Journal of Machine Learning Research 17 (2016) 1-51 Submitted 3/15; Revised 2/16; Published 2/16

On Lower and Upper Bounds in Smooth and Strongly Convex Optimization

Yossi Arjevani YOSSI.ARJEVANI@WEIZMANN.AC.IL
Department of Computer Science and Applied Mathematics
Weizmann Institute of Science
Rehovot 7610001, Israel

Shai Shalev-Shwartz SHAIS@CS.HUJI.AC.IL
School of Computer Science and Engineering
The Hebrew University
Givat Ram, Jerusalem 9190401, Israel

Ohad Shamir OHAD.SHAMIR@WEIZMANN.AC.IL
Department of Computer Science and Applied Mathematics
Weizmann Institute of Science
Rehovot 7610001, Israel

Editor: Mark Schmidt

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.

Keywords: smooth and strongly convex optimization, full gradient descent, accelerated gradient descent, heavy ball method

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[1]. Formally, we consider continuously differentiable $f : \mathbb{R}^d \to \mathbb{R}$ which are *L-smooth*, i.e.,

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

and *μ-strongly convex*, that is,

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

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?[2] 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, Nemirovsky and Yudin (1983) 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[3]. 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 $f : \mathbb{R}^d \to \mathbb{R}$, 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 $\kappa \triangleq L/\mu$ denotes the so-called *condition number*.

1. More generally, one may consider smooth and strongly convex functions over some Hilbert space.
2. 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 Polyak (1987)).
3. 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.

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 Polyak 1987), 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 the center-of-gravity method, e.g., Nemirovski 2005) 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. This can be made more solid using the following simple observation.

When applied on quadratic functions, the update rule of many optimization algorithms reduces to a recursive application of a 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 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, Newton's method (see Nesterov 2004), The Heavy Ball method Polyak (1987), SDCA (see Shalev-Shwartz and Zhang 2013) and SAG (see Roux et al. 2012); cyclic update rules, e.g,. SVRG (see Johnson and Zhang 2013); and piecewise-stationary update rules, e.g., Accelerated SDCA. 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 necessary to generate new points. The quantity p may be instructively interpreted as a limit on the amount of memory at the algorithm's disposal.

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. Specifically, 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. It should be noted 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., Polyak (1987) and more recently Lessard et al. (2014). 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[4], 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, i.e.,

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

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., Marden 1966; Rahman and Schmeisser 2002; Milovanovic et al. 1994; Walsh 1922; Milovanović and Rassias 2000; Fell 1980), to the best of our knowledge, few of them address lower bounds (see Higham and Tisseur 2003). 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 (see Wolkowicz and Styan 1980; Zhong and Huang 2008; Horne 1997; Huang and Wang 2007). 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., Mason and Handscomb 2002).

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, 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, some of which 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.

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

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). Prior to the work of Nemirovsky and Yudin, it was known that full Gradient Descent (FGD) obtains an ϵ-optimal solution by issuing no more than

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

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 (see Nesterov 1983), 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. Beck and Teboulle 2009; Baes 2009; Tseng 2008; Sutskever et al. 2013; Allen-Zhu and Orecchia 2014). 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, in order of appearance, 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), each of which corresponds to an optimal solution 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.

1.1 Notation

We denote scalars with lower case letters and vectors with bold face letters. We use $\mathbb{R}^{++}$ to denote the set of all positive real numbers. All functions in this paper are defined over Euclidean spaces equipped with the standard Euclidean norm and all matrix-norms are assumed to denote the spectral norm.

We denote a block-diagonal matrix whose blocks are $A_1, \ldots, A_k$ by the conventional direct sum notation, i.e., $\oplus_{i=1}^{k} A_k$. We devote a special operator symbol for scalar matrices $\mathrm{Diag}\,(a_1, \ldots, a_d) = \oplus_{i=1}^{d} a_i$. The spectrum of a square matrix A and its spectral radius, the maximum magnitude over its eigenvalues, are denoted by $\sigma(A)$ and $\rho(A)$, respectively. Recall that the eigenvalues of a square matrix $A \in \mathbb{R}^{d\times d}$ are exactly the roots of the characteristic polynomial which is defined as follows

$$\chi_A(\lambda) = \det(A - \lambda I_d),$$

where I_d denotes the identity matrix. Since polynomials in this paper have their origins as characteristic polynomials of some square matrices, by a slight abuse of notation, we will denote the roots of a polynomial $q(z)$ and its root radius, the maximum modulus over its roots, by $\sigma(q(z))$ and $\rho(q(z))$, respectively, as well.

The following notation for quadratic functions and matrices will be of frequent use,

$$\mathcal{S}^d(\Sigma) \triangleq \left\{ A \in \mathbb{R}^{d\times d} \middle| A \text{ is symmetric and } \sigma(A) \subseteq \Sigma \right\},$$
$$\mathcal{Q}^d(\Sigma) \triangleq \left\{ f_{A,\mathbf{b}}(\mathbf{x}) \;\middle|\; A \in \mathcal{S}^d(\Sigma)\,, \mathbf{b} \in \mathbb{R}^d \right\},$$

where Σ denotes a non-empty set of positive reals, and where $f_{A,\mathbf{b}}(\mathbf{x})$ denotes the following quadratic function

$$f_{A,\mathbf{b}}(\mathbf{x}) \triangleq \frac{1}{2}\mathbf{x}^\top A\mathbf{x} + \mathbf{b}^\top \mathbf{x}, \quad A \in \mathcal{S}^d(\Sigma)\,.$$

2. Framework

In the sequel we establish our framework for analyzing optimization algorithms for minimizing smooth and strongly convex functions. First, to motivate this technique, we show that the analysis of SDCA presented in Shalev-Shwartz and Zhang (2013) is tight by using a similar method. Next, we lay the foundations of the framework by generalizing and formalizing various aspects of the SDCA case. We then examine some popular optimization algorithms through this formulation. Apart from setting the boundaries for this work, this inspection gives rise to, otherwise subtle, distinctions between different optimization algorithms. Lastly, we discuss the computational complexity of p-SCLIs, as well as their convergence properties.

2.1 Case Study - Stochastic Dual Coordinate Ascent

We consider the optimization algorithm Stochastic Dual Coordinates Ascent (SDCA[5]) for solving Regularized Loss Minimization (RLM) problems (6), which are of great significance for the field of Machine Learning. It is shown that applying SDCA on quadratic loss functions allows one to reformulate it as a recursive application of linear transformations. The relative simplicity of such processes is then exploited to derive a lower bound on the convergence rate.

A smooth-RLM problem is an optimization task of the following form:

$$\min_{\mathbf{w}\in\mathbb{R}^d} P(\mathbf{w}) \triangleq \frac{1}{n}\sum_{i=1}^{n}\phi_i(\mathbf{w}^\top \mathbf{x}_i) + \frac{\lambda}{2}\left\|\mathbf{w}\right\|^2, \tag{6}$$

where ϕ_i are $1/\gamma$-smooth and convex, $\mathbf{x}_1,\ldots,\mathbf{x}_n$ are vectors in $\mathbb{R}^d$ and λ is a positive constant. For ease of presentation, we further assume that ϕ_i are non-negative, $\phi_i(0)\leq 1$ and $\|\mathbf{x}_i\|\leq 1$ for all i.

The optimization algorithm SDCA works by minimizing an equivalent optimization problem

$$\min_{\alpha\in\mathbb{R}^n} D(\boldsymbol{\alpha}) \triangleq \frac{1}{n}\sum_{i=1}^{n}\phi_i^\star(\alpha_i) + \frac{1}{2\lambda n^2}\left\|\sum_{i=1}^{n}\alpha_i\mathbf{x}_i\right\|^2,$$

where $\phi^\star$ denotes the Fenchel conjugate of ϕ, by repeatedly picking $z\sim\mathcal{U}([n])$ uniformly and minimizing $D(\boldsymbol{\alpha})$ over the z'th coordinate. The latter optimization problem is referred to as the *dual problem*, while the problem presented in (6) is called the *primal problem*. As shown in Shalev-Shwartz and Zhang (2013), it is possible to convert a high quality solution of the dual problem into a high quality solution of the primal problem. This allows one to bound from above the number of iterations required for obtaining a prescribed level of accuracy $\epsilon>0$ by

$$\tilde{\mathcal{O}}\left(\left(n+\frac{1}{\lambda\gamma}\right)\ln(1/\epsilon)\right).$$

We now show that this analysis is indeed tight. First, let us define the following 2-smooth functions:

$$\phi_i(y) = y^2, \quad i=1,\ldots,n$$

and let us define $\mathbf{x}_1=\mathbf{x}_2=\cdots=\mathbf{x}_n=\frac{1}{\sqrt{n}}\mathbb{1}$. This yields

$$D(\boldsymbol{\alpha}) = \frac{1}{2}\boldsymbol{\alpha}^\top\left(\frac{1}{2n}I+\frac{1}{\lambda n^2}\mathbb{1}\mathbb{1}^\top\right)\boldsymbol{\alpha}. \tag{7}$$

5. For a detailed analysis of SDCA, please refer to Shalev-Shwartz and Zhang 2013.

Clearly, the unique minimizer of $D(\boldsymbol{\alpha})$ is $\boldsymbol{\alpha}^* \triangleq 0$. Now, given $i \in [n]$ and $\boldsymbol{\alpha} \in \mathbb{R}^n$, it is easy to verify that

$$\underset{\alpha' \in \mathbb{R}}{\operatorname{argmin}} D(\alpha_1, \ldots, \alpha_{i-1}, \alpha', \alpha_{i+1}, \ldots, \alpha_n) = \frac{-2}{2+\lambda n} \sum_{j \neq i} \alpha_j. \tag{8}$$

Thus, the next test point $\boldsymbol{\alpha}^+$, generated by taking a step along the i'th coordinate, is a linear transformation of the previous point, i.e.,

$$\boldsymbol{\alpha}^+ = \left(I - \mathbf{e}_i \mathbf{u}_i^\top\right) \boldsymbol{\alpha}, \tag{9}$$

where

$$\mathbf{u}_i^\top \triangleq \left(\frac{2}{2+\lambda n}, \ldots, \frac{2}{2+\lambda n}, \underbrace{1}_{i\text{'s entry}}, \frac{2}{2+\lambda n}, \ldots, \frac{2}{2+\lambda n} \right).$$

Let $\boldsymbol{\alpha}^k,\ k = 1, \ldots, K$ denote the k'th test point. The sequence of points $(\boldsymbol{\alpha}^k)_{k=1}^K$ is randomly generated by minimizing $D(\boldsymbol{\alpha})$ over the z_i'th coordinate at the i'th iteration, where $z_1, z_2, \ldots, z_K \sim \mathcal{U}([n])$ is a sequence of K uniform distributed i.i.d random variables. Applying (9) over and over again starting from some initialization point $\boldsymbol{\alpha}^0$ we obtain

$$\boldsymbol{\alpha}^k = \left(I - \mathbf{e}_{z_K} \mathbf{u}_{z_K}^\top\right) \left(I - \mathbf{e}_{z_{K-1}} \mathbf{u}_{z_{K-1}}^\top\right) \cdots \left(I - \mathbf{e}_{z_1} \mathbf{u}_{z_1}^\top\right) \boldsymbol{\alpha}^0.$$

To compute $\mathbb{E}[\boldsymbol{\alpha}^K]$ note that by the i.i.d hypothesis and by the linearity of the expectation operator,

$$\begin{aligned}
\mathbb{E}\left[\boldsymbol{\alpha}^K\right] &= \mathbb{E}\left[\left(I - \mathbf{e}_{z_K} \mathbf{u}_{z_K}^\top\right) \left(I - \mathbf{e}_{z_{K-1}} \mathbf{u}_{z_{K-1}}^\top\right) \cdots \left(I - \mathbf{e}_{z_1} \mathbf{u}_{z_1}^\top\right) \boldsymbol{\alpha}^0\right] \\
&= \mathbb{E}\left[\left(I - \mathbf{e}_{z_K} \mathbf{u}_{z_K}^\top\right)\right] \mathbb{E}\left[\left(I - \mathbf{e}_{z_{K-1}} \mathbf{u}_{z_{K-1}}^\top\right)\right] \cdots \mathbb{E}\left[\left(I - \mathbf{e}_{z_1} \mathbf{u}_{z_1}^\top\right)\right] \boldsymbol{\alpha}^0 \\
&= \mathbb{E}\left[\left(I - \mathbf{e}_z \mathbf{u}_z^\top\right)\right]^K \boldsymbol{\alpha}^0.
\end{aligned} \tag{10}$$

The convergence rate of the latter is governed by the spectral radius of

$$E \triangleq \mathbb{E}\left[I - \mathbf{e}_z \mathbf{u}_z^\top\right].$$

A straightforward calculation shows that the eigenvalues of E, ordered by magnitude, are

$$\underbrace{1 - \frac{1}{2/\lambda + n}, \ldots, 1 - \frac{1}{2/\lambda + n}}_{n-1 \text{ times}}, 1 - \frac{2+\lambda}{2+\lambda n}. \tag{11}$$

By choosing $\boldsymbol{\alpha}^0$ to be the following normalized eigenvector which corresponds to the largest eigenvalue

$$\boldsymbol{\alpha}^0 = \left(\frac{1}{\sqrt{2}}, -\frac{1}{\sqrt{2}}, 0, \ldots, 0\right),$$

and plugging it into Equation (10), we can now bound from below the distance of $\mathbb{E}[\boldsymbol{\alpha}^K]$ to the optimal point $\boldsymbol{\alpha}^* = 0$,

$$\begin{aligned}
\left\|\mathbb{E}\left[\boldsymbol{\alpha}^K\right] - \boldsymbol{\alpha}^*\right\| &= \left\|\mathbb{E}\left[\left(I - \mathbf{e}_z\mathbf{u}_z^\top\right)\right]^K \boldsymbol{\alpha}^0\right\| \\
&= \left(1 - \frac{1}{2/\lambda + n}\right)^K \left\|\boldsymbol{\alpha}^0\right\| \\
&= \left(1 - \frac{2}{(4/\lambda + 2n - 1) + 1}\right)^K \\
&\geq \left(\exp\left(\frac{-1}{2/\lambda + n - 1}\right)\right)^K,
\end{aligned}$$

where the last inequality is due to the following inequality,

$$1 - \frac{2}{x+1} \geq \exp\left(\frac{-2}{x-1}\right), \quad \forall x \geq 1. \tag{12}$$

We see that the minimal number of iterations required for obtaining a solution whose distance from the $\boldsymbol{\alpha}^*$ is less than $\epsilon > 0$ must be greater than

$$(2/\lambda + n - 1)\ln(1/\epsilon),$$

thus showing that, up to logarithmic factors, the analysis of the convergence rate of SDCA is tight.

2.2 Definitions

In the sequel we introduce the framework of p-SCLI optimization algorithms which generalizes the analysis shown in the preceding section.

We denote the set of $d \times d$ symmetric matrices whose spectrum lies in $\Sigma \subseteq \mathbb{R}^{++}$ by $\mathcal{S}^d(\Sigma)$ and denote the following set of quadratic functions

$$f_{A,\mathbf{b}}(\mathbf{x}) \triangleq \frac{1}{2}\mathbf{x}^\top A\mathbf{x} + \mathbf{b}^\top\mathbf{x}, \quad A \in \mathcal{S}^d(\Sigma),$$

by $\mathcal{Q}^d(\Sigma)$. Note that since twice continuous differentiable functions $f(\mathbf{x})$ are L-smooth and μ-strongly convex if and only if

$$\sigma\big(\nabla^2(f(\mathbf{x}))\big) \subseteq [\mu, L] \subseteq \mathbb{R}^{++}, \quad \mathbf{x} \in \mathbb{R}^d,$$

we have that $\mathcal{Q}^d([\mu, L])$ comprises L-smooth μ-strongly convex quadratic functions. Thus, any optimization algorithm designed for minimizing smooth and strongly convex functions can be used to minimize functions in $\mathcal{Q}^d([\mu, L])$. The key observation here is that since the gradient of $f_{A,\mathbf{b}}(\mathbf{x})$ is linear in $\mathbf{x}$, when applied to quadratic functions, the update rules of many optimization algorithms also become linear in $\mathbf{x}$. This formalizes as follows.

Definition 1 (p-SCLI optimization algorithms) *An optimization algorithm $\mathcal{A}$ is called a p-stationary canonical linear iterative (abbr. p-SCLI) optimization algorithm over $\mathbb{R}^d$ if there exist $p+1$ mappings $C_0(X), C_1(X), \ldots, C_{p-1}(X), N(X)$ from $\mathbb{R}^{d\times d}$ to $\mathbb{R}^{d\times d}$-valued random variables, such that for any $f_{A,\mathbf{b}}(\mathbf{x}) \in \mathcal{Q}^d(\Sigma)$ the corresponding initialization and update rules take the following form:*

$$\mathbf{x}^0, \mathbf{x}^1, \ldots, \mathbf{x}^{p-1} \in \mathbb{R}^d \tag{13}$$

$$\mathbf{x}^k = \sum_{j=0}^{p-1} C_j(A)\mathbf{x}^{k-p+j} + N(A)\mathbf{b}, \quad k = p, p+1, \ldots \tag{14}$$

We further assume that in each iteration $C_j(A)$ and $N(A)$ are drawn independently of previous realizations[6]*, and that $\mathbb{E}C_i(A)$ are finite and simultaneously triangularizable*[7]*.*

Let us introduce a few more definitions and terminology which will be used throughout this paper. The number of previous points p by which new points are generated is called the *lifting factor*. The matrix-valued random variables $C_0(X), C_1(X), \ldots, C_{p-1}(X)$ and $N(X)$ are called *coefficient matrices* and *inversion matrix*, respectively. The term inversion matrix refers to the mapping $N(X)$, as well as to a concrete evaluation of it. It will be clear from the context which interpretation is being used. The same holds for coefficient matrices.

