D) can be solved numerically using an SDP solver (*e.g.*, CVX [23, 24]). The next two sections analytically specify feasible points of (D) for GM and OGM-G, which were numerically first identified to be solutions of (D) for each method by the authors. These feasible points provide the exact worst-case analytical gradient bounds for GM and OGM-G.

5 Applying the Relaxed PEP to GM

Inspired by the numerical solutions of (D) for GM using CVX [23, 24], we next specify a feasible point of (D) for GM.

Lemma 5.1 *For GM,* i.e. *the FSFOM with $h_{i+1,k}$ having 1 for $k=i$ and 0 otherwise, the following set of dual variables:*

$$\begin{cases} a_i = \frac{2(N+i)}{(N-i+1)(2N+1)} = \frac{N+i}{N-i+1}e, & i=1,\dots,N,\\ b_i = \begin{cases} \frac{2}{N(2N+1)} = \frac{1}{N}e, & i=0,\\ \frac{2}{(N-i)(N-i+1)}, & i=1,\dots,N-1, \end{cases}\\ c = e = \frac{2}{2N+1}, \end{cases} \tag{12}$$

is a feasible point of (D).

Proof *Obviously,* (12) *satisfies the equality conditions of* (D)*, and the rest of proof shows the positive semidefinite condition of* (D).

For any $\boldsymbol{h}$ *and* $(\boldsymbol{a},\boldsymbol{b},c,e)\in\Lambda$, *the* (i,j)*th entry of the symmetric matrix* (11) *can be rewritten as*

$$
\begin{aligned}
&[2\boldsymbol{S}(\boldsymbol{h},\boldsymbol{a},\boldsymbol{b},c)]_{ij} \qquad (13)\\
&=\begin{cases}
a_1+b_0\left(1-2\sum_{l=1}^N h_{l,0}\right), & i=0,\ j=i,\\
a_i+a_{i+1}+b_i\left(1-2\sum_{l=i+1}^N h_{l,i}\right), & i=1,\dots,N-1,\ j=i,\\
a_N+\sum_{l=0}^{N-1} b_l+c-2=2(a_N-1), & i=N,\ j=i,\\
a_i(h_{i,i-1}-1)-b_i\sum_{l=i+1}^N h_{l,i-1}-b_{i-1}\sum_{l=i+1}^N h_{l,i}, & i=1,\dots,N-1,\ j=i-1,\\
a_N(h_{N,N-1}-1)-b_{N-1}, & i=N,\ j=i-1,\\
a_ih_{i,j}-b_i\sum_{l=i+1}^N h_{l,j}-b_j\sum_{l=i+1}^N h_{l,i}, & i=2,\dots,N-1,\\
& j=0,\dots,i-2,\\
a_Nh_{N,j}-b_j, & i=N,\ j=0,\dots,i-2.
\end{cases}
\end{aligned}
$$

Substituting the step coefficients $\boldsymbol{h}$ *for GM and the dual variables* (12) *in* (13) *yields*

$$
[2\boldsymbol{S}(\boldsymbol{h},\boldsymbol{a},\boldsymbol{b},c)]_{ij}=\begin{cases}
a_1-b_0=e, & i=0,\ j=i,\\
a_i+a_{i+1}-b_i=2a_i, & i=1,\dots,N-1,\ j=i,\\
2(a_N-1), & i=N,\ j=i,\\
-b_j, & i=1,\dots,N,\ j=0,\dots,i-1,
\end{cases}
\qquad (14)
$$

The matrix (14) *has nonnegative diagonal entries, and thus showing the diagonal dominance of the matrix* (14) *implies its positive semidefiniteness.*

A sum of absolute values of nondiagonal elements for each row is

$$
\begin{aligned}
\sum_{\substack{j=0\\ j\neq i}}^N \left|[2\boldsymbol{S}(\boldsymbol{h},\boldsymbol{a},\boldsymbol{b},c)]_{ij}\right| &=\begin{cases}
Nb_0, & i=0,\\
b_0+(N-1)b_1 & i=1,\\
\sum_{j=0}^{i-1} b_l+(N-i)b_i & i=2,\dots,N-1,\\
\sum_{j=0}^{N-1} b_j & i=N,
\end{cases} \qquad (15)\\
&=\begin{cases}
\frac{2}{2N+1}, & i=0,\\
\frac{2}{N(2N+1)}+\frac{2}{N}=\frac{4(N+1)}{N(2N+1)}, & i=1,\\
\frac{2}{N(2N+1)}+\frac{2}{N-i+1}-\frac{2}{N}+\frac{2}{N-i+1}=\frac{4(N+i)}{(N-i+1)(2N+1)}, & i=2,\dots,N-1,\\
\frac{2}{N(2N+1)}+2-\frac{2}{N}=\frac{2(2N-1)}{2N+1}, & i=N,
\end{cases}\\
&=\begin{cases}
e, & i=0,\\
\frac{2(N+i)}{(N-i+1)}e, & i=1,\dots,N-1,\\
2(2Ne-1), & i=N,
\end{cases}
\end{aligned}
$$

and this satisfies $[2\boldsymbol{S}(\boldsymbol{h},\boldsymbol{a},\boldsymbol{b},c)]_i=\sum_{\substack{j=0\\ j\neq i}}^N\left|[2\boldsymbol{S}(\boldsymbol{h},\boldsymbol{a},\boldsymbol{b},c)]_{ij}\right|$ *for all* i, i.e., *the matrix* (14) *is diagonally dominant, and this concludes the proof.* □

The next theorem provides the worst-case convergence gradient bound of GM.

Theorem 5.1 *Assume that* $f\in\mathscr{F}_L(\mathbb{R}^d)$, $X_*(f)\neq\emptyset$, *and* $f(\boldsymbol{x}_0)-f_*\leq\frac{1}{2}LR^2$ (IFC). *Let* $\boldsymbol{x}_0,\dots,\boldsymbol{x}_N\in\mathbb{R}^d$ *be generated by GM,* i.e., *the FSFOM with* $h_{i+1,k}$ *having* 1 *for* $k=i$ *and* 0 *otherwise. Then, for any* $N\geq1$,

$$
||\nabla f(\boldsymbol{x}_N)||^2\leq\frac{L^2R^2}{2N+1}. \qquad (16)
$$

Proof *Using Lemma 5.1 for the step coefficients* $\boldsymbol{h}$ *of GM, we have*

$$
||\nabla f(\boldsymbol{x}_N)||^2\leq L^2R^2\mathscr{B}_{\mathrm{D}}(\boldsymbol{h},N)\leq L^2R^2\frac{1}{2N+1}.
$$

□

The PEP proof of Theorem 5.1, using Lemma 5.1, can be used to construct a conventional proof that derives inequality (16) by a weighted sum of the inequalities (8). Specifically, one can use a weighted sum of inequalities using the dual variables $(\boldsymbol{a},\boldsymbol{b},c,e)$ in (12) as weights:

$$\begin{aligned} \frac{1}{2L}||\nabla f(\boldsymbol{x}_{i-1})-\nabla f(\boldsymbol{x}_i)||^2 \leq f(\boldsymbol{x}_{i-1})-f(\boldsymbol{x}_i)-\langle\nabla f(\boldsymbol{x}_i),\boldsymbol{x}_{i-1}-\boldsymbol{x}_i\rangle \quad &: \quad a_i \\ \frac{1}{2L}||\nabla f(\boldsymbol{x}_N)-\nabla f(\boldsymbol{x}_i)||^2 \leq f(\boldsymbol{x}_N)-f(\boldsymbol{x}_i)-\langle\nabla f(\boldsymbol{x}_i),\boldsymbol{x}_N-\boldsymbol{x}_i\rangle \quad &: \quad b_i \\ \frac{1}{2L}||\nabla f(\boldsymbol{x}_N)||^2 \leq f(\boldsymbol{x}_N)-f_* \quad &: \quad c \\ f(\boldsymbol{x}_0)-f_* \leq \frac{1}{2}LR^2 \quad &: \quad e, \end{aligned} \tag{17}$$

which simplifies to

$$\frac{1}{L}||\nabla f(\boldsymbol{x}_N)||^2+\sum_{i=1}^{N}\sum_{j=0}^{i-1}\frac{b_j}{2L}\left|\left|\nabla f(\boldsymbol{x}_i)-\nabla f(\boldsymbol{x}_j)\right|\right|^2 \leq \frac{LR^2}{2N+1}, \tag{18}$$

and this yields (16).

We next show that the bound (16) is exact by specifying a certain worst-case function. This implies that the feasible point in (12) is an optimal point of (D) for GM.

