Aalto University publication series
DOCTORAL THESES 166/2024

Generalized Accelerated Optimization Framework for Big Data Processing

Endrit Dosti

A doctoral thesis completed for the degree of Doctor of Science (Technology) to be defended, with the permission of the Aalto University School of Electrical Engineering, at a public examination held at the lecture hall R001/Y124 Hall E of the school on 6 September 2024 at 12:00.

Aalto University
School of Electrical Engineering
Department of Information and Communications Engineering

Supervising professor
Prof. Sergiy A. Vorobyov, Aalto University, Finland

Thesis advisor
Prof. Themistoklis Charalambous, University of Cyprus, Cyprus

Preliminary examiners
Prof. Jean-Christophe Pesquet, University of Paris-Saclay, France
Prof. Panagiotis Patrinos, KU Leuven, Belgium

Opponent
Prof. Jean-Christophe Pesquet, University of Paris-Saclay, France

Aalto University publication series
DOCTORAL THESES 166/2024

ISBN 978-952-64-1968-8 (printed)
ISBN 978-952-64-1969-5 (pdf)
ISSN 1799-4934 (printed)
ISSN 1799-4942 (pdf)
http://urn.fi/URN:ISBN:978-952-64-1969-5

Unigrafia Oy
Helsinki 2024

Finland

NORDIC SWAN ECOLABEL
Printed matter
4041-0619

Author
Endrit Dosti

Name of the doctoral thesis
Generalized Accelerated Optimization Framework for Big Data Processing

Publisher School of Electrical Engineering

Unit Department of Information and Communications Engineering

Series Aalto University publication series DOCTORAL THESES 166/2024

Field of research Signal Processing Technology

Manuscript submitted 26 April 2024 **Date of the defence** 6 September 2024

Permission for public defence granted (date) 24 June 2024 **Language** English

☐ **Monograph** ☒ **Article thesis** ☐ **Essay thesis**

Abstract

Large-scale optimization problems arise in different fields of engineering and science. Due to the large number of parameters and different structures that these problems can have, black-box first-order methods are widely used in solving them. Among the existing first-order methods, the ones that are most widely used are different variants of Fast Gradient Methods (FGM). Such methods are devised in the context of the estimating sequences framework and exhibit desirable properties such as fast convergence rate and low per iteration complexity. In this Thesis, we devise new estimating sequences and show that they can be used to construct accelerated first-order methods. We start by considering the simplest case, i.e., minimizing smooth and convex objective functions. For this class of problems we present a class of generalized estimating sequences, constructed by exploiting the history of the estimating functions that are obtained during the minimization process. Using these generalized estimating sequences, we devise a new accelerated gradient method and prove that it can converge to an ϵ neighborhood of the optimal solution in at most $\sqrt{\frac{\kappa}{2}}\left(\ln\frac{1}{\epsilon}+\mathcal{O}(1)\right)$ iterations. We then consider a more general class of optimization problems, namely composite objectives. For this class of problems, we introduce the class of composite estimating sequences, which are obtained by making use of the gradient mapping framework and a tight lower bound on the function that should be minimized. Using these composite estimating sequences, we devise a composite objective accelerated multi-step estimating sequence technique, and prove its accelerated convergence rate. Last, embedding the memory term coming from the previous iterates into the composite estimating sequences, we obtain the generalized composite estimating sequences. Using these estimating sequences, we construct another accelerated gradient method and prove its accelerated convergence rate. The methods devised for solving composite objective functions that we introduce in this thesis are also equipped with efficient backtracking line-search strategies, which enable more accurate estimates of the step-size. Our results are validated by a large number of computational experiments on different types of loss functions, wherein both simulated and publicly available real-world datasets are considered. Our numerical experiments also highlight the robustness of our newly introduced methods to the usage of inexact values for of the Lipschitz constant and the strong convexity parameter.

Keywords

ISBN (printed) 978-952-64-1968-8 **ISBN (pdf)** 978-952-64-1969-5

ISSN (printed) 1799-4934 **ISSN (pdf)** 1799-4942

Location of publisher Helsinki **Location of printing** Helsinki **Year** 2024

Pages 173 **urn** http://urn.fi/URN:ISBN:978-952-64-1969-5

Preface

My sincere gratitude goes to Prof. Sergiy A. Vorobyov, my supervisor, whose mentorship and unceasing support have been indispensable throughout my doctoral studies. I would also like to extend my utmost appreciation to Prof. Themistoklis Charalambous, my thesis advisor, for his help and endless guidance throughout this journey.

I would like to express my deepest appreciation to Prof. Jean-Christophe Pesquet for his invaluable contributions as pre-examiner and opponent. Furthermore I wish to express my gratitude to Prof. Panagiotis Patrinos the contributions as pre-examiner of my thesis. Their insightful feedback has been crucial to the development of this thesis.

I would like to thank the School of Electrical Engineering in Aalto University for giving me the opportunity to work in one of the best research environments in the world. Many thanks also for supporting me financially through the system of incentive scholarships from the start of my doctoral studies. I am also grateful to the Nokia Foundation and HPY Research Foundation for supporting my studies by granting me their incentive scholarships.

I would also like to thank the colleagues in Aalto University for their friendship and support throughout this journey. I would also like to extend my gratitude to the colleagues in Huawei and Nokia who helped me learn how to put my knowledge into practice.

Last, but not least, I would like to thank my parents for their unconditional love and for always believing in me.

Espoo, August 6, 2024,

Endrit Dosti

Contents

List of Publications

This thesis consists of an overview and of the following publications which are referred to in the text by their Roman numerals.

I E. Dosti, S. A. Vorobyov, T. Charalambous. Generalizing Nesterov's Acceleration Framework by Embedding Momentum Into Estimating Sequences: New Algorithm and Bounds. In *IEEE International Symposium on Information Theory (ISIT)*, Helsinki, Finland, 1506-1511, June 2022.

II E. Dosti, S. A. Vorobyov, T. Charalambous. Embedding a Heavy-Ball type of Momentum into the Estimating Sequences. *Journal Submission*, March 2024.

III E. Dosti, S. A. Vorobyov, T. Charalambous. A new accelerated gradient-based estimating sequence technique for solving large-scale optimization problems with composite structure. In *IEEE 61st Conference on Decision and Control (CDC)*, Cancun, Mexico, 7516-7521, January 2023.

IV E. Dosti, S. A. Vorobyov, T. Charalambous. A new class of composite objective multistep estimating sequence techniques. *Signal Processing*, 206, 108889, December 2022.

V E. Dosti, S. A. Vorobyov, T. Charalambous. Generalizing the estimating sequences with memory terms for minimizing convex composite functions. *Journal Submission*, March 2024.

Author's Contribution

Publication I: "Generalizing Nesterov's Acceleration Framework by Embedding Momentum Into Estimating Sequences: New Algorithm and Bounds"

The author formulated the problem, derived the proofs and algorithm, performed the numerical simulations and was the main writer of the article while also incorporating comments from other co-authors.

Publication II: "Embedding a Heavy-Ball type of Momentum into the Estimating Sequences"

The author proposed the idea, developed the theoretical framework, implemented the computational experiments and lead the work on drafting the manuscript.

Publication III: "A new accelerated gradient-based estimating sequence technique for solving large-scale optimization problems with composite structure"

The author developed the algorithm, computational experiments and was the main driver of writing the article.

Publication IV: "A new class of composite objective multistep estimating sequence techniques"

The author suggested the problem, established the theoretical results, developed the simulations and lead the preparation of the manuscript.

Publication V: "Generalizing the estimating sequences with memory terms for minimizing convex composite functions"

The author devised the algorithm, computational experiments and wrote the majority of the article.

Abbreviations

AGM Accelerated Gradient Method

AMGS Accelerated Multistep Gradient Scheme

COMET Composite Objective Multistep Estimating Sequence Technique

CSS1 Constant Step Scheme 1

FGM Fast Gradient Method

FISTA Fast Iterative Shrinkage Thresholding

GDM Geometric Descent Method

LHS Left Hand Side

ODE Ordinary Differential Equations

RHS Right Hand Side

SDP SemiDefinite Program

TMM Triple Momentum Method

Symbols

f Smooth and convex cost function

f^* Minimal value of the smooth and convex cost function

k Iteration counter

L Lipschitz constant

$\ln$ Natural logarithm

$m_L(y; x)$ Upper bound on the composite objective function

R_0 Euclidean distance from the first iterate to the optimal solution

$r_L(y)$ Reduced composite gradient

s_L Subgradient of the non-smooth term of the convex composite objective function

span Linear span of vectors

$T_L(y)$ Composite gradient mapping

v_k Center of the scanning function

x Optimization variable

x^* Value of the optimization variable that yields minimal value of f

$x_{\Phi_k^*}$ Minimizer of the estimating function

y Real vector

$\mathcal{I}$ Set containing the span of all the iterates

$\mathcal{O}$ Big O notation

$\mathcal{Q}$ Subset of real numbers containing nonnegative values

$\mathcal{R}$ Set of real numbers

α Parameter of the estimating function

$\beta_{i,k}$ Momentum coefficients

γ_k Radius of the scanning function

ϵ Accuracy tolerance

η_u Parameter controlling the increase of the step-size estimate

η_d Parameter controlling the decrease of the step-size estimate

κ Condition number

λ_k Estimating sequence component

μ Strong convexity parameter

$\prod$ Product notation

$\sum$ Summation notation

τ Hyperparameter

$\Phi_k(x)$ Estimating sequence component

Φ_k^* Minimal value of the estimating function

$\psi_k(x)$ Heavy ball type of momentum term

Ψ_k Upper bound on $\psi_k(x)$

∇ Gradient notation

∂ Subdifferential notation

∞ Infinity notation

$\cup$ Union of sets notation

$\{\cdot\}_k$ Sequence notation

$||\cdot||$ Euclidean norm

$\cdot^n$ n-dimensional vectors

$\cdot^+$ Non-negative values

$\cdot^T$ Transposition operator

1. Introduction

1.1 Big Data and Optimization

In today's digital age, we are surrounded by a massive amount of data. This phenomena comes at large as a result of the spreading of the use of online social media, internet, and global communication [1]. The need to make use of all this available data for the purpose of building more efficient and robust models has fueled the field of data-driven statistical learning [2]. Enabling efficient ways to harness the information available in the "big data" has reshaped many aspects of the modern world, such as businesses which are now using data-driven approaches to adapt their strategies [3], researchers who are revolutionizing methodologies [4], and governments who are making more informed decisions [5].

Recent research on big data keeps revealing its boundless potential. Researchers have shown that exploiting the information available in the data can result in substantial economic growth and improve the daily life for everyone in myriads of ways [6]. For instance, in the medical fields we can use the information available in the data to aid the fight against the spread of different diseases in a more effective manner [7]. Another area that has witnessed significant impact is online marketing [8]. We can also delve deep and obtain a better understanding of the mechanisms that influence the financial markets [9], create complex networks that are easier to understand [10], analyze social-computational systems [11], and ensure the robustness and security of important systems such as the internet and power grids [12].

As discussed above, harnessing the power of big data is reshaping entire industries, steering government policies, and paving path to more sustainable societies [13]. Considering the ongoing data-driven industrial revolution, understanding and being able to efficiently utilize this data are the essential elements for succeeding in this changing environment [14]. Optimization problems emerge naturally across various fields of engi-

neering and science. We often find ourselves looking for the best possible solution to a problem that we face, or situation that we encounter. In this case, our own intuition is casting an optimization problem and seeking to find a solution for it. In a very similar way, in engineering and science we find ourselves converting this intuition into a mathematical formulation, i.e., we cast different optimization problems [15]. These problems arise often in various disciplines such as signal processing, control, wireless communications and many more [2, 16, 17].

Obtaining mathematical models for different problems is important, however, we are typically interested in finding the optimal solution, which is far from straightforward. One of the major challenges faced along the path of finding the optimal solution for an optimization problem, is the fact that many of these problems are unsolvable. This leads us to seek for approximate solutions [18]. As is normal with all approximations, a natural question that arises is: "How reliable are the obtained solutions?" Answering such a question is tightly coupled with understanding the computational aspects and limitations associated with solving an optimization problem [19].

Coupling the computational aspects with the problem of creating a mathematical model for the problem of interest typically takes significant amount of time and effort. It is often the case, that researchers need to trade off between an "exact" formulation which might not be solvable, with an approximate" model, which can be solved efficiently. In practice, the latter models are typically preferred [20].

Such solutions have been observed in different fields in science and engineering, with the Linear Model serving as the canonical example. Their popularity is attributed to their simple nature, which also enables solvable models. Another inherent benefit of linear methods is that it is usually possible to interpret the obtained solution [21]. However, as we know from the outburst of data-driven algorithms, the linear approximations tend to be limited, and are not very successful at capturing the non-linear structures which are present in the data [22].

1.2 Background

All the optimization problems that are encountered in science and engineering, can be either non-convex or convex [20, 32, 24]. Despite the recent advances in optimization theory, finding and certifying their global solutions remain challenging [25]. On the other hand, the class of convex optimization problems has gathered significant attention in the research community [20, 25, 26]. Different from the case of non-convex problems, for the class of convex problems it is possible to find and certify the global solutions (or an arbitrarily tight approximation of them) [20].

Such problems arise often in the context of applications in several fields such as signal processing, information theory and wireless communications [27, 28, 29, 30, 31, 32]. A myriad of the aforementioned problems can be solved exactly. Nevertheless, in the context of modern engineering applications which are enabled by big data, we are more interested in finding approximate solutions, which can be computed efficiently [33].

Depending on the size of the underlying datasets, it is natural to seek to optimize the trade-off between a high per-iteration complexity and convergence speed. Methods that exhibit a higher per iteration complexity, such as Newton and/or quasi-Newton type methods (e.g., L-BFGS) also exhibit fast convergence. However, as the size of the datasets grows large, it becomes necessary to seek to devise methods that exhibit a low per-iteration complexity, and as fast as possible convergence. One of the most popular tools used to solve the large-scale optimization problems, are gradient-based methods designed to be agnostic to the problem formulation, i.e., considering the black-box framework. At each iteration, these methods query a black-box oracle to obtain relevant insight about the function that is being minimized [19]. To build efficient gradient-based methods, the following aspects need to be considered: *i)* They need to converge to a neighborhood of the optimal solution; *ii)* The number of first-order oracle calls, together with additional computations, need to be minimized [25, 26]. The performance bounds for different black-box gradient-based methods for different types of convex problems have been thoroughly investigated and established in [19, 29, 30, 34].

In this thesis, we consider the problem of devising accelerated methods for solving smooth and non-smooth convex optimization problems. Considering only the problem of devising efficient gradient-based methods for solving convex optimization problems with smooth objective, one of the most celebrated results is the development of the Fast Gradient Method (FGM) [35]. Based on the framework devised in [19], FGM is referred to as an optimal method, i.e., the method minimizes the calls of a first-order oracle while exhibiting a convergence rate $\mathcal{O}(1/k^2)$, where k is the iteration counter. On a framework level, one of the most significant advances was the development of the estimating sequences framework, initially introduced in [36] and later refined in [15, 37]. Using this framework, further variants of FGM constructing for solving optimization problems which have smooth and strongly convex cost functions [36], [15, Constant Step Scheme I]. These variants of FGM require at most $\sqrt{\kappa}\left(\ln\frac{1}{\epsilon}+\mathcal{O}(1)\right)$ iterations to converge to a point x with $f(x)-f^*\leq\epsilon$, where $\kappa=\frac{L}{\mu}$ and L,μ denote the Lipschitz constant and strong convexity parameter.

Despite the consideration that the complexity bounds reached by FGM-type methods are only proportional to the fundamental performance bounds introduced in [19], FGM and its different variants have always been regarded in the literature to be optimal methods. Interestingly, these meth-

ods started gathering more attention only after the publication of the seminal work on smoothing techniques [38], wherein the author approximated a non-smooth convex cost function by another smooth convex cost function. FGM was then used to minimize the approximated function. The authors in [39] further extended the work by devising new interior gradient algorithms which also exhibit an accelerated convergence rate. In another line of work, detailed in [40, 41, 42], several researchers have studied the problem of robustness of FGM-type methods with respect to the usage of inaccurate gradients of the objective function in the minimization process.

More recently, in addition to the estimating sequences framework, researchers have also focused on studying other approaches that can be used to accelerate gradient-based methods. In the line of work presented in [43, 44, 45], existing links between the integration of ordinary differential equations (ODE) and optimization were considered in the context of devising a different perspective on acceleration of first-order methods. More specifically, in [44] the authors derive a second-order ODE which is the limit of FGM. In [43], the authors show that different accelerated gradient methods can be reformulated as constant parameter second-order ODEs. Moreover, they show the equivalence between the stability of such systems and the accelerated convergence rate. Last, in [45] the authors demonstrate that different variants of FGM can be viewed as a structured approach to transition from the continuous-time curves created by the Bregman Lagrangian to accelerated algorithms. In another line of work, the authors of [46] show that it is possible to devise different variants of FGM by making use of the linear coupling between mirror and gradient descent. Yet another line of work has been introduced in [47, 48]. Specifically, the authors of [47] develop an accelerated gradient method by extending the results exiting for the ellipsoid method. The resulting method called Geometric Descent is more efficient than FGM, however suffers the drawback that it requires an exact line search to ensure accelerated convergence. The links between the Geometric Descent Method (GDM) introduced in [47] and the strongly-convex variants of FGM were later established in [48]. Another line of work presented in [49] used principles of robust control theory to derive convergence rate results for accelerated gradient methods. The authors in [50] use the analysis presented in [49] to construct a more efficient method, which they name as Triple Momentum Method (TMM). TMM is more efficient than FGM, in the sense that it exhibits a faster convergence rate, however it suffers the drawback that it is defined only for strongly convex objective functions. Even for this class of problems, when the value of the condition number is large, TMM exhibits slower convergence than FGM (for more details see [51, Figure 1]).

Another interesting line of work has been introduced in [52]. Therein, the authors cast a semidefinite program (SDP) which is used to model the

improvement of the worst accuracy that a black-box numerical method can exhibit. Later, in [53] the authors analyze the tightness of the worst-case accuracies that the SDP yields. These results paved path to the development of new classes of optimal methods for minimizing smooth and non-strongly convex cost functions [54, 55]. Using the framework introduced in [52], the authors of [51] develop an optimal method for solving smooth and strongly convex optimization problems. The method proposed therein reaches the complexity bounds established in [19], however it suffers from several drawbacks. First, it is difficult to extend the framework to broader and more practical optimization setups, such as non-smooth optimization, stochastic optimization, etc. Second, the results demonstrated for the method are achieved by assuming that parameters relevant to the objective function (e.g., μ, L) are known. The sensitivity and robustness of the method to the inexact values of these parameters in the context of practical deployments requires further analysis and evaluation.

Different from all the other frameworks which have been used to develop accelerated gradient-based algorithms, estimating sequences have been consistently used to develop numerical methods that exhibit a competitive performance in a myriad of applications and optimization setups. In the context of applications, a myriad of novel results have been presented in [56, 57, 58, 59]. Specifically, in [56] the authors devise an accelerated gradient method used for minimizing a smooth loss function regularized by the trace norm of the matrix variable. In [57], the authors develop efficient distributed methods and show that their results match the existing results for FGM, with the additional cost coming from the communication constrains. Moreover, the authors in [58] consider the coupling of FGM-type of acceleration, multi-consensus and gradient tracking to devise algorithms that achieve optimal computation complexity and near-optimal communication complexity. Last, in [59] the authors develop an efficient variant of FGM by using the principle of differential quantization.

Estimating sequence-based approaches have also been successfully extended to other optimization setups. A myriad of interesting results have been established in the context of stochastic optimization [60, 61, 62]. In [60], the authors develop a stochastic accelerated gradient method for solving regularized risk minimization problems. An accelerated stochastic approximation algorithm based on FGM is presented in [61]. A new class of stochastic estimating sequences is presented in [62]. These stochastic estimating sequences are then used to devise efficient and robust stochastic methods. The development of non-Euclidean methods has also been widely studied in the recent years [63, 64]. The new estimating functions introduced in [64] are used to devise a novel bound on the nonlinear metric distortion to devise a Riemannian version of FGM. The method proposed therein exhibits accelerated convergence rate for finding the optimal solution of geodesically convex problems, which are smooth and strongly

convex. In [64], the authors present new estimating functions, which are used to devise the first global accelerated gradient method for Riemannian manifolds. Another relevant setup to which the estimating sequences framework has been successfully extended is the design of higher-order methods [37, 65, 66]. In [37], the author presents a unified framework which can be used for studying estimating sequences methods, and shows how to use the framework to devise a myriad of accelerated algorithms. An accelerated version of the Newton method is presented in [65]. Moreover, accelerated high-order proximal methods developed using the inexact oracle framework are presented in [66]. Another setup wherein the acceleration effect obtained by utilizing the estimating sequences framework becomes relevant is related to non-convex problems [67, 68]. The generalization of FGM to non-convex setups is introduced in [67]. Moreover, for nonconvex function with Lipschitz continuous first and second derivatives, the authors present a Hessian-free accelerated gradient method [68].

Estimating sequences can also be considered to devise efficient methods to solve constrained optimization problems. The fundamentals behind such extensions are introduced in [15, Chapters 2.2.4 - 2.2.5]. The key behind such extension lies in exploring the coupling of the estimating sequences framework together with the gradient mapping framework [19]. A similar approach can also be used for solving problems with convex composite objective functions. Extensions of these frameworks to solving such problems are introduced in [69, 70, 71]. In [69], the author also introduces a new class of estimating sequences and uses them to devise an accelerated gradient method called AMGS. Together with AMGS, the author also introduces a backtracking strategy which is used to estimate the value of the Lipschitz constant. In the same work, the author also presents an efficient technique for approximating the strong convexity parameter of the cost function. The main drawback of AMGS comes due to its high per iteration complexity because for each iteration it needs two projection-like operations. This issue has been mitigated with the development of FISTA. The method exhibits an accelerated rate of convergence and a lower per iteration complexity. Despite the attractive properties, FISTA does not convergence as fast as AMGS when considered in practical deployments [72, 73]. Another class of composite estimating sequences are introduced in [71]. Different from the estimating sequences presented in [69], the composite estimating sequences are used to devise accelerated gradient-based schemes which require one projection-like operation per iteration. Moreover, the method constructed therein, converges faster than both AMGS and FISTA when tested on practical problems and real-world datasets.

A plethora of gradient-based methods have already been studied in the literature in the context of different applications and optimization setups. Despite the framework that is used for designing the methods, in order for them to be considered as optimal when considering smooth

convex optimization the following are important: *i*) the method achieves an accelerated convergence rate; *ii*) the estimated number of iterations is proportional to the complexity bounds given in [19]. In the context of composite objectives with non-smooth term, it is desirable for the resulting methods to exhibit an accelerated convergence rate. A unified framework that can be used for devising gradient-based algorithms is presented in [74]. In [53], the authors compute the exact worst-case bounds for the variant of FGM presented in [35]. As we discussed earlier, the variant of FGM built using the estimating sequences framework is presented in [36, 15]. In [15], the author argues that one of the most relevant considerations for designing optimal methods relates to parsing global topological information about the cost function. The collection of such information is enabled by the estimating sequences. They consist of the sequences $\{\lambda_k\}_k$ and $\{\phi_k(x)\}_k$, which enable the computation of the rate of convergence for the iterates and accumulation of information around them.

Considering the popularity of estimating sequence methods, one can easily conclude that such an intuition is correct. A major challenge with the framework arises because the estimating functions are not unique. Finding a structure for estimating functions that always result in the most efficient (both when considering the theoretical bounds and practical performance) methods that can be devised for the corresponding problem classes remains an open question. As we have already discussed, different variants FGM , e.g., the ones presented in [15, Constant Step Scheme I], [35], [69], etc., are built using different estimating functions. Nevertheless, they are all very efficient and enjoy the accelerated convergence rate properties. Despite the different structures for the estimating functions, all variants of FGM share the commonality that the parameters in iteration $k+1$, are updated by considering the values of the parameters in iteration k.

1.3 Objectives

Considering the plethora of frameworks for devising accelerated first-order methods, together with the possibilities to construct more efficient estimating functions, it is natural to ask: “Is it possible to construct more efficient methods by changing the structure of the estimating functions?". This is a central question in the thesis which is positively answered in terms of minimizing smooth and convex cost functions, as well as composite objective functions with a non-smooth term. The main contributions of the thesis are summarized in the next Section.

1.4 Contributions

- In Publication I, we introduce the generalized estimating sequences which contain additional momentum terms and show how they can be utilized to devise a generalized version of FGM. We also show how to derive FGM based on our proposed framework.

- In Publication II, we formalize the generalized estimating sequences framework and provide links between the momentum terms used to construct the proposed generalized estimating sequences with the heavy-ball momentum. Moreover, therein we establish the convergence results of our proposed method and prove that it converges faster than FGM. We also demonstrate numerically on real data and popular problems the robustness of the methods devised within our proposed framework to the inexact information on L and μ.

- In Publication III, we present the composite estimating sequences and show how it can be coupled with the gradient mapping framework to construct an accelerated gradient method for minimizing convex composite cost functions.

- In Publication IV, we prove new results and implications of using our proposed composite estimating sequences. Furthermore, we formalize and establish the convergence of our proposed composite objective multistep estimating sequence technique (COMET). We show that COMET requires only one projection-like operation per iteration and is more efficient than existing numerical methods for minimizing functions with composite structure.

- In Publication V, we show how to further extend the generalized estimating sequences framework for minimizing convex and composite cost functions. We embed the heavy-ball type of momentum introduced in Publications I and II into the composite estimating sequences presented in Publications III and IV. We use the new estimating functions to construct another numerical method and demonstrate its efficiency in solving practical problems with real-world datasets.

1.5 Thesis structure

The remainder of the thesis is organized as follows. Chapter 2 introduces the generalized estimating sequences framework for smooth functions. In the same Chapter, we also present the associated method and establish its

convergence. Chapters 3 and 4 focus on further extending the framework to composite objectives. Chapter 3 uses a tight bound for the cost function and the gradient mapping framework, to construct composite estimating sequences and the corresponding method for minimizing convex and composite cost functions. Chapter 4 further extends the work and introduces the generalized composite estimating sequences. Using these estimating sequences, we build yet another algorithm and establish its accelerated rate. Chapter 5 presents our final remarks of the work and highlights several remaining open problems.

2. Generalizing the estimating sequences

A myriad of accelerated gradient-based algorithms have been devised based on the estimating sequences framework [35, 35, 15, 37]. In this Chapter, we begin by looking at the simplest case of smooth and strongly convex objective functions and highlight the main findings of Publications I and II. We have already discussed in Chapter 1.2 that the existing variants of FGM that are built using different estimating sequences share the commonalty that updates at iteration $k+1$ are obtained by considering *only* the updates at iteration k. Considering the existing results on the heavy-ball method [75], we thus formulate the first research questions of this thesis: *i)* Can we construct estimating sequences which also consider information coming from the past iterates? *ii)* How does this impact the resulting optimization method?

In the sequel, we present our answers to the aforementioned questions.[1] The main contributions are summarized as follows:

- We introduce new estimating functions, whose values are dependent on the history of iterates.[2]

- We revisit the lemmas and theorems derived in the context of the classical estimating sequences framework and introduce new approaches to establish our findings. We also highlight the intuition behind the selection of the estimating sequences and design of the corresponding methods.

- For black-box optimization, we introduce a novel type of heavy-ball momentum, and show how to couple it with the estimating sequences framework. Different from the framework presented in [75], wherein

[1] Note that the detailed derivations used to establish the Lemmas and Theorems presented in the rest of the Thesis are provided as part of the individual contributions.

[2] The proposed framework allows for embedding any form of information that can accelerate the convergence of iterates.

the heavy-ball momentum stabilizes the iterates, our newly introduced momentum stabilizes the estimating functions themselves.

- We introduce a new gradient-based algorithm which allows for embedding our newly introduced momentum term into the classical FGM. We also show how FGM can be derived by discarding the additional memory terms.

- We improve upon the existing convergence results for FGM. We establish the optimality of our proposed method, and prove that the bound on the iterations becomes $\sqrt{\frac{L}{2\mu}}\left(\ln\left(\frac{\mu R_0^2}{2\epsilon}\right) + \ln(5)\right)$, where $R_0 = ||x_0 - x^*||$ and $\epsilon \leq \frac{\mu}{2}R_0^2$. This results in an improvement over the bound reached from FGM by more than $\frac{1}{\sqrt{2}}$.

- The newly introduced convergence results allow for setting $\gamma_0 = 0$. In our publications, we show numerically that this result alone enables a faster convergence over FGM. We note that such result is an extension of the existing analysis for FGM, wherein the convergence was established only for $\gamma_0 \in [\mu, 3L + \mu]$. Moreover, it enables the robustness of the initialization of our method to the inexact estimate of μ.

- In our simulations, we highlight the efficiency of utilizing our method to solve problems that arise often in signal processing. Both simulated and publicly available datasets are used.

2.1 Proposed method

Let us beging by considering the following problem

$$\underset{x\in\mathcal{R}^n}{\text{minimize}}\ f(x), \tag{2.1}$$

where $f : \mathcal{R}^n \to \mathcal{R}$ has strong convexity parameter μ and Lipschitz continuous gradient L, defined by a deterministic black-box oracle.

First, let $\mathcal{I} = x_0 + \mathbf{span}\{\nabla f(x_0), \dots \nabla f(x_{k-1})\}$ for $k = 0, 1, 2, \dots, t$, where t is the current iteration. Next, we highlight the following definition.

Definition 1. *The sequences $\{\Phi_k\}_k$ and $\{\lambda_k\}_k$, $\lambda_k \geq 0$, are called generalized estimating sequences of the function $f(\cdot)$, if there exists a sequence of bounded functions $\{\psi_k\}_k$, $\lambda_k \to 0$, and $x \in \mathcal{I}, \forall k$ we have*

$$\Phi_k(x) \leq \lambda_k\Phi_0(x) + (1 - \lambda_k)\left(f(x) - \psi_k(x)\right). \tag{2.2}$$

Using $\psi_k(x)$ in (2.2) allows for including more information on the cost function that can enable the faster convergence. Let us now show how to

use the generalized estimating sequences to measure the rate of convergence for the iterates formed during the minimization process.

Lemma 1. *If for some sequence of points* $\{x_k\}_k$ *we have* $f(x_k) \leq \Phi_k^* \triangleq \min_{x\in\mathcal{I}} \Phi_k(x)$*, then* $f(x_k) - f(x^*) \leq \lambda_k \left[\Phi_0(x^*) - f(x^*)\right] - (1-\lambda_k)\,\psi_k(x^*)$*, where* $x^* = \arg\min_{x\in\mathcal{R}^n} f(x)$.

Let us next proceed to presenting our proposed definitions for the terms that comprise the generalized estimating sequences.

Lemma 2. *Assume that there exist sequences* $\{\alpha_k\}_k$*, where* $\alpha_k \in (0,1)$ *and* $\sum_{k=0}^{\infty}\alpha_k = \infty$*, and* $\{y_k\}_k$*, where* $y_k \in \mathcal{R}^n$*, and a sequence of functions* $\{\psi_k\}_k$*, with an upper bound* Ψ_k*, such that* $\psi_k(x) \geq 0$*,* $\forall k$*. Let* $\psi_0(x) = 0$ *and* $\lambda_0 = 1$*. Then, the sequences* $\{\Phi_k\}_k$ *and* $\{\lambda_k\}_k$*, which are defined recursively as*

$$\lambda_{k+1} = (1-\alpha_k)\lambda_k, \tag{2.3}$$

$$\begin{aligned}\Phi_{k+1}(x) = (1-\alpha_k)(\Phi_k(x)+\psi_k(x)) - \psi_{k+1}(x) - \Psi_k + \alpha_k\psi_k(x)\\ +\alpha_k\Big(f(y_k)+\nabla f(y_k)^T(x-y_k) + \frac{\mu}{2}||x-y_k||^2\Big),\end{aligned} \tag{2.4}$$

are generalized estimating sequences.

Recall that the structure for $\{\Phi_k(x)\}_k$ has not been presented yet. As discussed in [15], accelerated methods needs to exploit some of the topological features of the cost function. Such observation can be validated based on existing results on second-order methods. Considering Newton's method, as shown in [20, Fig. 9.19], making use of the information available in the Hessian enables the construction of ellipsoids around each iterate. Such ellipsoids facilitate corrections of the selected descent direction. For gradient-based methods, which do not have access to Hessian-related information, we can devise balls around each x_k, without "discriminating" the different search directions. Mathematically, this is modeled by using isotropic functions, which scan with radius γ_k. The resulting Hessian then becomes $\nabla^2\phi_k(x) = \gamma_k I$. The estimating function is

$$\phi_k(x) = \phi_k^* + \frac{\gamma_k}{2}||x-v_k||^2,\ \forall k, \tag{2.5}$$

and has minimum value ϕ_k^*, radius $\gamma_k \in \mathcal{R}^+$ and is centered around $v_k \in \mathcal{R}^n$. Similar structure as (2.5) is also used for constructing FGM [15]. Different from (2.5), we let

$$\Phi_k(x) = \phi_k^* + \frac{\gamma_k}{2}||x-v_k||^2 - \psi_k(x).\ \forall k, \tag{2.6}$$

The added term $\psi_k(x)$ is

$$\psi_k(x) \triangleq \sum_{i=0}^{k-1}\beta_{i,k}\frac{\gamma_i}{2}||x-v_i||^2.\ \forall k, \tag{2.7}$$

A simple example for $\beta_{i,k}$ is

$$\beta_{i,k} = \begin{cases} \min\left(1, \frac{\mu}{\gamma_{k-1}}\right), & \text{if } i = k-1, \\ 0, & \text{otherwise.} \end{cases} \tag{2.8}$$

Considering the black-box setting, wherein prior knowledge of the structure of the objective function is not available, we allow our newly introduced scanning functions to "self-regulate" and encompass information that was already available from the earlier iterates. Selecting $\beta_{i,k}$ according to (2.8) ensures that (2.7) remains finite since only the estimating function in iteration $k-1$ is used.

Let us now present the recursive relations for ϕ_k^*, γ_k, and v_k.

Lemma 3. *Assume that the coefficients $\beta_{i,k}$ are selected according to (2.8), and let $\Phi_0(x) = \phi_0^* + \frac{\gamma_0}{2}||x - v_0||^2$. Then, the process defined in Lemma 2 preserves the quadratic canonical structure of the scanning function introduced in (2.5). Moreover, the sequences $\{\gamma_k\}_k$, $\{v_k\}_k$ and $\{\phi_k^*\}_k$ can be computed as*

$$\gamma_{k+1} = (1-\alpha_k)\gamma_k + \alpha_k\left(\mu + \sum_{i=0}^{k-1}\beta_{i,k}\gamma_i\right), \tag{2.9}$$

$$v_{k+1} = \frac{1}{\gamma_{k+1}}\left((1-\alpha_k)\gamma_k v_k + \mu\alpha_k\left(y_k - \frac{1}{\mu}\nabla f(y_k) + \sum_{i=0}^{k-1}\frac{\beta_{i,k}\gamma_i}{\mu}v_i\right)\right), \tag{2.10}$$

$$\begin{aligned} \phi_{k+1}^* = {} & \alpha_k f(y_k) + (1-\alpha_k)\phi_k^* + \frac{\alpha_k\gamma_k(1-\alpha_k)(\mu + \sum_{i=1}^{k-1}\beta_{i,k}\gamma_i)}{2\gamma_{k+1}}||y_k - v_k||^2 \\ & + \frac{\alpha_k^3}{\gamma_{k+1}}\sum_{i=0}^{k-1}\beta_{i,k}\gamma_i||v_i - y_k||\,||\nabla f(y_k)|| - \frac{\alpha_k^2||\nabla f(y_k)||^2}{2\gamma_{k+1}} \\ & + \frac{\alpha_k(1-\alpha_k)\gamma_k}{\gamma_{k+1}}\left((v_k - y_k)^T\nabla f(y_k) + \sum_{i=0}^{k-1}\beta_{i,k}\gamma_i||y_k - v_i||||y_k - v_k||\right) \\ & + (1-\alpha_k)\frac{\gamma_k}{2}||x_{\Phi_k}^* - v_k||^2 + \alpha_k\sum_{i=0}^{k-1}\frac{\beta_{i,k}\gamma_i}{2}||y_k - v_i||^2 \\ & + \frac{(1-\alpha_k)\,\alpha_k^2}{\gamma_{k+1}}\sum_{i=0}^{k-1}\beta_{i,k}\gamma_i(v_i - y_k)^T\nabla f(y_k) + \sum_{i=0}^{k-1}\beta_{i,k}\frac{\gamma_i}{2}||x_{\Phi_k}^* - v_i||^2. \end{aligned} \tag{2.11}$$

We will choose $\{x_k\}_k$, $\{y_k\}_k$ and $\{v_k\}_k$ to ensure that $f(x_k) \leq \Phi_k^*$, $\forall k$. For iteration k, suppose that $\phi_k^* \geq f(x_k)$. At iteration $k+1$, by relaxing (2.11)

and making some algebraic manipulations, we reach

$$\phi^*_{k+1} \geq f(y_k) + (1-\alpha_k)\nabla f(y_k)^T(x_k - y_k) - \frac{\alpha_k^2}{2\gamma_{k+1}}||\nabla f(y_k)||^2 \tag{2.12}$$

$$+\frac{\alpha_k(1-\alpha_k)\gamma_k}{\gamma_{k+1}}(v_k - y_k)^T\nabla f(y_k) + (1-\alpha_k)\frac{\alpha_k^2}{\gamma_{k+1}}\sum_{i=0}^{k-1}\beta_{i,k}\gamma_i(v_i - y_k)^T\nabla f(y_k).$$

The necessary conditions of Lemma 1 are fulfilled if $\phi^*_{k+1} \geq f(x_{k+1})$. Thus, we further relax the lower bound by making use of

$$f(y_k) - \frac{1}{2L}||\nabla f(y_k)||^2 \geq f(x_{k+1}). \tag{2.13}$$

To ensure that (2.13) is satisfied, it suffices to take a gradient step for y_k [15, Theorem 2.1.5]. This allows for computing α_k as

$$\alpha_k = \sqrt{\frac{\gamma_{k+1}}{L}}. \tag{2.14}$$

Considering (2.9), we can write

$$\alpha_k = \frac{\left(\mu + \sum_{i=1}^{k-1}\beta_{i,k}\gamma_i - \gamma_k\right)}{2L} + \frac{\sqrt{\left(\mu + \sum_{i=1}^{k-1}\beta_{i,k}\gamma_i - \gamma_k\right)^2 + 4L\gamma_k}}{2L}. \tag{2.15}$$

Substituting the expression for α_k presented in (2.15), we can revise (2.12) as

$$\phi^*_{k+1} \geq f(x_{k+1}) + (1-\alpha_k)\nabla f(y_k)^T\Big((x_k - y_k) \tag{2.16}$$

$$+\frac{\alpha_k\gamma_k}{\gamma_{k+1}}(v_k - y_k) + \frac{\alpha_k^2}{\gamma_{k+1}}\sum_{i=0}^{k-1}\beta_{i,k}\gamma_i(v_i - y_k)\Big).$$

The terms of $\{y_k\}_k$ can be acquired from

$$x_k - y_k + \frac{\alpha_k\gamma_k}{\gamma_{k+1}}(v_k - y_k) + \frac{\alpha_k^2}{\gamma_{k+1}}\sum_{i=0}^{k-1}\beta_{i,k}\gamma_i(v_i - y_k) = 0.$$

This yields

$$y_k = \frac{\gamma_{k+1}x_k + \alpha_k\gamma_k v_k + \alpha_k^2\sum_{i=0}^{k-1}\beta_{i,k}\gamma_i v_i}{\gamma_{k+1} + \alpha_k\gamma_k + \alpha_k^2\sum_{i=0}^{k-1}\beta_{i,k}\gamma_i}. \tag{2.17}$$

The complete procedure is presented in Algorithm 1.

Let us next compare Algorithm 1 with [15, (2.2.19)]. First, observe that the relations for computing α_k and γ_k are different due to the different estimating functions. A similar observation can be made by looking at the recurrent relation for computing y_k, $\forall k$. An important difference is the range of values for which γ_0 can be selected. The existing convergence results for FGM are limited to the range $\gamma_0 \in [\mu, 3L + \mu]$. Our algorithm

Algorithm 1 Proposed Method

Input $x_0 \in \mathcal{R}^n$, set $\gamma_0 \in [0, \mu[\cup[2\mu, 3L+\mu]$ and $v_0 = x_0$.
while stopping criterion is not meet **do**
  $\alpha_k \leftarrow \frac{\left(\mu+\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i-\gamma_k\right)+\sqrt{\left(\mu+\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i-\gamma_k\right)^2+4L\gamma_k}}{2L}$
  $\gamma_{k+1} \leftarrow (1-\alpha_k)\gamma_k + \alpha_k\left(\mu + \sum_{i=0}^{k-1}\beta_{i,k}\gamma_i\right)$
  $y_k \leftarrow \frac{\gamma_{k+1}x_k+\alpha_k\gamma_k v_k+\alpha_k^2\sum_{i=0}^{k-1}\beta_{i,k}\gamma_i v_i}{\gamma_{k+1}+\alpha_k\gamma_k+\alpha_k^2\sum_{i=0}^{k-1}\beta_{i,k}\gamma_i}$
  $x_{k+1} \leftarrow y_k - \frac{1}{L}\nabla f(y_k)$
  $v_{k+1} \leftarrow \frac{1}{\gamma_{k+1}}\left((1-\alpha_k)\gamma_k v_k + \mu\alpha_k\left(y_k - \frac{1}{\mu}\nabla f(y_k) + \sum_{i=0}^{k-1}\frac{\beta_{i,k}\gamma_i}{\mu}v_i\right)\right)$
end while
Output x_{k+1}

converges for a larger range of γ_0. The extension of the convergence results to cover also the case where $\gamma_0 = 0$ enables the robustness of the initialization of our method when using an inexact μ. Computing the exact value for μ would require additional computations. Moreover, the additional terms coming from using $\{\psi_k\}_k$ appear as multipliers of α_k^2. They are also present in the update of v_{k+1}. Last, observe that FGM can be derived by letting $\beta_{i,k} = 0, \quad \forall i, k$.

2.2 Bounds on convergence rate

Let us now present the key convergence results for our proposed method. First, we show that the convergence of the iterates obtained during the minimization process is dependent on both $\{\lambda_k\}_k$ and $\{\psi_k\}_k$.

Theorem 1. *If we let $\lambda_0 = 1$ and $\lambda_k = \prod_{i=0}^{k-1}(1-\alpha_i)$, Algorithm 1 generates a sequence of points $\{x_k\}_k$ such that*

$$f(x_k) - f^* \leq \lambda_k\left[f(x_0) - f(x^*) + \frac{\gamma_0}{2}||x_0 - x^*||^2\right] - (1-\lambda_k)\psi_k(x^*). \quad (2.18)$$

From Lemma 2, we have that $\{\lambda_k\}_k \to 0$ when $k \to \infty$. The estimate of the rate of convergence for $\{\lambda_k\}_k$ is given in the following Lemma.

Lemma 4. *For all $k \geq 0$, Algorithm 1 guarantees that*

$$\begin{aligned}\lambda_k &\leq \frac{2\mu}{L\left(e^{\frac{k+1}{2}\sqrt{\frac{\mu+\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i}{L}}} - e^{-\frac{k+1}{2}\sqrt{\frac{\mu+\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i}{L}}}\right)^2} \\ &\leq \frac{2\mu}{\left(\mu+\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i\right)(k+1)^2}.\end{aligned} \quad (2.19)$$

Last, we demonstrate that Algorithm 1 is optimal.

Theorem 2. *In Algorithm 1, let $\mu > 0$. Then, the scheme generates a sequence of points such that*

$$f_k - f^* \leq \frac{\mu||x_0 - x^*||^2}{\left(e^{\frac{k+1}{2}\sqrt{\frac{\mu+\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i}{L}}} - e^{-\frac{k+1}{2}\sqrt{\frac{\mu+\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i}{L}}}\right)^2} - (1-\lambda_k)\psi_k(x^*). \quad (2.20)$$

where $f_k = f(x_k)$ and $f^* = f(x^*)$. This means that the method is optimal when the accuracy $\epsilon \leq \frac{\mu}{2}||x_0 - x^*||^2$.

For the class of problems considered in this Chapter, FGM reaches the following bound [15, (2.2.17)]

$$k_{FGM} \geq \sqrt{\frac{L}{\mu}}\left(\ln\left(\frac{\mu R_0^2}{2\epsilon}\right) + \ln(23/3)\right). \quad (2.21)$$

On the other hand, if we select $\beta_{i,k}$ according to (2.8), the proposed method reaches the following bound on the iterations

$$k_{Proposed} \geq \sqrt{\frac{L}{\mu + \min\left(\gamma_{k-1}, \mu\right)}}\left(\ln\left(\frac{\mu R_0^2}{2\epsilon}\right) + \ln(5)\right). \quad (2.22)$$

Observe that the bound presented in (2.22) is impacted by the rate of increase for the terms in $\{\gamma_k\}_k$. As we demonstrate in our Publications I and II, the terms $\{\gamma_k\}_k$ grow exponentially in k, and converge to 2μ. Thus, the bound to the required number of iterations converges to

$$k_{Proposed} \to \sqrt{\frac{L}{2\mu}}\left(\ln\left(\frac{\mu R_0^2}{2\epsilon}\right) + \ln(5)\right). \quad (2.23)$$

Comparing the convergence results presented in (2.20) to the existing lower bound for FGM given in (2.21), we highlight the improvement by at least a factor of $1/\sqrt{2}$.

3. Extending the existing estimating sequence framework to composite objectives

In this Chapter, we focus on a broader class of convex problems, which are expressed as the sum of a smooth convex function together with a non-smooth convex function. In the sequel, we present the main findings of Publications III and IV. For this class of problems, several estimating sequences methods have been introduced in [69, 72, 73]. Links between methods that were not originally devised by using the estimating sequences framework, such as FISTA, with estimating sequences methods have been presented in [73]. Despite these methods being devised using different frameworks, they all share in common the accelerated rate of convergence. Nevertheless, when comparing their performance in solving practical problems with real-world data, we have observed that they exhibit different convergence properties. Moreover, comparing the original FGM with FISTA and AMGS for minimizing smooth convex functions, we have observed that FGM is more efficient. Thus, it becomes relevant to extend the estimating sequences framework used for devising FGM to the setup of composite objectives.

In the sequel, we introduce our proposed composite estimating sequences and show how to construct a composite objective estimating sequence technique that exhibits an accelerated rate of convergence. The main contributions are summarized as follows:

- We present new estimating sequences that are useful for devising numerical methods for minimizing the broader class of composite functions.

- We introduce new composite estimating functions, devised by coupling the gradient mapping framework introduced in [19] together with a tight bound on the composite cost function.

- Different from the "classical" functions used in [15], our proposed composite estimating functions exploit a tight bound on the composite cost function, together with its subgradients. This allows for devising accelerated gradient-based methods applicable to more general optimization

problems.

- Based on the composite estimating sequences, we devise the Composite Objective Multi-step Estimating-sequence Techniques (COMET). Our proposed method, is equipped with an efficient step-size adaption strategy. Different from AMGS, COMET requires only one projection-like operation per iteration.

- We prove that COMET exhibits an accelerate rate despite the inexact information of the Lipschitz constant.

- Through computational experiments, we highlight the robustness of COMET to the inexact information of μ.

- Moreover, we conduct extensive computational experiments for different practical data processing problems modeled through composite and convex cost functions. We demonstrate the superior performance of our proposed method relative to the existing benchmarks. To highlight the efficiency and robustness of our proposed method in a reliable manner, we consider real-world datasets.

3.1 Preliminaries

The problems of interest have the following structure

$$\underset{x\in\mathcal{R}^n}{\text{minimize}} \quad F(x) = f(x) + \tau g(x), \quad \tau > 0, \tag{3.1}$$

Transferring the strong convexity parameter of $g(x)$ inside $F(x)$ yields

$$F(x) = \Big(f(x) + \frac{\tau\mu_g}{2}||x-x_0||^2\Big) + \tau\Big(g(x) - \frac{\mu_g}{2}||x-x_0||^2\Big) = \hat{f}(x) + \tau\hat{g}(x). \tag{3.2}$$

Considering the above-mentioned strong convexity transfer, we can write $L_{\hat{f}} = L_f + \tau\mu_g$ and $\mu_{\hat{f}} = \mu_f + \tau\mu_g$. Moreover, we observe that $\mu_{\hat{g}} = 0$.

Recall that for $\hat{f}(x)$ we can write

$$\hat{f}(x) \leq \hat{f}(y) + \nabla\hat{f}(y)^T(x-y) + \frac{L_{\hat{f}}}{2}||y-x||^2, \tag{3.3}$$

$$\hat{f}(x) \geq \hat{f}(y) + \nabla\hat{f}(y)^T(x-y) + \frac{\mu_{\hat{f}}}{2}||y-x||^2. \tag{3.4}$$

In a similar manner, by definition of the subgradient of a function, we can write

$$\hat{g}(x) \geq \hat{g}(y) + s(y)^T(x-y), \tag{3.5}$$

where $s(y)$ denotes a subgradient of $\hat{g}(y)$. Furthermore, consider

$$m_L(y;x) \triangleq \hat{f}(y)+\nabla \hat{f}(y)^T(x-y)+\frac{L}{2}||x-y||^2+\tau\hat{g}(x), \tag{3.6}$$

where $L \geq L_{\hat{f}}$. Substituting (3.3) in (3.6), yields

$$m_L(y;x) \geq F(x), \forall x, y \in \mathcal{R}^n. \tag{3.7}$$

Next, let us introduce the composite gradient mapping

$$T_L(y) \triangleq \arg\min_{x\in\mathcal{R}^n} m_L(y;x). \tag{3.8}$$

Morever, we define the composite reduced gradient

$$r_L(y) \triangleq L\left(y - T_L(y)\right). \tag{3.9}$$

Considering the special case $\tau = 0$, (3.2) results in $\hat{f}(x) = f(x)$. Observe that this would be the case wherein $m_L(y;x)$ would be differentiable in its variables. Applying the optimality condition for (3.8), we can write $\nabla m_L(y;x) = 0$. Replacing (3.6) in (3.8), and evaluating the first order condition, yields $T_L(y) = y - \frac{\nabla \hat{f}(y)}{L}$. Replacing such result in (3.9), we obtain $r_L(y) = \nabla F(y) = \nabla f(y)$. Considering the broader case where $\tau \neq 0$, based on the optimality criteria for (3.8), we can write

$$\partial m_L(y;T_L(y))^T(x - T_L(y)) \geq 0,$$

$$\left(\nabla \hat{f}(y)+L(T_L(y)-y)+\tau s_L(y)\right)^T (x-T_L(y)) \geq 0, \tag{3.10}$$

where ∂ is the subdifferential of $m_L(y;T_L(y))$ and $s_L(y) \in \partial\hat{g}(T_L(y))$. Lettin g the first factor of (3.10) be equal to 0, together with utilizing (3.9), results in

$$r_L(y) = L(y - T_L(y)) = \nabla \hat{f}(y) + \tau s_L(y). \tag{3.11}$$

The next theorem introduces a tighter lower bound than the one given in (3.4) for $F(x)$.

Theorem 3. *Let $F(x)$ be a composition of an $L_{\hat{f}}$-smooth and $\mu_{\hat{f}}$-strongly convex function $\hat{f}(x)$, and a simple convex function $\hat{g}(x)$, as given in (3.2). For $L \geq L_{\hat{f}}$, and $x, y \in \mathcal{R}^n$ we have*

$$F(x) \geq \hat{f}(T_L(y)) + \tau\hat{g}(T_L(y)) + r_L(y)^T(x - y) + \frac{\mu_{\hat{f}}}{2}||x - y||^2 + \frac{1}{2L}||r_L(y)||^2. \tag{3.12}$$

3.2 Proposed method

Similar to the previous Chapter, let us define the following.

Definition 2. *The sequences* $\{\phi_k\}_k$ *and* $\{\lambda_k\}_k$, $\lambda_k \geq 0$, *are called composite estimating sequences of the function* $F(\cdot)$ *defined in* (3.2), *if* $\lambda_k \to 0$ *as* $k \to \infty$, *and* $\forall x \in \mathcal{R}^n$, $\forall k \geq 0$, *we have*

$$\phi_k(x) \leq \lambda_k \phi_0(x) + (1 - \lambda_k) F(x). \tag{3.13}$$

Observe that the proposed composite estimating sequences can estimate the rate of convergence of $\{x_k\}_k$. This is captured in the sequel.

Lemma 5. *If for some sequence of points* $\{x_k\}_k$ *we have* $F(x_k) \leq \phi_k^* \triangleq \min_{x \in \mathcal{R}^n} \phi_k(x)$, *then* $F(x_k) - F(x^*) \leq \lambda_k \left[\phi_0(x^*) - F(x^*)\right]$, *where* $x^* = \arg\min_{x \in \mathcal{R}^n} F(x)$.

The terms comprising the composite estimating sequences are computed recursively as shown below.

Lemma 6. *Assume that there exists a sequence* $\{\alpha_k\}_k$, *where* $\alpha_k \in (0, 1)$ $\forall k$, *such that* $\sum_{k=0}^{\infty} \alpha_k = \infty$, *and an arbitrary sequence* $\{y_k\}_{k=0}^{\infty}$. *Furthermore, let* $\lambda_0 = 1$ *and assume that the estimates* L_k, $\forall k$, *of the Lipschitz constant* $L_{\hat{f}}$ *are selected in a way that inequality* (3.3) *is satisfied for all the iterates* x_k *and* y_k. *Then, the sequences* $\{\phi_k\}_k$ *and* $\{\lambda_k\}_k$, *which are defined recursively as*

$$\lambda_{k+1} = (1 - \alpha_k)\lambda_k, \tag{3.14}$$

$$\begin{aligned} \phi_{k+1}(x) = (1 - \alpha_k)\phi_k(x) + \alpha_k F\left(T_{L_k}(y_k)\right) + \alpha_k \frac{1}{2L_k} ||r_{L_k}(y_k)||^2 \\ + \alpha_k \left(r_{L_k}(y_k)^T (x - y_k) + \frac{\mu_f}{2} ||x - y_k||^2 \right), \end{aligned} \tag{3.15}$$

are composite estimating sequences.

Let us now compare our findings presented in Definition 2, Lemma 5 and Lemma 6 with the results obtained in [15, Definition 2.2.1, Lemma 2.2.1, Lemma 2.2.2]. If the objective function were differentiable, the proposed Definition 2 and Lemma 5 would reduce to the baseline results introduced for FGM, which are obtained under the assumption of smooth objective function. From this viewpoint, our proposed framework extends results introduced in [15] to a broader setup. Second, based on the results proved for Lemma 5, the rate of convergence of $\{x_k\}_k$ would be characterized by the rate that $\lambda_k \to 0$. Third, (3.15) highlights the effect of using the tighter bound introduced in Theorem 3. Last, the objective function given in (3.15) is now computed based on the composite gradient mapping. Unlike the case of FGM, the proposed composite estimating functions exploit the subgradients of the non-smooth cost function to build $\{\phi_k\}_k$.

The terms of the sequence $\{\phi_k\}_k$ can be computed as follows

$$\phi_k(x) = \phi_k^* + \frac{\gamma_k}{2} ||x - v_k||^2, \ \forall k = 1, 2, \ldots. \tag{3.16}$$

We highlight that there could be choices for $\phi_k(x)$, which can lead to different algorithms (see [63, 64]). We can now proceed to presenting the recursive relations for the terms $\{\gamma_k\}_k$, $\{v_k\}_k$, and $\{\phi_k^*\}_k$.

Lemma 7. *Let* $\phi_0(x) = \phi_0^* + \frac{\gamma_0}{2}||x - v_0||^2$*, where* $\gamma_0 \in \mathcal{R}^+$ *and* $v_0 \in \mathcal{R}^n$*. Then, the process defined in Lemma 6 preserves the canonical form of the function presented in* (3.16)*, where the sequences* $\{\gamma_k\}_k$*,* $\{v_k\}_k$*, and* $\{\phi_k^*\}_k$ *can be computed as follows*

$$\gamma_{k+1} = (1 - \alpha_k)\gamma_k + \alpha_k \mu_{\hat{f}}, \tag{3.17}$$

$$v_{k+1} = \frac{1}{\gamma_{k+1}}\Big((1 - \alpha_k)\gamma_k v_k + \alpha_k \left(\mu_{\hat{f}} y_k - L_k(y_k - T_{L_k}(y_k))\right)\Big), \tag{3.18}$$

$$\begin{aligned}\phi_{k+1}^* = (1 - \alpha_k)\phi_k^* + \alpha_k \Big(F(T_{L_k}(y_k)) + \frac{1}{2L_k}||r_{L_k}(y_k)||^2\Big) - \frac{L_k^2 \alpha_k^2}{2\gamma_{k+1}}||y_k - T_{L_k}(y_k)||^2 \\ + \frac{\mu_{\hat{f}} \alpha_k \gamma_k (1-\alpha_k)}{2\gamma_{k+1}}||y_k - v_k||^2 + \frac{L_k \alpha_k \gamma_k (1 - \alpha_k)}{\gamma_{k+1}}(y_k - v_k)^T (y_k - T_{L_k}(x_k)).\end{aligned} \tag{3.19}$$

Different from the results given in [15], our proposed framework also allows for the line search adaptation.[1] To enable faster convergence to the optimal solution, it would be desirable to choose the smallest value L_k for which (3.3), wherein $L_{\hat{f}} = L_k$, is satisfied $\forall k = 0, 1, \ldots$. Then, it would be desirable to control the increase of its value throughout the minimization process. Such approach would enforce the algorithm to perform "larger steps towards x^*" during the first few iterations. In the later iterations, i.e., when x_k is closer to x^*, having large L_k would prevent the method from overshooting past x^*. Unfortunately, such approach relies on the assumption that $L_{\hat{f}}$ is perfectly known. This makes it unsuitable for practical setups. Instead, we choose a line search strategy which enables the following: *i)* Robustness of the algorithm with respect to the selection of L_0; *ii)* Dynamic changes of the values of L_k, $\forall k = 0, 1, \ldots$. Our proposed line search strategy utilizes $\eta_u > 1$, which increases the value of L_k, and $\eta_d \in]0, 1[$, which decreases the value of L_k. As we have shown in our articles, the impact of the additional backtracks is minimal. This has also been observed in [72, 73]. The proposed algorithm for solving problems with composite objectives is given in Algorithm 2. In line 3 of Algorithm 2, we use $K_{\max}$ to denote the maximum allowed number of iterations. Its value can be chosen to optimize the trade-off between the needed accuracy, and computations/processing time. Comparing the method devised in this Chapter to FGM (CSS I in [15]), we can observe that $\{\alpha_k\}_k$ and $\{\gamma_k\}_k$ share similar recursive structures. The update of y_k is different. In the case of our proposed method, y_k is computed independently of the value of $\mu_{\hat{f}}$. The update rule for the iterates x_k is also different. Because of the structure of the cost function, the next iterate is obtained through a proximal gradient step. The assumption on the simplicity of the non-smooth term $g(x)$ ensures that the proximal term can be computed with complexity $\mathcal{O}(n)$ [76]. The

[1] A myriad of backtracking line search strategies have already been presented in the literature (see [69, 70]).

Algorithm 2 Proposed Method

Input $x_0 \in \mathcal{R}^n$, $L_0 > 0$, $\mu_{\hat{f}}$, $\gamma_0 \in [0, 3L_0 + \mu_{\hat{f}}]$,
$\eta_u > 1$ and $\eta_d \in]0, 1[$.
Set $k = 0$, $i = 0$ and $v_0 = x_0$.
while $k \leq K_{\max}$ **do**
  $\hat{L}_i \leftarrow \eta_d L_k$
  while True **do**
    $\hat{\alpha}_i \leftarrow \frac{(\mu_{\hat{f}} - \gamma_k) + \sqrt{(\mu_{\hat{f}} - \gamma_k)^2 + 4\hat{L}_i\gamma_k}}{2\hat{L}_i}$
    $\hat{\gamma}_{i+1} \leftarrow (1 - \hat{\alpha}_i)\gamma_k + \hat{\alpha}_i\mu_{\hat{f}}$
    $\hat{y}_i \leftarrow \frac{\hat{\gamma}_{i+1}x_k + \hat{\alpha}_i\gamma_k v_k}{\hat{\gamma}_{i+1} + \hat{\alpha}_i\gamma_k}$
    $\hat{x}_{i+1} \leftarrow \mathbf{prox}_{\frac{1}{\hat{L}_i}\hat{g}}\left(\hat{y}_i - \frac{1}{\hat{L}_i}\nabla f(\hat{y}_i)\right)$
    $\hat{v}_{i+1} \leftarrow \frac{1}{\hat{\gamma}_{i+1}}\left((1 - \hat{\alpha}_i)\gamma_k v_k + \hat{\alpha}_i\left(\mu_{\hat{f}}\hat{y}_i - \hat{L}_i\left(\hat{y}_i - \hat{x}_{i+1}\right)\right)\right)$
    if $F(\hat{x}_{i+1}) \leq m_{\hat{L}_i}(\hat{y}_i, \hat{x}_{i+1})$ **then**
      Break from loop
    else
      $\hat{L}_{i+1} \leftarrow \eta_u\hat{L}_i$
    end if
    $i \leftarrow i + 1$
  end while
  $L_{k+1} \leftarrow \hat{L}_i$, $x_{k+1} \leftarrow \hat{x}_i$, $\alpha_k \leftarrow \hat{\alpha}_{i-1}$,
  $y_k \leftarrow \hat{y}_{i-1}$, $i \leftarrow 0$, $k \leftarrow k + 1$
end while
Output x_k

update v_k is also computed differently. In the case of our proposed method, we can observe the effect of using the composite reduced gradient. Last, observe that our proposed convergence analysis enables the converge of the proposed method for a wider selection of γ_0. This is different from the existing results presented in [15, Lemma 2.2.4], wherein convergence is established only for $\gamma_0 \in [\mu_{\hat{f}}; 3L_{\hat{f}} + \mu_{\hat{f}}]$. Choosing $\gamma_0 = 0$, also enables the robustness of the initialization of our proposed algorithm with respect to the imperfect knowledge of $\mu_{\hat{f}}$.

