Math. Program., Ser. A (2016) 160:307–320
DOI 10.1007/s10107-016-0984-8

FULL LENGTH PAPER

On the worst case performance of the steepest descent algorithm for quadratic functions

Clóvis C. Gonzaga[1]

Received: 11 September 2014 / Accepted: 23 January 2016 / Published online: 18 February 2016

Abstract The existing choices for the step lengths used by the classical steepest descent algorithm for minimizing a convex quadratic function require in the worst case $\mathcal{O}(C\log(1/\varepsilon))$ iterations to achieve a precision ε, where C is the Hessian condition number. We show how to construct a sequence of step lengths with which the algorithm stops in $\mathcal{O}(\sqrt{C}\log(1/\varepsilon))$ iterations, with a bound almost exactly equal to that of the Conjugate Gradient method.

Mathematics Subject Classification 90C30 (main) · 65K05 · 68Q25

1 Introduction

We study the quadratic minimization problem

$$(P_z) \quad \underset{z\in\mathbb{R}^n}{\text{minimize}}\ \bar{f}(z) = c^T z + \frac{1}{2} z^T H z,$$

where $c \in \mathbb{R}^n$ and $H \in \mathbb{R}^{n\times n}$ is symmetric with eigenvalues

$$0 < \mu_1 \le \mu_2 \le \cdots \le \mu_n,$$

and condition number $C = \mu_n/\mu_1$. The problem has a unique solution $z^* \in \mathbb{R}^n$.

The author was partially supported by CNPq under Grant 301705/2006-2.

✉ Clóvis C. Gonzaga
ccgonzaga1@gmail.com

[1] Department of Mathematics, Federal University of Santa Catarina, Cx. Postal 5210, 88040-970 Florianópolis, SC, Brazil

The steepest descent algorithm, also called gradient method, is a memoryless method defined by

$$z^0 \in \mathbb{R}^n \text{ given}, \; z^{k+1} = z^k - \lambda_k \nabla f(z^k). \tag{1}$$

The only distinction among different steepest descent algorithms is in the choice of the step lengths λ_k. The worst-case performance of a scheme is based on a stopping rule, which depends on a precision $\varepsilon > 0$.

Assume that an initial point $z^0 \in \mathbb{R}^n$ is given. Let $\varepsilon > 0$ be a given precision. Let M be a matrix whose columns M_i, $i = 1, \ldots, m$, are orthonormal eigenvectors of H. We define four different stopping rules referred as τ_ε, which will be used by algorithms:

i. $\bar{f}(z) - \bar{f}(z^*) \leq \varepsilon^2(\bar{f}(z^0) - \bar{f}(z^*))$.
ii. $\|z - z^*\| \leq \varepsilon \|z^0 - z^*\|$.
iii. $\|\nabla \bar{f}(z)\| \leq \varepsilon \|\nabla \bar{f}(z^0)\|$.
iv. $|M_i^T(z - z^*)| \leq \varepsilon |M_i^T(z^0 - z^*)|$, i=1,…,n.

The first and second stopping rules are useful for characterizing the performance bounds for the algorithms, but they are usually not implementable. The third rule is practical. The fourth rule will be discussed in Sect. 3 and will be used in this paper. It means that all components of $z - z^*$ in the basis defined the columns of M are reduced by a factor of ϵ in absolute values. It implies all the others.

The steepest descent problem The problem to be solved in this paper is: given $\epsilon > 0$ and $x^0 \in \mathbb{R}^n$, find an integer $k > 0$ and a set $\{\lambda_0, \lambda_1, \ldots, \lambda_{k-1}\}$ of positive numbers such that the point z^k produced by the algorithm (1) satisfies the stopping rule (iv).

The steepest descent method, also called gradient method, was devised by Augustine Cauchy [3]. He studied a minimization problem and described a steepest descent step with exact line search, which we shall call "Cauchy step". For (P_z), the Cauchy step is

$$\lambda_k = \underset{\lambda \geq 0}{\operatorname{argmin}} \; \bar{f}(x^k - \lambda \nabla \bar{f}(x^k)). \tag{2}$$

The steepest descent method with Cauchy steps will be called Cauchy algorithm.

Steepest descent is the most basic algorithm for the unconstrained minimization of continuously differentiable functions, with step lengths computed by a multitude of line search schemes.

The quadratic problem is the simplest non-trivial non-linear programming problem. Being able to solve it is a pre-requisite for any method for more general problems, and this is the first reason for the great effort dedicated to its solution. A second reason is that the optimal solution of (P_z) is the solution of the linear system $Hz = -c$.