As demonstrated by the following definition, coefficients matrices of p-SCLIs can be equivalently described in terms of polynomial matrices[8]. This correspondence will soon play a pivotal role in the analysis of p-SCLIs.

Definition 2 *The characteristic polynomial of a given p-SCLI optimization algorithm $\mathcal{A}$ is defined by*

$$\mathcal{L}_{\mathcal{A}}(\lambda, X) \triangleq I_d\lambda^p - \sum_{j=0}^{p-1} \mathbb{E}C_j(X)\lambda^j, \tag{15}$$

where $C_j(X)$ denote the coefficient matrices. Moreover, given $X \in \mathbb{R}^{d\times d}$ we define the root radius of $\mathcal{L}_{\mathcal{A}}(\lambda, X)$ by

$$\rho_\lambda(\mathcal{L}_{\mathcal{A}}(\lambda, X)) = \rho(\det \mathcal{L}_{\mathcal{A}}(\lambda, X)) = \max\left\{|\lambda'| \;\middle|\; \det \mathcal{L}_{\mathcal{A}}(\lambda', X) = 0\right\}.$$

For the sake of brevity, we sometimes specify a given p-SCLI optimization algorithm $\mathcal{A}$ using an ordered pair of a characteristic polynomial and an inversion matrix as follows

$$\mathcal{A} \triangleq (\mathcal{L}_{\mathcal{A}}(\lambda, X), N(X)).$$

Furthermore, we may omit the subscript $\mathcal{A}$, when it is clear from the context.

6. We shall refer to this assumption as *stationarity*.
7. Intuitively, having this technical requirement is somewhat similar to assuming that the coefficients matrices commute (see Drazin et al. 1951 for a precise statement), and as such does not seem to restrict the scope of this work. Indeed, it is common to have $\mathbb{E}C_i(A)$ as polynomials in A or as diagonal matrices, in which case the assumption holds true.
8. For a detailed cover of polynomial matrices see Gohberg et al. (2009).

Lastly, note that nowhere in the definition of p-SCLIs did we assume that the optimization process converges to the minimizer of the function under consideration - an assumption which we refer to as *consistency*.

Definition 3 (Consistency of p-SCLI optimization algorithms) *A p-SCLI optimization algorithm $\mathcal{A}$ is said to be consistent with respect to a given $A \in \mathcal{S}^d(\Sigma)$ if for any $\mathbf{b} \in \mathbb{R}^d$, $\mathcal{A}$ converges to the minimizer of $f_{A,\mathbf{b}}(\mathbf{x})$, regardless of the initialization point. That is, for $\left(\mathbf{x}^k\right)_{k=1}^{\infty}$ as defined in (13,14) we have that*

$$\mathbf{x}^k \to -A^{-1}\mathbf{b},$$

for any $\mathbf{b} \in \mathbb{R}^d$. Furthermore, if $\mathcal{A}$ is consistent with respect to all $A \in \mathcal{S}^d(\Sigma)$, then we say that $\mathcal{A}$ is consistent with respect to $\mathcal{Q}^d(\Sigma)$.

2.3 Specifications for Some Popular Optimization Algorithms

Having defined the framework of p-SCLI optimization algorithms, a natural question now arises: how broad is the scope of this framework and what does characterize optimization algorithms which it applies to? Loosely speaking, any optimization algorithm whose update rules depend linearly on the first and the second order derivatives of the function under consideration is eligible for this framework. Instead of providing a precise characterization for such algorithms, we apply various popular optimization algorithms on a general quadratic function $f_{A,\mathbf{b}}(\mathbf{x}) \in \mathcal{Q}^d([\mu, L])$ and then express them as p-SCLI optimization algorithms.

Full Gradient Descent (FGD) is a 1-SCLI optimization algorithm with

$$\begin{aligned} \mathbf{x}^0 &\in \mathbb{R}^d, \\ \mathbf{x}^{k+1} &= \mathbf{x}^k - \beta\nabla f(\mathbf{x}^k) = \mathbf{x}^k - \beta(A\mathbf{x}^k + \mathbf{b}) = (I - \beta A)\mathbf{x}^k - \beta\mathbf{b}, \\ \beta &= \frac{2}{\mu + L}. \end{aligned}$$

See Nesterov (2004) for more details.

Newton method is a 0-SCLI optimization algorithm with

$$\begin{aligned} \mathbf{x}^0 &\in \mathbb{R}^d, \\ \mathbf{x}^{k+1} &= \mathbf{x}^k - (\nabla^2 f(\mathbf{x}^k))^{-1}\nabla f(\mathbf{x}^k) = \mathbf{x}^k - A^{-1}(A\mathbf{x}^k + \mathbf{b}) \\ &= (I - A^{-1}A)\mathbf{x}^k - A^{-1}\mathbf{b} = -A^{-1}\mathbf{b}. \end{aligned}$$

Note that Newton method can be also formulated as a degenerate p-SCLI for some $p \in \mathbb{N}$, whose coefficients matrices vanish. See Nesterov (2004) for more details.

The Heavy Ball Method is a 2-SCLI optimization algorithm with

$$\begin{aligned}\mathbf{x}^{k+1} &= \mathbf{x}^k - \alpha\nabla f(\mathbf{x}^k) + \beta(\mathbf{x}^k - \mathbf{x}^{k-1}) \\ &= \mathbf{x}^k - \alpha(A\mathbf{x}^k + \mathbf{b}) + \beta(\mathbf{x}^k - \mathbf{x}^{k-1}) \\ &= ((1+\beta)I - \alpha A)\,\mathbf{x}^k - \beta I\mathbf{x}^{k-1} - \alpha\mathbf{b}, \\ \alpha = \frac{4}{\left(\sqrt{L}+\sqrt{\mu}\right)^2}, &\quad \beta = \left(\frac{\sqrt{L}-\sqrt{\mu}}{\sqrt{L}+\sqrt{\mu}}\right)^2.\end{aligned}$$

See Polyak (1987) for more details.

Accelerated Gradient Descent (AGD) is a 2-SCLI optimization algorithm with

$$\begin{aligned}\mathbf{x}^0 &= \mathbf{y}^0 \in \mathbb{R}^d, \\ \mathbf{y}^{k+1} &= \mathbf{x}^k - \frac{1}{L}\nabla f(\mathbf{x}^k), \\ \mathbf{x}^{k+1} &= (1+\alpha)\,\mathbf{y}^{k+1} - \alpha\mathbf{y}^k, \\ \alpha &= \frac{\sqrt{L}-\sqrt{\mu}}{\sqrt{L}+\sqrt{\mu}},\end{aligned}$$

which can be rewritten as follows:

$$\begin{aligned}&\mathbf{x}^0 \in \mathbb{R}^d, \\ \mathbf{x}^{k+1} &= (1+\alpha)\left(\mathbf{x}^k - \frac{1}{L}\nabla f(\mathbf{x}^k)\right) - \alpha\left(\mathbf{x}^{k-1} - \frac{1}{L}\nabla f(\mathbf{x}^{k-1})\right) \\ &= (1+\alpha)\left(\mathbf{x}^k - \frac{1}{L}(A\mathbf{x}^k + \mathbf{b})\right) - \alpha\left(\mathbf{x}^{k-1} - \frac{1}{L}(A\mathbf{x}^{k-1} + \mathbf{b})\right) \\ &= (1+\alpha)\left(I - \frac{1}{L}A\right)\mathbf{x}^k - \alpha\left(I - \frac{1}{L}A\right)\mathbf{x}^{k-1} - \frac{1}{L}\mathbf{b}.\end{aligned}$$

Note that here we employ a stationary variant of AGD. See Nesterov (2004) for more details.

Stochastic Coordinate Descent (SCD) is a 1-SCLI optimization algorithm. This is a generalization of the example shown in Section 2.1. SCD acts by repeatedly minimizing a uniformly randomly drawn coordinate in each iteration. That is,

$$\begin{aligned}&\mathbf{x}^0 \in \mathbb{R}^d, \\ &\text{Pick } i \sim \mathcal{U}([d]) \text{ and set } \mathbf{x}^{k+1} = \left(I - \frac{1}{A_{i,i}}\mathbf{e}_i\mathbf{a}_{i,\star}^\top\right)\mathbf{x}^k - \frac{b_i}{A_{i,i}}\mathbf{e}_i,\end{aligned}$$

where $\mathbf{a}_{i,\star}^\top$ denotes the i'th row of A and $\mathbf{b} \triangleq (b_1, b_2, \ldots, b_d)$. Note that the expected update rule of this method is equivalent to the well-known Jacobi's iterative method.

We now describe some popular optimization algorithms which do not fit this framework, mainly because the stationarity requirement fails to hold. The extension of this framework to cyclic and piecewise stationary optimization algorithms is left to future work.

Conjugate Gradient Descent (CGD) can be expressed as a non-stationary iterative method

$$\mathbf{x}^{k+1} = ((1+\beta_k)I - \alpha_k A)\,\mathbf{x}^k - \beta_k I \mathbf{x}^{k-1} - \alpha_k \mathbf{b},$$

where α_k and β_k are computed at each iteration based on $\mathbf{x}^k, \mathbf{x}^{k-1}, A$ and b. Note the similarity of CGD and the heavy ball method. See Polyak (1987); Nemirovski (2005) for more details. In the context of this framework, CGD forms the 'most non-stationary' kind of method in that its coefficients α_k, β_k are highly dependent on time and the function at hand.

Stochastic Gradient Descent (SGD) A straightforward extension of the deterministic FGD. Specifically, let $(\Omega, \mathcal{F}, \mathcal{P})$ be a probability space and let $G(\mathbf{x}, \omega) : \mathbb{R}^d \times \Omega \to \mathbb{R}^d$ be an unbiased estimator of $\nabla f(\mathbf{x})$ for any $\mathbf{x}$. That is,

$$\mathbb{E}[G(\mathbf{x}, \omega)] = \nabla f(\mathbf{x}) = A\mathbf{x} + \mathbf{b}, \quad \mathbf{x} \in \mathbb{R}^d.$$

Equivalently, define $\mathbf{e}(\mathbf{x}, \omega) = G(\mathbf{x}, \omega) - (A\mathbf{x} + \mathbf{b})$ and assume $\mathbb{E}[\mathbf{e}(\mathbf{x}, \omega)] = 0,\ \mathbf{x} \in \mathbb{R}^d$. SGD may be defined using a suitable sequence of step sizes $(\gamma_i)_{i=1}^{\infty}$ as follows:

$$\begin{aligned} \text{Generate } \omega_k \text{ randomly and set } \mathbf{x}^{k+1} &= \mathbf{x}^k - \gamma_i G(\mathbf{x}^k, \omega_k) \\ &= (I - \gamma_i A)\,\mathbf{x}^k - \gamma_i \mathbf{b} - \gamma_i \mathbf{e}(\mathbf{x}, \omega). \end{aligned}$$

Clearly, some types of noise may not form a p-SCLI optimization algorithm. However, for some instances, e.g., quadratic learning problems, we have

$$\mathbf{e}(\mathbf{x}, \omega) = A_\omega \mathbf{x} + \mathbf{b}_\omega,$$

such that

$$\mathbb{E}[A_\omega] = 0, \quad \mathbb{E}[\mathbf{b}_\omega] = 0.$$

If, in addition, the step size is fixed then we get a 1-SCLI optimization algorithm. See Kushner and Yin (2003); Spall (2005); Nemirovski (2005) for more details.

2.4 Computational Complexity

The stationarity property of general p-SCLIs optimization algorithms implies that the computational cost of minimizing a given quadratic function $f_{A,\mathbf{b}}(\mathbf{x})$, assuming $\Theta(1)$ cost for all arithmetic operations, is

$$\#\text{ Iterations } \times \begin{cases} \text{Generating coefficient and inversion matrices randomly} \\ \qquad + \\ \text{Executing update rule (14) based on the previous } p \text{ points} \end{cases}$$

The computational cost of the execution of update rule (14) scales quadratically with d, the dimension of the problem, and linearly with p, the lifting factor. Thus, the running time of p-SCLIs is mainly affected by the iterations number and the computational cost of

randomly generating coefficient and inversion matrices each time. Notice that for deterministic p-SCLIs one can save running time by computing the coefficient and inversion matrices once, prior to the execution of the algorithm. Not surprisingly, but interesting nonetheless, there is a law of conservation which governs the total amount of computational cost invested in both factors: the more demanding is the task of randomly generating coefficient and inversion matrices, the less is the total number of iterations required for obtaining a given level of accuracy, and vice versa. Before we can make this statement more rigorous, we need to present a few more facts about p-SCLIs. For the time being, let us focus on the *iteration complexity*, i.e., the total number iterations, which forms our analogy for black box complexity.

The *iteration complexity* of a p-SCLI optimization algorithm $\mathcal{A}$ with respect to an accuracy level ϵ, initialization points $\mathcal{X}^0$ and a quadratic function $f_{A,\mathbf{b}}(\mathbf{x})$, symbolized by

$$\mathcal{IC}_{\mathcal{A}}\left(\epsilon, f_{A,\mathbf{b}}(\mathbf{x}), \mathcal{X}^0\right),$$

is defined to be the minimal number of iterations K such that

$$\left\|\mathbb{E}[\mathbf{x}^k - \mathbf{x}^*]\right\| < \epsilon, \quad \forall k \geq K,$$

where $\mathbf{x}^* = -A^{-1}\mathbf{b}$ is the minimizer of $f_{A,\mathbf{b}}(\mathbf{x})$, assuming $\mathcal{A}$ is initialized at $\mathcal{X}^0$. We would like to point out that although iteration complexity is usually measured through

$$\mathbb{E}\left\|\mathbf{x}^k - \mathbf{x}^*\right\|,$$

here we employ a different definition. We will discuss this issue shortly.

In addition to showing that the iteration complexity of p-SCLI algorithms scales logarithmically with $1/\epsilon$, the following theorem provides a characterization for the iteration complexity in terms of the root radius of the characteristic polynomial.

Theorem 4 *Let $\mathcal{A}$ be a p-SCLI optimization algorithm over $\mathbb{R}^d$ and let $f_{A,\mathbf{b}}(\mathbf{x}) \in \mathcal{Q}^d(\Sigma)$, $(\Sigma \subseteq \mathbb{R}^{++})$ be a quadratic function. Then, there exists $\mathcal{X}^0 \in (\mathbb{R}^d)^p$ such that*

$$\mathcal{IC}_{\mathcal{A}}\left(\epsilon, f_{A,\mathbf{b}}(\mathbf{x}), \mathcal{X}^0\right) = \tilde{\Omega}\left(\frac{\rho}{1-\rho}\ln(1/\epsilon)\right),$$

and for all $\mathcal{X}^0 \in (\mathbb{R}^d)^p$, it holds that

$$\mathcal{IC}_{\mathcal{A}}\left(\epsilon, f_{A,\mathbf{b}}(\mathbf{x}), \mathcal{X}^0\right) = \tilde{\mathcal{O}}\left(\frac{1}{1-\rho}\ln(1/\epsilon)\right),$$

where ρ denotes the root radius of the characteristic polynomial evaluated at $X = A$.

The full proof for this theorem is somewhat long and thus provided in Section C.1. Nevertheless, the intuition behind it is very simple and may be sketched as follows:

- First, we express update rule (14) as a single step rule by introducing new variables in some possibly higher-dimensional Euclidean space $(\mathbb{R}^d)^p$,

$$\mathbf{z}^0 = \left(\mathbf{x}^0, \mathbf{x}^1, \ldots, \mathbf{x}^{p-1}\right)^\top \in \mathbb{R}^{pd}, \quad \mathbf{z}^k = M(X)\mathbf{z}^{k-1} + UN(X)\mathbf{b}, \quad k = 1, 2, \ldots$$

  Recursively applying this rule and taking expectation w.r.t the coefficient matrices and the inversion matrix yields

$$\mathbb{E}\left[\mathbf{z}^k - \mathbf{z}^*\right] = \mathbb{E}[M]^k(\mathbf{z}^0 - \mathbf{z}^*).$$

- Then, to derive the lower bound, we use the Jordan form of $\mathbb{E}[M]$ to show that there exists some non-zero vector $\mathbf{r} \in (\mathbb{R}^d)^p$ such that if $\langle \mathbf{z}^0 - \mathbf{z}^*, \mathbf{r}\rangle \neq 0$, then $\|\mathbb{E}[M]^k(\mathbf{z}^0 - \mathbf{z}^*)\|$ is asymptotically bounded from below by some geometric sequence. The upper bound follows similarly.

- Finally, we express the bound on the convergence rate of $(\mathbf{z}^k)$ in terms of the original space.