3.3 Bounds on the convergence rate

First, we demonstrate that the convergence rate of the minimization process is characterized by the rate that $\lambda_k \to 0$.

Theorem 4. *If we let* $\lambda_0 = 1$ *and* $\lambda_k = \prod_{i=0}^{k-1}(1 - \alpha_i)$*, Algorithm 2 generates a sequence of points* $\{x_k\}_{k=0}^{\infty}$ *such that*

$$F(x_k) - F(x^*) \leq \lambda_k\left[F(x_0) - F(x^*) + \frac{\gamma_0}{2}||x_0 - x^*||^2\right]. \tag{3.20}$$

Since $\lambda_k \to 0$, Theorem 4 suffices to conclude that the iterates generated by our proposed method converge to x^*. We are now ready to estimate the rate at which $\lambda_k \to 0$.

Lemma 8. *For all $k \geq 0$, Algorithm 2 guarantees that*

1. If $\gamma_0 \in [0, \mu_{\hat{f}}[$, then

$$\lambda_k \leq \frac{2\mu_{\hat{f}}}{L_k\left(e^{\frac{k+1}{2}\sqrt{\frac{\mu_{\hat{f}}}{L_k}}} - e^{-\frac{k+1}{2}\sqrt{\frac{\mu_{\hat{f}}}{L_k}}}\right)^2} \leq \frac{2}{(k+1)^2}. \tag{3.21}$$

2. If $\gamma_0 \in [\mu_{\hat{f}}, 3L_0 + \mu_{\hat{f}}]$, then

$$\lambda_k \leq \frac{4\mu_{\hat{f}}}{(\gamma_0 - \mu_{\hat{f}})\left(e^{\frac{k+1}{2}\sqrt{\frac{\mu_{\hat{f}}}{L_k}}} - e^{-\frac{k+1}{2}\sqrt{\frac{\mu_{\hat{f}}}{L_k}}}\right)^2} \leq \frac{4L_k}{(\gamma_0 - \mu_{\hat{f}})(k+1)^2}. \tag{3.22}$$

Contrasting the results in Lemma 8 with their counterpart, i.e., [15, Lemma 2.2.4], we have the following main differences. First, we prove the convergence of the iterates also in the absence of the exact knowledge of the Lipschitz constant. Moreover, we prove the convergence of the minimization process for a wider range of γ_0. Such a finding is important for several reasons: *i)* The proposed method enjoyes a faster theoretical and practical convergence when $\gamma_0 = 0$; *ii)* Setting $\gamma_0 = 0$ provides robustness to the inexact knowledge of $\mu_{\hat{f}}$.

The following lemma yields an upper bound on the distance $F(x_0) - F(x^*)$.

Lemma 9. *Let $F(x)$ be a convex function with composite structure as shown in (2.1). Moreover, let $T_L(y)$ and $r_L(y)$ be computed as given in (3.8) and (3.11), respectively. Then, for any starting point x_0 in the domain of $F(x)$, we have*

$$F(x_0) - F(x^*) \leq \frac{L_0}{2}||x_0 - x^*||^2. \tag{3.23}$$

Combining Lemmas 8 and 9 with Theorem 4, yields the following convergence rate for our proposed method.

Theorem 5. *Algorithm 2 generates a sequence of points such that*

1. If $\gamma_0 \in [0, \mu_{\hat{f}}[$, then

$$F(x_k) - F(x^*) \leq \frac{\mu_{\hat{f}}(L_0 + \gamma_0)||x_0 - x^*||^2}{L_k\left(e^{\frac{k+1}{2}\sqrt{\frac{\mu_{\hat{f}}}{L_k}}} - e^{-\frac{k+1}{2}\sqrt{\frac{\mu_{\hat{f}}}{L_k}}}\right)^2}. \tag{3.24}$$

2. If $\gamma_0 \in [\mu_{\hat{f}}, 3L_0 + \mu_{\hat{f}}]$*, then*

$$F(x_k) - F(x^*) \leq \frac{2\mu_{\hat{f}}(L_0 + \gamma_0)||x_0 - x^*||^2}{(\gamma_0 - \mu_{\hat{f}})\left(e^{\frac{k+1}{2}\sqrt{\frac{\mu}{L_k}}} - e^{-\frac{k+1}{2}\sqrt{\frac{\mu}{L_k}}}\right)^2}. \tag{3.25}$$

Based on Theorem 5, we can observe that our proposed method converges over a wider interval than its counterpart devised for the class of smooth and strongly convex functions. Initializing $\gamma_0 = 0$ guarantees the fastest convergence of the method. Such a result is relevant in the context of practical deployments of the proposed method, since $\mu_{\hat{f}}$ and $L_{\hat{f}}$ are not known and their values need to be derived based on the available data. The convergence rate of the iterates is also dependent on the value of L_0. Based on (3.24) and (3.25), we can see that choosing small values for L_0 enables faster convergence of the proposed method.

4. Generalizing the estimating sequences framework for problems with composite objectives

In this Chapter, we further extend the results presented in Chapters 2 and 3. So far, we have proposed estimating sequences constructions that extend in different directions. In Chapter 2, we proposed a new class of generalized estimating sequences that support the embedding of a heavy-ball type of momentum into the classical estimating sequences. Based on the framework introduced in Chapter 2, we established that it is possible to devise a method that enjoys a provably faster convergence rate than FGM. In Chapter 3, we proposed a new class of estimating sequences that can be used for solving optimization problems with composite objectives. Therein, we showed that our proposed black-box method also enjoys the same acceleration as the existing benchmarks among black-box methods, i.e., AMGS and FISTA, however it is more efficient than them. The remaining question of interest relates to exploring the coupling of the frameworks introduced in Chapters 2 and 3.

In the sequel, we present the final class of estimating sequences that we devise in this thesis, which we name generalized composite estimating sequences, and show that they enable the construction of a class of very efficient accelerated algorithms. The main contributions are summarized as follows:

- We introduce a new structure for the estimating functions, which we call the *generalized composite estimating functions*. The proposed estimating functions are constructed by making use of the generalized estimating sequences, which contain a heavy-ball type of momentum embedded in them, together with the gradient mapping technique [19]. Similar to Chapter 3, we use a tighter global lower bound on the objective function than the one obtained from the Taylor series expansion of a convex function.

- We use the proposed generalized composite estimating sequences to devise a new class of accelerated gradient methods, which are also equipped with an efficient backracking line-search technique. Similarly to the

method introduced in Chapter 3, and different from AMGS, our proposed method also requires one projection-like operation per iteration.

- Independent from the knowledge of the true value of the Lipschitz constant, we prove that our proposed method enjoys the accelerated convergence rate.

- We prove that the initialization of our proposed method can be made robust to the imperfect knowledge of the strong convexity parameter. This reduces the computational burden of computing a tight estimate of the strong convexity parameter.

- We also demonstrate numerically the superiority of our proposed method when compared to the existing benchmarks. Such superiority is also visible when the performance of the methods is tested on real-world datasets and when inexact values of the strong convexity parameter and the Lipschitz constant are selected.

4.1 Proposed method

Let us now present the last structure of estimating sequences that we have devised in this thesis.

Definition 3. *The sequences* $\{\Phi_k\}_k$ *and* $\{\lambda_k\}_k$, $\lambda_k \geq 0$, *are called generalized composite estimating sequences of the function* $F(\cdot)$ *defined in* (3.2), *if there exists a sequence of bounded functions* $\{\psi_k\}_k$, $\lambda_k \to 0$ *as* $k \to \infty$, *and* $\forall x \in \mathcal{I}$, $\forall k \geq 0$ *we have*

$$\Phi_k(x) \leq \lambda_k \Phi_0(x) + (1 - \lambda_k)\left(F(x) - \psi_k(x)\right). \tag{4.1}$$

Similar to other estimating sequences, we can use the generalized composite estimating sequences to characterize the convergence rate of the minimization process.

Lemma 10. *If for some sequence* $\{x_k\}_k$ *we have* $F(x_k) \leq \Phi_k^* \triangleq \min_{x \in \mathcal{Q}} \Phi_k(x)$, *then* $F(x_k) - F(x^*) \leq \lambda_k \left[\Phi_0(x^*) - F(x^*)\right] - (1 - \lambda_k)\psi_k(x^*)$, *where* $x^* = \arg\min_{x \in \mathcal{Q}} F(x)$.

To construct our proposed method, we will need the following recurrent definitions of the estimating functions.

Lemma 11. *Assume that there exist sequences* $\{\alpha_k\}_k$, *where* $\alpha_k \in (0, 1)$ $\forall k$, *such that* $\sum_{k=0}^{\infty} \alpha_k = \infty$, $\{\psi_k\}_k$ *with an upper bound* Ψ_k, *such that* $\{\psi_k\}_k \geq 0$ *and an arbitrary sequence* $\{y_k\}_k$. *Furthermore, let* $\psi_0(x) = 0$,

$\lambda_0 = 1$ and assume that the estimates L_k, $\forall k = 0, 1, \ldots$, of the Lipschitz constant $L_{\hat{f}}$ are selected in a way that inequality (3.3) is satisfied for all the iterates x_k and y_k, $\forall k = 0, 1, \ldots$. Then, the sequences $\{\Phi_k\}_k$ and $\{\lambda_k\}_k$, which are defined recursively as

$$\lambda_{k+1} = (1 - \alpha_k)\lambda_k, \tag{4.2}$$

$$\begin{aligned}\Phi_{k+1}(x) = (1 - \alpha_k)\left(\Phi_k(x) + \psi_k(x)\right) - \psi_{k+1}(x) - \Psi_k + \alpha_k\left(\frac{\mu_{\hat{f}}}{2}||x - y_k||^2\right) \\ + \alpha_k\left(F\left(T_{L_k}(y_k) + r_{L_k}(y_k)^T(x - y_k)\right) + \psi_k(x) + \frac{1}{2L_k}||r_{L_k}(y_k)||^2\right),\end{aligned} \tag{4.3}$$

are generalized composite estimating sequences.

Observe that the estimating sequences used for devising FGM in [15, Lemma 2.2.4] are obtained as the special case of our generalized composite estimating sequences when $\tau = 0$ and $\psi_k(x) = 0, \forall k = 0, 1, \ldots$. Similarly, the generalized estimating sequences devised in Chapter 2 give the special case of our proposed generalized composite estimating sequences obtained when $\tau = 0$. Last, the composite estimating sequences presented in Chapter 3 correspond to the special case obtained when $\{\psi_k\}_k = \{0\}_k$. In this sense, the generalized composite estimating sequences presented in this Chapter encompass different variants of estimating sequences presented in the literature.

Considering $\gamma_k \in \mathcal{R}^+$, $v_k \in \mathcal{R}^n$, $\forall k = 0, 1, \ldots$, let us choose $\{\Phi_k\}_k$ as (2.5), $\{\psi_k(x)\}_k$ as (2.7), and $\beta_{i,k}$ as (2.8). Based on these selections, the minimal value of the estimating function introduced in (2.5) is

$$\Phi_k^* = \min_x \Phi_k(x) = \phi_k^* + \frac{\gamma_k}{2}||x_{\Phi_k}^* - v_k||^2 - \sum_{i=1}^{k-1}\frac{\beta_{i,k}\gamma_i}{2}||x_{\Phi_k}^* - v_i||^2, \tag{4.4}$$

where $x_{\Phi_k}^* = \arg\min_x \Phi_k(x)$. The recursive relations for the parameters of $\{\phi_k\}_k$ are presented in the following Lemma.

Lemma 12. *Let $\phi_0(x) = \phi_0^* + \frac{\gamma_0}{2}||x - v_0||^2$, where $\gamma_0 \in \mathcal{R}^+$ and $v_0 \in \mathcal{R}^n$. Then, the process defined in Lemma 11 preserves the canonical form of the function $\{\Phi_k(x)\}_k$ presented in (2.5), where the sequences $\{\gamma_k\}_k$, $\{v_k\}_k$ and $\{\phi_k^*\}k$ can be computed as follows*

$$\gamma_{k+1} = (1 - \alpha_k)\gamma_k + \alpha_k\left(\mu_{\hat{f}} + \sum_{i=1}^{k-1}\beta_{i,k}\gamma_i\right), \tag{4.5}$$

$$v_{k+1} = \frac{1}{\gamma_{k+1}}\left((1-\alpha_k)\gamma_k v_k + \alpha_k\left(\mu_{\hat{f}} y_k + \sum_{i=1}^{k-1}\beta_{i,k}\gamma_i - L(y_k - T_{L_k}(y_k))\right)\right), \tag{4.6}$$

$$\phi_{k+1}^* = (1 - \alpha_k)\phi_k^* + \alpha_k\left(F\left(T_{L_k}(y_k)\right) + \frac{1}{2L_k}||r_{L_k}(y_k)||^2 + \sum_{i=1}^{k-1}\beta_{i,k}\gamma_i||y_k - v_i||^2\right)$$

$$
\begin{aligned}
&-\frac{L_k^2\alpha_k^2}{2\gamma_{k+1}}||y_k-T_{L_k}(y_k)||^2+\frac{\alpha_k\gamma_k(1-\alpha_k)\left(\mu_{\hat{f}}+\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i\right)}{2\gamma_{k+1}}||y_k-v_k||^2\\
&+\frac{(1-\alpha_k)\gamma_k}{\gamma_{k+1}}||x^*_{\Phi_k}-v_k||^2+\sum_{i=1}^{k}\frac{\beta_{i,k+1}\gamma_i}{2}||x^*_{\Phi_{k+1}}-v_i||^2\\
&+\frac{\alpha_k^2(1-\alpha_k)}{\gamma_{k+1}}\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i(v_i-y_k)^T r_{L_k}(y_k)\\
&+\frac{\alpha_k^3}{\gamma_{k+1}}\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i||v_i-y_k||\;||r_{L_k}(y_k)||\\
&+\frac{\alpha_k\gamma_k(1-\alpha_k)}{\gamma_{k+1}}\left((v_k-y_k)^T r_{L_k}(y_k)+\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i||y_k-v_i||\;||y_k-v_k||\right).
\end{aligned}
\tag{4.7}
$$

Comparing between our results presented in Lemma 12 to that of [15, Lemma 2.2.3], we can highlight that the recursive relations obtained for computing the terms $\{v_k\}_k$ and $\{\phi_k^*\}_k$ now reflect the usage of a new lower bound on the function that is being minimized. The impact of using the proposed reduced composite gradient is also visible. Moreover, observe that the computation of the terms $\{\gamma_k\}_k$, $\{v_k\}_k$, and $\{\phi_k^*\}_k$ highlight the presence of the heavy-ball type of momentum term that was used to construct them. Comparing the above obtained results to the ones devised in Chapter 2, we can observe the presence of the subgradients of the objective function together with the multistep nature of our newly obtained method. Last, different from the results highlighted in Chapter 3, we can observe the additional terms coming from the newly introduced heavy-ball type of momentum.

Similar to the previous Chapters, we will devise our proposed method by using an induction-based argument. Suppose that at step k we have

$$
\Phi_k^* \overset{(4.4)}{=} \phi_k^*+\frac{\gamma_k}{2}||x^*_{\Phi_k}-v_k||^2-\sum_{i=1}^{k-1}\frac{\beta_{i,k}\gamma_i}{2}||x^*_{\Phi_k}-v_i||^2\geq F(x_k). \tag{4.8}
$$

We need to prove that $\Phi_{k+1}^*\geq F(x_{k+1})$. Using (4.8) and (3.9) in (4.7), we obtain

$$
\begin{aligned}
\phi_{k+1}^*\geq{}&(1-\alpha_k)F(x_k)+\alpha_k\left(F\left(T_{L_k}(y_k)\right)+\frac{1}{2L_k}||r_{L_k}(y_k)||^2+\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i||y_k-v_i||^2\right)\\
&-\frac{L_k^2\alpha_k^2}{2\gamma_{k+1}}||y_k-T_{L_k}(y_k)||^2+\frac{\alpha_k\gamma_k(1-\alpha_k)\left(\mu_{\hat{f}}+\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i\right)}{2\gamma_{k+1}}||y_k-v_k||^2\\
&+\frac{(1-\alpha_k)\gamma_k}{\gamma_{k+1}}||x^*_{\Phi_k}-v_k||^2+\sum_{i=1}^{k}\frac{\beta_{i,k+1}\gamma_i}{2}||x^*_{\Phi_{k+1}}-v_k||^2
\end{aligned}
$$

$$+\frac{\alpha_k^2(1-\alpha_k)}{\gamma_{k+1}}\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i(v_i-y_k)^T r_{L_k}(y_k)$$
$$+\frac{\alpha_k^3}{\gamma_{k+1}}\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i||v_i-y_k||||r_{L_k}(y_k)||$$
$$+\frac{\alpha_k\gamma_k(1-\alpha_k)}{\gamma_{k+1}}\left((v_k-y_k)^T r_{L_k}(y_k)+\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i||y_k-v_i||||y_k-v_k||\right). \tag{4.9}$$

Substituting (3.12) into (4.9), we have

$$\phi_{k+1}^* \geq (1-\alpha_k)\left(F(T_{L_k}(y_k))+r_{L_k}(y_k)^T(x_k-y_k)+\frac{\mu}{2}||x_k-y_k||^2+\frac{1}{2L_k}||r_{L_k}(y_k)||^2\right)$$
$$+\alpha_k\left(F\left(T_{L_k}(y_k)\right)+\frac{1}{2L_k}||r_{L_k}(y_k)||^2+\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i||y_k-v_i||^2\right)$$
$$-\frac{L_k^2\alpha_k^2}{2\gamma_{k+1}}||y_k-T_{L_k}(y_k)||^2+\frac{\alpha_k\gamma_k(1-\alpha_k)\left(\mu_{\hat{f}}+\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i\right)}{2\gamma_{k+1}}||y_k-v_k||^2$$
$$+\frac{(1-\alpha_k)\gamma_k}{\gamma_{k+1}}||x_{\Phi_k}^*-v_k||^2+\sum_{i=1}^{k}\frac{\beta_{i,k+1}\gamma_i}{2}||x_{\Phi_{k+1}}^*-v_k||^2$$
$$+\frac{\alpha_k^2(1-\alpha_k)}{\gamma_{k+1}}\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i(v_i-y_k)^T r_{L_k}(y_k)$$
$$+\frac{\alpha_k^3}{\gamma_{k+1}}\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i||v_i-y_k||||r_{L_k}(y_k)||$$
$$+\frac{\alpha_k\gamma_k(1-\alpha_k)}{\gamma_{k+1}}\left((v_k-y_k)^T r_{L_k}(y_k)+\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i||y_k-v_i||||y_k-v_k||\right). \tag{4.10}$$

Making some manipulations in (4.10), we reach

$$\phi_{k+1}^* \geq F(T_{L_k}(y_k))+(1-\alpha_k)r_{L_k}(y_k)^T(x_k-y_k)+\sum_{i=1}^{k}\frac{\beta_{i,k+1}\gamma_i}{2}||x_{\Phi_{k+1}}^*-v_i||^2$$
$$+\left(\frac{1}{2L_k}-\frac{\alpha_k^2}{2\gamma_{k+1}}\right)||r_{L_k}(y_k)||^2+\frac{\alpha_k^2(1-\alpha_k)}{\gamma_{k+1}}\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i(v_i-y_k)^T r_{L_k}(y_k)$$
$$+\frac{\alpha_k\gamma_k(1-\alpha_k)}{\gamma_{k+1}}(v_k-y_k)^T r_{L_k}(y_k). \tag{4.11}$$

Next, we add $\frac{\gamma_{k+1}}{2}||x_{\Phi_{k+1}}^*-v_{k+1}||^2$ to the LHS of (4.11) and move $\sum_{i=1}^{k}\frac{\beta_{i,k+1}\gamma_i}{2}$

$||x^*_{\Phi_{k+1}} - v_i||^2$ to the LHS of (4.11). This results in

$$\Phi^*_{k+1} \geq F(T_{L_k}(y_k)) + (1-\alpha_k) r_{L_k}(y_k)^T (x_k - y_k) + \left(\frac{1}{2L_k} - \frac{\alpha_k^2}{2\gamma_{k+1}}\right) ||r_{L_k}(y_k)||^2 + \frac{\alpha_k^2(1-\alpha_k)}{\gamma_{k+1}} \sum_{i=1}^{k-1} \beta_{i,k}\gamma_i (v_i - y_k)^T r_{L_k}(y_k) + \frac{\alpha_k\gamma_k(1-\alpha_k)}{\gamma_{k+1}} (v_k - y_k)^T r_{L_k}(y_k). \tag{4.12}$$

We can simplify (4.12) by letting

$$\alpha_k = \sqrt{\frac{\gamma_{k+1}}{L_k}}. \tag{4.13}$$

Plugging (4.5) into (4.13), yields the following recurrent relation for α_k

$$\alpha_k = \frac{\mu_{\hat{f}} + \sum_{i=1}^{k-1} \beta_{i,k}\gamma_i - \gamma_k + \sqrt{\left(\mu_{\hat{f}} + \sum_{i=1}^{k-1} \beta_{i,k}\gamma_i - \gamma_k\right)^2 + 4L_k\gamma_k}}{2L_k}. \tag{4.14}$$

Thus, we can re-write (4.12) as

$$\Phi^*_{k+1} \geq F(T_{L_k}(y_k)) + (1-\alpha_k) r_{L_k}(y_k)^T (x_k - y_k) + \frac{\alpha_k^2(1-\alpha_k)}{\gamma_{k+1}} \sum_{i=1}^{k-1} \beta_{i,k}\gamma_i (v_i - y_k)^T r_L(y_k) + \frac{\alpha_k\gamma_k(1-\alpha_k)}{\gamma_{k+1}} (v_k - y_k)^T r_L(y_k).$$

Letting

$$x_k - y_k + \frac{\alpha_k\gamma_k}{\gamma_{k+1}}(v_k - y_k) + \frac{\alpha_k^2}{\gamma_{k+1}} \sum_{i=1}^{k-1} \beta_{i,k}\gamma_i (v_i - y_k) = 0, \tag{4.15}$$

results in the following recurrent relation for y_k.

$$y_k = \frac{\gamma_{k+1}x_k + \alpha_k\gamma_k v_k + \alpha_k^2 \sum_{i=1}^{k-1} \beta_{i,k}\gamma_i v_i}{\gamma_{k+1} + \alpha_k\gamma_k + \alpha_k^2 \sum_{i=1}^{k-1} \beta_{i,k}\gamma_i}. \tag{4.16}$$

Finally, to establish $\Phi^*_{k+1} \geq F(x_{k+1})$, we can now simply let $x_{k+1} = T_{L_k}(y_k)$. Our proposed method is summarized in Algorithm 3.

Comparing between our proposed method and FGM, we highlight based on lines 6 and 7 in Algorithm 3, that the updates for α_k and γ_k are now computed differently from the ones for FGM. For our proposed method, their values exhibit dependency on the heavy-ball type of momentum term that is utilized in building the estimating sequences. The update for y_k is also computed differently. Furthermore, the value is not dependent on $\mu_{\hat{f}}$. Another significant difference is the update for x_k, which is now obtained through a proximal gradient step. The last difference can be observed from the update of v_k, whose value now reflects our selected subgradient. Further, comparing between our proposed method in this

Algorithm 3 Proposed Method

Input $x_0 \in \mathcal{R}^n$, $L_0 > 0$, $\mu_{\hat{f}}$, $\gamma_0 \in [0, \mu_{\hat{f}}[\cup[2\mu_{\hat{f}}, 3L_0 + \mu_{\hat{f}}]$, $\eta_u > 1$ **and** $\eta_d \in]0, 1[$.
Set $k = 0$, $i = 0$ **and** $v_0 = x_0$.
while $k \leq K_{\max}$ **do**
  $\hat{L}_i \leftarrow \eta_d L_k$
  while True **do**
    $\hat{\alpha}_i \leftarrow \frac{\mu_{\hat{f}} + \sum_{i=1}^{k-1} \beta_{i,k}\hat{\gamma}_i - \gamma_k + \sqrt{\left(\mu_{\hat{f}} + \sum_{i=1}^{k-1} \beta_{i,k}\hat{\gamma}_i - \gamma_k\right)^2 + 4\hat{L}_i\gamma_k}}{2\hat{L}_i}$
    $\hat{\gamma}_{i+1} \leftarrow (1 - \hat{\alpha}_i)\gamma_k + \hat{\alpha}_i\left(\mu_{\hat{f}} + \sum_{i=1}^{k-1} \beta_{i,k}\hat{\gamma}_i\right)$
    $\hat{y}_i \leftarrow \frac{\hat{\gamma}_{i+1}x_k + \hat{\alpha}_i\gamma_k v_k + \hat{\alpha}_i^2 \sum_{i=1}^{k-1} \beta_{i,k}\hat{\gamma}_i v_i}{\hat{\gamma}_{i+1} + \hat{\alpha}_i\gamma_k + \hat{\alpha}_i^2 \sum_{i=1}^{k-1} \beta_{i,k}\hat{\gamma}_i}$
    $\hat{x}_{i+1} \leftarrow \mathbf{prox}_{\frac{1}{\hat{L}_i}\hat{g}}\left(\hat{y}_i - \frac{1}{\hat{L}_i}\nabla f(\hat{y}_i)\right)$
    $\hat{v}_{i+1} \leftarrow \frac{1}{\hat{\gamma}_{i+1}}\left((1 - \hat{\alpha}_i)\gamma_k v_k + \hat{\alpha}_i\left(\mu_{\hat{f}}\hat{y}_i + \sum_{i=1}^{k-1} \beta_{i,k}\hat{\gamma}_i - \hat{L}_i\left(\hat{y}_i - \hat{x}_{i+1}\right)\right)\right)$
    if $F(\hat{x}_{i+1}) \leq m_{\hat{L}_i}(\hat{y}_i, \hat{x}_{i+1})$ **then**
      Break from loop
    else
      $\hat{L}_{i+1} \leftarrow \eta_u \hat{L}_i$
    end if
    $i \leftarrow i + 1$
  end while
  $L_{k+1} \leftarrow \hat{L}_i$, $x_{k+1} \leftarrow \hat{x}_i$, $\alpha_k \leftarrow \hat{\alpha}_{i-1}$, $y_k \leftarrow \hat{y}_{i-1}$, $\gamma_{k+1} \leftarrow \hat{\gamma}_i$, $v_{k+1} \leftarrow \hat{v}_i$, $i \leftarrow 0$, $k \leftarrow k + 1$
end while
Output x_k

Chapter and the one presented in Chapter 2, we highlight the differences coming due to the usage of the proposed subgradient of the objective function and due to the multistep structure of our proposed generalized composite estimating sequences. Last, comparing between the method proposed in this Chapter and the one presented in Chapter 3, we can see that the biggest differences arise from the usage of the heavy-ball type of momentum term. Observe that the recursive relations obtained for our method presented in Algorithm 3 reduce to the ones obtained for FGM when $\tau = 0$ and $\psi_k(x) = 0, \forall k = 0, 1, \ldots$. Further, observe that our method presented in Algorithm 3 reduces to the method introduced in Chapter 2 when $\tau = 0$. Last, observe that our method presented in Algorithm 3 reduces to the one presented in Chapter 3 when $\psi_k(x) = 0, \forall k = 0, 1, \ldots$. In this sense, the method presented in Algorithm 3 is a generalization of all the aforementioned algorithms.

4.2 Convergence Analysis

Based on Lemma 10, we can deduce that the convergence rate of the minimization process is dependent on the sequences $\{\lambda_k\}_k$ and $\{\psi_k\}_k$. This is clarified in the following Theorem.

Theorem 6. *If we let $\lambda_0 = 1$ and $\lambda_k = \prod_{i=0}^{k-1}(1-\alpha_i)$, Algorithm 3 generates a sequence of points $\{x_k\}_k$ such that*

$$F(x_k) - F(x^*) \leq \lambda_k \left(F(x_0) - F(x^*) + \frac{\gamma_0}{2}||x_0 - x^*||^2 \right) - (1-\lambda_k)\psi_k(x). \tag{4.17}$$

The rate of convergence for $\{\lambda_k\}_k$ is characterized in the sequel.

Lemma 13. *For all $k \geq 0$, Algorithm 3 guarantees that*

1. If $\gamma_0 \in [0, \mu_{\hat{f}}[$, then

$$\lambda_k \leq \frac{2\mu_{\hat{f}}}{L_k \left(e^{\frac{k+1}{2}\sqrt{\frac{\left(\mu_{\hat{f}} + \sum_{i=1}^{k-1}\beta_{i,k}\gamma_i\right)}{L_k}}} - e^{-\frac{k+1}{2}\sqrt{\frac{\left(\mu_{\hat{f}} + \sum_{i=1}^{k-1}\beta_{i,k}\gamma_i\right)}{L_k}}} \right)^2} \leq \frac{2}{(k+1)^2}. \tag{4.18}$$

2. If $\gamma_0 \in [2\mu_{\hat{f}}, 3L_0 + \mu_{\hat{f}}]$, then

$$\begin{aligned} \lambda_k &\leq \frac{4\mu_{\hat{f}}}{(\gamma_0 - \mu_{\hat{f}}) \left(e^{\frac{k+1}{2}\sqrt{\frac{\left(\mu_{\hat{f}} + \sum_{i=1}^{k-1}\beta_{i,k}\gamma_i\right)}{L_k}}} - e^{-\frac{k+1}{2}\sqrt{\frac{\left(\mu_{\hat{f}} + \sum_{i=1}^{k-1}\beta_{i,k}\gamma_i\right)}{L_k}}} \right)^2} \\ &\leq \frac{4L_k}{(\gamma_0 - \mu_{\hat{f}})(k+1)^2}. \end{aligned} \tag{4.19}$$

Comparing to [15, Lemma 2.2.4], the results obtained in this Chapter in Lemma 13 highlight the following benefits: *i)*: The method presented in Algorithm 3 converges also when the exact value of $L_{\hat{f}}$ is not known. *ii)* The method presented in Algorithm 3 converges for a wider range of γ_0. Such finding is important as it suggests that the initialization of our method presented in Algorithm 3 is robust to the inexact knowledge of $\mu_{\hat{f}}$.

To establish the accelerated convergence rate of our method presented in Algorithm 3 it suffices to combine Lemma 13 and Theorem 3 with Theorem 6 to come to the following finalizing theorems.

Theorem 7. *Algorithm 3 generates a sequence of points such that*

1. If $\gamma_0 \in [0, \mu_{\hat{f}}[$*, then*

$$F(x_k) - F(x^*) \leq \frac{\mu_{\hat{f}}(L_0 + \gamma_0)||x_0 - x^*||^2}{L_k \left(e^{\frac{k+1}{2}\sqrt{\frac{\mu_{\hat{f}}}{L_k}}} - e^{-\frac{k+1}{2}\sqrt{\frac{\mu_{\hat{f}}}{L_k}}} \right)^2}. \tag{4.20}$$

2. If $\gamma_0 \in [2\mu_{\hat{f}}, 3L_0 + \mu_{\hat{f}}]$*, then*

$$F(x_k) - F(x^*) \leq \frac{2\mu_{\hat{f}}(L_0 + \gamma_0)||x_0 - x^*||^2}{(\gamma_0 - \mu_{\hat{f}}) \left(e^{\frac{k+1}{2}\sqrt{\frac{\mu}{L_k}}} - e^{-\frac{k+1}{2}\sqrt{\frac{\mu}{L_k}}} \right)^2}. \tag{4.21}$$

5. Conclusion

In this Thesis we have presented several accelerated first-order estimating sequence methods that can be used for minimizing different classes of convex functions. In Chapter 2, we have established the generalized estimating sequences framework, and have shown that it enables the coexistence of two different fundamental principles of accelerating first-order methods, i.e., the heavy-ball momentum and Nesterov's acceleration in a single algorithm. We have proven that coupling the two acceleration principles results in schemes that minimize regularized versions of the objective function. For our proposed method, we have proven a faster convergence than FGM, and have demonstrated through our publications the achievability of our theoretical findings also in terms of computational experiments. In Chapter 3 we have presented the class of composite estimating sequences and have shown that they can be used to devise efficient accelerated methods for minimizing convex function with composite structure. Then, in Chapter 4, we have introduced the generalized composite estimating sequences, which encompass all the previously introduced classes of estimating sequences. These estimating sequences have also been used to define an accelerated gradient-based method, which is more efficient than the existing benchmarks. Based on the convergence results presented in this Thesis, we have established that for non-strongly convex problems our proposed methods retains the $\mathcal{O}(1/k^2)$ convergence rate. However, for arbitrarily small values of the strong convexity parameter, our proposed methods exhibit an accelerated linear convergence rate. Moreover, different from classical FGM-type of methods, the initialization of our proposed methods can be made robust to the imperfect knowledge of the strong convexity parameter. Moreover, for the methods presented in Chapters 3 and 4, we have also introduced an efficient backtracking line search strategy.

We now conclude this Thesis by introducing several open problems that arise based on our newly introduced framework.

- Several open questions relate to the selection of the structure for the

terms $\{\psi_k(x)\}_k$ and the choice of the coefficients $\beta_{i,k}$. Obtaining more efficient constructions for these terms can be used to devise more efficient first-order methods. It would also be of interest to evaluate their impact in designing methods that are optimal in decreasing the norm of the gradient for the case of smooth objective functions. Devising such methods is particularly riveting in the context of nonconvex optimization [67, 77, 78], which aim to find stationary points of the objective function.

- Finding alternative constructions for $\psi_k(x)$ would also be of interest, both in the context of black-box optimization and beyond. A related concept is introduced in [79], wherein the authors develop the notions of relative smoothness and relative strong convexity. Considering twice differentiable functions, the relative smoothness and strong convexity parameters are influenced by the weighted difference of the Hessians of the objective function with a differentiable and convex reference function [79, Proposition 1.1]. In this Thesis, we used a similar approach in establishing our estimating functions, with the main difference being that our proposed construction for $\psi_k(x)$ is dynamically changing over iterations. From the perspective of the framework introduced in [79], our selection of the coefficients $\beta_{i,k}$ suggests that the relative strong convexity parameter between $f(x)$ and $\psi_k(x)$ is not unique. As a mater of fact, it is contained in an interval which diminishes as the value of k increases, and as $k \to \infty$, it is restrained in $[0, 1]$. Thus, it is desirable to assess the co-existence aspects of these frameworks.

- In practice, the performance of FGM-type methods can be improved by restarting them. Several restarting conditions have been presented in the literature [80, 81]. It is of interest to assess if similar conditions can be devised for our proposed algorithm and measure the improvements in their performance. In this Thesis, we purposely avoided making use of heuristic approaches such as restarting for further improving the efficiency of our proposed methods. Nevertheless, we believe that it would be beneficial to devise restarting conditions applicable to our proposed methods.

- We also think that it would be relevant to extend our proposed frameworks to broader optimization setups, such as nonconvex, stochastic and distributed optimization. We have already discussed in the thesis that several extensions of the estimating sequences framework used for devising FGM have already been presented in the literature. Considering the gains observed for the foundational setups, we believe that it would be of interest to extend our proposed estimating sequences to such optimization setups.

Bibliography

[1] M. Najafabadi, F. Villanustre, T. Khoshgoftaar, N. Seliya, R. Wald and E. Muharemagic, "Deep learning applications and challenges in big data analytics," *ournal of big data*, vol. 2, no. 1, pp. 1–21, Dec. 2015.

[2] K. Slavakis, G. B. Giannakis and G. Mateos, "Modeling and optimization for big data analytics: (Statistical) learning tools for our era of data deluge," *IEEE Signal Processing Magazine*, vol. 31, no. 5, pp. 18–31, Aug. 2014.

[3] S. Athey, "The economics of artificial intelligence: An agenda," *University of Chicago Press*, pp. 507–547, Jan. 2018.

[4] C. Rudin and K. L. Wagstaff, "Machine learning for science and society," *Machine Learning*, vol. 95, no. 2, pp. 1–9, Apr. 2014.

[5] S. Eisenberg, "Chapter 3 - Machine Learning for the Government: Challenges and Statistical Difficulties, in Federal Data Science" *Academic Press*, pp. 29–40, Jan. 2018.

[6] I. Bose, and R. K. Mahapatra, "Business data mining—a machine learning perspective," *Information & management*, vol. 39, no. 3, pp. 211–225, Dec. 2001.

[7] R. C. Deo, "Machine learning in medicine," *Circulation*, vol. 132, no. 20, pp. 1920–1930, Nov. 2015.

[8] L. Ma and B. Sun, "Machine learning and AI in marketing–Connecting computing power to human insights," *International Journal of Research in Marketing*, vol. 37, no. 3, pp. 481–504, Sep. 2020.

[9] M. F. Dixon, I. Halperin and P. Bilokon "Machine learning in finance" *Springer*, vol. 1170, 2020.

[10] T. C. Silva and L. Zhao "Machine learning in complex networks" *Springer*, 2016.

[11] M. Aggarwal and N. M. Murty "Machine learning in social networks: embedding nodes, edges, communities, and graphs" *Springer Nature*, vol 1, 2020.

[12] S. M. Ibrahim, W. Dong and Q. Yang, "Machine learning driven smart electric power systems: Current trends and new perspectives," *Applied Energy*, vol. 272, pp. 115237, Aug. 2020.

[13] R. Garg, H. Aggarwal, P. Centobelli and R. Cerchione, "Extracting knowledge from big data for sustainability: A comparison of machine learning techniques," *Sustainability*, vol. 11, no. 23, pp. 6669, Nov. 2019.

[14] R. Pugliese, S. Regondi and R. Marini, "Machine learning-based approach: Global trends, research directions, and regulatory standpoints," *Data Science and Management*, vol. 4, pp. 19–29, Dec. 2021.

[15] Y. Nesterov, *Lectures on convex optimization*. Springer, vol. 137, Dec. 2018.

[16] M. Shehab, H. Alves, E. Jorswieck, E. Dosti and M. Latva-aho, "Effective energy efficiency of ultrareliable low-latency communication," *IEEE Internet of Things Journal*, vol. 8, no. 14 pp. 11135–11149, Jan. 2021.

[17] E. Dosti, T. Charalambous and R. Wichman, "Power Allocation for ARQ Two-Hop Cooperative Networks for Ultra-Reliable Communication," *IEEE International Conference on Communications (ICC)*, Jun. 2020, pp. 1–7.

[18] J. Nocedal and S. Wright, *Numerical optimization*. Springer, Aug. 1999.

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

[20] S. Boyd and L. Vandenberghe, *Convex optimization*, Cambridge University Press, Mar. 2004.

[21] S. Boyd and L. Vandenberghe, *Vectors, matrices, and least squares*, Working draft, available: stanford. edu/class/ee103/mma. pdf, 2016.

[22] I. Goodfellow, Y. Bengio and A. Courville, *Deep Learning*, MIT Press, Nov. 2016.

[23] E. Dosti, U. L. Wijewardhana, H. Alves and M. Latva-aho, "Ultra reliable communication via optimum power allocation for type-I ARQ in finite block-length," *IEEE International Conference on Communications (ICC)*, Paris, France, Jun. 2017, pp. 1–6.

[24] K. R. Kesari and J. Honorio, "First order methods take exponential time to converge to global minimizers of non-convex functions," *IEEE International Symposium on Information Theory* , Melbourne, Australia, Jul. 2021, pp. 2322-2327.

[25] A. d'Aspremont, D. Scieur, and A. Taylor, *Acceleration Methods*. Foundations and Trends in Optimization, Now Publisher, Jan. 2021.

[26] A. Beck, *First-order Methods in Optimization*. SIAM, vol. 25, Oct. 2017.

[27] M. S. Ibrahim, A. Konar and N. D. Sidiropoulos, "Fast algorithms for joint multicast beamforming and antenna selection in massive MIMO," *IEEE Transactions on Signal Processing*, vol. 68, pp. 1897–1909, Mar. 2020.

[28] R. Gu and A. Dogandžić, "Projected Nesterov's proximal-gradient algorithm for sparse signal recovery," *IEEE Transactions on Signal Processing*, vol. 65, no. 13, pp. 3510–3525, May. 2017.

[29] M. Raginsky and A. Rakhlin, "Information-based complexity, feedback and dynamics in convex programming," *IEEE Transactions on Information Theory*, vol. 57, no. 10, pp. 7036-7056, Oct. 2011.

[30] A. Agarwal, P. L. Bartlett, P. Ravikumar and M. J. Wainwright, "Information-theoretic lower bounds on the oracle complexity of stochastic convex optimization," *IEEE Transactions on Information Theory*, vol. 58, no. 5, pp. 3235-3249, May. 2012.

[31] M. Shehab, E. Dosti, H. Alves and M. Latva-aho, "On the effective capacity of MTC networks in the finite blocklength regime," *European Conference on Networks and Communications (EuCNC)*, Oulu, Finland, Jun. 2017, pp. 1–5.

[32] E. Dosti, M. Shehab, H. Alves and M. Latva-aho, "Ultra reliable communication via CC-HARQ in finite block-length," *European Conference on Networks and Communications (EuCNC)*, Oulu, Finland, Jun. 2017, pp. 1–5.

[33] Y. Nesterov, "Subgradient methods for huge-scale optimization problems," *Mathematical Programming*, vol. 146, no. 1, pp. 275–297, Aug. 2014.

[34] A. Pensia, V. Jong and P. L. Loh, "Generalization error bounds for noisy, iterative algorithms," *IEEE International Symposium on Information Theory*, Colorado, USA, Jun. 2018, pp. 546-550.

[35] Y. Nesterov, "A method for solving the convex programming problem with convergence rate $\mathcal{O}(1/k^2)$," *Doklady AN USSR*, vol. 269, pp. 543–547, 1983.

[36] Y. Nesterov, "On an approach to the construction of optimal methods of minimization of smooth convex functions," *Ekonomika i Mateaticheskie Metody*, vol. 24, no. 3, pp. 509–517, Nov. 1988.

[37] M. Baes, "Estimate sequence methods: Extensions and approximations," *Institute for Operations Research, ETH, Zürich, Switzerland*, Aug. 2020.

[38] Y. Nesterov, "Smooth minimization of non-smooth functions," *Mathematical Programming*, vol. 103, no. 1, pp. 127–152, May. 2005.

[39] A. Auslender and M. Teboulle, "Interior Gradient and Proximal Methods for Convex and Conic Optimization," *SIAM Journal on Optimization*, vol. 16, no. 3, pp. 697–725, Jul. 2006.

[40] A. d'Aspremont, "Smooth optimization with approximate gradient," *SIAM Journal on Optimization*, vol. 19, no. 3, pp. 1171–1183, Oct. 2008.

[41] O. Devolder, F. Glineur and Y. Nesterov, "First-order methods of smooth convex optimization with inexact oracle," *Mathematical Programming*, vol. 146, no. 1, pp. 37–75, Aug. 2014.

[42] M. Schmidt, N. L. Roux and F. R. Bach, "Convergence rates of inexact proximal-gradient methods for convex optimization," in *Proc. 25th Annual Conference on Neural Information Processing Systems*, Granada, Spain, Dec. 2011, pp. 1458–1466.

[43] N. Flammarion and F. Bach, "From Averaging to Acceleration, There is Only a Step-size," in *Proc. Conference on Learning Theory*, Paris, France, July 2015, pp. 658–695.

[44] W. Su, S. Boyd and E. J. Candès, "A differential equation for modeling Nesterov's accelerated gradient method: Theory and insights," *Journal of Machine Learning Research*, vol. 17, no. 153, pp. 1–43, Jan. 2016.

[45] A. Wibisono, A. C. Wilson and M. I. Jordan, "A variational perspective on accelerated methods in optimization," *Proceedings of the National Academy of Sciences*, vol. 113, no. 47, pp. E7351–E7358, Nov. 2016.

[46] Z. Allen-Zhu and L. Orecchia, "Linear coupling: An ultimate unification of gradient and mirror descent," *arXiv: 1407.1537*, Nov. 2016.

[47] S. Bubeck, Y. T. Lee and M. Singh, "A geometric alternative to Nesterov's accelerated gradient descent," *arXiv: 1506.08187*, Jun. 2015.

[48] D. Drusvyatskiy, M. Fazel and S. Roy, "An optimal first order method based on optimal quadratic averaging," *SIAM Journal on Optimization*, vol. 28, no. 1, pp. 251–271, Feb. 2018.

[49] L. Lessard, B. Recht and A. Packard, "Analysis and design of optimization algorithms via integral quadratic constraints," *SIAM Journal on Optimization*, vol. 26, no. 1, pp. 57–95, Jan. 2016.

[50] B. Van Scoy, R. A. Freeman and K. M. Lynch, "The fastest known globally convergent first-order method for minimizing strongly convex functions," *IEEE Control Systems Letters*, vol. 2, no. 1, pp. 49-54, Jan. 2018.

[51] A. Taylor and Y. Drori, "An optimal gradient method for smooth strongly convex minimization," *arXiv: 2101.09741*, Jan. 2021.

[52] Y. Drori and M. Teboulle, "Performance of first-order methods for smooth convex minimization: a novel approach," *Mathematical Programming*, vol. 145, no. 1, pp. 451–482, Jun. 2014.

[53] A. B. Taylor, J. M. Hendrickx and F. Glineur, "Smooth strongly convex interpolation and exact worst-case performance of first-order methods," *Mathematical Programming*, vol. 161, no. 1-2, pp. 307—345, Jan. 2014.

[54] D. Kim and J. A. Fessler, "Optimized first-order methods for smooth convex minimization," *Mathematical Programming*, vol. 159, no. 1, pp. 81–107, Sep. 2016.

[55] Y. Drori, "The exact information-based complexity of smooth convex minimization," *Journal of Complexity*, vol. 39, pp. 1—16, Nov. 2016.

[56] S. Ji and J. Ye, "An accelerated gradient method for trace norm minimization," in *Proc of the 26th Annual International Conference on Machine Learning*, Montreal, Canada, Jun. 2009, pp. 457–464.

[57] C. A. Uribe, S. Lee, A. Gasnikov and A. Nedić, "A dual approach for optimal algorithms in distributed optimization over networks," *Proc. Information Theory and Applications Workshop*, California, USA, Feb. 2020, pp. 1–37.

[58] H. Ye, L. Luo, Z. Zhou and T. Zhang, "Multi-consensus decentralized accelerated gradient descent," *arXiv: 2005.00797*, May. 2020.

[59] C. Lin, V. Kostina and B. Hassibi, "Differentially quantized gradient methods," *IEEE Transactions on Information Theory* (Early Access), Apr. 2022.

[60] C. Hu, W. Pan and J. Kwok, "Accelerated gradient methods for stochastic optimization and online learning," in *Proc. 22nd Annual Conference on Neural Information Processing Systems*, Vancouver, Canada, Dec. 2009, pp. 781–789.

[61] G. Lan, "An optimal method for stochastic composite optimization," *Mathematical Programming*, vol. 133, no. 1, pp. 365–397, Jun. 2016.

[62] A. Kulunchakov and J. Mairal, "Estimate Sequences for stochastic composite optimization: Variance reduction, acceleration, and robustness to noise," *Journal of Machine Learning Research*, vol. 21, no. 155, pp. 1–52, Jul. 2020.

[63] H. Zhang and S. Sra, "An estimate sequence for geodesically convex optimization," in *Proc. Conference on Learning Theory*, Stockholm, Sweden, Jul. 2018, pp. 1703–1723.

[64] K. Ahn and S. Sra, "From Nesterov's estimate sequence to Riemannian acceleration," in *Proc. Conference on Learning Theory*, Graz, Austria, Jul. 2020, pp. 88–118.

[65] Y. Nesterov, "Accelerating the cubic regularization of Newton's method on convex problems," *Mathematical Programming*, vol. 112, no. 1, pp. 159–181, Mar. 2008.

[66] Y. Nesterov, "Inexact high-order proximal-point methods with auxiliary search procedure," *SIAM Journal on Optimization*, vol. 31, no. 4, pp. 2807–2828, Nov. 2021.

[67] S. Ghadimi and G. Lan, "Accelerated gradient methods for nonconvex nonlinear and stochastic programming," *Mathematical Programming*, vol. 156, no. 1-2, pp. 59–99, Mar. 2016.

[68] Y. Carmon, J. C. Duchi, O. Hinder and A. Sidford, "Accelerated methods for nonconvex optimization," *SIAM Journal on Optimization*, vol. 28, no. 2, pp. 1751–1772, Jun. 2018.

[69] Y. Nesterov, "Gradient methods for minimizing composite objective function," *Mathematical Programming*, vol. 140, no. 1, pp. 125–161, Aug. 2013.

[70] A. Beck and M. Teboulle, "A fast iterative shrinkage-thresholding algorithm for linear inverse problems," *SIAM Journal on Imaging Sciences*, vol. 2, no. 1, pp. 183–202, Mar. 2009.

[71] E. Dosti, S. A. Vorobyov, and T. Charalambous, "A new class of composite objective multi-step estimating sequence techniques," *arXiv:2111.06763*, Nov. 12, 2021.

[72] M. I. Florea and S. A. Vorobyov, "An accelerated composite gradient method for large-scale composite objective problems," *IEEE Transactions on Signal Processing*, vol. 67, no. 2 pp. 444-–459, Jan. 2019.

[73] M. I. Florea and S. A. Vorobyov, "A generalized accelerated composite gradient method: Uniting Nesterov's fast gradient method and FISTA," *IEEE Transactions on Signal Processing*, vol. 68, pp. 3033–3048, Jul. 2020.

[74] P. Tseng, "On accelerated proximal gradient methods for convex-concave optimization," *submitted to SIAM Journal on Optimization*, May. 2008.

[75] B. T. Polyak, "Some methods of speeding up the convergence of iteration methods," *USSR Computational Mathematics and Mathematical Physics*, vol. 4, no. 5, pp. 1–17, 1964.

[76] N. Parikh, S. Boyd, "Proximal Algorithms," *Foundations and Trends in Optimization*, vol. 1, no. 3, pp. 127–239, Jan. 2014.

[77] Y. Carmon, J. C. Duchi, O. Hinder and A. Sidford, "Lower bounds for finding stationary points II: First-order methods," *Mathematical Programming*, vol. 185, no. 1, pp. 315–355, Jan. 2021.

[78] J. Liang and R. D. C. Monteiro, "An average curvature accelerated composite gradient method for nonconvex smooth composite optimization problems," *SIAM Journal on Optimization*, vol. 31, no. 1, pp. 217–243, Jan. 2021.

[79] H. Lu, R. M. Freund and Y. Nesterov, "Relatively smooth convex optimization by first-order methods, and applications," *SIAM Journal on Optimization*, vol. 28, no. 1, pp. 333–354, Feb. 2021.

[80] B. O'Donoghue and E. J. Candès, "Adaptive restart for accelerated gradient schemes," *Foundations of Computational Mathematics*, vol. 15, no. 3, pp. 715–732, Jun. 2015.

[81] P. Giselsson and S. Boyd, "Monotonicity and restart in fast gradient methods," in *53rd IEEE Conference on Decision and Control*, Los Anngeles, California, Dec. 2015, pp. 5058—5063.

Publication I

E. Dosti, S. A. Vorobyov, T. Charalambous. Generalizing Nesterov's Acceleration Framework by Embedding Momentum Into Estimating Sequences: New Algorithm and Bounds. In *IEEE International Symposium on Information Theory (ISIT)*, Helsinki, Finland, 1506-1511, June 2022.

Generalizing Nesterov's Acceleration Framework by Embedding Momentum Into Estimating Sequences: New Algorithm and Bounds

Endrit Dosti[1], Sergiy A. Vorobyov[1], and Themistoklis Charalambous[1, 2]

[1]School of Electrical Engineering, Aalto University, Espoo, Finland
[2]School of Engineering, University of Cyprus, Nicosia, Cyprus
Email: firstname.lastname@aalto.fi

2022 IEEE International Symposium on Information Theory (ISIT) | 978-1-6654-2159-1/22/$31.00 ©2022 IEEE | DOI: 10.1109/ISIT50566.2022.9834684

***Abstract*—We present a new type of heavy-ball momentum term, which is used to construct a class of generalized estimating sequences. These allow for accelerating the minimization process by exploiting the information accumulated in the previous iterates. Combining a newly introduced momentum term with the estimating sequences framework, we devise, as an example, a new black-box accelerated first-order method for solving smooth unconstrained optimization problems. We prove that the proposed method exhibits an improvement over the rate of the celebrated fast gradient method by at least a factor of $\frac{1}{\sqrt{2}}$, and establish that lower bound on the number of iterations carried through until convergence is $\mathcal{O}\left(\sqrt{\frac{\kappa}{2}}\right)$. Finally, the practical performance benefits of the proposed method are demonstrated by numerical experiments.**

***Index Terms*—estimating sequences, black-box methods, complexity analysis, optimization**

I. Introduction

Large-scale optimization is the basic enabling tool in many areas of information sciences, wherein a large amount of data needs to be processed efficiently. To comply with the complexity requirements, large-scale data processing methods are typically limited by first-order algorithms, which consist of sequential procedures that repeatedly query a black-box oracle for information about the objective function [1], [2]. The oracle can be deterministic, i.e., providing the exact value of the objective function and its gradient, or stochastic. In information theory, the fundamental bounds on the oracle complexity in the presence of deterministic and stochastic oracles have been investigated [2]–[5]. The performance of different first-order methods has also been analyzed in the case of minimizing non-convex objectives [6]. In the context of minimizing smooth and strongly convex objectives, Gradient Descent (GD) converges at the suboptimal rate of $\mathcal{O}(\kappa)$, where κ is the condition number [7]. The rate of GD has been accelerated by the Fast Gradient Method (FGM) [8], which is optimal in view of classic complexity theory for convex optimization [2] and reaches the complexity of $\mathcal{O}\left(\sqrt{\kappa}\right)$. FGM and its variants introduced in [7], [9]–[11] have been successfully applied for solving a myriad of machine learning and data analysis problems [12]–[14].

Understanding the intuition and machinery behind the acceleration principle utilized in FGM is challenging, and many of the recent works have focused on providing new perspectives on it, as well as different reasons behind acceleration [15]–[19]. In [15], the authors have introduced a new accelerated gradient method which is inspired by the ellipsoid method. Using theory from robust control, the convergence rates for FGM have been obtained in [16], [17]. In [18], the continuous time-limit of FGM is modelled as a second-order ordinary differential equation. Another framework for the study and analysis of accelerated gradient methods, which relies on the observation that the worst-case performance of a first-order black-box optimization method is itself a semidefinite program, has appeared in [19]. Within this framework, for the case of strongly-convex problems, optimal methods with faster rates than FGM have been introduced in [20]. However, the complexity of the method proposed therein can significantly exceed that of FGM for the case of ill-conditioned problems (see [20, Table 2]).

The key behind constructing optimal methods is the accumulation of global information of the function that is being minimized [7]. In our framework of interest [21], this is achieved by utilizing the estimating sequences, consisting of the pair $\{\phi_k(x)\}_k$ and $\{\lambda_k\}_k$, which allow for constructing upper bounds around the iterates, and simultaneously measure the convergence rate of the iterates. In the existing estimating sequence methods, the advancement of the iterates at step $k+1$ is done by utilizing only the information available at step k. From the existing results on other principles of acceleration, such as the heavy-ball method [22], it has been shown that accounting for the information contained in the previous iterates improves the performance.

In this work, we show that the original construction of the estimating functions can be generalized by incorporating extra terms that depend on the previous iterates. Within the black-box setting, we introduce a new type of heavy-ball momentum, which is captured by the terms of a new sequence $\{\psi_k\}_k$. Using the newly introduced heavy-ball type of momentum term within the estimating sequences framework, we construct a new method and show that FGM can be obtained as the

special case when the memory terms are not considered. We prove that our proposed method is optimal, and show that it outperforms FGM by at least a factor of $\frac{1}{\sqrt{2}}$. Moreover, we show that the initialization of our proposed method is robust to the imperfect knowledge of μ, which is of a high importance in practice, since accurate estimation of μ is typically computationally expensive.

II. Proposed Method

In this work, we focus on the problem of minimizing the smooth and strongly convex objective function, that is,

$$\underset{x\in\mathcal{R}^n}{\text{minimize}}\ f(x), \tag{1}$$

where $f:\mathcal{R}^n\to\mathcal{R}$ is a μ-strongly convex function with L-Lipschitz continuous gradient defined by a deterministic black-box oracle.

Let us begin by assuming that there exists a procedure that produces points $x\in\mathcal{R}^n$ and let $\mathcal{I}=\textbf{conv}(x_0,x_1,x_2,\ldots,x^*)$ be a closed convex set which is comprised of the convex hull of the finite number of iterates that are formed during the minimization process. Next, we define the generalized estimating sequences as follows.

Definition 1. *The sequences $\{\Phi_k\}_k$, and $\{\lambda_k\}_k$, $\lambda_k\geq 0$, are called generalized estimating sequences of the function $f(\cdot)$, if $\exists\psi_k:\mathcal{I}\to\mathcal{Q}\subset\mathcal{R}^+$, $\lambda_k\to 0$, and $\forall x\in\mathcal{I}$, $\forall k$ we have*

$$\Phi_k(x)\leq\lambda_k\Phi_0(x)+(1-\lambda_k)\left(f(x)-\psi_k(x)\right). \tag{2}$$

Thus, the inclusion of $\psi_k(x)$ in the definition of the estimating sequences allows for embedding additional information that can aid in improving the convergence properties of the methods. As discussed earlier, one of the major benefits of the estimating sequences, is the fact that they allow for estimating the convergence rate of the minimization process. The following Lemma yields a precise characterization.

Lemma 1. *If for some sequence of points $\{x_k\}_k$ we have $f(x_k)\leq\Phi_k^*\triangleq\min_{x\in\mathcal{I}}\Phi_k(x)$, then $f(x_k)-f(x^*)\leq\lambda_k\left[\Phi_0(x^*)-f(x^*)\right]-(1-\lambda_k)\,\psi_k(x^*)$, where $x^*=\arg\min_{x\in\mathcal{R}^n}f(x)$.*

All the proofs of the lemmas and theorems in this short paper can be found in our full paper [24].

Prior to presenting the structure of the estimating sequences that are utilized for constructing our proposed method, let us define the following upper bound on the terms in the sequence $\{\psi_k(x)\}_k$

$$\Psi_k=\begin{cases}\sup\limits_{m\in\{1,2,\ldots k\},x\in\mathcal{I}}\psi_m(x), & \text{if } k>0,\\ 0, & \text{otherwise.}\end{cases} \tag{3}$$

In words, Ψ_k is the supremum of the infinite sequence of finite values of $\psi_m(x)$. An explicit construction for $\psi_m(x)$ will be presented later in the paper. At this point, we can introduce our proposed construction for the generalized estimating sequences.

Lemma 2. *Assume that there exist sequences $\{\alpha_k\}_k$, where $\alpha_k\in(0,1)$ and $\sum_{k=0}^{\infty}\alpha_k=\infty$, and $\{y_k\}_k$, where $y_k\in\mathcal{R}^n$, and a sequence of functions $\{\psi_k\}_k$ such that $\psi_k(x)\geq 0$, $\forall k$. Let $\psi_0(x)=0$ and $\lambda_0=1$. Then, the sequences $\{\Phi_k\}_k$ and $\{\lambda_k\}_k$, which are defined recursively as*

$$\lambda_{k+1}=(1-\alpha_k)\lambda_k, \tag{4}$$

$$\begin{aligned}\Phi_{k+1}(x)=&(1-\alpha_k)(\Phi_k(x)+\psi_k(x))-\psi_{k+1}(x)-\Psi_k+\alpha_k\psi_k(x)\\&+\alpha_k\Big(f(y_k)+\nabla f(y_k)^T(x-y_k)+\frac{\mu}{2}||x-y_k||^2\Big),\end{aligned} \tag{5}$$

are generalized estimating sequences. Here, $(\cdot)^T$ denotes the transposition operator and $||\cdot||$ stands for the Euclidean norm of a vector.

