On the steepest descent algorithm for quadratic functions

Clóvis C. Gonzaga* Ruana M. Schneider†

July 9, 2015

Abstract

The steepest descent algorithm with exact line searches (Cauchy algorithm) is inefficient, generating oscillating step lengths and a sequence of points converging to the span of the eigenvectors associated with the extreme eigenvalues. The performance becomes very good if a short step is taken at every (say) 10 iterations. We show a new method for estimating short steps, and propose a method alternating Cauchy and short steps. Finally, we use the roots of a certain Chebyshev polynomial to further accelerate the method.

1 Introduction

We study the quadratic minimization problem

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

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

$$0 < d_1 < d_2 < \ldots < d_n,$$

and condition number $C = d_n/d_1$. The problem has a unique solution $z^* \in \mathbb{R}^n$.

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

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

The only distinction among different steepest descent algorithms is in the choice of the step lengths λ_k.

For the analysis, the problem may be simplified by assuming that $z^* = 0$, and so $f(z) = z^T Hz/2$. The matrix H may be diagonalized by setting $z = Mx$, where M has orthonormal eigenvectors of H as columns. Then the function becomes

$$f(x) = \frac{1}{2} x^T D x, \qquad D = \text{diag}(d_1, d_2, \ldots, d_n), \tag{2}$$

so that for $z \in \mathbb{R}^m$, $\bar{f}(z) = f(M^T z)$. M defines a similarity transformation, and hence for $z = Mx$,

$$\|z\| = \|x\|\,, \ \left\|\nabla \bar{f}(z)\right\| = \|\nabla f(x)\|\,, \ \nabla \bar{f}(z) = M \nabla f(x),$$

using throughout the paper the 2-norm $\|z\|^2 = z^T z$.

[2]Department of Mathematics, Federal University of Santa Catarina, Florianópolis, SC, Brazil; e-mail: ccgonzaga1@gmail.com. The author was partially supported by CNPq under grant 308413/2009-1.

[2]Department of Mathematics, Federal University of Santa Catarina, Florianópolis, SC, Brazil; e-mail: ruanamaira@gmail.com.

We define the diagonalized problem

$$(P) \qquad \underset{x\in\mathbb{R}^n}{\text{minimize}}\ f(x) = \frac{1}{2}x^T Dx.$$

The steepest descent iterations with step lengths λ_k for minimizing respectively $f(\cdot)$ from the initial point $x^0 = M^T z^0$, and $\bar{f}(\cdot)$ from the initial point z^0, are related by $z^k = Mx^k$. Thus, we may restrict our study to the diagonalized problem.

The steepest descent method, was devised by Augustine Cauchy [**?**] in 1847. He studied the quadratic minimization problem, using in each iteration the "Cauchy step"

$$\lambda_k = \underset{\lambda\geq 0}{\text{argmin}}\ f(x^k - \lambda\nabla f(x^k)). \tag{3}$$

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 [**?**] in 1959, and further developed by Forsythe [**?**]. A clear explanation of its consequences is found in Nocedal, Sartenaer and Zhu [**?**]. 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 [**?**] proposed a new step length computation with surprisingly good properties, which was further extended to non-quadratic problems by Raydan [**?**], and studied by Dai [**?**], Raydan and Svaiter [**?**], Birgin, Martínez and Raydan [**?**], among others. In another line of research, several methods were developed to enhance the Cauchy algorithm by breaking its zig-zagging pattern. For the latest developments of this subject, see Asmundis et al. [**?**, **?**].

This paper has two goals. In section 2 we write complete proofs for the main properties of the Cauchy algorithm, including the classical results by Akaike, hopefully simplifying the treatment. In section 3 we develop new ways of breaking the oscillatory behavior of the algorithm and compare several enhancements.

We do not attempt to generalize these procedures to non-quadratic or to constrained problems.

2 The Cauchy algorithm

In this section we intend to state and prove the main asymptotic properties of the Cauchy algorithm. These properties will be summarized in Theorem **??** below, and then the whole section is dedicated to prove it and describe some other properties of the sequences generated by the algorithm. The reader may well accept the theorem without going through the proofs, and proceed to the next section. The reason for proving these results, which are not original, is that the original Akaike and Forsythe papers, not being aimed exclusively at these results, are frequently not considered easy to read. We also prove simplified versions of results in [**?**, **?**, **?**], with a unified notation.

Given a point $x \in \mathbb{R}^n$, with $g = \nabla f(x)$, the Cauchy step from x defined as

$$x^+ = (I - \lambda D)x, \quad g^+ = (I - \lambda D)g.$$

where

$$\lambda = \underset{\sigma\geq 0}{\text{argmin}}\ f(x - \sigma g) = \frac{g^T g}{g^T Dg}.$$

If follows from the definition of λ that $g^T g^+ = 0$.

We shall frequently use the value $\mu = 1/\lambda = (g^T Dg)/\|g\|^2$. The following sequences will be associated with an application of the steepest descent algorithm from an initial point x^0 with $g^0 = \nabla f(x^0)$:
(x^k): iterates: $x^{k+1} = (I - \lambda_k D)x^k$.
(g^k): gradients: $g^{k+1} = (I - \lambda_k D)g^k$.
(λ_k): Cauchy step length from x^k.
(μ_k): $\mu_k = 1/\lambda_k$.
(y^k): normalized gradients $y^k = g^k/\left\|g^k\right\|$.
Assumption: We assume that g^0 has no zero components. This is done because if a component is null, then it will remain null forever.

It is well known that $x^k \to 0$ and $g^k \to 0$. The main goal of this section is the study of the asymptotic properties of the sequence of normalized gradients y^k, following Akaike [**?**] and Forsythe [**?**].

2.1 The main theorem

The theorem below summarizes some of the main results of the cited references. It is stated, commented, and then proved in the following section.