It was soon noticed that the Cauchy algorithm generates inefficient zig-zagging sequences. This phenomenon was established by Akaike [1], and further developed by Forsythe [7]. A clear explanation is found in Nocedal, Sartenaer and Zhu [13]. For

some time the steepest descent method was displaced by methods using second order information.

In the last years gradient methods returned to the scene due to the need to tackle large scale problems, with millions of variables, and due to novel methods for computing the step lengths. Barzilai and Borwein [2] proposed a new step length computation with surprisingly good properties, which was further extended to non-quadratic problems by Raydan [15], and studied by Dai [4], Raydan and Svaiter [16] among others. In another line of research, several methods were developed to enhance the Cauchy algorithm by breaking its zig-zagging pattern. These methods, which will not be studied in this paper, are explained in De Asmundis et al. [5,6] and in our paper [10].

None of these papers studies the worst-case performance of the algorithm applied to quadratic problems. This will be our task.

1.1 Complexity results

We are studying performance bounds for first-order methods – methods that use only function and derivative values. The best such method for the quadratic problem is the Conjugate Gradient method (and equivalent Krylov space methods): it is known that no first order method can be faster than it, and its performance bound (which then is the *complexity* of the problem with first order oracle) is given by

$$k \leq \left\lceil \frac{\sqrt{\mathcal{C}}}{2} \log\left(\frac{2}{\varepsilon}\right) \right\rceil \tag{3}$$

iterations to achieve the stopping rule (i), where $\lceil a \rceil$ denotes the smallest integer $\bar{a}$ such that $\bar{a} \geq a$.

This complexity study is found in Nemirovsky and Yudin [12], in Polyak [14], and a detailed proof of (3) is in Shewchuk [17]. See [9] for a tutorial on basic complexity results.

The two most widely known choices of step lengths for which there are complexity studies are:

- The Cauchy step, or exact step (2), the unique minimizer of f along the direction $-g$ with $g = \nabla f(x^k)$, given by

$$\lambda_k = \frac{g^T g}{g^T H g}. \tag{4}$$

- The short step: $\lambda_k = 1/\mu_n$, a fixed step length.

The complexity results for these methods are: the first stopping rule is achieved in

$$k \leq \left\lceil \frac{\mathcal{C}}{4} \log\left(\frac{1}{\varepsilon}\right) \right\rceil, \qquad k \leq \left\lceil \frac{\mathcal{C}}{2} \log\left(\frac{1}{\varepsilon}\right) \right\rceil,$$

respectively for the Cauchy and short step lengths. These bounds are tight, and we are unaware of any steepest descent algorithm with a better worst case performance.

In this paper we show that given μ_1, μ_n and ε, there exist $k \in \mathbb{N}$ and a set $\{\lambda_j \mid j = 1 \ldots k\}$ such that the steepest descent method applied from any initial point with the step lengths λ_j in any order produces a point that satisfies all four stopping rules. The bound to be found in this paper is

$$k = \left\lceil \frac{\cosh^{-1}\left(\frac{2}{\varepsilon}\right)}{\cosh^{-1}\left(1 + \frac{2}{C-1}\right)} \right\rceil \approx \left\lceil \frac{\sqrt{C}}{2} \log\left(\frac{2}{\varepsilon}\right) \right\rceil$$

These two values differ by less than 1 for $\varepsilon < 0.1$ and $C > 2$, as can be seen by plotting both functions. A weaker relationship between both bounds will also be derived in Sect. 3.

The values λ_j will be the inverses of the roots of a Chebyshev polynomial to be constructed in Sect. 3. In Sect. 2 we list some properties of Chebyshev polynomials and hyperbolic functions, which will be used in Sect. 4 to prove the performance bound.

The association of Chebyshev polynomials to steepest descent has been used by numerical analysts, but we are unaware of any complexity studies along this line. See Frank [8].

2 Tools

Let us list some well-known facts on Chebyshev polynomials[1] (see for instance [11]) and a technical result on hyperbolic functions.

Chebyshev polynomials. The Chebyshev polynomial of first kind $T_k(\cdot)$ satisfies:

- For $x \in [-1, 1]$, $T_k(x) = \cos(k \cos^{-1}(x)) \in [-1, 1]$, with $T_k(1) = 1$.
- For $x > 1$, $T_k(x) = \cosh(k \cosh^{-1}(x)) > 1$
- The roots of T_k are

$$x_j = \cos\left(\frac{1+2j}{2k}\pi\right), \; j = 0, 1, \ldots, k-1.$$

- The maximizers of $|T_k(x)|$ in $[-1, 1]$ are $\bar{x}_j = \cos(j\pi/k)$, $j = 0, 1, \ldots, k$.