Lemma 5.2 *For the following Huber function in $\mathscr{F}_L(\mathbb{R}^d)$ for all $d \geq 1$:*

$$\phi(\boldsymbol{x}) = \begin{cases} \frac{LR}{\sqrt{2N+1}}||\boldsymbol{x}||-\frac{LR^2}{2(2N+1)}, & ||\boldsymbol{x}|| \geq \frac{R}{\sqrt{2N+1}}, \\ \frac{L}{2}||\boldsymbol{x}||^2, & \textit{otherwise}, \end{cases} \tag{19}$$

GM exactly achieves the bound (16) with $\boldsymbol{x}_0$ satisfying $\phi(\boldsymbol{x}_0)-\phi_* = \frac{1}{2}LR^2$.

Proof *Starting from $\boldsymbol{x}_0 = \frac{N+1}{\sqrt{2N+1}}R\boldsymbol{\nu}$ that satisfies $\phi(\boldsymbol{x}_0)-\phi_* = \frac{1}{2}LR^2$ (IFC) for any unit-norm vector $\boldsymbol{\nu}$, the iterates of GM are as follows:*

$$\boldsymbol{x}_i = \boldsymbol{x}_0-\frac{1}{L}\sum_{k=0}^{i-1}\nabla\phi(\boldsymbol{x}_k) = \left(\frac{N+1}{\sqrt{2N+1}}-\frac{i}{\sqrt{2N+1}}\right)R\boldsymbol{\nu}, \quad i=0,\ldots,N,$$

where all the iterates stay in the affine region of the function $\phi(\boldsymbol{x})$ with the same gradient $\nabla\phi(\boldsymbol{x}_i) = \frac{LR}{\sqrt{2N+1}}\boldsymbol{\nu}$, $i = 0,\ldots,N$. Therefore, after N iterations of GM, we have $||\nabla\phi(\boldsymbol{x}_N)||^2 = \frac{L^2R^2}{2N+1}$, which concludes the proof. □

Remark 5.1 For $f \in \mathscr{F}_L(\mathbb{R}^d)$, and for some $\boldsymbol{x}_* \in X_*(f)$ and $||\boldsymbol{x}_0-\boldsymbol{x}_*|| \leq \overline{R}$ (IDC), the Nth iterate $\boldsymbol{x}_N$ of GM has the following exact worst-case cost function bound [1, Theorems 1 and 2]:

$$f(\boldsymbol{x}_N)-f(\boldsymbol{x}_*) \leq \frac{L\overline{R}^2}{2(2N+1)}, \tag{20}$$

where this exact upper bound is equivalent to the exact worst-case gradient bound (16) of GM up to a constant $\frac{\overline{R}^2}{2LR^2}$. A similar relationship appears in [22, Table 3] for nonsmooth composite convex minimization.

The preceding results in this section assume that there is a finite minimizer. There are applications that do not have a finite minimizer $\boldsymbol{x}_* \in X_*(f)$, *e.g.*, an unregularized logistic regression of an overparameterized model for separable datasets [18, 19]. The following corollary extends the analysis to such cases.

Corollary 5.1 *For $f \in \mathscr{F}_L(\mathbb{R}^d)$, let $\boldsymbol{x}_0,\ldots,\boldsymbol{x}_N \in \mathbb{R}^d$ be generated by GM. Assume that $f(\boldsymbol{x}_0)-f(\boldsymbol{x}_N) \leq \frac{1}{2}LR_N^2$ (IFC′). Then, for any $N \geq 1$,*

$$||\nabla f(\boldsymbol{x}_N)||^2 \leq \frac{L^2R_N^2}{2N}. \tag{21}$$

Proof *Equation* (18) *consists of a weighted sum of the third and fourth inequalities of* (17)*, scaled by* $c = e = \frac{2}{2N+1}$ *in* (12)*:*

$$\frac{c}{2L}||\nabla f(\boldsymbol{x}_N)||^2 + c(f(\boldsymbol{x}_0) - f(\boldsymbol{x}_N)) \le \frac{c}{2}LR^2.$$

The third inequality of (17) *assumes* $X_*(f) \neq \emptyset$ *(see footnote* 2*), so we derive a bound without the above inequality. Replacing the above inequality in the weighted summation for deriving* (18) *by* (IFC$'$) *scaled by* c, $c(f(\boldsymbol{x}_0) - f(\boldsymbol{x}_N)) \le \frac{c}{2}LR_N^2$*, yields*

$$\frac{1}{L}\left(1 - \frac{c}{2}\right)||\nabla f(\boldsymbol{x}_N)||^2 + \sum_{i=1}^{N}\sum_{j=0}^{i-1}\frac{b_j}{2L}\left|\left|\nabla f(\boldsymbol{x}_i) - \nabla f(\boldsymbol{x}_j)\right|\right|^2 \le \frac{LR_N^2}{2N+1},$$

which concludes the proof. □

6 Optimizing FSFOM Using the Relaxed PEP

This section optimizes the step coefficients of FSFOM using the relaxed PEP (D) to develop an efficient first-order method for decreasing the gradient of smooth convex functions.

6.1 Numerically Optimizing FSFOM Using the Relaxed PEP

To optimize the step coefficients of $\boldsymbol{h}$ of FSFOM for each given N, we are interested in solving

$$\tilde{\boldsymbol{h}} := \arg\min_{\boldsymbol{h}} \mathscr{B}_{\mathrm{D}}(\boldsymbol{h}, N), \tag{HD}$$

which is nonconvex. However, the problem (HD) is bi-convex over $\boldsymbol{h}$ and $(\boldsymbol{a}, \boldsymbol{b}, c, e, \gamma)$, so for each given N we numerically solved (HD) by an alternating minimization approach using CVX [23,24]. Inspired by those numerical solutions, the next section specifies a feasible point of (HD).

6.2 A Feasible Point of the Relaxed PEP

The following lemma specifies a feasible point of (HD).

Lemma 6.1 *The following step coefficients of FSFOM:*

$$\tilde{h}_{i+1,k} = \begin{cases} \frac{\tilde{\theta}_{k+1}-1}{\tilde{\theta}_k}\tilde{h}_{i+1,k+1}, & k = 0,\dots,i-2, \\ \frac{\tilde{\theta}_{k+1}-1}{\tilde{\theta}_k}(\tilde{h}_{i+1,i} - 1), & k = i-1, \\ 1 + \frac{2\tilde{\theta}_{i+1}-1}{\tilde{\theta}_i}, & k = i, \end{cases} \tag{22}$$

and the following set of dual variables:

$$a_i = \frac{1}{\tilde{\theta}_i^2}, \quad i = 1,\dots,N, \quad b_i = \frac{1}{\tilde{\theta}_i\tilde{\theta}_{i+1}^2}, \quad i = 0,\dots,N-1, \quad c = e = \frac{2}{\tilde{\theta}_0^2}, \tag{23}$$

constitute a feasible point of (HD) *for the parameters:*

$$\tilde{\theta}_i = \begin{cases} \frac{1+\sqrt{1+8\tilde{\theta}_{i+1}^2}}{2}, & i = 0, \\ \frac{1+\sqrt{1+4\tilde{\theta}_{i+1}^2}}{2}, & i = 1,\dots,N-1, \\ 1, & i = N. \end{cases} \tag{24}$$

Proof *The appendix first derives properties of the step coefficients* $\tilde{\boldsymbol{h}} = \{\tilde{h}_{i,k}\}$ (22) *that are used in the proof:*

$$\tilde{h}_{i,j} = \frac{\tilde{\theta}_i^2(2\tilde{\theta}_i - 1)}{\tilde{\theta}_j\tilde{\theta}_{j+1}^2}, \quad i = 2,\dots,N,\ j = 0,\dots,i-2, \tag{25}$$

$$\sum_{l=i+1}^{N} \tilde{h}_{l,j} = \begin{cases} \frac{1}{2}(\tilde{\theta}_0 + 1), & i = 0,\ j = i, \\ \tilde{\theta}_i, & i = 1,\dots,N-1,\ j = i, \\ \frac{\tilde{\theta}_{i+1}^4}{\tilde{\theta}_j\tilde{\theta}_{j+1}^2}, & i = 1,\dots,N-1,\ j = 0,\dots,i-1. \end{cases} \tag{26}$$

By definition of $\tilde{\theta}_i$ (24)*, we also have*

$$\tilde{\theta}_i^2 = \begin{cases} \tilde{\theta}_i + 2\tilde{\theta}_{i+1}^2, & i = 0, \\ \tilde{\theta}_i + \tilde{\theta}_{i+1}^2, & i = 1,\dots,N-1. \end{cases} \tag{27}$$