Theorem 1. *Consider the sequences with elements $x^k, g^k, y^k \in \mathbb{R}^n$, λ_k and $\mu_k = 1/\lambda_k$ generated by the Cauchy algorithm from an initial point $x^0 \in \mathbb{R}^n$, assuming that $n > 1$ and $x_1^0, x_n^0 \neq 0$. Then there exist $\mu, \mu' \in (d_1, d_n)$, $r, r' \in \mathbb{R}^n$ and $\alpha \in (0,1)$ such that*
(i) $\mu_{2k} \to \mu$, $\mu_{2k+1} \to \mu'$, with $\mu + \mu' = d_1 + d_n$.
(ii) $y_i^k \to 0$ for $i = 2, 3, \ldots, n-1$.
(iii) $y^{2k} \to r$, $y^{2k+1} \to r'$, with $r, r' \in \mathcal{L}(e_1, e_n)$, where $\mathcal{L}(e_1, e_n)$ denotes the subspace generated by e_1 and e_n.
(iv) $\lim\limits_{k\to\infty} \left|\dfrac{g_i^{k+2}}{g_i^k}\right| \leq 1$ for all $i = 1, \ldots, n$ such that $g_i^k \neq 0$ for $k \in \mathbb{N}$.
(v) $\lim\limits_{k\to\infty} \dfrac{g_i^{k+2}}{g_i^k} = \lim\limits_{k\to\infty} \dfrac{\|g^{k+2}\|}{\|g^k\|} = \alpha$ for $i = 1$ and $i = n$, with

$$\alpha \geq 1 - 2\frac{d_1 d_n}{\tilde{d}^2 - \delta^2},$$

where $\tilde{d} = (d_1 + d_n)/2$ and $\delta = \min\{|d_i - \tilde{d}| \,|\, i = 1, \ldots, n\}$.
(vi) The limiting values for μ, μ' are bounded by

$$\mu_{min} = \tilde{d} - \sqrt{(\tilde{d}^2 + \delta^2)/2} \leq \mu, \mu' \leq \tilde{d} + \sqrt{(\tilde{d}^2 + \delta^2)/2} = \mu_{max}.$$

(vii) For large k, $|g_n^k|$ oscillates around the values of $|g_1^k|$, with

$$\left|\frac{r_1}{r_n}\right| = \left|\frac{r_n'}{r_1'}\right| = \frac{\mu - d_1}{\mu' - d_1}.$$

One of the scopes of this paper is to present a complete proof of this theorem. The proof will be postponed to the next section, after a qualitative analysis of the asymptotic properties of the algorithm.

For this discussion, assume that the condition number C is large, say, $C >> 10$, and that the space dimension is $n > 2$, possibly large. The variables will be loosely classified as light, medium and heavy, associated respectively with small, medium and large eigenvalues.

In a typical step, $g_i^+ = (1 - \lambda d_i)g_i$, $\lambda = 1/\mu$. Let us comment on step sizes

Short steps: safe and frequent. We call 'short' a step with $\mu \geq d_n/2$. Short steps are:

- Harmless: all $|g_i|$ decrease.
- Efficient for heavy variables.
- Inefficient for light variables: if $\lambda << 1/d_i$, then $|g_i^+|/|g_i| = |1 - \lambda d_i|$ is near 1.

We conclude that short steps reduce heavy variables, with little effect on light variables. About one half of the steps will be short.

Large steps: dangerous but needed. We call 'large' a step with $\mu < d_n/2$. Large steps are:

- Dangerous: the heavy variables ($|g_n|$) increase.
- Reasonably efficient for light variables. Very light variables (d_i near d_1) can only be reduced by very large steps. For example, if $d_1 = 0.001$, $d_n = 1$, a step $\lambda = 500 >> 2/d_n$ produces $g_1^+ = g_1/2$ and $g_n^+ = -499g_1$.

We conclude that only very large steps are efficient for reducing $|g_1|$, and they increase very much the heavy variables. "Medium" steps are also inefficient for the light variables.

Cauchy steps: safe and essential. Cauchy steps satisfy $\mu \in (d_1, d_n)$, and hence with $g_i^+ = (1 - d_i/\mu)g_i$, g_1 always decreases in absolute value, keeping the sign unchanged; g_n changes sign at all iterations, and increases in absolute value if the step is large.

The Cauchy algorithm generates a sequence of steps $\lambda_k = 1/\mu_k$ that oscillate, with (μ_k, μ_{k+1}) converging to the two limit points μ, μ', with $\mu + \mu' = d_1 + d_n$. Then (say) $\mu < (d_1 + d_n)/2$ and $\mu' > (d_1 + d_n)/2 > d_n/2$. So, for large k, the steps alternate between short and large,

Cauchy steps are in general "medium short", i.e., they are not very large.

The sequence of normalized gradients (y^k) zig-zags, so that the pairs of consecutive iterations (y^{2k}, y^{2k+1}) converge to a pair of vectors (r, r') in $\mathcal{L}(e_1, e_n)$. In the original space, this is the subspace generated by the eigenvectors associated with the two extreme eigenvalues d_1, d_n. The step lengths also zig-zags, so that $(\lambda_{2k}, \lambda_{2k+1}) \to (\lambda, \lambda')$, or equivalently $(\mu_{2k}, \mu_{2k+1}) \to (\mu, \mu')$, with $\mu + \mu' = d_1 + d_n$.

For large k, the absolute values of all gradient components must decrease in each pair of iterations. In fact, g_1^k never changes sign and decreases in absolute value in all iterations; g_n^k changes size in all iterations, while $|g_n^k|$ oscillates around $|g_1^k|$ for large k.

Let us follow the iterates of the Cauchy algorithm for an example with 50 variables, with eigenvalues $0.01 \leq d_i \leq 1$, $x_i^0 = 1/\sqrt{d_i}$.

Here are some numerical values for this example:

$$\mu_{min} = 0.148, \quad \mu = 0.467, \quad \mu' = 0.543, \quad \mu_{max} = 0.862.$$

$$\alpha = 0.922, \quad |r_n/r_1| = 0.926, \quad |r'_n/r'_1| = 1.080.$$

The value α in item (v) deserves some attention: if C is large and δ is small (there exists an eigenvalue near $\tilde{d}$), then $\alpha > 1 - 8d_1 = 1 - 8/C$, very near the worst case convergence rate known for the Cauchy algorithm, given by $((C-1)/(C+1))^2 \approx 1 - 4/C$, counting pairs of iterations.

The figures show the values of $|g_i|$, with the eigenvalues in the horizontal axis. Each figure shows three iterates g^k, g^{k+1}, g^{k+2}, with vertical lines at the points μ_k, μ_{k+1} for $k = 0$, $k = 5$ and $k = 12$. One sees that the gradient components associated with eigenvalues near μ_k are substantially reduced in the iteration k. The last figure reproduces the third one using a logarithmic scale in the horizontal axis.