Carefully inspecting the proof idea shown above reveals that the lower bound remains valid even in cases where the initialization points are drawn randomly. The only condition for this to hold is that the underlying distribution is reasonable, in the sense that it is absolutely continuous w.r.t. the Lebesgue measure, which implies that $\Pr[\langle \mathbf{z}^0 - \mathbf{z}^*, \mathbf{r}\rangle \neq 0] = 1$.

We remark that the constants in the asymptotic behavior above may depend on the quadratic function under consideration, and that the logarithmic terms depend on the distance of the initialization points from the minimizer, as well as the lifting factor and the spectrum of the Hessian. For the sake of clarity, we omit the dependency on these quantities.

There are two, rather subtle, issues regarding the definition of iteration complexity which we would like to address. First, observe that in many cases a given point $\tilde{\mathbf{x}} \in \mathbb{R}^d$ is said to be ϵ-optimal w.r.t some real function $f : \mathbb{R}^d \to \mathbb{R}$ if

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

However, here we employ a different measure for optimality. Fortunately, in our case either can be used without essentially affecting the iteration complexity. That is, although in general the gap between these two definitions can be made arbitrarily large, for L-smooth μ-strongly convex functions we have

$$\frac{\mu}{2}\left\|\mathbf{x} - \mathbf{x}^*\right\|^2 \leq f(\mathbf{x}) - f(\mathbf{x}^*) \leq \frac{L}{2}\left\|\mathbf{x} - \mathbf{x}^*\right\|^2.$$

Combining these two inequalities with the fact that the iteration complexity of p-SCLIs depends logarithmically on $1/\epsilon$ implies that in this very setting these two distances are interchangeable, up to logarithmic factors.

Secondly, here we measure the sub-optimality of the k'th iteration by $\left\|\mathbb{E}[\mathbf{x}^k - \mathbf{x}^*]\right\|$, whereas in many other stochastic settings it is common to derive upper and lower bounds on $\mathbb{E}\left[\left\|\mathbf{x}^k - \mathbf{x}^*\right\|\right]$. That being the case, by

$$\mathbb{E}\left[\left\|\mathbf{x}^k - \mathbf{x}^*\right\|^2\right] = \mathbb{E}\left[\left\|\mathbf{x}^k - \mathbb{E}\mathbf{x}^k\right\|^2\right] + \left\|\mathbb{E}\left[\mathbf{x}^k - \mathbf{x}^*\right]\right\|^2,$$

we see that if the variance of the k'th point is of the same order of magnitude as the norm of the expected distance from the optimal point, then both measures are equivalent. Consequently, our upper bounds imply upper bounds on $\mathbb{E}\left[\left\|\mathbf{x}^k - \mathbf{x}^*\right\|^2\right]$ for deterministic algorithms (where the variance term is zero), and our lower bounds imply lower bounds on $\mathbb{E}\left[\left\|\mathbf{x}^k - \mathbf{x}^*\right\|^2\right]$, for both deterministic and stochastic algorithms (since the variance is non-negative). We defer a more adequate treatment for this matter to future work.

3. Deriving Bounds for p-SCLI Algorithms

The goal of the following section is to show how the framework of p-SCLI optimization algorithms can be used to derive lower and upper bounds. Our presentation follows from the simplest setting to the most general one. First, we present a useful characterization of consistency (see Definition 3) of p-SCLIs using the characteristic polynomial. Next, we demonstrate the importance of consistency through a simplified one dimensional case. This line of argument is then generalized to any finite dimensional space and is used to explain the role of the inversion matrix. Finally, we conclude this section by providing a schematic description of this technique for the most general case which is used both in Section (4) to establish lower bounds on the convergence rate of p-SCLIs with diagonal inversion matrices, and in Section (5) to derive efficient p-SCLIs.

3.1 Consistency

Closely inspecting various specifications for p-SCLI optimization algorithms (see Section (2.3)) reveals that the coefficient matrices always sum up to $I + \mathbb{E}N(X)X$, where $N(X)$ denotes the inversion matrix. It turns out that this is not a mere coincidence, but an extremely useful characterization for consistency of p-SCLIs. To see why this condition must hold, suppose $\mathcal{A}$ is a deterministic p-SCLI algorithm over $\mathbb{R}^d$ whose coefficient matrices and inversion matrix are $C_0(X), \ldots, C_{p-1}(X)$ and $N(X)$, respectively, and suppose that $\mathcal{A}$ is consistent w.r.t some $A \in \mathcal{S}^d(\Sigma)$. Recall that every $p+1$ consecutive points generated by $\mathcal{A}$ are related by (14) as follows

$$\mathbf{x}^k = \sum_{j=0}^{p-1} C_j(A)\mathbf{x}^{k-p+j} + N(A)\mathbf{b}, \quad k = p, p+1, \ldots$$

Taking limit of both sides of the equation above and noting that by consistency

$$\mathbf{x}^k \to -A^{-1}\mathbf{b}$$

for any $\mathbf{b} \in \mathbb{R}^d$, yields

$$-A^{-1}\mathbf{b} = -\sum_{j=0}^{p-1} C_j(A)A^{-1}\mathbf{b} + N(A)\mathbf{b}.$$

Thus,

$$-A^{-1} = -\sum_{j=0}^{p-1} C_j(A)A^{-1} + N(A).$$

Multiplying by A and rearranging, we obtain

$$\sum_{j=0}^{p-1} C_j(A) = I_d + N(A)A. \tag{16}$$

On the other hand, if instead of assuming consistency we assume that $\mathcal{A}$ generates a convergent sequence of points and that Equation (16) holds, then the arguments used above show that the limit point must be $-A^{-1}\mathbf{b}$. In terms of the characteristic polynomial of p-SCLIs, this formalized as follows.

Theorem 5 (Consistency via Characteristic Polynomials) *Suppose* $\mathcal{A} \triangleq (\mathcal{L}(\lambda, X), N(X))$ *is a p-SCLI optimization algorithm. Then,* $\mathcal{A}$ *is consistent with respect to* $A \in \mathcal{S}^d(\Sigma)$ *if and only if the following two conditions hold:*

$$1.\ \mathcal{L}(1, A) = -\mathbb{E}N(A)A \tag{17}$$

$$2.\ \rho_\lambda(\mathcal{L}(\lambda, A)) < 1 \tag{18}$$

The proof for the preceding theorem is provided in Section C.2. This result will be used extensively throughout the reminder of this work.

3.2 Simplified One-Dimensional Case

To illustrate the significance of consistency in the framework of p-SCLIs, consider the following simplified case. Suppose $\mathcal{A}$ is a deterministic 2-SCLI optimization algorithm over $\mathcal{Q}^1([\mu, L])$, such that its inversion matrix $N(x)$ is some constant scalar $\nu \in \mathbb{R}$ and its coefficient matrices $c_0(x), c_1(x)$ are free to take any form. The corresponding characteristic polynomial is

$$\mathcal{L}(\lambda, x) = \lambda^2 - c_1(x)\lambda - c_0(x).$$

Now, let $f_{a,b}(x) \in \mathcal{Q}^1([\mu, L])$ be a quadratic function. By Theorem 4, we know that $\mathcal{A}$ converges to the minimizer of $f_{a,b}(x)$ with an asymptotic geometric rate of $\rho_\lambda(\mathcal{L}(\lambda, a))$, the maximal modulus root. Thus, ideally we would like to set $c_j(x) = 0,\ j = 0, 1$. However, this might violate the consistency condition (17), according to which, one must maintain

$$\mathcal{L}(1, a) = -\nu a.$$

That being the case, how little can $\rho_\lambda\left(\mathcal{L}(\lambda,a)\right)$ be over all possible choices for $c_j(a)$ which satisfy $\mathcal{L}(1,a)=-\nu a$? Formally, we seek to solve the following minimization problem

$$\rho_* = \min\left\{\rho_\lambda(\mathcal{L}(\lambda,a)) \mid \mathcal{L}(\lambda,a) \text{ is a real monic quadratic polynomial in } \lambda \text{ and } \mathcal{L}(1)=-\nu a\right\}.$$

By consistency we also have that ρ_* must be strictly less than one. This readily implies that $-\nu a > 0$. In which case, Lemma 6 below gives

$$\rho_* \geq \rho\left(\left(\lambda - 1 - \sqrt{-\nu a}\right)^2\right) = \left|\sqrt{-\nu a} - 1\right|. \tag{19}$$

The key observation here is that ν cannot be chosen so as to be optimal for all $\mathcal{Q}^1([\mu,L])$ simultaneously. Indeed, the preceding inequality holds in particular for $a=\mu$ and $a=L$, by which we conclude that

$$\rho_* \geq \max\left\{\left|\sqrt{-\nu\mu}-1\right|, \left|\sqrt{-\nu L}-1\right|\right\} \geq \frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1}, \tag{20}$$

where $\kappa \triangleq L/\mu$. Plugging in Inequality (20) into Theorem 4 implies that there exists $f_{a,b}(x) \in \mathcal{Q}^1([\mu,L])$ such that the iteration complexity of $\mathcal{A}$ for minimizing it is

$$\tilde{\Omega}\left(\frac{\sqrt{\kappa}-1}{2}\ln(1/\epsilon)\right).$$

To conclude, by applying this rather natural line of argument we have established a lower bound on the convergence rate of any 2-SCLI optimization algorithms for smooth and strongly convex function over $\mathbb{R}$, e.g., AGD and HB.

3.3 The General Case and the Role of the Inversion Matrix

We now generalize the analysis shown in the previous simplified case to any deterministic p-SCLI optimization algorithm over any finite dimensional space. This generalization relies on a useful decomposability property of the characteristic polynomial, according to which deriving a lower bound on the convergence rate of p-SCLIs over $\mathbb{R}^d$ is essentially equivalent for deriving d lower bounds on the maximal modulus of the roots of d polynomials over $\mathbb{R}$.

Let $\mathcal{A} \triangleq (\mathcal{L}(\lambda,X), N(X))$ be a consistent deterministic p-SCLI optimization algorithm and let $f_{A,\mathbf{b}}(\mathbf{x}) \in \mathcal{Q}^d(\Sigma)$ be a quadratic function. By consistency (see Theorem 5) we have

$$\mathcal{L}(1,A) = -NA$$

(for brevity we omit the functional dependency on X). Since coefficient matrices are assumed to be simultaneously triangularizable, there exists an invertible matrix $Q \in \mathbb{R}^{d\times d}$ such that

$$T_j \triangleq Q^{-1}C_jQ, \quad j=0,1,\ldots,p-1$$

are upper triangular matrices. Thus, by the definition of the characteristic polynomial (Definition 2) we have

$$\det\mathcal{L}(\lambda,X) = \det\left(Q^{-1}\mathcal{L}(\lambda,X)Q\right) = \det\left(I_d\lambda^p - \sum_{j=0}^{p-1}T_j\lambda^j\right) = \prod_{j=1}^{d}\ell_j(\lambda), \tag{21}$$

where

$$\ell_j(\lambda) = \lambda^p - \sum_{k=0}^{p-1} \sigma_j^k \lambda^k, \tag{22}$$

and where $\sigma_1^j, \ldots, \sigma_d^j,\ j = 0, \ldots, p-1$ denote the elements on the diagonal of T_j, or equivalently the eigenvalues of C_j ordered according to Q. Hence, the root radius of the characteristic polynomial of $\mathcal{A}$ is

$$\rho_\lambda(\mathcal{L}(\lambda, X)) = \max\{|\,\lambda\,| \mid \ell_i(\lambda) = 0 \text{ for some } i \in [d]\}. \tag{23}$$

On the other hand, by consistency condition (17) we get that for all $i \in [d]$,

$$\ell_i(1) = \sigma_i\left(\mathcal{L}(1)\right) = \sigma_i\left(-NA\right). \tag{24}$$

It remains to derive a lower bound on the maximum modulus of the roots of $\ell_i(\lambda)$, subject to constraint (24). To this end, we employ the following lemma whose proof can be found in Section C.3.

Lemma 6 *Suppose $q(z)$ is a real monic polynomial of degree p. If $q(1) < 0$, then*

$$\rho(q(z)) > 1.$$

Otherwise, if $q(1) \geq 0$, then

$$\rho(q(z)) \geq \left|\, \sqrt[p]{q(1)} - 1 \right|.$$

In which case, equality holds if and only if

$$q(z) = \left(z - (1 - \sqrt[p]{q(1)})\right)^p.$$

We remark that the second part of Lemma 6 implies that subject to constraint (24), the lower bound stated above is unimprovable. This property is used in Section 5 where we aim to obtain optimal p-SCLIs by designing $\ell_j(\lambda)$ accordingly. Clearly, in the presence of additional constraints, one might be able to improve on this lower bound (see Section 4.2).

Since $\mathcal{A}$ is assumed to be consistent, Lemma 6 implies that $\sigma(-N(A)A) \subseteq \mathbb{R}^{++}$, as well as the following lower bound on the root radius of the characteristic polynomial,

$$\rho_\lambda(\mathcal{L}(\lambda, X)) \geq \max_{i\in[d]} \left|\, \sqrt[p]{\sigma_i(-N(A)A)} - 1 \,\right|. \tag{25}$$

Noticing that the reasoning above can be readily applied to stochastic p-SCLI optimization algorithms, we arrive at the following corollary which combines Theorem 4 and Inequality (25).

Corollary 7 *Let $\mathcal{A}$ be a consistent p-SCLI optimization algorithm with respect to some $A \in \mathcal{S}^d(\Sigma)$, let $N(X)$ denote the corresponding inversion matrix and let*

$$\rho^* = \max_{i\in[d]} \left|\, \sqrt[p]{\sigma_i(-\mathbb{E}N(A)A)} - 1 \,\right|,$$

then the iteration complexity of $\mathcal{A}$ for any $f_{A,\mathbf{b}}(\mathbf{x}) \in \mathcal{Q}^d(\Sigma)$ is lower bounded by

$$\tilde{\Omega}\left(\frac{\rho^*}{1-\rho^*}\ln(1/\epsilon)\right). \tag{26}$$

Using Corollary 7, we are now able to provide a concise 'plug-and-play' scheme for deriving lower bounds on the iteration complexity of p-SCLI optimization algorithms. To motivate this scheme, note that the effectiveness of the lower bound stated in Corollary 7 is directly related to the magnitude of the eigenvalues of $-N(X)X$. To exemplify this, consider the inversion matrix of Newton method (see Section 2.3)

$$N(X) = -X^{-1}.$$

Since

$$\sigma(-N(X)X) = \{1\},$$

the lower bound stated above is meaningless for this case. Nevertheless, the best computational cost for computing the inverse of $d \times d$ regular matrices known today is super-quadratic in d. As a result, this method might become impractical in large scale scenarios where the dimension of the problem space is large enough. A possible solution is to employ inversion matrices whose dependence on X is simpler. On the other hand, if $N(X)$ approximates $-X^{-1}$ very badly, then the root radius of the characteristic polynomial might get too large. For instance, if $N(X) = 0$ then

$$\sigma(-N(X)X) = \{0\},$$

contradicting the consistency assumption, regardless of the choice of the coefficient matrices.

In light of the above, many optimization algorithms can be seen as strategies for balancing the computational cost of obtaining a good approximation for the inverse of X and executing large number of iterations. Put differently, various structural restrictions on the inversion matrix yield different $\sigma(-N(X)X)$, which in turn lead to a lower bound on the root radius of the corresponding characteristic polynomial. This gives rise to the following scheme:

Scheme 1	Lower bounds
Parameters:	• A family of quadratic functions $\mathcal{Q}^d(\Sigma)$ • An inversion matrix $N(X)$ • A lifting factor $p \in \mathbb{N}$,
Choose	$\mathcal{S}' \subseteq \mathcal{S}^d(\Sigma)$
Verify	$\forall A \in \mathcal{S}',\ \sigma(-\mathbb{E}N(A)A) \subseteq (0, 2^p)$ to ensure consistency (Theorem 5)
Bound	$\max_{A \in \mathcal{S}', i \in [d]} \left\lvert \sqrt[p]{\sigma_i(-\mathbb{E}N(A)A)} - 1 \right\rvert$ from below by some $\rho_* \in [0, 1)$
Lower bound:	$\tilde{\Omega}\left(\frac{\rho_*}{1-\rho_*} \ln(1/\epsilon)\right)$

This scheme is implicitly used in the previous Section (3.2), where we established a lower bound on the convergence rate of 2-SCLI optimization algorithms over $\mathbb{R}$ with constant inversion matrix and the following parameters

$$\Sigma = [\mu, L], \quad \mathcal{S}' = \{\mu, L\}.$$

In Section 4 we will make this scheme concrete for scalar and diagonal inversion matrices.

3.4 Bounds Schemes

In spite of the fact that Scheme 1 is expressive enough for producing meaningful lower bounds under various structures of the inversion matrix, it does not allow one to incorporate other lower bounds on the root radius of characteristic polynomials whose coefficient matrices admit certain forms, e.g., linear coefficient matrices (see 35 below). Abstracting away from Scheme 1, we now formalize one of the main pillar of this work, i.e., the relation between the amount of computational cost one is willing to invest in executing each iteration and the total number of iterations needed for obtaining a given level of accuracy. We use this relation to form two schemes for establishing lower and upper bounds for p-SCLIs.