Obviously, (23) *satisfies the equality conditions of* (D)*, and the rest of proof shows the positive semidefinite condition of* (D)*. Substituting the step coefficients* $\tilde{\boldsymbol{h}}$ (22) *and the dual variables* (23) *with their properties* (25), (26) *and* (27) *in* (13) *yields*

$$\begin{aligned}
&[2\boldsymbol{S}(\boldsymbol{h},\boldsymbol{a},\boldsymbol{b},c)]_{ij} \\
&= \begin{cases}
\frac{1}{\tilde{\theta}_1^2} + \frac{1}{\tilde{\theta}_0\tilde{\theta}_1^2}(1 - (\tilde{\theta}_0 + 1)), \quad i = 0,\ j = i, \\
\frac{1}{\tilde{\theta}_i^2} + \frac{1}{\tilde{\theta}_{i+1}^2} + \frac{1}{\tilde{\theta}_i\tilde{\theta}_{i+1}^2}\left(1 - 2\tilde{\theta}_i\right) = \frac{\tilde{\theta}_{i+1}^2 + \tilde{\theta}_i - \tilde{\theta}_i^2}{\tilde{\theta}_i^2\tilde{\theta}_{i+1}^2}, \quad i = 1,\dots,N-1,\ j = i, \\
2\left(\frac{1}{\tilde{\theta}_N^2} - 1\right), \quad i = N,\ j = i, \\
\frac{1}{\tilde{\theta}_i^2}\frac{2\tilde{\theta}_i - 1}{\tilde{\theta}_{i-1}} - \frac{1}{\tilde{\theta}_i\tilde{\theta}_{i+1}^2}\frac{\tilde{\theta}_{i+1}^4}{\tilde{\theta}_{i-1}\tilde{\theta}_i^2} - \frac{1}{\tilde{\theta}_{i-1}\tilde{\theta}_i^2}\tilde{\theta}_i = \frac{(2\tilde{\theta}_i - 1)\tilde{\theta}_i - \tilde{\theta}_{i+1}^2 - \tilde{\theta}_i^2}{\tilde{\theta}_{i-1}\tilde{\theta}_i^3}, \ i = 1,\dots,N-1,\ j = i-1, \\
\frac{1}{\tilde{\theta}_N^2}\frac{2\tilde{\theta}_N - 1}{\tilde{\theta}_{N-1}} - \frac{1}{\tilde{\theta}_{N-1}\tilde{\theta}_N^2}, \quad i = N,\ j = i-1, \\
\frac{1}{\tilde{\theta}_i^2}\frac{\tilde{\theta}_i^2(2\tilde{\theta}_i - 1)}{\tilde{\theta}_j\tilde{\theta}_{j+1}^2} - \frac{1}{\tilde{\theta}_i\tilde{\theta}_{i+1}^2}\frac{\tilde{\theta}_{i+1}^4}{\tilde{\theta}_j\tilde{\theta}_{j+1}^2} - \frac{1}{\tilde{\theta}_j\tilde{\theta}_{j+1}^2}\tilde{\theta}_i = \frac{(2\tilde{\theta}_i - 1)\tilde{\theta}_i - (\tilde{\theta}_i - 1)^2\tilde{\theta}_i^2 - \tilde{\theta}_i^2}{\tilde{\theta}_j\tilde{\theta}_{j+1}^2\tilde{\theta}_i}, \quad i = 2,\dots,N-1, \\
\qquad\qquad\qquad j = 0,\dots,i-2, \\
\frac{1}{\tilde{\theta}_N^2}\frac{1}{\tilde{\theta}_j\tilde{\theta}_{j+1}^2} - \frac{1}{\tilde{\theta}_j\tilde{\theta}_{j+1}^2}, \quad i = N,\ j = 0,\dots,i-2,
\end{cases} \\
&= \boldsymbol{0},
\end{aligned}$$

which concludes the proof. □

The next theorem provides the worst-case convergence gradient bound of FSFOM with step coefficients (22).

Theorem 6.1 *Assume that* $f \in \mathscr{F}_L(\mathbb{R}^d)$*,* $X_*(f) \neq \emptyset$*, and* $f(\boldsymbol{x}_0) - f_* \leq \frac{1}{2}LR^2$ (IFC)*. Let* $\boldsymbol{x}_0,\dots,\boldsymbol{x}_N \in \mathbb{R}^d$ *be generated by FSFOM with step coefficients* (22)*. Then, for any* $N \geq 1$,

$$||\nabla f(\boldsymbol{x}_N)||^2 \leq \frac{L^2R^2}{\tilde{\theta}_0^2} \leq \frac{2L^2R^2}{(N+1)^2}. \tag{28}$$

Proof *Using Lemma 6.1, we have* $||\nabla f(\boldsymbol{x}_N)||^2 \leq L^2R^2\mathscr{B}_{\mathrm{D}}(\boldsymbol{h},N) \leq L^2R^2\frac{1}{\tilde{\theta}_0^2}$*. We can easily show that* $\tilde{\theta}_i$ (24) *satisfies* $\tilde{\theta}_i \geq \frac{N-i+2}{2}$ *for* $i = 1,\dots,N$ *by induction, and this then yields* $\tilde{\theta}_0 \geq \frac{N+1}{\sqrt{2}}$*, which concludes the proof.* □

Similar to (18), the PEP proof of Theorem 6.1, using Lemma 6.1, can be used to construct a conventional proof by a weighted sum of inequalities (17) using the dual variables $(\boldsymbol{a},\boldsymbol{b},c,e)$ in (23) as weights. This weighted sum leads to

$$\frac{1}{L}||\nabla f(\boldsymbol{x}_N)||^2 \leq \frac{LR^2}{\tilde{\theta}_0^2} \tag{29}$$

and yields (28).

The bound (28) of FSFOM with (22) is optimal up to a constant because Nemirovsky shows in [4] that the worst-case rate for the gradient decrease of large-dimensional convex *quadratic* function is $O(1/N^2)$ under (IFC). This result fills in Table 1, improving upon best known rates.

The following corollary examines the rate of FSFOM with (22) for cases where a finite minimizer might not exist.

Corollary 6.1 *For $f \in \mathscr{F}_L(\mathbb{R}^d)$, let $\boldsymbol{x}_0, \ldots, \boldsymbol{x}_N \in \mathbb{R}^d$ be generated by FSFOM with step coefficients (22). Assume that $f(\boldsymbol{x}_0) - f(\boldsymbol{x}_N) \le \frac{1}{2}LR^2$ (IFC′). Then, for any $N \ge 1$,*

$$||\nabla f(\boldsymbol{x}_N)||^2 \le \frac{L^2R_N^2}{\tilde{\theta}_0^2 - 1} \tag{30}$$

Proof *Equation (29) consists of a weighted sum of the third and fourth inequalties of (17), scaled by $c = e = \frac{2}{\tilde{\theta}_0^2}$ in (23):*

$$\frac{c}{2L}||\nabla f(\boldsymbol{x}_N)||^2 + c(f(\boldsymbol{x}_0) - f(\boldsymbol{x}_N)) \le \frac{c}{2}LR^2. \tag{31}$$

The third inequality of (17) assumes $X_*(f) \neq \emptyset$ (see footnote 2), so we derive a bound without the above inequality. Replacing the above inequality in the weighted summation for deriving (29) by (IFC′) scaled by c, $c(f(\boldsymbol{x}_0) - f(\boldsymbol{x}_N)) \le \frac{c}{2}LR_N^2$, yields

$$\frac{1}{L}\left(1 - \frac{c}{2}\right)||\nabla f(\boldsymbol{x}_N)||^2 \le \frac{LR_N^2}{\tilde{\theta}_0^2},$$

which concludes the proof. □

The per-iteration computational complexity of the FSFOM with (22) would be expensive if implemented directly via (4), compared to GM, FGM and OGM, so the next section provides an efficient form.

6.3 An Efficient Form of the Proposed Optimized Method: OGM-G

This section develops an efficient form of FSFOM with the step coefficients (22), named OGM-G. This form is similar to that of OGM [5], which is further studied in Sect. 6.6.