Hyperbolic functions. The hyperbolic cosine satisfies

$$\cosh(x) = \frac{e^x + e^{-x}}{2} \geq \frac{e^x}{2}, \; \cosh'(x) = \sinh(x), \; \cosh''(x) = \cosh(x).$$

Its Taylor approximation at 0 gives for $x \geq 0$

$$\cosh(x) = 1 + \frac{x^2}{2} + \frac{\cosh(\delta)x^4}{4!} \geq 1 + \frac{x^2}{2},$$

[1] A brief tutorial on Chebyshev polynomials is found in Wikipedia, https://en.wikipedia.org/wiki/Chebyshev_polynomials.

where $\delta \in [0, x]$. It follows that $\cosh(x) - 1$ is well approximated by $x^2/2$ for small values of $x \geq 0$. Setting $y = x^2/2$ and taking the inverse function, it follows that for small values of $y > 0$,

$$\sqrt{2y} \geq \cosh^{-1}(1 + y) \approx \sqrt{2y}.$$

This fact will be useful, and so we give a formal proof here, whose only purpose is to quantify the '$\approx$' above.

Lemma 1 *Let $\bar{x} > 0$ be a given real number. Then for any $x \in [0, \bar{x}]$,*

$$\sqrt{2x} \geq \cosh^{-1}(1 + x) \geq \gamma(\bar{x})\sqrt{2x},$$

where $\gamma(0) = 1$ and for $y > 0$, $\gamma(y) = \dfrac{\cosh^{-1}(1 + y)}{\sqrt{2y}} < 1$.

Proof Note that for $x > 0$, $\cosh^{-1}(1 + x) = \gamma(x)\sqrt{2x}$. All we need is to prove that $x \in \mathbb{R}_+ \mapsto \gamma(x)$ is continuous and decreases for $x > 0$ (and then $\gamma(x) \geq \gamma(\bar{x})$ for $x \in [0, \bar{x}]$).

Continuity: using l'Hôspital's rule,

$$\lim_{x \to 0+} \gamma(x) = \lim_{x \to 0+} \frac{\sqrt{2}}{\sqrt{x + 2}} = 1.$$

Computing the derivative for $x > 0$,

$$\frac{d}{dx}\gamma(x) = \frac{2\sqrt{\frac{x}{x+2}} - \cosh^{-1}(1 + x)}{2x\sqrt{2x}}.$$

The denominator is positive. So we must prove that the numerator

$$N(x) = 2\sqrt{\frac{x}{x + 2}} - \cosh^{-1}(1 + x)$$

is negative for $x > 0$.

Since $N(0) = 0$, $N(x)$ will be negative for $x > 0$ if $N'(x) < 0$. Computing this derivative, for $x > 0$,

$$N'(x) = -\frac{\sqrt{x}}{(x + 2)^{3/2}} < 0,$$

completing the proof. □

3 An infinite dimensional problem

Problem (P_z) has a unique solution z^*. For the analysis, the problem may be simplified by assuming that $z^* = 0$, and so $\bar{f}(z) = z^T H z/2$. The matrix H may be diagonalized

by setting $z = My$, where M has orthonormal eigenvectors of H as columns. Then the function becomes

$$f(y) = \frac{1}{2} y^T D y, \qquad D = \mathrm{diag}(\mu_1, \mu_2, \ldots, \mu_n). \tag{5}$$

M defines a similarity transformation, and hence for $z = My$,

$$\|z\| = \|y\|, \ \|\nabla \bar{f}(z)\| = \|\nabla f(y)\|, \ \nabla \bar{f}(z) = M \nabla f(y).$$

We define the diagonalized problem

$$(P_y) \quad \underset{y \in \mathbb{R}^n}{\text{minimize}} \ f(y) = \frac{1}{2} y^T D y.$$

The steepest descent iterations with step lengths λ_j for minimizing respectively $f(\cdot)$ from the initial point $y^0 = M^T z^0$, and $\bar{f}(\cdot)$ from the initial point z^0, are related by $z^k = My^k$. The stopping rule (iv) for the diagonalized problem becomes $|y_i| \leq \epsilon |y_i^0|$, $i = 1, \ldots, n$, and clearly implies all the others. For instance, if rule (iv) holds at y,

$$f(y) = \frac{1}{2} \sum_{i=1}^{n} \mu_i y_i^2 \leq \frac{\epsilon^2}{2} \sum_{i=1}^{n} \mu_i (y_i^0)^2 = \epsilon^2 f(y^0),$$

and rule (i) holds.

The stopping rules are equivalent for these two sequences. Thus, we may restrict our study to the diagonalized problem.