In the beginning iterations the heavy variables are reduced (in absolute value), and soon the oscillatory pattern is achieved. The values of μ_k, μ_{k+1} converge to the values μ, μ' as in the theorem,

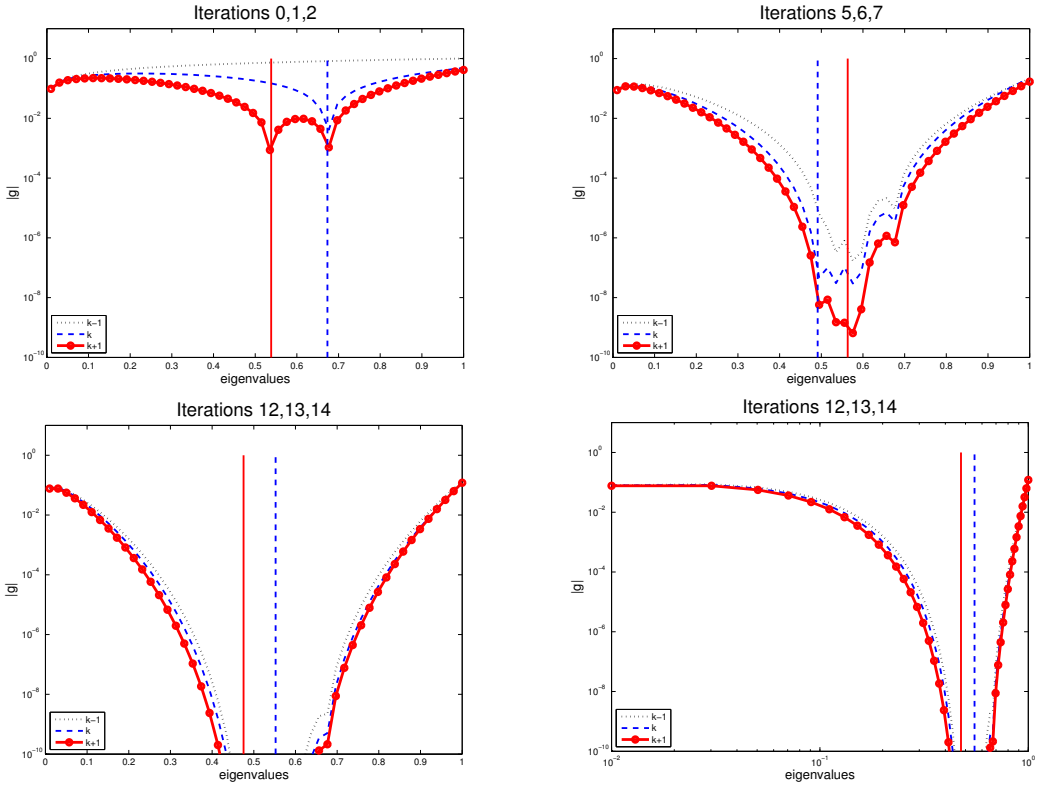

Figure 1: Absolute values of gradient components in consecutive iterations, showing the values of $\mu_k = 1/\lambda_k$.

the medium gradient components are reduced and $|g_n|$ oscillates around $|g_1|$, being slowly reduced at each pair of iterations. In the last figure we see that the algorithm affects the "medium heavy" variables, with little effect of the light variables. Large steps (small values of μ) are needed to reduce the very light components, and they never happen.

Fig. **??** plots the evolution of μ_k (left) and of $|g_1|$ and $|g_n|$, showing the oscillatory behavior. The step lengths stabilize very fast. As $|g_n^0| >> |g_1^0|$, $|g_n|$ decreases (in an oscillatory fashion) in the beginning iterations to match $|g_1|$.

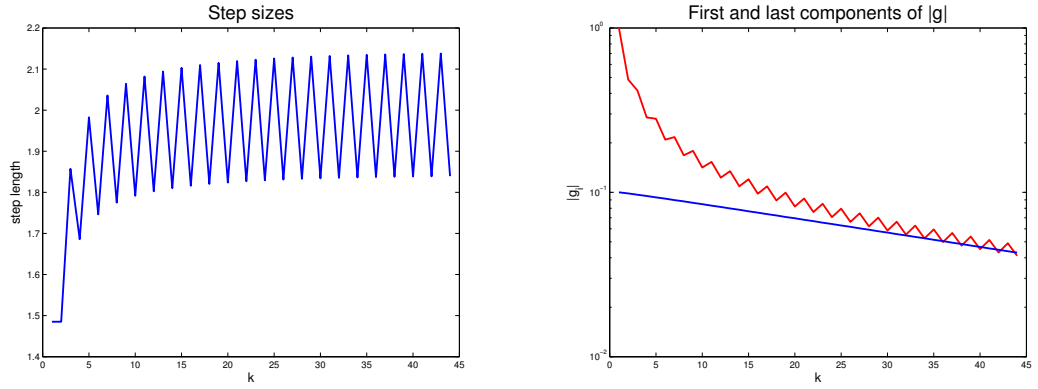

Figure 2: Behavior of λ_k and of $\left|g_1^k\right|, \left|g_n^k\right|$. λ_k oscillates, $\left|g_1^k\right|$ decreases and $\left|g_n^k\right|$ oscillates.

2.2 The Akaike-Forsythe results: proof of the main theorem

The results in this subsection are a simplified version of those in Forsythe [**?**], reduced to the case studied in this paper. We follow his proofs almost step by step, hopefully simplifying the treatment.

The study is simplified by the introduction of the following sequence:
(w^k): defined by $w^0 = g^0$, $w^{k+1} = (\mu_k I - D)w^k$.

It is easy to see that $w^k = g^k \prod_{i=0}^{k-1} \mu_k$. The sequence (w^k) does not seem to have much interest in itself, but as we shall see it is very handy for the study of each iteration of the algorithm. Note that the normalized gradients satisfy $y^k = g^k/\left\|g^k\right\| = w^k/\left\|w^k\right\|$.

Notation. We are mostly interested in the sequence of gradient vectors. When studying an iteration, we use the following maps, defined for a vector $g \in \mathbb{R}^n$, $g \neq 0$:

$$\mu(g) = \frac{g^T D g}{g^T g} \in [d_1, d_n], \quad g^+(g) = (I - \frac{1}{\mu(g)} D)g, \quad y^+(g) = \frac{g^+(g)}{\|g^+(g)\|}. \tag{4}$$

Whenever no confusion is possible, we omit the argument. If $g^+ \neq 0$, we also define $\mu^+ = \mu(g^+)$ and g^{++}. The fact that $\mu \in [d_1, d_n]$ is a standard result in linear algebra. If g has at least two non-zero components, then $\mu(g) \in (d_1, d_n)$.

When dealing with the sequence (w^k), we use the following notation:

$$w' = (\mu I - D)w, \quad w'' = (\mu^+ I - D)w', \tag{5}$$

when $w' \neq 0$. Note that $\mu(g^k) = \mu(w^k) = \mu(y^k)$.

By construction of the Cauchy steps, $g^T g^+ = w^T w' = y^T y^+ = 0$.