> **OGM-G**
>
> Input: $f \in \mathscr{F}_L(\mathbb{R}^d)$, $\boldsymbol{x}_0 = \boldsymbol{y}_0 \in \mathbb{R}^d$, $N \ge 1$.
>
> $$\tilde{\theta}_i = \begin{cases} \frac{1+\sqrt{1+8\tilde{\theta}_{i+1}^2}}{2}, & i = 0, \\ \frac{1+\sqrt{1+4\tilde{\theta}_{i+1}^2}}{2}, & i = 1, \ldots, N-1, \\ 1, & i = N, \end{cases}$$
>
> For $i = 0, \ldots, N-1$,
>
> $$\boldsymbol{y}_{i+1} = \boldsymbol{x}_i - \frac{1}{L}\nabla f(\boldsymbol{x}_i),$$
>
> $$\boldsymbol{x}_{i+1} = \boldsymbol{y}_{i+1} + \frac{(\tilde{\theta}_i - 1)(2\tilde{\theta}_{i+1} - 1)}{\tilde{\theta}_i(2\tilde{\theta}_i - 1)}(\boldsymbol{y}_{i+1} - \boldsymbol{y}_i) + \frac{2\tilde{\theta}_{i+1} - 1}{2\tilde{\theta}_i - 1}(\boldsymbol{y}_{i+1} - \boldsymbol{x}_i).$$

Proposition 6.1 *The sequence $\{\boldsymbol{x}_0, \ldots, \boldsymbol{x}_N\}$ generated by FSFOM with (22) is identical to the corresponding sequence generated by OGM-G.*

Proof *We first show that the step coefficients $\{\tilde{h}_{i+1,k}\}$ (22) are equivalent to*

$$\tilde{h}'_{i+1,k} = \begin{cases} \frac{(\tilde{\theta}_i - 1)(2\tilde{\theta}_{i+1} - 1)}{\tilde{\theta}_i(2\tilde{\theta}_i - 1)}\tilde{h}'_{i,k}, & k = 0, \ldots, i-2, \\ \frac{(\tilde{\theta}_i - 1)(2\tilde{\theta}_{i+1} - 1)}{\tilde{\theta}_i(2\tilde{\theta}_i - 1)}(\tilde{h}'_{i,i-1} - 1), & k = i-1, \\ 1 + \frac{2\tilde{\theta}_{i+1} - 1}{\tilde{\theta}_i}, & k = i. \end{cases} \tag{32}$$

Obviously, $\tilde{h}_{i+1,i} = \tilde{h}'_{i+1,i}$, $i = 0, \ldots, N-1$, and we have

$$\begin{aligned} \tilde{h}_{i+1,i-1} &= \frac{\tilde{\theta}_i - 1}{\tilde{\theta}_{i-1}}(\tilde{h}_{i+1,i} - 1) = \frac{(\tilde{\theta}_i - 1)(2\tilde{\theta}_{i+1} - 1)}{\tilde{\theta}_{i-1}\tilde{\theta}_i} = \frac{(\tilde{\theta}_i - 1)(2\tilde{\theta}_{i+1} - 1)}{\tilde{\theta}_i(2\tilde{\theta}_i - 1)}\frac{2\tilde{\theta}_i - 1}{\tilde{\theta}_{i-1}} \\ &= \frac{(\tilde{\theta}_i - 1)(2\tilde{\theta}_{i+1} - 1)}{\tilde{\theta}_i(2\tilde{\theta}_i - 1)}(\tilde{h}'_{i,i-1} - 1) = \tilde{h}'_{i+1,i-1} \end{aligned}$$

for $i = 1, \dots, N-1$.

We next use induction by assuming $\tilde{h}_{i+1,k} = \tilde{h}'_{i+1,k}$ *for* $i = 0, \dots, n-1$, $k = 0, \dots, i$. *We then have*

$$\begin{aligned}
\tilde{h}_{n+1,k} &= \frac{\tilde{\theta}_{k+1} - 1}{\tilde{\theta}_k} \tilde{h}_{n+1,k+1} = \left(\prod_{j=k}^{n-1} \frac{\tilde{\theta}_{l+1} - 1}{\tilde{\theta}_l} \right) (\tilde{h}_{n+1,n} - 1) \\
&= \left(\prod_{j=k}^{n-2} \frac{\tilde{\theta}_{l+1} - 1}{\tilde{\theta}_l} \right) (\tilde{h}_{n,n-1} - 1) \frac{\tilde{\theta}_n - 1}{\tilde{\theta}_{n-1}} \frac{\tilde{h}_{n+1,n} - 1}{\tilde{h}_{n,n-1} - 1} \\
&= \tilde{h}_{n,k} \frac{\tilde{\theta}_n - 1}{\tilde{\theta}_{n-1}} \frac{(2\tilde{\theta}_{n+1} - 1)\tilde{\theta}_{n-1}}{\tilde{\theta}_n (2\tilde{\theta}_n - 1)} = \frac{(\tilde{\theta}_n - 1)(2\tilde{\theta}_{n+1} - 1)}{\tilde{\theta}_n (2\tilde{\theta}_n - 1)} \tilde{h}'_{n,k} = \tilde{h}'_{n+1,k}
\end{aligned}$$

for $k = 0, \dots, n-2$, *where the fourth equality uses the definition of* $\tilde{h}_{n,k}$. *This proves the first claim that the step coefficients* $\{\tilde{h}_{i+1,k}\}$ (22) *and* $\{\tilde{h}'_{i+1,k}\}$ (32) *are equivalent.*

We finally use induction to show the equivalence between the generated sequences of FSFOM with (32) *and OGM-G. For clarity, we use the notation* $\boldsymbol{x}'_0, \dots, \boldsymbol{x}'_N$ *and* $\boldsymbol{y}'_0, \dots, \boldsymbol{y}'_N$ *for OGM-G. Obviously,* $\boldsymbol{x}_0 = \boldsymbol{x}'_0$, *and we have*

$$\begin{aligned}
\boldsymbol{x}_1 &= \boldsymbol{x}_0 - \frac{1}{L} \tilde{h}'_{1,0} \nabla f(\boldsymbol{x}_0) = \boldsymbol{x}_0 - \frac{1}{L} \left(1 + \frac{2\tilde{\theta}_1 - 1}{\tilde{\theta}_0} \right) \nabla f(\boldsymbol{x}_0) \\
&= \boldsymbol{y}'_1 - \frac{1}{L} \frac{(2\tilde{\theta}_0 - 1)(2\tilde{\theta}_1 - 1)}{\tilde{\theta}_0 (2\tilde{\theta}_0 - 1)} \nabla f(\boldsymbol{x}'_0) \\
&= \boldsymbol{y}'_1 - \frac{1}{L} \left(\frac{(\tilde{\theta}_0 - 1)(2\tilde{\theta}_1 - 1)}{\tilde{\theta}_0 (2\tilde{\theta}_0 - 1)} + \frac{2\tilde{\theta}_1 - 1}{2\tilde{\theta}_0 - 1} \right) \nabla f(\boldsymbol{x}'_0) = \boldsymbol{x}'_1.
\end{aligned}$$

Assuming $\boldsymbol{x}_i = \boldsymbol{x}'_i$ *for* $i = 0, \dots, n$, *we have*

$$\begin{aligned}
\boldsymbol{x}_{n+1} &= \boldsymbol{x}_n - \frac{1}{L} \tilde{h}'_{n+1,n} \nabla f(\boldsymbol{x}_n) - \frac{1}{L} \tilde{h}'_{n+1,n-1} \nabla f(\boldsymbol{x}_{n-1}) - \frac{1}{L} \sum_{k=0}^{n-2} \tilde{h}'_{n+1,k} \nabla f(\boldsymbol{x}_k) \\
&= \boldsymbol{x}_n - \frac{1}{L} \left(1 + \frac{2\tilde{\theta}_{n+1} - 1}{\tilde{\theta}_n} \right) \nabla f(\boldsymbol{x}_n) - \frac{1}{L} \frac{(\tilde{\theta}_n - 1)(2\tilde{\theta}_{n+1} - 1)}{\tilde{\theta}_n (2\tilde{\theta}_n - 1)} (\tilde{h}_{n,n-1} - 1) \nabla f(\boldsymbol{x}_{n-1}) \\
&\quad - \frac{1}{L} \frac{(\tilde{\theta}_n - 1)(2\tilde{\theta}_{n+1} - 1)}{\tilde{\theta}_n (2\tilde{\theta}_n - 1)} \sum_{k=0}^{n-2} \tilde{h}_{n,k} \nabla f(\boldsymbol{x}_k) \\
&= \boldsymbol{x}_n - \frac{1}{L} \left(1 + \frac{2\tilde{\theta}_{n+1} - 1}{2\tilde{\theta}_n - 1} \right) \nabla f(\boldsymbol{x}_n) \\
&\quad + \frac{(\tilde{\theta}_n - 1)(2\tilde{\theta}_{n+1} - 1)}{\tilde{\theta}_n (2\tilde{\theta}_n - 1)} \left(-\frac{1}{L} \nabla f(\boldsymbol{x}_n) + \frac{1}{L} \nabla f(\boldsymbol{x}_{n-1}) - \frac{1}{L} \sum_{k=0}^{n-1} \tilde{h}_{n,k} \nabla f(\boldsymbol{x}_k) \right) \\
&= \boldsymbol{y}'_{n+1} + \frac{(\tilde{\theta}_n - 1)(2\tilde{\theta}_{n+1} - 1)}{\tilde{\theta}_n (2\tilde{\theta}_n - 1)} (\boldsymbol{y}'_{n+1} - \boldsymbol{y}'_n) + \frac{2\tilde{\theta}_{n+1} - 1}{2\tilde{\theta}_n - 1} (\boldsymbol{y}'_{n+1} - \boldsymbol{x}'_n) = \boldsymbol{x}'_{n+1}.
\end{aligned}$$

□

6.4 **Two Worst-Case Iterative Behaviors of OGM-G**

This section specifies two worst-case problem instances for OGM-G, associated with Huber and quadratic functions respectively, that make the bound (28) exact. These examples imply that the feasible point in (23) is an optimal point of (D) for OGM-G.