Given $y^0 \in \mathbb{R}^n$, consider the sequence generated by the steepest descent algorithm defined by

$$y^{j+1} = y^j - \lambda_j \nabla f(y^j), \tag{6}$$

where λ_j is the step length at the j-iteration. As $\nabla f(y^j) = Dy^j$ and D is diagonal, we have, for all $i = 1, \ldots, n$,

$$y_i^{j+1} = (1 - \lambda_j \mu_i) y_i^j.$$

Using this recursively, we obtain

$$y_i^k = \prod_{j=0}^{k-1} (1 - \lambda_j \mu_i) y_i^0. \tag{7}$$

So, each variable may be seen independently, and the order of the step lengths $\{\lambda_0, \ldots, \lambda_{k-1}\}$ is irrelevant with respect to satisfying the stopping criterion. The stopping rule (iv), $|y_i^k| \leq \varepsilon |y_i^0|$ for $i = 1, \ldots, n$, will be satisfied if

$$\left| \prod_{j=0}^{k-1} (1 - \lambda_j \mu_i) \right| \leq \varepsilon, \quad i = 1, \ldots, n,$$

which in its turn is implied by

$$\left|\prod_{j=0}^{k-1}(1-\lambda_j w)\right| \le \varepsilon, \quad w \in [\mu_1, \mu_n].$$

The left hand side defines a polynomial $p(\cdot)$ such that $|p(w)| \le \epsilon$ for $w \in [\mu_1, \mu_n]$ and $p(0) = 1$, with roots $1/\lambda_j$, $j = 1, \ldots, k$. Dividing this inequality by ϵ, it can be satisfied by solving the following infinite dimensional problem:

Find a polynomial $p_k(\cdot)$ of degree $k \in \mathbb{N}$ such that $|p_k(w)| \le 1$ for $w \in [\mu_1, \mu_n]$ and $p(0) = 1/\epsilon$, with roots $1/\lambda_j > 0$, $j = 1, \ldots, k$.

This may be finally be formulated as

(P_w) Given $0 < \mu_1 < \mu_n$, find a polynomial $p_k(\cdot)$ of degree $k \in \mathbb{N}$ such that

$$\max_{w\in[\mu_1,\mu_n]} |p_k(w)| \le 1, \quad p(0) \ge 1/\epsilon,$$

with roots $1/\lambda_j > 0$, $j = 1, \ldots, k$.

The following fact summarizes our development up to now:

Let p_k be a solution of (P_w). Then the steepest descent algorithm with step lengths $\{\lambda_j,\ j = 0, \ldots, k-1\}$ (in any order) applied to (P_z) from any initial point $z^0 \in \mathbb{R}^n$ produces a point z^k that satisfies all four stopping rules.

Solution of ($\mathbf{P}_w$). Our task is to find a polynomial with degree as low as we can which solves the problem. We do it by constructing a Chebyshev polynomial. First, we change variables so that the set $[\mu_1, \mu_n]$ is mapped onto $[-1, 1]$. Set

$$w = \frac{\mu_n - \mu_1}{2}x + \frac{\mu_n + \mu_1}{2}, \quad \text{or} \quad x = \frac{2w}{\mu_n - \mu_1} - \frac{\mu_n + \mu_1}{\mu_n - \mu_1}.$$

Then $x = 0$ for $w = (\mu_1 + \mu_n)/2$, $x = -1$ for $w = \mu_1$ and $x = 1$ for $w = \mu_n$.

For $w = 0$, $x = -\dfrac{\mu_n + \mu_1}{\mu_n - \mu_1} = -\dfrac{C+1}{C-1}$.

With this change of variables, the problem is solved by a Chebyshev polynomial T_k (see Fig. 1). We must satisfy the conditions $|T_k(x)| \le 1$ for $x \in [-1, 1]$, which is trivial, and $|T_k(-(C+1)/(C-1))| \ge 1/\varepsilon$. Due to the symmetry of $|T_k|$ and to the fact that $T_k(x) > 1$ for $x > 1$, this condition is equivalent to

$$\left|T_k\left(\frac{C+1}{C-1}\right)\right| = T_k\left(1 + \frac{2}{C-1}\right) \ge \frac{1}{\varepsilon}.$$

Thus, using the properties of Chebyshev polynomials, we must satisfy

$$\cosh\left(k \cosh^{-1}\left(1 + \frac{2}{C-1}\right)\right) \ge \frac{1}{\varepsilon}.$$