We note that the structure for the terms in the sequence $\{\Phi_k(x)\}_k$ has not been introduced yet. From Lemma 1, we can see that their importance lies in the fact that they allow for constructing upper bounds on $f(x_k)$. As discussed in [7], accelerated methods must make use of some global topological properties of the objective function. This intuition is also asserted by the performance of second-order methods. For instance, in the case of Newton's method, by utilizing the information in the Hessian it is possible to construct ellipsoids around each iterate [23, Fig. 9.19]. These ellipsoids then aid in correcting the search direction. In the case of first-order methods, wherein the Hessian information is not available, we can consider constructing balls in the locality around each iterate x_k, without "discriminating" any search direction. This can be achieved by utilizing a sequence of isotropic scanning functions with scanning radius γ_k, whose Hessian is $\nabla^2\phi_k(x)=\gamma_k I$. The canonical structure of such scanning functions can be written as

$$\phi_k(x)=\phi_k^*+\frac{\gamma_k}{2}||x-v_k||^2,\ \forall k, \tag{6}$$

where ϕ_k^* is the minimal value, $\gamma_k\in\mathcal{R}^+$ is the radius and $v_k\in\mathcal{R}^n$ is the center of the scanning function. We note that this is also the canonical form of scanning function, which has been utilized in constructing FGM [7]. We have already discussed that the goal of the paper is to devise a new class of algorithms, which can benefit from both Nesterov's acceleration and the heavy-ball momentum. To achieve this goal, we introduce the following structure for the scanning function

$$\Phi_k(x)=\phi_k^*+\frac{\gamma_k}{2}||x-v_k||^2-\psi_k(x),\ \forall k, \tag{7}$$

where the heavy-ball type of momentum term $\psi_k(x)$ is

$$\psi_k(x)\triangleq\sum_{i=0}^{k-1}\beta_{i,k}\frac{\gamma_i}{2}||x-v_i||^2,\ \forall k, \tag{8}$$

and

$$\beta_{i,k}=\begin{cases}\min\left(1,\frac{\mu}{\gamma_{k-1}}\right), & \text{if } i=k-1,\\ 0, & \text{otherwise.}\end{cases} \tag{9}$$

Indeed, since in the black-box setting there is no prior information about the function that can be exploited to improve the convergence of the method, we construct the terms in our proposed scanning functions such that they can "self-regulate" by accounting for the information that has been obtained in the previous iterations. Lastly, we note that the terms $\beta_{i,k}$ ensure that only the estimating function constructed at step $k-1$ is accounted, which asserts that the values in the summation given in (8) remain finite.

Let us now formally express that the estimating sequences presented in Lemma 2 preserve the quadratic structure of our scanning functions $\Phi_k(x)$, and show how the terms ϕ_k^*, γ_k and v_k can be calculated. This is depicted in the sequel.

Lemma 3. *Assume that the coefficients $\beta_{i,k}$ are selected according to* (9)*, and let $\Phi_0(x) = \phi_0^* + \frac{\gamma_0}{2}||x-v_0||^2$. Then, the process defined in Lemma 2 preserves the quadratic canonical structure of the scanning function introduced in* (7)*. Moreover, the sequences $\{\gamma_k\}_k$, $\{v_k\}_k$ and $\{\phi_k^*\}_k$ can be computed as given by* (10)*,* (11) *and* (12) *shown at the top of the next page.*

In the sequel, we will select the sequences $\{x_k\}_k$, $\{y_k\}_k$ and $\{v_k\}_k$ such that $f(x_k) \leq \Phi_k^*$, $\forall k$. Let us assume that for some step k, we have $\phi_k^* \geq f(x_k)$. Then, at step $k+1$, relaxing (12), as well as making some manipulations, we can write[1]

$$\begin{aligned}\phi_{k+1}^* \geq\, & f(y_k) + (1-\alpha_k)\nabla f(y_k)^T(x_k - y_k) \\ & - \frac{\alpha_k^2}{2\gamma_{k+1}}||\nabla f(y_k)||^2 + \frac{\alpha_k(1-\alpha_k)\gamma_k}{\gamma_{k+1}}(v_k - y_k)^T\nabla f(y_k) \\ & + (1-\alpha_k)\,\frac{\alpha_k^2}{\gamma_{k+1}}\sum_{i=0}^{k-1}\beta_{i,k}\gamma_i(v_i - y_k)^T\nabla f(y_k).\end{aligned} \tag{13}$$

To satisfy the necessary conditions for Lemma 1, we need to ensure that $\phi_{k+1}^* \geq f(x_{k+1})$. For this reason, let us relax the lower bound even further by utilizing the relation

$$f(y_k) - \frac{1}{2L}||\nabla f(y_k)||^2 \geq f(x_{k+1}), \tag{14}$$

which can be guaranteed by a gradient descent step on y_k [7, Theorem 2.1.5]. Thus, we obtain α_k as

$$\alpha_k = \sqrt{\frac{\gamma_{k+1}}{L}}. \tag{15}$$

Utilizing the recursion (10), we have

$$\begin{aligned}\alpha_k = & \frac{\left(\mu + \sum_{i=1}^{k-1}\beta_{i,k}\gamma_i - \gamma_k\right)}{2L} \\ & + \frac{\sqrt{\left(\mu + \sum_{i=1}^{k-1}\beta_{i,k}\gamma_i - \gamma_k\right)^2 + 4L\gamma_k}}{2L}.\end{aligned} \tag{16}$$

[1]For more detailed derivations see [24].

Algorithm 1: Proposed Method

Choose $x_0 \in \mathcal{R}^n$, set $\gamma_0 = 0$ and $v_0 = x_0$.
while stopping criterion is not met **do**
Compute $\alpha_k \in [0,1]$ as
$\alpha_k = \frac{\left(\mu+\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i-\gamma_k\right)+\sqrt{\left(\mu+\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i-\gamma_k\right)^2+4L\gamma_k}}{2L}$.
Set $\gamma_{k+1} = (1-\alpha_k)\gamma_k + \alpha_k\left(\mu + \sum_{i=0}^{k-1}\beta_{i,k}\gamma_i\right)$.
Choose $y_k = \frac{\gamma_{k+1}x_k + \alpha_k\gamma_k v_k + \alpha_k^2\sum_{i=0}^{k-1}\beta_{i,k}\gamma_i v_i}{\gamma_{k+1}+\alpha_k\gamma_k+\alpha_k^2\sum_{i=0}^{k-1}\beta_{i,k}\gamma_i}$.
Set $x_{k+1} = y_k - \frac{1}{L}\nabla f(y_k)$
Set $v_{k+1} = \frac{1}{\gamma_{k+1}}\Big((1-\alpha_k)\gamma_k v_k + \mu\alpha_k\Big(y_k - \frac{1}{\mu}\nabla f(y_k) + \sum_{i=0}^{k-1}\frac{\beta_{i,k}\gamma_i}{\mu}v_i\Big)\Big)$.
end while

Choosing α_k as given in (16), we can rewrite (13) as

$$\begin{aligned}\phi_{k+1}^* \geq\, & f(x_{k+1}) + (1-\alpha_k)\nabla f(y_k)^T\Big((x_k - y_k) \\ & + \frac{\alpha_k\gamma_k}{\gamma_{k+1}}(v_k - y_k) + \frac{\alpha_k^2}{\gamma_{k+1}}\sum_{i=0}^{k-1}\beta_{i,k}\gamma_i(v_i - y_k)\Big).\end{aligned} \tag{17}$$

Finally, we can obtain the update for $\{y_k\}_k$ by letting

$$x_k - y_k + \frac{\alpha_k\gamma_k}{\gamma_{k+1}}(v_k - y_k) + \frac{\alpha_k^2}{\gamma_{k+1}}\sum_{i=0}^{k-1}\beta_{i,k}\gamma_i(v_i - y_k) = 0.$$

This results in

$$y_k = \frac{\gamma_{k+1}x_k + \alpha_k\gamma_k v_k + \alpha_k^2\sum_{i=0}^{k-1}\beta_{i,k}\gamma_i v_i}{\gamma_{k+1} + \alpha_k\gamma_k + \alpha_k^2\sum_{i=0}^{k-1}\beta_{i,k}\gamma_i}. \tag{18}$$

Our findings are summarized in Algorithm 1.

Comparing Algorithm 1 with [7, (2.2.19)], we can see that the computation of the terms α_k and γ_k reflects the different types of estimating sequences that were used in constructing the methods. This is also reflected in the computation of the points $y_k, \forall k$. Another important difference is the initialization of the parameter γ_0. In the case of FGM, the scheme is guaranteed to converge for $\gamma_0 \in [\mu, 3L+\mu]$. On the other hand, as we will show in Section III, our proposed method is guaranteed to converge even when $\gamma_0 = 0$. This ensures the robustness of the initialization of our proposed method with respect to the imperfect knowledge of the strong convexity parameter μ, whose exact value is difficult to be estimated efficiently in practice. Moreover, in our proposed method, the extra terms contributed from the generalized estimating sequences come up as coefficients of α_k^2. Such additional terms are also observed in the computation of v_{k+1}. Lastly, we note that FGM is obtained by setting the terms $\beta_{i,k} = 0$, $\forall i, k$. Such a result is coherent with the fact that the estimating sequences utilized in constructing FGM are the special case of the generalized estimating sequences that we used in constructing our proposed method, which arises when $\psi_k(x) = 0$, $\forall k$.

III. Bounds on convergence rate

In this section we introduce the main convergence results of our proposed method. Let us begin by showing that the

$$\gamma_{k+1} = (1-\alpha_k)\gamma_k + \alpha_k\left(\mu + \sum_{i=0}^{k-1}\beta_{i,k}\gamma_i\right), \tag{10}$$

$$v_{k+1} = \frac{1}{\gamma_{k+1}}\left((1-\alpha_k)\gamma_k v_k + \mu\alpha_k\left(y_k - \frac{1}{\mu}\nabla f(y_k) + \sum_{i=0}^{k-1}\frac{\beta_{i,k}\gamma_i}{\mu}v_i\right)\right), \tag{11}$$

$$\begin{aligned}\phi^*_{k+1} = {} & \alpha_k f(y_k) + (1-\alpha_k)\phi^*_k + \frac{\alpha_k\gamma_k(1-\alpha_k)(\mu+\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i)}{2\gamma_{k+1}}||y_k - v_k||^2 + \frac{\alpha_k^3}{\gamma_{k+1}}\sum_{i=0}^{k-1}\beta_{i,k}\gamma_i||v_i - y_k||\,||\nabla f(y_k)|| \\ & - \frac{\alpha_k^2||\nabla f(y_k)||^2}{2\gamma_{k+1}} + \frac{\alpha_k(1-\alpha_k)\gamma_k}{\gamma_{k+1}}\left((v_k-y_k)^T\nabla f(y_k) + \sum_{i=0}^{k-1}\beta_{i,k}\gamma_i||y_k - v_i||||y_k - v_k||\right) + (1-\alpha_k)\frac{\gamma_k}{2}||x^*_{\Phi_k} - v_k||^2 \\ & + \alpha_k\sum_{i=0}^{k-1}\frac{\beta_{i,k}\gamma_i}{2}||y_k - v_i||^2 + \frac{(1-\alpha_k)\,\alpha_k^2}{\gamma_{k+1}}\sum_{i=0}^{k-1}\beta_{i,k}\gamma_i(v_i - y_k)^T\nabla f(y_k) + \sum_{i=0}^{k-1}\beta_{i,k}\frac{\gamma_i}{2}||x^*_{\Phi_k} - v_i||^2.\end{aligned} \tag{12}$$

convergence rate of the minimization process will depend on both the sequences $\{\lambda_k\}_k$ and $\{\psi_k\}_k$.

Theorem 1. *If we let $\lambda_0 = 1$ and $\lambda_k = \prod_{i=0}^{k-1}(1-\alpha_i)$, Algorithm 1 generates a sequence of points $\{x_k\}_k$ such that*

$$\begin{aligned}f(x_k) - f^* \le {} & \lambda_k\left[f(x_0) - f(x^*) + \frac{\gamma_0}{2}||x_0 - x^*||^2\right] \\ & - (1-\lambda_k)\psi_k(x^*).\end{aligned} \tag{19}$$

From Lemma 2, we can see that the terms in the sequence $\{\lambda_k\}_k$ will converge to 0 as $k \to \infty$. The bound on the rate at which the terms in the sequence $\{\lambda_k\}_k$ decrease is presented in the following Lemma.

Lemma 4. *For all $k \ge 0$, Algorithm 1 guarantees that*

$$\lambda_k \le \frac{2\mu}{L\left(e^{\frac{k+1}{2}\sqrt{\frac{\mu+\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i}{L}}} - e^{-\frac{k+1}{2}\sqrt{\frac{\mu+\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i}{L}}}\right)^2} \tag{20}$$

$$\le \frac{2\mu}{\left(\mu + \sum_{i=1}^{k-1}\beta_{i,k}\gamma_i\right)(k+1)^2}. \tag{21}$$

The optimality of Algorithm 1 is established next.

Theorem 2. *In Algorithm 1, let $\mu > 0$. Then, the scheme generates a sequence of points such that:*

$$\begin{aligned}f_k - f^* \le {} & \frac{\mu||x_0 - x^*||^2}{\left(e^{\frac{k+1}{2}\sqrt{\frac{\mu+\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i}{L}}} - e^{-\frac{k+1}{2}\sqrt{\frac{\mu+\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i}{L}}}\right)^2} \\ & - (1-\lambda_k)\psi_k(x^*).\end{aligned} \tag{22}$$

where $f_k = f(x_k)$ and $f^* = f(x^*)$. This means that the method is optimal when the accuracy $\epsilon \le \frac{\mu}{2}||x_0 - x^*||^2$.

For the problem of minimizing smooth and strongly convex objective functions, FGM reaches the following bound on the number of iterations [7, (2.2.17)]

$$k_{FGM} \ge \sqrt{\frac{L}{\mu}}\left(\ln\left(\frac{\mu R_0^2}{2\epsilon}\right) + \ln(23/3)\right). \tag{23}$$

On the other hand, when the coefficients $\beta_{i,k}$ are selected as given in (9), the lower bound on the number of iterations for our proposed method becomes

$$k_{Proposed} \ge \sqrt{\frac{L}{\mu + \min(\gamma_{k-1}, \mu)}}\left(\ln\left(\frac{\mu R_0^2}{2\epsilon}\right) + \ln(5)\right). \tag{24}$$

From (24), we can see that the lower bound of the number of iterations is influenced by the increase of the values of the terms in the sequence $\{\gamma_k\}_k$. As we thoroughly establish in our full paper [24], and illustrate numerically in Section IV, the growth of the terms in the sequence $\{\gamma_k\}_k$ is exponential in the iteration counter k, and it converges to 2μ. Therefore, (24) converges to

$$k_{Proposed} \to \sqrt{\frac{L}{2\mu}}\left(\ln\left(\frac{\mu R_0^2}{2\epsilon}\right) + \ln(5)\right). \tag{25}$$

Comparing the convergence rate given in (22), and the lower bound on the number of iterations that need to be carried through until convergence for our proposed method, to the results presented for FGM in (23), we can see the improvement by at least a factor of $1/\sqrt{2}$.

IV. Simulation Results

In this section, we consider the classical task in data analysis of minimizing a regularized quadratic loss function

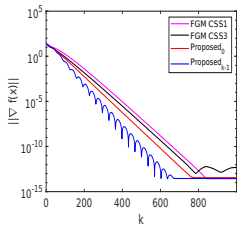
$$\underset{x\in\mathcal{R}^n}{\text{minimize}} \quad \frac{1}{2m}\sum_{i=1}^{m}(a_i^T x - y_i)^2 + \frac{\tau}{2}||x||^2, \tag{26}$$

where $a_i \in \mathcal{R}^n$ is a vector containing the data points, $x \in \mathcal{R}^n$ is a vector consisting of the parameters that need to be estimated, $y_i \in \mathcal{R}$ corresponds to the labels and $\tau \ge 0$ is a

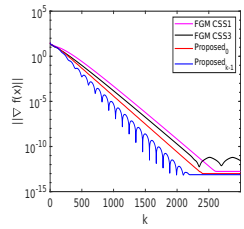
regularization parameter. We benchmark against two instances of FGM Constant Step Scheme I (CSS1), which is obtained when $\gamma_0 = L$ and is referred to as FGM CSS1, as well as $\gamma_0 = \mu$, which also corresponds to FGM CSS3 [7, Chapter 2.2]. For our proposed method, we consider the case $\beta_{0,k} = 1$ and $\beta_{i,k} = 0, \forall i = 1, \dots k$, which is referred to as Proposed$_0$. We note that when $\gamma_0 = 0$, this algorithm corresponds to FGM. However, the original analysis of FGM does not guarantee convergence of the method with this selection of γ_0. Such a fact is important as it ensures the robustness of the initialization of our proposed method with respect to the imperfect knowledge of μ. We also consider the instance obtained when the terms $\beta_{i,k}$ are selected according to (9), which is referred to as Proposed$_{k-1}$. Lastly, the starting point x_0 is randomly selected and all algorithms are initiated in it.

To control the condition number of the data matrix A, we generate a symmetric positive definite diagonal matrix $A \in \mathcal{R}^{mxm}$, whose elements a_{ii} are drawn from the discrete set $\{10^0, 10^{-1}, 10^{-2}, \dots 10^{-\xi}\}$ uniformly at random. This ensures $\kappa = 10^{\xi}$. Moreover, this results in $L = 1$ and $\mu = 10^{-\xi}$. We obtain the elements of the labels vector $y \in \mathcal{R}^m$ by sampling them uniformly at random from the box $[0, 1]^n$. Lastly, to simulate a large and ill-conditioned problem, we set $m = 1000$, $\xi \in \{3, 4\}$ and $\tau \in \{10^{-3}, 10^{-4}\}$. Our results are reported in Fig. 1.

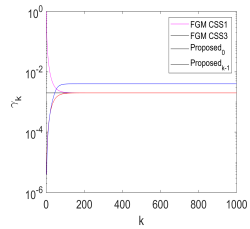
From Fig. 1, we can distinguish the efficiency of our proposed method. First, observe that all instances of the proposed method require a lower number of iterations to decrease the norm of the gradient. Second, we can see that the memoryless version of the proposed method, behaves similar to the considered instances of FGM, however, it exhibits a faster convergence. These results confirm the theoretical findings presented in Section III. Let us now focus on the variant of the proposed method that utilizes the information accumulated at step $k-1$. From Figs. 1(a) and 1(b), we can see that the proposed method with memory term γ_{k-1} converges approximately 30% faster than FGM CSS3, which is the fastest instance of FGM. This result is again coherent with the theoretical asymptotic bound obtained in (25), that also suggests an approximate improvement of 30% over FGM. An important observation can be made from Figs. 1(c) and 1(d), which depict the exponential convergence of the term γ_{k-1} as the iteration counter k grows large. As we already discussed in Section III, this ensures that the convergence to the bound presented in (25) is fast. Lastly, we note that the number of iterations that need to be carried through until convergence, increases significantly with the condition number of the problem. For instance, in the case when $\kappa = 10^3$, Fig. 1(a) depicts that the performance difference between algorithms tested is of the order of hundreds of iterations. Then, as the condition number of the objective function increases to $\kappa = 10^4$, Fig. 1(b) illustrates even larger differences between algorithms, as measured by the number of iterations that need to be carried through until convergence.



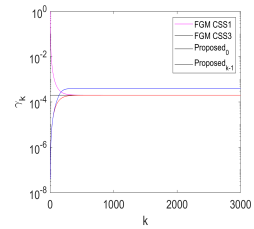
(a) Decreasing the norm of the gradient, $\kappa = 10^3$.



(b) Decreasing the norm of the gradient, $\kappa = 10^4$.



(c) Convergence of $\{\gamma_k\}_{k=0}^{\infty}$, $\kappa = 10^3$.



(d) Convergence of $\{\gamma_k\}_{k=0}^{\infty}$, $\kappa = 10^4$.

Fig. 1. Comparison between various features of interest of the algorithms. The goal is to minimize the quadratic loss function, for which $A \in \mathcal{R}^{1000 \times 1000}$ and its entries are randomly generated.

V. Conclusion and Future Work

A new form of heavy-ball momentum term has been introduced and it has been shown that it can be used to construct a newly introduced class of generalized estimating sequences. This has paved path to constructing a new class of optimal gradient methods. At their core, our proposed method benefits from the co-existence of two fundamentally different acceleration principles, i.e., the heavy-ball momentum and Nesterov's acceleration. We have shown that FGM can be obtained as the special case of our proposed method when the momentum term is not utilized. Moreover, we have managed to prove that the convergence rate, as well as the lower bound on the number of iterations, of the proposed method are better than FGM by at least a factor of $1/\sqrt{2}$. The practical superiority of all instances of the proposed method over FGM has been established throughout the simulations.

As future work, it would be of interest to characterize the amount of information that is captured at each iteration by the terms in the sequence $\{\psi_k(x)\}_k$. We believe that this would pave path to finding the optimal selection of the coefficients $\beta_{i,k}$, which can enable further improvements of the efficiency of the proposed method. It would also be of interest to explore the possibility of finding new constructions for $\{\psi_k(x)\}_k$, which can exploit both black and white box information about the function that is being minimized, and use it to further accelerate the minimization process. Last, it is also of interest to consider extensions of the results presented herein to account for stochastic oracles, as well as investigate extensions to other optimization setups such as the design of new methods for solving nonsmooth and potentially nonconvex problems.

REFERENCES

[1] A. Beck, *First-order Methods in Optimization*. SIAM, vol. 25, Oct. 2017.

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

[3] M. Raginsky and A. Rakhlin, "Information-based complexity, feedback and dynamics in convex programming," *IEEE Transactions on Information Theory*, vol. 57, no. 10, pp. 7036-7056, Oct. 2011.

[4] A. Agarwal, P. L. Bartlett, P. Ravikumar and M. J. Wainwright, "Information-theoretic lower bounds on the oracle complexity of stochastic convex optimization," *IEEE Transactions on Information Theory*, vol. 58, no. 5, pp. 3235-3249, May. 2012.

[5] A. Pensia, V. Jong and P. L. Loh, "Generalization error bounds for noisy, iterative algorithms," *IEEE International Symposium on Information Theory*, Colorado, USA, Jun. 2018, pp. 546-550.

[6] K. R. Kesari and J. Honorio, "First order methods take exponential time to converge to global minimizers of non-convex functions," *IEEE International Symposium on Information Theory* , Melbourne, Australia, Jul. 2021, pp. 2322-2327.

[7] Y. Nesterov, *Lectures on convex optimization*, Springer, vol. 137, Dec. 2018.

[8] Y. Nesterov, "A method for solving the convex programming problem with convergence rate $\mathcal{O}(1/k^2)$," *Doklady AN USSR*, vol. 269, pp. 543–547, 1983.

[9] Y. Nesterov, "Gradient methods for minimizing composite objective function," *Mathematical Programming*, vol. 140, no. 1, pp. 125–161, Aug. 2013.

[10] A. Beck and M. Teboulle, "A fast iterative shrinkage-thresholding algorithm for linear inverse problems," *SIAM Journal on Imaging Sciences*, vol. 2, no. 1, pp. 183–202, Mar. 2009.

[11] M. I. Florea and S. A. Vorobyov, "A generalized accelerated composite gradient method: Uniting Nesterov's fast gradient method and FISTA," *IEEE Transactions on Signal Processing*, vol. 68, pp. 3033–3048, Jul. 2020.

[12] K. Slavakis, G. B. Giannakis and G. Mateos, "Modeling and optimization for big data analytics: (Statistical) learning tools for our era of data deluge," *IEEE Signal Processing Magazine*, vol. 31, no. 5, pp. 18–31, Aug. 2014.

[13] D. Granziol, B. Ru, S. Zohren, X. Dong, M. Osborne, and S. Roberts, "MEMe: An accurate maximum entropy method for efficient approximations in large-scale machine learning," *Entropy*, vol. 21, no. 6, pp. 551, May. 2019.

[14] E. Dosti, S. A. Vorobyov and T. Charalambous, "A New Class of Composite Objective Multi-step Estimating-sequence Techniques (COMET)," *arXiv: 2111.06763*, Nov. 2021.

[15] S. Bubeck, Y. T. Lee and M. Singh, "A geometric alternative to Nesterov's accelerated gradient descent," *arXiv: 1506.08187*, Jun. 2015.

[16] L. Lessard, B. Recht and A. Packard, "Analysis and design of optimization algorithms via integral quadratic constraints," *SIAM Journal on Optimization*, vol. 26, no. 1, pp. 57–95, Jan. 2016.

[17] S. Safavi, B. Joshi, G. França and J. Bento, "An explicit convergence rate for Nesterov's method from SDP," *IEEE International Symposium on Information Theory* , Colorado, USA, Jun. 2018, pp. 1560-1564.

[18] W. Su, S. Boyd and E. J. Candès, "A differential equation for modeling Nesterov's accelerated gradient method: Theory and insights," *Journal of Machine Learning Research*, vol. 17, no. 153, pp. 1–43, Jan. 2016.

[19] Y. Drori and M. Teboulle, "Performance of first-order methods for smooth convex minimization: a novel approach," *Mathematical Programming*, vol. 145, no. 1, pp. 451–482, Jun. 2014.

[20] B. Van Scoy, R. A. Freeman and K. M. Lynch, "The fastest known globally convergent first-order method for minimizing strongly convex functions," *IEEE Control Systems Letters*, vol. 2, no. 1, pp. 49-54, Jan. 2018.

[21] M. Baes, "Estimate sequence methods: Extensions and approximations," *Institute for Operations Research, ETH, Zürich, Switzerland*, Aug. 2020.

[22] B. T. Polyak, "Some methods of speeding up the convergence of iteration methods," *USSR Computational Mathematics and Mathematical Physics*, vol. 4, no. 5, pp. 1–17, 1964.

[23] S. Boyd and L. Vandenberghe, *Convex optimization*, Cambridge University Press, Mar. 2004.

[24] E. Dosti, S. A. Vorobyov and T. Charalambous, "Embedding a heavy-ball type of momentum into the estimating sequences," *arXiv: 2008.07979*, Aug. 2020.

Publication II

E. Dosti, S. A. Vorobyov, T. Charalambous. Embedding a Heavy-Ball type of Momentum into the Estimating Sequences. *Journal Submission*, March 2024.

Embedding a Heavy-Ball type of Momentum into the Estimating Sequences

Endrit Dosti[a,*], Sergiy A. Vorobyov[a], Themistoklis Charalambous[a,b]

[a]*School of Electrical Engineering, Aalto University, Espoo, Finland*
[b]*Department of Electrical and Computer Engineering, University of Cyprus, Nicosia, Cyprus*

Abstract

We present a new accelerated gradient-based method for solving smooth unconstrained optimization problems. The new method exploits additional information about the objective function and is built by embedding a heavy-ball type of momentum into the Fast Gradient Method (FGM). For doing so, we devise a generalization of the estimating sequences, which allows for encoding any form of information about the objective function that can aid in further accelerating the minimization process. In the black box framework, we propose a construction for the generalized estimating sequences, which is obtained by exploiting the history of the previously constructed estimating functions. Moreover, we prove that the proposed method requires at most $\sqrt{\frac{\kappa}{2}}\left(\ln\frac{1}{\epsilon}+\mathcal{O}(1)\right)$ iterations to find a point x with $f(x)-f^*\leq\epsilon$, where ϵ is the desired tolerance and κ is the condition number of the problem. Our theoretical results are further corroborated by numerical experiments on various types of optimization problems, often dealt with in various areas of the information processing sciences. Both synthetic and real-world datasets are utilized to demonstrate the efficiency of our proposed method in terms of decreasing the distance to the optimal solution, the norm of the gradient and the function value.

Keywords: Accelerated first-order methods, large-scale optimization, estimating sequence

*Corresponding author
Email addresses: endrit.dosti@aalto.fi (Endrit Dosti), sergiy.vorobyov@aalto.fi (Sergiy A. Vorobyov), themistoklis.charalambous@aalto.fi (Themistoklis Charalambous)

1. Introduction

Making the best possible inferences on large datasets, while optimizing for the computational budget is among the major goals of machine learning and other modern information processing sciences [1–3]. Among the existing techniques for large-scale data processing, first-order methods designed within the black-box framework have gained a lot of popularity as they have been shown to fulfil the computational complexity requirements, while also ensuring convergence to a neighborhood of the optimal solution [4]. These methods are comprised of recursive procedures that query a black-box oracle to obtain relevant information about the objective function [5]. The oracle being queried can have a deterministic nature (in the sense that it provides the exact value of the function of interest), or a stochastic nature.

In this work, we draw attention to the new generalized estimating sequences and convergence analysis for accelerated first-order methods in their purity. Therefore, we focus on constructing a first-order method for solving the problem of minimizing smooth and strongly convex objectives.[1] Within this class of methods, one of the most important breakthroughs is the family of Fast (or Accelerated) Gradient Methods (FGM) presented in [6] for solving problems with non-strongly convex objective functions, and in [7, Constant Step Scheme I] for solving problems with strongly convex objective functions. For minimizing smooth and strongly convex objective functions, under the assumption of known strong convexity parameter μ and Lipschitz constant L, the method requires at most $\sqrt{\kappa}\left(\ln\frac{1}{\epsilon}+\mathcal{O}(1)\right)$ iterations to find a point x with $f(x)-f^*\leq\epsilon$, where $\kappa=\frac{L}{\mu}$. In view of classic complexity theory for convex optimization [5], the method is optimal in the sense that it minimizes the number of calls of a first-order oracle required to reach a desired tolerance ϵ. We note that the bound obtained for FGM is proportional to the complexity bounds established in [5].

[1] The results obtained in this paper can be extended as well to solve composite objective problems with a non-smooth term, which is an issue that will be addressed in a later work.

In this work, we will show how to further improve the proportionality constant for FGM-type methods designed within the estimating sequences framework.

Interest in FGM surged with the paper on smoothing techniques [8]. Therein, a smooth approximation of a non-smooth objective function is constructed, and then the variant of FGM designed for solving non-strongly convex problems is used to efficiently find the optimal solution. Following up on the work, several extensions were proposed. More recently, motivated by the need to construct even faster algorithms to solve large-scale problems with smooth objective functions, new perspectives on FGM and different reasons behind acceleration have been discussed, leading to new algorithms that achieve the optimal rate [9–12]. In [9], the continuous-time limits of FGM are modeled as a second-order ordinary differential equations. Another perspective of the FGM appears in [10], where it is shown that both the strongly and non-strongly convex variants of FGM can be obtained by exploiting the linear coupling between gradient and mirror descent. In [11], the authors have developed an alternative accelerated gradient method, which is inspired by the ellipsoid method. In [12] the authors have introduced the Triple Momentum Method (TMM). The method is defined only for $\mu > 0$, and for smooth and strongly convex functions it enjoys a faster convergence rate than FGM. However, as demonstrated in [12, Table 2], the constant terms present in the bound on the number of iterations needed until convergence for TMM depend on the condition number of the problem. In case of ill-conditioned problems, it exceeds the bound of FGM, thus, requiring more iterations to converge [13, Figure 1]. Another optimal first-order method for minimizing smooth and strongly convex functions has been developed in [13]. The method proposed therein achieves the bound presented in [5] with equality, however its generalizability properties to problems that are relevant in practical applications are limited. For instance, it is not clear how to extend the method and the framework proposed therein to broader setups, which also arise more often in practice, such as solving problems with constrains, composite objective structure, etc. Moreover, its robustness (and sensitivity) to the imperfect knowledge of L and μ needs to be more thoroughly evaluated. Last, note that it

is not clear how to extend the framework to design efficient stochastic methods, distributed methods or higher order methods.

On the other hand, the estimating sequences framework, and methods built using it, have been thoroughly researched for several decades. For instance, by following the steps described in [7, Chapters 2.2.4 - 2.2.5] estimating sequence methods (which also include the method that we will propose in the sequel) can be extended to solve constrained problems. Similarly, using the gradient mapping framework presented recently in [26], the proposed generalized estimating sequences framework that we will present in Section 3 can be extended to the setup of composite objectives with non-smooth terms. Furthermore, following the analysis presented in [23] it is also possible to use a backtracking line-search strategy for estimating the value of the Lipschitz constant. An efficient strategy for estimating the value of the strong convexity parameter is also presented therein. Moreover, in Section 5, we will also demonstrate the robustness of our proposed method with respect to the imperfect knowledge of the strong convexity parameter. Last, we note that the acceleration of first-order methods which is obtained by utilizing the estimating sequences framework [7], has also been extended to other optimization settings, such as stochastic optimization [15, 16], non-Euclidean optimization [17, 18], higher-order methods [19, 20] and non-convex optimization [21, 22].

An optimization method is considered optimal if it enjoys the following properties: *i*) it exhibits the accelerated convergence rate; *ii*) it reaches a complexity that is proportional to the lower complexity bounds. For the case of first-order methods, the complexity bounds have been introduced in [5]. Several frameworks for constructing such methods have already been presented in the literature [6, 8, 7]. In [7], it is argued that the key behind constructing optimal methods is the accumulation of global information of the function that is being minimized. For this purpose, the estimating sequences are introduced. They consist of the pair $\{\phi_k(x)\}_k$, $\{\lambda_k\}_k$ and allow for parsing information around carefully selected points at each iteration, while also measuring the rate of convergence of the iterates. In the case of first-order methods, this intuition is

provably correct; however, the construction of the estimating sequences is not unique, and finding a better construction, in the sense that it leads to more efficient methods in practice, is an open question. A simple, self-contained and unified framework for the study of methods devised within the estimating sequence framework has been introduced in [19]. Therein, the author shows how several accelerated schemes can be obtained, and provide some guidelines on the design of estimating sequence methods. Evidently, picking the right functions to construct the estimating sequences, can lead to practically much faster algorithms. For example, the variant of FGM constructed in [7, Constant Step Scheme I] and its extension to convex composite objectives, i.e., the Accelerated Multistep Gradient Scheme (AMGS) [23] have been constructed using different variants of estimating sequences and are both optimal methods. The link between the two estimating sequences, as well as its implications, have been investigated in [24] and [25], and a new class of composite objective multi-step estimating sequence techniques has been designed in [26].

Despite being based on different variants of estimating sequences, both the strongly convex variant of FGM[2] and AMGS share the fact that the update of iterates at step $k+1$ is done by utilizing the information available at step k. From the theory of the heavy-ball method [27], it is known that parsing information from iterates at step $k-1$ can accelerate the minimization process. Naturally, the following question arises: "Is it possible to explicitly embed information from earlier iterates into the family of FGM?". We answer this question affirmatively, and propose a way to generalize the design of estimating sequences by including a newly introduced heavy-ball type of momentum term in them.[3] We show that embedding our proposed type of heavy-ball momentum term into Nesterov's acceleration framework leads to a more powerful class of algorithms. Our main

[2]For brevity, from this point onwards, by FGM we will refer to [7, Constant Step Scheme I].

[3]Our framework, however, can be thought as a general way of encoding any form of information about the objective function that can aid in further accelerating the minimization process.

contributions here are the following.

- We show that the original construction of the estimating functions can be generalized by incorporating extra terms that depend on previous iterates.
- To establish the properties of our newly introduced generalized estimating sequences, we revise the key lemmas and results for the classical estimating sequences. Moreover, we utilize novel tools to introduce new results, as well as a more thorough analysis of estimating sequence methods.
- We present a new type of heavy-ball momentum, which is captured by the newly introduced sequence of quadratic functions. Unlike the classical method introduced in [27], wherein the heavy-ball momentum is utilized to stabilize the oscillations of the iterates, *our proposed type of heavy-ball momentum is utilized for stabilizing the estimating sequences.*
- We develop a new method and show that (in the black-box framework) it allows for embedding a heavy-ball type of momentum into FGM. Moreover, we show that FGM can be obtained as a special case when the heavy-ball type of momentum is not considered.
- We show that the original convergence results obtained for FGM can be improved. We prove that our proposed method is also an optimal method, and show analytically that its lower bound on the number of iterations becomes $\sqrt{\frac{L}{2\mu}}\left(\ln\left(\frac{\mu R_0^2}{2\epsilon}\right) +\ln(5)\right)$, where $R_0 = ||x_0 - x^*||$, $||\cdot||$ denotes the l_2 norm of a vector, and the tolerance $\epsilon \leq \frac{\mu}{2}R_0^2$. In other words, our proposed method outperforms FGM by at least a factor of $\frac{1}{\sqrt{2}}$.
- Our proposed convergence analysis allows for initializing the parameter $\gamma_0 \in [0, \mu[\cup[2\mu, 3L+\mu]$. As shown in Section 5, this yields an improvement over FGM. Moreover, note that this result is an extension of the existing analysis for FGM, the convergence of which was proved only when $\gamma_0 \in [\mu, 3L+\mu]$. At the same time, such an extension of the analysis allows for initializing $\gamma_0 = 0$, which makes the initialization of the proposed method more robust to the imperfect knowledge of μ.
- We show through extensive simulations the efficiency of our method in solving problems using both synthetic and real-world datasets.

2. Preliminaries

Some of the most popular methods used for solving convex problems are relaxation methods [7], wherein the Gradient Method (GM) is the most widely used algorithm. GM produces a sequence of points $\{x_k\}_k,\ k = 1, 2, \ldots$ that converges to x^* at a linear rate [28]. As discussed in [7], the greedy approach of solving a convex optimization problem is not optimal. Relaxation itself is too microscopic to enable optimal convergence. Instead, it is suggested that optimal methods must make use of global topological properties of the objective function. This intuition is also confirmed by the performance of second-order methods.

As can be seen from [28, Fig. 9.19], Newton's method is constructing ellipsoids around each iterate, which aid in correcting the search direction. Therein, the ellipsoids are obtained by exploiting the information contained in the Hessian of the objective function. In the case of first-order methods, such information about the Hessian is not available. Therefore, instead of constructing ellipsoids, one can consider constructing balls in the locality of the iterate, which allow for accounting for any feasible direction. This suggests utilizing an isotropic scanning function, which at step $k = 0$ would be: $\Phi_0 : \mathcal{R}^n \to \mathcal{R}$. All that is known about this function is that

$$\nabla^2 \Phi_0(x) = \gamma_0 I, \tag{1}$$

where γ_0 is the radius of the ball and I is the identity matrix of size $n \times n$. Then, integrating (1) twice over x, the following construction is obtained:

$$\Phi_0(x) = \Phi_0^* + \frac{\gamma_0}{2}||x - x_0||^2, \tag{2}$$

where Φ_0^* is the integration constant that characterizes the value of the function $\Phi_0(x)$ when $x = x_0$. As we will see in the sequel, recursively constructing such simple scanning functions as (2), for which we coin the term *scanning functions*, is an integral component in the construction of the estimating sequences. Accordingly, the radius γ_0 is referred to as the *scanning radius*.

Next, we can exploit the information coming from the fact that the objective function is L-smooth and μ-strongly convex. Let $\mathcal{I} \subset \mathcal{R}^n$ and $x, y \in \mathcal{I}$. Then, from [7, Theorem 2.1.5] we have

$$0 \leq f(x) - f(y) - \nabla f(y)^T(x-y) \leq \frac{L}{2}||y-x||^2. \tag{3}$$

Considering the definition of strongly convex function [7, Definition 2.1.3] yields

$$f(x) \geq f(y) + \nabla f(y)^T(x-y) + \frac{\mu}{2}||y-x||^2. \tag{4}$$

The above bounds suggest the need of utilizing gradient and function evaluation oracles. In the sequel, we assume that the computational cost of computing the gradient is comparable to the cost of computing the function values.

3. The Proposed Method

We focus on finding the optimal solution for problems of the form

$$\underset{x\in\mathcal{R}^n}{\text{minimize}}\ f(x), \tag{5}$$

where $f : \mathcal{R}^n \to \mathcal{R}$ is a μ-strongly convex function with L-Lipschitz continuous gradient defined by a black-box oracle.[4]

Let us begin by defining the generalized estimating sequences as follows.

Definition 1. *The sequences $\{\Phi_k(x)\}_k$ and $\{\lambda_k\}_k$, $\lambda_k \geq 0$, are called generalized estimating sequences of the function $f(\cdot)$, if there exists a sequence of bounded functions $\{\psi_k\}_k : \mathcal{I} \subset \mathcal{R}^n \to \mathcal{Q} \subset \mathcal{R}^+$, $\lambda_k \to 0$ as $k \to \infty$, and $\forall x \in \mathcal{I}$, $\forall k = 0, 1, \ldots$ we have*

$$\Phi_k(x) \leq \lambda_k \Phi_0(x) + (1-\lambda_k)\left(f(x) - \psi_k(x)\right). \tag{6}$$

Unlike the classical definition of the estimating sequences utilized for constructing FGM [7, Definition 2.2.1], the introduction of $\psi_k(x)$ allows for encoding any form of information about the objective function that will be useful in improving the speed at which $x_k \to x^*$. One can also think of it as a *control sequence* that, at each iteration, modifies the function that is to be optimized. This modification can be done in several ways, e.g., in white-box implementations $\psi_k(x)$ can be some prior information about the structure of $f(x)$, that would make the resulting function $f(x) - \psi_k(x)$ easier to optimize. In the black-box framework, which is central to our paper, such prior information is not available. Nevertheless, as we will show later, other choices are also possible. For

[4] However, the principles developed in this work are generalizable to other frameworks, while here we aim to present these principles in their purity.

now, we note that by setting $\psi_k(x) = 0, \forall k$, we recover the estimating sequence structure used for FGM. In this sense, Definition 1 is a generalization of the classical estimating sequences.

Now, we show that the generalized estimating sequences also allow for measuring the convergence rate to optimality.

Lemma 1. *If for some sequence of points* $\{x_k\}_k$ *we have* $f(x_k) \leq \Phi_k^* \triangleq \min_{x \in \mathcal{I}} \Phi_k(x)$*, then* $f(x_k) - f(x^*) \leq \lambda_k \left[\Phi_0(x^*) - f(x^*)\right] - (1 - \lambda_k)\, \psi_k(x^*)$.

Proof. Please see Appendix A in the Supplementary Material. □

We can now show how to construct the generalized estimating sequences.

Lemma 2. *Assume that there exist sequences* $\{\alpha_k\}_k$*, where* $\alpha_k \in (0, 1)$*,* $\forall k$*,* $\sum_{k=0}^{\infty} \alpha_k = \infty$*,* $\{y_k\}_k$ *and* $\{\psi_k(x)\}_k$ *such that* $\psi_k(x) \geq 0$*,* $\forall k$*. Let* Ψ_k *be an upper bound of* $\{\psi_k(x)\}_k$*. Moreover, let* $\psi_0(x) = 0$ *and* $\lambda_0 = 1$*. Then, the sequences* $\{\Phi_k(x)\}_k$ *and* $\{\lambda_k\}_k$*, which are defined recursively as*

$$\lambda_{k+1} = (1 - \alpha_k)\lambda_k, \tag{7}$$

$$\begin{aligned} \Phi_{k+1}(x) = (1 - \alpha_k)\left(\Phi_k(x) + \psi_k(x)\right) - \psi_{k+1}(x) - \Psi_k + \alpha_k \psi_k(x) \\ + \alpha_k \left(f(y_k) + \nabla f(y_k)^T (x - y_k) + \frac{\mu}{2} ||x - y_k||^2 \right), \end{aligned} \tag{8}$$

are generalized estimating sequences.

Proof. Please see Appendix B in the Supplementary Material. □

Different from the earlier results summarized in [7], Lemma 1 has the following benefits. First, since $\lambda_k \geq 0,\ \forall k$, it clarifies why the construction of the regularizing term should be such that $\psi_k(x) \geq 0,\ \forall k$. Second, it shows that the convergence rate to optimality now depends on both the sequence $\{\lambda_k\}_k$ and the sequence $\{\psi_k(x)\}_k$. Furthermore, the result of Lemma 2 suggests the sufficient rules for updating the generalized estimating sequences.

Note that the canonical structures for the terms in the sequences $\{\Phi_k(x)\}_k$ and $\{\psi_k(x)\}_k$ have not been introduced yet, and Lemmas 1 and 2 hold for any construction of the generalized estimating sequences. Such results stress on the generality of our newly proposed estimating sequences. Let us present the constructions that will be used throughout the paper.

First, we define $\Phi_k(x) \triangleq \phi_k(x) - \psi_k(x)$, where $\phi_k(x) \triangleq \phi_k^* + \frac{\gamma_k}{2}||x - v_k||^2$ with $v_k \in \mathcal{I}$. Such structure for $\phi_k(x)$ is utilized as a general construction for a quadratic function. We have already discussed that the function $\psi_k(x)$ can be selected in many ways. Since our goal here is to construct a generalized version of FGM which operates in a black-box setup, the simplest and quite generic approach to designing $\psi_k(x)$ is to let the terms in the sequence $\{\Phi_k(x)\}_k$ "self-regulate". Indeed, as the algorithm iterates towards optimality, several scanning functions are constructed. This also allows for defining $\psi_k(x)$ as a momentum term (or a "heavy ball") that is not directly applied to the iterates, but to the scanning function whenever its value at iteration $k-1$ does not exceed a finite (but allowed to be very large) threshold value. As we will see later, this enables a better control of the parameters of $\Phi_k(x)$. Thus, let us first define $\psi_k(x)$ as

$$\psi_k(x) \triangleq \sum_{i=1}^{k-1} \beta_{i,k} \frac{\gamma_i}{2} ||x - v_i||^2, \quad \forall k, \tag{9}$$

where $\beta_{i,k} \in [0,1], \forall i = 1, \ldots, k-1$ are weights assigned to each of the previously constructed scanning functions. Note that we allow the coefficients $\beta_{i,k}$ to change dynamically across the iterations. It is worth stressing that when designing numerical methods for solving (5), the goal is to produce a sequence of points which converges to a neighborhood of the optimal solution x^*. In practice, this is achieved by running the numerical procedure for a sufficiently large, but finite number of iterations $k = 1, 2, \ldots k_{\max}$, wherein $x_{k_{\max}}$ is a solution of our problem of interest with the required accuracy.[5] Thus, we construct a convergent method to operate in some bounded convex set $\mathcal{I} \subset \mathcal{R}^n$. For example, $\mathcal{I}$ can be $\mathcal{I} = \text{conv}(x_0, x_1, x_2, \ldots, x_{k_{\max}}, x^*)$, i.e., the convex hull of all the iterates that are formed during the minimization process. Thus, $\psi_k(x)$ does not take infinite value in any possible $\mathcal{I}$ as given above. Indeed, $\psi_k(x)$ is finite since $x \in I \subset R^n$, which is always the case in practice.

We note that the model defined in (9) is similar to the heavy-ball momentum in the sense that it also encompasses the information contained in the history of

[5] This will be made more precise later in Theorem 2 .

the minimization process [29, Section 3.2]. Moreover, as we will see in the end of this section, wherein we will present the updates of our proposed algorithm, the additional terms that arise from introducing the sequence $\{\psi_k\}_k$, get added to the updates of the terms in the sequences $\{\gamma_k\}_k, \{\alpha_k\}_k, \{y_k\}_k, \{v_k\}_k$. Different from the momentum term in the heavy-ball method, which accounts for the history of the iterates, the term $\psi_k(x)$ accounts for the history of the scanning functions that were created during the minimization process. Thus, to highlight the fact that our method is also designed in the spirit of "accounting for the history of the process to improve convergence", we refer to it as a heavy-ball *type* of momentum. From this perspective, the canonical structure of the new scanning function for all values of k becomes

$$\Phi_k(x) = \phi_k^* + \frac{\gamma_k}{2}||x - v_k||^2 - \sum_{i=1}^{k-1} \beta_{i,k} \frac{\gamma_i}{2}||x - v_i||^2. \tag{10}$$

Note that we will rigorously establish later that the canonical structure for $\Phi_k(x)$ presented in (10) is preserved by the recursive definition introduced in (8). For now, let us observe that at iteration $k = 0$, (10) is the same as the construction used for FGM. For $k > 0$, the memory term begins to affect all the coefficients. From this perspective, a natural question to ask is: "How large can the term $\sum_{i=1}^{k-1} \beta_{i,k} \frac{\gamma_i}{2}||x - v_i||^2$ become?" To answer this question, we note that the simplest way to guarantee that the necessary condition for Lemma 1 holds, is to restrict $\Phi_k(x)$ to be convex $\forall k$. Therefore, utilizing the second order condition of convexity, we must have $\nabla^2 \Phi_k(x) \geq 0$. This implies that

$$\sum_{i=1}^{k-1} \beta_{i,k} \gamma_i \leq \gamma_k. \tag{11}$$

Furthermore, in (6), we also restrict the difference of functions $f(x) - \psi_k(x)$ to be convex $\forall k$. Since both functions are (by assumption) differentiable, from the second-order condition of convexity, it is sufficient to ensure that $\nabla^2 \left(f(x) - \psi_k(x)\right) \geq 0$. This results in

$$\sum_{i=1}^{k-1} \beta_{i,k} \gamma_i \leq \mu. \tag{12}$$

Combining (12) with (11), we reach

$$\sum_{i=1}^{k-1} \beta_{i,k}\gamma_i \leq \min\left(\gamma_k, \mu\right). \tag{13}$$

Let us now analyze the minimal values that the terms in $\{\Phi_k(x)\}_k$ can have. First, define $x^*_{\Phi_k} \triangleq \arg\min_x \Phi_k(x)$. Then, utilizing (10), we can write

$$\Phi^*_k = \min_x \Phi_k(x) = \phi^*_k + \frac{\gamma_k}{2}||x^*_{\Phi_k} - v_k||^2 - \sum_{i=1}^{k-1} \beta_{i,k}\frac{\gamma_i}{2}||x^*_{\Phi_k} - v_i||^2. \tag{14}$$

Note that the coefficients ϕ^*_k, γ_k and v_k are unknown and need to be found. Thus, the following lemma is in order.

Lemma 3. *Let the coefficients $\beta_{i,k}$ be selected in a way that* (13) *is satisfied. Let $\Phi_0(x) = \phi^*_0 + \frac{\gamma_0}{2}||x - v_0||^2$. Then, the process defined in Lemma 2 preserves the quadratic canonical structure of the scanning function introduced in* (10)*. Moreover, the sequences $\{\phi^*_k\}_k$, $\{\gamma_k\}_k$ and $\{v_k\}_k$ can be computed as given by* (15)*,* (16) *and* (17)*.*

$$\begin{aligned}
\phi^*_{k+1} =& \alpha_k f(y_k) + (1-\alpha_k)\phi^*_k + \frac{\alpha_k\gamma_k(1-\alpha_k)(\mu + \sum_{i=1}^{k-1}\beta_{i,k}\gamma_i)}{2\gamma_{k+1}}||y_k - v_k||^2 \\
&+ \frac{\alpha_k^3}{\gamma_{k+1}}\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i||v_i - y_k||\,||\nabla f(y_k)|| - \frac{\alpha_k^2||\nabla f(y_k)||^2}{2\gamma_{k+1}} \\
&+ \frac{\alpha_k(1-\alpha_k)\gamma_k}{\gamma_{k+1}}\left((v_k - y_k)^T\nabla f(y_k) + \sum_{i=1}^{k-1}\beta_{i,k}\gamma_i||y_k - v_i||||y_k - v_k||\right) \\
&+ (1-\alpha_k)\frac{\gamma_k}{2}||x^*_{\Phi_k} - v_k||^2 + \alpha_k\sum_{i=1}^{k-1}\frac{\beta_{i,k}\gamma_i}{2}||y_k - v_i||^2 \\
&+ \frac{(1-\alpha_k)\,\alpha_k^2}{\gamma_{k+1}}\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i(v_i - y_k)^T\nabla f(y_k) + \sum_{i=1}^{k}\beta_{i,k+1}\frac{\gamma_i}{2}||x^*_{\Phi_{k+1}} - v_i||^2,
\end{aligned} \tag{15}$$

$$\gamma_{k+1} = (1-\alpha_k)\gamma_k + \alpha_k\left(\mu + \sum_{i=1}^{k-1}\beta_{i,k}\gamma_i\right), \tag{16}$$

$$v_{k+1} = \frac{1}{\gamma_{k+1}}\left((1-\alpha_k)\gamma_k v_k + \mu\alpha_k\left(y_k - \frac{1}{\mu}\nabla f(y_k) + \sum_{i=1}^{k-1}\frac{\beta_{i,k}\gamma_i}{\mu}v_i\right)\right). \tag{17}$$

Proof. Please see Appendix C in the Supplementary Material. □

Next, assume that at iteration k, we have

$$\Phi^*_k \overset{(14)}{=} \phi^*_k + \frac{\gamma_k}{2}||x^*_{\Phi_k} - v_k||^2 - \sum_{i=1}^{k-1}\beta_{i,k}\frac{\gamma_i}{2}||x^*_{\Phi_k} - v_i||^2 \geq f(x_k). \tag{18}$$

Then, from Lemma 3, at iteration $k+1$ we obtain

$$\begin{aligned}\phi^*_{k+1} \geq & \alpha_k f(y_k) + (1-\alpha_k) f(x_k) + \frac{\alpha_k \gamma_k (1-\alpha_k)(\mu + \sum_{i=1}^{k-1} \beta_{i,k}\gamma_i)}{2\gamma_{k+1}} ||y_k - v_k||^2 \\ & + \alpha_k \sum_{i=1}^{k-1} \frac{\beta_{i,k}\gamma_i}{2} ||y_k - v_i||^2 - \frac{\alpha_k^2}{2\gamma_{k+1}} ||\nabla f(y_k)||^2 + \frac{(1-\alpha_k)\,\alpha_k^2}{\gamma_{k+1}} \sum_{i=1}^{k-1} \beta_{i,k}\gamma_i (v_i - y_k)^T \nabla f(y_k) \\ & + \frac{\alpha_k^3}{\gamma_{k+1}} \sum_{i=1}^{k-1} \beta_{i,k}\gamma_i ||v_i - y_k||\, ||\nabla f(y_k)|| + \sum_{i=1}^{k} \beta_{i,k+1} \frac{\gamma_i}{2} ||x^*_{\Phi_{k+1}} - v_i||^2 \quad (19) \\ & + \frac{\alpha_k(1-\alpha_k)\gamma_k}{\gamma_{k+1}} \left((v_k - y_k)^T \nabla f(y_k) + \sum_{i=1}^{k-1} \beta_{i,k}\gamma_i ||y_k - v_i||\, ||y_k - v_k|| \right).\end{aligned}$$

From (19), utilizing the lower bound (4) on $f(x_k)$ we arrive to (20) shown below.

$$\begin{aligned}\phi^*_{k+1} \geq & \alpha_k f(y_k) + (1-\alpha_k) \left(f(y_k) + \nabla f(y_k)^T (x_k - y_k) + \frac{\mu}{2} ||y_k - x_k||^2 \right) \\ & - \frac{\alpha_k^2}{2\gamma_{k+1}} ||\nabla f(y_k)||^2 + \frac{\alpha_k \gamma_k (1-\alpha_k)(\mu + \sum_{i=1}^{k-1} \beta_{i,k}\gamma_i)}{2\gamma_{k+1}} ||y_k - v_k||^2 \quad (20) \\ & + \frac{\alpha_k^3}{\gamma_{k+1}} \sum_{i=1}^{k-1} \beta_{i,k}\gamma_i ||v_i - y_k||\, ||\nabla f(y_k)|| + \sum_{i=1}^{k} \beta_{i,k+1} \frac{\gamma_i}{2} ||x^*_{\Phi_{k+1}} - v_i||^2 \\ & + \frac{(1-\alpha_k)\,\alpha_k^2}{\gamma_{k+1}} \sum_{i=1}^{k-1} \beta_{i,k}\gamma_i (v_i - y_k)^T \nabla f(y_k) + \alpha_k \sum_{i=1}^{k-1} \frac{\beta_{i,k}\gamma_i}{2} ||y_k - v_i||^2 \\ & + \frac{\alpha_k(1-\alpha_k)\gamma_k}{\gamma_{k+1}} \left((v_k - y_k)^T \nabla f(y_k) + \sum_{i=1}^{k-1} \beta_{i,k}\gamma_i ||y_k - v_i||\, ||y_k - v_k|| \right).\end{aligned}$$

From (20), we discard all positive terms, relax the lower bound and reach

$$\begin{aligned}\phi^*_{k+1} \geq & (1-\alpha_k) \nabla f(y_k)^T (x_k - y_k) - \frac{\alpha_k^2}{2\gamma_{k+1}} ||\nabla f(y_k)||^2 + \frac{\alpha_k(1-\alpha_k)\gamma_k}{\gamma_{k+1}} (v_k - y_k)^T \nabla f(y_k) \\ & + \sum_{i=1}^{k} \frac{\beta_{i,k+1}\gamma_i}{2} ||x^*_{\Phi_{k+1}} - v_i||^2 + (1-\alpha_k) \frac{\alpha_k^2}{\gamma_{k+1}} \sum_{i=1}^{k-1} \beta_{i,k}\gamma_i (v_i - y_k)^T \nabla f(y_k) + f(y_k).\end{aligned} \quad (21)$$

For Lemma 1 to be valid, we must guarantee that $\Phi^*_{k+1} \geq f(x_{k+1})$. Observe that by adding $\frac{\gamma_{k+1}}{2} ||x^*_{\Phi_{k+1}} - v_{k+1}||^2$ to the left-hand side (LHS) of (21), we have

$$\phi^*_{k+1} + \frac{\gamma_{k+1}}{2} ||x^*_{\Phi_{k+1}} - v_{k+1}||^2 - \sum_{i=1}^{k} \frac{\beta_{i,k+1}\gamma_i}{2} ||x^*_{\Phi_{k+1}} - v_i||^2 \overset{(14)}{=} \Phi^*_{k+1}.$$

This yields

$$\begin{aligned}\Phi^*_{k+1} \geq & f(y_k) + (1-\alpha_k) \nabla f(y_k)^T (x_k - y_k) + \frac{\alpha_k(1-\alpha_k)\gamma_k}{\gamma_{k+1}} (v_k - y_k)^T \nabla f(y_k) \\ & + (1-\alpha_k) \frac{\alpha_k^2}{\gamma_{k+1}} \sum_{i=1}^{k-1} \beta_{i,k}\gamma_i (v_i - y_k)^T \nabla f(y_k) - \frac{\alpha_k^2}{2\gamma_{k+1}} ||\nabla f(y_k)||^2. \quad (22)\end{aligned}$$

We remark that the term $f(x_{k+1})$ can be obtained from (22) in several ways. Here, we choose to relax the lower bound even further by using the following

$$f(y_k) - \frac{1}{2L}||\nabla f(y_k)||^2 \geq f(x_{k+1}), \tag{23}$$

which can be guaranteed by a simple gradient descent step on y_k, that is $x_{k+1} = y_k - h_k \nabla f(y_k)$. As can be seen from (3), it suffices to let $h_k = \frac{1}{L}$. Thus, we compute α_k to have $\frac{1}{2L}$ as the coefficient for $||\nabla f(y_k)||^2$ in (22). This yields

$$\alpha_k = \sqrt{\frac{\gamma_{k+1}}{L}}. \tag{24}$$

Then, utilizing the recursive relation for γ_{k+1} given in (16), its value can be computed in closed form by solving the quadratic equation as

$$\alpha_k = \frac{\left(\mu + \sum_{i=1}^{k-1} \beta_{i,k}\gamma_i - \gamma_k\right)}{2L} + \frac{\sqrt{\left(\mu + \sum_{i=1}^{k-1} \beta_{i,k}\gamma_i - \gamma_k\right)^2 + 4L\gamma_k}}{2L}. \tag{25}$$

Making the above-mentioned selection for α_k, we can now rewrite (22) as

$$\begin{aligned}\Phi^*_{k+1} \geq f(x_{k+1}) + (1-\alpha_k)\nabla f(y_k)^T\big((x_k - y_k) + \frac{\alpha_k\gamma_k}{\gamma_{k+1}}(v_k - y_k) \\ + \frac{\alpha_k^2}{\gamma_{k+1}}\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i(v_i - y_k)\big).\end{aligned} \tag{26}$$

From (26), we can observe an important result from the computational point of view. It is the fact that the sequence of points $\{y_k\}_k$ "comes for free", in the sense that every point y_k can be computed without the need to query a first-order oracle at point x_k. To obtain the update rule for the terms $\{y_k\}_k$, we equate the multiplier of the term $(1-\alpha_k)\nabla f(y_k)$ to 0 in (26), and obtain

$$y_k = \frac{\gamma_{k+1}x_k + \alpha_k\gamma_k v_k + \alpha_k^2\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i v_i}{\gamma_{k+1} + \alpha_k\gamma_k + \alpha_k^2\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i}. \tag{27}$$

The expression for the points $\{y_k\}_k$ obtained in (27) again highlights the benefits of utilizing the generalized estimating sequence construction. Notice that the result of FGM is preserved, and the other terms come up as coefficients of the term α_k^2. Setting $\beta_{i,k} = 0$, $\forall i = 1, 1, \ldots, k-1$, i.e., $\psi_k(x) = 0$, we recover FGM.