Lemma 6.2 *For the following Huber and quadratic functions in* $\mathscr{F}_L(\mathbb{R}^d)$*:*

$$\phi_1(\boldsymbol{x}) = \begin{cases} \frac{LR}{\tilde{\theta}_0} ||\boldsymbol{x}|| - \frac{LR^2}{2\tilde{\theta}_0^2}, & ||\boldsymbol{x}|| \geq \frac{R}{\tilde{\theta}_0}, \\ \frac{L}{2} ||\boldsymbol{x}||^2, & \text{otherwise}, \end{cases} \quad \text{and} \quad \phi_2(\boldsymbol{x}) = \frac{L}{2} ||\boldsymbol{x}||^2, \tag{33}$$

for all $d \geq 1$, *OGM-G exactly achieves the bound* (28) *with an initial point* $\boldsymbol{x}_0$ *satisfying* $\phi_1(\boldsymbol{x}_0) - \phi_{1,*} = \phi_2(\boldsymbol{x}_0) - \phi_{2,*} = \frac{1}{2} LR^2$.

Proof We first consider $\phi_1(\boldsymbol{x})$. Starting from an initial point $\boldsymbol{x}_0 = \frac{\tilde{\theta}_0^2+1}{2\tilde{\theta}_0} R\boldsymbol{\nu}$ that satisfies $\phi_1(\boldsymbol{x}_0) - \phi_{1,*} = \frac{1}{2}LR^2$ (IFC) for any unit-norm vector $\boldsymbol{\nu}$, we have

$$\boldsymbol{x}_N = \boldsymbol{x}_0 - \frac{1}{L}\sum_{j=1}^{N}\sum_{k=0}^{j-1}\tilde{h}_{j,k}\nabla f(\boldsymbol{x}_k) = \left(\frac{\tilde{\theta}_0^2+1}{2\tilde{\theta}_0} - \frac{\tilde{\theta}_0^2-1}{2\tilde{\theta}_0}\right) R\boldsymbol{\nu},$$

since

$$\sum_{j=1}^{N}\sum_{k=0}^{j-1}\tilde{h}_{j,k} = \frac{1}{2}(\tilde{\theta}_0+1) + \sum_{j=1}^{N-1}\tilde{\theta}_j = \frac{1}{2}(\tilde{\theta}_0 + 1 + 2\tilde{\theta}_1^2 - 2) = \frac{1}{2}(\tilde{\theta}_0^2 - 1)$$

that uses (26) *and* (27). *Here, all the iterates stay in the affine region of the function* $\phi_1(\boldsymbol{x})$ *with the same gradient* $\nabla\phi_1(\boldsymbol{x}) = \frac{LR}{\tilde{\theta}_0}\boldsymbol{\nu}$, $i = 0,\ldots,N$. *Therefore, after N iterations of OGM-G, we have* $||\nabla\phi_1(\boldsymbol{x}_N)||^2 = \frac{L^2R^2}{\tilde{\theta}_0^2}$.

We next consider $\phi_2(\boldsymbol{x})$. *Starting from an initial point* $\boldsymbol{x}_0 = R\boldsymbol{\nu}$ *that satisfies* $\phi_2(\boldsymbol{x}_0) - \phi_{2,*} = \frac{1}{2}LR^2$ (IFC) *for any unit-norm vector* $\boldsymbol{\nu}$, *we have*

$$\boldsymbol{x}_1 = -\frac{1}{L}\frac{2\tilde{\theta}_1 - 1}{\tilde{\theta}_0}\nabla f(\boldsymbol{x}_0) = -\frac{2\tilde{\theta}_1 - 1}{\tilde{\theta}_0}\boldsymbol{x}_0,$$

and we have

$$\boldsymbol{x}_{i+1} = -\frac{1}{L}\frac{2\tilde{\theta}_{i+1} - 1}{2\tilde{\theta}_i - 1}\nabla f(\boldsymbol{x}_i) = -\frac{2\tilde{\theta}_{i+1} - 1}{2\tilde{\theta}_i - 1}\boldsymbol{x}_i = (-1)^i\frac{2\tilde{\theta}_{i+1} - 1}{2\tilde{\theta}_1 - 1}\boldsymbol{x}_1, \;\; i = 1,\ldots,N-1,$$

using $\boldsymbol{y}_i = \boldsymbol{0}$, $i = 1,\ldots,N$. *Therefore, we have* $||\nabla\phi_2(\boldsymbol{x}_N)||^2 = L^2||\boldsymbol{x}_N||^2 = \frac{L^2R^2}{\tilde{\theta}_0^2}$. □

The iterates of OGM-G for the Huber worst-case function ϕ_1 stay in one side of the affine region of the function, while those for the quadratic worst-case function ϕ_2 always overshoot the optimum. These are extreme cases, and it is notable that some other first-order methods also have two such worst-case iterative behaviors. Specifically, in [17,25], first-order methods that have such two types of worst-case iterative behaviors in Lemma 6.2, associated with Huber and quadratic functions, respectively, were found to have an optimal worst-case bound among a certain subset of first-order methods. This leads us to conjecture that the exact worst-case bound (28) of OGM-G may be optimal, but proving it remains an open problem.

6.5 **Worst-Case Rate Behaviors of OGM-G under Initial Distance Condition**

This section further studies the worst-case rate behaviors of OGM-G under initial distance condition (IDC). Table 2 presents exact numerical worst-case rates of OGM-G (under a large-dimensional condition), using the performance estimation toolbox, named PESTO[3] [28], based on PEP [1, 17].

Table 2 Exact values of the reciprocals of the worst-case cost function inaccuracy $\left(\frac{L\overline{R}^2}{f(\boldsymbol{x}_N)-f(\boldsymbol{x}_*)}\right)$ in (5) and the worst-case gradient inaccuracy $\left(\frac{L^2\overline{R}^2}{||\nabla f(\boldsymbol{x}_N)||^2}, \frac{L^2R^2}{||\nabla f(\boldsymbol{x}_N)||^2}, \text{ or } \frac{L^2R_N^2}{||\nabla f(\boldsymbol{x}_N)||^2}\right)$ in (3) of OGM-G under one of the conditions (IDC), (IFC) or (IFC′).

OGM-G Efficiency	Initial cond.	Number of iterations							
		1	2	4	10	20	30	40	50
Cost func. (5)	(IDC)	8.0	10.0	9.7	8.9	8.5	8.3	8.3	8.2
Gradient (3)	(IDC)	4.0	8.1	19.5	79.5	262.5	547.8	934.6	1422.6
	(IFC)	4.0	8.1	19.5	79.5	262.5	547.8	934.6	1422.6
	(IFC′)	3.0	7.1	18.5	78.5	261.5	546.8	933.6	1421.6

Table 2 illustrates that the worst-case gradient rates of OGM-G are equivalent numerically under both (IDC) and (IFC). This is because the worst-case problem instance of OGM-G in Lemma 6.2 associated with the quadratic function under (IFC) also serves as a worst-case of OGM-G under (IDC), as formally discussed next.

[3] In PESTO toolbox [28], we used the SDP solver SeDuMi [26] interfaced through Yalmip [27]. The OGM-G method is implemented in the PESTO toolbox.

Corollary 6.2 *Let $\boldsymbol{x}_0,\dots,\boldsymbol{x}_N \in \mathbb{R}^d$ be generated by OGM-G. Then, for any $N \geq 1$,*

$$\frac{L^2\overline{R}^2}{\tilde{\theta}_0^2} \leq \min_{\substack{f\in\mathscr{F}_L(\mathbb{R}^d),\\ \boldsymbol{x}_*\in X_*(f),\\ ||\boldsymbol{x}_0-\boldsymbol{x}_*||\leq\overline{R}}} ||\nabla f(\boldsymbol{x}_N)||^2. \tag{34}$$

Proof *Consider the quadratic function $\phi_2(\boldsymbol{x}) = \frac{L}{2}||\boldsymbol{x}||^2$ in Lemma 6.2 associated with the initial point $\boldsymbol{x}_0 = R\boldsymbol{\nu}$ for any unit-norm vector $\boldsymbol{\nu}$. This initial point $\boldsymbol{x}_0$ satisfies $||\boldsymbol{x}_0 - \boldsymbol{x}_*|| = R$ as well as $\phi_2(\boldsymbol{x}_0) - \phi_{2,*} = \frac{1}{2}LR^2$, which implies the inequality (34) based on Lemma 6.2.* □

We conjecture that the lower bound (34) of OGM-G under (IDC) is exact, based on numerical evidence in Table 2. This is a bit disappointing, because it appears that a method that is optimal under one initial condition is far from optimal for another initial condition. It is also unfortunate that OGM-G has a poor worst-case rate for decreasing the cost function under (IDC). An open problem is finding a method that achieves optimal rates invariant to worst-case rate measures and initial conditions.