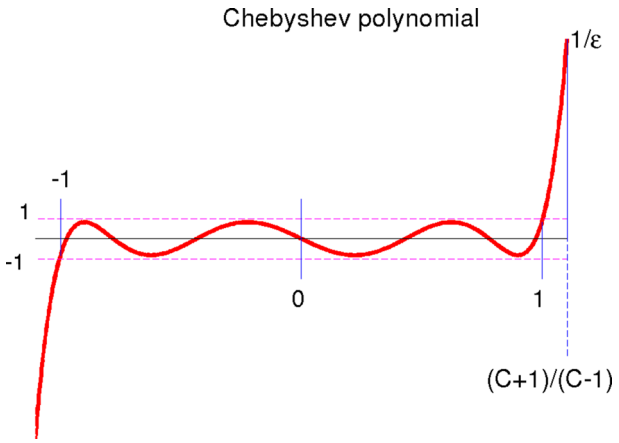

Fig. 1 Chebyshev polynomial

The smallest integer that satisfies this is

$$k(C,\epsilon) = \left\lceil \frac{\cosh^{-1}\left(\frac{1}{\varepsilon}\right)}{\cosh^{-1}\left(1+\frac{2}{C-1}\right)} \right\rceil. \tag{8}$$

We have proved our main result, expressed in the next theorem.

Theorem 1 *Consider the problem (P_z), assume that the eigenvalues of H belong to the interval $[\mu^-, \mu^+]$, $\mu^- > 0$ and set $C = \mu^+/\mu^-$. Then the steepest descent algorithm with step lengths $\{\lambda_j = 1/w_j,\ j = 0, 1, \ldots, k-1\}$, where*

$$w_j = \frac{\mu^+ - \mu^-}{2} \cos\left(\frac{1+2j}{2k}\pi\right) + \frac{\mu^+ + \mu^-}{2}$$

and $k = k(C,\epsilon)$ defined in (8), satisfies the stopping rule in k steps, for any initial point z^0. The step lengths λ_j can be applied in any order.

Proof The result follows directly by a change of variables and the reasoning above.□

A simpler bound. Let us express the bound (8) in the shape $\mathcal{O}(\sqrt{C}\log(1/\varepsilon))$. Numerical values for the parameter appearing in the corollary below will follow its proof.

Corollary 1 *In the conditions of Theorem 1, for $C \geq \bar{C} > 1$,*

$$k \leq \left\lceil \beta(\bar{C})\frac{\sqrt{C}}{2}\log\left(\frac{2}{\varepsilon}\right) \right\rceil,$$

where

$$\beta(\bar{C}) = \frac{2}{\cosh^{-1}(1 + 2/(\bar{C} - 1))\sqrt{\bar{C} - 1}}$$

Proof Let us examine numerator and denominator of (8).

Numerator: since $\cosh(x) > e^x/2$ for $x \in \mathbb{R}$, $\cosh(\log(2/\varepsilon)) > 1/\varepsilon$ for $\varepsilon < 1$, and then $\cosh^{-1}(1/\varepsilon) < \log(2/\varepsilon)$.

Denominator: By Lemma 1, given $\bar{C} > 1$ and $C \geq \bar{C}$ (and hence $2/(C-1) \leq 2/(\bar{C} - 1)$),

$$\cosh^{-1}\left(1 + \frac{2}{C-1}\right) \geq \gamma\left(\frac{2}{\bar{C}-1}\right)\sqrt{\frac{4}{C-1}}, \tag{9}$$

with

$$\gamma\left(\frac{2}{\bar{C}-1}\right) = \frac{\cosh^{-1}\left(1 + \frac{2}{\bar{C}-1}\right)}{\sqrt{\frac{4}{\bar{C}-1}}}$$

This value is near 1, as we see by calculating some values. Let us denote

$$\beta(\bar{C}) = \frac{1}{\gamma(2/(\bar{C}-1))} = \frac{2}{\cosh^{-1}(1 + 2/(\bar{C} - 1))\sqrt{\bar{C} - 1}},$$

and obtain from (9):

$$\frac{1}{\cosh^{-1}(1 + 2/(C-1))} \leq \beta(\bar{C})\frac{\sqrt{C-1}}{2}.$$

Finally, putting together the results for numerator and denominator, (8) is satisfied by

$$k = \left\lceil \beta(\bar{C})\frac{\sqrt{C-1}}{2}\log(2/\varepsilon) \right\rceil \leq \left\lceil \beta(\bar{C})\frac{\sqrt{C}}{2}\log(2/\varepsilon) \right\rceil,$$

for any $C \geq \bar{C}$. □

Numerical values: for $\bar{C} = 2, 4, 10, 100$, the values of β are respectively 1.14, 1.06, 1.02, 1.002. The corresponding values of γ (which appears in the proof) are 0.88, 0.95, 0.98, 0.998.

4 Complexity for unknown eigenvalue bounds

In this section we study the diagonalized problem. All complexity results will also be valid for the original problem.