Assuming that the coefficients $\beta_{i,k}$ are selected to comply with (13), we come to Algorithm 1. Comparing our proposed method with [7, (2.2.19)], we first note that the selection of the next iterate is done in the same way in both

algorithms. The reason for this update stems from the fact that both methods use (23) to compute x_{k+1}. Moreover, a similar type of update rule is also applied for the terms α_k and γ_k. Evidently, in this case both methods reflect the different types of estimating sequences that were used in constructing them. The computation of the points $\{y_k\}_k$ shares the same structure in both algorithms. In Algorithm 1, the extra terms contributed from the generalized estimating sequence come up as coefficients of α_k^2. The extra terms also appear in the update rule for v_{k+1}. Last, we emphasize that if we set the term $\psi_k(x) = 0, \quad \forall k$, then Algorithm 1 reduces to the regular FGM. This is consistent with the fact that the estimating sequences utilized in constructing FGM are a special case of the generalized estimating sequences that we used in constructing Algorithm 1.

Algorithm 1. Proposed Method

Input: $x_0 \in \mathcal{R}^n$, $\gamma_0 \in [0, \mu[\cup[2\mu, 3L+\mu]$ and $v_0 = x_0$.

1: **while** stopping criterion is not meet **do**

2: Compute $\alpha_k \in [0,1]$ as $\alpha_k = \frac{\left(\mu+\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i-\gamma_k\right)+\sqrt{\left(\mu+\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i-\gamma_k\right)^2+4L\gamma_k}}{2L}$.

3: Set $\gamma_{k+1} = (1-\alpha_k)\gamma_k + \alpha_k\left(\mu + \sum_{i=1}^{k-1}\beta_{i,k}\gamma_i\right)$.

4: Choose $y_k = \frac{\gamma_{k+1}x_k+\alpha_k\gamma_k v_k+\alpha_k^2\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i v_i}{\gamma_{k+1}+\alpha_k\gamma_k+\alpha_k^2\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i}$.

5: Set $x_{k+1} = y_k - \frac{1}{L}\nabla f(y_k)$

6: Set $v_{k+1} = \frac{1}{\gamma_{k+1}}\Bigg((1-\alpha_k)\gamma_k v_k$

7: $\quad +\mu\alpha_k\Big(y_k - \frac{1}{\mu}\nabla f(y_k) + \sum_{i=1}^{k-1}\frac{\beta_{i,k}\gamma_i}{\mu}v_i\Big)\Bigg)$.

8: **end while**

Output: x_k

4. Convergence analysis

As can be anticipated from Lemma 1, the convergence rate of Algorithm 1 depends on $\{\lambda_k\}_k$ and $\{\psi_k(x)\}_k$. The following theorem makes this statement precise and allows us to present the convergence rate of Algorithm 1.

Theorem 1. *Let $\lambda_0 = 1$ and $\lambda_k = \prod_{i=1}^{k-1}(1-\alpha_i)$. Then, Algorithm 1 generates a sequence of points $\{x_k\}_k$ such that*

$$f(x_k) - f^* \leq \lambda_k\left[f(x_0) - f(x^*) + \frac{\gamma_0}{2}||x_0 - x^*||^2\right] - (1-\lambda_k)\psi_k(x^*). \qquad (28)$$

Proof. Please see Appendix D in the Supplementary Material. □

Comparing the result presented in Theorem 1 to [7, Theorem 2.2.1], we observe that as long as $\psi_k > 0$, we should expect Algorithm 1 to yield a faster convergence to optimality than the one exhibited by FGM. For this reason, we will refer to our proposed Algorithm 1 as SuperFGM (SFGM).

To analyse the rate of convergence, we start by computing the rate at which the sequence $\{\lambda_k\}_k$ decreases. The following lemma is in order.

Lemma 4. *Let the coefficients $\beta_{i,k}$ be selected in a way that* (13) *is satisfied and* $h(k) = \left(e^{\frac{k+1}{2}\sqrt{\frac{\mu+\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i}{L}}} - e^{-\frac{k+1}{2}\sqrt{\frac{\mu+\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i}{L}}}\right)^2$. *For all $k \geq 0$, Algorithm 1 guarantees that*

1. *If $\gamma_0 \in [0, \mu[$, then: $\lambda_k \leq \frac{2\mu}{Lh(k)}$.*
2. *If $\gamma_0 \in [2\mu, 3L+\mu]$, then: $\lambda_k \leq \frac{4\left(\mu+\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i\right)}{(\gamma_0-\mu-\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i)h(k)}$.*

Proof. Please see Appendix E in the Supplementary Material. □

Comparing the results obtained in Lemma 4 with their conterpart presented in [7, Lemma Lemma 2.2.4], we can observe that our analysis enables convergence over a wider range of the initialization of the parameter γ_0. The only region for which the initialization of the methods does not overlap is $\gamma_0 \in [\mu, 2\mu]$. The reason is because within this range of initializing the parameter γ_0, it is not possible to ensure that the term ξ_k defined in the proof of Lemma 4 is a real number, and the induction steps presented therein cannot be established. Nevertheless, the results proved for the terms λ_k hint that we should expect a faster rate of convergence for our proposed SFGM.

Theorem 2. *In Algorithm 1, let $\mu > 0$. Then, the algorithm generates a sequence of points such that*

1. *If $\gamma_0 \in [0, \mu[$, then*

$$f(x_k) - f(x^*) \leq \frac{\mu||x_0 - x^*||^2}{h(k)} - (1-\lambda_k)\psi_k(x^*). \tag{29}$$

2. *If $\gamma_0 \in [2\mu, 3L+\mu]$, then*

$$f(x_k) - f(x^*) \leq \frac{2L(\mu + \sum_{i=1}^{k-1}\beta_{i,k}\gamma_i)||x_0 - x^*||^2}{(\gamma_0 - \mu - \sum_{i=1}^{k-1}\beta_{i,k}\gamma_i)h(k)} - (1-\lambda_k)\psi_k(x^*). \tag{30}$$

Thus, the method is optimal when the tolerance ϵ is small enough

$$0 < \epsilon \leq \frac{\mu}{2} R_0^2. \tag{31}$$

The lower bound on the number of iterations is

$$k_{\text{SFGM}} \geq \sqrt{\frac{L}{\mu + \sum_{i=1}^{k-1} \beta_{i,k}\gamma_i}} \left(ln\left(\frac{\mu R_0^2}{2\epsilon}\right) + ln(5) \right) \tag{32}$$

Proof. Please see Appendix F in the Supplementary Material. □

Next, we compare our proposed SFGM to FGM [7, (2.2.17)], which requires

$$k_{\text{FGM}} \geq \sqrt{\frac{L}{\mu}} \left(\ln\left(\frac{\mu R_0^2}{2\epsilon}\right) + \ln(23/3) \right). \tag{33}$$

Comparing the bound in (33) to the bound obtained from our proposed method in (32), we can observe that the instance of SFGM obtained when initializing $\gamma_0 = 0$ always outperforms FGM despite any valid selection of the coefficients $\beta_{i,k}$. Under the selection $\beta_{i,k} = 0, \forall i = 1, \ldots, k-1$, which reduces SFGM to FGM, we observe that we still have an improvement of a constant number of iterations. This stems from the fact that our result obtained in Lemma 4 yields a tighter bound on the sequence $\{\lambda_k\}_k$. Moreover, it also supports the smallest possible starting value for initializing the sequence $\{\gamma_k\}_k$, i.e., $\gamma_0 = 0$, which is not supported by the existing analysis for FGM.

Allowing for nonzero values of $\beta_{i,k}$, a better scaling factor than for FGM is also obtained. Moreover, note that the bound obtained in ((32)) is dynamic, and if $\sum_{i=1}^{k-1} \beta_{i,k}\gamma_i \to \mu$, then we obtain the tightest provable bound on the performance of SFGM. Here, we remark that (32) is still an upper bound on the true performance of SFGM. The reason for that is that the proof of Theorem 2 does not fully account for the extra terms coming from the sequence $\{\psi_k(x)\}_k$. The rationale behind this approach stems from the difficulty of estimating the size of the terms in the sequence $\{\psi_k(x)\}_k$.

So far, no explicit construction about the coefficients $\beta_{i,k}$ has been given. Evidently, they act as weights that allow us to parse function information. From the result of Lemma 4, we observe that it is beneficial to allow the term

$\sum_{i=1}^{k-1} \beta_{i,k}\gamma_i$ to be as large as possible. The bound for this term has been obtained in (13). There are several ways to select the coefficients $\beta_{i,k}$, $\forall i, k$, and at the same time satisfy the bound. For instance, $\beta_{i,k}$ can be selected to account for certain samples of the previously constructed scanning functions, or a window of the previous scanning functions, or they can span the entire range of the scanning functions with some weight. For this paper, we pose the optimal selection of these coefficients as an open problem. Instead, considering also practical requirements, such as the minimization the number of computations performed by the method per iteration, storage of additional parameters etc., we select the simplest choice for the coefficients $\beta_{i,k}$, that enables the convergence of the resulting method with the fastest rate that we can prove for SFGM, that is,

$$\beta_{i,k} = \begin{cases} \min\left(1, \frac{\mu}{\gamma_{k-1}}\right), & \text{if } i = k-1, \\ 0, & \text{otherwise.} \end{cases} \tag{34}$$

Such choice of the coefficients $\beta_{i,k}$'s ensures that (13) is satisfied. Thus, with such $\beta_{i,k}$'s, Algorithm 1 remains within the setup of Lemma 3, and the theoretical results developed earlier apply to the produced iterates. Considering $\beta_{i,k}$'s selected according to (34), the lower bound on the number of iterations becomes

$$k_{\text{SFGM}} \geq \sqrt{\frac{L}{\mu + \min\left(\gamma_{k-1}, \mu\right)}} \left(\ln\left(\frac{\mu R_0^2}{2\epsilon}\right) + \ln(5)\right). \tag{35}$$

It can be observed from (35) that the introduction of the term $\psi_k(x)$ in the generalized estimating sequences yields more flexibility for improving the lower complexity bound of the proposed back-box SFGM. It was found in [8] that the lower complexity bound can be controlled by the norm selection in the objective function (for the case of unconstrained optimization). We also find, thus, that the lower complexity bound can be controlled by $\psi_k(x)$.[6] Considering (13), (16) and choosing a sufficiently large $k_{\max}$, we obtain

$$k_{\text{SFGM}} \geq \sqrt{\frac{L}{2\mu}} \left(\ln\left(\frac{\mu R_0^2}{2\epsilon}\right) + \ln(5)\right). \tag{36}$$

[6]Since we are free to select $\psi_k(x)$, in general, it can be choosen such that it accumulates available prior information about the data as well.

Let us now analyze the relative behavior of terms α_k and γ_k. From the update rule of the sequence $\{\alpha_k\}_k$, (24), we can observe that $\alpha_k \propto \gamma_{k+1}$. Therefore, if the value of γ_{k+1} increases, the value of α_k also increases. From the relationship for computing γ_{k+1} obtained in (16), we can see that also γ_{k+1} increases with α_k. Therefore, we can conclude that these two terms recursively increase the value of one-another. In Lemma 1, we established that $\gamma_0 = 0$. Then, from the update rule of the sequence $\{\gamma_k\}_k$, we can see that $\gamma_1 > \gamma_0$. This results in a value of $\alpha_0 > 0$, which then causes the values of the sequence $\{\gamma_k\}_k$ to increase. Therefore, as k increases, γ_{k-1} also increases, and the bound in (35) converges to (36). Moreover, the LHS in (35) converges to (36) very quickly due to the exponential growth of the terms in the sequence $\{\gamma_k\}_k$. Analytically, this can be seen by writing $\alpha_k = \sqrt{\gamma_{k+1}/L} = 1 - \lambda_{k+1}/\lambda_k$, and observing from Lemma 4 that the terms of the sequence $\{\lambda_k\}_k$ decrease exponentially. If we choose $k_{\max}$ to be sufficiently large, then the terms in the sequence $\{\lambda_k\}_k$ can become sufficiently small and close to 0, in the sense that the tolerance ϵ is achieved. Numerically, this is also shown in Subsection 5.1.

5. Numerical study

In this section, we test the efficiency of several instances of the proposed method both in terms of decreasing the distance to optimality, as well as in decreasing the norm of the gradient. Both synthetic and real data are utilized to analyze different aspects of the proposed algorithm. The synthetic data, which are randomly generated, are used to have a better insight on how the performance of the methods scales with the condition number of the problem. On the other hand, the real-world datasets are drawn from the Library for Support Vector Machines (LIBSVM) [30]. Datasets are selected according to the specific problem instances. We utilize CVX [31] to find the optimal solutions.

We benchmark against two instances of FGM Constant Step Scheme I (CSS1), specifically, we choose $\gamma_0 = L$, which we refer to as FGM CSS1, and $\gamma_0 = \mu$, which yields the best performance for FGM. The latter also corresponds to Constant Step Scheme III (CSS3) [7, Chapter 2.2]. For the proposed SFGM, we consider the simplest instances of the algorithm, respectively selecting $\beta_{0,k} = 1$

and $\beta_{i,k} = 0, \forall i = 1, \ldots, k$. This instance of the algorithm is referred to as memoryless SFGM. We note that when $\gamma_0 = 0$, this algorithm corresponds to FGM. However, the original analysis of FGM does not guarantee convergence of the method with $\gamma_0 = 0$, whereas SFGM guarantees the convergence, and achieves it in a smaller number of iterations. The other instance of the algorithm that is considered, is the one introduced in (34). This instance is referred to as SFGM with memory term γ_{k-1}. Relative to the CSS1 of FGM, this instance of SFGM requires the storage of an extra vector and scalar. Regarding the computations, it performs four more scalar additions and one more vector addition. Nevertheless, despite this slight increase in computational burden, we have already proved that SFGM with memory term γ_{k-1} is an optimal method. Lastly, the starting point x_0 is randomly selected and all algorithms are initiated in it.

5.1. Decreasing the distance to optimality

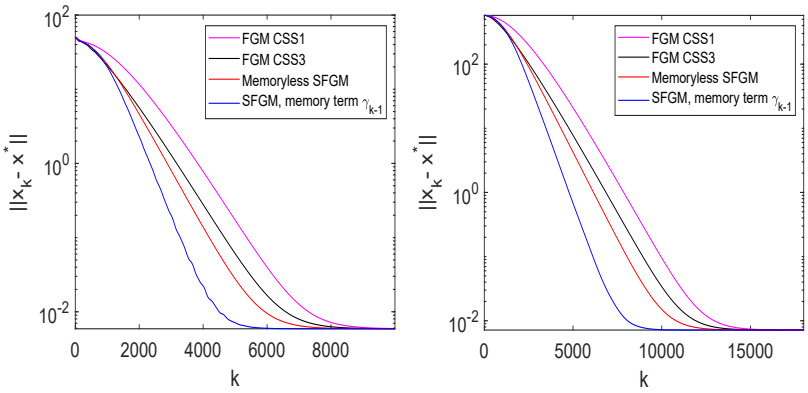
We start by solving problems of the form

$$\underset{x \in \mathcal{R}^n}{\text{minimize}} \quad \frac{1}{2m} \sum_{i=1}^{m} (a_i^T x - y_i)^2 + \frac{\tau}{2} ||x||^2, \tag{37}$$

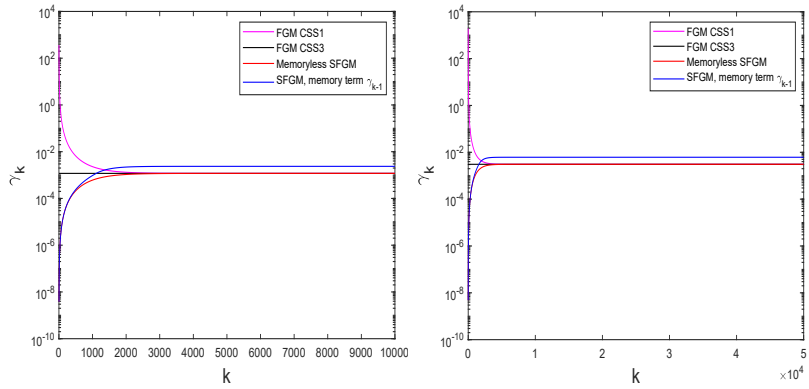
The main goal of this section is to show that the theoretical convergence guarantees obtained in Section 4 yield a realistic description of the practical performance of the methods. Moreover, we analyze how the performance of the methods scales with the condition number of the problem. We also show the fast convergence of the terms in the sequence $\{\gamma_k\}_k$.

Let us begin by considering the simplest case, $\tau = 0$. The entries of the vector a_i are sampled from a uniform distribution, whereas the values of the vector $y \in \mathcal{R}^m$ are uniformly drawn from the box $[0, 1]^n$. In our simulations, we set $m \in \{100, 1000\}$ and the resulting condition number is $\kappa \in \{3 \cdot 10^5, 9 \cdot 10^6\}$.

Our findings are reported in Fig. 1 that shows the performance gains of the proposed SFGM. The quality of the obtained solution, as measured by the distance to the optimal solution x^*, is similar to that obtained by FGM, however the number of iterations required by SFGM is smaller. In the case of the memoryless version of SFGM, we can observe that it exhibits the same behavior as FGM, however it converges faster. This is coherent with the theoretical



(a) Decreasing the distance to x^*, $\kappa = 3 \cdot 10^5$. (b) Decreasing the distance to x^*, $\kappa = 9 \cdot 10^6$.



(c) Convergence of the sequence $\{\gamma_k\}_k$, $\kappa = 3 \cdot 10^5$. (d) Convergence of the sequence $\{\gamma_k\}_k$, $\kappa = 9 \cdot 10^6$.

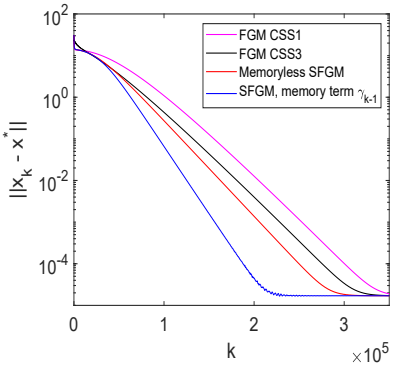
Figure 1: Comparison between various features of interest of the algorithms tested. The data is randomly generated and the goal is to minimize the quadratic loss function.

bounds established in Section 4. A similar observation can also be made for the case of SFGM with memory term γ_{k-1}. From Figs. 1(a) and 1(b), we can see that the method yields an improvement of approximately 30% over FGM CSS3. This result is also coherent with the theoretical asymptotic bound obtained in (36), that also suggests an improvement of 30% over FGM. Moreover, from Figs. 1(c) and 1(d), we can observe the exponential convergence of the term γ_{k-1} to μ. Last, as the condition number of the problem increases, all methods require a larger number of iterations to converge. For instance, from Fig. 1(a), we can see that when $\kappa = 3 \cdot 10^5$ the performance difference between the algorithms

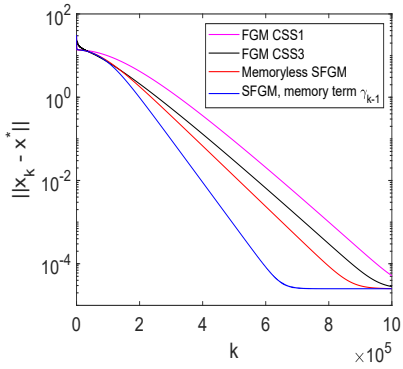
tested is of the order of few thousands of iterations. Then, when $\kappa = 9 \cdot 10^6$, from Fig. 1(b), we can see that the difference between algorithms increases. We will promptly demonstrate that for more ill-conditioned problems, the differences between the algorithms tested become even larger.

Next, we proceed by considering the more general case, $\tau \neq 0$. We let $A \in \mathcal{R}^{mxn}$ and $b \in \mathcal{R}^m$ and start with the case when $m < n$. Both synthetic and real data are utilized. To diversify the type of synthetic data used, here we do not impose any particular structure on A. We simply draw the elements for both A and b from a standard normal distribution and set $m = 800$ and $n = 1000$. Regarding real data, we utilize the "colon-cancer" dataset, for which $m = 62$ and $n = 2000$. The data that is used also dictates the values of L and μ. In practice, estimating μ is challenging and computationally expensive. For this reason, the common approach that is followed is to assume that the strong convexity parameter of the data is 0. In all the numerical experiments that will be presented in the sequel, we also follow this approach, and equate μ to the regularization parameter $\frac{\tau}{2}$. On the other hand, similar to the previous computational experiments (and to be coherent with the theoretical analysis) we estimate the Lipschitz constant directly from the data. Nevertheless, we note that several efficient backtracking strategies for estimating L already exist in the literature [23]. For the datasets that we are utilizing, the respective Lipschitz constants are $L_{\text{“random”}} = 3567.1$ and $L_{\text{“colon-cancer”}} = 1927.4$. Moreover, for both data types, we let the regularizer term $\tau \in \{10^{-5}, 10^{-6}\}$. Evidently, this selection of the regularizer term ensures that the condition number of the problems that are being solved is quite high. The numerical results are presented in Fig. 2. From this figure, we can observe that SFGM with memory term γ_{k-1} again outperforms FGM CSS3 by approximately $35\% - 40\%$.

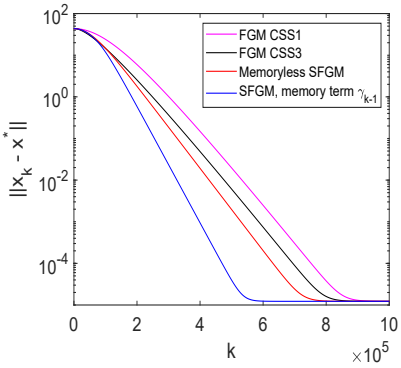
Finally, we analyze the remaining case, in which the matrix A is a tall matrix. For this, we only consider real data. The datasets that we select are "triazine" and "a1a". For the former dataset, we have $m = 186$ and $n = 60$. For the latter, we have $m = 1605$ and $n = 123$. The corresponding Lipschitz constants are $L_{\text{“triazines”}} = 632.2804$ and $L_{\text{“a1a”}} = 10061$. The regularizer term is set



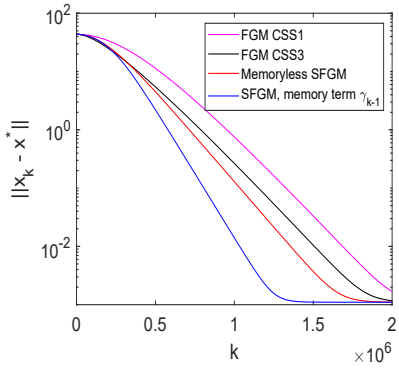
(a) Random data, $\tau = 10^{-5}$.



(b) Random data, $\tau = 10^{-6}$.



(c) The dataset is "colon-cancer", $\tau = 10^{-5}$.



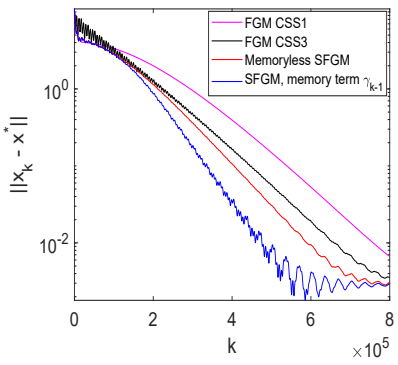
(d) The dataset is "colon-cancer", $\tau = 10^{-6}$.

Figure 2: Comparison between the efficiency of the algorithms tested in minimizing the regularized quadratic loss function in the case where $m < n$, i.e., A is a fat matrix.

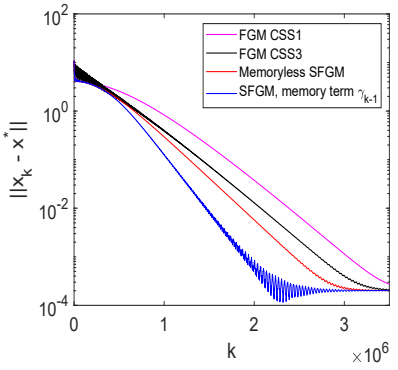
$\tau \in \{10^{-7}, 10^{-8}\}$. The results are reported in Fig. 3. Despite the fact that the problems being solved are extremely ill-conditioned, we can see that the fastest version of SFGM retains its theoretical gains of approximately $30\% - 35\%$ across all datasets, when compared to the fastest version of FGM, which is CSS3.

5.2. Decreasing the norm of the gradient

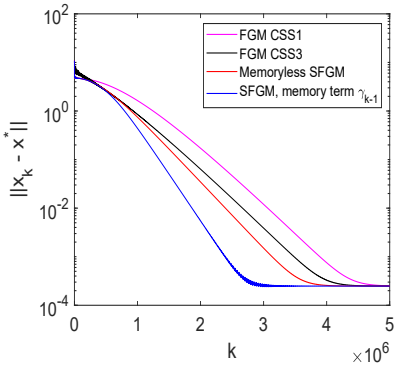
Particularly, in many practical problems, it is of high interest to find points with small norm of the gradient$||\nabla f(x)|| \leq \eta$, where η denotes the desired tol-



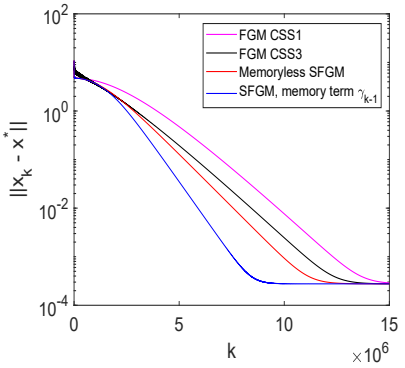
(a) The dataset is "triazines" and $\tau = 10^{-7}$.



(b) The dataset is "triazines" and $\tau = 10^{-8}$.



(c) The dataset is "a1a" and $\tau = 10^{-7}$.



(d) The dataset is "a1a" and $\tau = 10^{-8}$.

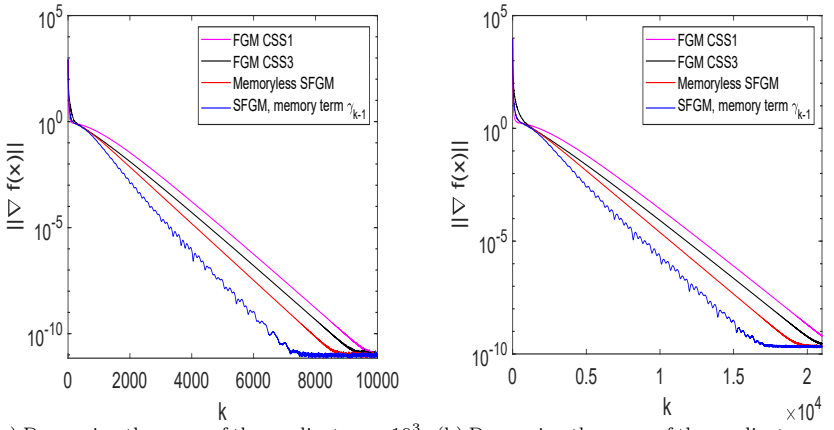
Figure 3: Comparison between the efficiency of the algorithms tested in minimizing the regularized quadratic loss function in the case where $m > n$, i.e., A is a tall matrix.

erance[7]. In [32] and [7], it is shown that FGM is not optimal in this sense. Instead, minimizing a regularized version of the objective function, which results in a reduction of the iteration complexity to $\mathcal{O}\sqrt{\frac{LR}{\epsilon}}\ln\left(\frac{LR}{\epsilon}\right)$ is suggested therein. From this perspective, utilizing the construction of $\psi_k(x)$ proposed in (9) in Definition 1, we can see that SFGM is minimizing a regularized version of the objective function. Moreover, when the generalized estimating sequences framework is used, it also provides the regularizer term, which consists of linear combinations of the previously constructed scanning functions weighted such that (13) is satisfied. In the sequel, we show that the simplest versions of SFGM

[7]The true value of the tolerance η is selected in practice depending on the application. This is different from ϵ, the true maximal value of which depends on R_0 that is typically unknown.

are more efficient than FGM in decreasing the norm of the gradient.

Prior to presenting the results for a new problem class, let us return to the setup of Fig. 1, and depict the decrease of the norm of the gradient. As can be seen from Figs. 4(a) and 4(b), the norm of the gradient of our proposed SFGM decreases much faster (approximately 30%) than that of FGM CSS3.



(a) Decreasing the norm of the gradient, $\kappa = 10^3$. (b) Decreasing the norm of the gradient, $\kappa = 10^4$.

Figure 4: Comparison between the efficiency of various algorithms in decreasing the norm of the gradient on randomly generated data.

To diversify the nature of the problems solved, in the sequel we consider the regularized logistic loss problem

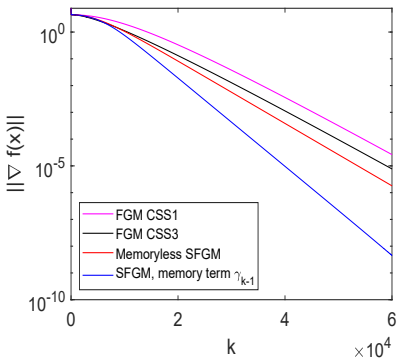
$$\underset{x\in\mathcal{R}^n}{\text{minimize}} \quad \frac{1}{m}\sum_{i=1}^{m}\log\left(1+\mathrm{e}^{-b_i a_i^T x}\right)+\frac{\tau}{2}||x||^2. \tag{38}$$

For this problem type, we reuse the datasets "colon-cancer" and "a1a", which were introduced in Subsection 5.1. We set $\tau \in \{10^{-5}, 10^{-7}\}$ for the "colon-cancer" dataset, and $\tau \in \{10^{-6}, 10^{-8}\}$ for the "a1a" dataset. The results are reported in Fig. 5. We can observe from Fig. 5 that SFGM outperforms FGM for both datasets. Specifically, SFGM with memory term γ_{k-1} is approximately $35\% - 40\%$ faster at decreasing the norm of the gradient than FGM CSS3.

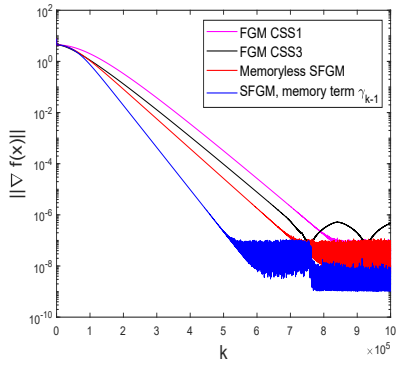
6. Conclusions and Discussion

6.1. Conclusions

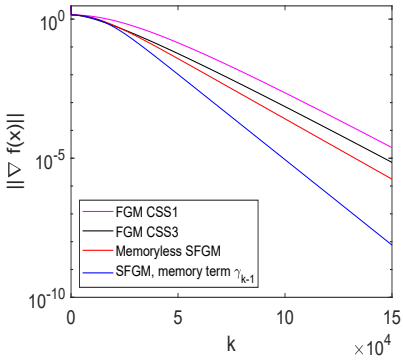
The way for embedding a new form of heavy-ball momentum into Nesterov's acceleration framework has been rigorously established, and shown to be of



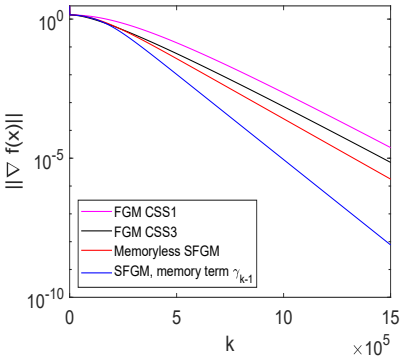
(a) The dataset is "colon-cancer" and $\tau = 10^{-5}$.



(b) The dataset is "colon-cancer" and $\tau = 10^{-7}$.



(c) The dataset is "a1a" and $\tau = 10^{-6}$.



(d) The dataset is "a1a" and $\tau = 10^{-8}$.

Figure 5: Comparison between the efficiency of the algorithms tested in minimizing the regularized logistic loss function.

practical significance. The faster convergence (than FGM) of the proposed accelerated algorithm that we name SFGM is established analytically and demonstrated through simulations and real data analysis. One more novelty important for this venue is that we provide intuition on the design of accelerated methods based on the example of the proposed SFGM, which was necessary for our objective of deriving new methods based on the embedding of different acceleration principles in one scheme.

6.2. Discussion

We conclude this work by discussing several open problems that arise from the proposed framework.

- It would be of interest to find the optimal selection of $\beta_{i,k}$. This would also produce the optimal regularizers for the objective function, which would result in faster algorithms. These optimal regularizers can further be used to enable methods that are optimal in decreasing the norm of the gradient, a topic which has gathered significant attention recently [22, 33].

- Another topic of interest is related to devising alternative candidate structures for the term $\psi_k(x)$, which can ideally encompass both black and white box information about the objective function. This is also relevant in the context of [34], as in this work we follow a similar approach in establishing (13), with the main difference being that $\psi_k(x)$ is dynamically changing over iterations. From the perspective of [34], (13) suggests that the relative strong convexity parameter between $f(x)$ and $\psi_k(x)$ is not unique. Instead, it is contained in an interval which shrinks over iterations. Thus, it is of interest to study how these frameworks can coexist.

- A strategy that is known to improve the performance of FGM is restarting [7]. In this work, we purposely avoided relying on heuristics like restarting for further improving the performance of SFGM. Nevertheless, it is of practical interest to establish restarting conditions applicable to SFGM.

- Last, it would be of interest to investigate extensions of the proposed framework to solve nonsmooth optimization problems. To solve such problems, variations of FGM already exist [23], suggesting that similar variants can be introduced in the context of our proposed SFGM.

References

[1] K. Slavakis, G. B. Giannakis and G. Mateos, "Modeling and optimization for big data analytics: (Statistical) learning tools for our era of data deluge," *IEEE Sig. Proc. Mag.*, vol. 31, no. 5, pp. 18–31, Aug. 2014.

[2] G. Lan and Y. Zhou, "Asynchronous decentralized accelerated stochastic

gradient descent," *IEEE Journal on Selected Areas in Information Theory*, vol. 2, no. 2, pp. 802–811, May 2021.

[3] E. Dosti et al., "Generalizing Nesterov's acceleration framework by embedding momentum into estimating sequences: New algorithm and bounds," *IEEE Int. Symp. on Inf. Theory*, Finland, Jun. 2022, pp. 1506–1511.

[4] A. d'Aspremont, D. Scieur, and A. Taylor, *Acceleration Methods*. Foundations and Trends in Optimization, Now Publisher, Jan. 2021.

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

[6] Y. Nesterov, "A method for solving the convex programming problem with convergence rate $\mathcal{O}(1/k^2)$," *Doklady AN USSR*, vol. 269, pp. 543–547, 1983.

[7] Y. Nesterov, *Lectures on convex optimization*. Springer, vol. 137, Dec. 2018.

[8] Y. Nesterov, "Smooth minimization of non-smooth functions," *Mathematical Programming*, vol. 103, no. 1, pp. 127–152, May. 2005.

[9] W. Su, S. Boyd and E. J. Candès, "A differential equation for modeling Nesterov's accelerated gradient method: Theory and insights," *Journal of Machine Learning Research*, vol. 17, no. 153, pp. 1–43, Jan. 2016.

[10] Z. Allen-Zhu and L. Orecchia, "Linear coupling: An ultimate unification of gradient and mirror descent," *arXiv: 1407.1537*, Nov. 2016.

[11] S. Bubeck, Y. T. Lee and M. Singh, "A geometric alternative to Nesterov's accelerated gradient descent," *arXiv: 1506.08187*, Jun. 2015.

[12] B. Van Scoy, R. A. Freeman and K. M. Lynch, "The fastest known globally convergent first-order method for minimizing strongly convex functions," *IEEE Control Systems Letters*, vol. 2, no. 1, pp. 49-54, Jan. 2018.

[13] A. Taylor and Y. Drori, "An optimal gradient method for smooth strongly convex minimization," *Mathematical Programming*, Jun. 2022.

[14] Y. Drori, "The exact information-based complexity of smooth convex minimization," *Journal of Complexity*, vol. 39, pp. 1–-16, Nov. 2016.

[15] A. Kulunchakov and J. Mairal, "Estimate Sequences for stochastic composite optimization: Variance reduction, acceleration, and robustness to noise," *Jour. of Mach. Learn. Research*, vol. 21, no. 155, pp. 1–52, Jul. 2020.

[16] G. Lan, "An optimal method for stochastic composite optimization," *Mathematical Programming*, vol. 133, no. 1, pp. 365–397, Jun. 2016.

[17] K. Ahn and S. Sra, "From Nesterov's estimate sequence to Riemannian acceleration," in *Conf. on Learn. Theory*, Austria, Jul. 2020, pp. 88–118.

[18] H. Zhang and S. Sra, "An estimate sequence for geodesically convex optimization," in *Conf. on Learn. Theory*, Sweden, Jul. 2018, pp. 1703–1723.

[19] M. Baes, "Estimate sequence methods: Extensions and approximations," *Institute for Operations Research, ETH, Zürich, Switzerland*, Aug. 2020.

[20] Y. Nesterov, "Inexact high-order proximal-point methods with auxiliary search procedure," *SIAM Journal on Optimization*, vol. 31, no. 4, pp. 2807–2828, Nov. 2021.

[21] Y. Carmon, J. C. Duchi, O. Hinder and A. Sidford, "Accelerated methods for nonconvex optimization," *SIAM Journal on Optimization*, vol. 28, no. 2, pp. 1751–1772, Jun. 2018.

[22] S. Ghadimi and G. Lan, "Accelerated gradient methods for nonconvex nonlinear and stochastic programming," *Mathematical Programming*, vol. 156, no. 1-2, pp. 59–99, Mar. 2016.

[23] Y. Nesterov, "Gradient methods for minimizing composite objective function," *Mathematical Programming*, vol. 140, no. 1, pp. 125–161, Aug. 2013.

[24] M. I. Florea and S. A. Vorobyov, "An accelerated composite gradient method for large-scale composite objective problems," *IEEE Transactions on Signal Processing*, vol. 67, no. 2 pp. 444-–459, Jan. 2019.

[25] M. I. Florea and S. A. Vorobyov, "A generalized accelerated composite gradient method: Uniting Nesterov's fast gradient method and FISTA," *IEEE Transactions on Signal Processing*, vol. 68, pp. 3033–3048, Jul. 2020.

[26] E. Dosti et al., "A new class of composite objective multi-step estimating sequence techniques," *arXiv:2111.06763*, Nov. 12, 2021.

[27] B. Polyak, "Some methods of speeding up the convergence of iteration methods," *USSR Com. Math. and Math. Phy.*, vol. 4, no. 5, pp. 1–17, 1964.

[28] S. Boyd and L. Vandenberghe, *Convex optimization*, Cambridge University Press, Mar. 2004.

[29] B. Polyak, "Introduction to optimization," *Publications Division*, 1987.

[30] C. C. Chang et al., "LIBSVM: A library for support vector machines," *ACM Trans. on Int. Syst. and Techn.*, vol. 2, no. 3, pp. 1–27, May. 2011.

[31] M. Grant, S. Boyd and Y. Ye, "CVX: Matlab software for disciplined convex programming (web page and software)," 2009.

[32] Y. Nesterov, "How to make the gradients small," *Optima*, vol. 88, pp. 10–11, 2012.

[33] Y. Carmon, J. C. Duchi, O. Hinder and A. Sidford, "Lower bounds for finding stationary points II: First-order methods," *Mathematical Programming*, vol. 185, no. 1, pp. 315–355, Jan. 2021.

[34] H. Lu, R. M. Freund and Y. Nesterov, "Relatively smooth convex optimization by first-order methods, and applications," *SIAM Journal on Optimization*, vol. 28, no. 1, pp. 333–354, Feb. 2021.

Embedding a Heavy-Ball type of Momentum into the Estimating Sequences (Supplementary Material)

Endrit Dosti[a,*], Sergiy A. Vorobyov[a], Themistoklis Charalambous[a,b]

[a] *School of Electrical Engineering, Aalto University, Espoo, Finland*
[b] *Department of Electrical and Computer Engineering, University of Cyprus, Nicosia, Cyprus*

Abstract

We present the appendixes containing the proofs for the lemmas and theorems formulated in the main paper. The document should be read in conjuction with the main paper.

Keywords: Accelerated first-order methods, large-scale optimization, estimating sequence

*Corresponding author
Email addresses: endrit.dosti@aalto.fi (Endrit Dosti), sergiy.vorobyov@aalto.fi (Sergiy A. Vorobyov), themistoklis.charalambous@aalto.fi (Themistoklis Charalambous)

Appendices

Appendix A Lemma 1 and Corresponding Proof

Lemma 1. *If for some sequence of points* $\{x_k\}_k$ *we have* $f(x_k) \leq \Phi_k^* \triangleq \min_{x\in\mathcal{I}} \Phi_k(x)$, *then*
$f(x_k) - f(x^*) \leq \lambda_k \left[\Phi_0(x^*) - f(x^*)\right] - (1-\lambda_k)\,\psi_k(x^*)$.

Proof. Based on the assumption that is made in the formulation of the lemma, we can write

$$f(x_k) \leq \Phi_k^* = \min_{x\in\mathcal{I}} \Phi_k(x) \overset{(6),\text{main paper}}{\leq} \min_{x\in\mathcal{I}} \left[\lambda_k \Phi_0(x) + (1-\lambda_k)\left(f(x) - \psi_k(x)\right)\right] \tag{38}$$

$$\leq \left[\lambda_k \Phi_0(x^*) + (1-\lambda_k)\left(f(x^*) - \psi_k(x^*)\right)\right].$$

Rearranging the terms yields the desired result. □

Appendix B Lemma 2 and Corresponding Proof

Lemma 2. *Assume that there exist sequences* $\{\alpha_k\}_k$, *where* $\alpha_k \in (0,1)$, $\forall k$, $\sum_{k=0}^{\infty} \alpha_k = \infty$, $\{y_k\}_k$ *and* $\{\psi_k(x)\}_k$ *such that* $\psi_k(x) \geq 0$, $\forall k$. *Let* Ψ_k *be an upper bound of* $\{\psi_k(x)\}_k$. *Moreover, let* $\psi_0(x) = 0$ *and* $\lambda_0 = 1$. *Then, the sequences* $\{\Phi_k(x)\}_k$ *and* $\{\lambda_k\}_k$, *which are defined recursively as*

$$\lambda_{k+1} = (1-\alpha_k)\lambda_k, \tag{7}$$

$$\Phi_{k+1}(x) = (1-\alpha_k)\left(\Phi_k(x) + \psi_k(x)\right) - \psi_{k+1}(x) - \Psi_k + \alpha_k \psi_k(x) \tag{8}$$
$$+ \alpha_k \left(f(y_k) + \nabla f(y_k)^T (x - y_k) + \frac{\mu}{2}||x - y_k||^2\right),$$

are generalized estimating sequences.

Proof. We prove the lemma by induction. At iteration $k = 0$, since $\psi_0(x) = 0$, $\Psi_0 = 0$ and $\lambda_0 = 1$, utilizing (6) in the main paper, we have $\Phi_0(x) \leq \lambda_0 \Phi_0(x) + (1-\lambda_0) f(x) \triangleq \Phi_0(x)$. Next, we assume that at some iteration k, (6) in the main paper holds true, which yields

$$\Phi_k(x) - (1-\lambda_k) f(x) \leq \lambda_k \Phi_0(x) - (1-\lambda_k)\psi_k(x). \tag{39}$$

Utilizing (4) in the main paper and (8), at iteration $k+1$ we can write

$$\Phi_{k+1}(x) \leq (1-\alpha_k)\left(\Phi_k(x)+\psi_k(x)\right) - \psi_{k+1}(x) - \Psi_k + \alpha_k\left(f(x)+\psi_k(x)\right). \tag{40}$$

From Definition 1 in the main paper, we notice that the function $\psi_k(x)$ maps to $\mathcal{Q} \subset \mathcal{R}^+$. This fact, together with the assumption that the upper bound Ψ_k cannot be infinite, are sufficient for establishing the results of the lemma. Exploiting the aforementioned observations, as well as adding and subtracting the same term to the RHS of (40), yields

$$\begin{aligned}\Phi_{k+1}(x) &\leq (1-\alpha_k)\Phi_k(x) - \psi_{k+1}(x) + \alpha_k f(x) + (1-\alpha_k)(1-\lambda_k)f(x) \\ &\quad - (1-\alpha_k)(1-\lambda_k)f(x) \\ &= (1-\alpha_k)\left[\Phi_k(x) - (1-\lambda_k)f(x)\right] - \psi_{k+1}(x) + \left(\alpha_k + (1-\lambda_k)(1-\alpha_k)\right) f(x).\end{aligned} \tag{41}$$

Utilizing (39) in (41), results in

$$\Phi_{k+1}(x) + \psi_{k+1}(x) \leq (1-\alpha_k)\left(\lambda_k\Phi_0(x) - (1-\lambda_k)\,\psi_k(x)\right) + (1-\lambda_k+\alpha_k\lambda_k)f(x). \tag{42}$$

Then, from the recursive relation (7), and also by relaxing the RHS of (42), we reach

$$\Phi_{k+1}(x) + \psi_{k+1}(x) \leq \lambda_{k+1}\Phi_0(x) + (1-\lambda_{k+1})f(x). \tag{43}$$

Finally, utilizing the fact that $\lambda_k \in [0,1]$, we obtain

$$\Phi_{k+1}(x) \leq \lambda_{k+1}\Phi_0(x) + (1-\lambda_{k+1})\left(f(x) - \psi_{k+1}(x)\right). \tag{44}$$

□

Appendix C Lemma 3 and Corresponding Proof

Lemma 3. *Let the coefficients $\beta_{i,k}$ be selected in a way that (??) is satisfied. Let $\Phi_0(x) = \phi_0^* + \frac{\gamma_0}{2}||x - v_0||^2$. Then, the process defined in Lemma 2 preserves*

the quadratic canonical structure of the scanning function introduced in (**??**). *Moreover, the sequences* $\{\phi_k^*\}_k$, $\{\gamma_k\}_k$ *and* $\{v_k\}_k$ *can be computed as given by* (15), (16) *and* (17).

$$\phi_{k+1}^* = \alpha_k f(y_k) + (1-\alpha_k)\phi_k^* + \frac{\alpha_k\gamma_k(1-\alpha_k)(\mu+\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i)}{2\gamma_{k+1}}||y_k - v_k||^2 \tag{15}$$

$$\begin{aligned}
&+\frac{\alpha_k^3}{\gamma_{k+1}}\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i||v_i-y_k||\,||\nabla f(y_k)|| - \frac{\alpha_k^2||\nabla f(y_k)||^2}{2\gamma_{k+1}}\\
&+\frac{\alpha_k(1-\alpha_k)\gamma_k}{\gamma_{k+1}}\left((v_k-y_k)^T\nabla f(y_k)+\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i||y_k-v_i||||y_k-v_k||\right)\\
&+(1-\alpha_k)\frac{\gamma_k}{2}||x_{\Phi_k}^*-v_k||^2+\alpha_k\sum_{i=1}^{k-1}\frac{\beta_{i,k}\gamma_i}{2}||y_k-v_i||^2\\
&+\frac{(1-\alpha_k)\,\alpha_k^2}{\gamma_{k+1}}\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i(v_i-y_k)^T\nabla f(y_k)+\sum_{i=1}^{k}\beta_{i,k+1}\frac{\gamma_i}{2}||x_{\Phi_{k+1}}^*-v_i||^2;
\end{aligned}$$

$$\gamma_{k+1} = (1-\alpha_k)\gamma_k + \alpha_k\left(\mu+\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i\right), \tag{16}$$

$$v_{k+1} = \frac{1}{\gamma_{k+1}}\left((1-\alpha_k)\gamma_k v_k + \mu\alpha_k\left(y_k - \frac{1}{\mu}\nabla f(y_k) + \sum_{i=1}^{k-1}\frac{\beta_{i,k}\gamma_i}{\mu}v_i\right)\right). \tag{17}$$

Proof. Let us begin by establishing that (8) preserves the quadratic structure of the terms in the sequence $\{\Phi_k\}_k$. Note that at step $k=0$, we have $\psi_0 = 0$. Therefore, $\nabla^2\Phi_0(x) = \nabla^2\phi_0(x) = \gamma_0 I$. Next, let us assume that for some step k, we have $\nabla^2\Phi_k(x) = \gamma_k - \sum_{i=1}^{k-1}\beta_{i,k}\gamma_i \overset{(13),\text{main paper}}{\geq} 0$. Then, by considering the Hessian of (8), we can write

$$\nabla^2\Phi_{k+1}(x) = (1-\alpha_k)\gamma_k I - \sum_{i=1}^{k}\beta_{i,k}\gamma_i I + \alpha_k\left(\mu+\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i\right)I. \tag{45}$$

Utilizing (16) in (45) we obtain

$$\nabla^2\Phi_{k+1}(x) = \gamma_{k+1}I - \sum_{i=1}^{k}\beta_{i,k}\gamma_i I. \tag{46}$$

Last, we note that selecting the coefficients $\beta_{i,k}$ to satisfy (13) in the main paper ensures that $\nabla^2\Phi_{k+1}(x) \geq 0$.

We proceed now to establishing the recursive relation for the terms in the sequence $\{v_k\}_k$. Let us start by substituting our proposed construction for the scanning function presented in (10) in the main paper into (8), and making the necessary manipulations to obtain

$$\phi^*_{k+1} + \frac{\gamma_{k+1}}{2}||x - v_{k+1}||^2 = (1-\alpha_k)\Big(\phi^*_k + \frac{\gamma_k}{2}||x-v_k||^2\Big) - \Psi_k \\ +\alpha_k\Big(f(y_k)+\nabla f(y_k)^T(x-y_k) + \frac{\mu}{2}||x-y_k||^2 + \psi_k(x)\Big). \tag{47}$$

First, observe that both the LHS and the RHS of (47) are convex functions in x, and minimizing them over all possible values of x yields two unconstrained optimization problems. Therefore, the solution needs to satisfy the optimality condition for unconstrained problems, which is that the gradient of the objective function with respect to the optimization parameter has to be equal to 0. Taking gradients with respect to x, yields

$$\gamma_{k+1}(x-v_{k+1}) = \gamma_k(1-\alpha_k)(x-v_k) + \alpha_k\big(\mu(x-y_k) + \nabla f(y_k) + \sum_{i=1}^{k-1}\beta_{i,k}\gamma_i(x-v_i)\big). \tag{48}$$

For now, assume that the points y_k are known and x's are unknown. By utilizing (16), we can reduce the unknown x's in (48). Then, after making some manipulations, we obtain

$$v_{k+1} = \frac{1}{\gamma_{k+1}}\big((1-\alpha_k)\gamma_k v_k + \mu\alpha_k(y_k - \frac{1}{\mu}\nabla f(y_k) + \sum_{i=1}^{k-1}\frac{\beta_{i,k}\gamma_i}{\mu}v_i)\big). \tag{49}$$

Notice that the values of the terms in the sequence $\{v_k\}_k$ depend on the corresponding terms in the sequence $\{y_k\}_k$, which are assumed to be known up to this point. We will show later how these values can be computed recursively. For now, let us focus on finding the smallest value of the scanning function ϕ^*_{k+1}. On a conceptual level, the simplest way to compute ϕ^*_{k+1} is to think that there is another scanning function $\Theta_k(y_k)$ for the sequence $\{y_k\}_k$, which has the same center and radius and as the sequence of functions $\{\Phi_k(x)\}_k$. So, for all values of the iterates k, we have

$$\Theta_k(y_k) = \theta^*_k + \frac{\gamma_k}{2}||y_k - v_k||^2 - \sum_{i=1}^{k-1}\beta_{i,k}\frac{\gamma_i}{2}||y_k - v_i||^2. \tag{50}$$

Then, utilizing (8) applied at $x = y_k$, yields

$$\Theta_{k+1}(y_k) = (1-\alpha_k)\left(\Theta_k(y_k)+\psi_k(y_k)\right) - \psi_{k+1}(y_k) - \Psi_k + \alpha_k\left(f(y_k)+\psi_k(y_k)\right). \tag{51}$$

Substituting (9) in the main paper and (50) into (51), as well as making the necessary relaxations, we obtain

$$\begin{aligned}\theta^*_{k+1} + \frac{\gamma_{k+1}}{2}||y_k - v_{k+1}||^2 \leq (1-\alpha_k)\left(\theta^*_k + \frac{\gamma_k}{2}||y_k - v_k||^2\right) \\ + \alpha_k\left(f(y_k) + \sum_{i=1}^{k-1}\frac{\beta_{i,k}\gamma_i}{2}||y_k - v_i||^2\right).\end{aligned} \tag{52}$$

From the recursive relation (49), we have

$$v_{k+1} - y_k = \frac{1}{\gamma_{k+1}}\left((1-\alpha_k)\gamma_k v_k + \mu\alpha_k\big(y_k - \frac{1}{\mu}\nabla f(y_k) + \sum_{i=1}^{k-1}\frac{\beta_{i,k}\gamma_i}{\mu}v_i\big) - \gamma_{k+1}y_k\right). \tag{53}$$

Then, substituting the recursive relation for the term γ_{k+1}, i.e., (16) into (53), yields

$$v_{k+1} - y_k = \frac{1}{\gamma_{k+1}}\big((1-\alpha_k)\gamma_k\left(v_k - y_k\right) - \alpha_k\nabla f(y_k) + \alpha_k\sum_{i=1}^{k-1}\frac{\beta_{i,k}\gamma_i}{\mu}\left(v_i - y_k\right)\big). \tag{54}$$

Taking $||\cdot||^2$ of both sides in (54), we obtain

$$||v_{k+1} - y_k||^2 = \frac{||\left(\gamma_k(1-\alpha_k)(v_k - y_k)\right) + \alpha_k\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i(v_i - y_k) - \alpha_k\nabla f(y_k)||^2}{\gamma^2_{k+1}}. \tag{55}$$

Then, multiplying both sides of (55) shown in the next page by $\frac{\gamma_{k+1}}{2}$ and ex-

panding the RHS, we reach

$$\frac{\gamma_{k+1}}{2}||v_{k+1}-y_k||^2 = \frac{(1-\alpha_k)^2\gamma_k^2}{2\gamma_{k+1}}||v_k-y_k||^2 + \frac{\alpha_k^2}{2\gamma_{k+1}}||\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i(v_i-y_k)||^2 \tag{56}$$
$$- \frac{2\alpha_k(1-\alpha_k)\gamma_k}{2\gamma_{k+1}}(v_k-y_k)^T\nabla f(y_k) + \frac{\alpha_k^2}{2\gamma_{k+1}}||\nabla f(y_k)||^2$$
$$+ \frac{(1-\alpha_k)\alpha_k\gamma_k}{\gamma_{k+1}}\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i(v_i-y_k)^T(v_k-y_k)$$
$$- \frac{\alpha_k^2}{\gamma_{k+1}}\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i(y_k-v_i)^T\nabla f(y_k).$$

Substituting (56) into (52) and doing the respective factorings, we obtain

$$\theta_{k+1}^* \leq \alpha_k f(y_k) + (1-\alpha_k)\theta_k^* + \frac{(1-\alpha_k)\gamma_k}{2}\left[\frac{\gamma_{k+1}}{\gamma_{k+1}} - \frac{(1-\alpha_k)\gamma_k}{\gamma_{k+1}}\right]||y_k-v_k||^2$$
$$+ \alpha_k\sum_{i=1}^{k-1}\frac{\beta_{i,k}\gamma_i}{2}||y_k-v_i||^2 - \frac{\alpha_k^2}{2\gamma_{k+1}}||\sum_{i=1}^{k-1}\frac{\beta_{i,k}\gamma_i}{2}(y_k-v_i)||^2$$
$$- \frac{\alpha_k^2}{2\gamma_{k+1}}||\nabla f(y_k)||^2 + \frac{\alpha_k^2}{\gamma_{k+1}}\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i(v_i-y_k)^T\nabla f(y_k)$$
$$+ \frac{\alpha_k(1-\alpha_k)\gamma_k}{\gamma_{k+1}}\left((v_k-y_k)^T\nabla f(y_k) - \sum_{i=1}^{k-1}\beta_{i,k}\gamma_i(y_k-v_i)^T(y_k-v_k)\right). \tag{57}$$

Making some further manipulations and relaxing the upper bound on θ_{k+1}^* in (57) yields

$$\theta_{k+1}^* \leq \alpha_k f(y_k) + (1-\alpha_k)\theta_k^* + \frac{\alpha_k\gamma_k(1-\alpha_k)(\mu+\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i)}{2\gamma_{k+1}}||y_k-v_k||^2$$
$$+ \alpha_k\sum_{i=1}^{k-1}\frac{\beta_{i,k}\gamma_i}{2}||y_k-v_i||^2 - \frac{\alpha_k^2}{2\gamma_{k+1}}||\nabla f(y_k)||^2$$
$$+ \frac{(1-\alpha_k)\,\alpha_k^2}{\gamma_{k+1}}\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i(v_i-y_k)^T\nabla f(y_k) + \frac{\alpha_k^3}{\gamma_{k+1}}\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i(v_i-y_k)^T\nabla f(y_k)$$
$$+ \frac{\alpha_k(1-\alpha_k)\gamma_k}{\gamma_{k+1}}\left((v_k-y_k)^T\nabla f(y_k) - \sum_{i=1}^{k-1}\beta_{i,k}\gamma_i(y_k-v_i)^T\,(y_k-v_k)\right). \tag{58}$$

Then, utilizing the Cauchy-Schwartz inequality in (58), as well as relaxing the upper bound, we obtain

$$\begin{aligned}\theta^*_{k+1} \leq & \alpha_k f(y_k) + (1-\alpha_k)\theta^*_k + \frac{\alpha_k\gamma_k(1-\alpha_k)(\mu + \sum_{i=1}^{k-1}\beta_{i,k}\gamma_i)}{2\gamma_{k+1}}||y_k - v_k||^2 \\ & + \alpha_k \sum_{i=1}^{k-1}\frac{\beta_{i,k}\gamma_i}{2}||y_k - v_i||^2 - \frac{\alpha_k^2}{2\gamma_{k+1}}||\nabla f(y_k)||^2 \\ & + \frac{(1-\alpha_k)\,\alpha_k^2}{\gamma_{k+1}}\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i(v_i - y_k)^T\nabla f(y_k) + \sum_{i=1}^{k}\beta_{i,k+1}\frac{\gamma_i}{2}||x^*_{\Phi_{k+1}} - v_i||^2 \\ & + \frac{\alpha_k^3}{\gamma_{k+1}}\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i||v_i - y_k||\;||\nabla f(y_k)|| + (1-\alpha_k)\frac{\gamma_k}{2}||x^*_{\Phi_k} - v_k||^2 \\ & + \frac{\alpha_k(1-\alpha_k)\gamma_k}{\gamma_{k+1}}\left((v_k - y_k)^T\nabla f(y_k) + \sum_{i=1}^{k-1}\beta_{i,k}\gamma_i||y_k - v_i||\;||y_k - v_k||\right).\end{aligned} \tag{59}$$

Last, since we would like the scanning function to be as close as possible to the objective function itself, we let θ_{k+1} equal to the tightest upper bound we can obtain analytically. Moreover, as discussed earlier, we let $\phi^*_k = \theta^*_k,\ \forall k$. This yields (15). □

Appendix D Theorem 1 and Corresponding Proof

Theorem 1. *Let $\lambda_0 = 1$ and $\lambda_k = \prod_{i=1}^{k-1}(1-\alpha_i)$. Then, Algorithm **??** generates a sequence of points $\{x_k\}_k$ such that*

$$f(x_k) - f^* \leq \lambda_k\left[f(x_0) - f(x^*) + \frac{\gamma_0}{2}||x_0 - x^*||^2\right] - (1-\lambda_k)\psi_k(x^*). \tag{28}$$

Proof. Let $\phi^*_0 = f(x_0)$. Then, by construction of the scanning function at $k = 0$, we have $f(x_0) \leq \Phi_0(x) = f(x_0) + \frac{\gamma_0}{2}||x - x_0||^2$. Moreover, we recall that the update rules of SFGM are devised to maintain the relation $f(x_k) \leq \Phi^*_k$. This is sufficient for the results proved in Lemma 1 to be applied. □

Appendix E Lemma 4 and Corresponding Proof

Lemma 4. *Let the coefficients $\beta_{i,k}$ be selected in a way that (13) in the main paper is satisfied and* $h(k) = \left(e^{\frac{k+1}{2}\sqrt{\frac{\mu+\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i}{L}}} - e^{-\frac{k+1}{2}\sqrt{\frac{\mu+\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i}{L}}}\right)^2$. *For all $k \geq 0$, Algorithm 1 in the main paper guarantees that*

1. *If $\gamma_0 \in [0, \mu[$, then: $\lambda_k \leq \frac{2\mu}{Lh(k)}$.*
2. *If $\gamma_0 \in [2\mu, 3L+\mu]$, then: $\lambda_k \leq \frac{4\left(\mu+\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i\right)}{(\gamma_0-\mu-\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i)h(k)}$.*

Proof. Recall that in Algorithm 1 in the main paper we initialize $\gamma_0 \in [0, \mu[\cup[2\mu, 3L+\mu]$. From (16), we can write

$$\begin{aligned}\gamma_{k+1} - \left(\mu + \sum_{i=1}^{k-1}\beta_{i,k}\gamma_i\right) &= (1-\alpha_k)\gamma_k + \alpha_k\left(\mu + \sum_{i=1}^{k-1}\beta_{i,k}\gamma_i\right) - \left(\mu + \sum_{i=1}^{k-1}\beta_{i,k}\gamma_i\right) \\ &= (1-\alpha_k)\lambda_0\left[\gamma_k - \left(\mu + \sum_{i=1}^{k-1}\beta_{i,k}\gamma_i\right)\right]. \end{aligned} \tag{60}$$

Then, utilizing the recursivity of (16) in (60), we obtain

$$\gamma_{k+1} - \left(\mu + \sum_{i=1}^{k-1}\beta_{i,k}\gamma_i\right) = \lambda_{k+1}\left[\gamma_0 - \left(\mu + \sum_{i=1}^{k-1}\beta_{i,k}\gamma_i\right)\right]. \tag{61}$$

Recalling that $\lambda_{k+1} = (1-\alpha_k)\lambda_k$ and considering (24) in the main paper, we have

$$\begin{aligned}\alpha_k = 1 - \frac{\lambda_{k+1}}{\lambda_k} = \sqrt{\frac{\gamma_{k+1}}{L}} &= \sqrt{\frac{\mu+\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i}{L} + \frac{\gamma_{k+1} - \left(\mu+\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i\right)}{L}} \\ &\overset{(61)}{=} \sqrt{\frac{\mu+\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i}{L} + \lambda_{k+1}\frac{\gamma_0 - \left(\mu+\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i\right)}{L}}.\end{aligned}$$

Moreover,

$$\begin{aligned}\frac{\lambda_k - \lambda_{k+1}}{\lambda_k} &= \sqrt{\lambda_{k+1}}\sqrt{\frac{\mu + \sum_{i=1}^{k-1}\beta_{i,k}\gamma_i}{\lambda_{k+1}L} + \frac{\gamma_0 - \left(\mu + \sum_{i=1}^{k-1}\beta_{i,k}\gamma_i\right)}{L}}, \\ \frac{\lambda_k - \lambda_{k+1}}{\lambda_k\lambda_{k+1}} &= \frac{1}{\sqrt{\lambda_{k+1}}}\sqrt{\frac{\mu + \sum_{i=1}^{k-1}\beta_{i,k}\gamma_i}{\lambda_{k+1}L} + \frac{\gamma_0 - \left(\mu + \sum_{i=1}^{k-1}\beta_{i,k}\gamma_i\right)}{L}}. \end{aligned} \tag{62}$$

Then, by writing the LHS of (62) as $\frac{\lambda_k-\lambda_{k+1}}{\lambda_k\lambda_{k+1}} = \frac{1}{\lambda_{k+1}} - \frac{1}{\lambda_k}$, and utilizing a difference of squares argument, we obtain

$$\left(\frac{1}{\sqrt{\lambda_{k+1}}}-\frac{1}{\sqrt{\lambda_k}}\right)\left(\frac{1}{\sqrt{\lambda_{k+1}}}+\frac{1}{\sqrt{\lambda_k}}\right)=\frac{1}{\sqrt{\lambda_{k+1}}} \sqrt{\frac{\mu+\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i}{\lambda_{k+1}L}+\frac{\gamma_0-\left(\mu+\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i\right)}{L}}. \tag{63}$$

In (63), we can lower bound the LHS by replacing $\frac{1}{\sqrt{\lambda_k}}$ with the larger number $\frac{1}{\sqrt{\lambda_{k+1}}}$. This results in

$$\frac{2}{\sqrt{\lambda_{k+1}}}\left(\frac{1}{\sqrt{\lambda_{k+1}}}-\frac{1}{\sqrt{\lambda_k}}\right)\geq\frac{1}{\sqrt{\lambda_{k+1}}}\sqrt{\frac{\mu+\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i}{\lambda_{k+1}L}+\frac{\gamma_0-\left(\mu+\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i\right)}{L}}. \tag{64}$$

To establish convergence for the entire range of values for the term γ_0, let us consider separately the regions $\mathcal{R}_1 = [0, \mu[$ and $\mathcal{R}_2 = [2\mu, 3L + \mu]$. Let us now begin by considering the region $\gamma_0 \in \mathcal{R}_1$. First, we can rewrite (64) as

$$\frac{2}{\sqrt{\lambda_{k+1}}}-\frac{2}{\sqrt{\lambda_k}}\geq\sqrt{\frac{\left(\mu+\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i\right)-\gamma_0}{L}}\sqrt{\frac{L\left(\mu+\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i\right)}{L\lambda_{k+1}\left(\mu+\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i-\gamma_0\right)}-1}. \tag{65}$$

Then, we define the following[1].

$$\xi_k \triangleq \sqrt{\frac{L}{\left[\left(\mu+\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i\right)-\gamma_0\right]\lambda_k}}. \tag{66}$$

Multiplying both sides of (65) by $\sqrt{\frac{L}{\mu+\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i-\gamma_0}}$ and utilizing the newly introduced (66), yields

$$\xi_{k+1}-\xi_k\geq\frac{1}{2}\sqrt{\frac{\left(\mu+\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i\right)\xi_{k+1}^2}{L}-1}. \tag{67}$$

[1]We note that restricting $\gamma_0 \in \mathcal{R}_1$ ensures that there is no division by 0 in the denominator of (66)

At this point, we make use of induction to prove the following bound on ξ_k

$$\xi_k \geq \frac{\sqrt{2}}{4\delta}\sqrt{\frac{L}{\mu}}\left[e^{(k+1)\delta} - e^{-(k+1)\delta}\right], \tag{68}$$

where $\delta \triangleq \frac{1}{2}\sqrt{\frac{\mu+\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i}{L}}$. At step $k=0$ we have

$$\xi_0 \overset{(66)}{=} \sqrt{\frac{L}{(\mu+\gamma_{-1}-\gamma_0)\lambda_0}} = \sqrt{\frac{L}{\mu-\gamma_0}} \geq \frac{1}{2}\sqrt{\frac{L}{\mu}}\left[e^{\frac{\sqrt{2}}{2}} - e^{-\frac{\sqrt{2}}{2}}\right] \geq \frac{\sqrt{2}}{4\delta}\sqrt{\frac{L}{\mu}}\left[e^{\delta} - e^{-\delta}\right], \tag{69}$$

where the second equality is obtained from the assumptions made in Lemma 2, i.e., $\lambda_0 = 1$ and $\gamma_k = 0$, $\forall k < 0$. From (1) in the main paper, we must have $\gamma_0 \geq 0$. Recalling that $\gamma_0 \in \mathcal{R}_1$ in (69) and multiplying it with a number that is smaller than 1, yields the first inequality. The last inequality in (69) follows because the RHS increases with δ, which by construction is $\delta < \frac{\sqrt{2}}{2}$.