Let us begin by isolating a special case (and later prove that it never happens):

Lemma 1. *Suppose that $\bar{w} = \alpha e_p$ for some $p \in \{1, 2, \ldots, n\}$, $\alpha \neq 0$. Then*

$$\mu(\bar{w}) = d_p, \quad \bar{w}' = 0, \quad \lim_{w \to \bar{w}} \frac{w'(w)}{\|w\|} = 0.$$

Proof. Clearly, $\mu(\bar{w}) = e_p^T D e_p = d_p$, and then $\bar{w}' = \alpha(d_p I - D)e_p = 0$, because $De_p = d_p e_p$. The last result follows from the continuity of $\mu(\cdot)$ (and hence of $w'(\cdot)$) at $\bar{w} \neq 0$. □

Lemma 2. *Let $p < q$ be the first and last non-zero components of a vector $w \in \mathbb{R}^n$. Then $\mu(w) \in (d_p, d_q)$ and $w'_p \neq 0$, $w'_q \neq 0$. For the Cauchy algorithm, if $g_1^0 \neq 0$ and $g_n^0 \neq 0$, then the same is true for all w^k and g^k.*

Proof. The result follows from (**??**) and the lemma above: $\mu(w) = d_p$ (or $\mu(w) = d_q$) can only occur if w_1 (or w_n) is its only non-zero component. The result on the sequence (g^k) follows by induction. □

From the lemmas above we see that the behavior of the Cauchy algorithm is anomalous at points g on the coordinate axes. Let us call such vectors "scalar", and eliminate them by defining the sets

$$\Omega = \{g \in \mathbb{R}^n \,|\, g \text{ is not scalar}\}, \quad Z = \{y \in \Omega \,|\, \|y\| = 1\}.$$

Continuous maps: now all the maps defined above $(\mu, \mu^+, g^+, g^{++}, \ldots)$ are continuous on Ω.

We use the simplified notation $\mu = \mu(w), \ldots$. We assume that the Cauchy algorithm is applied to the problem (P) from a point $g^0 \in \Omega$ with no null component.

Theorem 2. *Let (w^k) be generated by the Cauchy algorithm. Then for $k = 0, 1, \ldots$,*

$$\frac{\|w^{k+1}\|}{\|w^k\|} = \cos\psi_k \frac{\|w^{k+2}\|}{\|w^{k+1}\|} \leq \frac{\|w^{k+2}\|}{\|w^{k+1}\|},$$

where ψ_k is the angle between w^k and w^{k+2}.

Proof. Using our simplified notation for $w = w^k$, we first prove that $\|w^+\|^2 = w^T w^{++}$: We have

$$\begin{aligned} w^T w^{++} &= w^T(\mu I - D)(\mu^+ I - D)w \\ \|w^+\|^2 &= w^T(\mu I - D)(\mu I - D)w. \end{aligned}$$

Subtracting, we obtain

$$w^T w^{++} - \|w^+\|^2 = w^T(\mu I - D)(\mu^+ - \mu)w = (\mu^+ - \mu)w^T w^+ = 0.$$

Now, using this and Cauchy-Schwartz,

$$w^T w^{++} = \|w\|\|w^{++}\|\cos(\psi) = \|w^+\|^2.$$

which divided by $\|w\|\|w^+\|$ gives the desired result, completing the proof. □

Theorem 3. *Let (w^k) be generated by the Cauchy algorithm from $g^0 \in \Omega$, and let ψ_k be the angle between w^k and w^{k+2}. Then:*
(i) The the sequence (ϕ_k), with $\phi_k = \|w^{k+1}\|/\|w^k\|$ increases towards a limit $L > 0$.
(ii) $\lim_{k\to\infty} \psi_k = 0$ *and* $\lim_{k\to\infty} \left\|y^k - y^{k+2}\right\| = 0$.
(iii) All limit points of (y^k) are in Z.

Proof. (i) The sequence increases as a consequence of Theorem **??**. Let us show that it is bounded: given $k \in \mathbb{N}$, $w^{k+1} = (\mu_k I - D)w^k$, and then

$$\|w^{k+1}\| \le \|\mu_k I - D\|\|w^k\|.$$

We know that $\mu_k \in (d_1, d_n)$, and hence

$$\|\mu_k I - D\| \le \max\{|\mu_k - d_i| \mid i = 1, 2, \ldots, n\} \le d_n - d_1, \quad \phi_k = \frac{\left\|w^{k+1}\right\|}{\left\|w^k\right\|} \le d_n - d_1.$$

Thus the sequence is increasing and bounded, implying that it converges to some value L.
(ii) From Theorem **??**, $\phi_k = \cos(\psi_k)\phi_{k+1}$, and taking limits,

$$\cos(\psi_k) = \frac{\phi_k}{\phi_{k+1}} \to 1,$$

showing that $\psi_k \to 0$. As $\|y^k\| = \|y^{k+2}\| = 1$ and ψ_k is the angle between these vectors, (ii) holds.
(iii) Assume that $y^k \overset{\mathcal{K}}{\to} r$, with $\mathcal{K} \subset \mathbb{N}$. If $r \notin Z$, then necessarily $r = \pm e_p$ for some $p = 1, \ldots, n$. Then, by Lemma **??**,

$$\lim_{y^k \to r} \frac{w'(y^k)}{\|y^k\|} = 0.$$

This contradicts (i), because

$$\frac{\|w^{k+1}\|}{\|w^k\|} = \frac{\|w'(y^k)\|}{\|y^k\|},$$

which converges to $L > 0$, completing the proof. □

Summing up, the algorithm generates a sequence (y^k) with $y^k = \dfrac{g^k}{\|g^k\|} = \dfrac{w^k}{\|w^k\|}$, whose limit points are all in the set Z. Then all transformations like $\mu(y), w^+(y), \ldots$ are continuous at any limit point r of (y^k).

For instance, assuming that $y^k \overset{\mathcal{K}}{\to} r$, and defining $r' = w'(r) = (\mu(r)I - D)r$, $r'' = w''(r) = (\mu^+(r)I - D)r'$, $r^+ = r'/\|r'\|$, $r^{++} = w''/\|w''\|$, we deduce that $y^{k+1} \overset{\mathcal{K}}{\to} r^+$ and $y^{k+2} \overset{\mathcal{K}}{\to} r^{++} = r$. This last equality follows from Theorem **??**. Also from this theorem, we see that $r'' = L^2 r$.

Remarks. There should be no confusion: given $r \in Z$ with unit norm, $r' = (\mu I - D)r$ has $\|r'\| = L$, and r^+ has unit norm. Also $r^{++} = r$.