Given a compatible set of parameters: a lifting factor p, an inversion matrix $N(X)$, set of quadratic functions $\mathcal{Q}^d(\Sigma)$ and a set of coefficients matrices $\mathcal{C}$, we denote by $\mathfrak{A}(p, N(X), \mathcal{Q}^d(\Sigma), \mathcal{C})$ the set of consistent p-SCLI optimization algorithms for $\mathcal{Q}^d(\Sigma)$ whose inversion matrix are $N(X)$ and whose coefficient matrices are taken from $\mathcal{C}$. Furthermore, we denote by $\mathfrak{L}(p, N(X), \mathcal{Q}^d(\Sigma), \mathcal{C})$ the following set of polynomial matrices

$$\left\{ \mathcal{L}(\lambda, X) \triangleq I_d \lambda^p - \sum_{j=0}^{p-1} \mathbb{E} C_j(X) \lambda^j \;\middle|\; C_j(X) \in \mathcal{C},\ \mathcal{L}(1, A) = -N(A)A,\ \forall A \in \mathcal{S}^d(\Sigma) \right\}.$$

Since both sets are determined by the same set of parameters, the specifications of which will be occasionally omitted for brevity. The natural one-to-one correspondence between these two set, as manifested by Theorem 4 and Corollary 5, yields

$$\boxed{\min_{\mathcal{A} \in \mathfrak{A}} \max_{f_{A,\mathbf{b}}(\mathbf{x}) \in \mathcal{Q}^d(\Sigma)} \rho_\lambda(\mathcal{L}_{\mathcal{A}}(\lambda, A)) = \min_{\mathcal{L}(\lambda, X) \in \mathfrak{L}} \max_{A \in \mathcal{S}^d(\Sigma)} \rho_\lambda(\mathcal{L}(\lambda, A))} \tag{27}$$

The importance of Equation (27) stems from its ability to incorporate any bound on the maximal modulus root of polynomial matrices into a general scheme for bounding the iteration complexity of p-SCLIs. This is summarized by the following scheme.

Scheme 2	Lower bounds
Given	a set of p-SCLI optimization algorithms $\mathfrak{A}(p, N(X), \mathcal{Q}^d(\Sigma), \mathcal{C})$
Find	$\rho_* \in [0, 1)$ such that $\min_{\mathcal{L}(\lambda, X) \in \mathfrak{L}} \max_{A \in \mathcal{S}^d(\Sigma)} \rho_\lambda(\mathcal{L}(\lambda, A)) \geq \rho_*$
Lower bound:	$\tilde{\Omega}\left(\frac{\rho_*}{1-\rho_*} \ln(1/\epsilon)\right)$

Thus, Scheme 1 is in effect an instantiation of the scheme shown above using Lemma 6. This correspondence of p-SCLI optimization algorithms and polynomial matrices can be also used contrariwise to derive efficient algorithm optimization. Indeed, in Section 2.3 we show how FGD, HB and AGD can be formed as optimal instantiations of the following dual scheme.

Scheme 3	Optimal p-SCLI Optimization Algorithms
Given	a set of polynomial matrices $\mathfrak{L}(p, N(X), \mathcal{Q}^d(\Sigma), \mathcal{C})$
Compute	$\rho^* = \min_{\mathcal{L}(\lambda,X)\in\mathfrak{L}} \max_{A\in\mathcal{S}^d(\Sigma)} \rho_\lambda\left(\mathcal{L}(\lambda, A)\right)$ and denote its minimizer by $\mathcal{L}^*(\lambda, A)$
Upper bound:	The corresponding p-SCLI algorithm for $\mathcal{L}^*(\lambda, A)$
Convergence rate:	$\mathcal{O}\left(\frac{1}{1-\rho^*}\ln(1/\epsilon)\right)$

4. Lower Bounds

In the sequel we derive lower bounds on the convergence rate of p-SCLI optimization algorithms whose inversion matrices are scalar or diagonal, and discuss the assumptions under which these lower bounds meet matching upper bounds. It is likely that this approach can be also effectively applied for block-diagonal inversion, as well as for a much wider set of inversion matrices whose entries depend on a relatively small set of entries of the matrix to be inverted.

4.1 Scalar and Diagonal Inversion Matrices

We derive a lower bound on the convergence rate of p-SCLI optimization algorithms for L-smooth μ-strongly convex functions over $\mathbb{R}^d$ with a scalar inversion matrix $N(X)$ by employing Scheme 1 (see Section 3.3). Note that since the one-dimensional case was already proven in Section 3.2, we may assume that $d \geq 2$.

First, we need to pick a 'hard' matrix in $\mathcal{S}^d([\mu, L])$. It turns out that any positive-definite matrix $A \in \mathcal{S}^d([\mu, L])$ for which

$$\{\mu, L\} \subseteq \sigma(A)\,, \tag{28}$$

will meet this criterion. For the sake of concreteness, let us define

$$A \triangleq \mathrm{Diag}(L, \underbrace{\mu, \ldots, \mu}_{d-1 \text{ times}}).$$

In which case,

$$-\nu\{\mu, L\} = \sigma(-\mathbb{E}N(A)A)\,,$$

where $\nu I = \mathbb{E}[N(A)]$. Thus, to maintain consistency, it must hold that[9]

$$\nu \in \left(\frac{-2^p}{L}, 0\right). \tag{29}$$

9. On a side note, this reasoning also implies that if the spectrum of a given matrix A contains both positive and negative eigenvalues then $A^{-1}b$ cannot be computed using p-SCLIs with scalar inversion matrices.

Next, to bound from below

$$\rho_* \triangleq \max_{i\in[d]} \left| \sqrt[p]{\sigma_i(-\nu A)} - 1 \right| = \max\left\{ |\sqrt[p]{-\nu\mu} - 1|, |\sqrt[p]{-\nu L} - 1| \right\},$$

we split the feasible range of ν (29) into three different sub-ranges as follows:

	$\sqrt[p]{-\nu\mu} - 1 < 0$	$\sqrt[p]{-\nu\mu} - 1 \geq 0$
$\sqrt[p]{-\nu L} - 1 \leq 0$	Case 1 Range: $[-1/L, 0)$ Minimizer: $\nu^* = -1/L$ Lower bounds: $1 - \sqrt[p]{\frac{\mu}{L}}$	N/A
$\sqrt[p]{-\nu L} - 1 > 0$	Case 2 Range: $(-1/\mu, -1/L)$ Minimizer: $-\left(\frac{2}{\sqrt[p]{L}+\sqrt[p]{\mu}}\right)^p$ Lower bound: $\frac{\sqrt[p]{L/\mu}-1}{\sqrt[p]{L/\mu}+1}$	Case 3 (requires: $p \geq \log_2 \kappa$) Range: $(-2^p/L, -1/\mu]$ Minimizer: $-1/\mu$ Lower Bound: $\sqrt[p]{\frac{L}{\mu}} - 1$

Table 1: Lower bound for ρ_* by subranges of ν

Therefore,

$$\rho_* \geq \min\left\{ 1 - \sqrt[p]{\frac{\mu}{L}}, \frac{\sqrt[p]{L/\mu} - 1}{\sqrt[p]{L/\mu} + 1}, \sqrt[p]{\frac{L}{\mu}} - 1 \right\} = \frac{\sqrt[p]{\kappa} - 1}{\sqrt[p]{\kappa} + 1}, \tag{30}$$

where $\kappa \triangleq L/\mu$, upper bounds the condition number of functions in $\mathcal{Q}^d([\mu, L])$. Thus, by Scheme 1, we get the following lower bound on the worse-case iteration complexity,

$$\tilde{\Omega}\left(\frac{\sqrt[p]{\kappa} - 1}{2} \ln(1/\epsilon) \right). \tag{31}$$

As for the diagonal case, it turns out that for any quadratic $f_{A,b}(\mathbf{x}) \in \mathcal{Q}^d([\mu, L])$ which has

$$\begin{pmatrix} \frac{L+\mu}{2} & \frac{L-\mu}{2} \\ \frac{L-\mu}{2} & \frac{L+\mu}{2} \end{pmatrix} \tag{32}$$

as a principal sub-matrix of A, the best p-SCLI optimization algorithm with a diagonal inversion matrix does not improve on the optimal asymptotic convergence rate achieved by scalar inversion matrices (see Section C.4). Overall, we obtain the following theorem.

Theorem 8 *Let $\mathcal{A}$ be a consistent p-SCLI optimization algorithm for L-smooth μ-strongly convex functions over $\mathbb{R}^d$. If the inversion matrix of $\mathcal{A}$ is diagonal, then there exists a quadratic function $f_{A,\mathbf{b}}(\mathbf{x}) \in \mathcal{Q}^d([\mu, L])$ such that*

$$\mathcal{IC}_{\mathcal{A}}\left(\epsilon, f_{A,\mathbf{b}}(\mathbf{x})\right) = \tilde{\Omega}\left(\frac{\sqrt[p]{\kappa} - 1}{2} \ln(1/\epsilon) \right), \tag{33}$$

where $\kappa = L/\mu$.

4.2 Is This Lower Bound Tight?

A natural question now arises: is the lower bound stated in Theorem 8 tight? In short, it turns out that for $p = 1$ and $p = 2$ the answer is positive. For $p > 2$, the answer heavily depends on whether a suitable spectral decomposition is within reach. Obviously, computing the spectral decomposition for a given positive definite matrix A is at least as hard as finding the minimizer of a quadratic function whose Hessian is A. To avoid this, we will later restrict our attention to linear coefficients matrices which allow efficient implementation.

A matching upper bound for $p = 1$ In this case the lower bound stated in Theorem 8 is simply attained by FGD (see Section 2.3).

A matching upper bound for $p = 2$ In this case there are two 2-SCLI optimization algorithm which attain this bound, namely, Accelerated Gradient Descent and The Heavy Ball method (see Section 2.3), whose inversion matrices are scalar and correspond to Case 1 and Case 2 in Table 1, i.e.,

$$N_{\text{HB}} = -\left(\frac{2}{\sqrt{L}+\sqrt{\mu}}\right)^2 I_d, \quad N_{\text{AGD}} = \frac{-1}{L} I_d.$$

Although HB obtains the best possible convergence rate in the class of 2-SCLIs with diagonal inversion matrices, it has a major disadvantage. When applied to general smooth and strongly-convex functions, one cannot guarantee global convergence. That is, in order to converge to the corresponding minimizer, HB must be initialized close enough to the minimizer (see Section 3.2.1 in Polyak 1987). Indeed, if the initialization point is too far from the minimizer then HB may diverge as shown in Section 4.5 in Lessard et al. (2014). In contrast to this, AGD attains a global linear convergence with a slightly worse factor. Put differently, the fact HB is highly adapted to quadratic functions prevents it from converging globally to the minimizers of general smooth and strongly convex functions.

A matching upper bound for $p > 2$ In Subsection A we show that when no restriction on the coefficient matrices is imposed, the lower bound shown in Theorem 8 is tight, i.e., for any $p \in \mathbb{N}$ there exists a matching p-SCLI optimization algorithm with scalar inversion matrix whose iteration complexity is

$$\tilde{\mathcal{O}}\big(\sqrt[p]{\kappa}\ln(1/\epsilon)\big). \tag{34}$$

In light of the existing lower bound which scales according to $\sqrt{\kappa}$, this result may seem surprising at first. However, there is a major flaw in implementing these seemingly ideal p-SCLIs. In order to compute the corresponding coefficients matrices one has to obtain a very good approximation for the spectral decomposition of the positive definite matrix which defines the optimization problem. Clearly, this approach is rarely practical. To remedy this situation we focus on linear coefficient matrices which admit a relatively low computational cost per iteration. That is, we assume that there exist real scalars $\alpha_1, \ldots, \alpha_{p-1}$ and $\beta_1, \ldots, \beta_{p-1}$ such that

$$C_j(X) = \alpha_j X + \beta_j I_d, \quad j = 0, 1, \ldots, p-1, \tag{35}$$

We believe that for these type of coefficient matrices the lower bound derived in Theorem 8 is not tight. Precisely, we conjecture that for any $0 < \mu < L$ and for any consistent p-SCLI optimization algorithm $\mathcal{A}$ with diagonal inversion matrix and linear coefficients matrices, there exists $f_{A,\mathbf{b}}(\mathbf{x}) \in \mathcal{Q}^d([\mu, L])$ such that

$$\rho_\lambda(\mathcal{L}_\mathcal{A}(\lambda, X)) \geq \frac{\sqrt{\kappa} - 1}{\sqrt{\kappa} + 1},$$

where $\kappa \triangleq L/\mu$. Proving this may allow to derive tight lower bounds for many optimization algorithm in the field of machine learning. Using Scheme 2, which allows to incorporate various lower bounds on the root radius of polynomials, one is able to equivalently express this conjecture as follows: suppose $q(z)$ is a p-degree monic real polynomial such that $q(1) = 0$. Then, for any polynomial $r(z)$ of degree $p-1$ and for any $0 < \mu < L$, there exists $\eta \in [\mu, L]$ such that

$$\rho(q(z) - \eta r(z)) \geq \frac{\sqrt{L/\mu} - 1}{\sqrt{L/\mu} + 1}.$$

That being so, can we do better if we allow families of quadratic functions $\mathcal{Q}^d(\Sigma)$ where Σ are not necessarily continuous intervals? It turns out that the answer is positive. Indeed, in Section B we present a 3-SCLI optimization algorithm with linear coefficient matrices which, by being intimately adjusted to quadratic functions whose Hessian admits large enough spectral gap, beats the lower bound of Nemirovsky and Yudin (3). This apparently contradicting result is also discussed in Section B, where we show that lower bound (3) is established by employing quadratic functions whose Hessian admits spectrum which densely populates $[\mu, L]$.

5. Upper Bounds

Up to this point we have projected various optimization algorithms on the framework of p-SCLI optimization algorithms, thereby converting questions on convergence properties into questions on moduli of roots of polynomials. In what follows, we shall head in the opposite direction. That is, first we define a polynomial (see Definition (2)) which meets a prescribed set of constraints, and then we form the corresponding p-SCLI optimization algorithm. As stressed in Section 4.2, we will focus exclusively on linear coefficient matrices which admit low per-iteration computational cost and allow a straightforward extension to general smooth and strongly convex functions. Surprisingly enough, this allows a systematic recovering of FGD, HB, AGD, as well as establishing new optimization algorithms which allow better utilization of second-order information. This line of inquiry is particularly important due to the obscure nature of AGD, and further emphasizes its algebraic characteristic. We defer stochastic coefficient matrices, as in SDCA, (Section 2.1) to future work.

This section is organized as follows. First we apply Scheme 3 to derive general p-SCLIs with linear coefficients matrices. Next, we recover AGD and HB as optimal instantiations under this setting. Finally, although general p-SCLI algorithms are exclusively specified for quadratic functions, we show how p-SCLIs with linear coefficient matrices can be extended to general smooth and strongly convex functions.

5.1 Linear Coefficient Matrices

In the sequel we instantiate Scheme 3 (see Section 3.4) for $\mathcal{C}_{\text{Linear}}$, the family of deterministic linear coefficient matrices.

First, note that due to consistency constraints, inversion matrices of constant p-SCLIs with linear coefficient matrices must be either constant scalar matrices or else be computationally equivalent to A^{-1}. Therefore, since our motivation for resorting to linear coefficient matrices was efficiency, we can safely assume that $N(X) = \nu I_d$ for some $\nu \in (-2^p/L, 0)$. Following Scheme 3, we now seek the optimal characteristic polynomial in $\mathfrak{L}_{\text{Linear}} \triangleq \mathfrak{L}(p, \nu I_d, \mathcal{Q}^d([\mu, L]), \mathcal{C}_{\text{Linear}})$ with a compatible set of parameters (see Section 3.4). In the presence of linearity, the characteristic polynomials takes the following simplified form

$$\mathcal{L}(\lambda, X) = \lambda^p - \sum_{j=0}^{p-1}(a_j X + b_j I_d)\lambda^j, \quad a_j, b_j \in \mathbb{R}.$$

By (23) we have

$$\rho_\lambda(\mathcal{L}(\lambda, X)) = \max\left\{|\lambda| \mid \exists i \in [d],\ \ell_i(\lambda) = 0\right\},$$

where $\ell_i(\lambda)$ denote the factors of the characteristic polynomial as in (22). That is, denoting the eigenvalues of X by $\sigma_1, \ldots, \sigma_d$ we have

$$\ell_i(\lambda) = \lambda^p - \sum_{j=0}^{p-1}(a_j\sigma_i + b_j)\lambda^j = \lambda^p - \sigma_i \sum_{j=0}^{p-1} a_j\lambda^j + \sum_{j=0}^{p-1} b_j\lambda^j.$$

Thus, we can express the maximal root radius of the characteristic polynomial over $\mathcal{Q}^d([\mu, L])$ in terms of the following polynomial

$$\ell(\lambda, \eta) = \lambda^p - (\eta a(\lambda) + b(\lambda)), \tag{36}$$

for some real univariate $p-1$ degree polynomials $a(\lambda)$ and $b(\lambda)$, whereby

$$\max_{A \in \mathcal{S}^d(\Sigma)} \rho_\lambda\left(\mathcal{L}(\lambda, A)\right) = \max_{\eta \in [\mu, L]} \rho\left(\ell(\lambda, \eta)\right).$$

That being the case, finding the optimal characteristic polynomial in $\mathfrak{L}_{\text{Linear}}$ translates to the following minimization problem,

$$\begin{aligned} \underset{\ell(\lambda,\eta) \in \mathfrak{L}_{\text{Linear}}}{\text{minimize}} \quad & \max_{\eta \in [\mu, L]} \rho_\lambda(\ell(\lambda, \eta)) \\ \text{s.t. } & \ell(1, \eta) = -\nu\eta, \quad \eta \in [\mu, L] && (37) \\ & \rho_\lambda(\ell(\lambda, \eta)) < 1 && (38) \end{aligned}$$

(Note that in this case we think of $\mathfrak{L}_{\text{Linear}}$ as a set of polynomials whose variable assumes scalars).

