On Lower and Upper Bounds in Smooth and Strongly Convex Optimization

Yossi Arjevani
Weizmann Institute of Science
Rehovot 7610001, Israel
yossi.arjevani@weizmann.ac.il

Shai Shalev-Shwartz
The Hebrew University
Givat Ram, Jerusalem 9190401, Israel
shais@cs.huji.ac.il

Ohad Shamir
Weizmann Institute of Science
Rehovot 7610001, Israel
ohad.shamir@weizmann.ac.il

Abstract

We develop a novel framework to study smooth and strongly convex optimization algorithms. Focusing on quadratic functions we are able to examine optimization algorithms as a recursive application of linear operators. This, in turn, reveals a powerful connection between a class of optimization algorithms and the analytic theory of polynomials whereby new lower and upper bounds are derived. Whereas existing lower bounds for this setting are only valid when the dimensionality scales with the number of iterations, our lower bound holds in the natural regime where the dimensionality is fixed. Lastly, expressing it as an optimal solution for the corresponding optimization problem over polynomials, as formulated by our framework, we present a novel systematic derivation of Nesterov's well-known Accelerated Gradient Descent method. This rather natural interpretation of AGD contrasts with earlier ones which lacked a simple, yet solid, motivation.

1 Introduction

In the field of mathematical optimization one is interested in efficiently solving a minimization problem of the form

$$\min_{\mathbf{x}\in X} f(\mathbf{x}), \tag{1}$$

where the *objective function* f is some real-valued function defined over the *constraints set* X. Many core problems in the field of Computer Science, Economic, and Operations Research can be readily expressed in this form, rendering this minimization problem far-reaching. That being said, in its full generality this problem is just too hard to solve or even to approximate. As a consequence, various structural assumptions on the objective function and the constraints set, along with better-suited optimization algorithms, have been proposed so as to make this problem viable.

One such case is smooth and strongly convex functions over some d-dimensional Euclidean space. Formally, we consider continuously differentiable $f : \mathbb{R}^d \to \mathbb{R}$ which are L*-smooth*, i.e.,

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

and μ*-strongly convex*, that is,

$$f(\mathbf{y}) \ge f(\mathbf{x}) + \langle \mathbf{y} - \mathbf{x}, \nabla f(\mathbf{x})\rangle + \frac{\mu}{2}\|\mathbf{y} - \mathbf{x}\|^2, \quad \forall \mathbf{x}, \mathbf{y} \in \mathbb{R}^d$$

A wide range of applications together with very efficient solvers have made this family of problems very important. Naturally, an interesting question arises: how fast can these kind of problems be solved? better said, what is the computational complexity of minimizing smooth and strongly-convex functions to a given degree of accuracy?[1] Prior to answering these, otherwise ill-defined, questions, one must first address the exact nature of the underlying computational model.

Although being a widely accepted computational model in the theoretical computer sciences, the Turing Machine Model presents many obstacles when analyzing optimization algorithms. In their seminal work, [15] evaded some of these difficulties by proposing the *black box computational model*, according to which information regarding the objective function is acquired iteratively by querying an *oracle*. This model does not impose any computational resource constraints[2]. Nemirovsky and Yudin showed that for any optimization algorithm which employs a first-order oracle, i.e. receives $(f(\mathbf{x}), \nabla f(\mathbf{x}))$ upon querying at a point $\mathbf{x} \in \mathbb{R}^d$, there exists an L-smooth μ-strongly convex quadratic function $f : \mathbb{R}^d \rightarrow \mathbb{R}$, such that for any $\epsilon > 0$ the number of oracle calls needed for obtaining an ϵ-*optimal* solution $\tilde{\mathbf{x}}$, i.e.,

$$f(\tilde{\mathbf{x}}) < \min_{\mathbf{x}\in\mathbb{R}^d} f(\mathbf{x}) + \epsilon, \tag{2}$$

must satisfy

$$\#\text{ Oracle Calls} \geq \tilde{\Omega}\left(\min\left\{d, \sqrt{\kappa}\ln(1/\epsilon\right\}\right), \tag{3}$$

where $\kappa := L/\mu$ denotes the so-called *condition number*.