Let us study the problem (P_y) with no information on the eigenvalues. It is well-known (see for instance [1]) that the Cauchy step $\lambda_C(y)$ from a point $y \in \mathbb{R}^n$ with $g = \nabla f(y)$ satisfies

$$\lambda_C(y) = \frac{g^T g}{g^T D g}, \quad \frac{1}{\lambda_C} \in [\mu_1, \mu_n].$$

So, we may assume that a number $\mu \in [\mu_1, \mu_n]$ is known.

The Cheby algorithm. Consider the problem (P_y) and values $l < u$, not necessarily bounds for the eigenvalues. We shall call 'Cheby' algorithm the steepest descent with steps λ_i given by Theorem 1 with $\mu^- = l$ and $\mu^+ = u$. After its application from the initial point y^0, $g^0 = \nabla f(y^0)$, we obtain a point $y = Cheby(y^0, l, u, \epsilon)$ such that

$$|y_j| \leq \epsilon |y_j^0|, \quad |g_j| \leq \epsilon |g_j^0|$$

for $j \in \{1, \ldots, n\}$ such that $l \leq \mu_j \leq u$.

The stopping rule. Let us use the stopping rule (iii), $\|\nabla f(y)\| \leq \|\nabla f(y^0)\|$, because the other rules are not implementable. The following scheme expands an interval $[l_i, u_i]$ and applies the Cheby algorithm, until the stopping rule is satisfied. Note that we do not propose this as a practical method: our intent is to show a complexity bound.

Algorithm
Data: $y^0 \in \mathbb{R}^n$, $g^0 = \nabla f(y^0)$, $\epsilon > 0$, $i = 1$, $\mu \in [\mu_1, \mu_n]$.
$l_1 = \mu/2$, $u_1 = 2\mu$, $C_1 = u_1/l_1 = 4$.
Apply the Cheby algorithm to find $y^1 = Cheby(y^0, l_1, u_1, \epsilon/2)$.
$g = \nabla f(y^1)$.
WHILE $\|g\| > \epsilon \|g^0\|$

$$\mu_C = \frac{1}{\lambda_C(y^i)} = \frac{g^T D g}{g^T g}.$$

IF $\mu_C > u_i/2$, set $u_{i+1} = 4u_i$, $l_{i+1} = l_i$.
ELSE set $l_{i+1} = l_i/4$, $u_{i+1} = u_i$.
Apply the Cheby algorithm to find $y^{i+1} = Cheby(y^0, l_{i+1}, u_{i+1}, \epsilon/2)$, and set $g = \nabla f(y^{i+1})$.
$i = i + 1$, $C_i = u_i/l_i = 4^i$.
END
$\bar{l} = l_i$, $\bar{u} = u_i$, $\bar{i} = i$.

Lemma 2 *At an iteration i of the algorithm:*

(i) If $u_i \geq 2\mu_n$ then $\bar{u} = u_i$ (u_i stops increasing).
(ii) If $l_i \leq \mu_1$ then $\bar{l} = l_i$ (l_i stops decreasing).
(iii) When the algorithm stops, $\bar{l} \geq \mu_1/4$, $\bar{u} \leq 8\mu_n$ and $C_{\bar{i}} \leq 32C$.

Proof Assume without loss of generality that $\|g^0\| = 1$, and let us examine an iteration i (if $\|g^1\| \leq \epsilon$, the result is trivial).

(i) Assume that $u_i \geq 2\mu_n$. We know that $\mu_C \in [\mu_1, \mu_n]$, and then $\mu_C \leq \mu_n \leq u_i/2$. By construction, $u_{i+1} = u_i$ (u_i stops increasing).
(ii) Assume that $l_i \leq \mu_1$. We must prove that $\mu_C > u_i/2$.

Let $\bar{j} = \max\left\{j \in \{1, \ldots, n\} \mid \mu_j \leq u_i\right\}$. By Theorem 1, after the application of the Cheby algorithm with precision $\epsilon/2$,

$$|g_j| \leq \frac{\epsilon}{2}|g_j^0|, \quad \text{for } j = 1, 2, \ldots, \bar{j}.$$

Hence

$$\sum_{j \leq \bar{j}} g_j^2 \leq \frac{\epsilon^2}{4} \sum_{j \leq \bar{j}} (g_j^0)^2 \leq \frac{\epsilon^2}{4} < \frac{\|g\|^2}{4},$$

and consequently

$$\sum_{j > \bar{j}} g_j^2 > \frac{3}{4}\|g\|^2 > \frac{3}{4}\epsilon^2 > \frac{1}{2}\epsilon^2.$$

It follows that

$$\sum_{j=1}^{n} \mu_j g_j^2 \geq u_i \sum_{j > \bar{j}} g_j^2 > \frac{u_i}{2}\|g\|^2.$$

Dividing both sides by $\|g\|^2$,

$$\mu_C = \frac{\sum_{j=1}^{n} \mu_j g_j^2}{\|g\|^2} > \frac{u_i}{2},$$

proving (ii).