This optimization task can be readily solved for the setting where the lifting factor is $p = 1$, the family of quadratic functions under considerations is $\mathcal{Q}^d([\mu, L])$ and the inversion matrix is $N(X) = \nu I_d,\ \nu \in (-2/L, 0)$. In which case (36) takes the following form

$$\ell(\lambda, \eta) = \lambda - \eta a_0 - b_0,$$

where a_0, b_0 are some real scalars. In order to satisfy (37) for all $\eta \in [\mu, L]$, we have no other choice but to set

$$a_0 = \nu, \quad b_0 = 1,$$

which implies

$$\rho_\lambda(\ell(\lambda, \eta)) = 1 + \nu\eta.$$

Since $\nu \in (-2/L, 0)$, condition 38 follows, as well. The corresponding 1-SCLI optimization algorithm is

$$\mathbf{x}^{k+1} = (I + \nu A)\mathbf{x}^k + \nu\mathbf{b},$$

and its first-order extension (see Section 5.3 below) is precisely FGD (see Section 2.3). Finally, note that the corresponding root radius is bounded from above by

$$\frac{\kappa - 1}{\kappa}$$

for $\nu = -1/L$, the minimizer in Case 2 of Table 1, and by

$$\frac{\kappa - 1}{\kappa + 1}$$

for $\nu = \frac{-2}{\mu + L}$, the minimizer in Case 3 of Table 1. This proves that FGD is optimal for the class of 1-SCLIs with linear coefficient matrices. Figure 5.1 shows how the root radius of the characteristic polynomial of FGD is related to the eigenvalues of the Hessian of the quadratic function under consideration.

5.2 Recovering AGD and HB

Let us now calculate the optimal characteristic polynomial for the setting where the lifting factor is $p = 2$, the family of quadratic functions under considerations is $\mathcal{Q}^d([\mu, L])$ and the inversion matrix is $N(X) = \nu I_d,\ \nu \in (-4/L, 0)$ (recall that the restricted range of ν is due to consistency). In which case (36) takes the following form

$$\ell(\lambda, \eta) = \lambda^2 - \eta(a_1\lambda + a_0) - (b_1\lambda + b_0), \tag{39}$$

for some real scalars a_0, a_1, b_0, b_1. Our goal is to choose a_0, a_1, b_0, b_1 so as to minimize

$$\max_{\eta \in [\mu, L]} \rho_\lambda(\ell(\lambda, \eta))$$

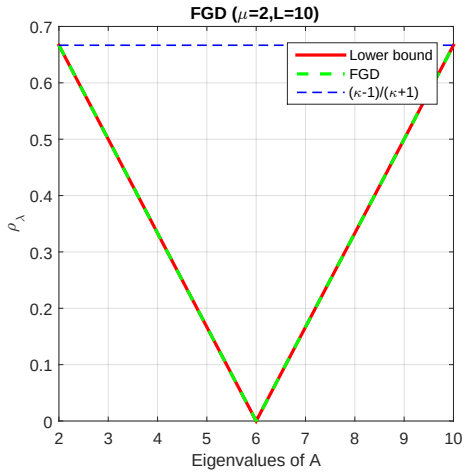

Figure 1: The root radius of FGD vs. various eigenvalues of the corresponding Hessian.

while preserving conditions (37) and (38). Note that $\ell(\lambda, \eta)$, when seen as a function of η, forms a linear path of quadratic functions. Thus, a natural way to achieve this goal is to choose $\ell(\lambda, \eta)$ so that $\ell(\lambda, \mu)$ and $\ell(\lambda, L)$ take the form of the 'economic' polynomials introduced in Lemma 6, namely

$$\left(\lambda - (1 - \sqrt{r})\right)^2$$

for $r = -\nu\mu$ and $r = -\nu L$, respectively, and hope that for others $\eta \in (\mu, L)$, the roots of $\ell(\lambda, \eta)$ would still be of small magnitude. Note that due to the fact that $\ell(\lambda, \eta)$ is linear in η, condition (37) readily holds for any $\eta \in (\mu, L)$. This yields the following two equations

$$\ell(\lambda, \mu) = \left(\lambda - (1 - \sqrt{-\nu\mu})\right)^2,$$
$$\ell(\lambda, L) = \left(\lambda - (1 - \sqrt{-\nu L})\right)^2.$$

Substituting (39) for $\ell(\lambda, \eta)$ and expanding the r.h.s. of the equations above we get

$$\lambda^2 - (a_1\mu + b_1)\lambda - (a_0\mu + b_0) = \lambda^2 - 2(1 - \sqrt{-\nu\mu})\lambda + (1 - \sqrt{-\nu\mu})^2,$$
$$\lambda^2 - (a_1 L + b_1)\lambda - (a_0 L + b_0) = \lambda^2 - 2(1 - \sqrt{-\nu L})\lambda + (1 - \sqrt{-\nu L})^2.$$

Which can be equivalently expressed as the following system of linear equations

$$-(a_1\mu + b_1) = -2(1 - \sqrt{-\nu\mu}), \tag{40}$$
$$-(a_0\mu + b_0) = (1 - \sqrt{-\nu\mu})^2, \tag{41}$$
$$-(a_1 L + b_1) = -2(1 - \sqrt{-\nu L}), \tag{42}$$
$$-(a_0 L + b_0) = (1 - \sqrt{-\nu L})^2. \tag{43}$$

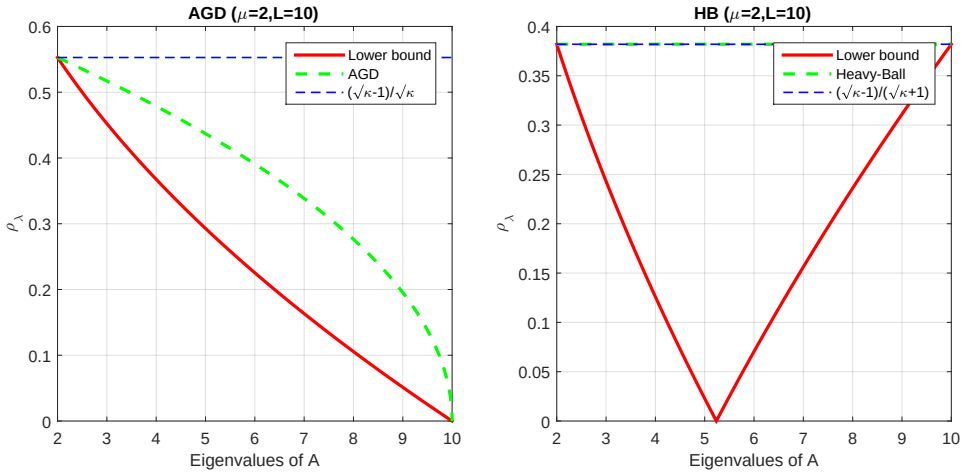

Figure 2: The root radius of AGD and HB vs. various eigenvalues of the corresponding Hessian.

Multiplying Equation (40) by -1 and add to it Equation (42). Next, multiply Equation (41) by -1 and add to it Equation (43) yields

$$a_1(\mu - L) = 2\sqrt{-\nu}(\sqrt{L} - \sqrt{\mu}),$$
$$a_0(\mu - L) = (1 - \sqrt{-\nu L})^2 - (1 - \sqrt{-\nu\mu})^2.$$

Thus,

$$a_1 = \frac{-2\sqrt{-\nu}}{\sqrt{\mu} + \sqrt{L}}, \qquad a_0 = \frac{2\sqrt{-\nu}}{\sqrt{\mu} + \sqrt{L}} + \nu.$$

Plugging in $\nu = -1/L$ (see Table 1) into the equations above and solving for b_1 and b_0 yields a 2-SCLI optimization algorithm whose extension (see Section 5.3 below) is precisely AGD. Following the same derivation only this time by setting (see again Table 1)

$$\nu = -\left(\frac{2}{\sqrt{L} + \sqrt{\mu}}\right)^2$$

yields the Heavy-Ball method .

Moreover, using standard formulae for roots of quadratic polynomials one can easily verify that

$$\rho_\lambda\left(\ell(\lambda, \eta)\right) \leq \frac{\sqrt{\kappa} - 1}{\sqrt{\kappa}}, \ \eta \in [\mu, L],$$

for AGD, and

$$\rho_\lambda\left(\ell(\lambda, \eta)\right) \leq \frac{\sqrt{\kappa} - 1}{\sqrt{\kappa} + 1}, \ \eta \in [\mu, L],$$

for HB. In particular, Condition 38 holds. Figure 5.2 shows how the root radii of the characteristic polynomials of AGD and HB are related to the eigenvalues of the Hessian of the quadratic function under consideration.

Unfortunately, finding the optimal p-SCLIs for $p > 2$ is open and is closely related to the conjecture presented in the end of Section 4.2.

5.3 First-Order Extension for p-SCLIs with Linear Coefficient Matrices

As mentioned before, since the coefficient matrices of p-SCLIs can take any form, it is not clear how to use a given p-SCLI algorithm, efficient as it may be, for minimizing general smooth and strongly convex functions. That being the case, one could argue that recovering the specifications of, say, AGD for quadratic functions does not necessarily imply how to recover AGD itself. Fortunately, consistent p-SCLIs with linear coefficients can be reformulated as optimization algorithms for general smooth and strongly convex functions in a natural way by substituting $\nabla f(\mathbf{x})$ for $A\mathbf{x} + \mathbf{b}$, while preserving the original convergence properties to a large extent. In the sequel we briefly discuss this appealing property, namely, canonical first-order extension, which completes the path from the world of polynomials to the world optimization algorithm for general smooth and strongly convex functions.

Let $\mathcal{A} \triangleq (\mathcal{L}_{\mathcal{A}}(\lambda, X), N(X))$ be a consistent p-SCLI optimization algorithm with a scalar inversion matrix, i.e., $N(X) \triangleq \nu I_d,\ \nu \in (-2^p/L, 0)$, and linear coefficient matrices

$$C_j(X) = a_j X + b_j I_d, \qquad j = 0, \ldots, p-1, \tag{44}$$

where $a_0, \ldots, a_{p-1} \in \mathbb{R}$ and $b_0, \ldots, b_{p-1} \in \mathbb{R}$ denote real scalars. Recall that by consistency, for any $f_{A,\mathbf{b}}(\mathbf{x}) \in \mathcal{Q}^d(\Sigma)$, it holds that

$$\sum_{j=0}^{p-1} C_j(A) = I + \nu A.$$

Thus,

$$\sum_{j=0}^{p-1} b_j = 1 \text{ and } \sum_{j=0}^{p-1} a_j = \nu. \tag{45}$$

By the definition of p-SCLIs (Definition 1), we have that

$$\mathbf{x}^k = C_0(A)\mathbf{x}^{k-p} + C_1(A)\mathbf{x}^{k-(p-1)} + \cdots + C_{p-1}(A)\mathbf{x}^{k-1} + \nu\mathbf{b}.$$

Substituting $C_j(A)$ for (44), gives

$$\mathbf{x}^k = (a_0 A + b_0)\mathbf{x}^{k-p} + (a_1 A + b_1)\mathbf{x}^{k-(p-1)} + \cdots + (a_{p-1} A + b_{p-1})\mathbf{x}^{k-1} + \nu\mathbf{b}.$$

Rearranging and plugging in 45, we get

$$\begin{aligned}\mathbf{x}^k = {} & a_0(A\mathbf{x}^{k-p} + \mathbf{b}) + a_1(A\mathbf{x}^{k-(p-1)} + \mathbf{b}) + \cdots + a_{p-1}(A\mathbf{x}^{k-1} + \mathbf{b}) \\ & + b_0\mathbf{x}^{k-p} + b_1\mathbf{x}^{k-(p-1)} + \cdots + b_{p-1}\mathbf{x}^{k-1}.\end{aligned}$$

Finally, by substituting $A\mathbf{x}+\mathbf{b}$ for its analog $\nabla f(\mathbf{x})$, we arrive at the following canonical first-order extension of $\mathcal{A}$

$$\mathbf{x}^k = \sum_{j=0}^{p-1} b_j \mathbf{x}^{k-(p-j)} + \sum_{j=0}^{p-1} a_j \nabla f(\mathbf{x}^{k-(p-j)}). \tag{46}$$

Being applicable to a much wider collection of functions, how well should we expect the canonical extensions to behave? The answer is that when initialized close enough to the minimizer, one should expect a linear convergence of essentially the same rate. A formal statement is given by the theorem below which easily follows from Theorem 1 in Section 2.1, Polyak (1987) for

$$g(\mathbf{x}^{k-p}, \mathbf{x}^{k-(p-1)}, \ldots, \mathbf{x}^{k-1}) = \sum_{j=0}^{p-1} b_j \mathbf{x}^{k-(p-j)} + \sum_{j=0}^{p-1} a_j \nabla f(\mathbf{x}^{k-(p-j)}).$$

Theorem 9 *Suppose $f:\mathbb{R}^d\to\mathbb{R}$ is an L-smooth μ-strongly convex function and let $\mathbf{x}^*$ denotes its minimizer. Then, for every $\epsilon>0$, there exist $\delta>0$ and $C>0$ such that if*

$$\left\|\mathbf{x}^j-\mathbf{x}^*\right\|\le\delta,\quad j=0,\ldots,p-1,$$

then

$$\left\|\mathbf{x}^k-\mathbf{x}^0\right\|\le C(\rho^*+\epsilon)^k,\quad k=p,p+1,\ldots,$$

where

$$\rho^*=\sup_{\eta\in\Sigma}\rho\left(\lambda^p-\sum_{j=0}^{p-1}(a_j\eta+b_j)\lambda^j\right).$$

Unlike general p-SCLIs with linear coefficient matrices which are guaranteed to converge only when initialized close enough to the minimizer, AGD converges linearly, regardless of the initialization points, for any smooth and strongly convex function. This fact merits further investigation as to the precise principles which underlie p-SCLIs of this kind.

Appendix A. Optimal p-SCLI for Unconstrained Coefficient Matrices

In the sequel we use Scheme 3 (see Section 3.4) to show that, when no constraints are imposed on the functional dependency of the coefficient matrices, the lower bound shown in Theorem 8 is tight. To this end, recall that in Lemma 6 we showed that the lower bound on the maximal modulus of roots of a polynomials which evaluate at $z=1$ to some $r\ge 0$ is uniquely attained by the following polynomial

$$q_r^*(z)\triangleq\left(z-(1-\sqrt[p]{r})\right)^p$$

Thus, by choosing coefficients matrices which admit the same form, we obtain the optimal convergence rate as stated in Theorem 8.

Concretely, let $p \in \mathbb{N}$ be some lifting factor, let $N(X) = \nu I_d,\ \nu \in (-2^p/L, 0)$ be a fixed scalar matrix and let $f_{A,\mathbf{b}}(\mathbf{x}) \in \mathcal{Q}^d(\Sigma)$ be some quadratic function. Lemma 6 implies that for each $\eta \in \sigma(-\nu A)$ we need the corresponding factor of the characteristic polynomial to be

$$\begin{aligned}\ell_j(\lambda) &= (\lambda - (1 - \sqrt[p]{\eta}))^p \\ &= \sum_{k=0}^{p} \binom{p}{k} \left(\sqrt[p]{-\nu\eta} - 1\right)^{p-k} \lambda^k \end{aligned} \tag{47}$$

This is easily accomplished using the spectral decomposition of A by

$$\Lambda \triangleq U^\top A U$$

where U is an orthogonal matrix and Λ is a diagonal matrix. Note that since A is a positive definite matrix such a decomposition must always exist. We define p coefficient matrices $C_0, C_1, \ldots, C_{p-1}$ in accordance with Equation (47) as follows

$$C_k = U \begin{pmatrix} -\binom{p}{k}\left(\sqrt[p]{-\nu\Lambda_{11}} - 1\right)^{p-k} & & & \\ & -\binom{p}{k}\left(\sqrt[p]{-\nu\Lambda_{22}} - 1\right)^{p-k} & & \\ & & \ddots & \\ & & & -\binom{p}{k}\left(\sqrt[p]{-\nu\Lambda_{dd}} - 1\right)^{p-k} \end{pmatrix} U^\top.$$

By using Theorem 5, it can be easily verified that these coefficient matrices form a consistent p-SCLI optimization algorithm whose characteristic polynomial's root radius is

$$\max_{j=1,\ldots,d} \left| \sqrt[p]{-\nu\mu_j} - 1 \right|.$$

Choosing

$$\nu = -\left(\frac{2}{\sqrt[p]{L} + \sqrt[p]{\mu}}\right)^p$$

according to Table 1, produces an optimal p-SCLI optimization algorithm for this set of parameters. It is noteworthy that other suitable decompositions can be used for deriving optimal p-SCLIs, as well.