The result of Nemirovsky and Yudin can be seen as the starting point of the present paper. The restricted validity of this lower bound to the first $\mathcal{O}(d)$ iterations is not a mere artifact of the analysis. Indeed, from an information point of view, a minimizer of any convex quadratic function can be found using no more than $\mathcal{O}(d)$ first-order queries. Noticing that this bound is attained by the Conjugate Gradient Descent method (CGD, see [18]), it seems that one cannot get a non-trivial lower bound once the number of queries exceeds the dimension d. Moreover, a similar situation can be shown to occur for more general classes of convex functions. However, the known algorithms which attain such behavior (such as CGD and center-of-gravity, e.g., [14]) require computationally intensive iterations, and are quite different than many common algorithms used for large-scale optimization problems, such as gradient descent and its variants. Thus, to capture the attainable performance of such algorithms, we must make additional assumptions on their structure. The following simple observation seems to be particularity true in large-scale scenarios:

When applied on quadratic functions, the update rules of many optimization algorithms reduce to a recursive application of some linear transformation which depends, possibly randomly, on the previous p query points.

Indeed, the update rule of CGD for quadratic functions is *non-stationary*, i.e. uses a different linear transformation at each iteration, as opposed to other optimization algorithms which utilize less complex update rules such as: stationary updates rule, e.g., Gradient Descent, Accelerated Gradient Descent (see scheme 2.2.11 in [17]), Newton's method, The Heavy Ball method [18], and in the field of machine learning: SDCA [22], SVRG [8] and SAG [20]; cyclic update rules, e.g,. Cyclic Coordinate Descent; and piecewise-stationary update rules, e.g., Accelerated SDCA [21]. Inspired by this observation, in the present work we explore the boundaries of optimization algorithms which admit stationary update rules. We call such algorithms p-Stationary Canonical Linear Iterative optimization algorithms (abbr. p-SCLI), where p designates the number of previous points which are required to generate new points. The quantity p can be interpreted as a limit on the amount of memory at the algorithm's disposal.

[1] Natural as these questions might look today, matters were quite different only few decades ago. In his book 'Introduction to Optimization' which dates back to 87', Polyak B.T devotes a whole section as to: 'Why Are Convergence Theorems Necessary?' (See section 1.6.2 in [18]).

[2] In a sense, this model is dual to the Turing Machine model where all the information regarding the parameters of the problem is available prior to the execution of the algorithm, but the computational resources are limited in time and space.

Similar to the analysis of power iteration methods, the convergence properties of such algorithms are intimately related to the eigenvalues of the corresponding linear transformation. Concretely, as the convergence rate of a recursive application of a linear transformation is essentially characterized by its largest magnitude eigenvalue, the asymptotic convergence rate of p-SCLI algorithms can be bounded from above and from below by analyzing the spectrum of the corresponding linear transformation. At this point we would like to emphasize that the technique of linearizing iterative procedures and analyzing their convergence behavior accordingly, which dates back to the pioneering work of the Russian mathematician Lyapunov, has been successfully applied in the field of mathematical optimization many times, e.g., [18] and more recently [9]. However, whereas previous works were primarily concerned with deriving upper bounds on the magnitude of the corresponding eigenvalues, in this work our reference point is lower bounds.

As eigenvalues are merely roots of characteristic polynomials[3], our approach involves establishing a lower bound on the maximal modulus (absolute value) of the roots of polynomials. Clearly, in order to find a meaningful lower bound, one must first find a condition which is satisfied by all characteristic polynomials that correspond to p-SCLIs. We show that such condition does exist by proving that characteristic polynomials of consistent p-SCLIs, which correctly minimize the function at hand, must have a specific evaluation at $\lambda = 1$. This in turn allows us to analyze the convergence rate purely in terms of the analytic theory of polynomials, namely,

$$\textbf{Find} \quad \min\{\rho(q(z)) \mid q(z) \text{ is a real monic polynomial of degree } p \text{ and } q(1) = r\} \tag{4}$$

where $r \in \mathbb{R}$ and $\rho(q(z))$ denotes the maximum modulus over all roots of $q(z)$. Although a vast range of techniques have been developed for bounding the moduli of roots of polynomials (e.g., [10, 19, 12, 25, 13, 4]), to the best of our knowledge, few of them address lower bounds (e.g., [5]). The minimization problem (4) is also strongly connected with the question of bounding the spectral radius of 'generalized' companion matrices from below. Unfortunately, this topic too lacks an adequate coverage in the literature (e.g., [26, 27, 6, 7]). Consequently, we devote part of this work to establish new tools for tackling (4). It is noteworthy that these tools are developed by using elementary arguments. This sharply contrasts with previously proof techniques used for deriving lower bounds on the convergence rate of optimization algorithms which employed heavy machinery from the field of extremal polynomials, such as Chebyshev polynomials (e.g., [11]).