In addition, we study how the worst-case rate under (IFC) transfers to that under (IDC) for given problem instance $(f,\boldsymbol{x}_0)$. We particularly focus on two worst-case problem instances of OGM-G in Lemma 6.2, while similar analysis can be done for the worst-case problem instance of GM in Lemma 5.2. For the worst-case of OGM-G associated with the Huber function $\phi_1(\boldsymbol{x})$, the constants R and $\overline{R}$ in (IFC) and (IDC) have the following relationship:

$$\overline{R} = ||\boldsymbol{x}_0 - \boldsymbol{x}_*|| = \frac{\tilde{\theta}_0^2+1}{2\tilde{\theta}_0}R \geq \frac{\tilde{\theta}_0}{2}R \geq \frac{N+1}{2\sqrt{2}}R. \tag{35}$$

We can then show the following upper bound associated with $\overline{R}$ after N iterations of OGM-G:

$$||\nabla\phi_1(\boldsymbol{x}_N)||^2 = \frac{L^2R^2}{\tilde{\theta}_0^2} \leq \frac{2L^2R^2}{(N+1)^2} \leq \frac{16L^2\overline{R}^2}{(N+1)^4}, \tag{36}$$

yielding $O(1/N^4)$, instead of the OGM-G rate $O(1/N^2)$, expressed by using $\overline{R}$ instead of R. On the other hand, for the worst-case of OGM-G associated with the quadratic function $\phi_2(\boldsymbol{x})$ in Lemma 6.2, we have the relationship $R = \overline{R}$, as mentioned in Corollary 6.2. These examples illustrate that comparing the worst-case rates under different initial conditions is subtle, and it would be incomplete to treat R and $\overline{R}$ as just arbitrary constants (unrelated to N) in Table 1.

6.6 Related Work: OGM

This section shows that the proposed OGM-G has a close relationship with the following OGM [5] (that was numerically first identified in [1]).

OGM [5]

Input: $f \in \mathscr{F}_L(\mathbb{R}^d)$, $\boldsymbol{x}_0 = \boldsymbol{y}_0 \in \mathbb{R}^d$, $N \geq 1$, $\hat{\theta}_0 = 1$.

For $i = 0,\dots,N-1$,

$$\begin{aligned}
\boldsymbol{y}_{i+1} &= \boldsymbol{x}_i - \frac{1}{L}\nabla f(\boldsymbol{x}_i),\\
\hat{\theta}_{i+1} &= \begin{cases} \frac{1+\sqrt{1+4\hat{\theta}_i^2}}{2}, & i < N-1,\\ \frac{1+\sqrt{1+8\hat{\theta}_i^2}}{2}, & i = N-1,\end{cases}\\
\boldsymbol{x}_{i+1} &= \boldsymbol{y}_{i+1} + \frac{\hat{\theta}_i - 1}{\hat{\theta}_{i+1}}(\boldsymbol{y}_{i+1} - \boldsymbol{y}_i) + \frac{\hat{\theta}_i}{\hat{\theta}_{i+1}}(\boldsymbol{y}_{i+1} - \boldsymbol{x}_i).
\end{aligned} \tag{37}$$

We can easily notice the symmetric relationship of the parameters

$$\hat{\theta}_i = \tilde{\theta}_{N-i}, \quad i = 0,\dots,N, \tag{38}$$

and the fact that OGM and OGM-G have forms that differ in the coefficients of the terms $\boldsymbol{y}_{i+1} - \boldsymbol{y}_i$ and $\boldsymbol{y}_{i+1} - \boldsymbol{x}_i$.

For $f \in \mathscr{F}_L(\mathbb{R}^d)$, $\boldsymbol{x}_* \in X_*(f)$ and $||\boldsymbol{x}_0 - \boldsymbol{x}_*|| \leq \overline{R}$ (IDC), the final Nth iterate $\boldsymbol{x}_N$ of OGM has the following exact worst-case cost function bound [5, Theorems 2 and 3]:

$$f(\boldsymbol{x}_N) - f(\boldsymbol{x}_*) \leq \frac{L\overline{R}^2}{2\hat{\theta}_N^2} \leq \frac{L\overline{R}^2}{(N+1)^2}, \tag{39}$$

where this exact upper bound is equivalent to the exact worst-case gradient bound (28) of OGM-G up to a constant $\frac{\overline{R}^2}{2LR^2}$. This equivalence is similar to the relationship between the exact worst-case bounds (16) and (20) of GM discussed in Remark 5.1. The worst-case rate (39) of OGM is exactly optimal for large-dimensional smooth convex minimization [6].

OGM is equivalent to FSFOM with the step coefficients [5, Proposition 4]:

$$\hat{h}_{i+1,k} = \begin{cases} \frac{\hat{\theta}_i - 1}{\hat{\theta}_{i+1}} \hat{h}_{i,k}, & k = 0, \dots, i-2, \\ \frac{\hat{\theta}_i - 1}{\hat{\theta}_{i+1}} (\hat{h}_{i,i-1} - 1), & k = i-1, \\ 1 + \frac{2\hat{\theta}_i - 1}{\hat{\theta}_{i+1}}, & k = i. \end{cases} \tag{40}$$

for $i = 0, \dots, N-1$. The following proposition shows the symmetric relationship between the step coefficients $\{\hat{h}_{i+1,k}\}$ (40) and $\{\tilde{h}_{i+1,k}\}$ (22) of OGM and OGM-G, respectively.

Proposition 6.2 *The step coefficients* $\{\hat{h}_{i+1,k}\}$ (40) *and* $\{\tilde{h}_{i+1,k}\}$ (22) *of OGM and OGM-G, respectively, have the following relationship*

$$\hat{h}_{i+1,k} = \tilde{h}_{N-k,N-i-1}, \quad i = 0, \dots, N-1, \; k = 0, \dots, i. \tag{41}$$

Proof *We use induction. Obviously,* $\hat{h}_{1,0} = \tilde{h}_{N,N-1}$. *Then, assuming* $\hat{h}_{i+1,k} = \tilde{h}_{N-k,N-i-1}$ *for* $i = 0, \dots, n-1$, *we have*

$$\begin{aligned} \hat{h}_{n+1,k} &= \begin{cases} \frac{\tilde{\theta}_{N-n} - 1}{\tilde{\theta}_{N-n-1}} \tilde{h}_{N-k,N-n}, & k = 0, \dots, n-2, \\ \frac{\tilde{\theta}_{N-n} - 1}{\tilde{\theta}_{N-n-1}} (\tilde{h}_{N-n+1,N-n} - 1), & k = n-1, \\ 1 + \frac{2\tilde{\theta}_{N-n} - 1}{\tilde{\theta}_{N-n-1}}, & k = n, \end{cases} \\ &= \tilde{h}_{N-k,N-n-1}. \end{aligned}$$

□

Building upon the relationships (38) and (41) between OGM and OGM-G, we numerically study the momentum coefficient values β_i and γ_i of OGM and OGM-G in the following form, to characterize the convergence behaviors of the methods.

Accelerated First-Order Method

Input: $f \in \mathscr{F}_L(\mathbb{R}^d)$, $\boldsymbol{x}_0 = \boldsymbol{y}_0 \in \mathbb{R}^d$, $N \geq 1$.

For $i = 0, \dots, N-1$,

$$\boldsymbol{y}_{i+1} = \boldsymbol{x}_i - \frac{1}{L} \nabla f(\boldsymbol{x}_i),$$

$$\boldsymbol{x}_{i+1} = \boldsymbol{y}_{i+1} + \beta_i (\boldsymbol{y}_{i+1} - \boldsymbol{y}_i) + \gamma_i (\boldsymbol{y}_{i+1} - \boldsymbol{x}_i).$$

Figure 1 compares the momentum coefficients (β_i, γ_i) of OGM and OGM-G for $N = 100$. It is notable that having *increasing* values of (β_i, γ_i) as i increases, except for the last iteration, yields the optimal (fast) worst-case rate for decreasing the cost function, whereas having *decreasing* values of (β_i, γ_i), except for the first iteration, yields the fast worst-case rate (that is optimal up to a constant) for decreasing the gradient. We leave further theoretical study on such choices of coefficients as future work.