As a side note, since the cost of computing each iteration in $\in \mathbb{R}^{pd}$ grows linearly with the lifting factor p, the optimal choice of p with respect to the condition number κ yields a p-SCLI optimization algorithm whose iteration complexity is $\Theta(\ln(\kappa)\ln(1/\epsilon))$. Clearly, this result is of theoretical interest only, as this would require a spectral decomposition of A, which, if no other structural assumptions are imposed, is an even harder task than computing the minimizer of $f_{A,\mathbf{b}}(\mathbf{x})$.

Appendix B. Lifting Factor ≥ 3

In Section 4.2 we conjecture that for any p-SCLI optimization algorithm $\mathcal{A} \triangleq (\mathcal{L}(\lambda, X), N(X))$, with diagonal inversion matrix and linear coefficient matrices there exists some $A \in \mathcal{Q}^d([\mu, L])$ such that

$$\rho_\lambda(\mathcal{L}(\lambda, X)) \geq \frac{\sqrt{\kappa} - 1}{\sqrt{\kappa} + 1}, \tag{48}$$

where $\kappa \triangleq L/\mu$. However, it may be possible to overcome this barrier by focusing on a subclass of $\mathcal{Q}^d([\mu, L])$. Indeed, recall that the polynomial analogy of this conjecture states that for any monic real p degree polynomial $q(z)$ such that $q(1) = 0$ and for any polynomial $r(z)$ of degree $p-1$, there exists $\eta \in [\mu, L]$ such that

$$\rho(q(z) - \eta r(z)) \geq \frac{\sqrt{\kappa} - 1}{\sqrt{\kappa} + 1}.$$

This implies that we may be able to tune $q(z)$ and $r(z)$ so as to obtain a convergence rate, which breaks Inequality (48), for quadratic function whose Hessian's spectrum does not spread uniformly across $[\mu, L]$.

Let us demonstrate this idea for $p = 3, \mu = 2$ and $L = 100$. Following the exact same derivation used in the last section, let us pick

$$q(z, \eta) \triangleq z^p - (\eta a(z) + b(z))$$

numerically, so that

$$\begin{aligned} q(z, \mu) &= \left(z - (1 - \sqrt[3]{-\nu\mu})\right)^3 \\ q(z, L) &= \left(z - (1 - \sqrt[3]{-\nu\mu})\right)^3 \end{aligned}$$

where

$$\nu = -\left(\frac{2}{\sqrt[3]{L} + \sqrt[3]{\mu}}\right)^3$$

The resulting 3-CLI optimization algorithm $\mathcal{A}_3$ is

$$\mathbf{x}^k = C_2(X)\mathbf{x}^{k-1} + C_1(X)\mathbf{x}^{k-2} + C_0(X)\mathbf{x}^{k-3} + N(X)b$$

where

$$\begin{aligned} C_0(X) &\approx 0.1958 I_d - 0.0038 X \\ C_1(X) &\approx -0.9850 I_d \\ C_2(X) &\approx 1.7892 I_d - 0.0351 X \\ N(X) &\approx -0.0389 I_d \end{aligned}$$

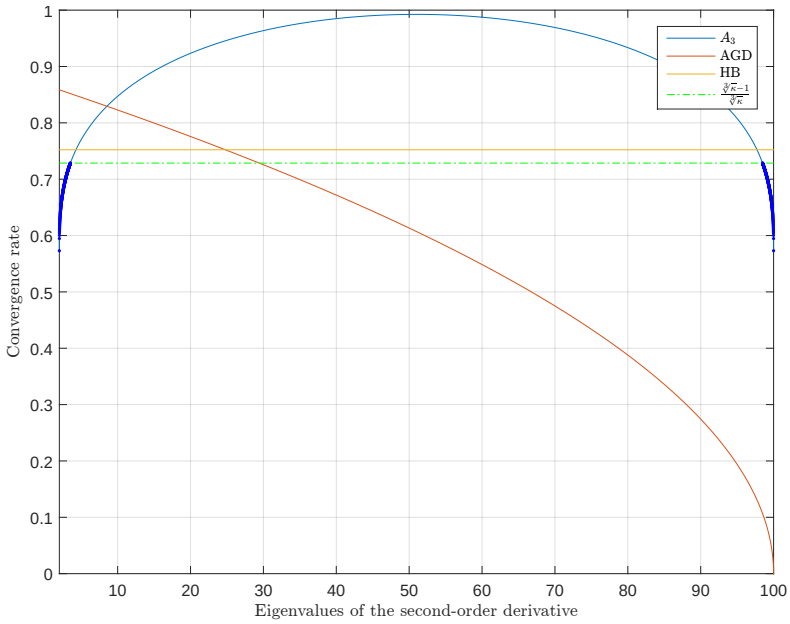

Figure 3: The convergence rate of AGD and $\mathcal{A}_3$ vs. the eigenvalues of the second-order derivatives. It can be seen that the asymptotic convergence rate of $\mathcal{A}_3$ for quadratic functions whose second-order derivative comprises eigenvalues which are close to the edges of $[2, 100]$, is faster than AGD and goes below the theoretical lower bound for first-order optimization algorithm $\frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1}$.

As opposed to the algorithm described in Section A, when employing linear coefficient matrices no knowledge regarding the eigenvectors of A is required. As each eigenvalue of the second-order derivative corresponds to a bound on the convergence rate, one can verify by Figure 3 that

$$\rho_\lambda\left(\mathcal{L}_{\mathcal{A}_3}(\lambda, X)\right) \leq \frac{\sqrt[3]{\kappa}-1}{\sqrt[3]{\kappa}}$$

for any $X \in \mathcal{Q}^d([2, 100])$ which satisfies

$$\sigma(A) \subseteq \hat{\Sigma} \triangleq [2, 2+\epsilon] \cup [100-\epsilon, 100], \quad \epsilon \approx 1.5.$$

Thus, $\mathcal{A}_3$ outperforms AGD for this family of quadratic functions.

Let us demonstrate the gain in the performance allowed by $\mathcal{A}_3$ in a very simple setting. Define A to be $\mathrm{Diag}\,(\mu, L)$ rotated counter-clockwise by 45°, that is

$$A = \mu \begin{pmatrix} \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \end{pmatrix} \begin{pmatrix} \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \end{pmatrix}^\top + L \begin{pmatrix} \frac{1}{\sqrt{2}} \\ \frac{-1}{\sqrt{2}} \end{pmatrix} \begin{pmatrix} \frac{1}{\sqrt{2}} \\ \frac{-1}{\sqrt{2}} \end{pmatrix}^\top = \begin{pmatrix} \frac{\mu+L}{2} & \frac{\mu-L}{2} \\ \frac{\mu-L}{2} & \frac{\mu+L}{2} \end{pmatrix}.$$

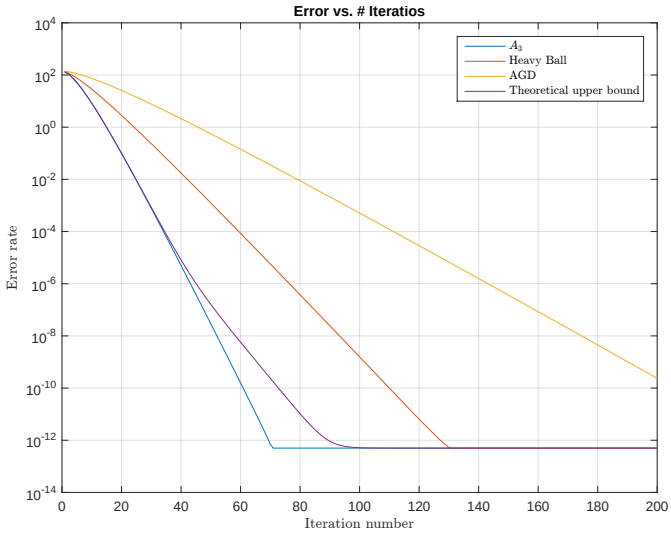

Figure 4: The error rate of $\mathcal{A}_3$, AGD and HB vs. # iterations for solving a simple quadratic minimization task. The convergence rate of $\mathcal{A}_3$ is bounded from above by $\frac{\sqrt[3]{\kappa}-1}{\sqrt[3]{\kappa}}$ as implied by theory.

Furthermore, define $\mathbf{b} = -A\left(100, 100\right)^\top$. Note that $f_{A,\mathbf{b}}(\mathbf{x}) \in \mathcal{Q}^2\left(\hat{\Sigma}\right)$ and that its minimizer is simply $\left(100, 100\right)^\top$. Figure 4 shows the error of $\mathcal{A}_3$, AGD and HB vs. iteration number. All algorithms are initialized at $\mathbf{x}^0 = 0$. Since $\mathcal{A}_3$ is a first-order optimization algorithm, by the lower bound shown in (3) there must exist some quadratic function $f_{A_{\text{lb}},\mathbf{b}_{\text{lb}}}(\mathbf{x}) \in \mathcal{Q}^2([\mu, L])$ such that

$$\mathcal{IC}_{\mathcal{A}_3}\left(\epsilon, f_{A_{\text{lb}},\mathbf{b}_{\text{lb}}}(\mathbf{x})\right) \geq \tilde{\Omega}\left(\sqrt{\kappa}\ln(1/\epsilon)\right). \tag{49}$$

But, since

$$\mathcal{IC}_{\mathcal{A}_3}\left(\epsilon, f_{A,\mathbf{b}}(\mathbf{x})\right) \leq \mathcal{O}\left(\sqrt[3]{\kappa}\ln(1/\epsilon)\right) \tag{50}$$

for every $f_{A,\mathbf{b}}(\mathbf{x}) \in \mathcal{Q}^2\left(\hat{\Sigma}\right)$, we must have $f_{A_{\text{lb}},\mathbf{b}_{\text{lb}}}(\mathbf{x}) \in \mathcal{Q}^2([\mu, L]) \setminus \mathcal{Q}^2\left(\hat{\Sigma}\right)$. Indeed, in the somewhat simpler form of the general lower bound for first-order optimization algorithms,

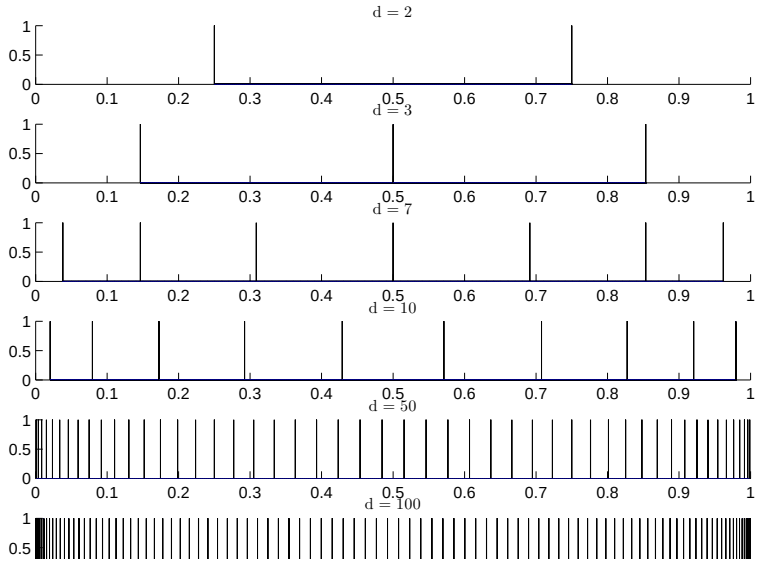

Figure 5: The spectrum of A_{lb}, as used in the derivation of Nesterov's lower bound, for problem space of various dimensions.

Nesterov (see Nesterov 2004) considers the following 1-smooth 0-strongly convex function[10]

$$A_{\text{lb}} = \frac{1}{4}\begin{pmatrix} 2 & -1 & 0 & \dots & & & 0 \\ -1 & 2 & -1 & 0 & \dots & & 0 \\ 0 & -1 & 2 & -1 & 0 & \dots & 0 \\ & & & \ddots & & & \\ 0 & & \dots & 0 & -1 & 2 & -1 \\ 0 & & & \dots & 0 & -1 & 2 \end{pmatrix}, \ \mathbf{b}_{\text{lb}} = -\begin{pmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{pmatrix}.$$

As demonstrated by Figure 5, $\sigma(A_{\text{lb}})$ densely fills $[\mu, L]$.

Consequently, we expect that whenever adjacent eigenvalues of the second-order derivatives are relatively distant, one should able be to minimize the corresponding quadratic function faster than the lower bound stated in 3. This technique can be further generalized to $p > 3$ using the same ideas. Also, a different approach is to use quadratic (or even higher degree) coefficient matrices to exploit other shapes of spectra. Clearly, the applicability of both approaches heavily depends the existence of spectra of this type in real applications.

10. Although $f_{A_{\text{lb}},\mathbf{b}_{\text{lb}}}(\mathbf{x})$ is not strongly convex, the lower bound for strongly convex function is obtained by shifting the spectrum using a regularization term $\mu/2\,\|\mathbf{x}\|^2$. In which case, the shape of the spectrum is preserved.

Appendix C. Proofs

C.1 Proof of Theorem 4

The simple idea behind proof of Theorem 4 is to express the dynamic of a given p-SCLI optimization algorithm as a recurrent application of linear operator. To analyze the latter, we employ the Jordan form which allows us to bind together the maximal magnitude eigenvalue and the convergence rate. Prior to proving this theorem, we first need to introduce some elementary results in linear algebra.

C.1.1 Linear Algebra Preliminaries

We prove two basic lemmas which allow to determine under what conditions does a recurrence application of linear operators over finite dimensional spaces converge, as well as to compute the limit of matrices powers series. It is worth noting that despite of being a very elementary result in Matrix theory and in the theory of power methods, the lower bound part of the first lemma does not seem to appear in this form in standard linear algebra literature.

Lemma 10 *Let A be a $d \times d$ square matrix.*

- *If $\rho(A) > 0$ then there exists $C_A > 0$ such that for any $\mathbf{u} \in \mathbb{R}^d$ and for any $k \in \mathbb{N}$ we have*

$$\left\| A^k \mathbf{u} \right\| \leq C_A k^{m-1} \rho(A)^k \left\| \mathbf{u} \right\|,$$

  where m denotes the maximal index of eigenvalues whose modulus is maximal.
  In addition, there exists $c_A > 0$ and $\mathbf{r} \in \mathbb{R}^d$ such that for any $\mathbf{u} \in \mathbb{R}^d$ which satisfies $\langle \mathbf{u}, \mathbf{r} \rangle \neq 0$ we have

$$\left\| A^k \mathbf{u} \right\| \geq c_A k^{m-1} \rho(A)^k \left\| \mathbf{u} \right\|,$$

  for sufficiently large $k \in \mathbb{N}$.

- *If $\rho(A) = 0$ then A is a nilpotent matrix. In which case, both lower and upper bounds mentioned above hold trivially for any $\mathbf{u} \in \mathbb{R}^d$ for sufficiently large k.*

Proof Let P be a $d \times d$ invertible matrix such that

$$P^{-1} A P = J,$$

where J is a Jordan form of A, namely, J is a block-diagonal matrix such that $J = \oplus_{i=1}^{s} J_{k_i}(\lambda_i)$ where $\lambda_1, \lambda_2, \ldots, \lambda_s$ are eigenvalues of A in a non-increasing order, whose indices are $k_1, \ldots, k_s$, respectively. w.l.o.g. we may assume that $|\lambda_1| = \rho(A)$ and that the corresponding index, which we denote by m, is maximal over all eigenvalues of maximal magnitude. Let $Q_1, Q_2, \cdots, Q_s$ and $R_1, R_2, \cdots, R_s$ denote partitioning of the columns of P and the rows of P^{-1}, respectively, which conform with the Jordan blocks of A.