The sequence (y^{2k}) satisfies $\left\|y^{k+2} - y^k\right\| \to 0$. It is then easy to prove that either it is convergent or it has no isolated limit point (actually, the set of limit points is a *continuum*).

Theorem 4. *Let (y^k) be the sequence of normalized gradients generated by the Cauchy algorithm from $y^0 = g^0/\|g^0\|$. Then (y^{2k}) converges to a non-scalar point $r \in \mathcal{L}(e_1, e_n)$, the linear space generated by the basis vectors e_1 and e_n.*

Proof. We already know that any limit point of (y^k) is non-scalar. Let r be a limit point of (y^{2k}). The proof will be done in three steps.

(i) There exist two indices $p < q$ such that $r, r^+ \in \mathcal{L}(e_p, e_q)$.

Let r be a limit point of (y^k), and define the vectors r', r'' as above. We have

$$r_i'' = (\mu - d_i)(\mu^+ - d_i)r_i = L^2 r_i, \quad i = 1, \ldots, n,$$

where μ, μ^+ are fixed. For each i, either $r_i = 0$ or $(\mu - d_i)(\mu^+ - d_i) = L^2$. This is a second degree equation, which will have two real solutions, say, $d_i = d_p$ and $d_i = d_q$, because Theorem **??**(iii) prevents a single solution.

If $r \in \mathcal{L}(e_p, e_q)$, then trivially $r' = (\mu I - D)r \in \mathcal{L}(e_p, e_q)$.

(ii) r is the unique limit of (y^{2k}).

Assume by contradiction that r is not a unique limit point, and hence a non-isolated limit point of (y^{2k}). Then there exist an infinite number of limit points $\tilde{r}$ satisfying $\|\tilde{r} - r\| < \min\{|r_p|, |r_q|\}$. All such points must belong to $\mathcal{L}(e_p, e_q)$, for the following reason: any limit point $\tilde{r}$ must belong to some space $\mathcal{L}(e_s, e_t)$, where s, t are indices. If $\{s, t\} \neq \{p, q\}$, then either $\tilde{r}_p = 0$ or $\tilde{r}_q = 0$, and then $\|\tilde{r} - r\| \geq \min |r_p|, |r_q|$.

Let us examine a point $r \in \mathcal{L}(e_p, e_q)$. We have

$$\mu = r_p^2 d_p + r_q^2 d_q,$$

and immediately,

$$\mu - d_p = r_q^2(d_p - d_q), \quad \mu - d_q = r_p^2(d_q - d_p).$$

We also know that $\|r'\| = L$, where

$$r_p' = (\mu - d_p)r_p, \quad r_q' = (\mu - d_q)r_q.$$

So,

$$\begin{aligned} L = (r_p')^2 + (r_q')^2 &= (d_q - d_p)^2(r_q^4 r_p^2 + r_p^4 r_q^2) \\ &= (d_q - d_p)^2 r_p^2 r_q^2 (r_q^2 + r_p^2) \\ &= (d_q - d_p)^2 r_p^2 r_q^2. \end{aligned}$$

Hence, we have the equations

$$\begin{aligned} r_p^2 r_q^2 &= L/(d_q - d_p)^2 \\ r_p^2 + r_q^2 &= 1, \end{aligned}$$

a system with a finite number of isolated solutions, establishing a contradiction and proving (ii).

(iii) $r \in \mathcal{L}(e_1, e_n)$.

Let us prove that $q = n$. The proof that $p = 1$ is similar. Assume by contradiction that $q < n$. We know that both r and r^+ belong to $\mathcal{L}(e_p, e_q)$, and by Lemma **??**, $\mu(r) < d_q < d_n$ and $\mu(r^+) < d_q$. Since r, r^+ are the unique limit points of the sequence (y^k), for k sufficiently large, say, $k \geq \bar{k}$, $\mu(y^k) < d_q$. For such k,

$$\frac{\left|g_q^{k+1}\right|}{\left|g_q^k\right|} = |\mu(y^k) - d_q| < |\mu(y^k) - d_q| = \frac{\left|g_n^{k+1}\right|}{\left|g_n^k\right|}.$$

Hence for all $k > \bar{k}$

$$\frac{\left|g_q^k\right|}{\left|g_q^{\bar{k}}\right|} < \frac{\left|g_n^k\right|}{\left|g_n^{\bar{k}}\right|},$$

contradicting the fact that $y_n^k \to 0$ and $y_q \to r_q \neq 0$. □

2.3 Properties of the limiting points

Now we know that the study of asymptotic properties of the sequences generated by the Cauchy algorithm may be reduced to the study of the limit points. This leads to simple results that reproduce properties described by Nocedal, Sartenaer and Zhu [**?**] and by De Asmundis et al [**?**, **?**].

Consider an application of the Cauchy algorithm as above, starting from x^0 with no null component, and define $r = \lim_{k\to\infty} y^{2k}$, with $y^k = g^k / \left\|g^k\right\|$. We know that the only non-zero components of r are r_1 and r_n. From the analysis above, we know that

$$\begin{aligned}
\|r\| &= 1 && (6)\\
\mu &= d_1 r_1^2 + d_n r_n^2, \quad \lambda = 1/\mu && (7)\\
r' &= (\mu I - D)r, \quad \left\|r'\right\| = L && (8)\\
\mu' &= (r')^T D r / \left\|r'\right\|^2, \quad \lambda' = 1/\mu' && (9)\\
r'' &= (\mu' I - D)r' = L^2 r, \quad \left\|r''\right\| = L^2. && (10)
\end{aligned}$$

Lemma 3. $\mu + \mu' = d_1 + d_n$.

Proof. As $r'' = (\mu I - D)(\mu' I - D)r = L^2 r$,

$$L^2 = (\mu - d_1)(\mu' - d_1) = (\mu - d_n)(\mu' - d_n).$$

Then

$$\begin{aligned}
-d_1(\mu + \mu') + d_1^2 &= -d_n(\mu + \mu') + d_n^2\\
(d_n - d_1)(\mu + \mu') &= d_n^2 - d_1^2.
\end{aligned}$$

The result follows by dividing by $(d_n - d_1)$, completing the proof. □

This completes the proofs of the items (i)-(iii) of Theorem **??**. Let us prove the remaining items, studying a double iteration for large k:

$$\lim_{k\to\infty} \frac{g_i^{k+2}}{g_i^k} = (1 - \frac{d_i}{\mu})(1 - \frac{d_i}{\mu'}) \leq (1 - \frac{d_n}{\mu})(1 - \frac{d_n}{\mu'}), \tag{11}$$

because $d_i \leq d_n$. As $g_i^k \to 0$, we must have

$$\lim_{k\to\infty} \left|\frac{g_i^{k+2}}{g_i^k}\right| \leq 1,$$

proving (iv). In view of the inequality above, this is reduced to

$$(1 - \frac{d_i}{\mu})(1 - \frac{d_i}{\mu'}) > -1, \quad i = 1, \ldots, n. \tag{12}$$

Developing (**??**), we obtain immediately

$$\mu\mu' \geq \frac{(\mu + \mu')d_i - d_i^2}{2} = \frac{(d_1 + d_n)d_i - d_i^2}{2}, \quad \mu + \mu' = d_1 + d_n. \tag{13}$$

Let us define $\tilde{d} = (d_1 + d_n)/2$, $\delta_i = \tilde{d} - d_i$, $i = 1, \ldots, n$ and $\delta = \text{argmin}\{|\delta_i| \,|\, i = 1, \ldots, n\}$.