(iii) Lower bound: if $\bar{l} = l_1$, the result is true because $l_1 = \mu/2 \geq \mu_1/2$. Otherwise, $\bar{l} = l_i/4$ for some i with $l_i > \mu_1$, and hence $\bar{l} > \mu_1/4$. Upper bound: if $\bar{u} = u_1$, $\bar{u} = 2\mu \leq 2\mu_n$. Otherwise, $\bar{u} = 4u_i$ for some i such that $u_i < 2\mu_C \leq 2\mu_n$, because $\mu_C \in [\mu_1, \mu_n]$. Hence $\bar{u} \leq 8\mu_n$, completing the proof. □

Lemma 3 *Let p be the smallest integer such that $32C \leq 4^p$. The algorithm will stop with $\bar{i} \leq p$, with a total number of steepest descent steps satisfying*

$$K = \sum_{i=1}^{p} k_i \leq 12\sqrt{C}\log(4/\epsilon).$$

Proof The number of steps at each iteration with $C \geq 4$ and precision $\epsilon/2$ is given by Corollary 1: as $\lceil a \rceil \leq a + 1$ for $a \in \mathbb{R}$,

$$k_i \leq 1.06\frac{\sqrt{C_i}}{2}\log(4/\epsilon) + 1 \leq \sqrt{C_i}\log(4/\epsilon).$$

The last inequality is easily checked for $C_i \geq 4$ and $\epsilon < 1$, because $1.06\log(4) + 1 < 2\log(4)$.

The number of iterations satisfies $\bar{i} \leq p$, because $4^{\bar{i}} = C_{\bar{i}} \leq 32C$. Then $C \geq 4^{\bar{i}-2}/2$,

$$\sqrt{C} \geq 2^{\bar{i}-2}/\sqrt{2}. \tag{10}$$

In $\bar{i}$ iterations, the total number of steepest descent steps is

$$K = \sum_{i=1}^{\bar{i}} k_i \leq \sum_{i=1}^{\bar{i}} \sqrt{C_i}\log(4/\epsilon) = \log(4/\epsilon)\sum_{i=1}^{\bar{i}} 2^i < \log(4/\epsilon)2^{\bar{i}+1}.$$

Finally, using (10), $2^{\bar{i}+1} \leq 8\sqrt{2}\sqrt{C} \leq 12\sqrt{C}$, completing the proof. □

Remark If either a good lower bound or upper bound for the eigenvalues is known, the scheme becomes much easier, just update the unknown bound by multiplying or dividing it by 4 in each iteration. The bound on K will have a lower constant:

$$K \leq 3\sqrt{C}\log(2/\epsilon).$$

Doing this simultaneously for both bounds would increase the complexity: the "difficult" thing was to decide which bound to change at each iteration, and it was done with the help of the inverse Cauchy step. If a scheme like this is to be used in practice, note that there must be more efficient ways of updating the upper bound (see for instance [5] for methods for estimating short steps). Estimating the smallest eigenvalue is more difficult. We assumed that all iterations start at the same initial point, which does not seem reasonable in practice: one should start each iteration i of the scheme from y^i.

5 Toward possible applications

The purpose of this paper is the establishment of a theoretical optimal performance bound. Nevertheless, some practical hints can be suggested from these results. These hints, together with efficient versions of steepest descent and practical methods for estimating the bounds $\bar{l}$ and $\bar{u}$, are explored in reference [10].

The Cauchy algorithm tends to generate steps that cycle around two limit points. Our results say that the steps should be spaced according to the Chebyshev roots: repeating step lengths is not as efficient as keeping them apart. They also suggest that the step lengths should have a sinusoidal distribution, with a higher density of step lengths near the extremities of the interval $[1/\mu_n, 1/\mu_1]$.

This is illustrated by the following strategy: assuming that μ_1, μ_n and ϵ are known, generate a list $L = l_1, l_2, \ldots, l_k$ of Chebyshev steps as in Theorem 1. Choose any steepest descent algorithm (Cauchy, short steps, Barzilai-Borwein,…) and do the following:

At iteration k, compute λ_k by the algorithm.
Find the item $\bar{i}$ in L that minimizes $|\lambda_k - l_i|$.
Set $\lambda_k = l_{\bar{i}}$ and remove the item $l_{\bar{i}}$ from L.