Based on the technique described above we present a novel lower bound on the convergence rate of p-SCLI optimization algorithms. More formally, we prove that any p-SCLI optimization algorithm over $\mathbb{R}^d$, whose iterations can be executed efficiently in a well defined sense, requires

$$\#\text{Oracle Calls} \geq \tilde{\Omega}\left(\sqrt[p]{\kappa}\ln(1/\epsilon)\right) \tag{5}$$

in order to obtain an ϵ-optimal solution, *regardless of the dimension of the problem*. This result partially complements the lower bound presented earlier in Inequality (3). More specifically, for $p = 1$, we show that the runtime of algorithms whose update rules do not depend on previous points (e.g. Gradient Descent) and can be computed efficiently scales linearly with the condition number. For $p = 2$, we get the optimal result for smooth and strongly convex functions. For $p > 2$, this lower bound is clearly weaker than the lower bound shown in (3) at the first d iterations. However, we show that it can be indeed attained by p-SCLI schemes, and surprisingly, some of them can be executed efficiently for certain classes of quadratic functions. Finally, we believe that a more refined analysis of problem (4) would show that this technique is powerful enough to meet the classical lower bound $\sqrt{\kappa}$ for any p, in the worst-case over all quadratic problems.

The last part of this work concerns a cornerstone in the field of mathematical optimization, i.e., Nesterov's well-known Accelerated Gradient Descent method (AGD). At the time the work of Nemirovsky and Yudin was published, it was known that Gradient Descent (GD) obtains an ϵ-optimal solution by issuing no more than

$$\mathcal{O}(\kappa\ln(1/\epsilon))$$

[3] In fact, we will use a polynomial matrix analogous of characteristic polynomials which will turns out to be more useful for our purposes.

first-order queries. The gap between this upper bound and the lower bound shown in (3) has intrigued many researchers in the field. Eventually, it was this line of inquiry that led to the discovery of AGD by Nesterov [16], a slight modification of the standard GD algorithm, whose iteration complexity is

$$\mathcal{O}\big(\sqrt{\kappa}\ln(1/\epsilon)\big).$$

Unfortunately, AGD lacks the strong geometrical intuition which accompanies many optimization algorithms, such as FGD and the Heavy Ball method. Primarily based on sophisticated algebraic manipulations, its proof strives for a more intuitive derivation (e.g., [3, 2, 24, 23, 1]). This downside has rendered the generalization of AGD to different optimization scenarios, such as constrained optimization problems, a highly non-trivial task which up to the present time does not admit a complete satisfactory solution. Surprisingly enough, by designing optimization algorithms whose characteristic polynomials are optimal with respect to a constrained version of (4), we have uncovered a novel simple derivation of AGD. This reformulation as an optimal solution for a constrained optimization problem over polynomials, shows that AGD and the Heavy Ball are essentially two sides of the same coin.

To summarize, our main contributions are the following:

- We define a class of algorithms (p-SCLI) in terms of linear operations on the last p iterations, and show that they subsume some of the most interesting algorithms used in practice.
- We prove that any p-SCLI optimization algorithm must use at least

  $$\tilde{\Omega}\left(\sqrt[p]{\kappa}\ln(1/\epsilon)\right)$$

  iterations in order to obtain an ϵ-optimal solution. As mentioned earlier, unlike existing lower bounds, our bound holds for every fixed dimensionality.
- We show that there exist matching p-SCLI optimization algorithms which attain the convergence rates stated above for all p. Alas, for $p \geq 3$, an expensive pre-calculation task renders these algorithms inefficient.
- As a result, we focus on a restricted subclass of p-SCLI optimization algorithms which can be executed efficiently. This yields a novel systematic derivation of Full Gradient Descent, Accelerated Gradient Descent, The Heavy-Ball method (and potentially other efficient optimization algorithms), all of which correspond to optimal solutions of a specific form of optimization problems on the moduli of polynomials' roots.
- We present new schemes which offer better utilization of second-order information by exploiting breaches in existing lower bounds. This leads to a new optimization algorithm which obtains a rate of $\sqrt[3]{\kappa}\ln(1/\epsilon)$ in the presence of large enough spectral gaps.