Note that for all $i \in [d]$, $J_{k_i}(0)$ is a nilpotent matrix of an order k_i. Therefore, for any (λ_i, k_i) and $k \geq k_i - 1$ we have

$$\begin{aligned} J_{k_i}(\lambda_i)^k &= (\lambda_i I_{k_i} + J_{k_i}(0))^k \\ &= \sum_{j=0}^{k} \binom{k}{j} \lambda_i^{k-j} J_{k_i}(0)^j \\ &= \sum_{j=0}^{k_i-1} \binom{k}{j} \lambda_i^{k-j} J_{k_i}(0)^j. \end{aligned}$$

Thus, for non-zero eigenvalues we have

$$\begin{aligned} J_{k_i}(\lambda_i)^k / (k^{m-1}\lambda_1^k) &= \sum_{j=0}^{k_i-1} \frac{\binom{k}{j}\lambda_i^{k-j} J_{k_i}(0)^j}{k^{m-1}\lambda_1^k} \\ &= \sum_{j=0}^{k_i-1} \frac{\binom{k}{j}}{k^{m-1}} \left(\frac{\lambda_i}{\lambda_1}\right)^k \frac{J_{k_i}(0)^j}{\lambda_i^j}. \end{aligned} \tag{51}$$

The rest of the proof pivots around the following equality which holds for any $\mathbf{u} \in \mathbb{R}^{pd}$,

$$\begin{aligned} \left\|A^k \mathbf{u}\right\| &= \left\|P J^k P^{-1} \mathbf{u}\right\| \\ &= \left\|\sum_{i=1}^{s'} Q_i J_{k_i}(\lambda_i)^k R_i \mathbf{u}\right\| \\ &= k^{m-1}\rho(A)^k \left\|\sum_{i=1}^{s'} Q_i \left(J_{k_i}(\lambda_i)/(k^{m-1}\lambda_1^k)\right) R_i \mathbf{u}\right\|, \end{aligned} \tag{52}$$

where s' denotes the smallest index such that $\lambda_i = 0$ for $i > s'$, in case there are zero eigenvalues. Plugging in 51 yields,

$$\left\|A^k \mathbf{u}\right\| = k^{m-1}\rho(A)^k \left\| \underbrace{\sum_{i=1}^{s'} Q_i \left(\sum_{j=0}^{k_i-1} \frac{\binom{k}{j}}{k^{m-1}} \left(\frac{\lambda_i}{\lambda_1}\right)^k \frac{J_{k_i}(0)^j}{\lambda_i^j}\right) R_i \mathbf{u}}_{\mathbf{w}_k} \right\|. \tag{53}$$

Let us denote the sequence of vectors in the r.h.s of the preceding inequality by $\{\mathbf{w}_k\}_{k=1}^{\infty}$. Showing that the norm of $\{\mathbf{w}_k\}_{k=1}^{\infty}$ is bounded from above and away from zero will conclude the proof. Deriving an upper bound is straightforward.

$$\begin{aligned} \|\mathbf{w}_k\| &\leq \sum_{i=1}^{s'} \left\| Q_i \left(\sum_{j=0}^{k_i-1} \frac{\binom{k}{j}}{k^{m-1}} \left(\frac{\lambda_i}{\lambda_1}\right)^k \frac{J_{k_i}(0)^j}{\lambda_i^j}\right) R_i \mathbf{u} \right\| \\ &\leq \|\mathbf{u}\| \sum_{i=1}^{s'} \|Q_i\| \, \|R_i\| \sum_{j=0}^{k_i-1} \left\| \frac{\binom{k}{j}}{k^{m-1}} \left(\frac{\lambda_i}{\lambda_1}\right)^k \frac{J_{k_i}(0)^j}{\lambda_i^j} \right\|. \end{aligned} \tag{54}$$

Since for all $i \in [d]$ we have

$$\frac{\binom{k}{j}}{k^{m-1}} \left(\frac{\lambda_i}{\lambda_1}\right)^k \to 0 \quad \text{or} \quad \left| \frac{\binom{k}{j}}{k^{m-1}} \left(\frac{\lambda_i}{\lambda_1}\right)^k \right| \to 1$$

it holds that Inequality (54) can be bounded from above by some positive scalar C_A. Plugging it in into 53 yields

$$\left\| A^k \mathbf{u} \right\| \leq C_A k^{m-1} \rho(A)^k \left\| \mathbf{u} \right\|.$$

Deriving a lower bound on the norm of $\{\mathbf{w}_k\}$ is a bit more involved. First, we define the following set of Jordan blocks which govern the asymptotic behavior of $\|\mathbf{w}_k\|$

$$\mathcal{I} \triangleq \{i \in [s] \mid |\lambda_i| = \rho(A) \text{ and } k_i = m\}.$$

Equation (51) implies that for all $i \notin \mathcal{I}$

$$J_{k_i}(\lambda_i)^k / (k^{m-1}\lambda_1^k) \to 0 \text{ as } k \to \infty.$$

As for $i \in \mathcal{I}$, the first $k_i - 1$ terms in Equation (51) tend to zero. The last term is a matrix whose entries are all zeros, except for the last entry in the first row which equals

$$\frac{\binom{k}{m-1}}{k^{m-1}} \left(\frac{\lambda_i}{\lambda_1}\right)^k 1/(\lambda_i^{m-1}) \sim \left(\frac{\lambda_i}{\lambda_1}\right)^k 1/(\lambda_i^{m-1})$$

(here, two positive sequences a_k, b_k are asymptotic equivalence, i.e., $a_k \sim b_k$, if $a_k/b_k \to 1$). By denoting the first column of each Q_i by q_i and the last row in each R_i by $r_i^\top$, we get

$$\begin{aligned} \|\mathbf{w}_k\| &\sim \left\| \sum_{i \in \mathcal{I}} \left(\frac{\lambda_i}{\lambda_1}\right)^k \frac{1}{\lambda_i^{m-1}} Q_i J_m(0)^{m-1} R_i \mathbf{u} \right\| \\ &= \left\| \sum_{i \in \mathcal{I}} \left(\frac{\lambda_i}{\lambda_1}\right)^k \frac{q_i r_i^\top \mathbf{u}}{\lambda_i^{m-1}} \right\|. \end{aligned}$$

Now, if $\mathbf{u}$ satisfies $r_1^\top \mathbf{u} \neq 0$ then since $q_1, q_2, \cdots, q_{|\mathcal{I}|}$ are linearly independent, we see that the preceding can be bounded from below by some positive constant $c_A > 0$ which does not depend on k. That is, there exists $c_A > 0$ such that $\|\mathbf{w}_k\| > c_A$ for sufficiently large k. Plugging it in into Equation (53) yields

$$\|A\mathbf{u}\| \geq c_A k^{m-1} \rho(A)^k \|\mathbf{u}\|$$

for any $\mathbf{u} \in \mathbb{R}^d$ such that $\langle \mathbf{u}, \mathbf{r}_1 \rangle \neq 0$ and for sufficiently large k. ■

The following is a well-known fact regarding *Neuman series*, sum of powers of square matrices, which follows easily from Lemma 10.

Lemma 11 *Suppose A is a square matrix. Then, the following statements are equivalent:*

1. $\rho(A) < 1$.
2. $\lim_{k\to\infty} A^k = 0$.
3. $\sum_{k=0}^{\infty} A^k$ *converges.*

In which case, $(I - A)^{-1}$ *exists and* $(I - A)^{-1} = \sum_{k=0}^{\infty} A^k$.

Proof First, note that all norms on a finite-dimensional space are equivalent. Thus, the claims stated in (2) and (3) are well-defined.
The fact that (1) and (2) are equivalent is a direct implication of Lemma 10. Finally, the equivalence of (2) and (3) may be established using the following identity

$$(I - A)\sum_{k=0}^{m-1} A^k = I - A^m, \quad m \in \mathbb{N}.$$

■

C.1.2 Convergence Properties

Let us now analyze the convergence properties of p-SCLI optimization algorithms. First, note that update rule (14) can be equivalently expressed as a single step rule by introducing new variables in some possibly higher-dimensional Euclidean space $\mathbb{R}^{pd}$,

$$\mathbf{z}^0 = \left(\mathbf{x}^0, \mathbf{x}^1, \ldots, \mathbf{x}^{p-1}\right)^\top \in \mathbb{R}^{pd}, \quad \mathbf{z}^k = M(X)\mathbf{z}^{k-1} + UN(X)\mathbf{b}, \quad k = 1, 2, \ldots \tag{55}$$

where

$$U \triangleq (\underbrace{0_d, \ldots, 0_d}_{p-1 \text{ times}}, I_d)^\top \in \mathbb{R}^{pd\times d}, \tag{56}$$

and where $M(X)$ is a mapping from $\mathbb{R}^{d\times d}$ to $\mathbb{R}^{pd\times pd}$-valued random variables which admits the following generalized form of companion matrices

$$\begin{pmatrix} 0_d & I_d & & & \\ & 0_d & I_d & & \\ & & \ddots & \ddots & \\ & & & 0_d & I_d \\ C_0(X) & & \ldots & C_{p-2}(X) & C_{p-1}(X) \end{pmatrix}. \tag{57}$$

Following the convention in the field of linear iterative methods, we call $M(X)$ the *iteration matrix*. Note that in terms of the formulation given in (55), consistency w.r.t $A \in \mathcal{S}^d(\Sigma)$ is equivalent to

$$\mathbb{E}\mathbf{z}^k \to \underbrace{\left(-A^{-1}\mathbf{b}, \ldots, -A^{-1}\mathbf{b}\right)^\top}_{p \text{ times}} \tag{58}$$

regardless of the initialization points and for any $\mathbf{b} \in \mathbb{R}^d$ and $\mathbf{z}^0 \in \mathbb{R}^{pd}$.

To improve readability, we shall omit the functional dependency of the iteration, inversion and coefficient matrices on X in the following discussion. Furthermore, Equation (55) can be used to derive a simple expression of $\mathbf{z}^k$, in terms of previous iterations as follows

$$
\begin{aligned}
\mathbf{z}^1 &= M^{(0)}\mathbf{z}^0 + UN^{(0)}\mathbf{b}, \\
\mathbf{z}^2 &= M^{(1)}\mathbf{z}^1 + UN^{(1)}\mathbf{b} = M^{(1)}M^{(0)}\mathbf{z}^0 + M^{(1)}UN^{(0)}\mathbf{b} + UN^{(1)}\mathbf{b}, \\
\mathbf{z}^3 &= M^{(2)}\mathbf{z}^2 + UN^{(2)}\mathbf{b} = M^{(2)}M^{(1)}M^{(0)}\mathbf{z}^0 + M^{(2)}M^{(1)}UN^{(0)}\mathbf{b} + M^{(2)}UN^{(1)}\mathbf{b} + UN^{(2)}\mathbf{b}, \\
&\vdots \\
\mathbf{z}^k &= \prod_{j=0}^{k-1} M^{(j)}\mathbf{z}^0 + \sum_{m=1}^{k-1}\prod_{j=m}^{k-1} M^{(j)}UN^{(m-1)}\mathbf{b} + UN^{(k-1)}\mathbf{b}, \\
&= \prod_{j=0}^{k-1} M^{(j)}\mathbf{z}^0 + \sum_{m=1}^{k}\left(\prod_{j=m}^{k-1} M^{(j)}\right)UN^{(m-1)}\mathbf{b}.
\end{aligned}
$$

where $\left(M^{(0)}, N^{(0)}\right), \ldots, \left(M^{(k-1)}, N^{(k-1)}\right)$ are k i.i.d realizations of the corresponding iteration matrix and inversion matrix, respectively. We follow the convention of defining an empty product as the identity matrix and defining the multiplication order of factors of abbreviated product notation as multiplication from the highest index to the lowest, i.e., $\prod_{j=1}^{k} M^{(j)} = M^{(k)} \cdots M^{(1)}$. Taking the expectation of both sides yields

$$
\mathbb{E}\mathbf{z}^k = \mathbb{E}[M]^k\mathbf{z}^0 + \left(\sum_{j=0}^{k-1}\mathbb{E}[M]^j\right)\mathbb{E}[UN\mathbf{b}]. \tag{59}
$$

By Lemma 11, if $\rho(\mathbb{E}M) < 1$ then the first term in the r.h.s of Equation (59) vanishes for any initialization point $\mathbf{z}^0$, whereas the second term converges to

$$
(I - \mathbb{E}M)^{-1}\mathbb{E}[UN\mathbf{b}],
$$

the fixed point of the update rule. On the other hand, suppose that $\left(\mathbb{E}\mathbf{z}^k\right)_{k=0}^{\infty}$ converges for any $\mathbf{z}^0 \in \mathbb{R}^d$. Then, this is also true for $\mathbf{z}^0 = 0$. Thus, the second summand in the r.h.s of Equation (59) must converge. Consequently, the sequence $\mathbb{E}[M]^k\mathbf{z}^0$, being a difference of two convergent sequences, converges for all $\mathbf{z}^0$, which implies $\rho(\mathbb{E}[M]) < 1$. This proves the following theorem.

Theorem 12 *With the notation above, $\left(\mathbb{E}\mathbf{z}^k\right)_{k=0}^{\infty}$ converges for any $\mathbf{z}^0 \in \mathbb{R}^d$ if and only if $\rho(\mathbb{E}[M]) < 1$. In which case, for any initialization point $\mathbf{z}^0 \in \mathbb{R}^d$, the limit is*

$$
\mathbf{z}^* \triangleq (I - \mathbb{E}M)^{-1}\,\mathbb{E}[UN\mathbf{b}]. \tag{60}
$$

We now address the more delicate question as to how fast do p-SCLIs converge. To this end, note that by Equation (59) and Theorem 12 we have

$$\begin{aligned}
\mathbb{E}\left[\mathbf{z}^k - \mathbf{z}^*\right] &= \mathbb{E}[M]^k \mathbf{z}^0 + \left(\sum_{l=0}^{k-1} \mathbb{E}[M]^i\right) \mathbb{E}[UN\mathbf{b}] - (I - \mathbb{E}M)^{-1} \mathbb{E}[UN\mathbf{b}] \\
&= \mathbb{E}[M]^k \mathbf{z}^0 + (I - \mathbb{E}M)^{-1} \left((I - \mathbb{E}M) \sum_{l=0}^{k-1} \mathbb{E}[M]^i - I\right) \mathbb{E}[UN\mathbf{b}] \\
&= \mathbb{E}[M]^k \mathbf{z}^0 - (I - \mathbb{E}M)^{-1} (\mathbb{E}M)^k \mathbb{E}[UN\mathbf{b}] \\
&= \mathbb{E}[M]^k (\mathbf{z}^0 - \mathbf{z}^*). \qquad (61)
\end{aligned}$$

Hence, to obtain a full characterization of the convergence rate of $\left\|\mathbb{E}\left[\mathbf{z}^k - \mathbf{z}^*\right]\right\|$ in terms of $\rho(\mathbb{E}M)$, all we need is to simply apply Lemma 10 with $\mathbb{E}M$.

C.1.3 Proof

We are now in position to prove Theorem 4. Let $\mathcal{A} \stackrel{\triangle}{=} (\mathcal{L}(\lambda, X), N(X))$ be a p-SCLI algorithm over $\mathbb{R}^d$, let $M(X)$ denote its iteration matrix and let $f_{A,\mathbf{b}}(\mathbf{x})$ be some quadratic function. According to the previous discussion, there exist $m \in \mathbb{N}$ and $C(A), c(A) > 0$ such that the following hold:

1. For any initialization point $\mathbf{z}^0 \in \mathbb{R}^{pd}$, we have that $(\mathbb{E}\mathbf{z}^k)_{k=1}^{\infty}$ converges to
$$\mathbf{z}^* \stackrel{\triangle}{=} (I - \mathbb{E}M(A))^{-1} \mathbb{E}\left[UN(A)\mathbf{b}\right]. \qquad (62)$$
2. For any initialization point $\mathbf{z}^0 \in \mathbb{R}^{pd}$ and for any $h \in \mathbb{N}$,
$$\left\|\mathbb{E}\left[\mathbf{z}^k - \mathbf{z}^*\right]\right\| \leq C_A k^{m-1} \rho(M(A))^k \left\|\mathbf{z}^0 - \mathbf{z}^*\right\|. \qquad (63)$$
3. There exists $\mathbf{r} \in \mathbb{R}^{pd}$ such that for any initialization point $\mathbf{z}^0 \in \mathbb{R}^{pd}$ which satisfies $\left\langle \mathbf{z}^0 - \mathbf{z}^*, \mathbf{r}\right\rangle \neq 0$ and sufficiently large $k \in \mathbb{N}$,
$$\left\|\mathbb{E}\left[\mathbf{z}^k - \mathbf{z}^*\right]\right\| \geq c_A k^{m-1} \rho(M(A))^k \left\|\mathbf{z}^0 - \mathbf{z}^*\right\|. \qquad (64)$$

Since iteration complexity is defined over the problem space, we need to derive the same inequalities in terms of

$$\mathbf{x}^k = U^\top \mathbf{z}^k.$$

Note that by linearity we have $\mathbf{x}^* = U^\top \mathbf{z}^*$. For bounding $(\mathbf{x}_k)_{k=1}^{\infty}$ from above we use (63),

$$\begin{aligned}
\left\|\mathbb{E}\left[\mathbf{x}^k - \mathbf{x}^*\right]\right\| &= \left\|\mathbb{E}\left[U^\top \mathbf{z}^k - U^\top \mathbf{z}^*\right]\right\| \\
&\leq \left\|U^\top\right\| \left\|\mathbb{E}\left[\mathbf{z}^k - \mathbf{z}^*\right]\right\| \\
&\leq \left\|U^\top\right\| C_A k^{m-1} \rho(M)^k \left\|\mathbf{z}^0 - \mathbf{z}^*\right\| \\
&= \left\|U^\top\right\| C_A k^{m-1} \rho(M)^k \left\|U\mathbf{x}^0 - U\mathbf{x}^*\right\| \\
&\leq \left\|U^\top\right\| \|U\| C_A k^{m-1} \rho(M)^k \left\|\mathbf{x}^0 - \mathbf{x}^*\right\|. \qquad (65)
\end{aligned}$$

Thus, the same rate as in (63), with a different constant, holds in the problem space. Although the corresponding lower bound takes a slightly different form, it proof is done similarly. Pick $\mathbf{x}^0, \mathbf{x}^1, \ldots, \mathbf{x}^{p-1}$ such that the corresponding $\mathbf{z}^0$ is satisfied the condition in (64). For sufficiently large $k \in \mathbb{N}$, it holds that

$$\begin{aligned}
\max_{k=0,\ldots,p-1} \left\| \mathbb{E}\mathbf{x}^{k+j} - \mathbb{E}\mathbf{x}^* \right\| &\geq \frac{1}{\sqrt{p}} \sqrt{\sum_{j=0}^{p-1} \left\| \mathbb{E}\mathbf{x}^{k+j} - \mathbb{E}\mathbf{x}^* \right\|^2} \\
&= \frac{1}{\sqrt{p}} \left\| \mathbb{E}\left[\mathbf{z}^k\right] - \mathbf{z}^* \right\| \\
&\geq \frac{c_A}{\sqrt{p}} k^{m-1} \rho(M)^k \left\| \mathbf{z}^0 - \mathbf{z}^* \right\| \\
&= \frac{c_A}{\sqrt{p}} k^{m-1} \rho(M)^k \sqrt{\sum_{j=0}^{p-1} \left\| \mathbf{x}^j - \mathbf{x}^* \right\|^2}. \qquad (66)
\end{aligned}$$

We arrived at the following corollary which states that the asymptotic convergence rate of any p-SCLI optimization algorithm is governed by the spectral radius of its iteration matrix.