We next compare OGM and OGM-G with their other equivalent efficient forms. Similar to [5, Algorithm OGM2] one can easily show that the last line of OGM is equivalent to

$$\begin{cases} \boldsymbol{z}_{i+1} = \boldsymbol{y}_{i+1} + (\hat{\theta}_i - 1)(\boldsymbol{y}_{i+1} - \boldsymbol{y}_i) + \hat{\theta}_i (\boldsymbol{y}_{i+1} - \boldsymbol{x}_i), \\ \boldsymbol{x}_{i+1} = \left(1 - \frac{1}{\hat{\theta}_{i+1}}\right) \boldsymbol{y}_{i+1} + \frac{1}{\hat{\theta}_{i+1}} \boldsymbol{z}_{i+1}, \end{cases} \tag{42}$$

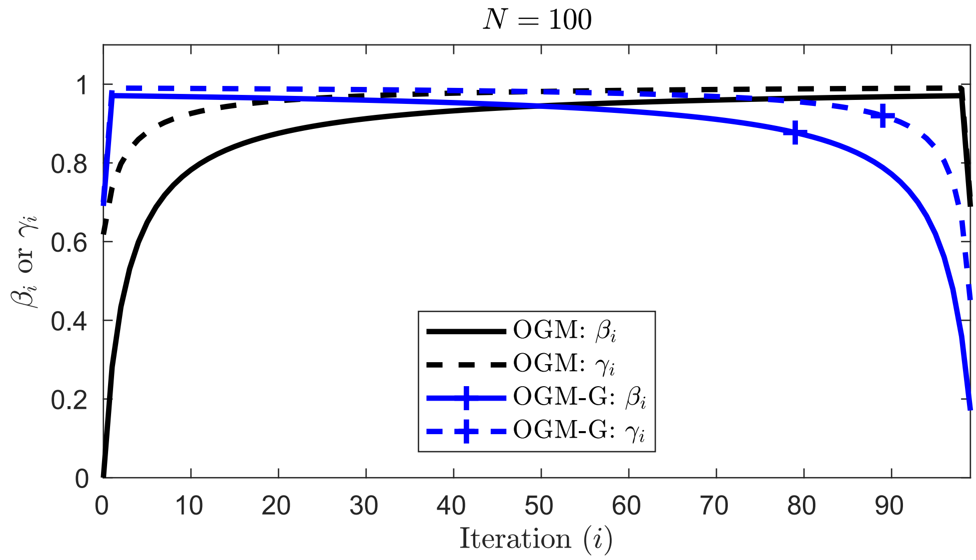

Fig. 1 Comparison of momentum coefficients (β_i,γ_i) of OGM and OGM-G.

while that of OGM-G is equivalent to

$$\begin{cases} \boldsymbol{z}_{i+1} = \boldsymbol{y}_{i+1} + (\tilde{\theta}_i - 1)(\boldsymbol{y}_{i+1} - \boldsymbol{y}_i) + \tilde{\theta}_i(\boldsymbol{y}_{i+1} - \boldsymbol{x}_i), \\ \boldsymbol{x}_{i+1} = \left(1 - \frac{2\tilde{\theta}_{i+1}-1}{\tilde{\theta}_i(2\tilde{\theta}_i-1)}\right)\boldsymbol{y}_{i+1} + \frac{2\tilde{\theta}_{i+1}-1}{\tilde{\theta}_i(2\tilde{\theta}_i-1)}\boldsymbol{z}_{i+1}. \end{cases} \tag{43}$$

This interpretation stems from a variant of FGM [29] that involves a convex combination of two points as above. [5] already showed that similar interpretation is possible for OGM, and the expression here also implies that decreasing gradient can be achieved via some convex combination of two points. Further analysis is left as future work.

7 Conclusions

This paper developed a first-order method named OGM-G that has an inexpensive per-iteration computational complexity and achieves the optimal worst-case bound for decreasing the gradient of large-dimensional smooth convex functions up to a constant, under the initial bounded function condition. A simple method in [8], using the OGM-G, also achieves the optimal worst-case gradient bound up to a constant, under the initial bounded distance condition. The OGM-G was derived by optimizing the step coefficients of first-order methods in terms of the worst-case gradient bound using the performance estimation problem (PEP) approach [1]. On the way, the exact worst-case gradient bound for a gradient method was studied.

A practical drawback of OGM-G is that one must choose the number of iterations N in advance. Finding a first-order method that achieves the optimal worst-case gradient bound (up to a constant), but that does not depend on selecting N in advance, remains an open problem. In addition, extending the approaches based on PEP in this paper to the initial bounded distance condition (IDC) will be interesting future work; this PEP approach with a strict relaxation (unlike this paper) has been studied in [14]. Further extensions of this paper to nonconvex problems and composite problems are also of interest.

Acknowledgements Part of this work was carried through while the first author was affiliated with the University of Michigan. The first author would like to thank Ernest K. Ryu for pointing out related references. The authors would like to thank associate editor and referees for useful comments, especially regarding the case where a finite minimizer does not exist. The first author was supported in part by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (No. 2019R1A5A1028324), and the POSCO Science Fellowship of POSCO TJ Park Foundation. The second author was supported in part by NSF grant IIS 1838179.

Appendix: Proof of Eqs. (25) and (26)

This proof shows the properties (25) and (26) of the step coefficients $\{\tilde{h}_{i,j}\}$ (22).

We first show (25). We can easily derive

$$\tilde{h}_{i,i-2} = \frac{(\tilde{\theta}_{i-1}-1)(2\tilde{\theta}_i-1)}{\tilde{\theta}_{i-2}\tilde{\theta}_{i-1}} = \frac{\tilde{\theta}_i^2(2\tilde{\theta}_i-1)}{\tilde{\theta}_{i-2}\tilde{\theta}_{i-1}^2}$$

for $i=2,\dots,N$ using (27). Again using the definition of (22) and (27), we have

$$\begin{aligned}
\tilde{h}_{i,j} &= \frac{\tilde{\theta}_{j+1}-1}{\tilde{\theta}_j}\tilde{h}_{i,j+1} = \cdots = \left(\prod_{l=j+1}^{i-2}\frac{\tilde{\theta}_l-1}{\tilde{\theta}_{l-1}}\right)\tilde{h}_{i,i-2} = \left(\prod_{l=j+1}^{i-1}\frac{\tilde{\theta}_l-1}{\tilde{\theta}_{l-1}}\right)\frac{2\tilde{\theta}_i-1}{\tilde{\theta}_{i-1}} \\
&= \frac{1}{\tilde{\theta}_j}\frac{1}{\tilde{\theta}_{j+1}}\frac{\tilde{\theta}_{j+1}-1}{\tilde{\theta}_{j+2}}\cdots\frac{\tilde{\theta}_{i-3}-1}{\tilde{\theta}_{i-2}}(\tilde{\theta}_{i-2}-1)(\tilde{\theta}_{i-1}-1)\frac{2\tilde{\theta}_i-1}{\tilde{\theta}_{i-1}} \\
&= \frac{1}{\tilde{\theta}_j}\frac{1}{\tilde{\theta}_{j+1}}\frac{\tilde{\theta}_{j+2}}{\tilde{\theta}_{j+1}}\cdots\frac{\tilde{\theta}_{i-2}}{\tilde{\theta}_{i-3}}(\tilde{\theta}_{i-2}-1)(\tilde{\theta}_{i-1}-1)\frac{2\tilde{\theta}_i-1}{\tilde{\theta}_{i-1}} \\
&= \frac{\tilde{\theta}_{i-2}(\tilde{\theta}_{i-2}-1)(\tilde{\theta}_{i-1}-1)(2\tilde{\theta}_i-1)}{\tilde{\theta}_j\tilde{\theta}_{j+1}^2\tilde{\theta}_{i-1}} = \frac{\tilde{\theta}_i^2(2\tilde{\theta}_i-1)}{\tilde{\theta}_j\tilde{\theta}_{j+1}^2},
\end{aligned}$$

for $i=2,\dots,N,\ j=0,\dots,i-3$, which concludes the proof of (25).

We next prove the first two lines of (26) using the induction. For $N=1$, we have $\tilde{\theta}_1=1$ and

$$\tilde{h}_{1,0} = 1+\frac{2\tilde{\theta}_1-1}{\tilde{\theta}_0} = 1+\frac{\tilde{\theta}_1^2}{\tilde{\theta}_0} = 1+\frac{\frac{1}{2}(\tilde{\theta}_0^2-\tilde{\theta}_0)}{\tilde{\theta}_0} = \frac{1}{2}(\tilde{\theta}_0+1),$$

where the third equality uses (27). For $N>1$, we have

$$\tilde{h}_{N,N-1} = 1+\frac{2\tilde{\theta}_N-1}{\tilde{\theta}_{N-1}} = 1+\frac{\tilde{\theta}_N^2}{\tilde{\theta}_{N-1}} = 1+\frac{\tilde{\theta}_{N-1}^2-\tilde{\theta}_{N-1}}{\tilde{\theta}_{N-1}} = \tilde{\theta}_{N-1},$$

where the third equality uses (27). Assuming $\sum_{l=j+1}^N \tilde{h}_{l,j} = \tilde{\theta}_j$ for $j=n,\dots,N-1$ and $n\ge 1$, we get

$$\begin{aligned}
\sum_{l=n}^N \tilde{h}_{l,n-1} &= 1+\frac{2\tilde{\theta}_n-1}{\tilde{\theta}_{n-1}}+\frac{\tilde{\theta}_n-1}{\tilde{\theta}_{n-1}}(\tilde{h}_{n+1,n}-1)+\frac{\tilde{\theta}_n-1}{\tilde{\theta}_{n-1}}\sum_{l=n+2}^N \tilde{h}_{l,n} \\
&= 1+\frac{\tilde{\theta}_n}{\tilde{\theta}_{n-1}}+\frac{\tilde{\theta}_n-1}{\tilde{\theta}_{n-1}}\sum_{l=n+1}^N \tilde{h}_{l,n} = \frac{\tilde{\theta}_{n-1}+\tilde{\theta}_n+(\tilde{\theta}_n-1)\tilde{\theta}_n}{\tilde{\theta}_{n-1}} = \frac{\tilde{\theta}_{n-1}+\tilde{\theta}_n^2}{\tilde{\theta}_{n-1}} \\
&= \begin{cases} \frac{1}{2}(\tilde{\theta}_0+1), & n=0, \\ \tilde{\theta}_n, & n=1,\dots,N-1, \end{cases}
\end{aligned}$$

where the last equality uses (27), which concludes the proof of the first two lines of (26).