It is straightforward to check that the numerator of (**??**) satisfies

$$(d_1 + d_n)d_i - d_i^2 = \tilde{d}^2 - \delta_i^2,$$

and hence we know that μ, μ' must satisfy

$$\mu\mu' \geq \frac{\tilde{d}^2 - \delta_i^2}{2}, \quad \mu + \mu' = 2\tilde{d}, \quad i = 1, \ldots, n,$$

which is equivalent to

$$\mu\mu' \geq \frac{\tilde{d}^2 - \delta^2}{2}, \quad \mu + \mu' = 2\tilde{d}. \tag{14}$$

The components $|g_1|, |g_n|$ are reduced by $\alpha = (1 - d_n/\mu)(1 - d_n/\mu')$. Developing this expression and substituting $\mu + \mu' = d_1 + d_n$ we obtain

$$\alpha = 1 - \frac{d_1 d_n}{\mu\mu'}. \tag{15}$$

From (**??**) and this expression we conclude that,

$$\alpha \geq 1 - 2\frac{d_1 d_n}{\tilde{d}^2 - \delta^2}, \tag{16}$$

proving item (v) of Theorem **??**.

Bounds for the inverse steps μ, μ' are obtained by solving the system (**??**), which reduces to a simple second degree equation whose solution is

$$\bar{\mu}, \bar{\mu}' = \tilde{d} \pm \sqrt{(\tilde{d}^2 + \delta^2)/2}, \tag{17}$$

and thus $\bar{\mu} \leq \mu, \mu' \leq \bar{\mu}'$, proving the item (vi).

The last item of the main theorem is proved in the following lemma:

Lemma 4. *There exists $c > 0$ such that*

$$c = \left|\frac{r_n}{r_1}\right| = \left|\frac{r'_1}{r'_n}\right| = \sqrt{\frac{\mu - d_1}{\mu' - d_1}}, \quad \textit{with } \frac{r_n}{r_1} = -\frac{r'_1}{r'_n}.$$

Proof. By construction of the Cauchy step, $r \perp r'$, i.e., $r_1 r'_1 + r_n r'_n = 0$. The first equalities follow by dividing this by $r_1 r_n$.

Using this result, consider $r^+ = (I - D/\mu)$. We have

$$r_n^+ = (1 - d_n/\mu)r_n, r_1^+ = (1 - d_1\mu)r_1,$$

and so

$$\frac{r_n^+}{r_1^+} = \frac{(1 - d_n/\mu)}{(1 - d_1/\mu)}\frac{r_n}{r_1} = \frac{\mu - d_n}{\mu - d_1}\frac{r_n}{r_1} = -\frac{r_1}{r_n}.$$

From this last equality, it follows that

$$c^2 = \frac{r_n^2}{r_1^2} = -\frac{\mu - d_1}{\mu - d_n} = \frac{\mu - d_1}{\mu' - d_1},$$

using the fact that $\mu + \mu' = d_1 + d_n$, completing the proof. □

Some interesting relations proved in [**?**] are now straightforward: from $|r_n| = c|r_1|$ and $r_1^2 + r_n^2 = 1$ and then $r_1^2 + c^2 r_1^2 = 1$ we obtain:

$$r_1^2 = \frac{1}{1+c^2}, \quad r_n^2 = \frac{c^2}{1+c^2}, \tag{18}$$

$$\mu = \frac{d_1 + c^2 d_n}{1+c^2} = d_1 \frac{1 + c^2 C}{1+c^2}, \tag{19}$$

$$\mu' = \frac{c^2 d_1 + d_n}{1+c^2} = d_1 \frac{c^2 + C}{1+c^2}. \tag{20}$$

Let us now relate the constants L and c. We have

$$\begin{aligned} L^2 &= (\mu - d_1)(\mu' - d_1), \\ \mu - d_1 &= d_1 \frac{1 + c^2 C}{1+c^2} - d_1 = d_1 \frac{c^2 (C-1)}{1+c^2} \\ \mu' - d_1 &= d_1 \frac{c^2 + C}{1+c^2} - d_1 = d_1 \frac{C-1}{1+c^2}. \end{aligned}$$

Simplifying these expressions and using Lemma **??** we obtain

$$L^2 = \frac{c^2 (d_n - d_1)^2}{(1+c^2)^2} = \mu\mu' - d_1 d_n. \tag{21}$$

3 Breaking the cycle

The Cauchy algorithm reduces the medium variables and acts very slowly on the light and heavy variables. The cycle may be broken by enforcing either a very short (μ near d_n) or a very large (μ near d_1) step. It is difficult to estimate the value d_1, and a very large step will cause a great increase in the heavy variables (in absolute values), and will possibly increase the function value. A very large step should only be allowed if it is obtained by a Cauchy iterate, which always reduces the function value.

A very short step is harmless, and reduces the heavy variables. If the medium variables are already small, the function will then be dominated by the light variables, and the next Cauchy step will be large, breaking the cycle.

This is done by Asmundis et al. [**?**]: if μ_k, μ_{k+1} are near μ, μ', then $\mu_k + \mu_{k+1} \approx d_1 + d_n$, and then the step $1/(\mu_k + \mu_{k+1})$ will be short. They check the evolution of μ_k and decide to periodically estimate and apply short steps by this method. Their algorithm was named "steepest descent with alignment" (SDA).

We shall follow their approach, but with a different way of estimating short steps: all steps will be Cauchy steps taken at some point, making sure that in all iterations satisfy $\mu_k \in (d_1, d_n)$.

In the algorithms below we call $\lambda-$gradient step a steepest descent step with step length λ. The algorithm below is the Cauchy algorithm modified by periodically applying short steps.