Theorem 13 *Suppose $\mathcal{A}$ is a p-SCLI optimization algorithm over $\mathcal{Q}^d(\Sigma)$ and let $M(X)$ denotes its iteration matrix. Then, there exists $m \in \mathbb{N}$ such that for any quadratic function $f_{A,\mathbf{b}}(\mathbf{x}) \in \mathcal{Q}^d(\Sigma)$ it holds that*

$$\left\| \mathbb{E}\left[\mathbf{x}^k - \mathbf{x}^*\right] \right\| = \mathcal{O}\Big(k^{m-1} \rho(M(X))^k \left\| \mathbf{x}^0 - \mathbf{x}^* \right\| \Big),$$

where $\mathbf{x}^*$ denotes the minimizer of $f_{A,b}(\mathbf{x})$. Furthermore, there exists an initialization point $\mathbf{x}^0 \in \mathbb{R}^d$, such that

$$\max_{k=0,\ldots,p-1} \left\| \mathbb{E}\mathbf{x}^{k+j} - \mathbb{E}\mathbf{x}^* \right\| = \Omega\left(\frac{k^{m-1}}{\sqrt{p}} \rho(M(X))^k \left\| \mathbf{x}^0 - \mathbf{x}^* \right\| \right).$$

Finally, in the next section we prove that the spectral radius of the iteration matrix equals the root radius of the determinant of the characteristic of polynomial by showing that

$$\det(\lambda I - M(X)) = \det(\mathcal{L}(\lambda, X)).$$

Combining this with the corollary above and by applying Inequality (12) and the like, concludes the proof for Theorem 4.

C.1.4 The Characteristic Polynomial of the Iteration Matrix

The following lemma provides an explicit expression for the characteristic polynomial of iteration matrices. The proof is carried out by applying elementary determinant manipulation rules.

Lemma 14 *Let $M(X)$ be the matrix defined in (57) and let A be a given $d \times d$ square matrix. Then, the characteristic polynomial of $\mathbb{E}M(A)$ can be expressed as the following matrix polynomial*

$$\chi_{\mathbb{E}M(A)}(\lambda) = (-1)^{pd} \det\left(\lambda^p I_d - \sum_{k=0}^{p-1} \lambda^k \mathbb{E}C_k(A)\right). \tag{67}$$

Proof As usual, for the sake of readability we omit the functional dependency on A, as well as the expectation operator symbol. For $\lambda \neq 0$ we get,

$$\begin{aligned}
\chi_{\mathcal{M}}(\lambda) &= \det(M - \lambda I_{pd}) \\
&= \det\left(\begin{array}{cccc|c}
-\lambda I_d & I_d & & & \\
& -\lambda I_d & I_d & & \\
& & \ddots & \ddots & \\
& & & -\lambda I_d & I_d \\
\hline
C_0 & & \dots & C_{p-2} & C_{p-1} - \lambda I_d
\end{array}\right) \\
&= \det\left(\begin{array}{cccc|c}
-\lambda I_d & I_d & & & \\
& -\lambda I_d & I_d & & \\
& & \ddots & \ddots & \\
& & & -\lambda I_d & I_d \\
\hline
0_d & C_1 + \lambda^{-1} C_0 & \dots & C_{p-2} & C_{p-1} - \lambda I_d
\end{array}\right) \\
&= \det\left(\begin{array}{ccccc|c}
-\lambda I_d & I_d & & & & \\
& -\lambda I_d & & I_d & & \\
& & & \ddots & \ddots & \\
& & & & -\lambda I_d & I_d \\
\hline
0_d & 0_d & C_2 + \lambda^{-1} C_1 + \lambda^{-2} C_0 \dots & C_{p-2} & C_{p-1} - \lambda I_d
\end{array}\right) \\
&= \det\left(\begin{array}{cccc|c}
-\lambda I_d & I_d & & & \\
& -\lambda I_d & I_d & & \\
& & \ddots & \ddots & \\
& & & -\lambda I_d & I_d \\
\hline
0_d & \dots & 0_d & & \sum_{k=1}^{p} \lambda^{k-p} C_{k-1} - \lambda I_d
\end{array}\right) \\
&= \det(-\lambda I_d)^{p-1} \det\left(\sum_{k=1}^{p} \lambda^{k-p} C_{k-1} - \lambda I_d\right)
\end{aligned}$$

$$= (-1)^{(p-1)d} \det\left(\sum_{k=1}^{p} \lambda^{k-1} C_{k-1} - \lambda^p I_d\right)$$
$$= (-1)^{pd} \det\left(\lambda^p I_d - \sum_{k=0}^{p-1} \lambda^k C_k\right).$$

By continuity we have that the preceding equality holds for $\lambda = 0$ as well. ■

C.2 Proof of Theorem 5

We prove that consistent p-SCLI optimization algorithms must satisfy conditions (17) and (18). The reverse implication is proven by reversing the steps of the proof.

First, note that (18) is an immediate consequence of Corollary 13, according to which p-SCLIs converge if and only if the the root radius of the characteristic polynomial is strictly smaller than 1. As for (18), let $\mathcal{A} \triangleq (\mathcal{L}(\lambda, X), N(X))$ be a consistent p-SCLI optimization algorithm over $\mathcal{Q}^d(\Sigma)$ and let $f_{A,\mathbf{b}}(\mathbf{x}) \in \mathcal{Q}^d(\Sigma)$ be a quadratic function. Furthermore, let us denote the corresponding iteration matrix by $M(X)$ as in (57). By Theorem 12, for any initialization point we have

$$\mathbb{E}\mathbf{z}^k \to (I - \mathbb{E}M(A))^{-1} U \mathbb{E}[N(A)]\mathbf{b},$$

where U is as defined in (56), i.e.,

$$U \triangleq (\underbrace{0_d, \ldots, 0_d}_{p-1 \text{ times}}, I_d)^\top \in \mathbb{R}^{pd \times d}.$$

For the sake of readability we omit the functional dependency on A, as well as the expectation operator symbol. Combining this with Equation (58) yields

$$U^\top (I - M)^{-1} U N \mathbf{b} = -A^{-1}\mathbf{b}.$$

Since this holds for any $\mathbf{b} \in \mathbb{R}^d$, we get

$$U^\top (I - M)^{-1} U N = -A^{-1}.$$

Evidently, N is an invertible matrix. Therefore,

$$U^\top (I - M)^{-1} U = -(NA)^{-1}. \tag{68}$$

Now, recall that

$$M = \begin{pmatrix} 0_d & I_d & & & \\ & 0_d & I_d & & \\ & & \ddots & \ddots & \\ & & & 0_d & I_d \\ C_0 & & \ldots & C_{p-2} & C_{p-1} \end{pmatrix},$$

where C_j denote the coefficient matrices. We partition M as follows

$$\left(\begin{array}{c|c} M_{11} & M_{12} \\ \hline M_{21} & M_{22} \end{array}\right) \triangleq \left(\begin{array}{cccc|c} 0_d & I_d & & & \\ & 0_d & I_d & & \\ & & \ddots & \ddots & \\ & & & 0_d & I_d \\ \hline C_0 & & \dots & C_{p-2} & C_{p-1} \end{array}\right).$$

The l.h.s of Equation (68) is in fact the inverse of the Schur Complement of $I - M_{11}$ in $I - M$, i.e.,

$$\begin{aligned} (I - M_{22} - M_{21}(I - M_{11})^{-1}M_{12})^{-1} &= -(NA)^{-1} \\ I - M_{22} - M_{21}(I - M_{11})^{-1}M_{12} &= -NA \\ M_{22} + M_{21}(I - M_{11})^{-1}M_{12} &= I + NA. \end{aligned} \tag{69}$$

Moreover, it is straightforward to verify that

$$(I - M_{11})^{-1} = \left(\begin{array}{cccc} I_d & & I_d & I_d \\ & I_d & & I_d \\ & & \ddots & \\ & & & I_d \end{array}\right)$$

Plugging in this into (69) yields

$$\sum_{i=0}^{p-1} C_i = I + NA,$$

or equivalently,

$$\mathcal{L}(1, A) = -NA \tag{70}$$

Thus concludes the proof.

C.3 Proof of Lemma 6

First, we prove the following Lemma. Let us denote

$$q_r^*(z) \triangleq \left(z - (1 - \sqrt[p]{r})\right)^p,$$

where r is some non-negative constant.

Lemma 15 *Suppose $q(z)$ is a monic polynomial of degree p with complex coefficients. Then,*

$$\rho(q(z)) \leq \left| \sqrt[p]{|\, q(1)\,|} - 1 \right| \iff q(z) = q^*_{|\, q(1)\,|}(z).$$

Proof As the $\Leftarrow$ statement is clear, we prove here only the $\Rightarrow$ part.
By the fundamental theorem of algebra $q(z)$ has p roots. Let us denote these roots by $\zeta_1, \zeta_2, \ldots, \zeta_p \in \mathbb{C}$. Equivalently,

$$q(z) = \prod_{i=1}^{p}(z - \zeta_i).$$

Let us denote $r \triangleq |\, q(1)\,|$. If $r \geq 1$ we get

$$\begin{aligned} r &= \left| \prod_{i=1}^{p}(1 - \zeta_i) \right| = \prod_{i=1}^{p} |\, 1 - \zeta_i \,| \leq \prod_{i=1}^{p}(1 + |\, \zeta_i \,|) \\ &\leq \prod_{i=1}^{p}(1 + \left|\, \sqrt[p]{r} - 1 \,\right|) = \prod_{i=1}^{p}(1 + \sqrt[p]{r} - 1) = r. \end{aligned} \tag{71}$$

Consequently, Inequality (71) becomes an equality. Therefore,

$$|\, 1 - \zeta_i \,| = 1 + |\, \zeta_i \,| = \sqrt[p]{r}, \quad \forall i \in [p]. \tag{72}$$

Now, for any two complex numbers $w, z \in \mathbb{C}$ it holds that

$$|\, w + z \,| = |\, w \,| + |\, z \,| \iff \mathrm{Arg}(w) = \mathrm{Arg}(z).$$

Using this fact in the first equality of Equation (72), we get that $\mathrm{Arg}(-\zeta_i) = \mathrm{Arg}(1) = 0$, i.e., ζ_i are negative real numbers. Writing $-\zeta_i$ in the second equality of Equation (72) instead of $|\, \zeta_i \,|$, yields $1 - \zeta_i = \sqrt[p]{r}$, concluding this part of the proof.

The proof for $r \in [0, 1)$ follows along the same lines, only this time we use the reverse triangle inequality,

$$\begin{aligned} r &= \prod_{i=1}^{p} |\, 1 - \zeta_i \,| \geq \prod_{i=1}^{p} (1 - |\, \zeta_i \,|) \geq \prod_{i=1}^{p} \left(1 - \left|\, \sqrt[p]{r} - 1 \,\right|\right) \\ &= \prod_{i=1}^{p} \left(1 - (1 - \sqrt[p]{r})\right) = r. \end{aligned}$$

Note that in the first inequality, we used the fact that $r \in [0, 1) \implies |\, \zeta_i \,| \leq 1$ for all i. ■

The proof for Lemma 6 now follows easily. In case $q(1) \geq 0$, if $q(z) = (z - (1 - \sqrt[p]{r}))^p$ then, clearly,

$$\rho(q(z)) = \rho\left((z - (1 - \sqrt[p]{r})^p)\right) = \left|\, 1 - \sqrt[p]{r} \,\right|.$$

Otherwise, according to Lemma 15

$$\rho(q(z)) > \left|\, 1 - \sqrt[p]{r} \,\right|.$$

In case $q(1) \leq 0$, we must use the assumption that the coefficients are reals (see Remark 16), in which case the mere fact that

$$\lim_{z \in \mathbb{R}, z \to \infty} q(z) = \infty$$

combined with the Mean-Value theorem implies $\rho(q(z)) \geq 1$. This concludes the proof.

Remark 16 *The requirement that the coefficients of $q(z)$ should be real is inevitable. To see why, consider the following polynomial,*

$$u(z) = \left(z - \left(1 - 0.5e^{\frac{i\pi}{3}}\right)\right)^3.$$

Although $u(1) = \left(1 - \left(1 - 1/2e^{\frac{i\pi}{3}}\right)\right)^3 = -1/8 \leq 0$, it holds that $\rho(u(z)) < 1$. Indeed, not all the coefficients of $u(z)$ are real. Notice that the claim does hold for degree ≤ 3, regardless of the additional assumption on the coefficients of $u(z)$.

C.4 Bounding the Spectral Radius of Diagonal Inversion Matrices from below Using Scalar Inversion Matrices

We prove a lower bound on the convergence rate of p-SCLI optimization algorithm with diagonal inversion matrices. In particular, we show that for any p-SCLI optimization algorithm whose inversion matrix is diagonal there exists a quadratic function for which it does not perform better than p-SCLI optimization algorithms with scalar inversion matrix. We prove the claim for $d = 2$. The general case follows by embedding the 2-dimensional case as a principal sub-matrix in some higher dimensional matrix in $\mathcal{S}^d([\mu, L])$. Also, although here we prove for deterministic p-SCLIs, the stochastic case is straightforward.

Let $\mathcal{A}$ be a p-SCLI optimization algorithm with iteration matrix $M(X)$ (defined in (57)) and diagonal inversion matrix $N(X)$. Define the following positive definite matrix

$$B = \begin{pmatrix} \frac{L+\mu}{2} & \frac{L-\mu}{2} \\ \frac{L-\mu}{2} & \frac{L+\mu}{2} \end{pmatrix}, \tag{73}$$

and note that $\sigma(B) = \{\mu, L\}$. As usual, we wish to derive a lower bound on $\rho(M(B))$. To this end, denote

$$N \triangleq N(B) = \begin{pmatrix} \alpha & 0 \\ 0 & \beta \end{pmatrix},$$

where $\alpha, \beta \in \mathbb{R}$. By a straightforward calculation we get that the eigenvalues of $-NB$ are

$$\begin{aligned} \sigma_{1,2}(\alpha, \beta) &= \frac{-(\alpha+\beta)(L+\mu)}{4} \pm \sqrt{\left(\frac{(\alpha+\beta)(L+\mu)}{4}\right)^2 - \alpha\beta L\mu} \\ &= \frac{-(\alpha+\beta)(L+\mu)}{4} \pm \sqrt{(\alpha+\beta)^2\frac{(L-\mu)^2}{16} + \frac{1}{4}(\alpha-\beta)^2 L\mu}. \end{aligned} \tag{74}$$

Using similar arguments to the ones which were applied in the scalar case, we get that both eigenvalues of $-NB$ must be strictly positive as well as satisfy

$$\rho(M) \geq \min_{\alpha,\beta} \max \left\{ \left| \sqrt[p]{\sigma_1(\alpha,\beta)} - 1 \right|, \left| \sqrt[p]{\sigma_2(\alpha,\beta)} - 1 \right| \right\}. \tag{75}$$

Equation (74) shows that the minimum of the preceding is obtained for $\nu = \frac{\alpha+\beta}{2}$, which simplifies to

$$\begin{aligned} \max \left\{ \left| \sqrt[p]{\sigma_1(\alpha,\beta)} - 1 \right|, \left| \sqrt[p]{\sigma_2(\alpha,\beta)} - 1 \right| \right\} &\geq \max \left\{ \left| \sqrt[p]{\sigma_1(\nu,\nu)} - 1 \right|, \left| \sqrt[p]{\sigma_2(\nu,\nu)} - 1 \right| \right\} \\ &= \max \left\{ \left| \sqrt[p]{-\nu\mu} - 1 \right|, \left| \sqrt[p]{-\nu L} - 1 \right| \right\}. \end{aligned}$$

The rest of the analysis is carried out similarly to the scalar case, resulting in

$$\rho(M(B)) \geq \frac{\sqrt[p]{\kappa} - 1}{\sqrt[p]{\kappa} + 1}.$$

References

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.

Michel Baes. Estimate sequence methods: extensions and approximations. 2009.

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.

MP Drazin, JW Dungey, and KW Gruenberg. Some theorems on commutative matrices. *Journal of the London Mathematical Society*, 1(3):221–228, 1951.

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

Israel Gohberg, Pnesteeter Lancaster, and Leiba Rodman. *Matrix polynomials*, volume 58. SIAM, 2009.

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

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

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.

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

Harold J Kushner and George Yin. *Stochastic approximation and recursive algorithms and applications*, volume 35. Springer, 2003.

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

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

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

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.

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

Arkadi Nemirovski. Efficient methods in convex programming. 2005.

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

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

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

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

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

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.

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.

James C Spall. *Introduction to stochastic search and optimization: estimation, simulation, and control*, volume 65. John Wiley & Sons, 2005.

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.

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

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

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

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.