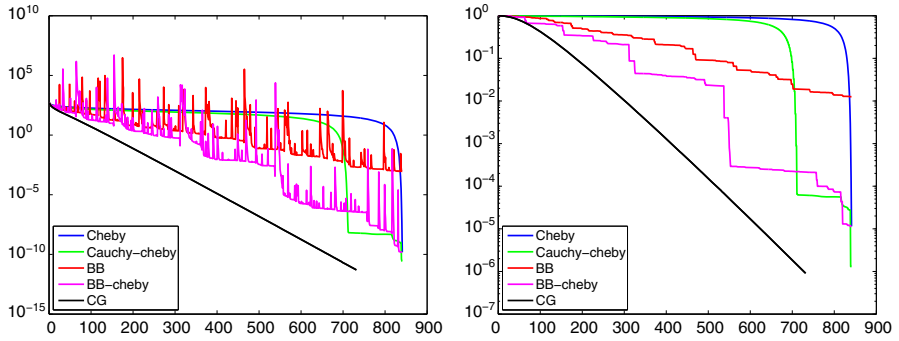

Fig. 2 Algorithms using Chebyshev roots

The algorithm will necessarily stop for $k \leq k(\mu_n/\mu_1, \epsilon)$. This procedure will hopefully enhance any steepest descent method, by forcing it to use Chebyshev steps. In Fig. 2 we show the effect of this enhancement on a run of the Cauchy and Barzilai-Borwein algorithms applied to a problem with logarithmically distributed eigenvalues, $C = 10^6$, $n = 10^4$, $\epsilon = 10^{-5}$. The plots show the evolution of function values for the algorithm with ordered Chebyshev roots, modified Cauchy (original Cauchy is too slow, not shown), original Barzilai-Borwein and enhanced Barzilai-Borwein. We see that the enhancement really did improve the convergence in this example.

At left, the figure shows function values; at right, $\min\left\{\|x^j\|_\infty \mid j \leq k\right\}$.

Acknowledgments I thank an anonymous referee for his great contribution to improving this paper.

References

1. Akaike, H.: On a successive transformation of probability distribution and its application to the analysis of the optimum gradient method. Ann. Inst. Stat. Math. Tokyo **11**, 1–17 (1959)
2. Barzilai, J., Borwein, J.M.: Two-point step size gradient methods. IMA J. Numer. Anal. **8**, 141–148 (1988)
3. Cauchy, A.: Méthode générale pour la résolution des systèmes d'équations simultanées. Comp. Rend. Acad. Sci. Paris **25**, 536–538 (1847)
4. Dai, Y.H.: Alternate step gradient method. Optimization **52**(4–5), 395–415 (2003)
5. de Asmundis, R., di Serafino, D., Hager, W., Toraldo, G., Zhang, H.: An efficient gradient method using the Yuan steplength. Comput. Optim. Appl. **59**(3), 541–563 (2014)
6. de Asmundis, R., di Serafino, D., Riccio, R., Toraldo, G.: On spectral properties of steepest descent mehtods. IMA J. Numer. Anal. **33**, 1416–1435 (2013)
7. Forsythe, G.E.: On the asymptotic directions of the s-dimensional optimum gradient method. Numerische Mathematik **11**, 57–76 (1968)
8. Frank, W.L.: Solution of linear systems by Richardson's method. J. ACM **7**(3), 274–286 (1960)
9. Gonzaga, C.C., Karas, E.W.: Complexity of first-order methods for differentiable convex optimization. Pesqui. Opera. Brazil **34**, 395–419 (2014)
10. Gonzaga, C.C., Schneider, R.M.: On the steepest descent algorithm for quadratic functions. Comput. Optim. Appl. (2015). doi:10.1007/s10589-015-9775-z
11. Mason, J., Handscomb, D.: Chebyshev Polynomials. Chapman and Hall, New York (2003)
12. Nemirovski, A.S., Yudin, D.B.: Problem Complexity and Method Efficiency in Optimization. Wiley, New York (1983)

13. Nocedal, J., Sartenaer, A., Zhu, C.: On the behavior of the gradient norm in the steepest descent method. Comput. Optim. Appl. **22**, 5–35 (2002)
14. Polyak, B.T.: Introduction to Optimization. Optimization Software Inc., New York (1987)
15. Raydan, M.: The Barzilai and Borwein gradient method for large scale unconstrained minimization problem. SIAM J. Optim. **7**, 26–33 (1997)
16. Raydan, M., Svaiter, B.: Relaxed steepest descent and Cauchy–Barzilai–Borwein method. Comput. Optim. Appl. **21**, 155–167 (2002)
17. Shewchuk, J.R.: An introduction to the conjugate gradient method without the agonizing pain. Technical report, School of Computer Science, Carnegie Mellon University (1994)