Stopping rule. Each step must be followed by the statement $k = k + 1$ and by testing a stopping rule. The usual test is $\left\|g^k\right\| \leq \epsilon$, for a given $\epsilon > 0$. In our examples with diagonal Hessian matrices we test the function values, because the optimal value is zero.

Algorithm 1. *Cauchy-short Algorithm*

Data: $x_0 \in \mathbb{R}^n$, $K_I, K_C, K_s \in \mathbb{N}$ (respectively $10, 6, 2$ in our choice), $k = 0$.
Take K_I Cauchy steps.
REPEAT
Compute an estimated short step λ_s.
Take K_s λ_s-gradient steps.
Take K_C Cauchy steps.

Remark. The choice of K_I, K_C, K_s is somewhat arbitrary, but we noticed in many tests that the oscillatory pattern is achieved in about 10 Cauchy iterations initially, and in about 6 Cauchy iterations after each set of short steps. Two consecutive short steps are sufficient to cause a substantial reduction in the large variables. See Fig. **??** for an example. Note that even if the estimated short step is not very near $1/d_n$, the short steps are harmless.

Estimating a short step: a "mock" large step. We intend to state an algorithm in which all iterations take Cauchy steps, and so all steps will satisfy $d_1 < \mu_k < d_n$. As the cycle pattern is established, an artificially very large step will cause a great increase in the heavy variables, and consequently the next Cauchy step will be short. The very large step will be computed but not applied. Let S be a very large step length ($S >> 1/d_1$). At an iteration k we compute the following short step:

$$\tilde{g} = (I - S\,D)g^k \tag{22}$$

$$\lambda_s = \frac{\tilde{g}^T\tilde{g}}{\tilde{g}^T D\tilde{g}}. \tag{23}$$

The reason why this will be a short step is the following: for a large value of S,

$$\tilde{g}_i = (1 - S\,d_i)g_i^k \approx -S\,d_i g_i, \quad i = 1, \ldots, n.$$

Hence the following Cauchy step will satisfy

$$\lambda_s = \frac{\tilde{g}^T\tilde{g}}{\tilde{g}^T D\tilde{g}} = \frac{\sum_{i=1}^{n-1} g_1^2 d_i^2 + g_n^2 d_n^2}{\sum_{i=1}^{n-1} g_1^2 d_i^3 + g_n^2 d_n^3}$$

If the medium components are small, then for $i = 1, \ldots, n-1$, $|g_i d_i| << |g_n d_n|$, and then $\lambda_s \approx 1/d_n$. A safeguard may be used to enforce short steps, by setting $\lambda_s = \min\{\lambda_s, \lambda_{k-1}, \lambda_{k-2}\}$, as $\min\{\lambda_{k-1}, \lambda_{k-2}\}$ should be a short step.

Fig.**??** shows iterates before and after the short steps for our example. Fig. **??** shows the behavior of the step lengths for the Cauchy-short algorithm: the cyclic pattern is broken by periodically forcing a short step, which is naturally followed by a large Cauchy step.

Alternating Cauchy and short steps. It is known that about one half of the steps will be short. The algorithm below uses the computation of short steps as above, but applies them after all Cauchy steps:

Algorithm 2. *Alternated Cauchy-short steps.*

Data: $x_0 \in \mathbb{R}^n$, K_I, K_C, K_s (respectively $10, 6, 2$ in our choice), $k = 0$.
Take K_I Cauchy steps.
REPEAT
Compute an estimated short step λ_s.
Take K_s λ_s-gradient steps.
Take K_C iterations composed of a Cauchy step followed by one λ_s-gradient step.

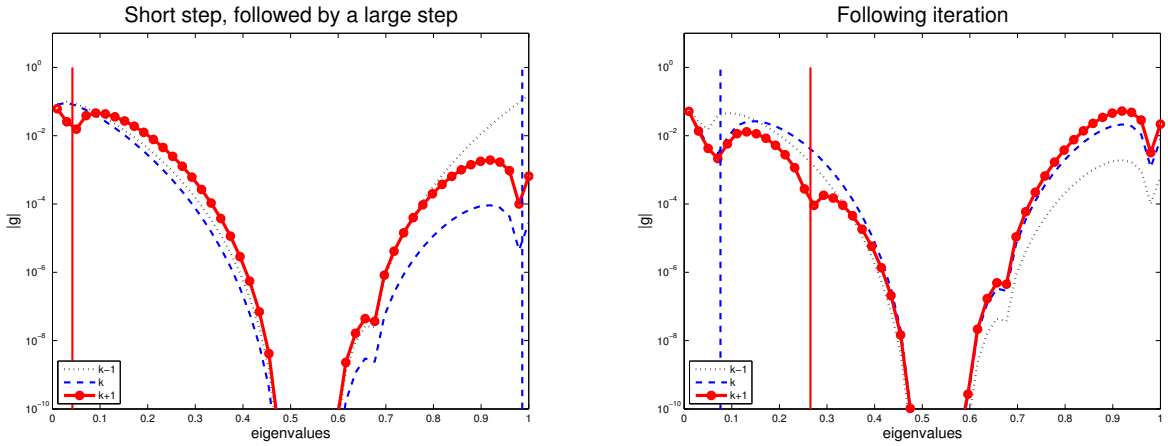

Figure 3: Before and after 2 short steps in CS algorithm.

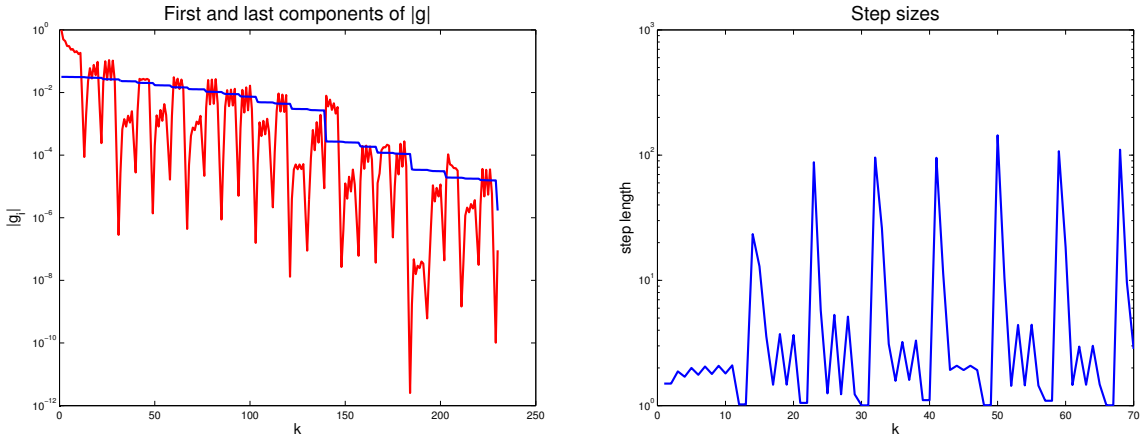

Figure 4: Behavior of λ_k and of $\left|g_1^k\right|, \left|g_n^k\right|$ in an application of the CS algorithm.