Next, we assume that (68) holds at iteration k and prove the same result for iteration $k+1$ via contradiction. Letting $\omega(t) = \frac{1}{4\delta}\sqrt{\frac{L}{\mu}}\left[e^{(t+1)\delta} - e^{-(t+1)\delta}\right]$, which is a convex function [1, Lemma 2.2.4], we have

$$\omega(t) \leq \xi_k \overset{(67)}{\leq} \xi_{k+1} - \frac{1}{2}\sqrt{\frac{\left(\mu+\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i\right)\xi_{k+1}^2}{L} - 1}. \tag{70}$$

Now, suppose $\xi_{k+1} < \omega(t+1)$. Substituting it into (70), yields

$$\omega(t) \overset{(70)}{<} \omega(t+1) - \frac{1}{2}\sqrt{\frac{\left(\mu+\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i\right)\xi_{k+1}^2}{L} - 1}. \tag{71}$$

Then, applying (68) and the definition of δ, yields

$$\begin{aligned}
\omega(t) &\leq \omega(t+1) - \frac{1}{2}\sqrt{4\delta^2\left[\frac{\sqrt{2}}{4\delta}\sqrt{\frac{L}{\mu}}\left(e^{(t+2)\delta} - e^{-(t+2)\delta}\right)\right]^2 - 1} \\
&= \omega(t+1) - \frac{2}{4}\sqrt{\frac{L}{\mu}}\left[e^{(t+2)\delta} + e^{-(t+2)\delta}\right] \\
&= \omega(t+1) + \omega(t+1)'\left(t - (t+1)\right) \leq \omega(t),
\end{aligned}$$

where the last inequality follows from the supporting hyperplane theorem of convex functions. Evidently, this leads to a contradiction with our earlier assumption, which implies that $\xi_{k+1} < \omega(k+1), \forall k$. Therefore, (68) must hold true.

From (66) we can write

$$\lambda_k = \frac{L}{\left(\mu + \sum_{i=1}^{k-1} \beta_{i,k}\gamma_i - \gamma_0\right)\xi_k^2}. \tag{72}$$

Then, recalling that $\gamma_0 \in \mathcal{R}_1$ and utilizing (68), we obtain

$$\frac{L}{\left(\mu + \sum_{i=1}^{k-1} \beta_{i,k}\gamma_i - \gamma_0\right)\xi_k^2} \leq \frac{\mu(4\delta)^2}{2\left(\mu + \sum_{i=1}^{k-1} \beta_{i,k}\gamma_i\right)\left[e^{(k+1)\delta} - e^{-(k+1)\delta}\right]^2}, \tag{73}$$

Last, applying the definition of δ in (73), we obtain the bound presented in the first point of the lemma.

Let us next consider the region $\gamma_0 \in \mathcal{R}_2$[2]. First, we can rewrite (64) as

$$\frac{2}{\sqrt{\lambda_{k+1}}} - \frac{2}{\sqrt{\lambda_k}} \geq \sqrt{\frac{\gamma_0 - \mu - \sum_{i=1}^{k-1} \beta_{i,k}\gamma_i}{L}} \sqrt{\frac{L\left(\mu + \sum_{i=1}^{k-1} \beta_{i,k}\gamma_i\right)}{L\lambda_{k+1}\left(\gamma_0 - \mu - \sum_{i=1}^{k-1} \beta_{i,k}\gamma_i\right)} + 1}. \tag{74}$$

Then, we define the following[3]

$$\xi_k \triangleq \sqrt{\frac{L}{\left[\left(\gamma_0 - \mu - \sum_{i=1}^{k-1} \beta_{i,k}\gamma_i\right)\right]\lambda_k}}. \tag{75}$$

Multiplying both sides of (74) by $\sqrt{\frac{L}{\gamma_0 - \mu - \sum_{i=1}^{k-1} \beta_{i,k}\gamma_i}}$ and utilizing the newly introduced (75), yields

$$\xi_{k+1} - \xi_k \geq \frac{1}{2}\sqrt{\frac{\left(\mu + \sum_{i=1}^{k-1} \beta_{i,k}\gamma_i\right)\xi_{k+1}^2}{L} + 1}. \tag{76}$$

[2]We note that this part follows mutatis mutandis the analysis given in [1, Lemma 2.2.4] and is presented here for completeness.

[3]We again note that restricting $\gamma_0 \in \mathcal{R}_2$ ensures that there is no division by 0 in the denominator of (75).

At this point, we make use of induction to prove the following bound on ξ_k

$$\xi_k \geq \frac{1}{4\delta}\left[e^{(k+1)\delta} - e^{-(k+1)\delta}\right], \tag{77}$$

where $\delta \triangleq \frac{1}{2}\sqrt{\frac{\mu+\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i}{L}}$. At step $k=0$ we have

$$\xi_0 \overset{(75)}{=} \sqrt{\frac{L}{(\gamma_0-\mu-\gamma_{-1})\lambda_0}} = \sqrt{\frac{L}{\gamma_0-\mu}} \geq \sqrt{\frac{1}{3}} \geq \frac{1}{2}\left[e^{\frac{1}{2}} - e^{-\frac{1}{2}}\right] \geq \frac{1}{4\delta}\left[e^{\delta} - e^{-\delta}\right], \tag{78}$$

where the second equality is obtained from the assumptions made in Lemma 2, i.e., $\lambda_0 = 1$ and $\gamma_k = 0, \forall k < 0$. From (1) in the main paper, we must have $\gamma_0 \geq 0$. Recalling that $\gamma_0 \in \mathcal{R}_2$ in (78) yields the first inequality. The last inequality in (78) follows because the RHS increases with δ, which by construction is $\delta < \frac{1}{2}$.

Next, we assume that (77) holds at iteration k and prove the same result for iteration $k+1$ via contradiction. Letting $\omega(t) = \frac{1}{4\delta}\left[e^{(t+1)\delta} - e^{-(t+1)\delta}\right]$, which is a convex function [1, Lemma 2.2.4], we have

$$\omega(t) \leq \xi_k \overset{(76)}{\leq} \xi_{k+1} - \frac{1}{2}\sqrt{\frac{\left(\mu+\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i\right)\xi_{k+1}^2}{L} + 1}. \tag{79}$$

Now, suppose $\xi_{k+1} < \omega(t+1)$. Substituting it into (79), yields

$$\omega(t) \overset{(79)}{<} \omega(t+1) - \frac{1}{2}\sqrt{\frac{\left(\mu+\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i\right)\xi_{k+1}^2}{L} + 1}. \tag{80}$$

Then, applying (77) and the definition of δ, yields

$$\begin{aligned}
\omega(t) &\leq \omega(t+1) - \frac{1}{2}\sqrt{4\delta^2\left[\frac{\sqrt{2}}{4\delta}\sqrt{\frac{L}{\mu}}\left(e^{(t+2)\delta} - e^{-(t+2)\delta}\right)\right]^2 + 1} \\
&= \omega(t+1) - \frac{2}{4}\sqrt{\frac{L}{\mu}}\left[e^{(t+2)\delta} + e^{-(t+2)\delta}\right] \\
&= \omega(t+1) + \omega(t+1)'\left(t-(t+1)\right) \leq \omega(t),
\end{aligned}$$

where the last inequality follows from the supporting hyperplane theorem of convex functions. Evidently, this leads to a contradiction with our earlier assumption, which implies that $\xi_{k+1} < \omega(k+1), \forall k$. Therefore, (77) must hold true.

From (75) we can write

$$\lambda_k = \frac{L}{\left(\gamma_0 - \mu - \sum_{i=1}^{k-1} \beta_{i,k}\gamma_i\right)\xi_k^2}. \tag{81}$$

Then, recalling that $\gamma_0 \in \mathcal{R}_2$ and utilizing (77), we obtain

$$\frac{L}{\left(\gamma_0 - \mu - \sum_{i=1}^{k-1} \beta_{i,k}\gamma_i\right)\xi_k^2} \leq \frac{L(4\delta)^2}{\left(\gamma_0 - \mu - \sum_{i=1}^{k-1} \beta_{i,k}\gamma_i\right)\left[e^{(k+1)\delta} - e^{-(k+1)\delta}\right]^2}, \tag{82}$$

Last, applying the definition of δ in (82) we obtain the second bound presented in the lemma. □

Appendix F Theorem 2 and Corresponding Proof

Theorem 2. *In Algorithm 1 in the main paper, let $\mu > 0$. Then, the algorithm generates a sequence of points such that*

1. *If $\gamma_0 \in [0, \mu[$, then*

$$f(x_k) - f(x^*) \leq \frac{\mu||x_0 - x^*||^2}{h(k)} - (1 - \lambda_k)\psi_k(x^*). \tag{29}$$

2. *If $\gamma_0 \in [2\mu, 3L + \mu]$, then*

$$f(x_k) - f(x^*) \leq \frac{2L(\mu + \sum_{i=1}^{k-1} \beta_{i,k}\gamma_i)||x_0 - x^*||^2}{(\gamma_0 - \mu - \sum_{i=1}^{k-1} \beta_{i,k}\gamma_i)h(k)} - (1 - \lambda_k)\psi_k(x^*). \tag{30}$$

Thus, the method is optimal when the tolerance ϵ is small enough

$$0 < \epsilon \leq \frac{\mu}{2}R_0^2. \tag{31}$$

The lower bound on the number of iterations is

$$k_{\text{SFGM}} \geq \sqrt{\frac{L}{\mu + \sum_{i=1}^{k-1} \beta_{i,k}\gamma_i}} \left(ln\left(\frac{\mu R_0^2}{2\epsilon}\right) + ln(5)\right) \tag{32}$$

Proof. Combining the result of Theorem 1 and the inequality $f(x_0) - f^* \leq \frac{L}{2}||x_0 - x^*||^2$, we obtain

$$f(x_k) - f(x^*) \leq \frac{\lambda_k(L + \gamma_0)}{2}||x_0 - x^*||^2 - (1 - \lambda_k)\psi_k(x^*) \tag{83}$$

Let us first consider the case wherein $\gamma_0 \in [0, \mu[$. In this case, substituting the bound on the term λ_k obtained in point 1 of Lemma 4 in (83), yields (29).

From (29), we can observe that it is increasing in the values of γ_0. Thus, in the following analysis we will choose the smallest value for this parameter, i.e., $\gamma_0 = 0$. Then, relaxing the upper bound in (83), yields

$$f(x_k) - f(x^*) \leq \frac{2\mu||x_0 - x^*||^2}{e^{(k+1)\sqrt{\frac{\mu+\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i}{L}}} - 1}. \tag{84}$$

Therefore, in view of (31), our problem will be solved for $k_{\text{SFGM}} > \sqrt{\frac{L}{\mu+\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i}}\ln\big(1+\frac{2\mu R_0^2}{\epsilon}\big)$. Moreover, we have

$$\ln\left(1+\frac{2\mu R_0^2}{\epsilon}\right) \overset{(31)}{\leq} \ln\left(\frac{\mu R_0^2}{2\epsilon}+\frac{2\mu R_0^2}{\epsilon}\right) = \ln\left(\frac{5\mu R_0^2}{2\epsilon}\right). \tag{85}$$

Finally, the lower bound on the number of iterations for Algorithm 1 is

$$k_{\text{SFGM}} \geq \sqrt{\frac{L}{\mu+\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i}}\left(\ln\left(\frac{\mu R_0^2}{2\epsilon}\right)+\ln(5)\right) \tag{86}$$

As we will show later, the terms in the sequence $\{\gamma_k\}_k$ converge to μ at a much faster rate than the convergence of the iterates to x^*. In view of (13) in the main paper, and by making the appropriate selection for the coefficients $\beta_{i,k}$, we can ensure that the term $\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i$ also converges to μ at a much faster rate than the convergence of the iterates to x^*. Thus, the right hand side (RHS) of (86) becomes $\sqrt{\frac{L}{2\mu}}\left(\ln\left(\frac{\mu R_0^2}{2\epsilon}\right)+\ln(5)\right)$. In the sequel, we also present a scheme that attains the aforementioned lower bound.

From the lower complexity bounds for the class of smooth and strongly convex functions [1, (2.2.16)], we have that

$$k_{\text{bound}} \geq \frac{\sqrt{L/\mu}-1}{4}\ln\left(\frac{\mu R_0^2}{2\epsilon}\right). \tag{87}$$

Clearly, the bound obtained in (86) is proportional to (87). Thus, we conclude that for $\gamma_0 \in [0, \mu[$ our proposed SFGM is optimal.

Next, let us consider the case when $\gamma_0 \in [2\mu, 3L+\mu]$. In this case, substituting the bound on the term λ_k obtained in point 2 of Lemma 4 in (83), yields (30). Observe that the upper bound (30) is decreasing in γ_0 and increasing in the terms $\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i$. Thus, choosing $\gamma_0 = 3L+\mu$, letting $\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i = 0$

and relaxing the upper bound in (83) and doing some algebraic manipulations, yields

$$f(x_k) - f(x^*) \leq \frac{10\mu||x_0 - x^*||^2}{3\left(e^{(k+1)\sqrt{\frac{\mu+\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i}{L}}} - 1\right)}, \tag{88}$$

which has the same structure as the bound obtained in [1, Theorem 2.2.2]. Therefore, based on the results presented therein, we can again conclude that our proposed SFGM is optimal when $\gamma_0 \in [2\mu, 3L + \mu]$. □

References

[1] Y. Nesterov, *Lectures on convex optimization*. Springer, vol. 137, Dec. 2018.

Publication III

E. Dosti, S. A. Vorobyov, T. Charalambous. A new accelerated gradient-based estimating sequence technique for solving large-scale optimization problems with composite structure. In *IEEE 61st Conference on Decision and Control (CDC)*, Cancun, Mexico, 7516-7521, January 2023.

2022 IEEE 61st Conference on Decision and Control (CDC)
December 6-9, 2022. Cancún, Mexico

A new accelerated gradient-based estimating sequence technique for solving large-scale optimization problems with composite structure

Endrit Dosti, Sergiy A. Vorobyov and Themistoklis Charalambous

2022 IEEE 61st Conference on Decision and Control (CDC) | 978-1-6654-6761-2/22/$31.00 ©2022 IEEE | DOI: 10.1109/CDC51059.2022.9993313

***Abstract*— Various problems arising in control and data analysis can be formulated as large-scale convex optimization problems with a composite objective structure. Within the black-box optimization framework, such problems are typically solved by using accelerated first-order methods. The celebrated examples of such methods are the Fast Gradient Method and the Accelerated Multistep Gradient Method, designed by using the estimating sequences framework. In this work, we present a new class of estimating sequences, which are constructed by making use of a tighter lower bound on the objective function together with the gradient mapping technique. Based on the newly introduced estimating sequences, we construct a new method, which is also equipped with an efficient line-search strategy that provides robustness to the imperfect knowledge of the Lipschitz constant. Our proposed method enjoys the accelerated convergence rate, and our theoretical results are corroborated by numerical experiments conducted on real-world datasets. The experimental results also demonstrate the robustness of the initialization of the proposed method to the imperfect knowledge of the strong convexity parameter of the objective function.**

I. Introduction

Consider large-scale convex optimization problems with a composite objective of the type

$$\underset{x\in\mathcal{R}^n}{\text{minimize}} \quad \{F(x) = f(x) + g(x)\}, \tag{1}$$

where the function $f : \mathcal{R}^n \to \mathcal{R}$ has Lipschitz continuous gradients with Lipschitz constant L_f and is strongly convex with parameter μ_f, where $0 < \mu_f \leq L_f$. The regularizer $g : \mathcal{R}^n \to \mathcal{R}$ is a "simple" convex lower semi-continuous function with strong convexity parameter $\mu_g \geq 0$. The simplicity of g implies that its proximal map,

$$\text{prox}_{\tau g} \triangleq \arg\min_{z\in\mathcal{R}^n} \left(g(z) + \frac{1}{2\tau}||z - x||^2 \right), \tag{2}$$

where $x \in \mathcal{R}^n$ and $\tau > 0$, is computed with complexity $\mathcal{O}(n)$ [1]. Here $||\cdot||$ refers to the l_2 norm. Problems that have a composite objective, as shown in (1), arise in various areas of control, such as model predictive control, adaptive control, distributed systems, etc., [2], [3], and are solved iteratively using different first-order optimization algorithms [4].

A large portion of the recent research in first-order optimization has been targeted at investigating different reasons behind acceleration, as well establishing alternative frameworks [5]–[10]. Among the existing frameworks for the acceleration of first-order methods, the estimating sequences framework has grasped a significant interest (see [4] and references therein). Several reasons that have led to the popularity of methods built within this framework are the following. First, gradient-based methods built based on the estimating sequences framework are optimal in the sense of [11]. Second, as shown in [12], they can be combined with backtracking line search strategies, while maintaining their efficient convergence properties. Third, the estimating sequences framework can be used to build efficient accelerated second-order methods [13], and higher-order methods [14]. Last, they have demonstrated competitive performance even when extended to other settings, such as distributed optimization [15], nonconvex optimization [16], stochastic optimization [17], non-Euclidean optimization [18], etc. Despite their wide applicability and many desirable properties from the perspective of designing accelerated methods, estimating sequences are not unique and suffer from the lack of a systematic methodology for devising them. Thus, selecting the right construction for estimating functions can lead to more efficient algorithms compared to the existing state-of-the-art methods for solving problems of the type of (1).

The framework for the study and analysis of estimating sequence-based methods has been established in [20]. Among the existing estimating sequence methods devised for solving problems of the form of (1), a popular algorithm is the Accelerated Multistep Gradient Scheme (AMGS) [22, Method (4.9)]. It exhibits the theoretical accelerated rate of convergence $\mathcal{O}(\frac{1}{k^2})$ and is also very efficient in practice. However, it suffers from the increase in the computational burden due to the fact that it requires two projection-like operations per iteration. As the dimensionality of the problems increases, this can significantly affect the run-time of the minimization process. This issue has been addressed with the development of the Fast Iterative Shrinkage-Thresholding Algorithm (FISTA) [21], which exhibits the accelerated convergence rate with one projection-like operation per iteration.

Comparing AMGS to the Fast Gradient Method (FGM), which has been devised for minimizing smooth convex functions [19, Constant Step Scheme I], it can be observed that despite both exhibiting the accelerated convergence rate, the equations for updating the iterates are significantly different. These dissimilarities arise because the methods were devised using different variants of the estimating sequences and cause the practical performance of the methods to vary when they are compared on real-world problems. Based on our numerical experiments, we have observed that for smooth and strongly convex problems, FGM outperforms

E. Dosti and S. A. Vorobyov are with the Department of Signal Processing and Acoustics, Aalto University, Espoo, Cyprus. Emails: firstname.lastname@aalto.fi.

T. Charalambous is with the Department of Electrical and Computer Engineering, University of Cyprus, Nicosia, Cyprus and the Department of Electrical Engineering and Automation, School of Electrical Engineering, Aalto University, Espoo, Finland. Email: themistoklis.charalambous@aalto.fi.

both AMGS and FISTA. Thus, it is of interest to extend the variant of estimating sequences introduced in [19] and devise methods that can be used to find the optimal solution of (1). In this work, we show that by utilizing the aforementioned variant of estimating sequences it is possible to devise very efficient accelerated gradient-based algorithms. More specifically, we present a new structure for the estimating functions, which we call the composite estimating functions hereafter and show that they satisfy the properties of the estimating sequences. Different from the classical estimating functions devised in [19], our proposed composite estimating functions make use of the subgradients, as well as a tighter lower bound of the objective function. Utilizing the newly introduced estimating functions together with the gradient mapping framework, we devise our proposed method, which enjoys the accelerated convergence rate even when the true value of the Lipschitz constant is not known. The efficiency of our proposed algorithm in solving problems with composite structure is illustrated based on real-world datasets. Our numerical results also demonstrate the robustness of the initialization of the proposed algorithm with respect to the imperfect knowledge of the strong convexity parameter. Note that the need to estimate the true value of the strong convexity parameter comes with an additional increase in the computational complexity in practical implementations [23]. We remark that in this work we only present the proofs of the statements which are crucial in the development of the algorithm. The remaining proofs can be found in our full paper [24].

II. Foundations for composite objective optimization

First, we transfer the strong convexity of $g(x)$ within the objective function in (1). Let $x_0 \in \mathcal{R}^n$ and consider

$$\begin{aligned} F(x) &= \Big(f(x) + \frac{\tau\mu_g}{2}||x-x_0||^2\Big) + \tau\Big(g(x) - \frac{\mu_g}{2}||x-x_0||^2\Big) \\ &= \hat{f}(x) + \tau\hat{g}(x). \end{aligned} \tag{3}$$

This strong convexity transfer also yields $L_{\hat{f}} = L_f + \tau\mu_g$ and $\mu_{\hat{f}} = \mu_f + \tau\mu_g$, as well as $\mu_{\hat{g}} = 0$.

Next, the following bounds for the smooth and strongly convex function $\hat{f}(x)$ are introduced.

$$\hat{f}(x) \le \hat{f}(y) + \nabla\hat{f}(y)^T(x-y) + \frac{L_{\hat{f}}}{2}||y-x||^2, \tag{4}$$

$$\hat{f}(x) \ge \hat{f}(y) + \nabla\hat{f}(y)^T(x-y) + \frac{\mu_{\hat{f}}}{2}||y-x||^2, \tag{5}$$

where $y \in \mathcal{R}^n$. Similarly, using the definition of the subgradient, the term $\hat{g}(x)$ is bounded below as

$$\hat{g}(x) \ge \hat{g}(y) + s(y)^T(x-y), \tag{6}$$

where $s(y)$ is a subgradient of the function $\hat{g}(y)$. Moreover, let $L \ge L_{\hat{f}}$, and define the following

$$m_L(y;x) \triangleq \hat{f}(y) + \nabla\hat{f}(y)^T(x-y) + \frac{L}{2}||x-y||^2 + \tau\hat{g}(x). \tag{7}$$

Utilizing (4) in (7), we have

$$m_L(y;x) \ge F(x), \forall x, y \in \mathcal{R}^n. \tag{8}$$

Then, the composite gradient mapping can be introduced as

$$T_L(y) \triangleq \arg\min_{x\in\mathcal{R}^n} m_L(y;x), \tag{9}$$

and the composite reduced gradient can be defined as

$$r_L(y) \triangleq L\left(y - T_L(y)\right). \tag{10}$$

Observe that when $\tau = 0$, in (3) we have $\hat{f}(x) = f(x)$.In this case, note that the function $m_L(y;x)$ would also be differentiable in both x and y. Thus, the optimality condition for (9), would be $\nabla m_L(y;x) = 0$. Substituting the definition of $m_L(y;x)$ given in (7) into (9), and analyzing the first order condition, we can write $T_L(y) = y - \frac{\nabla\hat{f}(y)}{L}$. Substituting this into (10), results in $r_L(y) = \nabla F(y) = \nabla f(y)$, i.e., the composite reduced gradient becomes the gradient of the objective function. On the other hand, when $\tau \neq 0$, utilizing the first-order optimality conditions for (9), we have

$$\begin{aligned} \partial m_L(y;T_L(y))^T(x - T_L(y)) \ge 0, \\ \Big(\nabla\hat{f}(y) + L(T_L(y) - y) + \tau s_L(y)\Big)^T (x - T_L(y)) \ge 0, \end{aligned} \tag{11}$$

where ∂ denotes the subdifferential of $m_L(y;T_L(y))$, $s_L(y) \in \partial\hat{g}(T_L(y))$ is a subgradient belonging to the subdifferential of $\hat{g}(T_L(y))$. Setting the first bracket of (11) to 0 and using (10), we can compute the composite reduced gradient as

$$r_L(y) = L(y - T_L(y)) = \nabla\hat{f}(y) + \tau s_L(y). \tag{12}$$

In words, the choice of the subgradient $r_L(y)$ as in (12) ensures that $0 \in \partial m_L(y;T_L(y))$.

Now, we present the following lower on the objective function $F(x)$, which is tighter than utilizing (5).

Theorem 1. *Let $F(x)$ be a composition of an $L_{\hat{f}}$-smooth and $\mu_{\hat{f}}$-strongly convex function $\hat{f}(x)$, and a simple convex function $\hat{g}(x)$, as given in (3). For $L \ge L_{\hat{f}}$, and $x, y \in \mathcal{R}^n$ we have*

$$\begin{aligned} F(x) \ge \hat{f}(T_L(y)) + \tau\hat{g}(T_L(y)) + r_L(y)^T(x-y) \\ + \frac{\mu_{\hat{f}}}{2}||x-y||^2 + \frac{1}{2L}||r_L(y)||^2. \end{aligned} \tag{13}$$

III. Proposed method

First, we introduce the composite estimating sequences.

Definition 1. *The sequences $\{\phi_k\}_{k=0}^{\infty}$ and $\{\lambda_k\}_{k=0}^{\infty}$, $\lambda_k \ge 0$, are called composite estimating sequences of the function $F(\cdot)$ defined in (3), if $\lambda_k \to 0$ as $k \to \infty$, and $\forall x \in \mathcal{R}^n$, $\forall k \ge 0$ we have*

$$\phi_k(x) \le \lambda_k\phi_0(x) + (1-\lambda_k)F(x). \tag{14}$$

These sequences can be used to measure the rate of convergence of the iterates, as shown in the following lemma.

Lemma 1. *If for some sequence of points $\{x_k\}_{k=0}^{\infty}$ we have $F(x_k) \le \phi_k^* \triangleq \min_{x\in\mathcal{R}^n}\phi_k(x)$, then $F(x_k) - F(x^*) \le \lambda_k\left[\phi_0(x^*) - F(x^*)\right]$, where $x^* = \arg\min_{x\in\mathcal{R}^n} F(x)$.*

The following recursive definition for the proposed composite estimating sequences is introduced.

Lemma 2. *Assume that there exists a sequence $\{\alpha_k\}_{k=0}^{\infty}$, where $\alpha_k \in (0,1)\ \forall k$, such that $\sum_{k=0}^{\infty}\alpha_k = \infty$, and an arbitrary sequence $\{y_k\}_{k=0}^{\infty}$. Furthermore, let $\lambda_0 = 1$ and assume that the estimates L_k of the Lipschitz constant $L_{\hat{f}}$ are selected in a way that inequality (4) is satisfied for all the iterates x_k and y_k. Then, the sequences $\{\phi_k\}_{k=0}^{\infty}$ and $\{\lambda_k\}_{k=0}^{\infty}$, which are defined recursively as*

$$\lambda_{k+1} = (1-\alpha_k)\lambda_k, \tag{15}$$

$$\begin{aligned}\phi_{k+1}(x) = {} & (1-\alpha_k)\phi_k(x) + \alpha_k F\left(T_{L_k}(y_k)\right) \\ & + \alpha_k\left(r_{L_k}(y_k)^T(x-y_k) + \frac{\mu_f}{2}||x-y_k||^2\right) \\ & + \alpha_k\frac{1}{2L_k}||r_{L_k}(y_k)||^2,\end{aligned} \tag{16}$$

are composite estimating sequences.

We proceed by contrasting our results introduced in Definition 1, Lemmas 1 and 2 with their counterpart devised for minimizing smooth convex functions presented in [19, Definition 2.2.1, Lemma 2.2.1, Lemma 2.2.2]. First, note that Definition 1 and Lemma 1 would reduce to the corresponding results introduced by Nesterov, which are limited to the case of minimizing differentiable objective functions. In this sense, the framework proposed here extends the work presented in [19] to solve a more general class of problems. Second, as established in Lemma 1, the rate of convergence of the iterates depends on the rate at which $\lambda_k \to 0$. Third, in (16) we can see the impact of the tighter lower bound on the objective function presented in Theorem 1.[1] Last, the cost function in (16) is now evaluated at the points given by the composite gradient mapping. Moreover, unlike FGM, which is defined only when the computation of the gradient of the objective function is possible, we can observe that our proposed composite estimating functions utilize subgradients of the non-smooth objective function to construct the sequence $\{\phi_k\}_{k=0}^{\infty}$.

Let us now introduce the following structure for the functions in the sequence $\{\phi_k\}_{k=0}^{\infty}$

$$\phi_k(x) = \phi_k^* + \frac{\gamma_k}{2}||x-v_k||^2,\ \forall k = 1,2,\ldots. \tag{17}$$

Note that the selection for the terms in $\{\phi_k(x)\}_{k=0}^{\infty}$ is not unique and that different choices for $\phi_k(x)$ can lead to different accelerated methods (see [18], [25]). Let us then show how the terms in the sequences $\{\gamma_k\}_{k=0}^{\infty}$, $\{v_k\}_{k=0}^{\infty}$ and $\{\phi_k^*\}_{k=0}^{\infty}$ can be computed recursively.

Lemma 3. *Let $\phi_0(x) = \phi_0^* + \frac{\gamma_0}{2}||x-v_0||^2$, where $\gamma_0 \in \mathcal{R}^+$ and $v_0 \in \mathcal{R}^n$. Then, the process defined in Lemma 2 preserves the canonical form of the function $\{\phi_k(x)\}_{k=0}^{\infty}$ presented in (17), where the sequences $\{\gamma_k\}_{k=0}^{\infty}$, $\{v_k\}_{k=0}^{\infty}$ and $\{\phi_k^*\}_{k=0}^{\infty}$ can be computed as follows*

$$\gamma_{k+1} = (1-\alpha_k)\gamma_k + \alpha_k\mu_{\hat{f}}, \tag{18}$$

$$v_{k+1} = \frac{1}{\gamma_{k+1}}\Big((1-\alpha_k)\gamma_k v_k + \alpha_k\left(\mu_{\hat{f}}y_k - L_k(y_k - T_{L_k}(y_k))\right)\Big), \tag{19}$$

$$\begin{aligned}\phi_{k+1}^* = {} & (1-\alpha_k)\phi_k^* + \alpha_k\left(F\left(T_{L_k}(y_k)\right) + \frac{1}{2L_k}||r_{L_k}(y_k)||^2\right) \\ & - \frac{L_k^2\alpha_k^2}{2\gamma_{k+1}}||y_k - T_{L_k}(y_k)||^2 + \frac{\mu_{\hat{f}}\alpha_k\gamma_k(1-\alpha_k)}{2\gamma_{k+1}}||y_k - v_k||^2 \\ & + \frac{L_k\alpha_k\gamma_k(1-\alpha_k)}{\gamma_{k+1}}(y_k - v_k)^T(y_k - T_{L_k}(x_k)).\end{aligned} \tag{20}$$

Proof: See Appendix I.

Due to space limitations, some derivations have been omitted at this point. They can which be found in our full paper [24]. Moreover, to obtain more intuition behind the estimating sequences and methods obtained by utilizing this framework, we refer the reader to [20], [26], [27].

Unlike the analysis presented in [19], the results obtained in this work also allow for the line search adaptation.[2] To achieve a faster progress to the optimal solution, it would be preferable to select the smallest constant L_k such that (4), with value $L_{\hat{f}} = L_k$, is satisfied $\forall k = 0,1,\ldots$, and then slightly increase its value across the iterations. This approach would ensure that the algorithm makes "larger steps towards the optimal solution" in the initial iterations. Then, as x_k approaches x^*, the larger values of L_k would ensure that the algorithm does not overshoot past x^* and behave erratically. Unfortunately, such an approach is not feasible because the true value of $L_{\hat{f}}$ is not known. Therefore, we introduce a line search strategy that has the following benefits: *i)* Guarantees the robustness of the method with respect to the initialization of the estimate of the Lipschitz constant. *ii)* Ensures a dynamic update of the step size across the iterations. The line search strategy that is utilized in this work makes use of a constant $\eta_u > 1$, which increases the value of the estimate and a constant $\eta_d \in]0,1[$, which decreases the value of the estimate of the Lipschitz constant. Finally, our proposed algorithm is presented in Algorithm 1.

Contrasting our proposed method and FGM, i.e., Constant Step Scheme I in [19], we can see that the terms in $\{\alpha_k\}_{k=0}^{\infty}$ and $\{\gamma_k\}_{k=0}^{\infty}$ are updated in a similar manner. The first dissimilarity can be observed in the updates of $\{y_k\}_{k=0}^{\infty}$, which for the proposed method are independent of $\mu_{\hat{f}}$. The second dissimilarity can be noticed from the update of x_k. Because of the composite structure of $F(x)$, the next iterate is now computed by taking a proximal gradient step. Note that the assumption on the simplicity of $g(x)$ ensures that the proximal term can be computed efficiently. The third dissimilarity is the way $\{v_k\}_{k=0}^{\infty}$ is computed. It reflects the usage of the proposed composite reduced gradient. Last, we note that the proposed convergence analysis ensures the converge of our proposed method for a wider range of values for γ_0 than what is supported by the existing convergence results for FGM [19, Lemma 2.2.4], which ensure convergence for $\gamma_0 \in [\mu_{\hat{f}}; 3L_{\hat{f}} + \mu_{\hat{f}}]$. As we will

[1] When $F(x)$ is a convex and differentiable function, the composite reduced gradient becomes the same as the gradient of the function.

[2] Several backtracking strategies have been proposed in the literature (see for instance [21], [22]).

[3] We note that $K_{\max}$ denotes the maximum number of iterations. Depending on the application, the value of $K_{\max}$ can be selected to trade-off between the required accuracy, and the needed processing time and computations.

Algorithm 1 Proposed Method

1: **Input** $x_0 \in \mathcal{R}^n$, $L_0 > 0$, $\mu_{\hat{f}}$, $\gamma_0 \in [0, 3L_0 + \mu_{\hat{f}}]$, $\eta_u > 1$ and $\eta_d \in]0, 1[$.
2: **Set** $k = 0$, $i = 0$ and $v_0 = x_0$.
3: **while** $k \leq K_{\max}$[3] **do**
4: $\quad \hat{L}_i \leftarrow \eta_d L_k$
5: $\quad$ **while** True **do**
6: $\quad\quad \hat{\alpha}_i \leftarrow \frac{(\mu_{\hat{f}} - \gamma_k) + \sqrt{(\mu_{\hat{f}} - \gamma_k)^2 + 4\hat{L}_i \gamma_k}}{2\hat{L}_i}$
7: $\quad\quad \hat{\gamma}_{i+1} \leftarrow (1 - \hat{\alpha}_i)\gamma_k + \hat{\alpha}_i \mu_{\hat{f}}$
8: $\quad\quad \hat{y}_i \leftarrow \frac{\hat{\gamma}_{i+1} x_k + \hat{\alpha}_i \gamma_k v_k}{\hat{\gamma}_{i+1} + \hat{\alpha}_i \gamma_k}$
9: $\quad\quad \hat{x}_{i+1} \leftarrow \text{prox}_{\frac{1}{\hat{L}_i}\hat{g}}\left(\hat{y}_i - \frac{1}{\hat{L}_i}\nabla f(\hat{y}_i)\right)$
10: $\quad\quad \hat{v}_{i+1} \leftarrow \frac{1}{\hat{\gamma}_{i+1}}\left((1 - \hat{\alpha}_i)\gamma_k v_k + \hat{\alpha}_i\left(\mu_{\hat{f}}\hat{y}_i - \hat{L}_i(\hat{y}_i - \hat{x}_{i+1})\right)\right)$
11: $\quad\quad$ **if** $F(\hat{x}_{i+1}) \leq m_{\hat{L}_i}(\hat{y}_i, \hat{x}_{i+1})$ **then**
12: $\quad\quad\quad$ Break from loop
13: $\quad\quad$ **else**
14: $\quad\quad\quad \hat{L}_{i+1} \leftarrow \eta_u \hat{L}_i$
15: $\quad\quad$ **end if**
16: $\quad\quad i \leftarrow i + 1$
17: $\quad$ **end while**
18: $\quad L_{k+1} \leftarrow \hat{L}_i$, $x_{k+1} \leftarrow \hat{x}_i$, $\alpha_k \leftarrow \hat{\alpha}_{i-1}$, $y_k \leftarrow \hat{y}_{i-1}$, $i \leftarrow 0$, $k \leftarrow k + 1$
19: **end while**
20: **Output** x_k

see later, setting $\gamma_0 = 0$, ensures the robustness of the initialization of the proposed method with respect to the imperfect knowledge of the strong convexity parameter.

Let us also analyze the behavior of the estimate of the Lipschitz constant. Depending on the selection of L_0, two possibilities exist. First, if $L_0 \in]0, L_{\hat{f}}[$, then from line 11 in Algorithm 1, we observe that the estimate of the Lipschitz constant at iteration k increases only if $L_{k-1} \leq L_{\hat{f}}$. Thus, it can be written that

$$L_0 \leq \hat{L}_i \leq L_k \leq \eta_u L_{\hat{f}}. \tag{21}$$

Second, if $L_0 \geq L_{\hat{f}}$, then the condition in line 11 of Algorithm 1 is satisfied, and estimate of the Lipschitz constant cannot increase further. Therefore, we would have

$$L_k \leq \eta_d L_0. \tag{22}$$

Combining (21) and (22), we note that despite of the value of L_0, we have

$$L_k \leq L_{\max} \triangleq \max\{\eta_d L_0, \eta_u L_{\hat{f}}\}. \tag{23}$$

Finally, we can characterize the convergence rate of the proposed method as follows.

Theorem 2. *Algorithm 1 generates a sequence of points such that*

1) *If* $\gamma_0 \in [0, \mu_{\hat{f}}[$, *then*

$$F(x_k) - F(x^*) \leq \frac{\mu_{\hat{f}}(L_0 + \gamma_0)||x_0 - x^*||^2}{L_k\left(e^{\frac{k+1}{2}\sqrt{\frac{\mu_{\hat{f}}}{L_k}}} - e^{-\frac{k+1}{2}\sqrt{\frac{\mu_{\hat{f}}}{L_k}}}\right)^2} \tag{24}$$

2) *If* $\gamma_0 \in [\mu_{\hat{f}}, 3L_0 + \mu_{\hat{f}}]$, *then*

$$F(x_k) - F(x^*) \leq \frac{2\mu_{\hat{f}}(L_0 + \gamma_0)||x_0 - x^*||^2}{(\gamma_0 - \mu_{\hat{f}})\left(e^{\frac{k+1}{2}\sqrt{\frac{\mu}{L_k}}} - e^{-\frac{k+1}{2}\sqrt{\frac{\mu}{L_k}}}\right)^2} \tag{25}$$

From Theorem 2 we observe that, compared to FGM, the proposed method converges over a larger selection of values of the term γ_0. Moreover, we can see that initializing $\gamma_0 = 0$ exhibits the best theoretical performance. This is important from a practical perspective, since in most cases the true values of $\mu_{\hat{f}}$ and $L_{\hat{f}}$ are not known and should be estimated. The convergence rate is also affected from the selection of L_0. From (24) and (25) we can see that the smaller L_0, the faster the convergence of the method. At this point, we stress that L_0 cannot be arbitrarily small, as it should still be chosen in a way that the upper bound (4) is satisfied. Moreover, it should also have a larger value than our estimate of the strong convexity parameter $\mu_{\hat{f}}$.

IV. Numerical Study

In this section, we test the performance of several instances of our proposed method in solving

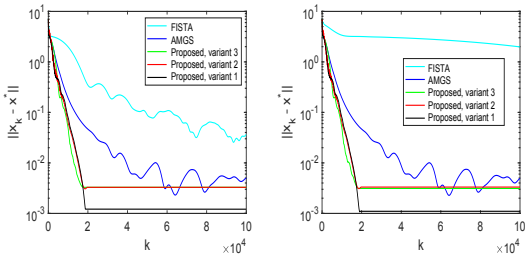
$$\underset{x \in \mathcal{R}^n}{\text{minimize}} \quad \frac{1}{2}\sum_{i=1}^{m}(a_i^T x - y_i)^2 + \frac{\tau_1}{2}||x||^2 + \tau_2||x||_1, \tag{26}$$

where $||\cdot||_1$ denotes the l_1 norm. The performance of our proposed method is compared to the state-of-the-art black-box methods, i.e., AMGS and FISTA. For the proposed method, we consider the variant that yields the best theoretical performance, i.e., when we initialize $\gamma_0 = 0$. In the plots, it is named "Proposed, variant 1". We also examine the variant for which (in theory) the convergence rate is slowest, i.e., we choose $\gamma_0 = 3L_0 + \mu_{\hat{f}}$, and it is named Proposed, variant 2". Lastly, we examine the instance of the proposed method that is obtained when $\gamma_0 = \mu_{\hat{f}}$, which is named "Proposed, variant 3". For both AMGS and FISTA we utilize the line-search strategies presented in the respective papers [21], [22]. We demonstrate the robustness of the line-search strategy that is used in the proposed method, we depict the following instances. *i)* We initialize the estimate of the Lipschitz constant to be 10-times smaller than the true value, i.e., $L_0 = 0.1L_{\hat{f}}$. *ii)* We initialize the estimate of the Lipschitz constant to be 10-times larger than the true value, i.e., $L_0 = 10L_{\hat{f}}$. Moreover, we choose the parameters $\eta_u = 2$ and $\eta_d = 0.9$ based on [30] because they ensure "a good performance of the methods across many applications". Regarding the strong convexity parameter, we have already discussed that, from a computational viewpoint, $\mu_{\hat{f}}$ is expensive to estimate in practice. Therefore, to decrease the number of computations, we equate the strong convexity parameter to that of the regularizer term in (26). Furthermore, we choose the starting point x_0 at random for all algorithms.

We compare the performance of the methods on real data, which are selected from the Library for Support Vector Machines (LIBSVM) [28]. Specifically, we consider the datasets "a1a" and "colon-cancer". For the "a1a" dataset,

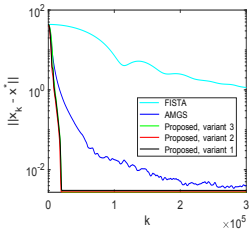
we have $A \in \mathcal{R}^{1605\times 123}$. On the other hand, for the "colon-cancer" dataset, we have $A \in \mathcal{R}^{62\times 2000}$. For these datasets, the respective true values for the Lipschitz constants are $L_{\text{"a1a"}} = 10061$ and $L_{\text{"colon-cancer"}} = 1927.4$. Moreover, we consider the following assignment for the regularizer term $\tau_1 = \tau_2 \in \{10^{-5}, 10^{-6}\}$. Evidently, such choice of the regularizer terms ensures a very large condition number $\kappa = \frac{L_{\hat{f}}}{\mu_{\hat{f}}}$ for the problems that are considered in this section. We find the optimal solutions via CVX [29].

From Fig. 1 we can observe that all the instances of the proposed method exhibit a much better performance than the existing benchmarks. First, observe that the final iterate produced by any of the variants of the proposed method is the closest to x^*. Second, notice that unlike the iterates produced by AMGS or FISTA, the sequence of iterates constructed by the proposed method converges to the optimal solution x^* in a much smaller number of iterations. Moreover, we can see that the performance of FISTA is visibly worse than the other accelerated methods. This occurs partly because the FISTA algorithm cannot exploit the strong convexity of the objective function. Third, notice that the practical performance of both the proposed method and AMGS is not altered by the inexact knowledge of L_0. However, unlike AMGS which requires two projection-like operations per iteration, our proposed method retains the robustness to L_0 at a lower computational cost. On the other hand, the results for FISTA suggest that the initialization of the Lipschitz constant significantly affects its performance. Last, observe that all the different variants of the proposed method exhibit very similar convergence properties and their differences in performance are minor. As can be seen from Figs. 1(a) and 1(b), the variant obained under the initialization $\gamma_0 = 0$ exhibits a faster convergence. Such a result is highly relevant in practical applications, wherein the exact values of $\mu_{\hat{f}}$ and $L_{\hat{f}}$ are approximated by using some numerical procedure. Thus, we can see that the variant of the proposed method which results from choosing $\gamma_0 = 0$ exhibits better convergence properties than the selected benchmarks, and at the same time is also more robust to the imperfect knowledge of the strong convexity parameter and the Lipschitz constant.

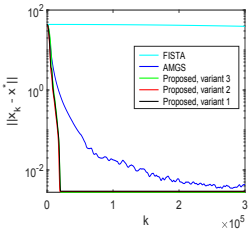


(a) Decreasing the distance to x^* on "a1a" dataset, $L_0 = 0.1L_{\text{"a1a"}}$ and $\tau_1 = \tau_2 = 10^{-5}$.

(b) Decreasing the distance to x^* on "a1a" dataset, $L_0 = 10L_{\text{"a1a"}}$ and $\tau_1 = \tau_2 = 10^{-5}$.



(c) Decreasing the distance to x^* on "colon-cancer" dataset, $L_0 = 0.1L_{\text{"colon-cancer"}}$ and $\tau_1 = \tau_2 = 10^{-6}$.



(d) Decreasing the distance to x^* on "colon-cancer" dataset, $L_0 = 10L_{\text{"colon-cancer"}}$ and $\tau_1 = \tau_2 = 10^{-6}$.

Fig. 1: Comparison between the efficiency and robustness with respect to the initialization of the Lipschitz constant of the algorithms tested in minimizing the quadratic loss function with elastic net regularizer on real data.

V. Conclusions and Discussion

A new accelerated black-box gradient-based estimating sequence method for solving problems with composite objective has been presented. The proposed method has been devised by utilizing a newly introduced class of estimating functions and it is equipped with an efficient line-search strategy. The newly introduced estimating functions have been used to construct upper bounds on the non-smooth function, as well as to measure the convergence rate of the minimization process. Different from the existing convergence results of FGM-type methods, our proposed analysis supports the adjustment of the estimate of the Lipschitz constant. Moreover, our proposed method converges when $\gamma_0 \in [0, 3L + \mu_{\hat{f}}]$. In practice, our findings establish the possibility of constructing accelerated estimating sequences methods, which also enjoy the robustness to the imperfect knowledge of the Lipschitz constant and strong convexity parameter. We note that the robustness to the strong convexity parameter is of significant importance in practice since its true value is computationally expensive to obtain. Our theoretical findings are supported by numerical experiments performed on real-world datasets.

The framework and results that were presented in this work can be extended in various directions. First, it would be of interest for networked control applications to investigate the possibilities of constructing a distributed variant of our proposed method, which would also improve the scalability of the proposed framework. Second, it would be of interest to construct extensions of our proposed framework to the stochastic optimization framework. Last, it would also be of interest to investigate the possibility of embedding other types of momentum terms (e.g., heavy-ball momentum) into the proposed estimating sequences to further improve the convergence properties of our proposed method.

Appendix I
Proof of Lemma 3

We will prove the first part by induction. First, considering $k = 0$, we can write $\nabla^2\phi_0(x) = \gamma_0 I$. Next, we suppose that for some step k we have $\nabla^2\phi_k(x) = \gamma_k I$. Then, at the next step $k+1$, it can be written that

$$\nabla^2\phi_{k+1}(x) \overset{(16)}{=} (1-\alpha_k)\gamma_k I + \alpha_k\mu_{\hat{f}} I \equiv \gamma_{k+1} I. \qquad (27)$$

At this point, we are ready to show how the recursive equations for updating $\{v_k\}_{k=0}^{\infty}$ and $\{\phi_k^*\}_{k=0}^{\infty}$ can be obtained.

$$\phi^*_{k+1} + \frac{\gamma_{k+1}}{2}||y_k - v_{k+1}||^2 = (1-\alpha_k)\left(\phi^*_k + \frac{\gamma_k}{2}||y_k - v_k||^2\right) + \alpha_k\left(F\left(T_{L_k}(y_k)\right) + \frac{1}{2L_k}||r_{L_k}(y_k)||^2\right). \tag{34}$$

$$\frac{\gamma_{k+1}}{2}||y_k - v_{k+1}||^2 = \frac{(1-\alpha_k)^2\gamma_k^2}{2\gamma_{k+1}}||v_k - y_k||^2 + \frac{\alpha_k^2 {L_k}^2}{2\gamma_{k+1}}||y_k - T_{L_k}(y_k)||^2 - \frac{2L_k\alpha_k(1-\alpha_k)\gamma_k}{2\gamma_{k+1}}(v_k - y_k)^T\nabla\left(y_k - T_{L_k}(y_k)\right). \tag{38}$$

Utilizing (17) in (16), as well as considering its first-order optimality conditions, yields

$$\gamma_{k+1}(x - v_{k+1}) = \gamma_k(1-\alpha_k)(x - v_k) + \alpha_k\left(\mu_{\hat{f}}(x - y_k) + r_{L_k}(y_k)\right). \tag{28}$$

Substituting (18) in (28), and discarding the terms that depend on x, it can be written that

$$-\gamma_{k+1}v_{k+1} = -(1-\alpha_k)\gamma_k v_k + \alpha_k\left(-\mu_{\hat{f}}y_k + r_{L_k}(y_k)\right). \tag{29}$$

Then, utilizing (10) in (29), yields (19).

Now, we are ready to proceed with establishing (20). Exploiting (17) in (16), now evaluated at the point $x = y_k$, we obtain (34), shown at the top of the page.

Then, we utilize (19) to find an alternative characterization for the second term in the left hand side (LHS) of (34). Let us examine the following

$$v_{k+1} - y_k = \frac{1}{\gamma_{k+1}}((1-\alpha_k)\gamma_k v_k + \alpha_k\mu_{\hat{f}}y_k - \alpha_k L_k\left(y_k - T_{L_k}(y_k)\right) - \gamma_{k+1}y_k). \tag{35}$$

Then, substituting (18) in (35), it can be written that

$$v_{k+1} - y_k = \frac{1}{\gamma_{k+1}}((1-\alpha_k)\gamma_k(v_k - y_k) - \alpha_k L_k\left(y_k - T_{L_k}(y_k)\right). \tag{36}$$

Considering the $||\cdot||^2$ of the LHS and RHS in (36), we reach

$$||y_k - v_{k+1}||^2 = \frac{||(1-\alpha_k)\gamma_k(v_k - y_k) - \alpha_k L_k(y_k - T_{L_k}(y_k))||^2}{\gamma_{k+1}^2}. \tag{37}$$

Multiplying the LHS and RHS of (37) by $\frac{\gamma_{k+1}}{2}$, and expanding the RHS, we obtain (38) shown at the top of the page. Utilizing (38) in (34), yields (20).

REFERENCES

[1] N. Parikh, S. Boyd, "Proximal Algorithms," *Foundations and Trends in Optimization*, vol. 1, no. 3, pp. 127–239, Jan. 2014.

[2] T. Alamo, P. Krupa and D. Limon, "Restart of accelerated first order methods with linear convergence under a quadratic functional growth condition," *IEEE Trans. on Autom. Cont.* (Eearly Access), Jan. 2022.

[3] J. F. Gaudio, T. E. Gibson, A. M. Annaswamy, M. A. Bolender and E. Lavretsky, "Connections between adaptive control and optimization in machine learning," in *58th IEEE Conference on Decision and Control*, Nice, France, Dec 2019, pp. 4563–4568.

[4] A. d'Aspremont, D. Scieur, and A. Taylor, *Acceleration Methods*. *arXiv:2101.09545*, Jan. 2021.

[5] Z. Allen-Zhu and L. Orecchia, "Linear coupling: An ultimate unification of gradient and mirror descent," *arXiv: 1407.1537*, Nov. 2016.

[6] S. Bubeck, Y. T. Lee and M. Singh, "A geometric alternative to Nesterov's accelerated gradient descent," *arXiv: 1506.08187*, Jun. 2015.

[7] W. Su, S. Boyd and E. J. Candès, "A differential equation for modeling Nesterov's accelerated gradient method: Theory and insights," *Journal of Machine Learning Research*, vol. 17, no. 153, pp. 1–43, Jan. 2016.

[8] L. Lessard, B. Recht and A. Packard, "Analysis and design of optimization algorithms via integral quadratic constraints," *SIAM Journal on Optimization*, vol. 26, no. 1, pp. 57–95, Jan. 2016.

[9] Y. Drori and M. Teboulle, "Performance of first-order methods for smooth convex minimization: a novel approach," *Mathematical Programming*, vol. 145, no. 1, pp. 451–482, Jun. 2014.

[10] B. Van Scoy, R. A. Freeman and K. M. Lynch, "The fastest known globally convergent first-order method for minimizing strongly convex functions," *IEEE Cont. Syst. Let.*, vol. 2, no. 1, pp. 49-54, Jan. 2018.

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

[12] M. I. Florea and S. A. Vorobyov, "An accelerated composite gradient method for large-scale composite objective problems," *IEEE Transactions on Signal Processing*, vol. 67, no. 2, pp. 444–459, Jan. 2019.

[13] Y. Nesterov, "Accelerating the cubic regularization of Newton's method on convex problems," *Mathematical Programming*, vol. 112, no. 1, pp. 159–181, Mar. 2008.

[14] Y. Nesterov, "Inexact high-order proximal-point methods with auxiliary search procedure," *CORE*, Oct. 2020.

[15] D. Jakovetić, J. Xavier and J. M. F. Moura, "Fast distributed gradient methods," *IEEE Transactions on Automatic Control*, vol. 59, no. 5, pp. 1131–1146, Jan. 2014.

[16] S. Ghadimi and G. Lan, "Accelerated gradient methods for nonconvex nonlinear and stochastic programming," *Mathematical Programming*, vol. 156, no. 1-2, pp. 59–99, Mar. 2016.

[17] A. Kulunchakov and J. Mairal, "Estimate Sequences for stochastic composite optimization: Variance reduction, acceleration, and robustness to noise," *Journal of Machine Learning Research*, vol. 21, no. 155, pp. 1–52, Jul. 2020.

[18] K. Ahn and S. Sra, "From Nesterov's estimate sequence to Riemannian acceleration," in *Proceedings Conference on Learning Theory*, Graz, Austria, Jul. 2020, pp. 88–118.

[19] Y. Nesterov, *Lectures on convex optimization*. Springer, vol. 137, Dec. 2018.

[20] M. Baes, "Estimate sequence methods: Extensions and approximations," *Inst. for Oper. Res., ETH, Switzerland*, Aug. 2020.