References

[1] Zeyuan Allen-Zhu and Lorenzo Orecchia. A novel, simple interpretation of nesterov's accelerated method as a combination of gradient and mirror descent. *arXiv preprint arXiv:1407.1537*, 2014.

[2] Michel Baes. Estimate sequence methods: extensions and approximations. 2009.

[3] Amir Beck and Marc Teboulle. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM Journal on Imaging Sciences*, 2(1):183–202, 2009.

[4] Harriet Fell. On the zeros of convex combinations of polynomials. *Pacific Journal of Mathematics*, 89(1):43–50, 1980.

[5] Nicholas J Higham and Françoise Tisseur. Bounds for eigenvalues of matrix polynomials. *Linear algebra and its applications*, 358(1):5–22, 2003.

[6] Bill G Horne. Lower bounds for the spectral radius of a matrix. *Linear algebra and its applications*, 263:261–273, 1997.

[7] Ting-Zhu Huang and Lin Wang. Improving bounds for eigenvalues of complex matrices using traces. *Linear Algebra and its Applications*, 426(2):841–854, 2007.

[8] Rie Johnson and Tong Zhang. Accelerating stochastic gradient descent using predictive variance reduction. In *Advances in Neural Information Processing Systems*, pages 315–323, 2013.

[9] Laurent Lessard, Benjamin Recht, and Andrew Packard. Analysis and design of optimization algorithms via integral quadratic constraints. *arXiv preprint arXiv:1408.3595*, 2014.

[10] Morris Marden. *Geometry of polynomials*. Number 3 in @. American Mathematical Soc., 1966.

[11] John C Mason and David C Handscomb. *Chebyshev polynomials*. CRC Press, 2002.

[12] Gradimir V Milovanovic, DS Mitrinovic, and Th M Rassias. Topics in polynomials. *Extremal Problems, Inequalities, Zeros, World Scientific, Singapore*, 1994.

[13] Gradimir V Milovanović and Themistocles M Rassias. Distribution of zeros and inequalities for zeros of algebraic polynomials. In *Functional equations and inequalities*, pages 171–204. Springer, 2000.

[14] Arkadi Nemirovski. Efficient methods in convex programming. 2005.

[15] AS Nemirovsky and DB Yudin. Problem complexity and method efficiency in optimization. 1983. *Willey-Interscience, New York*, 1983.

[16] Yurii Nesterov. *A method of solving a convex programming problem with convergence rate O (1/k2)*. @, 1983.

[17] Yurii Nesterov. *Introductory lectures on convex optimization*, volume 87. Springer Science & Business Media, 2004.

[18] Boris T Polyak. *Introduction to optimization*. Optimization Software New York, 1987.

[19] Qazi Ibadur Rahman and Gerhard Schmeisser. *Analytic theory of polynomials*. Number 26 in @. Oxford University Press, 2002.

[20] Nicolas Le Roux, Mark Schmidt, and Francis Bach. A stochastic gradient method with an exponential convergence rate for finite training sets. *arXiv preprint arXiv:1202.6258*, 2012.

[21] Shai Shalev-Shwartz and Tong Zhang. Accelerated proximal stochastic dual coordinate ascent for regularized loss minimization. *arXiv preprint arXiv:1309.2375*, 2013.

[22] Shai Shalev-Shwartz and Tong Zhang. Stochastic dual coordinate ascent methods for regularized loss. *The Journal of Machine Learning Research*, 14(1):567–599, 2013.

[23] Ilya Sutskever, James Martens, George Dahl, and Geoffrey Hinton. On the importance of initialization and momentum in deep learning. In *Proceedings of the 30th International Conference on Machine Learning (ICML-13)*, pages 1139–1147, 2013.

[24] Paul Tseng. On accelerated proximal gradient methods for convex-concave optimization. submitted to siam j. *J. Optim*, 2008.

[25] JL Walsh. On the location of the roots of certain types of polynomials. *Transactions of the American Mathematical Society*, 24(3):163–180, 1922.

[26] Henry Wolkowicz and George PH Styan. Bounds for eigenvalues using traces. *Linear Algebra and Its Applications*, 29:471–506, 1980.

[27] Qin Zhong and Ting-Zhu Huang. Bounds for the extreme eigenvalues using the trace and determinant. *Journal of Information and Computing Science*, 3(2):118–124, 2008.