We finally prove the last line of (26) using the induction. For $i\ge 1$, we have

$$\sum_{l=i+1}^N \tilde{h}_{l,i-1} = \sum_{l=i}^N \tilde{h}_{l,i-1} - \tilde{h}_{i,i-1} = \tilde{\theta}_{i-1} - \left(1+\frac{2\tilde{\theta}_i-1}{\tilde{\theta}_{i-1}}\right) = \frac{(\tilde{\theta}_i-1)^2}{\tilde{\theta}_{i-1}} = \frac{\tilde{\theta}_{i+1}^4}{\tilde{\theta}_{i-1}\tilde{\theta}_i^2},$$

where the third and fourth equalities use (27). Then, assuming $\sum_{l=i+1}^N \tilde{h}_{l,j} = \frac{\tilde{\theta}_i^4}{\tilde{\theta}_j\tilde{\theta}_{j+1}^2}$ for $i=n,\dots,N-1,\ j=0,\dots,i-1$ with $n\ge 1$, we get:

$$\sum_{l=n}^N \tilde{h}_{l,j} = \sum_{l=n+1}^N \tilde{h}_{l,j} + \tilde{h}_{n,j} = \frac{\tilde{\theta}_{n+1}^4}{\tilde{\theta}_j\tilde{\theta}_{j+1}^2} + \frac{\tilde{\theta}_n^2(2\tilde{\theta}_n-1)}{\tilde{\theta}_j\tilde{\theta}_{j+1}^2} = \frac{\tilde{\theta}_n^2(\tilde{\theta}_n-1)^2+\tilde{\theta}_n^2(2\tilde{\theta}_n-1)}{\tilde{\theta}_j\tilde{\theta}_{j+1}^2} = \frac{\tilde{\theta}_n^4}{\tilde{\theta}_j\tilde{\theta}_{j+1}^2},$$

where the second and third equalities use (25), which concludes the proof. □

References

1. Drori, Y., Teboulle, M.: Performance of first-order methods for smooth convex minimization: A novel approach. Mathematical Programming **145**(1-2), 451–82 (2014). DOI 10.1007/s10107-013-0653-0
2. Nesterov, Y.: A method for unconstrained convex minimization problem with the rate of convergence $O(1/k^2)$. Dokl. Akad. Nauk. USSR **269**(3), 543–7 (1983)
3. Nesterov, Y.: Introductory lectures on convex optimization: A basic course. Kluwer (2004). DOI 10.1007/978-1-4419-8853-9
4. Nemirovsky, A.S.: Information-based complexity of linear operator equations. J. of Complexity **8**(2), 153–75 (1992). DOI 10.1016/0885-064X(92)90013-2
5. Kim, D., Fessler, J.A.: Optimized first-order methods for smooth convex minimization. Mathematical Programming **159**(1), 81–107 (2016). DOI 10.1007/s10107-015-0949-3
6. Drori, Y.: The exact information-based complexity of smooth convex minimization. J. Complexity **39**, 1–16 (2017). DOI 10.1016/j.jco.2016.11.001

7. Kim, D., Fessler, J.A.: Optimizing the efficiency of first-order methods for decreasing the gradient of smooth convex functions (2018). URL http://arxiv.org/abs/1803.06600
8. Nesterov, Y., Gasnikov, A., Guminov, S., Dvurechensky, P.: Primal-dual accelerated gradient methods with small-dimensional relaxation oracle. Optim. Meth. Software (2020). DOI 10.1080/10556788.2020.1731747
9. Nesterov, Y.: How to make the gradients small (2012). URL http://www.mathopt.org/?nav=optima_newsletter. Optima 88
10. Allen-Zhu, Z.: How to make the gradients small stochastically: Even faster convex and nonconvex SGD. In: NIPS (2018)
11. Drori, Y., Shamir, O.: The complexity of finding stationary points with stochastic gradient descent. In: ICML (2020)
12. Carmon, Y., Duchi, J.C., Hinder, O., Sidford, A.: Lower bounds for finding stationary points II: First-order methods. Mathematical Programming (2019). DOI 10.1007/s10107-019-01431-x
13. Kim, D., Fessler, J.A.: Another look at the Fast Iterative Shrinkage/Thresholding Algorithm (FISTA). SIAM J. Optim. **28**(1), 223–50 (2018). DOI 10.1137/16M108940X
14. Kim, D., Fessler, J.A.: Generalizing the optimized gradient method for smooth convex minimization. SIAM J. Optim. **28**(2), 1920–50 (2018). DOI 10.1137/17m112124x
15. Ghadimi, S., Lan, G.: Accelerated gradient methods for nonconvex nonlinear and stochastic programming. Mathematical Programming **156**(1), 59–99 (2016). DOI 10.1007/s10107-015-0871-8
16. Monteiro, R.D.C., Svaiter, B.F.: An accelerated hybrid proximal extragradient method for convex optimization and its implications to second-order methods. SIAM J. Optim. **23**(2), 1092–1125 (2013). DOI 10.1137/110833786
17. Taylor, A.B., Hendrickx, J.M., Glineur, F.: Smooth strongly convex interpolation and exact worst-case performance of first- order methods. Mathematical Programming **161**(1), 307–45 (2017). DOI 10.1007/s10107-016-1009-3
18. Nacson, M.S., Lee, J.D., Gunasekar, S., Savarese, P.H.P., Srebro, N., Soudry, D.: Convergence of gradient descent on separable data. In: AISTATS (2019)
19. Soudry, D., Hoffer, E., Nacson, M.S., Srebro, N.: The implicit bias of gradient descent on separable data. In: Proc. Intl. Conf. on Learning Representations (2018)
20. Beck, A., Teboulle, M.: A fast iterative shrinkage-thresholding algorithm for linear inverse problems. SIAM J. Imaging Sci. **2**(1), 183–202 (2009). DOI 10.1137/080716542
21. Drori, Y., Taylor, A.B.: Efficient first-order methods for convex minimization: a constructive approach. Mathematical Programming (2019). DOI 10.1007/s10107-019-01410-2
22. Taylor, A.B., Hendrickx, J.M., Glineur, F.: Exact worst-case convergence rates of the proximal gradient method for composite convex minimization. J. Optim. Theory Appl. **178**(2), 455–76 (2018)
23. CVX Research Inc.: CVX: Matlab software for disciplined convex programming, version 2.0. http://cvxr.com/cvx (2012)
24. Grant, M., Boyd, S.: Graph implementations for nonsmooth convex programs. In: V. Blondel, S. Boyd, H. Kimura (eds.) Recent Advances in Learning and Control, Lecture Notes in Control and Information Sciences, pp. 95–110. Springer-Verlag Limited (2008). http://stanford.edu/~boyd/graph_dcp.html
25. Kim, D., Fessler, J.A.: On the convergence analysis of the optimized gradient methods. J. Optim. Theory Appl. **172**(1), 187–205 (2017). DOI 10.1007/s10957-016-1018-7
26. Sturm, J.: Using SeDuMi 1.02, A MATLAB toolbox for optimization over symmetric cones. Optim. Meth. Software **11**(1), 625–53 (1999). DOI 10.1080/10556789908805766
27. Löfberg, J.: YALMIP: A toolbox for modeling and optimization in MATLAB. In: Proc. of the CACSD Conference (2004)
28. Taylor, A.B., Hendrickx, J.M., Glineur, F.: Performance Estimation Toolbox (PESTO): automated worst-case analysis of first-order optimization methods. In: Proc. Conf. Decision and Control, pp. 1278–83 (2017). DOI 10.1109/CDC.2017.8263832
29. Nesterov, Y.: Smooth minimization of non-smooth functions. Mathematical Programming **103**(1), 127–52 (2005). DOI 10.1007/s10107-004-0552-5