[21] A. Beck and M. Teboulle, "A fast iterative shrinkage-thresholding algorithm for linear inverse problems," *SIAM Journal on Imaging Sciences*, vol. 2, no. 1, pp. 183–202, Mar. 2009.

[22] Y. Nesterov, "Gradient methods for minimizing composite objective function," *Math. Prog.*, vol. 140, no. 1, pp. 125–161, Aug. 2013.

[23] E. K. Ryu and S. Boyd, "Primer on monotone operator methods," *Applied Computational Mathematics*, vol. 15, no. 1, pp. 3–43, Jan. 2016.

[24] E. Dosti, S. A. Vorobyov and T. Charalambous, "A New Class of Composite Objective Multi-step Estimating-sequence Techniques (COMET)," *arXiv: 2111.06763*, Nov. 2021.

[25] H. Zhang and S. Sra, "An estimate sequence for geodesically convex optimization," in *Proceedings of Conference on Learning Theory*, Stockholm, Sweden, Jul. 2018, pp. 1703–1723.

[26] E. Dosti, S. A. Vorobyov and T. Charalambous, "Generalizing Nesterov's acceleration framework by embedding momentum into estimating sequences: New algorithm and bounds," in *IEEE Int. Symp. on Inf. Theo.*, Helsinki, Finland, Jul. 2022, pp. 1703–1723.

[27] E. Dosti et al., "Embedding a heavy-ball type of momentum into the estimating sequences," *arXiv: 2008.07979*, Aug. 2020.

[28] C. C. Chang and C. J. Lin, "LIBSVM: A library for support vector machines," *ACM Transactions on Intelligent Systems and Technology*, vol. 2, no. 3, pp. 1–27, May. 2011.

[29] M. Grant, S. Boyd and Y. Ye, "CVX: Matlab software for disciplined convex programming (web page and software)," 2009.

[30] S. R. Becker, E. J. Candès and M. Grant, "Templates for convex cone problems with applications to sparse signal recovery," *Mathematical Programming: Computation*, vol. 3, no. 3, pp. 165, Sep. 2011.

Publication IV

E. Dosti, S. A. Vorobyov, T. Charalambous. A new class of composite objective multistep estimating sequence techniques. *Signal Processing*, 206, 108889, December 2022.

Signal Processing 206 (2023) 108889

Contents lists available at ScienceDirect

Signal Processing

journal homepage: www.elsevier.com/locate/sigpro

A new class of composite objective multistep estimating sequence techniques

Endrit Dosti [a], Sergiy A. Vorobyov [a,*], Themistoklis Charalambous [a,b]

[a] School of Electrical Engineering, Aalto University, Espoo, Finland
[b] Department of Electrical and Computer Engineering, University of Cyprus, Nicosia, Cyprus

ARTICLE INFO

Article history:
Received 22 April 2022
Revised 31 October 2022
Accepted 12 December 2022
Available online 16 December 2022

Keywords:
Accelerated first-order methods
Large-scale optimization
Composite objective
Estimating sequence
Gradient mapping
Line-search

ABSTRACT

A plethora of problems arising in signal processing, machine learning and statistics can be cast as large-scale optimization problems with a composite objective structure. Such problems are typically solved by utilizing iterative first-order algorithms. In this work, we devise a new accelerated gradient-based estimating sequence technique for solving large-scale optimization problems with composite objective structure. Specifically, we introduce a new class of estimating functions, which are obtained by utilizing both a tight lower bound on the objective function, as well as the gradient mapping technique. Then, using the proposed estimating functions, we construct a class of Composite Objective Multi-step Estimating-sequence Techniques (COMET), which are endowed with an efficient line-search procedure. We prove that our proposed COMET enjoys the accelerated convergence rate, and our newly established convergence results allow for step-size adaptation. Our theoretical findings are supported by extensive computational experiments on various problem types and real-world datasets. Moreover, our numerical results show evidence of the robustness of the proposed method to the imperfect knowledge of the smoothness and strong convexity parameters.

1. Introduction

In this work, we devise accelerated black-box methods for solving large-scale convex optimization problems with a composite objective structure by using only first-order information. The typical structure of such problems is

$$\underset{x\in\mathcal{R}^n}{\text{minimize}} \quad F(x) = f(x) + \tau g(x), \tau > 0, \tag{1}$$

where the function $f : \mathcal{R}^n \to \mathcal{R}$ is an L_f-smooth and μ_f-strongly convex function with $0 \le \mu_f \le L_f$. The regularizer $g : \mathcal{R}^n \to \mathcal{R}$ is a simple convex lower semi-continuous function with strong convexity parameter μ_g. Typically, in signal processing applications, the function $g(x)$ is "simple", meaning that a closed-form solution for minimizing the summation of g and some auxiliary functions can be easily found [1]. In more practical terms, the assumption on the simplicity of g implies that its proximal map, defined as

$$\text{prox}_{\tau g} \triangleq \underset{z\in\mathcal{R}^n}{\text{argmin}} \quad \left(g(z) + \tfrac{1}{2\tau}||z-x||^2\right), x \in \mathcal{R}^n, \tag{2}$$

is computed with complexity $\mathcal{O}(n)$ [2]. Herein $||\cdot||$ denotes the l_2 norm.

Problems that share the same structure as (1) arise quite often in different scientific disciplines, such as signal and image processing, data analysis, and machine learning. Typical applications in which the formulation given in (1) is relevant include compressive sensing, phase retrieval problems, medical imaging, dictionary learning, and many more (see [3,5–7,4] and references therein). When considering applications, the variable x represents the model parameters, whereas the role of $f(x)$ is to ensure a good fit between the observed data and the estimated parameters. In signal processing applications, $g(x)$ acts as a regularizer and typically takes the form of some parameter shrinkage norm, i.e., l_2 norm [8,9], sparsity-enforcing norm, i.e., l_1 norm [10–12], or its counterpart for the rank function, i.e., the nuclear norm [13,14]. Another popular structure for $g(x)$ is the Chebyshev norm, i.e., the l_∞ norm [15]. The function $g(x)$ can also be used to embed convex constraints, in which case it would act as an indicator function of some closed convex set [1].

In the context of large-scale optimization [16], problems that share the same structure as (1) are solved iteratively using different first-order optimization algorithms [17,18]. The bounds on the performance of black-box first-order methods have been es-

* Corresponding author.
E-mail addresses: endrit.dosti@aalto.fi (E. Dosti), sergiy.vorobyov@aalto.fi (S.A. Vorobyov), themistoklis.charalambous@aalto.fi (T. Charalambous).

https://doi.org/10.1016/j.sigpro.2022.108889

tablished by Nemirovsky and Yudin [19]. Loosely speaking, a first-order method is optimal in the black-box framework if it achieves the accelerated convergence rate with respect to the iteration counter k, while at the same time complying with the lower complexity bounds. The question of how to construct practical methods that are optimal has attracted the attention of the research community over decades. One of the first methods that managed to achieve the accelerated convergence rate in the black-box framework was the heavy ball method [20]. Therein, the acceleration is achieved by adding a momentum term to the gradient step, which nudges the new iterate in the direction of the previous step. The first method that is optimal in the sense of Nemirovsky and Yudin [19] is the Fast Gradient Method (FGM) [21]. It is built based on the mathematical machinery of estimating sequences, and has been since widely studied [22–27].

Finding different reasons behind acceleration has attracted significant attention in the recent research on first-order optimization. In [28], the authors have constructed accelerated first-order methods by exploiting the linear coupling between mirror and gradient descent. The framework presented therein leads to a myriad of applications wherein classical accelerated gradient methods do not apply, however all these applications are limited to the case of differentiable objective functions. The authors of [29] have derived an accelerated first-order method, which was inspired by the ellipsoid method. The proposed method is efficient; however, it suffers from the drawback that it requires an exact line search. An interesting framework is established in Flammarion and Bach [30], Su et al. [31], wherein the authors model the continuous-time limit of FGM as a second-order differential equation (ODE). Then, FGM equations can be obtained based on such a framework. Specifically, in Flammarion and Bach [30], the authors show that several accelerated schemes can be formulated as constant parameter ODE algorithms, wherein the stability of the systems would be equivalent to covergence at rate $\mathcal{O}(1/n^2)$. The limitation of the work is that the analysis presented therein is restricted only to the class of smooth and non-strongly convex problems. Moreover, in Su et al. [31] the authors show that the ODE type of analysis allows for a better understanding of Nesterov's scheme. However, the family of methods obtained therein, exhibits a similar convergence rate to FGM. Similar convergence rate as those obtained for FGM can also be derived by using theory from robust control [32]. A novel approach for analyzing the worst-case performance of first-order black-box methods has appeared in Drori and Teboulle [33]. The analysis conducted therein relies on the observation that the worst-case behavior improvement of a black-box method is itself an optimization problem, which is referred to as the performance estimation problem. By utilizing this approach, the authors of Kim and Fessler [34], 35] have introduced optimized first-order methods that are efficient and achieve a convergence bound that is two times smaller than the one attained by FGM. However, the development of these algorithms is restricted to solving problems with smooth objective functions.

Among the various approaches to the acceleration of first-order methods that were discussed above, the methods that were built based on the machinery of estimating sequences have attracted a lot of attention (see d'Aspremont et al. [18], Bubeck [36] and references therein). Several reasons that have led to their success are summarized in the sequel. First, on a theoretical level, FGM-type methods are proven to be optimal in the sense of Nemirovsky and Yudin [19]. Second, their practical performance is competitive even when they are used in conjunction with simple line search strategies, such as backtracking [37,38]. Third, they can be scaled to construct accelerated second-order methods [39,40] and accelerated higher-order methods [41,42]. Last, they have been shown to excel in performance even when they have been extended to other settings, such as distributed optimization [43,44], nonconvex optimization [45,46], stochastic optimization [47,48], non-Euclidean optimization [49,50], etc. In [51], it is argued that the key behind constructing optimal methods lies in the accumulation of some global information on the objective function. The mathematical objects which enable for capturing the relevant topological information on the function that is to be minimized are the estimating sequences. Typically, they consist of a pair of sequences, that simultaneously allow for parsing global information around the iterates, as well as for measuring the convergence rate of the minimization process. Despite their remarkable properties, estimating sequences exhibit the issue that there is no unique or systematic approach for constructing them. As we will see in the sequel, making the adequate choice of the estimating functions that comprise the estimating sequences can significantly impact the practical performance of the resulting algorithm.

The estimating sequences framework for the study and analysis of various methods has been presented in Baes [52]. An existing estimating sequence method that can directly solve (1) is the Accelerated Multistep Gradient Scheme (AMGS) [1]. The method is proven to enjoy the accelerated rate of convergence $\mathcal{O}(\frac{1}{k^2})$. Despite its notable theoretical and practical performance as measured by the number of iterations carried through until convergence, the method suffers the drawback that it requires two projection-like operations per iteration. This results in an increase of the computational burden, which (in the case of large-scale problems) is also reflected in an increase of the runtime of the method. This problem has been solved by the development of the Fast Iterative Shrinkage-Thresholding Algorithm (FISTA) [53]. The method also enjoys the accelerated convergence rate of $\mathcal{O}(\frac{1}{k^2})$, while at the same time requiring only one projection-like operation per iteration. Similarly to Nesterov [21], FISTA does not explicitly utilize the machinery of estimating sequences. However, as has been demonstrated in Florea and Vorobyov [54], by properly selecting the estimating functions it is possible to establish links between FISTA and estimating sequence methods.

As discussed above, many of the existing seminal methods such as AMGS, FISTA and FGM [51, Constant Step Scheme I (2.2.19)], were obtained by explicitly (or implicitly) using the estimating sequences framework, and they all enjoy the theoretical accelerated rate of convergence. Despite being accelerated in theory, these methods still exhibit the following differences: *i*) The algorithmic structure of the methods changes depending on the different estimating sequences that are used in devising these algorithms. *ii)* The practical performance of the methods varies significantly when they are tested on real-world problems and datasets. Moreover, based on preliminary experiments that we have conducted for the cases of differentiable convex functions, we have observed that FGM converges faster than both AMGS and FISTA. Thus, the question of how to construct newer classes of estimating sequences that can be used to build more efficient methods for solving problems with composite objective structure arises. In this work, we answer this question affirmatively, and show that, by constructing the appropriate estimating functions, it is possible to devise very efficient accelerated first-order methods. More specifically, the main contributions of the article are as follows.

- In this work, we extend the existing estimating sequences framework presented in Nesterov [51] for minimizing differentiable objective functions, to the broader class of solving problems with composite structure given in (1).
- We introduce a new structure for the estimating functions, which we call the *composite estimating functions*. The proposed estimating functions are constructed by utilizing the gradient mapping technique [19] together with a tighter global lower bound on the objective function than the one obtained from the Taylor series expansion of a convex function.

- We show that our proposed estimating functions can be used to efficiently parse information around all the iterates, as well as measure the convergence rate of the minimization process. Unlike the estimating functions devised in Nesterov [51], which are only defined for the problem of minimizing smooth functions, our proposed composite estimating functions make use of the tighter lower bound on the objective function, as well as the subgradients of the objective function. This allows for designing methods that are used for solving a broader class of problems.
- We show how the proposed estimating sequences can be used to produce a new class of Composite Objective Multi-step Estimating-sequence Techniques (COMET), which are also endowed with an efficient line-search strategy. Unlike AMGS, the resulting algorithms require only one projection-like operation per iteration.
- We prove that COMET enjoys the accelerated convergence rate even when the Lipschitz constant is not known and needs to be estimated.
- We establish that the initialization of COMET can be made robust to the imperfect knowledge of the strong convexity parameter. Such a fact is very important for many practical applications, as computing the true value of the strong convexity parameter is computationally expensive.
- Through extensive simulations for various typical signal processing problems with composite structure, we show that the proposed method yields a better performance than the existing benchmarks. Furthermore, we also show the robustness of the selected instances of COMET with respect to the imperfect knowledge of the strong convexity parameter and the Lipschitz constant. To demonstrate the robustness, as well as the reliability of our proposed method, we test its performance on real-world datasets.

The article is organized as follows. In Section 2, we introduce the key assumptions of the paper, as well as some of the main concepts that are used in developing our method. In Section 3, we introduce the proposed estimating sequences for composite objectives and devise COMET based on them. In Section 4, we formally establish the convergence of COMET and derive the convergence rate for the minimization process. Then, in Section 5, we illustrate the performance of our proposed method in solving several optimization problems and show that it outperforms the existing benchmarks. Last, in Section 6, we present our conclusions and discuss possible future research directions.

2. Preliminaries

Assume that the objective function is bounded below, i.e., (1) has a solution. Another key assumption, which holds true for typical signal processing applications, is that the function and gradient computations have approximately the same complexity. For the problem setting of interest, the necessary oracle functions are the function evaluators, $f(x)$, $g(x)$, gradient evaluator $\nabla f(x)$, and proximal evaluator $\text{prox}_{\tau g}(x)$.

To simplify our analysis, let us relocate the strong convexity of $g(x)$ within the objective function in (1). Let $x_0 \in \mathcal{R}^n$ and consider that

$$F(x) = \left(f(x) + \frac{\tau\mu_g}{2}||x - x_0||^2\right) + \tau\left(g(x) - \frac{\mu_g}{2}||x - x_0||^2\right) = \hat{f}(x) + \tau\hat{g}(x). \tag{3}$$

The resulting function $\hat{f}(x)$ has a Lipschitz constant $L_{\hat{f}} = L_f + \tau\mu_g$ and strong convexity parameter $\mu_{\hat{f}} = \mu_f + \tau\mu_g$. On the other hand, the function $\hat{g}(x)$ has a strong convexity parameter $\mu_{\hat{g}} = 0$.

Recall that it is possible to construct upper and lower bounds for the smooth and strongly convex function $\hat{f}(x)$ by using the following relations:

$$\hat{f}(x) \le \hat{f}(y) + \nabla\hat{f}(y)^T(x - y) + \frac{L_{\hat{f}}}{2}||y - x||^2, \tag{4}$$

$$\hat{f}(x) \ge \hat{f}(y) + \nabla\hat{f}(y)^T(x - y) + \frac{\mu_{\hat{f}}}{2}||y - x||^2, \tag{5}$$

for all points $y \in \mathcal{R}^n$. Similarly, we can construct the following lower bound for the non-smooth term

$$\hat{g}(x) \ge \hat{g}(y) + s(y)^T(x - y), \tag{6}$$

where $s(y)$ is a subgradient of the function $\hat{g}(\cdot)$ at the point y. Moreover, for all $y \in \mathcal{R}^n$ and $L \ge L_{\hat{f}}$, we define

$$m_L(y; x) \triangleq \hat{f}(y) + \nabla\hat{f}(y)^T(x - y) + \frac{L}{2}||x - y||^2 + \tau\hat{g}(x). \tag{7}$$

Using the upper bound on the function established in (4), it can be seen that

$$m_L(y; x) \ge F(x), \forall x, y \in \mathcal{R}^n. \tag{8}$$

At this point, the composite gradient mapping can be introduced as

$$T_L(y) \triangleq \arg\min_{x\in\mathcal{R}^n} m_L(y; x). \tag{9}$$

Lastly, the composite reduced gradient can be defined as

$$r_L(y) \triangleq L(y - T_L(y)). \tag{10}$$

Let us now make a digression and note that when $\tau = 0$, we have the following: *i)* $\hat{f}(x) = f(x)$, which follows from (3); *ii)* $T_L(y) = y - \frac{\nabla\hat{f}(y)}{L}$, which follows from (9) and (7). Substituting these results into the definition given in (10), yields $r_L(y) = \nabla F(y) = \nabla f(y)$, i.e., the composite reduced gradient becomes the gradient of the objective function.

Returning back to the more general case, i.e., $\tau \neq 0$, from the first-order optimality conditions for (9), we can write

$$\begin{aligned} &\nabla m_L(y; T_L(y))^T(x - T_L(y)) \ge 0, \\ &(\nabla f(y) + L(T_L(y) - y) + \tau s_L(y))^T(x - T_L(y)) \ge 0, \end{aligned} \tag{11}$$

where $s_L(y) \in \partial F(T_L(y))$ is a subgradient belonging to the subdifferential of $F(T_L(y))$, whose value depends on the point y. Equating the first bracket of (11) to 0, as well as recalling definition (10), we obtain the following relation, which is useful for computing the value of the composite reduced gradient

$$r_L(y) = L(y - T_L(y)) = \nabla f(y) + \tau s_L(y). \tag{12}$$

Last, we present a tighter lower bound on the objective function.

Theorem 1. *Let $F(x)$ be a composition of an $L_{\hat{f}}$-smooth and $\mu_{\hat{f}}$-strongly convex function $\hat{f}(x)$, and a simple convex function $\hat{g}(x)$, as given in (3). For $L \ge L_{\hat{f}}$, and $x, y \in \mathcal{R}^n$ we have*

$$F(x) \ge \hat{f}(T_L(y)) + \tau\hat{g}(T_L(y)) + r_L(y)^T(x - y) + \frac{\mu_{\hat{f}}}{2}||x - y||^2 + \frac{1}{2L}||r_L(y)||^2, \tag{13}$$

where $T_L(y)$ and $r_L(y)$ are defined in (9) and (10), respectively.

Proof. See Appendix A. □

3. COMET

In this section, we devise our proposed method. We start by introducing the composite estimating sequences, and then show why these sequences are useful. We also present a pair of composite estimating functions and show how to compute them recursively. Then, utilizing the proposed construction of the composite estimating functions, we derive COMET.

We begin by defining the composite estimating sequences.

Definition 1. The sequences $\{\phi_k\}_{k=0}^{\infty}$ and $\{\lambda_k\}_{k=0}^{\infty}$, $\lambda_k \geq 0$, are called composite estimating sequences of the function $F(\cdot)$ defined in (3), if $\lambda_k \to 0$ as $k \to \infty$, and $\forall x \in \mathcal{R}^n$, $\forall k \geq 0$ we have

$$\phi_k(x) \leq \lambda_k \phi_0(x) + (1-\lambda_k)F(x). \tag{14}$$

These composite estimating sequences allow for measuring the convergence rate to optimality, which is characterized in the following lemma.

Lemma 1. *If for some sequence of points $\{x_k\}_{k=0}^{\infty}$ we have $F(x_k) \leq \phi_k^* \triangleq \min_{x\in\mathcal{R}^n} \phi_k(x)$, then $F(x_k) - F(x^*) \leq \lambda_k[\phi_0(x^*) - F(x^*)]$, where $x^* = \arg\min_{x\in\mathcal{R}^n} F(x)$.*

Proof. See Appendix B. □

We are now ready to show how the composite estimating sequences can be defined recursively.

Lemma 2. *Assume that there exists a sequence $\{\alpha_k\}_{k=0}^{\infty}$, where $\alpha_k \in (0,1)$ $\forall k$, such that $\sum_{k=0}^{\infty}\alpha_k = \infty$, and an arbitrary sequence $\{y_k\}_{k=0}^{\infty}$. Furthermore, let $\lambda_0 = 1$ and assume that the estimates L_k of the Lipschitz constant $L_{\hat{f}}$ are selected in a way that inequality (4) is satisfied for all the iterates x_k and y_k. Then, the sequences $\{\phi_k\}_{k=0}^{\infty}$ and $\{\lambda_k\}_{k=0}^{\infty}$, which are defined recursively as*

$$\lambda_{k+1} = (1-\alpha_k)\lambda_k, \tag{15}$$

$$\begin{aligned}\phi_{k+1}(x) = {} & (1-\alpha_k)\phi_k(x) + \alpha_k\Big(F\big(T_{L_k}(y_k)\big) + \frac{1}{2L_k}\|r_{L_k}(y_k)\|^2\Big) \\ & + \alpha_k\Big(r_{L_k}(y_k)^T(x-y_k) + \frac{\mu_f}{2}\|x-y_k\|^2\Big),\end{aligned} \tag{16}$$

are composite estimating sequences.

Proof. See Appendix C. □

At this point, we provide a comparison between the results obtained in Lemmas 1 and 2 to their counterpart devised for the simpler case of minimizing smooth convex functions presented in Nesterov [51]. First, we can see from Lemma 1 that the convergence rate of the minimization process depends entirely on the rate at which $\lambda_k \to 0$. Moreover, the result hints that for problem (1) we should expect a similar convergence rate as in the simpler case of minimizing a differentiable convex function. Then, in Lemma 2, we have shown how to form the estimating functions. It can also be seen from (16) that we are utilizing a tighter lower bound than the one used for deriving FGM for the smooth strongly convex case.[1] Furthermore, it can be noted that the cost function is evaluated at specific points in its domain, which are produced by the composite gradient mapping. Last, it can be observed that the subgradient of the non-smooth objective function is needed to construct the estimating functions $\{\phi_k\}_{k=0}^{\infty}$.

Until now, no particular structure for the functions in the sequence $\{\phi_k\}_{k=0}^{\infty}$ has been proposed yet. Inspired by the analysis for FGM in the setup of smooth convex functions [51], in the sequel we let

$$\phi_k(x) \triangleq \phi_k^* + \frac{\gamma_k}{2}||x-v_k||^2, \ \forall k = 1,2,\ldots, \tag{17}$$

where $\gamma_k \in \mathcal{R}^+$ and $v_k \in \mathcal{R}^n$, $\forall k = 0,1,\ldots$. Nevertheless, we stress that this selection is not unique. As a matter of fact, different choices of the canonical structure for the function $\phi_k(x)$ can lead to entirely different algorithms, see for example [49,56,55]. Next, in Lemma 3 we show how the terms $\{\gamma_k\}_{k=0}^{\infty}$, $\{v_k\}_{k=0}^{\infty}$ and $\{\phi_k^*\}_{k=0}^{\infty}$ can be computed recursively.

Lemma 3. *Let $\phi_0(x) = \phi_0^* + \frac{\gamma_0}{2}||x-v_0||^2$, where $\gamma_0 \in \mathcal{R}^+$ and $v_0 \in \mathcal{R}^n$. Then, the process defined in Lemma 2 preserves the canonical form of the function $\{\phi_k(x)\}_{k=0}^{\infty}$ presented in (17), where the sequences $\{\gamma_k\}_{k=0}^{\infty}$, $\{v_k\}_{k=0}^{\infty}$ and $\{\phi_k^*\}_{k=0}^{\infty}$ can be computed as follows*

$$\gamma_{k+1} = (1-\alpha_k)\gamma_k + \alpha_k\mu_{\hat{f}}, \tag{18}$$

$$v_{k+1} = \frac{1}{\gamma_{k+1}}\Big((1-\alpha_k)\gamma_k v_k + \alpha_k\big(\mu_{\hat{f}}y_k - L_k\big(y_k - T_{L_k}(y_k)\big)\big)\Big), \tag{19}$$

$$\begin{aligned}\phi_{k+1}^* = {} & (1-\alpha_k)\phi_k^* + \alpha_k\Big(F\big(T_{L_k}(y_k)\big) + \frac{1}{2L_k}||r_{L_k}(y_k)||^2\Big) \\ & - \frac{L_k^2\alpha_k^2}{2\gamma_{k+1}}||y_k - T_{L_k}(y_k)||^2 + \frac{\mu_{\hat{f}}\alpha_k\gamma_k(1-\alpha_k)}{2\gamma_{k+1}}||y_k - v_k||^2 \\ & + \frac{L_k\alpha_k\gamma_k(1-\alpha_k)}{\gamma_{k+1}}(y_k - v_k)^T(y_k - T_{L_k}(x_k)).\end{aligned} \tag{20}$$

Proof. See Appendix D. □

Comparing the result obtained in Lemma 3 with its counterpart constructed for minimizing smooth objective functions [51, Lemma 2.2.3], it can be seen that the recursion for computing the elements in the sequences $\{v_k\}_{k=0}^{\infty}$ and $\{\phi_k^*\}_{k=0}^{\infty}$ has changed. It now reflects both the different lower bound on the objective function, as well as the reduced composite gradient, which were utilized for constructing the composite estimating functions.

Let us now proceed to constructing the algorithm via induction. First, let $\phi_0^* = F(x_0)$. Next, assume that for some iteration k, we have: $\phi_k^* \geq F(x_k)$. To conclude the induction argument, we need to show that $\phi_{k+1}^* \geq F(x_{k+1})$. Using the aforementioned assumption for iteration k into (20), it can be written that

$$\begin{aligned}\phi_{k+1}^* \geq {} & (1-\alpha_k)F(x_k) + \alpha_k\Big(F\big(T_{L_k}(y_k)\big) + \frac{1}{2L_k}||r_{L_k}(y_k)||^2\Big) \\ & - \frac{L_k^2\alpha_k^2}{2\gamma_{k+1}}||y_k - T_{L_k}(y_k)||^2 + \frac{\mu_{\hat{f}}\alpha_k\gamma_k(1-\alpha_k)}{2\gamma_{k+1}}||y_k - v_k||^2 \\ & + \frac{L_k\alpha_k\gamma_k(1-\alpha_k)}{\gamma_{k+1}}(v_k - y_k)^T(y_k - T_{L_k}(y_k)).\end{aligned} \tag{21}$$

Then, substituting the bound obtained in Theorem 1, as well as (10) into (21), we obtain

$$\begin{aligned}\phi_{k+1}^* \geq {} & (1-\alpha_k)\Big(F(T_{L_k}(y_k)) + r_{L_k}(y_k)^T(x_k - y_k) + \frac{\mu}{2}||x_k - y_k||^2 \\ & + \frac{1}{2L_k}||r_{L_k}(y_k)||^2\Big) + \alpha_k\Big(F\big(T_{L_k}(y_k)\big) + \frac{1}{2L_k}||r_{L_k}(y_k)||^2\Big) \\ & - \frac{\alpha_k^2}{2\gamma_{k+1}}||r_{L_k}(y_k)||^2 + \frac{\mu\alpha_k\gamma_k(1-\alpha_k}{2\gamma_{k+1}}||y_k - v_k||^2 \\ & + \frac{\alpha_k\gamma_k(1-\alpha_k)}{\gamma_{k+1}}r_{L_k}(y_k)^T(v_k - y_k).\end{aligned} \tag{22}$$

Making some algebraic manipulations and factoring in (23), we reach

[1] Recall that when $F(x)$ is smooth and convex function, the composite reduced gradient becomes just the gradient of the function.

$$\phi^*_{k+1} \geq F(T_{L_k}(y_k)) + \left(\frac{1}{2L_k} - \frac{\alpha_k^2}{2\gamma_{k+1}}\right)||r_{L_k}(y_k)||^2 + (1-\alpha_k) r_{L_k}(y_k)^T \left(x_k - y_k + \frac{\alpha_k \gamma_k}{\gamma_{k+1}}(v_k - y_k)\right). \tag{23}$$

At this point, a relation for the unknown terms in the sequences $\{\alpha_k\}_{k=0}^{\infty}$ and $\{y\}_{k=0}^{\infty}$ needs to be found. Observe that in (24) we can obtain the update rule for the terms in the sequence $\{\alpha_k\}_{k=0}^{\infty}$ as

$$\alpha_k = \sqrt{\frac{\gamma_{k+1}}{L_k}}. \tag{24}$$

Utilizing the recursion for γ_{k+1} given by (18), and solving the resulting quadratic equation yields

$$\alpha_k = \frac{\mu_{\hat{f}} - \gamma_k + \sqrt{\left(\mu_{\hat{f}} - \gamma_k\right)^2 + 4L_k\gamma_k}}{2L_k}. \tag{25}$$

Making the aforementioned selection for α_k, (24) can now be written as

$$\phi^*_{k+1} \geq F(T_{L_k}(y_k)) + (1-\alpha_k) r_{L_k}(y_k)^T \left(x_k - y_k + \frac{\alpha_k\gamma_k}{\gamma_{k+1}}(v_k - y_k)\right). \tag{26}$$

Thus, the update rule for the term y_k can be obtained by setting

$$x_k - y_k + \frac{\alpha_k\gamma_k}{\gamma_{k+1}}(v_k - y_k) = 0. \tag{27}$$

This results in

$$y_k = \frac{\gamma_{k+1}x_k + \alpha_k\gamma_k v_k}{\gamma_{k+1} + \alpha_k\gamma_k}. \tag{28}$$

To establish that $\phi_{k+1} \geq F(x_{k+1})$, it suffices to let $x_{k+1} = T_{L_k}(y_k)$.

Last, another major difference between our proposed method and its counterpart for minimizing differentiable convex functions [51], is the fact that our analysis allows for the line search adaptation.[2] The goal of our proposed line-search strategy is to select the smallest constant L_k such that (4) is satisfied $\forall k = 0, 1, \ldots$. To progress faster towards x^* in the initial iterations, we would want to initialize $L_0 \in]0, L_{\hat{f}}[$, and then gradually increase the value of the estimate of the Lipschitz constant across the iterations. However, since the true value of $L_{\hat{f}}$ is not known, this approach cannot be used. Therefore, it would be more preferable to select the line search strategy such that it ensures the robustness of the method with respect to the initialization of the estimate of the Lipschitz constant and ensure a dynamic update of the step size. Such a scheme would be of importance for many applications in signal processing (see Florea and Vorobyov [54] and the references therein). For this purpose, the following two parameters can be utilized: *i)* a constant $\eta_u > 1$, which increases the value of the estimate; *ii)* a constant $\eta_d \in]0, 1[$, which decreases the value of the estimate of the Lipschitz constant. Finally, the proposed method is summarized in Algorithm 1.

Comparing between our proposed method and FGM (Constant Step Scheme I in Nesterov [51]), we can observe from lines 6 and 7 in Algorithm 1, the similarities in updating the sequences $\{\alpha_k\}_{k=0}^{\infty}$ and $\{\gamma_k\}_{k=0}^{\infty}$. A difference can, however, be noticed in the update of the terms in the sequence $\{y_k\}_{k=0}^{\infty}$, whose value becomes independent of $\mu_{\hat{f}}$. Additionally, a key difference between the methods is in the update of the iterates x_k. Due to the composite structure of the objective function of interest, the next iterate x_{k+1} is computed by taking a proximal gradient step. Note that as long as the non-smooth term $g(x)$ has a simple structure, the proximal term can be computed efficiently. Another major difference between the methods lies in the update of the terms in the sequence $\{v_k\}_{k=0}^{\infty}$, which now reflect the usage of the proposed subgradient. Last, the parameter γ_0 can now be selected over a wider range of parameters than what is guaranteed by the existing convergence results for FGM established in Nesterov [51, Lemma 2.2.4]. The rationale behind this result will become clear in the sequel.

Algorithm 1 COMET.

Input: $x_0 \in \mathcal{R}^n$, $L_0 > 0$, $\mu_{\hat{f}}$, $\gamma_0 \in [0, 3L_0 + \mu_{\hat{f}}]$, $\eta_u > 1$ and $\eta_d \in]0, 1[$.

1: **while** $k \leq K_{\max}$ **do**
2: $\quad \hat{L}_i \leftarrow \eta_d L_k$
3: $\quad$ **while** True **do**
4: $\quad\quad \hat{\alpha}_i \leftarrow \frac{(\mu_{\hat{f}} - \gamma_k) + \sqrt{(\mu_{\hat{f}} - \gamma_k)^2 + 4\hat{L}_i\gamma_k}}{2\hat{L}_i}$
5: $\quad\quad \hat{\gamma}_{i+1} \leftarrow (1 - \hat{\alpha}_i)\gamma_k + \hat{\alpha}_i\mu_{\hat{f}}$
6: $\quad\quad \hat{y}_i \leftarrow \frac{\hat{\gamma}_{i+1}x_k + \hat{\alpha}_i\gamma_k v_k}{\hat{\gamma}_{i+1} + \hat{\alpha}_i\gamma_k}$
7: $\quad\quad \hat{x}_{i+1} \leftarrow \text{prox}_{\frac{1}{\hat{L}_i}\hat{g}}\left(\hat{y}_i - \frac{1}{\hat{L}_i}\nabla f(\hat{y}_i)\right)$
8: $\quad\quad \hat{v}_{i+1} \leftarrow \frac{1}{\hat{\gamma}_{i+1}}\left((1 - \hat{\alpha}_i)\gamma_k v_k + \hat{\alpha}_i\left(\mu_{\hat{f}}\hat{y}_i - \hat{L}_i(\hat{y}_i - \hat{x}_{i+1})\right)\right)$
9: $\quad\quad$ **if** $F(\hat{x}_{i+1}) \leq m_{\hat{L}_i}(\hat{y}_i, \hat{x}_{i+1})$ **then**
10: $\quad\quad\quad$ Break from loop
11: $\quad\quad$ **else**
12: $\quad\quad\quad \hat{L}_{i+1} \leftarrow \eta_u \hat{L}_i$
13: $\quad\quad$ **end if**
14: $\quad\quad i \leftarrow i + 1$
15: $\quad$ **end while**
16: $\quad L_{k+1} \leftarrow \hat{L}_i$, $x_{k+1} \leftarrow \hat{x}_i$, $\alpha_k \leftarrow \hat{\alpha}_{i-1}$, $y_k \leftarrow \hat{y}_{i-1}$, $i \leftarrow 0$, $k \leftarrow k + 1$
17: **end while**

Output: x_k

Before we proceed to analyzing the convergence rate of the minimization process, let us evaluate the behavior of the estimate of the Lipschitz constant. Depending on the initialization of L_0, there are two scenarios.

i) If $L_0 \in]0, L_{\hat{f}}[$, then from line 11 in Algorithm 1, it can be observed that the estimate of the Lipschitz constant at iteration k increases only if $L_{k-1} \leq L_{\hat{f}}$. Therefore, we can write

$$L_0 \leq \hat{L}_i \leq L_k \leq \eta_u L_{\hat{f}}. \tag{29}$$

ii) If $L_0 \geq L_{\hat{f}}$, then the condition in line 11 of Algorithm 1 is satisfied, and estimate of the Lipschitz constant cannot increase further. This yields

$$L_k \leq \eta_d L_0. \tag{30}$$

Combining the bounds (30) and (31), we can see that despite the initialization of L_0, it is always true that

$$L_k \leq L_{\max} \triangleq \max\{\eta_d L_0, \eta_u L_{\hat{f}}\}. \tag{31}$$

To obtain an easier understanding of the proposed method, we also present the flowchart in Fig. 1. As can be seen from the flowchart, at any iteration k the inputs are feed into the outer loop, which starts by decreasing the estimate of the Lipschitz constant (see line 2 in Algorithm 1). The inner loop then updates the parameters and takes *one* proximal gradient step to produce the iterate at iteration $k+1$ (see lines 4–8 in Algorithm 1). As long as a function-based stopping criterion is not satisfied, the inner loop also corrects the value of the estimate of the Lipschitz contant, which corresponds to line 12 in Algorithm 1. After the

[2] Note that several backtracking strategies have already been proposed in the literature (see for example Nesterov [1], Beck and Teboulle [53], Tseng [57]).

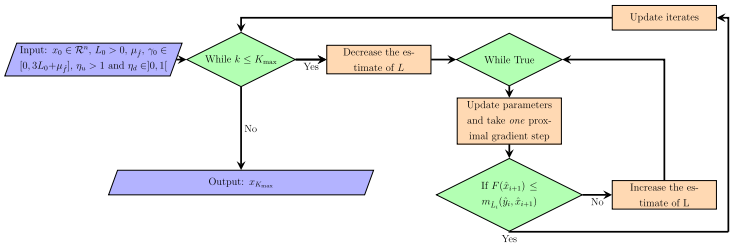


Fig. 1. Flowchart that depicts the main building blocks of our proposed method.

function-based stopping criterion is satisfied, the inner loop is terminated and the method proceeds to the next iterate (see line 16 in Algorithm 1). The numerical procedure terminates after the iteration-based stopping criterion is satisfied, and outputs $x_{K_{\max}}$. Contrasting our proposed COMET to AMGS and FISTA we can highlight several differences. First, with respect to AMGS, we note that the methods require different input parameters. Moreover, observe that our proposed COMET only queries one proximal and one gradient oracle to update the iterates. On the other hand, AMGS requires double the queries. As we will see in Section 5.3, this translates into an increase in the runtime of AMGS. Comparing our proposed COMET to FISTA, we note that they both query a single proximal and gradient oracle to update the iterates. The first difference in the methods lies in the line-search procedure that is employed by COMET, which is more efficient as it allows for dynamically updating the estimate of the Lipschitz constant. On the other hand, the line-search procedure proposed for FISTA only allows for increasing the estimate of the Lipschitz constant. Another major difference between the methods lies in the fact that the methods are initialized using different input parameters. Similar to the differences with AMGS, this arises because the methods were devised using different principles of acceleration of first-order methods.

4. Convergence analysis

Let us begin by noting that the result obtained in Lemma 1 suggests that the convergence rate of the minimization process will be the same as the rate at which $\lambda_k \to 0$. This is made more precise in the following theorem.

Theorem 2. *If we let $\lambda_0 = 1$ and $\lambda_k = \prod_{i=0}^{k-1}(1-\alpha_i)$, Algorithm 1 generates a sequence of points $\{x_k\}_{k=0}^{\infty}$ such that*

$$F(x_k) - F(x^*) \leq \lambda_k \left[F(x_0) - F(x^*) + \frac{\gamma_0}{2}||x_0 - x^*||^2 \right]. \tag{32}$$

Proof. See Appendix E. □

Now, recall that from Definition 1, we must have $\lambda_k \to 0$. Therefore, the result of Theorem 2 is sufficient to establish the fact that the sequence of iterates produced by our proposed algorithm converges to the optimal solution. The next step is to evaluate the rate of convergence of this process. Let us begin by characterizing the rate at which $\lambda_k \to 0$.

Lemma 4. *For all $k \geq 0$, Algorithm 1 guarantees that*

1. *If $\gamma_0 \in [0, \mu_{\hat{f}}[$, then*

$$\lambda_k \leq \frac{2\mu_{\hat{f}}}{L_k\left(e^{\frac{k+1}{2}\sqrt{\frac{\mu_{\hat{f}}}{L_k}}} - e^{-\frac{k+1}{2}\sqrt{\frac{\mu_{\hat{f}}}{L_k}}}\right)^2} \leq \frac{2}{(k+1)^2}. \tag{33}$$

2. *If $\gamma_0 \in [\mu_{\hat{f}}, 3L_0 + \mu_{\hat{f}}]$, then*

$$\lambda_k \leq \frac{4\mu_{\hat{f}}}{(\gamma_0 - \mu_{\hat{f}})\left(e^{\frac{k+1}{2}\sqrt{\frac{\mu_{\hat{f}}}{L_k}}} - e^{-\frac{k+1}{2}\sqrt{\frac{\mu_{\hat{f}}}{L_k}}}\right)^2} \leq \frac{4L_k}{(\gamma_0 - \mu_{\hat{f}})(k+1)^2}. \tag{34}$$

Proof. See Appendix F. □

Comparing the results obtained in Lemma 4 with the earlier results obtained in Nesterov [51, Lemma 2.2.4], we can see two major differences. First, our proposed analysis establishes the convergence of the method even when the true value of the Lipschitz constant is not known. Second, we can see that it is possible to establish the convergence of the method in minimizing objective functions with composite structure for a wider initialization range of the parameter γ_0. The importance of this result arises from the fact that the method exhibits a faster theoretical and practical convergence when $\gamma_0 = 0$, which is not supported by the existing analysis for FGM. At the same time, the initialization $\gamma_0 = 0$ also provides robustness with respect to the imperfect knowledge of $\mu_{\hat{f}}$.

From Theorem 2, we can see that the convergence rate of the minimization process depends on the distance $F(x_0) - F(x^*)$. The following lemma yields an upper bound on it.

Lemma 5. *Let $F(x)$ be a convex function with composite structure as shown in (1). Moreover, let $T_L(y)$ and $r_L(y)$ be computed as given in (9) and (12), respectively. Then, for any starting point x_0 in the domain of $F(x)$, we have*

$$F(x_0) - F(x^*) \leq \frac{L_0}{2}||x_0 - x^*||^2. \tag{35}$$

Proof. See Appendix G. □

Combining the results of Lemmas 4 and 5 with Theorem 2, we can immediately obtain the convergence rate for COMET as follows.

Theorem 3. *Algorithm 1 generates a sequence of points such that*

1. *If $\gamma_0 \in [0, \mu_{\hat{f}}[$, then*

$$F(x_k) - F(x^*) \leq \frac{\mu_{\hat{f}}(L_0 + \gamma_0)||x_0 - x^*||^2}{L_k\left(e^{\frac{k+1}{2}\sqrt{\frac{\mu_{\hat{f}}}{L_k}}} - e^{-\frac{k+1}{2}\sqrt{\frac{\mu_{\hat{f}}}{L_k}}}\right)^2} \tag{36}$$

2. *If $\gamma_0 \in [\mu_{\hat{f}}, 3L_0 + \mu_{\hat{f}}]$, then*

$$F(x_k) - F(x^*) \leq \frac{2\mu_{\hat{f}}(L_0 + \gamma_0)||x_0 - x^*||^2}{(\gamma_0 - \mu_{\hat{f}})\left(e^{\frac{k+1}{2}\sqrt{\frac{\mu}{L_k}}} - e^{-\frac{k+1}{2}\sqrt{\frac{\mu}{L_k}}}\right)^2} \tag{37}$$

From the result of Theorem 3 we can see that our proposed method is guaranteed to converge over a wider interval than its counterpart designed for minimizing smooth and strongly convex objectives. Notice that initializing $\gamma_0 = 0$ would guarantee the fastest convergence of the method. Such a result is important when considering many practical applications, wherein the true values of $\mu_{\hat{f}}$ and $L_{\hat{f}}$ are often not known and should be estimated. Another factor that impacts the rate of convergence of the minimization process is also the initialization of L_0. From (37), (38) we can see that the smaller the value of L_0, the faster the convergence of the method.

5. Numerical study

In this section, we compare the numerical performance of the proposed method against the two seminal black-box methods, namely, AMGS and FISTA, in solving several optimization problems, which arise often in many signal and image processing, statistics and data science applications. The selected loss functions are the quadratic and logistic loss functions, both with elastic net regularization. Moreover, we also test the performance of our proposed COMET in solving the regularized image deblurring problem. As we will see in the sequel, controlling the parameters of the elastic net regularizer allows for simulating extremely ill-conditioned examples. For the constructed examples, we show that COMET outperforms the selected benchmarks in terms of minimizing the number of iterations needed to achieve a certain tolerance level. To provide reliable results, we utilize both synthetic and real data, that are selected from the Library for Support Vector Machines [58]. To find the optimal solutions, we use CVX [59].

In the first example, we illustrate the performance of three variants of COMET: *1)* we consider the variant that in theory is expected to result in the fastest convergence, which is obtained when we initialize for $\gamma_0 = 0$, and it is referred to as "COMET, variant 1"; *2)* we also consider the variant that is expected to produce the slowest convergence, which happens when we initialize $\gamma_0 = 3L_0 + \mu_{\hat{f}}$, and it is labeled as "COMET, variant 2"; *3)* we also implement the variant of COMET that is obtained when $\gamma_0 = \mu_{\hat{f}}$, which is referred to as "COMET, variant 3". When comparing the performance of the methods under the condition where the Lipschitz constant is not known, for both AMGS and FISTA we utilize the line-search strategies presented in the respective works [1,53]. We note that throughout all the simulations the starting point x_0 is randomly selected and all algorithms are initialized in it. The numerical experiments are conducted using an Intel(R) Core(TM) i7-8665U 1.90 GHz CPU and the methods are implemented using Matlab.

5.1. Minimizing the quadratic loss function

Consider one of the most popular problems in signal processing and statistics

$$\underset{x\in\mathcal{R}^n}{\text{minimize}} \quad \frac{1}{2}\sum_{i=1}^{m}\left(a_i^T x - y_i\right)^2 + \frac{\tau_1}{2}\|x\|^2 + \tau_2\|x\|_1, \tag{38}$$

where $||\cdot||_1$ denotes the l_1 norm. The objective is to show that the theoretical gains of COMET, which are discussed in Section 4, are also reflected in the practical performance of the methods. Moreover, we analyze how the performance of the methods scales with the condition number of the problem. We also illustrate the practical benefits of utilizing the proposed line-search strategy.

Let us first consider the simplest case, where the Lipschitz constant is assumed to be known. It allows for an objective assessment of the effectiveness of the methods in finding the optimal solution. For this example, we utilize synthetic data. We consider the diagonal matrix $A \in \mathcal{R}^{m\times m}$ and sample the elements a_{ii} from the discrete set $\{10^0, 10^{-1}, 10^{-2}, \ldots, 10^{-\xi}\}$ uniformly at random. This choice of selecting A ensures that $L = 1$ and $\mu_f = 10^{-\xi}$, which results in the condition number 10^{ξ}. Then, we select the elements of the vector $y \in \mathcal{R}^m$ by uniformly drawing them from the box $[0, 1]^n$. Lastly, we note that in our computational experiments we set $m \in \{500, 1000, 1500, 2000\}$, $\xi \in \{3, 4, 7, 8\}$ and $\tau_1 = \tau_2 \in \{10^{-3}, 10^{-4}, 10^{-7}, 10^{-8}\}$.

From Fig. 2, we can observe that the proposed method significantly outperforms all the existing benchmarks. First, notice that the larger the condition number of the problems becomes, the more iterations, and consequently computations, are required by the methods to obtain a good solution. Comparing between the methods, we can observe that all instances of COMET yield a better quality of the obtained solution, as measured by the distance to x^*. Moreover, we can clearly see that the iterates produced by COMET converge to x^* in a much smaller number of iterations. Another important observation that can be made from the figure is that the proposed method exhibits better monotonic properties than both AMGS and FISTA. Comparing the performance of different variants of COMET, we can observe that their behavior is similar and the differences in performance are not too large. We can see that the variant that yields the best performance is the one obtained when $\gamma_0 = 0$, which is coherent with the theoretical results established in Section 4.

Next, we proceed to analyzing a more realistic scenario. We assume that the Lipschitz constant is not known, and needs to be estimated by using a line-search procedure. To demonstrate the robustness of the line-search strategy to be utilized in conjunction with COMET, we consider the following cases. *i)* The Lipschitz constant is underestimated by a factor of 10, i.e., $L_0 = 0.1L_{\hat{f}}$. *ii)* The Lipschitz constant is overestimated by a factor of 10, i.e., $L_0 = 10L_{\hat{f}}$. Moreover, we note that we selected $\eta_u = 2$ and $\eta_d = 0.9$, which were suggested in Becker et al. [60] because they ensure a good performance of the methods in many applications. Another parameter that is computationally expensive to be estimated in practice is the strong convexity parameter $\mu_{\hat{f}}$. To avoid an increase in computations, in all the following simulations we equate the value of the strong convexity parameter to that of the regularization term in the objective function in (41). Lastly, we note that for all the examples that will be shown in the sequel, we utilize the datasets "a1a" and "colon-cancer". The former dataset has data matrix $A \in \mathcal{R}^{1605\times 123}$, whereas the latter has $A \in \mathcal{R}^{62\times 2000}$.

For the datasets that we are utilizing, the respective Lipschitz constants are $L_{\text{"a1a'prime}} = 10061$ and $L_{\text{"colon-cancer'prime}} = 1927.4$. Moreover, we let the regularizer term $\tau_1 = \tau_2 \in \{10^{-5}, 10^{-6}\}$. Evidently, this selection of the regularizer terms guarantees a very large condition number $\kappa = \frac{L_{\hat{f}}}{\mu_{\hat{f}}}$ for the problems that are being solved. The numerical results are presented in Fig. 3, from which we can observe that all the instances of COMET significantly outperform the existing benchmarks. First, the final iterate produced by the first variant of COMET is the closest to x^*. This is most visible from the numerical experiments conducted on the "a1a" dataset, which are depicted in Fig. 3(a) and (b). Second, the iterates produced by the proposed COMET converge to x^* by requiring a significantly smaller number of iterations, when compared to AMGS and FISTA. Third, the performance of FISTA largely depends on the initialization of the Lipschitz constant. On the other hand, we can observe that for both datasets, the performance of both AMGS and COMET remains unaffected by the value of L_0. We stress that COMET retains the robustness to L_0 at the lower computational cost of only one projection-like operation per iteration, whereas AMGS requires double of that. Last, comparing the performance between the selected variants of COMET, we can see that in practice their performance differences are minor. Neverthe-

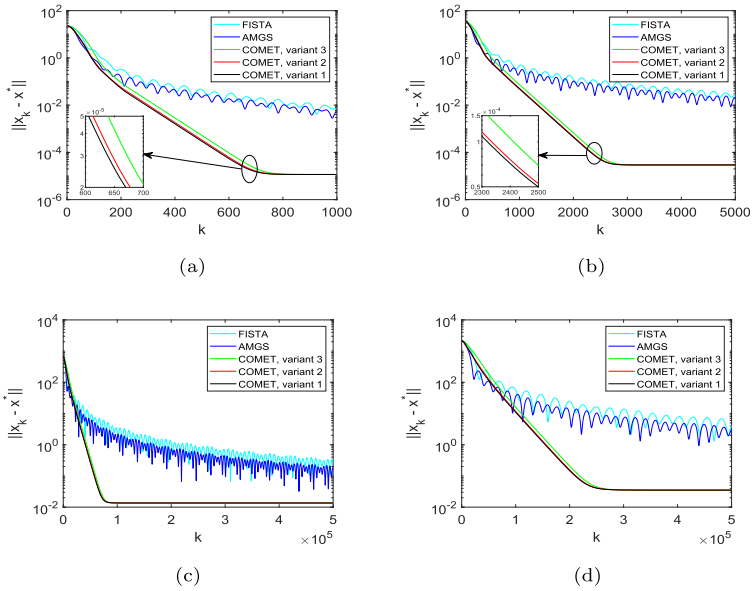


Fig. 2. Comparison between the efficiency of the algorithms tested in minimizing the quadratic loss function with elastic net regularizer on randomly generated data.

less, our results shown in Fig. 3(a) and (b) suggest that the version of COMET which is obtained when $\gamma_0 = 0$ yields a better performance. This becomes important particularly when considering practical applications, wherein the true values of $\mu_{\hat{f}}$ and $L_{\hat{f}}$ are typically not known and their true values can only be estimated within some error bounds. From this perspective, we can conclude that the instance of COMET obtained by setting $\gamma_0 = 0$ enjoys both the faster convergence of the iterates and the robustness with respect to the imperfect knowledge of $\mu_{\hat{f}}$ and $L_{\hat{f}}$.

5.2. Minimizing the logistic loss function

To demonstrate the versatility of the proposed black-box method, let us now compare its performance to the selected benchmarks in minimizing a regularized logistic loss function with elastic net regularizer

$$\underset{x\in\mathcal{R}^n}{\text{minimize}} \quad \frac{1}{m}\sum_{i=1}^{m}\log\left(1+\mathrm{e}^{-b_i x a_i}\right)+\frac{\tau_1}{2}\|x\|^2+\tau_2\|x\|_1. \tag{39}$$

For this problem type, we diversify the utilized datasets and select "triazine", as well as a subset of "rcv1.binary". For the chosen datasets, we have $A_{\text{//triazine'prime}} \in \mathcal{R}^{186\times 61}$ and $A_{\text{//rcv1.binary'prime}} \in \mathcal{R}^{1000\times 2000}$. Moreover, from the results of Fig. 3, we have observed that the performance of FISTA has been dependent on the initial estimate of the Lipschitz constant and has been overall worsened when $L_{\hat{f}}$ is unknown. Therefore, to provide the fairest comparison with respect to FISTA, for this set of examples we estimate the value of L directly from the data. More specifically, we have $L_{\text{//triazine'prime}} = 25.15$ and $L_{\text{//rcv1.binary'prime}} = 1.13$. On the other hand, similar to the earlier computational experiments, we equate the value of the strong convexity parameter to that of the regularization term in the objective function in (40). Last, we note that for this set of numerical experiments we consider the cases when $\tau_1 \neq \tau_2$. The results are reported in Fig. 4, wherein the specific values for τ_1 and τ_2 are also presented.

From Fig. 4, we can observe that for both datasets, COMET outperforms and exhibits better monotonic properties than AMGS or FISTA. Moreover, all variants of COMET require a much lower number of iterations to produce iterates which are closest to x^*. Last, for the selected problem type, the variant of COMET which is constructed when $\gamma_0 = 0$ yields the best practical performance, although the true value of $\mu_{\hat{f}}$ is not known.

5.3. Application to the regularized image deblurring problem

Let us now consider solving the problem of regularized image deblurring, which we formulate as follows

$$\underset{x\in\mathcal{R}^n}{\text{minimize}} \quad \|RWx-y\|^2+\frac{\tau_1}{2}\|x\|^2+\tau_2\|x\|_1, \tag{40}$$

where R represents the blur operator and W is the inverse three-stage Haar wavelet transform. In this example, $x \in \mathcal{R}^{256\times 256}$ is the cameraman test image [53]. To blurr the image, we scale its pixels in the range [0,1], add zero-mean Gaussian noise with standard deviation 10^{-3} and apply the blur operator R. Moreover, we set the regularizer parameters $\tau_1 = 1\times\ 10^{-3}$ and $\tau_2 = 10^{-5}$. For this problem, we initialize $L_0 = L_F$, which is obtained as the maximum eigenvalue of $(RW)^T(RW)$, and set $\mu_F = \tau_1$. Different from the previous sections, herein we report the CPU runtime (in seconds) that was needed to decrease the value of the objective function. For a more extensive comparison, herein we have also included the Accelerated Composite Gradient Method (ACGM) [37], which is built on top of the estimating sequences variant that was used for designing AMGS. Moreover, we have also included the variant of FISTA presented in Chambolle and Pock [61], which is designed to exploit the strong convexity information that might be available about the objective function.

Our findings are summarized in Table 1. The first column was obtained by computing the values of the objective function that

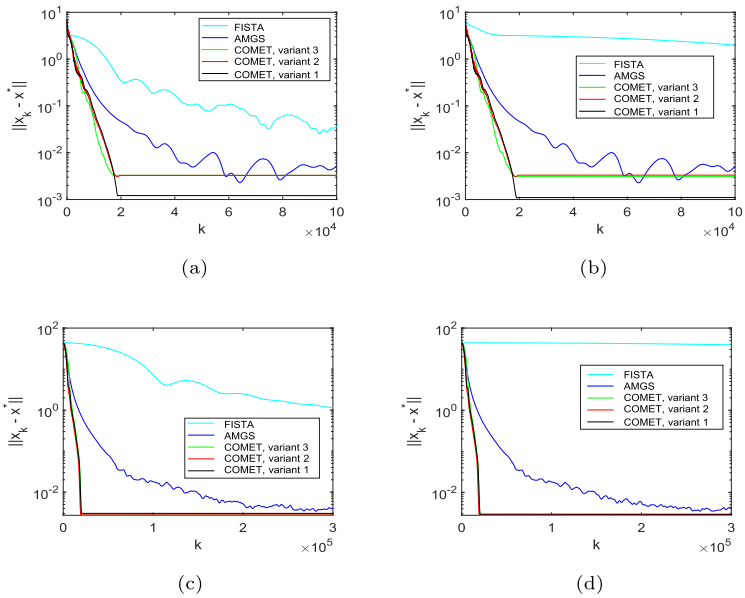


Fig. 3. Comparison between the efficiency and robustness with respect to the initialization of the Lipschitz constant of the algorithms tested in minimizing the quadratic loss function with elastic net regularizer on real data.

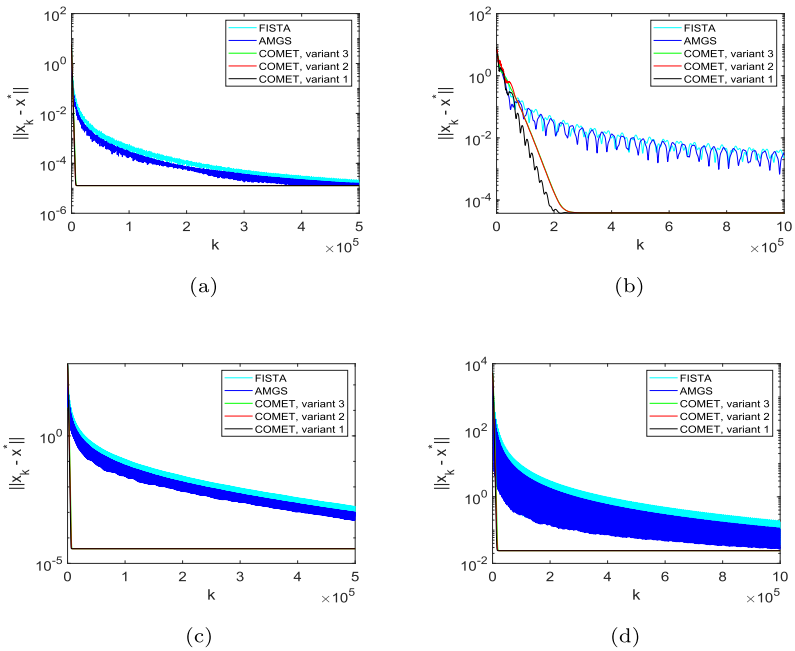


Fig. 4. Comparison between the efficiency of the algorithms tested in minimizing the logistic loss function with elastic net regularizer on real data.

Table 1
Comparison between the CPU runtimes (in seconds) of the algorithms tested in solving the image deblurring problem.

F(x)	COMET, variant 1	COMET, variant 2	COMET, variant 3	AMGS	ACGM	FISTA CP	FISTA
45.74	1.33	1.21	1.87	2.52	1.76	1.92	2.16
25.61	2.77	2.35	3.14	3.98	3.45	3.57	3.67
13.22	4.19	3.78	4.52	6.21	4.93	5.23	5.84
5.83	5.49	4.98	6.02	9.42	6.76	7.38	7.69
3.25	6.97	5.89	7.32	13.21	8.35	9.21	9.84
1.11	8.29	7.82	8.75	17.65	10.79	12.41	12.73
0.63	9.72	9.46	10.06	22.08	13.24	15.86	16.25
0.51	11.14	11.31	12.69	26.39	15.65	17.13	17.97
0.44	13.53	13.93	14.21	34.11	17.23	19.32	20.15
0.37	15.86	16.56	16.72	41.28	19.86	23.57	24.43
0.35	17.30	18.27	18.96	49.36	25.57	28.39	32.07

were obtained by running the first variant of COMET in intervals of 20 iterations. The other entries in the table were obtained by computing the time spent by the other methods to achieve the same decrease in the values of the objective function. Analyzing the obtained results, we can observe that the different variants of the estimating sequences methods are very efficient. Different from the other estimating sequence methods, we can see that the performance of AMGS is significantly affected by the need to compute an additional proximal step per iteration. Comparing to FISTA, every variant of COMET and ACGM perform more computations per iteration. Nevertheless, we can see that the improvement in runtime is significant. Comparing among the estimating sequence methods, we can observe that the fastest variant of COMET converges approximately 30% faster than AMGS. Last, we note that the differences in runtime among all variants of COMET are marginal. Nevertheless, we note that the variant of COMET which is obtained by initializing $\gamma_0 = 0$ is more efficient, while also enjoying the robustness to the imperfect knowledge of the strong convexity parameter.

6. Conclusions and discussion

The problem of constructing accelerated black-box first-order methods for solving optimization problems with composite structure by utilizing the estimating sequences framework has been considered, and a new class of estimating functions has been introduced. It has been shown that by exploiting these estimating sequences together with the gradient mapping technique, it is possible to construct very efficient gradient-based methods, which we named COMET. Unlike the existing results on the convergence of FGM-type methods, the novel convergence analysis established in this work allows for the adaptation of the step-size. Another major contribution which stemmed from the proposed convergence analysis is the fact that COMET is guaranteed to converge when $\gamma_0 \in [0, 3L + \mu_{\hat{f}}]$. The practical implication of these two observations is the fact that it is possible to construct efficient accelerated methods which are also robust to the imperfect knowledge of the smoothness and strong convexity parameters. Our theoretical findings were corroborated by extensive numerical experiments, wherein both synthetic and real-world data were utilized.

The results that were established in this work can be further developed in different directions. Particularly, it is interesting to investigate the possibilities of embedding the heavy-ball momentum into COMET. Another attractive research direction is the investigation of the possibility of coupling between the proposed framework and the inexact oracle framework, as well as the framework for constructing distributed proximal gradient methods. Lastly, we note that it is also interesting to investigate the possible extensions to designing accelerated algorithms for solving non-convex optimization problems.

Declaration of Competing Interest

Authors declare that they have no conflict of interest.

CRediT authorship contribution statement

Endrit Dosti: Conceptualization, Formal analysis, Methodology, Software, Visualization, Writing – review & editing. **Sergiy A. Vorobyov:** Conceptualization, Formal analysis, Methodology, Supervision, Writing – review & editing. **Themistoklis Charalambous:** Conceptualization, Methodology, Supervision, Writing – review & editing.

Data Availability

Data will be made available on request.

Appendix A. Proof of Theorem 1

We start by showing that $m_L(y; x)$ is an L-strongly convex function in x. Notice that it is defined to be the sum of convex functions. Therefore, it is itself a convex function. Now, consider that

$$m_L(y; y) - m_L(y; T_L(y)) \geq \frac{L}{2}||y - T_L(y)||^2. \tag{41}$$

By the definition given in (9), $T_L(y)$ is the minimizer of $m_L(y; x)$ over all $x \in \mathcal{R}^n$. Therefore, we can conclude that $m_L(y; x)$ is a strongly convex function with strong convexity parameter L.

Now, we can proceed to deriving the lower bound. From (5), (6), it can be written that

$$F(x) \geq \hat{f}(y) + \tau\hat{g}(y) + \left(\nabla\hat{f}(y) + \tau s_L(y)\right)^T (x - y) + \frac{\mu_{\hat{f}}}{2}||x - y||^2. \tag{42}$$

Then, from the definition of $m_L(y, y)$ given in (7), as well as (12), the right-hand side (RHS) of (43) can be rewritten as

$$\begin{aligned}&\hat{f}(y) + \tau\hat{g}(y) + \left(\nabla\hat{f}(y) + \tau s_L(y)\right)^T (x - y) + \frac{\mu_{\hat{f}}}{2}\|x - y\|^2 \\ &= m_L(y; y) + r_L(y)^T (x - y) + \frac{\mu_{\hat{f}}}{2}\|x - y\|^2. \end{aligned}\tag{43}$$

Moreover, substituting (42) in (44), the lower bound of the RHS of (44) becomes

$$\begin{aligned}m_L(y; y) + r_L(y)^T (x - y) + \frac{\mu_{\hat{f}}}{2}\|x - y\|^2 \geq m_L(y; T_L(y)) \\ + \frac{L}{2}\|y - T_L(y)\|^2 + r_L(y)^T (x - y) + \frac{\mu_{\hat{f}}}{2}\|x - y\|^2.\end{aligned}$$

Utilizing the definition of the reduced composite gradient given in (10), yields

$$m_L(y;T_L(y))+\frac{L}{2}\|y-T_L(y)\|^2+r_L(y)^T(x-y)+\frac{\mu_{\hat{f}}}{2}\|x-y\|^2$$
$$=m_L(y;T_L(y))+\frac{1}{2L}\|r_L(y)\|^2+r_L(y)^T(x-y)+\frac{\mu_{\hat{f}}}{2}\|x-y\|^2. \quad (44)$$

Finally, taking a proximal gradient descent step on $f(x)$, which by assumption has Lipschitz continuous gradient, we can obtain (13). This completes the proof.

Appendix B. Proof of Lemma 1

By the assumption of Lemma 1, we have

$$F(x_k) \le \phi_k^* = \min_{x\in\mathcal{R}^n}\phi_k(x) \overset{(14)}{\le} \min_{x\in\mathcal{R}^n}[\lambda_k\phi_0(x)+(1-\lambda_k)F(x)]$$
$$\le \lambda_k\phi_0(x^*)+(1-\lambda_k)F(x^*).$$

Rearranging the terms yields the desired result.

Appendix C. Proof of Lemma 2

We prove this lemma by induction. Let us begin by analyzing iteration $k=0$. By assumption, we have $\lambda_0=1$. Utilizing (14), we obtain $\phi_0(x)\le\lambda_0\phi_0(x)+(1-\lambda_0)F(x)\equiv\phi_0(x)$. Then, assuming that (14) holds true at some iteration k, it can be written that

$$\phi_k(x)-(1-\lambda_k)F(x)\le\lambda_k\phi_0(x). \quad (45)$$

Substituting the bound obtained in Theorem 1, i.e., (13) in (16), we obtain

$$\phi_{k+1}(x)\le(1-\alpha_k)\phi_k(x)+\alpha_kF(x). \quad (46)$$

Then, adding and subtracting the same term to the RHS of (47), we reach

$$\phi_{k+1}(x) \le (1-\alpha_k)\phi_k(x)+\alpha_kF(x)+(1-\alpha_k)(1-\lambda_k)F(x)-(1-\alpha_k)$$
$$(1-\lambda_k)F(x)=(1-\alpha_k)[\phi_k(x)-(1-\lambda_k)F(x)]$$
$$+(\alpha_k+(1-\lambda_k)(1-\alpha_k))F(x). \quad (47)$$

Using the bound obtained in (46) in (48), we have

$$\phi_{k+1}(x)\le(1-\alpha_k)\lambda_k\phi_0(x)+(1-\lambda_k+\alpha_k\lambda_k)F(x). \quad (48)$$

Lastly, after utilizing (15), the proof is concluded.

Appendix D. Proof of Lemma 3

Let us begin with establishing the first part of the proof through a mathematical induction argument. At iteration $k=0$, we have $\nabla^2\phi_0(x)=\gamma_0 I$. Next, assuming that at some iteration k it is true that $\nabla^2\phi_k(x)=\gamma_k I$, at iteration $k+1$ it can be written that

$$\nabla^2\phi_{k+1}(x)\overset{(16)}{=}(1-\alpha_k)\gamma_k I+\alpha_k\mu_{\hat{f}}I\equiv\gamma_{k+1}I. \quad (49)$$

We then proceed to establishing the proposed recurrent relations for updating the terms in the sequences $\{v_k\}_{k=0}^{\infty}$ and $\{\phi_k^*\}_{k=0}^{\infty}$. Substituting (17) into (16), and analyzing its first-order optimality conditions we obtain

$$\gamma_{k+1}(x-v_{k+1})=\gamma_k(1-\alpha_k)(x-v_k)+\alpha_k\big(\mu_{\hat{f}}(x-y_k)r_{L_k}(y_k)\big). \quad (50)$$

We can then reduce the terms that depend on x by using (18) in (51), and reach

$$-\gamma_{k+1}v_{k+1} \quad =-(1-\alpha_k)\gamma_kv_k+\alpha_k\big(-\mu_{\hat{f}}y_k+r_{L_k}(y_k)\big). \quad (51)$$

Then, substituting (10) in (52), we obtain (19).

To establish (20), let us begin by substituting (17) in (16), now evaluated at the point $x=y_k$. This way we obtain

$$\phi_{k+1}^* \ +\frac{\gamma_{k+1}}{2}\|y_k-v_{k+1}\|^2=(1-\alpha_k)\Big(\phi_k^*+\frac{\gamma_k}{2}\|y_k-v_k\|^2\Big)$$
$$+\alpha_k\Big(F\big(T_{L_k}(y_k)\big)+\frac{1}{2L_k}\|r_{L_k}(y_k)\|^2\Big). \quad (52)$$

We proceed by utilizing (19) to compute the second term in the left hand side (LHS) of (53). Consider the following

$$v_{k+1}-y_k=\frac{1}{\gamma_{k+1}}((1-\alpha_k)\gamma_kv_k+\alpha_k\mu_{\hat{f}}y_k$$
$$-\alpha_kL_k\big(y_k-T_{L_k}(y_k)\big)-\gamma_{k+1}y_k). \quad (53)$$

Then, utilizing (18) in (54), we obtain

$$v_{k+1}-y_k=\frac{1}{\gamma_{k+1}}((1-\alpha_k)\gamma_k(v_k-y_k)-\alpha_kL_k\big(y_k-T_{L_k}(y_k)\big). \quad (54)$$

Taking $||\cdot||^2$ of both sides in (55), yields

$$||y_k-v_{k+1}||^2=\frac{||(1-\alpha_k)\gamma_k(v_k-y_k)-\alpha_kL_k\big(y_k-T_{L_k}(y_k)\big)||^2}{\gamma_{k+1}^2}. \quad (55)$$

Finally, multiplying both sides of (56) by $\frac{\gamma_{k+1}}{2}$ and expanding the RHS, we reach

$$\frac{\gamma_{k+1}}{2}\|y_k-v_{k+1}\|^2=\frac{(1-\alpha_k)^2\gamma_k^2}{2\gamma_{k+1}}\|v_k-y_k\|^2+\frac{\alpha_k^2{L_k}^2}{2\gamma_{k+1}}\|y_k-T_{L_k}(y_k)\|^2$$
$$-\frac{2L_k\alpha_k(1-\alpha_k)\gamma_k}{2\gamma_{k+1}}(v_k-y_k)^T\nabla\big(y_k-T_{L_k}(y_k)\big). \quad (56)$$

Substituting (57) in (53), and making some straightforward algebraic manipulations, we obtain (20).

Appendix E. Proof of Theorem 2

Set $\phi_0^*=f(x_0)$. Then, considering (17) evaluated at iteration $k=0$ and $x=x_0$, we obtain $\phi_0(x_0)=f(x_0)+\frac{\gamma_0}{2}||x_0-v_0||^2$. In Algorithm 1, we initialize $v_0=x_0$, which is sufficient to guarantee that $f(x_0)\le\phi_0^*$ at step $k=0$. Moreover, recall that we designed the update rules of the proposed method to guarantee that $f(x_k)\le\phi_k^*$, $\forall k=1,2,\ldots$. Therefore, the necessary conditions for the results proved in Lemma 1 to be applied are satisfied.

Appendix F. Proof of Lemma 4

Let $\gamma_0\in[0,3L_0+\mu_{\hat{f}}]$ and consider applying (18) to the following

$$\gamma_{k+1}-\mu_{\hat{f}}=(1-\alpha_k)\gamma_k+\alpha_k\mu_{\hat{f}}-\mu_{\hat{f}}. \quad (57)$$

Then, utilizing the assumption that $\lambda_0=1$ in (58), it can be written that

$$\gamma_{k+1}-\mu_{\hat{f}}=(1-\alpha_k)\lambda_0\big[\gamma_k-\mu_{\hat{f}}\big]. \quad (58)$$

Using the recursivity of (18) in (59), yields

$$\gamma_{k+1}-\mu_{\hat{f}}=\lambda_{k+1}\big[\gamma_0-\mu_{\hat{f}}\big]. \quad (59)$$

Let us now exploit the connection between relations (15) and (25), which can be linked through the term α_k as follows

$$\alpha_k=1-\frac{\lambda_{k+1}}{\lambda_k}=\sqrt{\frac{\gamma_{k+1}}{L_k}}=\sqrt{\frac{\mu_{\hat{f}}}{L_k}+\frac{\gamma_{k+1}-\mu_{\hat{f}}}{L_k}}. \quad (60)$$

Substituting (60) in the RHS of (61) and making some manipulations, we get

$$\frac{1}{\lambda_{k+1}}-\frac{1}{\lambda_k}=\frac{1}{\sqrt{\lambda_{k+1}}}\sqrt{\frac{\mu_{\hat{f}}}{\lambda_{k+1}L_k}+\frac{\gamma_0-\mu_{\hat{f}}}{L_k}}. \quad (61)$$

Then, through a difference of squares argument, we reach

$$\left(\frac{1}{\sqrt{\lambda_{k+1}}}-\frac{1}{\sqrt{\lambda_k}}\right)\left(\frac{1}{\sqrt{\lambda_{k+1}}}+\frac{1}{\sqrt{\lambda_k}}\right) = \frac{1}{\sqrt{\lambda_{k+1}}}\sqrt{\frac{\mu_{\hat{f}}}{\lambda_{k+1}L_k}+\frac{\gamma_0-\mu_{\hat{f}}}{L_k}}. \tag{62}$$

Let us now analyze the behavior of the terms in the sequence $\{\lambda_k\}_{k=0}^{\infty}$. First, recall that from Lemma 2 we have $\alpha_k \in [0,1]$. Then, considering (15), we can conclude that the terms λ_k are non-increasing in the iteration counter k. Therefore, we can substitute the term $\frac{1}{\sqrt{\lambda_k}}$ in the LHS of (63) with the larger number $\frac{1}{\sqrt{\lambda_{k+1}}}$. This results in

$$\frac{2}{\sqrt{\lambda_{k+1}}}\left(\frac{1}{\sqrt{\lambda_{k+1}}}-\frac{1}{\sqrt{\lambda_k}}\right) \geq \frac{1}{\sqrt{\lambda_{k+1}}}\sqrt{\frac{\mu_{\hat{f}}}{\lambda_{k+1}L_k}+\frac{\gamma_0-\mu_{\hat{f}}}{L_k}} \tag{63}$$

Note that the practical performance of the proposed method depends on the initialization of the parameter γ_0. To allow for the widest possible range of selection for this parameter, we need to consider separately the regions $\mathcal{R}_1 = [0, \mu_{\hat{f}}[$ and $\mathcal{R}_2 = [\mu_{\hat{f}}, 3L_k + \mu_{\hat{f}}]$. The results for the case when $\gamma_0 \in \mathcal{R}_2$ can be established by following the analysis conducted for FGM in Nesterov [51, Lemma 2.2.4]. Therefore, in the sequel we will thoroughly establish the results only for the case when $\gamma_0 \in \mathcal{R}_1$, which is the novel part of the proof. Let us begin by defining the following quantity

$$\xi_{k,\mathcal{R}_1} \triangleq \sqrt{\frac{L_{\max}}{(\mu_{\hat{f}}-\gamma_0)\lambda_k}}, \tag{64}$$

where $L_{\max}$ was defined in (32). Next, (64) can be rewritten as

$$\frac{2}{\sqrt{\lambda_{k+1}}}-\frac{2}{\sqrt{\lambda_k}} \geq \sqrt{\frac{\mu_{\hat{f}}-\gamma_0}{L_k}}\sqrt{\frac{\mu_{\hat{f}}L_k}{L_k\lambda_{k+1}(\mu_{\hat{f}}-\gamma_0)}-1}. \tag{65}$$

Then, relaxing the bound in (66) and multiplying it with $\sqrt{\frac{L_{\max}}{\mu_{\hat{f}}-\gamma_0}}$, we obtain

$$\xi_{k+1,\mathcal{R}_1}-\xi_{k,\mathcal{R}_1} \geq \frac{1}{2}\sqrt{\frac{\mu_{\hat{f}}\xi_{k+1,\mathcal{R}_1}^2}{L_{\max}}-1}. \tag{66}$$

We then proceed to establish via induction the following lower bound

$$\xi_{k,\mathcal{R}_1} \geq \frac{\sqrt{2}}{4\delta}\sqrt{\frac{L_k}{\mu_{\hat{f}}-\gamma_0}}\left[e^{(k+1)\delta}-e^{(k+1)\delta}\right], \tag{67}$$

where $\delta \triangleq \frac{1}{2}\sqrt{\frac{\mu_{\hat{f}}}{L_{\max}}}$. Utilizing (65) at step $k=0$, we have

$$\xi_{0,\mathcal{R}_1} = \sqrt{\frac{L_{\max}}{(\mu_{\hat{f}}-\gamma_0)\lambda_0}} = \sqrt{\frac{L_{\max}}{\mu_{\hat{f}}-\gamma_0}}, \tag{68}$$

where the second equality is obtained because $\lambda_0 = 1$. Then, substituting (32) into (69), we obtain

$$\xi_{0,\mathcal{R}_1} \geq \frac{\sqrt{2}}{2}\sqrt{\frac{L_k}{\mu_{\hat{f}}-\gamma_0}}\left[e^{1/2}-e^{-1/2}\right] \geq \frac{\sqrt{2}}{4\delta}\sqrt{\frac{L_k}{\mu_{\hat{f}}-\gamma_0}}\left[e^{\delta}-e^{-\delta}\right]. \tag{69}$$

Note that the second row in (70) follows because the RHS is increasing in δ, which by construction is always $\delta < 0.5$.

As it is common with induction-type of proofs, the next step is to assume that (68) is satisfied for some iteration k. To establish that the relation would hold true at the next iteration as well, we proceed via contradiction. Define $\omega(t) \triangleq \frac{\sqrt{2}}{4\delta}\sqrt{\frac{L_k}{\mu_{\hat{f}}-\gamma_0}}\left[e^{(t+1)\delta}-e^{-(t+1)\delta}\right]$, and note that from Nesterov [51, Lemma 2.2.4] it is a convex function. Therefore, it can be written that

$$\omega(t) \leq \xi_{k,\mathcal{R}_1} \overset{(67)}{\leq} \xi_{k+1,\mathcal{R}_1}-\frac{1}{2}\sqrt{\frac{\mu_{\hat{f}}\xi_{k+1,\mathcal{R}_1}^2}{L_{\max}}-1}. \tag{70}$$

Assuming that $\xi_{k+1,\mathcal{R}_1} < \omega(t+1)$ and substituting it into (71), we have

$$\omega(t) < \omega(t+1)-\frac{1}{2}\sqrt{\frac{\mu_{\hat{f}}\xi_{k+1,\mathcal{R}_1}^2}{L_{\max}}-1}. \tag{71}$$

Then, utilizing the definition of δ, as well as (68), we obtain

$$\begin{aligned}\omega(t) &\leq \omega(t+1)-\frac{1}{2}\sqrt{4\delta^2\left[\frac{\sqrt{2}}{4\delta}\sqrt{\frac{L_k}{\mu_{\hat{f}}-\gamma_0}}\left(e^{(t+2)\delta}-e^{-(t+2)\delta}\right)\right]^2-1}\\ &\leq \omega(t+1)-\frac{\sqrt{2}}{4}\sqrt{\frac{L_k}{\mu_{\hat{f}}-\gamma_0}}\left[e^{(t+2)\delta}+e^{-(t+2)\delta}\right] = \omega(t+1)\\ &\quad +\omega'(t+1)(t-(t+1)) \leq \omega(t),\end{aligned} \tag{72}$$

where the last inequality follows from the supporting hyperplane theorem of convex functions. Notice that this result contradicts the earlier assumption that $\xi_{k+1,\mathcal{R}_1} < \omega(t+1)$. Thus, the inductive argument asserts that we have established the lower bound (68) to be true for all values of $k = 0, 1, \ldots$.

We are finally ready to establish (34). From (65), it can be written that

$$\lambda_k = \frac{L_{\max}}{\xi_{k+1,\mathcal{R}_1}^2(\mu_{\hat{f}}-\gamma_0)}. \tag{73}$$

Utilizing (68) in the RHS of (74), we reach

$$\lambda_k \leq \frac{(4\delta)^2 L_{\max}}{2L_k\left[e^{(k+1)\delta}-e^{(k+1)\delta}\right]^2}, \tag{74}$$

The first inequality in (34) is obtained by substituting the definition of δ in (75).

To establish the remaining inequality in (34), we first analyze the following

$$\left(e^{\frac{k+1}{2}\sqrt{\frac{\mu_{\hat{f}}}{L_k}}}-e^{-\frac{k+1}{2}\sqrt{\frac{\mu_{\hat{f}}}{L_k}}}\right)^2 = e^{(k+1)\sqrt{\frac{\mu_{\hat{f}}}{L_k}}}-e^{-(k+1)\sqrt{\frac{\mu_{\hat{f}}}{L_k}}}-2. \tag{75}$$

Then, utilizing the definition of the hyperbolic cosine function in (76), we obtain

$$\left(e^{\frac{k+1}{2}\sqrt{\frac{\mu_{\hat{f}}}{L_k}}}-e^{-\frac{k+1}{2}\sqrt{\frac{\mu_{\hat{f}}}{L_k}}}\right)^2 = 2\cosh\left(\sqrt{\frac{\mu_{\hat{f}}}{L_k}}(k+1)-2\right). \tag{76}$$

Using the Taylor expansion of the hyperbolic cosine function, yields

$$\begin{aligned}\left(e^{\frac{k+1}{2}\sqrt{\frac{\mu_{\hat{f}}}{L_k}}}-e^{-\frac{k+1}{2}\sqrt{\frac{\mu_{\hat{f}}}{L_k}}}\right)^2 &= -2+2+2\frac{\mu_{\hat{f}}(k+1)^2}{2L_k}\\ &\quad +2\frac{\mu_{\hat{f}}^2(k+1)^4}{4!{L_k}^2}+\ldots.\end{aligned} \tag{77}$$

The next step is to truncate the RHS of (78). This results in

$$\left(e^{\frac{k+1}{2}\sqrt{\frac{\mu_{\hat{f}}}{L_k}}}-e^{-\frac{k+1}{2}\sqrt{\frac{\mu_{\hat{f}}}{L_k}}}\right)^2 \geq \frac{\mu_{\hat{f}}}{L_k}(k+1)^2. \tag{78}$$

All that remains for establishing the second inequality of (34), is to substitute (79) into the denominator of the first inequality of (34).

Appendix G. Proof of Lemma 5

We begin by substituting the upper bound (4) evaluated at the point $y = x^*$ into (3), and obtain that

$$F(x_0) = \hat{f}(x_0) + \tau\hat{g}(x_0) \leq \hat{f}(x^*) + \nabla\hat{f}(x^*)^T(x_0 - x^*) + \frac{L_0}{2}||x_0 - x^*||^2 + \tau\hat{g}(x_0). \tag{79}$$

Then, from the equality established in (12), the RHS of (80) can be written as

$$\begin{aligned} F(x_0) \leq \hat{f}(x^*) + \nabla\hat{f}(x^*)^T(x_0 - x^*) + \frac{L_0}{2}\|x_0 - x^*\|^2 + \tau\hat{g}(x_0) = \hat{f}(x^*) \\ + \left(\tau s_{L_0}(x^*) - L_0\left(x^* - T_{L_0}(x^*)\right)\right)^T(x_0 - x^*) + \frac{L_0}{2}\|x_0 - x^*\|^2 \\ + \tau\hat{g}(x_0). \end{aligned} \tag{80}$$

From the definition of the composite gradient mapping given in (9), we can see that when $y = x^*$, then $T_{L_0}(x^*) = x^*$. Therefore, the RHS of (81) becomes

$$F(x_0) \leq \hat{f}(x^*) - \tau s_{L_0}(x^*)^T(x^* - x_0) + \frac{L_0}{2}||x_0 - x^*||^2 + \tau\hat{g}(x_0). \tag{81}$$

Lastly, utilizing (6) in the RHS of (82) completes the proof.

References

[1] Y. Nesterov, Gradient methods for minimizing composite objective function, Math. Program. 140 (1) (Aug. 2013) 125–161.
[2] N. Parikh, S. Boyd, Proximal algorithms, Found. Trends Optim. 1 (3) (Jan. 2014) 127–239.
[3] V. Cevher, S. Becker, M. Schmidt, Convex optimization for big data: scalable, randomized, and parallel algorithms for big data analytics, IEEE Signal Process. Mag. 31 (5) (Aug. 2014) 32–43.
[4] K. Slavakis, G.B. Giannakis, G. Mateos, Modeling and optimization for big data analytics: (statistical) learning tools for our era of data deluge, IEEE Signal Process. Mag. 31 (5) (Aug. 2014) 18–31.
[5] A.P. Liavas, G. Kostoulas, G. Lourakis, K. Huang, N.D. Sidiropoulos, Nesterov-based alternating optimization for nonnegative tensor factorization: algorithm and parallel implementation, IEEE Trans. Signal Process. 66 (4) (Nov. 2018) 944–953.
[6] M.S. Ibrahim, A. Konar, N.D. Sidiropoulos, Fast algorithms for joint multicast beamforming and antenna selection in massive MIMO, IEEE Trans. Signal Process. 68 (Mar. 2020) 1897–1909.
[7] R. Gu, A. Dogandžić, Projected Nesterov's proximal-gradient algorithm for sparse signal recovery, IEEE Trans. Signal Process. 65 (13) (May. 2017) 3510–3525.
[8] K. Elkhalil, A. Kammoun, X. Zhang, M. Alouini, T. Al-Naffouri, Risk convergence of centered kernel ridge regression with large dimensional data, IEEE Trans. Signal Process. 68 (Feb. 2020) 1574–1588.
[9] M.J. Wainwright, Structured regularizers for high-dimensional problems: statistical and computational issues, Annu. Rev. Stat. Appl. 1 (1) (Jan. 2014) 233–253.
[10] R. Tibshirani, Regression shrinkage and selection via the lasso, J. R. Stat. Soc. 58 (1) (Jan. 1996) 267–288.
[11] J. Tropp, S.J. Wright, Computational methods for sparse solution of linear inverse problems, Proc. IEEE 98 (6) (Apr. 2010) 948–958.
[12] P.L. Combettes, J.C. Pesquet, Proximal splitting methods in signal processing, Fixed-Point Algorithms Inverse Probl.Sci. Eng. 49 (May. 2011) 185–212.
[13] E.J. Candès, Y.C. Eldar, T. Strohmer, V. Voroninski, Phase retrieval via matrix completion, SIAM Rev. 57 (2) (May. 2015) 225–251.
[14] A. Yurtsever, Y.P. Hsieh, V. Cevher, Scalable convex methods for phase retrieval, in: Proc. IEEE 6th International Workshop on Computational Advances in Multi-Sensor Adaptive Processing, Cancun, Mexico, Dec. 2015, pp. 381–384.
[15] C. Studer, T. Goldstein, W. Yin, R.G. Baraniuk, Democratic representations, arXiv:1401.3420(Apr. 2015).
[16] Y. Nesterov, Subgradient methods for huge-scale optimization problems, Math. Program. 146 (1) (Aug. 2014) 275–297.
[17] A. Beck, First-order Methods in Optimization, vol. 25, SIAM, Oct. 2017.
[18] A. d'Aspremont, D. Scieur, A. Taylor, Acceleration methods, Found. Trends Optim. 5 (1–2) (Dec. 2021) 1–245.
[19] A. Nemirovsky, D. Yudin, Problem Complexity and Method Efficiency in Optimization, Wiley, 1983.
[20] B.T. Polyak, Some methods of speeding up the convergence of iteration methods, USSR Comput. Math. Math. Phys. 4 (5) (1964) 1–17.
[21] Y. Nesterov, A method for solving the convex programming problem with convergence rate $\mathcal{O}(1/k^2)$, Doklady USSR 269 (1983) 543–547.
[22] A. Auslender, M. Teboulle, Interior gradient and proximal methods for convex and conic optimization, SIAM J. Optim. 16 (3) (Jul. 2006) 697–725.
[23] G. Lan, Z. Lu, R.D.C. Monteiro, Primal-dual first-order methods with $\mathcal{O}(1/\epsilon)$ iteration-complexity for cone programming, Math. Program. 126 (1) (Jan. 2011) 1–29.
[24] B. O'donoghue, E. Candès, Adaptive restart for accelerated gradient schemes, Found. Comput. Math. 15 (3) (Jun. 2015) 715–732.
[25] A. d'Aspremont, Smooth optimization with approximate gradient, SIAM J. Optim. 19 (3) (Oct. 2008) 1171–1183.
[26] M. Schmidt, N.L. Roux, F.R. Bach, Convergence rates of inexact proximal-gradient methods for convex optimization, in: Proc. 25th Annual Conference on Neural Information Processing Systems, Granada, Spain, 2011, pp. 1458–1466.
[27] O. Devolder, F. Glineur, Y. Nesterov, First-order methods of smooth convex optimization with inexact oracle, Math. Program. 146 (1) (Aug. 2014) 37–75.
[28] Z. Allen-Zhu, L. Orecchia, Linear coupling: an ultimate unification of gradient and mirror descent, Nov. 2016, arXiv:1407.1537.
[29] S. Bubeck, Y.T. Lee, M. Singh, A geometric alternative to Nesterov's accelerated gradient descent, Jun. 2015. arXiv:1506.08187.
[30] N. Flammarion, F. Bach, From averaging to acceleration, there is only a step-size, in: Proc. Conference on Learning Theory, Paris, France, 2015, pp. 658–695.
[31] W. Su, S. Boyd, E.J. Candès, A differential equation for modeling Nesterov's accelerated gradient method: theory and insights, J. Mach. Learn. Res. 17 (153) (Jan. 2016) 1–43.
[32] L. Lessard, B. Recht, A. Packard, Analysis and design of optimization algorithms via integral quadratic constraints, SIAM J. Optim. 26 (1) (Jan. 2016) 57–95.
[33] Y. Drori, M. Teboulle, Performance of first-order methods for smooth convex minimization: a novel approach, Math. Program. 145 (1) (Jun. 2014) 451–482.
[34] D. Kim, J.A. Fessler, Optimized first-order methods for smooth convex minimization, Math. Program. 159 (1) (Sep. 2016) 81–107.
[35] D. Kim, J.A. Fessler, Generalizing the optimized gradient method for smooth convex minimization, SIAM J. Optim. 28 (2) (Jun. 2018) 1920–1950.
[36] S. Bubeck, Convex optimization: algorithms and complexity, Found. Trends Mach. Learn. (May. 2014) 231–357.
[37] M.I. Florea, S.A. Vorobyov, An accelerated composite gradient method for large-scale composite objective problems, IEEE Trans. Signal Process. 67 (2) (Jan. 2019) 444–459.
[38] Y. Nesterov, Universal gradient methods for convex optimization problems, Math. Program. 152 (1) (Aug. 2015) 381–404.
[39] Y. Nesterov, Accelerating the cubic regularization of Newton's method on convex problems, Math. Program. 112 (1) (Mar. 2008) 159–181.
[40] X. Chen, B. Jiang, T. Lin, S. Zhang, Accelerating adaptive cubic regularization of Newton's method via random sampling, J. Mach. Learn. Res. 23 (Mar. 2022) 1–38.
[41] Y. Nesterov, Inexact high-order proximal-point methods with auxiliary search procedure, SIAM J. Optim. 31 (4) (Nov. 2021) 2807–2828.
[42] N. Doikov, Y. Nesterov, High-order optimization methods for fully composite problems, SIAM J. Optim. 32 (3) (Sep. 2022) 2402–2427.
[43] X. Zeng, J. Lei, J. Chen, Dynamical primal-dual accelerated method with applications to network optimization, IEEE Trans. Autom. Control (2022), doi:10.1109/TAC.2022.3152720. (Early Access)
[44] J. Gao, X. Liu, Y. Dai, Y. Huang, P. Yang, A family of distributed momentum methods over directed graphs with linear convergence, IEEE Trans. Autom. Control (2022), doi:10.1109/TAC.2022.3160684. (Early Access)
[45] S.S. Mannelli, P. Urbani, Analytical study of momentum-based acceleration methods in paradigmatic high-dimensional non-convex problems, Adv. Neural Inf. Process. Syst. 34 (Dec. 2021) 187–199.
[46] X. Xie, P. Zhou, H. Li, Z. Lin, S. Yan, ADAN: adaptive Nesterov momentum algorithm for faster optimizing deep models, Aug. 2022, arXiv:2208.06677.
[47] A. Kulunchakov, J. Mairal, Estimate sequences for stochastic composite optimization: variance reduction, acceleration, and robustness to noise, J. Mach. Learn. Res. 21 (155) (Jul. 2020) 1–52.
[48] M. Even, R. Berthier, F. Bach, N. Flammarion, P. Gaillard, H. Hendrikx, L. Massoulié, A. Taylor, A continuized view on Nesterov acceleration for stochastic gradient descent and randomized gossip, Jun. 2021. arXiv:2106.07644.
[49] K. Ahn, S. Sra, From Nesterov's estimate sequence to Riemannian acceleration, in: Proc. Conference on Learning Theory, Graz, Austria, 2020, pp. 88–118.
[50] J. Kim, I. Yang, Nesterov acceleration for Riemannian optimization, Feb. 2022, arXiv:2202.02036.
[51] Y. Nesterov, Lectures on Convex Optimization, vol. 137, Springer, Dec. 2018.
[52] M. Baes, Estimate sequence methods: extensions and approximations, Institute for Operations Research, ETH, Zürich, Switzerland, Aug. 2009.
[53] A. Beck, M. Teboulle, A fast iterative shrinkage-thresholding algorithm for linear inverse problems, SIAM J. Imaging Sci. 2 (1) (Mar. 2009) 183–202.
[54] M.I. Florea, S.A. Vorobyov, A generalized accelerated composite gradient method: uniting Nesterov's fast gradient method and FISTA, IEEE Trans. Signal Process. 68 (Jul. 2020) 3033–3048.
[55] E. Dosti, S.A. Vorobyov, T. Charalambous, Embedding a heavy-ball type of momentum into the estimating sequences, Aug. 2020, arXiv:2008.07979.
[56] E. Dosti, S.A. Vorobyov, T. Charalambous, Generalizing Nesterov's acceleration framework by embedding momentum into estimating sequences: new algorithm and bounds, in: IEEE International Symposium on Information Theory (ISIT), Helsinki, Finland, Jun 2022, pp. 1506–1511.

[57] P. Tseng, On accelerated proximal gradient methods for convex-concave optimization, online available at https://www.mit.edu/~dimitrib/PTseng/papers/apgm.pdf.
[58] C.C. Chang, C.J. Lin, LIBSVM: a library for support vector machines, ACM Trans. Intell. Syst. Technol. 2 (3) (May. 2011) 1–27.
[59] M. Grant, S. Boyd, Y. Ye, CVX: Matlab software for disciplined convex programming (web page and software), 2009.
[60] S.R. Becker, E.J. Candès, M. Grant, Templates for convex cone problems with applications to sparse signal recovery, Math. Program. 3 (3) (2011) Sep.165.
[61] A. Chambolle, T. Pock, An introduction to continuous optimization for imaging, Acta Numer. 25 (2016) 161–319.

Publication V

E. Dosti, S. A. Vorobyov, T. Charalambous. Generalizing the estimating sequences with memory terms for minimizing convex composite functions. *Journal Submission*, March 2024.

GENERALIZING THE ESTIMATING SEQUENCES WITH MEMORY TERMS FOR MINIMIZING CONVEX COMPOSITE FUNCTIONS

ENDRIT DOSTI*, SERGIY A. VOROBYOV†, AND THEMISTOKLIS CHARALAMBOUS††

Abstract. In this work, we present a new class of generalized composite estimating sequences, devised by exploiting the information contained in the iterates that are formed during the minimization process. Based on the newly introduecd generalize estimating sequences, we present a new accelerated first-order methods for minimizing convex functions with composite objective structure. Our proposed method is equipped with backtracking line-search, and exhibits an accelerated convergence rate independent of whether the true value of the Lipschitz constant is known. Moreover, our proposed method is robust to the inexact knowledge of the strong convexity parameter. The efficiency of the proposed method together with its robustness properties are confirmed by extensive numerical evaluations on both synthetic and real-world data.

Key words. Accelerated first-order methods, large-scale optimization, composite objective, estimating sequence, gradient mapping, line-search

AMS subject classifications. 65B99, 65K10, 65K05, 65Y20

1. Introduction.

1.1. Motivation.

Recent research in first-order methods has been largely focused around exploring different approaches to acceleration of gradient-based methods. For the problem of minimizing smooth convex functions, an accelerated method built by making use of the linear coupling between gradient and mirror descent was introduced in [1]. Another accelerated method inspired by the ellipsoid method was presented in [2]. It converges faster than the Fast Gradient Method (FGM) [3, 4], however it exhibits higher per-iteration complexity because of the need for an exact line search. In yet another framework, the continuous-time limit of FGM has been modeled as a second-order differential equation [5, 6, 7]. In a newly developed framework [8], the authors have cast the improvement of the worst-case behavior of an algorithm as an optimization problem. Based on this framework, an optimal method for minimizing smooth convex functions has been presented in [9]. Despite the promising theoretical analysis, the applicability of these methods in the current form is restricted only to minimizing smooth convex functions and their generalization capabilities remain unclear.

Considering the different strategies that have been developed for accelerating gradient-based methods, estimating sequence methods continue to play a central role in the field (see [10] and references therein). First, for the case of differentiable convex functions such methods are optimal in the sense of [11]. Second, they are efficient in practice and can work well with backtracking line-search [12, 13]. Third, they can be used to devise fast second-order and higher-order methods [14, 15]. Fourth, their efficiency has also been established in the context of applications to distributed optimization, nonconvex optimization, stochastic optimization, and many more (see [16, 17, 18, 19, 20] and the references therein). As discussed in [4], different estimating sequences can be used to enable the accumulation of global information of the objective function. One of the main challenges with the framework is the design of estimating functions that are used to construct the estimating sequences.

*Department of Information and Communication Engineering (firstname.lastname@aalto.fi).

†Department of Electrical and Computer Engineering, University of Cyprus, Nicosia, Cyprus (charalambous.themistoklis@ucy.ac.cy)

The estimating sequences framework has been formalized in [21]. For the broader class of minimizing convex functions with composite structure, which is important to this paper, a popular method is the Accelerated Multistep Gradient Scheme (AMGS) [22], which exhibits an accelerated convergence rate. The method has the disadvantage of requiring two projection-like operations per iteration, which translates in an increased runtime of the method and inhibits its deployment to practical large-scale optimization setups [23]). Another popular method is the Fast Iterative Shrinkage-Thresholding Algorithm (FISTA) [24]. Unlike AMGS, it requires one projection-like operation per iteration and has been proven to exhibit an accelerated convergence rate. Nevertheless, as we will also see in the numerical section, the method converges slower than AMGS. At first glance, FISTA does not appear as an estimating sequence method. Links between FISTA and estimating sequence methods have been established in [25]. In [26, 27] the authors have introduced COMET, which is a new estimating sequence method, which is built on top of the estimating sequences framework used for devising FGM. Similar to FISTA, the method proposed therein requires one projection-like operation per iteration, and is more efficient than AMGS.

1.2. Preliminaries to build on. In the sequel, we will focus on devising an accelerated black-box method for solving convex optimization problems with composite objective functions. The typical structure for such problems is

$$F(x) = f(x) + \tau g(x), \quad \tau > 0, \tag{1.1}$$

where $f : \mathcal{R}^n \to \mathcal{R}$ is a differentiable convex function and $g : \mathcal{R}^n \to \mathcal{R}$ is a simple convex lower semi-continuous function. The simplicity of g implies that the complexity of computing the proximal map

$$\text{prox}_{\tau g} \triangleq \arg\min_{z \in \mathcal{R}^n} \quad \left(g(z) + \frac{1}{2\tau} ||z - x||^2 \right), \quad x \in \mathcal{R}^n, \tag{1.2}$$

is $\mathcal{O}(n)$ [28]. Herein $|| \cdot ||$ denotes the l_2 norm.

Assuming that $g(x)$ has strong convexity parameter $\mu_g \geq 0$, we use the following strong convexity transfer

$$F(x) = \left(f(x) + \frac{\tau \mu_g}{2} ||x - x_0||^2 \right) + \tau \left(g(x) - \frac{\mu_g}{2} ||x - x_0||^2 \right) = \hat{f}(x) + \tau \hat{g}(x), \tag{1.3}$$

to facilitate the tractability of the derivations presented in the sequel. Based on (1.3), we have $L_{\hat{f}} = L_f + \tau \mu_g$, $\mu_{\hat{f}} = \mu_f + \tau \mu_g$ and $\mu_{\hat{g}} = 0$.

For all $y \in \mathcal{Q}$, where $\mathcal{Q}$ is a closed convex set and $L \geq L_{\hat{f}}$, let us define

$$m_L(y; x) \triangleq \hat{f}(y) + \nabla \hat{f}(y)^T (x - y) + \frac{L}{2} ||x - y||^2 + \tau \hat{g}(x). \tag{1.4}$$

The following bounds for $\hat{f}(x)$ and $\hat{g}(x)$ will be useful in the analysis

$$\hat{f}(x) \leq \hat{f}(y) + \nabla \hat{f}(y)^T (x - y) + \frac{L_{\hat{f}}}{2} ||y - x||^2, \tag{1.5}$$

$$\hat{g}(x) \geq \hat{g}(y) + s(y)^T (x - y), \tag{1.6}$$

Considering (1.4) and (1.5), we have

$$m_L(y; x) \geq F(x), \forall x, y \in \mathcal{Q}. \tag{1.7}$$

Next, we define the composite gradient mapping as

$$T_L(y) \triangleq \arg\min_{x \in \mathcal{Q}} m_L(y; x). \tag{1.8}$$

We conclude by introducing the reduced composite gradient

$$r_L(y) \triangleq L\left(y - T_L(y)\right). \tag{1.9}$$

Consider now the optimality conditions for (1.8):

$$\begin{aligned} \nabla m_L(y; T_L(y))^T (x - T_L(y)) &\geq 0, \\ \left(\nabla f(y) + L(T_L(y) - y) + \tau s_L(y)\right)^T \left(x - T_L(y)\right) &\geq 0, \end{aligned} \tag{1.10}$$

where $s_L(y)$ is a subgradient and $F(T_L(y))$ is the subdifferential. In (1.10), let

$$\nabla f(y) + L(T_L(y) - y) + \tau s_L(y) = 0. \tag{1.11}$$

Considering (1.9) and (1.11) yields

$$r_L(y) = L(y - T_L(y)) = \nabla f(y) + \tau s_L(y). \tag{1.12}$$

Last, we note that in the paper we will make use of the following bounds [26, 27]

$$F(x) \geq \hat{f}(T_L(y)) + \tau \hat{g}(T_L(y)) + r_L(y)^T (x - y) + \frac{\mu_{\hat{f}}}{2}||x - y||^2 + \frac{1}{2L}||r_L(y)||^2, \tag{1.13}$$

$$F(x_0) \leq F(x^*) + \frac{L_0}{2}||x_0 - x^*||^2. \tag{1.14}$$

In this paper, we will focus on designing first-order methods. For such methods, at any iteration t, the iterates are in the span of the gradients, i.e., $x_k \in x_0 + \mathbf{span}\{\nabla f(x_0), \ldots \nabla f(x_{k-1})\}$ for $k = 0, 1, 2, \ldots, t$. Then, set $\mathcal{Q} = \mathbf{span}(x_1, x_2, \ldots)$.

1.3. The main idea. Contrasting the analysis conducted for AMGS in [22] with FGM in [4], we can see that different estimating functions were used. The method in [26, 27] is also devised using the estimating sequences framework. As discussed earlier, the lack of uniqueness of the estimating sequences is one of the main challenges we face in developing methods under such framework. In theory, when used to solve convex problems, both methods exhibit an accelerated convergence rate. Moreover, in [29, 30] the authors have shown how to devise generalized estimating sequences, which can be used to construct faster algorithms. Thus, it is of practical interest to develop the framework for non-differentiable functions.

1.4. Contributions. The main contributions of the article are as follows.

- We introduce a new structure for the estimating functions, which we call the *generalized composite estimating functions*. The proposed estimating functions are devised by making use of the following: *i)* A new term created by adding the previously constructed estimating functions *ii)* The gradient mapping framework [11]. *iii)* The tighter lower bound on the objective function presented in (1.13).
- Using our proposed estimating sequences, we devise a new accelerated method for minimizing (1.1). Moreover, we present an efficient line-search strategy which is used to estimate the step size. Our proposed method requires only one projection-like operation per iteration, which is lower than the respective requirement for AMGS.

- We prove that our proposed method exhibits an accelerated convergence rate, despite the imperfect knowledge of the Lipschitz constant.
- We prove that the way our proposed method is initialized is robust to the inexact knowledge of $\mu_{\hat{f}}$, which further reduces the additional computational burden of having to estimate such parameter.
- We demonstrate the efficiency of our proposed method as compared to the existing benchmarks. Using real-world datasets, in our computational experiments we also highlight the robustness of our proposed method in cases when $\mu_{\hat{f}}$ and $L_{\hat{f}}$ are not known.

1.5. Contents. The article is organized as follows. In Section 2, we present the generalized composite estimating sequences and show how they can be used to build our proposed method. In Section 3, we prove the convergence results for our proposed method. In Section 4, we depict the numerical performance of our proposed method and compare with several existing benchmarks. We consider several types of optimization problems and demonstrate the efficiency of our proposed metod. Last, in Section 5, we summarize the main findings of the paper.

2. Proposed Method. Consider the following definition for the generalized composite estimating sequences.

DEFINITION 2.1. *The sequences $\{\Phi_k\}_k$ and $\{\lambda_k\}_k$, $\lambda_k \geq 0$, are called generalized composite estimating sequences of the function $F(\cdot)$ defined in (1.3), if there exists a sequence of bounded functions $\{\psi_k\}_k$, $\lambda_k \to 0$ as $k \to \infty$, and $\forall x \in \mathcal{Q}$, $\forall k \geq 0$ we have*

$$\Phi_k(x) \leq \lambda_k \Phi_0(x) + (1-\lambda_k)\left(F(x) - \psi_k(x)\right). \tag{2.1}$$

Next, let us use the generalized composite estimating sequences to characterize the convergence rate of the minimization process

LEMMA 2.2. *If for some sequence $\{x_k\}_k$ we have $F(x_k) \leq \Phi_k^* \triangleq \min_{x\in\mathcal{Q}} \Phi_k(x)$, then $F(x_k) - F(x^*) \leq \lambda_k \left[\Phi_0(x^*) - F(x^*)\right] - (1-\lambda_k)\psi_k(x^*)$, where $x^* = \arg\min_{x\in\mathcal{Q}} F(x)$.*

Proof. By the assumption of Lemma 2.2, we have

$$\begin{aligned} F(x_k) \leq \Phi_k^* = \min_{x\in\mathcal{Q}} \Phi_k(x) &\overset{(2.1)}{\leq} \min_{x\in\mathcal{Q}} \lambda_k \Phi_0(x) + (1-\lambda_k)\left(F(x) - \psi_k(x)\right) \\ &\leq \lambda_k \Phi_0(x^*) + (1-\lambda_k)\left(F(x^*) - \psi_k(x^*)\right). \end{aligned}$$

Regrouping the terms concludes the proof. ∎

So far, we have presented the generalized composite estimating sequences and shown why they are useful. In the sequel, we present the estimating functions that will be used to devise our proposed method.

LEMMA 2.3. *Assume that there exist sequences $\{\alpha_k\}_k$, where $\alpha_k \in (0,1)$ $\forall k$, such that $\sum_{k=0}^{\infty} \alpha_k = \infty$; $\{\psi_k\}_k$ with an upper bound Ψ_k, such that $\{\psi_k\}_k \geq 0$; and an arbitrary sequence $\{y_k\}_k$. Furthermore, let $\psi_0(x) = 0$, $\lambda_0 = 1$ and assume that the estimates L_k of the Lipschitz constant $L_{\hat{f}}$ are selected in a way that inequality (1.5) is satisfied for all the iterates x_k and y_k. Then, the sequences $\{\Phi_k\}_k$ and $\{\lambda_k\}_k$, which*

are defined recursively as

$$\lambda_{k+1} = (1-\alpha_k)\lambda_k, \tag{2.2}$$

$$\begin{aligned}\Phi_{k+1}(x) = (1-\alpha_k)\left(\Phi_k(x)+\psi_k(x)\right) - \psi_{k+1}(x) - \Psi_k \\ +\alpha_k\left(F\left(T_{L_k}(y_k)\right) + \psi_k(x) + \frac{1}{2L_k}||r_{L_k}(y_k)||^2\right) \\ +\alpha_k\left(r_{L_k}(y_k)^T(x-y_k) + \frac{\mu_{\hat{f}}}{2}||x-y_k||^2\right),\end{aligned} \tag{2.3}$$

are composite estimating sequences.

Proof. We prove this by induction. At step $k=0$, considering (2.1) together with the facts that $\lambda_0 = 1$ and $\psi_0(x) = 0$, we can write: $\Phi_0(x) \leq \lambda_0\Phi_0(x) + (1-\lambda_0)F(x) \equiv \Phi_0(x)$. At iteration k, assume (2.1) holds true, which results in

$$\Phi_k(x) - (1-\lambda_k)F(x) \leq \lambda_k\Phi_0(x) - (1-\lambda_k)\psi_k(x). \tag{2.4}$$

Utilizing (1.13) in (2.3), yields

$$\Phi_{k+1}(x) \leq (1-\alpha_k)\Big(\Phi_k(x)+\psi_k(x)\Big) + \alpha_k\left(F(x)+\psi_k(x)\right) - \psi_{k+1}(x) - \Psi_k. \tag{2.5}$$

Considering that Ψ_k is an upper bound on $\psi_k(x)$, and adding to the right-hand side (RHS) of (2.5), results in

$$\begin{aligned}\Phi_{k+1}(x) \leq (1-\alpha_k)\Phi_k(x) + \alpha_k F(x) + (1-\alpha_k)(1-\lambda_k)F(x) \\ -(1-\alpha_k)(1-\lambda_k)F(x) - \psi_{k+1}(x).\end{aligned} \tag{2.6}$$

Relaxing the RHS of (2.6), yields

$$\Phi_{k+1}(x) \leq (1-\alpha_k)\Big(\Phi_k(x) - (1-\lambda_k)F(x)\Big) + (\alpha_k + (1-\lambda_k)(1-\alpha_k))F(x) - \psi_{k+1}(x). \tag{2.7}$$

Substituting (2.4) in (2.7), results in

$$\Phi_{k+1}(x) \leq (1-\alpha_k)\lambda_k\left(\Phi_0(x) - (1-\lambda_k)\psi_k(x)\right) + (1-\lambda_k+\alpha_k\lambda_k)F(x) - \psi_{k+1}(x). \tag{2.8}$$

Last, relaxing the RHS of (2.8) and using (2.2) yields

$$\Phi_{k+1}(x) \leq \lambda_{k+1}\Phi_0(x) + (1-\lambda_{k+1})\left(F(x) - \psi_{k+1}(x)\right). \tag{2.9}$$

∎

Let us now compare between the different estimating sequence constructions that exist in the literature. First, observe that the estimating sequences used to construct FGM in [4, Lemma 2.2.4] are the instance of our proposed generalized composite estimating sequences obtained when $\tau = 0$ and $\{\psi_k\}_k = 0$. Moreover, both types of estimating sequences can be used to measure the convergence rate of the minimization process. In this sense, the framework presented herein, is a generalization of the estimating sequences framework. Comparing our generalized composite estimating sequences to

[26, 27], we can see that the introduction of the terms $\{\psi_k\}_k$ can have an additional impact on the convergence rate of the minimization process.

We observe now that there are different ways to choose the terms $\{\Phi_k\}_k$ and $\{\psi_k(x)\}_k$. Let $\gamma_k \in \mathcal{R}^+$, $v_k \in \mathcal{R}^n$, $\forall k = 0, 1, \dots$ and define the terms $\{\Phi_k\}_k$ as

$$\Phi_k(x) \triangleq \phi_k^* + \frac{\gamma_k}{2}||x - v_k||^2 - \psi_k(x), \ \forall k = 1, 2, \dots, \tag{2.10}$$

If we have no prior knowledge about the particular structure of $F(x)$, the terms of the sequence $\{\psi_k(x)\}_k$ can be chosen to account for the accumulation of the terms in the sequence $\{\Phi_k(x)\}_k$ as follows

$$\psi_k(x) \triangleq \sum_{i=1}^{k-1} \beta_{i,k} \frac{\gamma_i}{2}||x - v_i||^2, \quad \forall k, \tag{2.11}$$

where $\beta_{i,k} \in [0, 1], \forall i = 1, \dots, k-1$.

Considering the definition introduced above for $\Phi_k(x)$ and $\psi_k(x)$, it is of interest to assess the conditions for $\psi_k(x)$ that ensure the convexity of $\Phi_k(x)$. Since both functions are twice differentiable, assessing the second order condition for (2.10), we have $\sum_{i=1}^{k-1} \beta_{i,k}\gamma_i \leq \gamma_k$. Moreover, we also restrict $\sum_{i=1}^{k-1} \beta_{i,k}\gamma_i \leq \mu$. Combining these conditions, we reach

$$\sum_{i=1}^{k-1} \beta_{i,k}\gamma_i \leq \min(\gamma_k, \mu). \tag{2.12}$$

We can find the minimal value of the estimating function introduced in (2.10) as

$$\Phi_k^* = \min_x \Phi_k(x) = \phi_k^* + \frac{\gamma_k}{2}||x_{\Phi_k}^* - v_k||^2 - \sum_{i=1}^{k-1} \frac{\beta_{i,k}\gamma_i}{2}||x_{\Phi_k}^* - v_i||^2, \tag{2.13}$$

where $x_{\Phi_k}^* = \arg\min_x \Phi_k(x)$. The values of the parameters still need to be computed in a recurrent manner. The following Lemma captures these relations for the components of $\{\Phi_k\}_k$ introduced in (2.10).

LEMMA 2.4. *Assume that the coefficients $\beta_{i,k}$ are selected such that (2.12) is satisfied and let $\phi_0(x) = \phi_0^* + \frac{\gamma_0}{2}||x - v_0||^2$, where $\gamma_0 \in \mathcal{R}^+$ and $v_0 \in \mathcal{R}^n$. Then, the process defined in Lemma 2.3 preserves the canonical form of the function $\Phi_k(x)$ presented in (2.10), where the sequences $\{\gamma_k\}_k$, $\{v_k\}_k$ and $\{\phi_k^*\}k$ can be computed as*

$$\gamma_{k+1} = (1 - \alpha_k)\gamma_k + \alpha_k \left(\mu_{\hat{f}} + \sum_{i=1}^{k-1} \beta_{i,k}\gamma_i\right), \tag{2.14}$$

$$v_{k+1} = \frac{1}{\gamma_{k+1}}\left((1-\alpha_k)\gamma_k v_k + \alpha_k\left(\mu_{\hat{f}} y_k + \sum_{i=1}^{k-1} \beta_{i,k}\gamma_i - L(y_k - T_{L_k}(y_k))\right)\right), \tag{2.15}$$

$$\phi^*_{k+1} = (1-\alpha_k)\phi^*_k + \alpha_k\left(F\left(T_{L_k}(y_k)\right) + \frac{1}{2L_k}||r_{L_k}(y_k)||^2 + \sum_{i=1}^{k-1}\beta_{i,k}\gamma_i||y_k - v_i||^2\right)$$

$$- \frac{L_k^2\alpha_k^2}{2\gamma_{k+1}}||y_k - T_{L_k}(y_k)||^2 + \frac{\alpha_k\gamma_k(1-\alpha_k)\left(\mu_{\hat{f}} + \sum_{i=1}^{k-1}\beta_{i,k}\gamma_i\right)}{2\gamma_{k+1}}||y_k - v_k||^2$$

$$+ \frac{(1-\alpha_k)\gamma_k}{\gamma_{k+1}}||x^*_{\Phi_k} - v_k||^2 + \sum_{i=1}^{k}\frac{\beta_{i,k+1}\gamma_i}{2}||x^*_{\Phi_{k+1}} - v_i||^2$$

$$+ \frac{\alpha_k^2(1-\alpha_k)}{\gamma_{k+1}}\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i(v_i - y_k)^T r_{L_k}(y_k) + \frac{\alpha_k^3}{\gamma_{k+1}}\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i||v_i - y_k||\,||r_{L_k}(y_k)||$$

(2.16)

$$+ \frac{\alpha_k\gamma_k(1-\alpha_k)}{\gamma_{k+1}}\left((v_k - y_k)^T r_{L_k}(y_k) + \sum_{i=1}^{k-1}\beta_{i,k}\gamma_i||y_k - v_i||\,||y_k - v_k||\right).$$

Proof. Recall that for $k = 0$, we have $\psi_0(x) = 0$. Thus, $\nabla^2\Phi_0(x) = \gamma_0 I$. Assume that for step k we have: $\nabla^2\Phi_k(x) = \gamma_k I - \sum_{i=1}^{k-1}\beta_{i,k}\gamma_i I$. For step $k+1$, consider the following

$$\nabla^2\Phi_{k+1}(x) \overset{(2.3)}{=} (1-\alpha_k)\gamma_k I - \sum_{i=1}^{k}\beta_{i,k}\gamma_i I + \alpha_k\left(\mu_{\hat{f}} + \sum_{i=1}^{k-1}\beta_{i,k}\gamma_i\right)I. \tag{2.17}$$

Massaging (2.17) we obtain:

$$\gamma_{k+1}I = (1-\alpha_k)\gamma_k I + \alpha_k\left(\mu_{\hat{f}} + \sum_{i=1}^{k-1}\beta_{i,k}\gamma_i\right)I. \tag{2.18}$$

Substituting (2.14) in (2.18) is sufficient to establish that the quadratic cannonical structure for $\{\Phi_k\}_k$ is preserved.

Let us next focus on finding the recurrent relations for the terms $\{v_k\}_k$. First, replacing (2.10) in (2.3) and making some algebraic manipulations, results in

$$\phi^*_{k+1} + \frac{\gamma_{k+1}}{2}||x - v_{k+1}||^2 = (1-\alpha_k)\left(\phi^*_k + \frac{\gamma_k}{2}||x - v_k||^2\right) - \Psi_k + \alpha_k\Bigg(F\left(T_{L_k}(y_k)\right) + \psi_k(x)$$

$$+ \frac{1}{2L_k}||r_{L_k}(y_k)||^2 + r_{L_k}(y_k)^T(x - y_k) + \frac{\mu_{\hat{f}}}{2}||x - y_k||^2\Bigg). \tag{2.19}$$

Observe that both sides of (2.19) are convex in x. From the first-order optimality conditions we have

(2.20)

$$\gamma_{k+1}(x - v_{k+1}) = \gamma_k(1-\alpha_k)(x - v_k) + \alpha_k\left(\mu_{\hat{f}}(x - y_k) + r_{L_k}(y_k) + \sum_{i=1}^{k-1}\beta_{i,k}\gamma_i(x - v_i)\right).$$

Substituting (2.14) in (2.20), and reducing the dependency on x results in

$$-\gamma_{k+1}v_{k+1} = \alpha_k\left(r_{L_k}(y_k) - \mu_{\hat{f}}y_k - \sum_{i=1}^{k-1}\beta_{i,k}\gamma_i v_i\right) - (1-\alpha_k)\gamma_k v_k. \tag{2.21}$$

Substituting (1.9) into (2.21) yields the desired (2.15).

Let us now focus on finding the terms $\{\phi_k^*\}_k$. A straightforward approach is to assume that there exists a sequence of estimating functions $\{\Theta_k(y_k)\}_k$ for the sequence $\{y_k\}_k$ that has the following structure

$$\Theta_k(y_k) = \theta_k^* + \frac{\gamma_k}{2}||y_k - v_k||^2 - \sum_{i=1}^{k-1} \frac{\beta_{i,k}\gamma_i}{2}||y_k - v_i||^2 \tag{2.22}$$

Next, consider (2.3) with $x = y_k$

$$\begin{aligned}\Theta_{k+1}(y_k) = &(1-\alpha_k)\left(\Theta_k(y_k) + \psi_k(y_k)\right) - \psi_{k+1}(y_k) - \Psi_k \\ &+\alpha_k\left(F\left(T_{L_k}(y_k)\right) + \psi_k(y_k) + \frac{1}{2L_k}||r_{L_k}(y_k)||^2\right).\end{aligned} \tag{2.23}$$

Substituting (2.11) and (2.22) into (2.23), and relaxing the RHS, results in

$$\begin{aligned}\theta_{k+1}^* + \frac{\gamma_{k+1}}{2}||y_k - v_{k+1}||^2 \leq &(1-\alpha_k)\left(\theta_k^* + \frac{\gamma_k}{2}||y_k - v_k||^2\right) + \alpha_k\Bigg(F\left(T_{L_k}(y_k)\right) \\ &+\frac{1}{2L_k}||r_{L_k}(y_k)||^2 + \sum_{i=1}^{k-1}\frac{\beta_{i,k}\gamma_i}{2}||y_k - v_i||^2\Bigg).\end{aligned} \tag{2.24}$$

Using (2.15), we can write

$$v_{k+1} - y_k = \frac{1}{\gamma_{k+1}}\left((1-\alpha_k)\gamma_k v_k + \alpha_k\left(\mu_{\hat{f}}y_k - r_{L_k}(y_k) + \sum_{i=1}^{k-1}\beta_{i,k}\gamma_i v_i\right) - \gamma_{k+1}y_k\right). \tag{2.25}$$

Substituting (2.14) into (2.25), and making some algebraic manipulations, results in

$$v_{k+1} - y_k = \frac{1}{\gamma_{k+1}}\left((1-\alpha_k)\gamma_k\Big(v_k - y_k\Big) + \alpha_k\Big(\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i(v_i - y_k) - r_{L_k}(y_k)\Big)\right). \tag{2.26}$$

Taking $||\cdot||^2$ of (2.26), multiplying with $\frac{\gamma_{k+1}}{2}$, and extending the RHS, we reach

$$\begin{aligned}\frac{\gamma_{k+1}}{2}||v_{k+1} - y_k||^2 = &\frac{(1-\alpha_k)^2\gamma_k^2}{2\gamma_{k+1}}||v_k - y_k||^2 + \frac{\alpha_k^2}{2\gamma_{k+1}}\Bigg(||r_{L_k}(y_k)||^2 \\ &+||\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i(v_i - y_k)||^2\Bigg) - \frac{\alpha_k^2}{\gamma_{k+1}}\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i(v_i - y_k)^T r_{L_k}(y_k) \\ &-\frac{\alpha_k(1-\alpha_k)\gamma_k}{\gamma_{k+1}}\left((v_k - y_k)^T r_{L_k}(y_k) - \sum_{i=1}^{k-1}\beta_{i,k}\gamma_i(v_i - y_k)^T(v_k - y_k)\right).\end{aligned} \tag{2.27}$$

Substituting (2.27) into (2.24), yields:

$$\theta^*_{k+1} \leq (1-\alpha_k)\theta^*_k + \frac{(1-\alpha_k)\gamma_k}{2}\left(1 - \frac{(1-\alpha_k)\gamma_k}{\gamma_{k+1}}\right)||y_k - v_k||^2 + \alpha_k\Bigg(F\left(T_{L_k}(y_k)\right) + \frac{1}{2L_k}||r_{L_k}(y_k)||^2 + \sum_{i=1}^{k-1}\frac{\beta_{i,k}\gamma_i}{2}||v_i - y_k||^2\Bigg) - \frac{\alpha_k^2}{2\gamma_{k+1}}\Bigg(||\sum_{i=1}^{k-1}\frac{\beta_{i,k}\gamma_i}{2}(y_k - v_i)||^2 + ||r_{L_k}(y_k)||^2\Bigg) + \frac{\alpha_k^2}{\gamma_{k+1}}\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i(v_i - y_k)^T r_{L_k}(y_k) + \frac{\alpha_k(1-\alpha_k)\gamma_k}{\gamma_{k+1}}\Bigg((v_k - y_k)^T r_{L_k}(y_k) - \sum_{i=1}^{k-1}\beta_{i,k}\gamma_i(v_i - y_k)^T(v_k - y_k)\Bigg). \tag{2.28}$$

In (2.28), using the Cauchy-Schwartz inequality and relaxing the upper bound, yields

$$\theta^*_{k+1} \leq (1-\alpha_k)\theta^*_k + \frac{\alpha_k\gamma_k(1-\alpha_k)(\mu_{\hat{f}} + \sum_{i=1}^{k-1}\beta_{i,k}\gamma_i)}{2\gamma_{k+1}}||y_k - v_k||^2 + \alpha_k\Bigg(F\left(T_{L_k}(y_k)\right) + \frac{1}{2L_k}||r_{L_k}(y_k)||^2 + \sum_{i=1}^{k-1}\frac{\beta_{i,k}\gamma_i}{2}||v_i - y_k||^2\Bigg)$$

$$- \frac{\alpha_k^2}{2\gamma_{k+1}}||r_{L_k}(y_k)||^2 + \frac{(1-\alpha_k)\gamma_k}{2}||x^*_{\Phi_k} - v_k||^2 + \frac{(1-\alpha_k)\alpha_k^2}{\gamma_{k+1}}\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i(v_i - y_k)^T r_{L_k}(y_k) + \frac{\alpha_k^3}{\gamma_{k+1}}\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i||v_i - y_k||||r_{L_k}(y_k)|| + \frac{\alpha_k^2}{\gamma_{k+1}}\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i(v_i - y_k)^T r_{L_k}(y_k) + \sum_{i=1}^{k}\frac{\beta_{i,k+1}\gamma_i}{2}||x^*_{\Phi_{k+1}} - v_i||^2 + \frac{\alpha_k(1-\alpha_k)\gamma_k}{\gamma_{k+1}}\Bigg((v_k - y_k)^T r_{L_k}(y_k) + \sum_{i=1}^{k-1}\beta_{i,k}\gamma_i||v_i - y_k||||v_k - y_k||\Bigg). \tag{2.29}$$

Last, recall that we want the estimating function to be as close to the objective function as possible. Thus, we let θ^*_{k+1} equal to the upper bound obtained in (2.29). Letting $\phi^*_k = \theta^*_k, \forall k$ concludes the proof. ∎

Comparing the result obtained in Lemma 2.4 with that of [4, Lemma 2.2.3], it can be seen that the recursive relations obtained for computing the elements of $\{v_k\}_k$ and $\{\phi^*_k\}_k$ now reflect on the usage of a new lower bound on the function that is being minimized, and the reduced composite gradient. Moreover, observe that the recurrent relations for computing $\{\gamma_k\}_k$, $\{v_k\}_k$ and $\{\phi^*_k\}_k$ all reflect the presence of the added memory terms that was used to construct them. Comparing the above obtained results [26, 27], we can observe the additional terms coming from the newly introduced memory terms into the generalized composite estimating sequences.

To devise our proposed method, we will use an inductive argument. Assume that for a step k we have

$$\Phi^*_k \overset{(2.13)}{=} \phi^*_k + \frac{\gamma_k}{2}||x^*_{\Phi_k} - v_k||^2 - \sum_{i=1}^{k-1}\frac{\beta_{i,k}\gamma_i}{2}||x^*_{\Phi_k} - v_i||^2 \geq F(x_k). \tag{2.30}$$

For the inductive argument to be complete, we need to establish that $\Phi^*_{k+1} \geq F(x_{k+1})$. Considering the assumption for iteration k, and using (1.9) in (2.16), yields

$$\begin{aligned}
\phi^*_{k+1} \geq\, & (1-\alpha_k)F(x_k)+\alpha_k\left(F\left(T_{L_k}(y_k)\right)+\frac{1}{2L_k}||r_{L_k}(y_k)||^2+\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i||y_k-v_i||^2\right)\\
& -\frac{L_k^2\alpha_k^2}{2\gamma_{k+1}}||y_k-T_{L_k}(y_k)||^2+\frac{\alpha_k\gamma_k(1-\alpha_k)\left(\mu_{\hat{f}}+\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i\right)}{2\gamma_{k+1}}||y_k-v_k||^2\\
& +\frac{(1-\alpha_k)\gamma_k}{\gamma_{k+1}}||x^*_{\Phi_k}-v_k||^2+\sum_{i=1}^{k}\frac{\beta_{i,k+1}\gamma_i}{2}||x^*_{\Phi_{k+1}}-v_k||^2\\
& +\frac{\alpha_k^2(1-\alpha_k)}{\gamma_{k+1}}\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i(v_i-y_k)^Tr_{L_k}(y_k)+\frac{\alpha_k^3}{\gamma_{k+1}}\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i||v_i-y_k||||r_{L_k}(y_k)||\\
& +\frac{\alpha_k\gamma_k(1-\alpha_k)}{\gamma_{k+1}}\left((v_k-y_k)^Tr_{L_k}(y_k)+\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i||y_k-v_i||||y_k-v_k||\right). \qquad (2.31)
\end{aligned}$$

Using (1.13) in (2.31), we reach

$$\begin{aligned}
\phi^*_{k+1} \geq\, & (1-\alpha_k)\left(F(T_{L_k}(y_k))+r_{L_k}(y_k)^T(x_k-y_k)+\frac{\mu}{2}||x_k-y_k||^2+\frac{1}{2L_k}||r_{L_k}(y_k)||^2\right)\\
& +\alpha_k\left(F\left(T_{L_k}(y_k)\right)+\frac{1}{2L_k}||r_{L_k}(y_k)||^2+\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i||y_k-v_i||^2\right)\\
& -\frac{L_k^2\alpha_k^2}{2\gamma_{k+1}}||y_k-T_{L_k}(y_k)||^2+\frac{\alpha_k\gamma_k(1-\alpha_k)\left(\mu_{\hat{f}}+\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i\right)}{2\gamma_{k+1}}||y_k-v_k||^2\\
& +\frac{(1-\alpha_k)\gamma_k}{\gamma_{k+1}}||x^*_{\Phi_k}-v_k||^2+\sum_{i=1}^{k}\frac{\beta_{i,k+1}\gamma_i}{2}||x^*_{\Phi_{k+1}}-v_k||^2\\
& +\frac{\alpha_k^2(1-\alpha_k)}{\gamma_{k+1}}\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i(v_i-y_k)^Tr_{L_k}(y_k)+\frac{\alpha_k^3}{\gamma_{k+1}}\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i||v_i-y_k||||r_{L_k}(y_k)||\\
& +\frac{\alpha_k\gamma_k(1-\alpha_k)}{\gamma_{k+1}}\left((v_k-y_k)^Tr_{L_k}(y_k)+\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i||y_k-v_i||||y_k-v_k||\right). \qquad (2.32)
\end{aligned}$$

Massaging (2.32) yields

$$\begin{aligned}
\phi^*_{k+1} \geq\, & F(T_{L_k}(y_k))+(1-\alpha_k)r_{L_k}(y_k)^T(x_k-y_k)+\sum_{i=1}^{k}\frac{\beta_{i,k+1}\gamma_i}{2}||x^*_{\Phi_{k+1}}-v_i||^2\\
& +\left(\frac{1}{2L_k}-\frac{\alpha_k^2}{2\gamma_{k+1}}\right)||r_{L_k}(y_k)||^2+\frac{\alpha_k^2(1-\alpha_k)}{\gamma_{k+1}}\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i(v_i-y_k)^Tr_{L_k}(y_k)\\
& +\frac{\alpha_k\gamma_k(1-\alpha_k)}{\gamma_{k+1}}(v_k-y_k)^Tr_{L_k}(y_k). \qquad (2.33)
\end{aligned}$$

Adding $\frac{\gamma_{k+1}}{2}||x^*_{\Phi_{k+1}}-v_{k+1}||^2$ to the left-hand side (LHS) of (2.33), as well as moving

the term $\sum_{i=1}^{k} \frac{\beta_{i,k+1}\gamma_i}{2}||x^*_{\Phi_{k+1}} - v_i||^2$ to the LHS, we can write

$$\Phi^*_{k+1} \geq F(T_{L_k}(y_k)) + (1-\alpha_k) r_{L_k}(y_k)^T (x_k - y_k) + \left(\frac{1}{2L_k} - \frac{\alpha_k^2}{2\gamma_{k+1}}\right) ||r_{L_k}(y_k)||^2$$

$$+ \frac{\alpha_k^2(1-\alpha_k)}{\gamma_{k+1}} \sum_{i=1}^{k-1} \beta_{i,k}\gamma_i (v_i - y_k)^T r_{L_k}(y_k) + \frac{\alpha_k\gamma_k(1-\alpha_k)}{\gamma_{k+1}} (v_k - y_k)^T r_{L_k}(y_k). \tag{2.34}$$

From (2.34), we have

$$\alpha_k = \sqrt{\frac{\gamma_{k+1}}{L_k}}. \tag{2.35}$$

Substituting (2.14) into (2.35), the solution for α_k is found as

$$\alpha_k = \frac{\mu_{\hat{f}} + \sum_{i=1}^{k-1} \beta_{i,k}\gamma_i - \gamma_k + \sqrt{\left(\mu_{\hat{f}} + \sum_{i=1}^{k-1} \beta_{i,k}\gamma_i - \gamma_k\right)^2 + 4L_k\gamma_k}}{2L_k}. \tag{2.36}$$

This allows to simplify (2.34) as

$$\Phi^*_{k+1} \geq F(T_{L_k}(y_k)) + (1-\alpha_k) r_{L_k}(y_k)^T (x_k - y_k) + \frac{\alpha_k^2(1-\alpha_k)}{\gamma_{k+1}} \sum_{i=1}^{k-1} \beta_{i,k}\gamma_i (v_i - y_k)^T r_L(y_k)$$
$$+ \frac{\alpha_k\gamma_k(1-\alpha_k)}{\gamma_{k+1}} (v_k - y_k)^T r_L(y_k).$$

Next, let us set

$$x_k - y_k + \frac{\alpha_k\gamma_k}{\gamma_{k+1}}(v_k - y_k) + \frac{\alpha_k^2}{\gamma_{k+1}} \sum_{i=1}^{k-1} \beta_{i,k}\gamma_i (v_i - y_k) = 0, \tag{2.37}$$

which yields

$$y_k = \frac{\gamma_{k+1}x_k + \alpha_k\gamma_k v_k + \alpha_k^2 \sum_{i=1}^{k-1} \beta_{i,k}\gamma_i v_i}{\gamma_{k+1} + \alpha_k\gamma_k + \alpha_k^2 \sum_{i=1}^{k-1} \beta_{i,k}\gamma_i}. \tag{2.38}$$

Letting $x_{k+1} = T_{L_k}(y_k)$ ensures that $\Phi_{k+1} \geq F(x_{k+1})$.

Before introducing our proposed method, let us also present a backtracking line-search strategy that will enable the convergence of the minimization process.[1] Since the true values of $L_{\hat{f}}$ and $\mu_{\hat{f}}$ are not known, and considering the typical applications [25], we prioritize: *i)* robustness to the imperfect initialization of the estimate of L at iteration $k = 0$; *ii)* the need to adjust the value of the estimates of $L_{\hat{f}}$. This is achieved by selecting the parameters $\eta_u > 1$ and $\eta_d \in]0, 1[$, which are used to increase and decrease the estimate of $L_{\hat{f}}$ across different iterations. Considering this choice of parameters η_u, η_d, despite the initialization of L_0, we can always write

$$L_k \leq L_{\max} \triangleq \max\{\eta_d L_0, \eta_u L_{\hat{f}}\}. \tag{2.39}$$

We conclude by presenting our proposed method in Algorithm 2.1.

[1] Note that several backtracking strategies have already been proposed in the literature (see for example [22, 24]).

Algorithm 2.1 Proposed Method

1: **Input** $x_0 \in \mathcal{R}^n$, $L_0 > 0$, $\mu_{\hat{f}}$, $\gamma_0 \in [0, \mu_{\hat{f}}[\cup[2\mu_{\hat{f}}, 3L_0 + \mu_{\hat{f}}]$, $\eta_u > 1$ and $\eta_d \in]0, 1[$.
2: **Set** $k = 0$, $i = 0$ and $v_0 = x_0$.
3: **while** $k \leq K_{\max}$[2] **do**
4: $\quad \hat{L}_i \leftarrow \eta_d L_k$
5: $\quad$ **while** True **do**
6: $\quad\quad \hat{\alpha}_i \leftarrow \frac{\mu_{\hat{f}} + \sum_{i=1}^{k-1} \beta_{i,k}\hat{\gamma}_i - \gamma_k + \sqrt{\left(\mu_{\hat{f}} + \sum_{i=1}^{k-1} \beta_{i,k}\hat{\gamma}_i - \gamma_k\right)^2 + 4\hat{L}_i\gamma_k}}{2\hat{L}_i}$
7: $\quad\quad \hat{\gamma}_{i+1} \leftarrow (1 - \hat{\alpha}_i)\gamma_k + \hat{\alpha}_i\left(\mu_{\hat{f}} + \sum_{i=1}^{k-1} \beta_{i,k}\hat{\gamma}_i\right)$
8: $\quad\quad \hat{y}_i \leftarrow \frac{\hat{\gamma}_{i+1}x_k + \hat{\alpha}_i\gamma_k v_k + \hat{\alpha}_i^2 \sum_{i=1}^{k-1} \beta_{i,k}\hat{\gamma}_i v_i}{\hat{\gamma}_{i+1} + \hat{\alpha}_i\gamma_k + \hat{\alpha}_i^2 \sum_{i=1}^{k-1} \beta_{i,k}\hat{\gamma}_i}$
9: $\quad\quad \hat{x}_{i+1} \leftarrow \operatorname{prox}_{\frac{1}{\hat{L}_i}\hat{g}}\left(\hat{y}_i - \frac{1}{\hat{L}_i}\nabla f(\hat{y}_i)\right)$
10: $\quad\quad \hat{v}_{i+1} \leftarrow \frac{1}{\hat{\gamma}_{i+1}}\left((1 - \hat{\alpha}_i)\gamma_k v_k + \hat{\alpha}_i\left(\mu_{\hat{f}}\hat{y}_i + \sum_{i=1}^{k-1} \beta_{i,k}\hat{\gamma}_i - \hat{L}_i\left(\hat{y}_i - \hat{x}_{i+1}\right)\right)\right)$
11: $\quad\quad$ **if** $F(\hat{x}_{i+1}) \leq m_{\hat{L}_i}(\hat{y}_i, \hat{x}_{i+1})$ **then**
12: $\quad\quad\quad$ Break from loop
13: $\quad\quad$ **else**
14: $\quad\quad\quad \hat{L}_{i+1} \leftarrow \eta_u \hat{L}_i$
15: $\quad\quad$ **end if**
16: $\quad\quad i \leftarrow i + 1$
17: $\quad$ **end while**
18: $\quad L_{k+1} \leftarrow \hat{L}_i$, $x_{k+1} \leftarrow \hat{x}_i$, $\alpha_k \leftarrow \hat{\alpha}_{i-1}$, $y_k \leftarrow \hat{y}_{i-1}$, $\gamma_{k+1} \leftarrow \hat{\gamma}_i$, $v_{k+1} \leftarrow \hat{v}_i$, $i \leftarrow 0$, $k \leftarrow k + 1$
19: **end while**
20: **Output** $x_k = 0$

Comparing our proposed method to FGM, we can observe (from lines 6 and 7 in Algorithm 2.1) the differences in computing the iterates α_k and γ_k. In our case, their values are also dependent on the memory terms that were used in devising the estimating sequences. The update of y_k is also different, and independent of $\mu_{\hat{f}}$. A major difference is the update for x_k, which is now done through a proximal gradient step. The last difference between the methods can be observed from the update of the iterates v_k, which now depend on the selected subgradient. Further, comparing between our proposed method to the one presented in [26, 27] for minimizing convex functions with composite structure, we can see that the major differences arise from making use of the additional memory terms. Observe that our proposed method reduces to FGM when $\tau = 0$ and $\psi_k(x) = 0, \forall k = 0, 1, \ldots$. Moreover, observe that our proposed method reduces to the method presented in [26, 27] when $\psi_k(x) = 0, \forall k = 0, 1, \ldots$. In this sense, our proposed method is a generalization of all the aforementioned estimating sequence methods.

3. Convergence Analysis. Considering the results established in Lemma 2.2, the convergence rate of the minimization process is controlled by the rate at which the terms $\{\lambda_k\}_k$ decrease and the rate at which the terms $\{\psi_k\}_k$ increase.

[2]Note that $K_{\max}$ denotes the maximum number of iterations.

THEOREM 3.1. *If we let* $\lambda_0 = 1$ *and* $\lambda_k = \prod_{i=0}^{k-1}(1-\alpha_i)$*, Algorithm 2.1 generates a sequence of points* $\{x_k\}_k$ *such that*

$$F(x_k) - F(x^*) \leq \lambda_k \left(F(x_0) - F(x^*) + \frac{\gamma_0}{2}||x_0 - x^*||^2 \right) - (1-\lambda_k)\psi_k(x). \tag{3.1}$$

Proof. Let us begin by setting $\Phi_0^* = F(x_0)$. Further, evaluating (2.10) for $k = 0$ and $x = x_0$ we have: $\Phi_0(x_0) = F(x_0) + \frac{\gamma_0}{2}||x_0 - v_0||^2$. Moreover, using the initialization $v_0 = x_0$ as suggested in Algorithm 2.1 we obtain $F(x_0) \leq \Phi_0^*$. Last, note that the proposed method is designed to ensure $F(x_k) \leq \Phi_k^*$, $\forall k = 1, 2, \ldots$. Applying the findings from Lemma 2.2 suffices to conclude the proof. ∎

Let us now establish the rate at which the terms $\{\lambda_k\}_k$ decrease.

LEMMA 3.2. *For all* $k \geq 0$*, Algorithm 2.1 guarantees that*

1. *If* $\gamma_0 \in [0, \mu_{\hat{f}}[$*, then*

$$\lambda_k \leq \frac{2\mu_{\hat{f}}}{L_k \left(e^{\frac{k+1}{2}\sqrt{\frac{\left(\mu_{\hat{f}} + \sum_{i=1}^{k-1}\beta_{i,k}\gamma_i\right)}{L_k}}} - e^{-\frac{k+1}{2}\sqrt{\frac{\left(\mu_{\hat{f}} + \sum_{i=1}^{k-1}\beta_{i,k}\gamma_i\right)}{L_k}}} \right)^2} \leq \frac{2}{(k+1)^2}. \tag{3.2}$$

2. *If* $\gamma_0 \in [2\mu_{\hat{f}}, 3L_0 + \mu_{\hat{f}}]$*, then*

$$\lambda_k \leq \frac{4\mu_{\hat{f}}}{(\gamma_0 - \mu_{\hat{f}}) \left(e^{\frac{k+1}{2}\sqrt{\frac{\left(\mu_{\hat{f}} + \sum_{i=1}^{k-1}\beta_{i,k}\gamma_i\right)}{L_k}}} - e^{-\frac{k+1}{2}\sqrt{\frac{\left(\mu_{\hat{f}} + \sum_{i=1}^{k-1}\beta_{i,k}\gamma_i\right)}{L_k}}} \right)^2}$$

$$\leq \frac{4L_k}{(\gamma_0 - \mu_{\hat{f}})(k+1)^2}. \tag{3.3}$$

Proof. Let $\gamma_0 \in [0, \mu_{\hat{f}}[\cup [2\mu_{\hat{f}}, 3L_0 + \mu_{\hat{f}}]$ and apply (2.14) to

$$\gamma_{k+1} - \left(\mu_{\hat{f}} + \sum_{i=1}^{k-1}\beta_{i,k}\gamma_i \right) = (1-\alpha_k)\gamma_k + \alpha_k \left(\mu_{\hat{f}} + \sum_{i=1}^{k-1}\beta_{i,k}\gamma_i \right) - \left(\mu_{\hat{f}} + \sum_{i=1}^{k-1}\beta_{i,k}\gamma_i \right). \tag{3.4}$$

Moreover, since $\lambda_0 = 1$, we can re-write (3.4) as

$$\gamma_{k+1} - \left(\mu_{\hat{f}} + \sum_{i=1}^{k-1}\beta_{i,k}\gamma_i \right), = (1-\alpha_k)\lambda_0 \left[\gamma_k - \left(\mu_{\hat{f}} + \sum_{i=1}^{k-1}\beta_{i,k}\gamma_i \right) \right]. \tag{3.5}$$

Substituting (2.14) into (3.5), results in

$$\gamma_{k+1} - \left(\mu_{\hat{f}} + \sum_{i=1}^{k-1}\beta_{i,k}\gamma_i \right) = \lambda_{k+1} \left[\gamma_0 - \left(\mu_{\hat{f}} + \sum_{i=1}^{k-1}\beta_{i,k}\gamma_i \right) \right]. \tag{3.6}$$

Next, we note that (2.2) and (2.35) are connected through α_k as follows

$$\alpha_k = 1 - \frac{\lambda_{k+1}}{\lambda_k} = \sqrt{\frac{\gamma_{k+1}}{L_k}} = \sqrt{\frac{\left(\mu_{\hat{f}} + \sum_{i=1}^{k-1}\beta_{i,k}\gamma_i\right)}{L_k} + \frac{\gamma_{k+1} - \left(\mu_{\hat{f}} + \sum_{i=1}^{k-1}\beta_{i,k}\gamma_i\right)}{L_k}}. \tag{3.7}$$

Moreover, replacing (3.6) in the RHS of (3.7), and making some manipulations yields

$$\frac{\lambda_k-\lambda_{k+1}}{\lambda_k\lambda_{k+1}}=\frac{1}{\sqrt{\lambda_{k+1}}}\sqrt{\frac{\left(\mu_{\hat{f}}+\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i\right)}{\lambda_{k+1}L_k}+\frac{\gamma_0-\left(\mu_{\hat{f}}+\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i\right)}{L_k}}. \tag{3.8}$$

Observe that LHS of (3.8) can be written as $\frac{1}{\lambda_{k+1}}-\frac{1}{\lambda_k}$. Replacing the relation for the difference of squares in the LHS of (3.8) results in

$$\left(\frac{1}{\sqrt{\lambda_{k+1}}}-\frac{1}{\sqrt{\lambda_k}}\right)\left(\frac{1}{\sqrt{\lambda_{k+1}}}+\frac{1}{\sqrt{\lambda_k}}\right)=\frac{1}{\sqrt{\lambda_{k+1}}}$$
$$\sqrt{\frac{\left(\mu_{\hat{f}}+\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i\right)}{\lambda_{k+1}L_k}+\frac{\gamma_0-\left(\mu_{\hat{f}}+\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i\right)}{L_k}}. \tag{3.9}$$

Observe that in Lemma 2.3 we define $\alpha_k\in[0,1]$. Moreover, based on (2.2) we can establish that λ_k are non-increasing in k. This allows for replacing $\frac{1}{\sqrt{\lambda_k}}$ in the LHS of (3.9) with $\frac{1}{\sqrt{\lambda_{k+1}}}$, which would have a bigger value. So, we obtain

(3.10)

$$\frac{2}{\sqrt{\lambda_{k+1}}}\left(\frac{1}{\sqrt{\lambda_{k+1}}}-\frac{1}{\sqrt{\lambda_k}}\right)\geq\frac{1}{\sqrt{\lambda_{k+1}}}\sqrt{\frac{\left(\mu_{\hat{f}}+\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i\right)}{\lambda_{k+1}L_k}+\frac{\gamma_0-\left(\mu_{\hat{f}}+\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i\right)}{L_k}}.$$

We can now observe that the convergence rate of the minimization process is dependent on the value of γ_0. We will prove convergence separately for $\gamma_0\in\mathcal{R}_1=[0,\mu_{\hat{f}}[$ and $\gamma_0\in\mathcal{R}_2=[2\mu_{\hat{f}},3L_k+\mu_{\hat{f}}]$. We start with $\gamma_0\in\mathcal{R}_1$ and introduce the following

$$\xi_{k,\mathcal{R}_1}\triangleq\sqrt{\frac{L_{\max}}{\left(\left(\mu_{\hat{f}}+\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i\right)-\gamma_0\right)\lambda_k}}. \tag{3.11}$$

Next, we can revise (3.10) as

$$\frac{2}{\sqrt{\lambda_{k+1}}}-\frac{2}{\sqrt{\lambda_k}}\geq\sqrt{\frac{\left(\mu_{\hat{f}}+\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i\right)-\gamma_0}{L_k}}$$
$$\sqrt{\frac{\mu_{\hat{f}}L_k}{L_k\lambda_{k+1}\left(\left(\mu_{\hat{f}}+\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i\right)-\gamma_0\right)}+1}. \tag{3.12}$$

Revising the LHS in (3.12) and multiplying by $\sqrt{\frac{L_{\max}}{\left(\mu_{\hat{f}}+\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i\right)-\gamma_0}}$, yields

$$\xi_{k+1,\mathcal{R}_1}-\xi_{k,\mathcal{R}_1}\geq\frac{1}{2}\sqrt{\frac{\left(\mu_{\hat{f}}+\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i\right)\xi_{k+1,\mathcal{R}_1}^2}{L_{\max}}+1}. \tag{3.13}$$

Next, we prove by induction that

$$\xi_{k,\mathcal{R}_1} \geq \frac{\sqrt{2}}{4\delta}\sqrt{\frac{L_k}{\mu_{\hat{f}}-\gamma_0}}\left[e^{(k+1)\delta}-e^{(k+1)\delta}\right], \tag{3.14}$$

where $\delta \triangleq \frac{1}{2}\sqrt{\frac{\left(\mu_{\hat{f}}+\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i\right)}{L_{\max}}}$. First, considering (3.11) at iteration $k=0$ and recalling that $\lambda_0=1$, yeids

$$\xi_{0,\mathcal{R}_1} = \sqrt{\frac{L_{\max}}{(\mu_{\hat{f}}+\gamma_{-1}-\gamma_0)\lambda_0}} = \sqrt{\frac{L_{\max}}{\mu_{\hat{f}}-\gamma_0}}, \tag{3.15}$$

Embedding (2.39) in (3.15), results in

$$\xi_{0,\mathcal{R}_1} \geq \frac{\sqrt{2}}{2}\sqrt{\frac{L_k}{\mu_{\hat{f}}-\gamma_0}}\left[e^{\sqrt{2}/2}-e^{-\sqrt{2}/2}\right] \geq \frac{\sqrt{2}}{4\delta}\sqrt{\frac{L_k}{\mu_{\hat{f}}-\gamma_0}}\left[e^{\delta}-e^{-\delta}\right]. \tag{3.16}$$

The last inequality in (3.16) holds true because the RHS increases together with δ, which is designed such that $\delta < \frac{\sqrt{2}}{2}$.

Now suppose that (3.14) holds true at step k, and prove the relation for step $k+1$ by contradiction. Let $\omega(t) \triangleq \frac{\sqrt{2}}{4\delta}\sqrt{\frac{L_k}{\mu_{\hat{f}}-\gamma_0}}\left[e^{(t+1)\delta}-e^{-(t+1)\delta}\right]$. Based on [4, Lemma 2.2.4] $\omega(t)$ is convex in t. So, we have

$$\omega(t) \leq \xi_{k,\mathcal{R}_1} \leq \xi_{k+1,\mathcal{R}_1} - \frac{1}{2}\sqrt{\frac{\left(\mu_{\hat{f}}+\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i\right)\xi_{k+1,\mathcal{R}_1}^2}{L_{\max}}-1}, \tag{3.17}$$

where the second inequality stems from (3.13). Moreover, suppose that $\xi_{k+1,\mathcal{R}_1} < \omega(t+1)$ and substitute the relation in (3.17). This yelds

$$\omega(t) < \omega(t+1) - \frac{1}{2}\sqrt{\frac{\left(\mu_{\hat{f}}+\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i\right)\xi_{k+1,\mathcal{R}_1}^2}{L_{\max}}-1}. \tag{3.18}$$

Applying the definition for δ, together with (3.14), results in

$$\begin{aligned}\omega(t) &\leq \omega(t+1) - \frac{1}{2}\sqrt{4\delta^2\left[\frac{\sqrt{2}}{4\delta}\sqrt{\frac{L_k}{\mu_{\hat{f}}-\gamma_0}}\left(e^{(t+2)\delta}-e^{-(t+2)\delta}\right)\right]^2-1} \\ &\leq \omega(t+1) - \frac{\sqrt{2}}{4}\sqrt{\frac{L_k}{\mu_{\hat{f}}-\gamma_0}}\left[e^{(t+2)\delta}+e^{-(t+2)\delta}\right] \\ &= \omega(t+1) + \omega'(t+1)\left(t-(t+1)\right) \leq \omega(t).\end{aligned} \tag{3.19}$$

The last inequality is obtained based on the supporting hyperplane theorem of convex functions. At this point, we highlight the contradiction with the earlier assumption, i.e., $\xi_{k+1,\mathcal{R}_1} < \omega(t+1)$. So, it must be true that (3.14) holds for all iterations $k=0,1,\ldots$.

We can now prove (3.2). Considering (3.11), we have

$$\lambda_k = \frac{L_{\max}}{\xi^2_{k+1,\mathcal{R}_1}\left(\left(\mu_{\hat{f}} + \sum_{i=1}^{k-1}\beta_{i,k}\gamma_i\right) - \gamma_0\right)}. \tag{3.20}$$

Substituting (3.14) into (3.20), yields

$$\lambda_k \leq \frac{(4\delta)^2 L_{\max}}{2L_k\left[e^{(k+1)\delta} - e^{(k+1)\delta}\right]^2}. \tag{3.21}$$

The first inequality in (3.2) is obtained by replacing the definition of δ in (3.21). The second inequality in (3.2) can be proved as follows. First, let us define the following abbreviation

$$\mathcal{A}_k \triangleq \left(e^{\frac{k+1}{2}\sqrt{\frac{\left(\mu_{\hat{f}}+\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i\right)}{L_k}}} - e^{-\frac{k+1}{2}\sqrt{\frac{\left(\mu_{\hat{f}}+\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i\right)}{L_k}}}\right)^2 \tag{3.22}$$

Now, consider

$$\mathcal{A}_k = e^{(k+1)\sqrt{\frac{\left(\mu_{\hat{f}}+\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i\right)}{L_k}}} - e^{-(k+1)\sqrt{\frac{\left(\mu_{\hat{f}}+\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i\right)}{L_k}}} - 2. \tag{3.23}$$

Applying the definition of the hyperbolic cosine function in (3.23), yields

$$\mathcal{A}_k = 2\cosh\left(\sqrt{\frac{\left(\mu_{\hat{f}} + \sum_{i=1}^{k-1}\beta_{i,k}\gamma_i\right)}{L_k}}\,(k+1) - 2\right). \tag{3.24}$$

Taking the Taylor expansion of $\cosh(\cdot)$, yields

$$\mathcal{A}_k = -2 + 2 + 2\frac{\left(\mu_{\hat{f}} + \sum_{i=1}^{k-1}\beta_{i,k}\gamma_i\right)(k+1)^2}{2L_k} + 2\frac{\left(\mu_{\hat{f}} + \sum_{i=1}^{k-1}\beta_{i,k}\gamma_i\right)^2 (k+1)^4}{4!{L_k}^2} + \ldots. \tag{3.25}$$

Discarding the additional terms in (3.25) we obtain

$$\mathcal{A}_k \geq \frac{\left(\mu_{\hat{f}} + \sum_{i=1}^{k-1}\beta_{i,k}\gamma_i\right)}{L_k}(k+1)^2. \tag{3.26}$$

Replacing (3.26) in the denominator of the first inequality of (3.2) concludes the first part of the proof. The results for the case when $\gamma_0 \in \mathcal{R}_2$ can be established by following the analysis conducted for FGM in [4, Lemma 2.2.4]. The main update would need to be the addition of the term $\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i$ in the update for the sequence $\{\gamma_k\}_k$. ∎

Compared to [4, Lemma 2.2.4], Lemma 3.2 exhibits the following benefits: *i)*: Convergence of our proposed method is established also for the cases when the exact value of $L_{\hat{f}}$ is not known. *ii)* Our proposed method converges for a broader range of γ_0. Such a result is relevant because it enables the robustness of the initialization of our proposed method in the absence of the true value of $\mu_{\hat{f}}$.

Combining Lemma 3.2 and (1.14) with Theorem 3.1, yields the following accelerated convergence rate for the proposed method.

THEOREM 3.3. *Algorithm 2.1 generates a sequence of points such that*

1. *If* $\gamma_0 \in [0, \mu_{\hat{f}}[$, *then*

(3.27)
$$F(x_k) - F(x^*) \leq \frac{\mu_{\hat{f}}(L_0+\gamma_0)||x_0 - x^*||^2}{L_k\left(e^{\frac{k+1}{2}\sqrt{\frac{\mu_{\hat{f}}+\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i}{L_k}}} - e^{-\frac{k+1}{2}\sqrt{\frac{\mu_{\hat{f}}+\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i}{L_k}}}\right)^2}$$

2. *If* $\gamma_0 \in [2\mu_{\hat{f}}, 3L_0 + \mu_{\hat{f}}]$, *then*

(3.28)
$$F(x_k) - F(x^*) \leq \frac{2\mu_{\hat{f}}(L_0+\gamma_0)||x_0 - x^*||^2}{(\gamma_0-\mu_{\hat{f}})\left(e^{\frac{k+1}{2}\sqrt{\frac{\mu_{\hat{f}}+\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i}{L_k}}} - e^{-\frac{k+1}{2}\sqrt{\frac{\mu_{\hat{f}}+\sum_{i=1}^{k-1}\beta_{i,k}\gamma_i}{L_k}}}\right)^2}$$

4. Numerical study. We now present the numerical performance of our proposed method and compare to the existing black-box benchmarks, specifically, AMGS and FISTA. We consider both quadratic and logistic loss functions. To simulate very ill-conditioned instances of our selected problems, we also use elastic net regularizer and select different values of the hyperparameters. Throughout all the tested instances, we demonstrate the efficiency of our proposed method when compared to the selected benchmarks. In our simulations, we make use of both synthetic and real-world datasets, the altter being chosen from the Library for Support Vector Machines [31]. Moreover, throughout our simulations, We find x^* by using CVX [32].

In our simulations, we choose the terms $\beta_{i,k} = \min\left(1, \frac{\mu}{\gamma_{k-1}}\right)$, for $i = k-1$. Depending on the selection of the terms γ_0, we will consider the following instances of our proposed method: *1)* We set $\gamma_0 = 0$, and refer to it as "Proposed 1"; *2)* We set $\gamma_0 = \mu_{\hat{f}}$, refer to it as "Proposed 2"; *3)* We set $\gamma_0 = 3L_0 + \mu_{\hat{f}}$, and refer to it as "Proposed 3". To estimate the value of the Lipschitz constant for AMGS and FISTA we make use of the line-search strategies introduced in the corresponding papers [22, 24]. Last, in all the computational examples shown below, we select the point x_0 at random and use it as a starting point for all the algorithms that are compared.

4.1. Minimizing the quadratic loss function. Let us begin with the following cost function

$$\underset{x\in\mathcal{R}^n}{\text{minimize}} \quad \frac{1}{2}\sum_{i=1}^{m}(a_i^T x - y_i)^2 + \frac{\tau_1}{2}||x||^2 + \tau_2||x||_1, \tag{4.1}$$

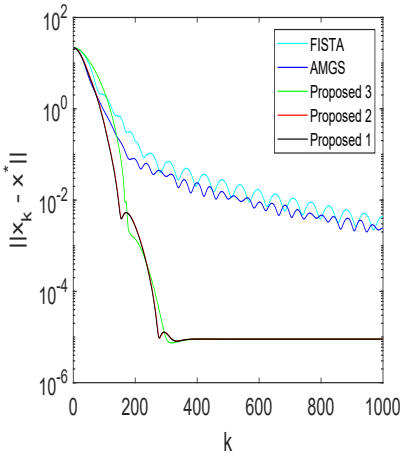
where $||\cdot||_1$ is the l_1 norm. The aim of the Section is to validate the theoretical results obtained above and demonstrate that such gains are also sustained when considering the practical deployments of the proposed method. For this purpose, we thoroughly evaluate the performance of the different benchmarks with respect to different values of the condition number of the problem. In our computational analysis, we also consider cases wherein the value of the Lipschitz constant is not known and needs to be estimated.

Let us start our evaluations by considering the cases where the Lipschitz constant and strong convexity parameters are known. This corresponds to the simplest case to analyze, and facilitates an unbiased evaluation of the efficiency of the methods that are being compared. For this setup, we will utilize simulated data which are generated by uniformly sampling m elements from the set $\{10^0, 10^{-1}, 10^{-2}, \dots, 10^{-\xi}\}$. These elements are then used to populate the diagonal of a sparse matrix $A \in \mathcal{R}^{m\times m}$. The other entries of A are set to 0. Considering the design of the matrix A, we have $L = 1$ and $\mu_f = 10^{-\xi}$. Thus, the condition number of the problem becomes $\kappa = 10^{\xi}$. The entries of $y \in \mathcal{R}^m$ are uniformly sampled from the interval $[0, 1]^n$. Last, the other simulation parameters are set to $m \in \{500, 1000\}$, $\xi \in \{3, 7\}$ and $\tau_1 = \tau_2 \in \{10^{-3}, 10^{-7}\}$.

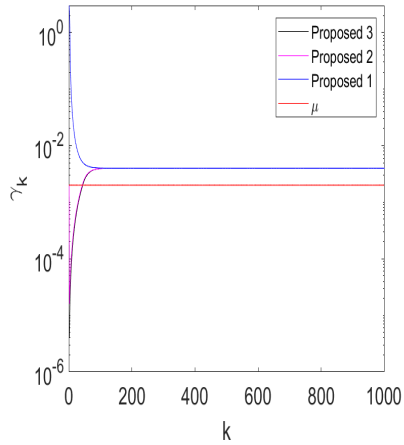
Our findings for the aforementioned simulation setup are summarized in Fig. 1. When compared to the selected benchmarks, we can observe that our proposed method is more efficient both in terms of the obtained distance to the optimal solution x^*, as well as in the number of iterations needed to converge to such solution. Another advantage of our proposed method is that it exhibits better monotonic properties. Moreover, observe that all the methods that are being evaluated are sensitive to the condition number of the problem. The higher the value of the condition number is, the more iterations the methods require to converge in the vicinity of x^*. Last, comparing between the selected instances of our proposed method, we can observe that they exhibit a commensurate degree of similarity, which is also clear based on our theoretical analysis. Nevertheless, we can see that the best performing instance is the one obtained when choosing $\gamma_0 = 0$.

Let us next consider the case where the true value of the Lipschitz contant is not known. For this purpose, we shall consider initial estimates of the Lipschitz constant that are 10 times higher and lower than the true value, i.e., $L_0 \in \{0.1L_{\hat{f}}, 10L_{\hat{f}}\}$. Following the recommendations presented in [33], for our line-search procedure we choose $\eta_u = 2$ and $\eta_d = 0.9$. We also assume the true value of the strong convexity parameter $\mu_{\hat{f}}$ is not known. Instead, we use the lower bound on the true value which can be controlled by the selection of the regularizer term in (4.1). In the following examples, we will use data from the "a1a" dataset, for which $A \in \mathcal{R}^{1605\times 123}$. For the considered dataset, the true values of the Lipschitz constant is $L_{\text{“a1a”}} = 10061$. The values of the regularizers are selected to be $\tau_1 = \tau_2 \in \{10^{-4}, 10^{-5}\}$, which ensures that the condition number of the problem $\kappa = \frac{L_{\hat{f}}}{\mu_{\hat{f}}}$ has a high value.

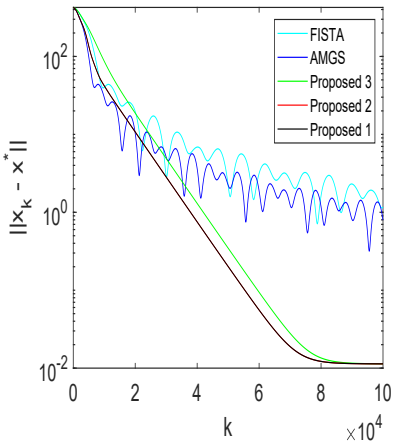
Our findings are summarized in Fig. 2. Therein, we can observed that our proposed method is more efficient than the selected benchmark. Similar to the results presented in Figure 1, the iterates produced from our proposed method exhibit better monotonic properties and have the smallest distance to the optimal solution. Moreover, accross all simulations, we can observe that our proposed method converges to x^* in a smaller number of iterations. Considering the result for different values of regularizers and Lipschitz constant estimates, we can observe the robustness of our proposed method and AMGS to the imperfect selection of L_0. A difference between these two methods, however, is that AMGS exhibits a higher per iteration complexity. Such results cannot be observed for FISTA, whose performance is very sensitive to the initialization of the Lipschitz contant estimate. This comes because the line-search strategy introduced for FISTA, does not allow for decreasing the estimate of the Lipschitz contant across iterates. Comparing between the different versions of our proposed method, we can observe that in most cases they are equally efficient. Nevertheless, the variant obtained when initializing $\gamma_0 = 0$ is preferred because it



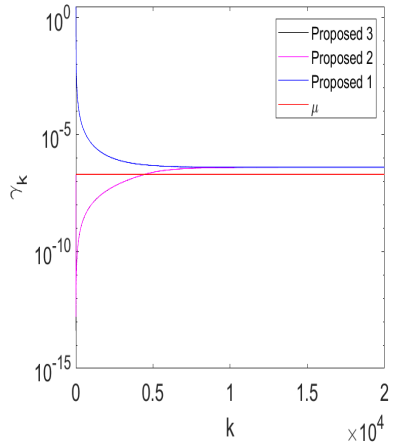
(a) Evaluating the distance to x^*, $m = 500$, $\kappa = 10^3$ and $\tau_1 = \tau_2 = 10^{-3}$.



(b) Convergence of the terms $\{\gamma_k\}_k$, $m = 500$, $\kappa = 10^3$ and $\tau_1 = \tau_2 = 10^{-3}$.



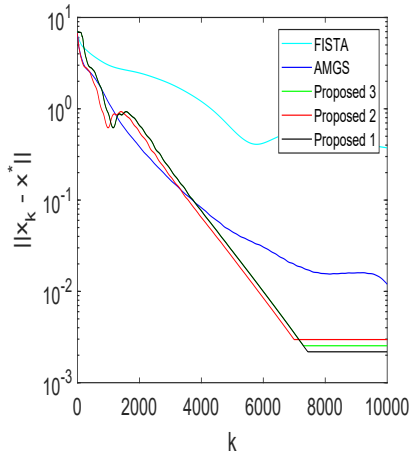
(c) Evaluating the distance to x^*, $m = 1000$, $\kappa = 10^7$ and $\tau_1 = \tau_2 = 10^{-7}$.



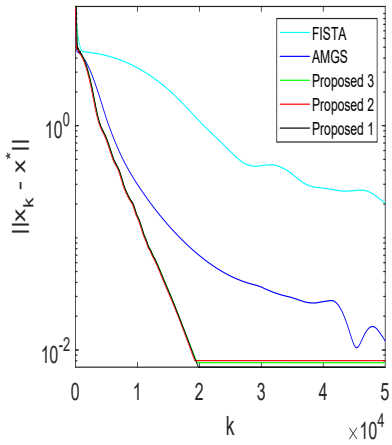
(d) Convergence of the terms $\{\gamma_k\}_k$, $m = 1000$, $\kappa = 10^7$ and $\tau_1 = \tau_2 = 10^{-7}$.

Fig. 1: Performance evaluation of our proposed method and the selected benchmarks on synthetic data. We consider quadratic objective function and elastic net regularizer.

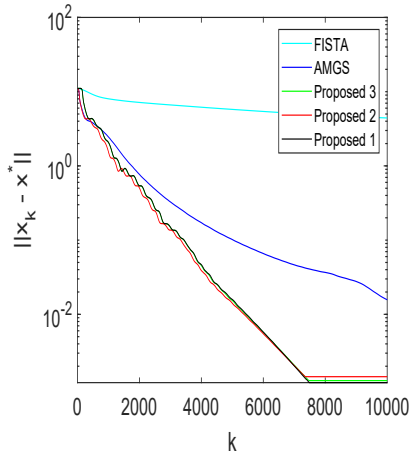
enables the robustness of the initialization of our proposed method with respect to the imperfect knowledge of $\mu_{\hat{f}}$.



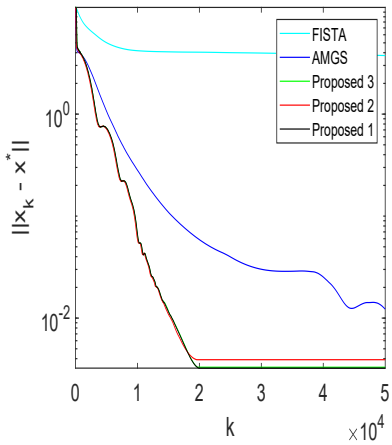
(a) Evaluating the distance to x^* for "a1a" dataset, $L_0 = 0.1L_{\text{"a1a"}}$ and $\tau_1 = \tau_2 = 10^{-4}$.



(b) Evaluating the distance to x^* for "a1a" dataset, $L_0 = 0.1L_{\text{"a1a"}}$ and $\tau_1 = \tau_2 = 10^{-5}$.



(c) Evaluating the distance to x^* for "a1a" dataset, $L_0 = 10L_{\text{"a1a"}}$ and $\tau_1 = \tau_2 = 10^{-4}$.



(d) Evaluating the distance to x^* for "a1a" dataset, $L_0 = 10L_{\text{"a1a"}}$ and $\tau_1 = \tau_2 = 10^{-5}$.

Fig. 2: Performance evaluation of our proposed method and the selected benchmarks on the "a1a" dataset. We consider quadratic objective function and elastic net regularizer, and assume that the true value of $L_{\hat{f}}$ is not known.

4.2. Minimizing the logistic loss function. We also test the performance of our algorithm and selected benchmarks in minimizing the following function.

$$\underset{x\in\mathcal{R}^n}{\text{minimize}} \quad \frac{1}{m}\sum_{i=1}^{m}\log\left(1+\mathrm{e}^{-b_i x a_i}\right)+\frac{\tau_1}{2}||x||^2+\tau_2||x||_1. \tag{4.2}$$

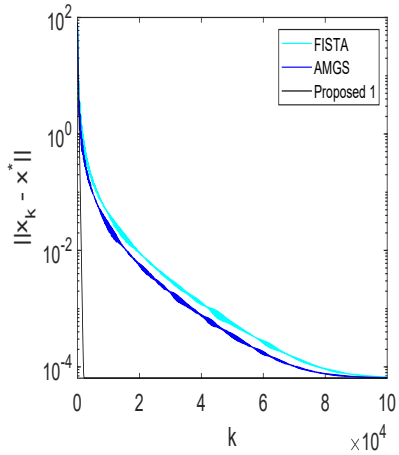
We also consider different datasets from the previous Section, namely "rcv1.binary", for which $A_{\text{"rcv1.binary"}} \in \mathcal{R}^{1000\times 2000}$, and a subset of "triazine", for which $A_{\text{"triazine"}} \in \mathcal{R}^{186\times 61}$. Moreover, in the earlier Section we observed that the convergence of FISTA is significantly affected by the selection of L_0, which happens because the line-search strategy proposed for FISTA does not allow for decreasing the estimate of the Lipschitz constant. Since in this paper the goal is to devise more efficient black-box algorithms, for the upcoming simulations we will assume that the true value of $L_{\hat{f}}$ is known. For the selected datasets, we have $L_{\text{"rcv1.binary"}} = 1.13$ and $L_{\text{"triazine"}} = 25.15$. Regarding the strong convexity parameter, we follow a similar approach as in the earlier examples and select its value to be the same as the l_2 regularizer term in (4.2), which are selected to be $\tau_1 = \tau_2 \in \{10^{-4}, 10^{-5}, 10^{-6}, 10^{-7}\}$. Last, since there is little performance difference between the different variants of our Proposed method, in the sequel we simulate only the first variant, namely Proposed 1. Our findings are depicted in Fig. 3, and from it we can clearly see that our proposed method significantly outperforms the selected benchmarks also in minimizing the regularized logistic loss function.

5. Conclusions and Discussion. A new class of generalized composite estimating sequences has been introduced for minimizing convex functions with composite structure with a non-smooth term. Using these newly introduced class of estimating sequences, a new accelerated black-box first method has been presented. The proposed method is endowed with an efficient backtracking line-search strategy, and exhibits an accelerated convergence rate even when the true value of the Lipschitz constant of the objective function is not known. The convergence results presented in the paper suggest that our proposed method exhibits such an accelerated convergence when $\gamma_0 \in [0, 3L + \mu_{\hat{f}}]$, i.e., the initialization of our proposed method is robust to the imperfect knowledge of the strong convexity parameter. From a computational viewpoint, our proposed method has been shown to outperform the existing benchmarks when tested in solving practical problems modeled by both simulated and real-world datasets.

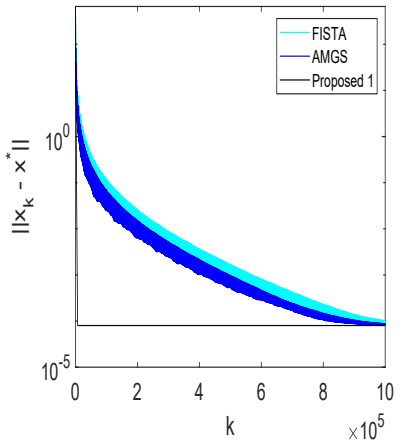
The results presented in this paper can be extended in multiple directions. First, it would be of interest to explore alternative structures for $\psi_k(x)$, which can be used for devising estimating sequences applicable to different optimization methods, e.g., higher-order methods, stochastic methods, non-convex methods etc. Another riveting research direction is related to investigating extensions of the framework devised herein in the context of the inexact oracle framework. Last, it would also be of interest to consider the impact of restarting in the practical performance of our proposed methods.

REFERENCES

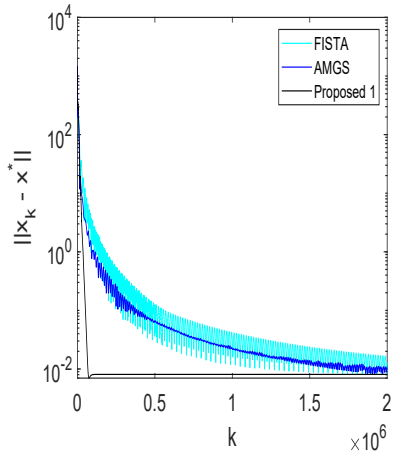
[1] Z. Allen-Zhu and L. Orecchia, "Linear coupling: An ultimate unification of gradient and mirror descent," *arXiv: 1407.1537*, Nov. 2016.
[2] S. Bubeck, Y. T. Lee and M. Singh, "A geometric alternative to Nesterov's accelerated gradient descent," *arXiv: 1506.08187*, Jun. 2015.
[3] Y. Nesterov, "A method for solving the convex programming problem with convergence rate $\mathcal{O}(1/k^2)$," *Doklady AN USSR*, vol. 269, pp. 543–547, 1983.



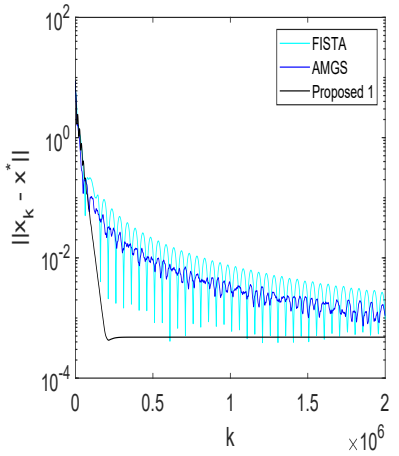
(a) Evaluating the distance to x^* for "rcv1.binary" dataset, $\tau_1 = \tau_2 = 10^{-4}$.



(b) Evaluating the distance to x^* for "rcv1.binary" dataset, $\tau_1 = \tau_2 = 10^{-5}$.



(c) Evaluating the distance to x^* for "triazine" dataset, $\tau_1 = \tau_2 = 10^{-6}$.



(d) Evaluating the distance to x^* for "triazine" dataset, $\tau_1 = \tau_2 = 10^{-7}$.

Fig. 3: Performance evaluation of our proposed method and the selected benchmarks on real data. We consider logistic objective function and elastic net regularizer.

[4] Y. Nesterov, *Lectures on convex optimization*. Springer, vol. 137, Dec. 2018.
[5] N. Flammarion and F. Bach, "From Averaging to Acceleration, There is Only a Step-size," in *Proc. Conference on Learning Theory*, Paris, France, July 2015, pp. 658–695.
[6] W. Su, S. Boyd and E. J. Candès, "A differential equation for modeling Nesterov's accelerated gradient method: Theory and insights," *Journal of Machine Learning Research*, vol. 17, no. 153, pp. 1–43, Jan. 2016.
[7] A. Wibisono, A. C. Wilson and M. I. Jordan, "A variational perspective on accelerated methods

in optimization," *Proceedings of the National Academy of Sciences*, vol. 113, no. 47, pp. E7351–E7358, Nov. 2016.
[8] Y. Drori and M. Teboulle, "Performance of first-order methods for smooth convex minimization: a novel approach," *Mathematical Programming*, vol. 145, no. 1, pp. 451–482, Jun. 2014.
[9] A. Taylor and Y. Drori, "An optimal gradient method for smooth strongly convex minimization," *Mathematical Programming*, Jun. 2022.
[10] A. d'Aspremont, D. Scieur, and A. Taylor, *Acceleration Methods*. Foundations and Trends® in Optimization, vol. 5, No. 1-2, pp 1–245 Dec. 2021.
[11] A. Nemirovsky and D. Yudin, *Problem Complexity and Method Efficiency in Optimization* Wiley, 1983.
[12] M. I. Florea and S. A. Vorobyov, "An accelerated composite gradient method for large-scale composite objective problems," *IEEE Transactions on Signal Processing*, vol. 67, no. 2, pp. 444–459, Jan. 2019.
[13] Y. Nesterov, "Universal gradient methods for convex optimization problems," *Mathematical Programming*, vol. 152, no. 1, pp. 381–404, Aug. 2015.
[14] Y. Nesterov, "Accelerating the cubic regularization of Newton's method on convex problems," *Mathematical Programming*, vol. 112, no. 1, pp. 159–181, Mar. 2008.
[15] Y. Nesterov, "Inexact high-order proximal-point methods with auxiliary search procedure," *SIAM Journal on Optimization*, vol. 31, no. 4, pp. 2807–2828, Nov. 2021.
[16] D. Jakovetić, J. Xavier and J. M. F. Moura, "Fast distributed gradient methods," *IEEE Transactions on Automatic Control*, vol. 59, no. 5, pp. 1131–1146, Jan. 2014.
[17] S. Ghadimi and G. Lan, "Accelerated gradient methods for nonconvex nonlinear and stochastic programming," *Mathematical Programming*, vol. 156, no. 1-2, pp. 59–99, Mar. 2016.
[18] A. Kulunchakov and J. Mairal, "Estimate Sequences for stochastic composite optimization: Variance reduction, acceleration, and robustness to noise," *Journal of Machine Learning Research*, vol. 21, no. 155, pp. 1–52, Jul. 2020.
[19] K. Ahn and S. Sra, "From Nesterov's estimate sequence to Riemannian acceleration," in *Proc. Conference on Learning Theory*, Graz, Austria, Jul. 2020, pp. 88–118.
[20] B. Li, M. Coutiño, G. B. Giannakis, "Revisit of estimate sequence for accelerated gradient methods," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*, Barcelona, Spain, May. 2020, pp. 3602–3606.
[21] M. Baes, "Estimate sequence methods: Extensions and approximations," *Institute for Operations Research, ETH, Zürich, Switzerland*, Aug. 2009.
[22] Y. Nesterov, "Gradient methods for minimizing composite objective function," *Mathematical Programming*, vol. 140, no. 1, pp. 125–161, Aug. 2013.
[23] Y. Nesterov, "Subgradient methods for huge-scale optimization problems," *Mathematical Programming*, vol. 146, no. 1, pp. 275–297, Aug. 2014.
[24] A. Beck and M. Teboulle, "A fast iterative shrinkage-thresholding algorithm for linear inverse problems," *SIAM Journal on Imaging Sciences*, vol. 2, no. 1, pp. 183–202, Mar. 2009.
[25] M. I. Florea and S. A. Vorobyov, "A generalized accelerated composite gradient method: Uniting Nesterov's fast gradient method and FISTA," *IEEE Transactions on Signal Processing*, vol. 68, pp. 3033–3048, Jul. 2020.
[26] E. Dosti, S. A. Vorobyov and T. Charalambous, "A new accelerated gradient-based estimating sequence technique for solving large-scale optimization problems with composite structure," in *IEEE Conference on Decision and Control*, Cancun, Mexico, Dec. 2022, pp. 7516–7521.
[27] E. Dosti, S. A. Vorobyov and T. Charalambous, "A new class of composite objective multistep estimating sequence techniques," *Signal Processing*, vol. 206, pp. 108889, May. 2023.
[28] N. Parikh, S. Boyd, "Proximal Algorithms," *Foundations and Trends in Optimization*, vol. 1, no. 3, pp. 127–239, Jan. 2014.
[29] E. Dosti, S. A. Vorobyov and T. Charalambous, "Embedding a heavy-ball type of momentum into the estimating sequences," in *IEEE International Symposium on Information Theory*, Helsinki, Finland, Jun. 2022, pp. 1506-1511.
[30] E. Dosti, S. A. Vorobyov, and T. Charalambous, "A new class of composite objective multi-step estimating sequence techniques," *arXiv:2111.06763*, Nov. 12, 2021.
[31] C. C. Chang and C. J. Lin, "LIBSVM: A library for support vector machines," *ACM Transactions on Intelligent Systems and Technology*, vol. 2, no. 3, pp. 1–27, May. 2011.
[32] M. Grant, S. Boyd and Y. Ye, "CVX: Matlab software for disciplined convex programming (web page and software)," 2009.
[33] S. R. Becker, E. J. Candès and M. Grant, "Templates for convex cone problems with applications to sparse signal recovery," *Mathematical Programming: Computation*, vol. 3, no. 3, pp. 165, Sep. 2011.