The algorithm has a stronger effect on the very heavy variables. Consequently the Cauchy and eventually the short steps increase. Fig.**??** shows the behavior of the function values on an example with $C = 1000$ and $n = 1000$, eigenvalues uniformly distributed between 0.001 and 1, $x_i^0 = 1/\sqrt{d_i}$. The figure shows the evolution of the Barzilai-Borwein (BB), the Cauchy-short (CS), the steepest descent with alignment (SDA), and the Alternated Cauchy-short (ACS) algorithms. It also shows the effect of using Chebyshev roots to correct the step lengths, as described below.

Using Chebyshev roots. The paper [**?**] has the following result: if bounds $l \le d_1$ and $u \ge d_n$ are known, then the stopping condition $|x_i^k| \le \epsilon |x_i^0|$ for $i = 1, \ldots, n$ is achieved by the following steepest descent scheme:

For $k = 0$ to $K - 1$, $g^{k+1} = (I - \Lambda_k D) g^k$,
where, using $C = u/l$,

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

and the set of step lengths is

$$\Lambda = \{1/\mu_k \,|\, k = 0, \ldots, K-1\}, \quad \mu_k = \frac{u-l}{2} \cos\left(\frac{1+2k}{2K}\pi\right) + \frac{u+l}{2}, \tag{24}$$

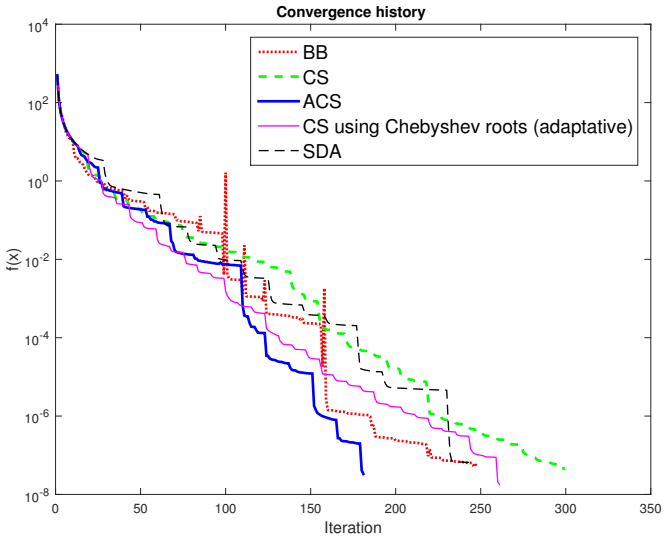

Figure 5: Function values for the BB, CS, ACS, SDA and CS with steps corrected to Chebyshev roots using adaptive bounds for the eigenvalues.

taken in any order. If the bounds l and u are exact, this number of iterations coincides almost exactly with the worst case performance of the Krylov space method, the best possible.

This means that if good bounds for the eigenvalues are known, then the number of iterations of any gradient algorithm may be limited to $K(C)$ if the step lengths are corrected to match elements of Λ. This is the resulting scheme:
Let $C = u/l$, compute $K(C)$ and the set Λ by (**??**).
Take any steepest descent method, and execute the following command after the computation of each step length λ_k;
Set $\lambda_k = \text{argmin}\{|\nu - \lambda_k| \, \nu \in \Lambda\}$.
Remove the element λ_k from the set Λ.

Adaptive scheme. This scheme depends on reliable lower and upper bounds l and u for the eigenvalues. When they are not available, we construct them by adding the following procedure to Algorithms **??** and **??**:
Initialization: choose an integer $k_0 > K_i + K_s$, take the first k_0 iterations of the algorithm and set $u = 1.2 \max_{k=0,\ldots,k_0-1} 1/\lambda_k$, $l = 0.25 \min_{k=0,\ldots,k_0-1} 1/\lambda_k$, $C = u/l$, and compute Λ by (**??**).
Adaptation: at all steps with $k > k_0$, perform the following adaptation of the set Λ:
IF $u < 1/\lambda_k$, set $u = 1.2\,u$, C=u/l and reset Λ by (**??**).
IF $l > 1/\lambda_k$, set $l = l/4$, C=u/l and reset Λ by (**??**).

The initial value of u will hopefully satisfy $u > d_n$, because the short step computation is efficient, both using our algorithms and ASD (with k_0 properly chosen). The lower bound will probably be updated, and each time l changes the size of Λ doubles.

We applied this scheme to our test problems, using the Cauchy-short algorithm. The result is shown in Fig. **??**. This scheme is only applicable to quadratic functions, but it reduces all variables to $|g_i^K| \leq \epsilon |g_i^0|$, which may be important in the solution of linear systems of equations and least squares problems.

Performance profile. Finally, Fig. **??** is a performance profile, computed as described in [**?**], using 120 quadratic problems with 1000 variables, three values of the condition number $C = 1000, 10000, 100000$, and each problem with a different randomly generated initial point. The eigenvalues are randomly generated with four different distributions, exemplified by Fig. **??**.

It shows that all new schemes are efficient, and seem to be superior to the Barzilai-Borwein method, with the advantage of generating monotonically decreasing function values.

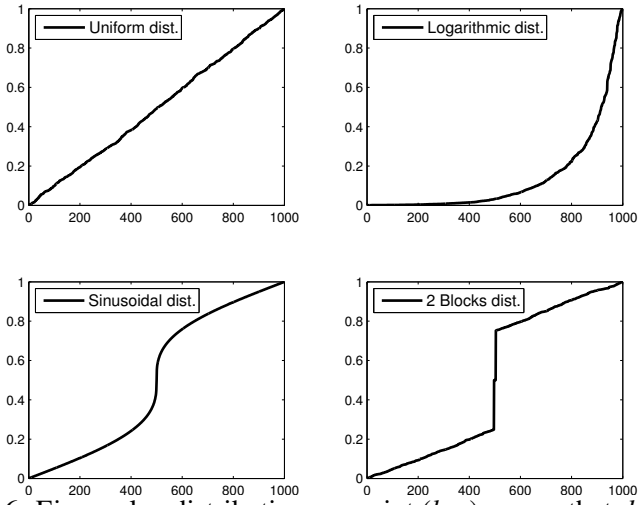

Figure 6: Eigenvalue distributions: a point (k, s) means that $d_k = sd_n$.

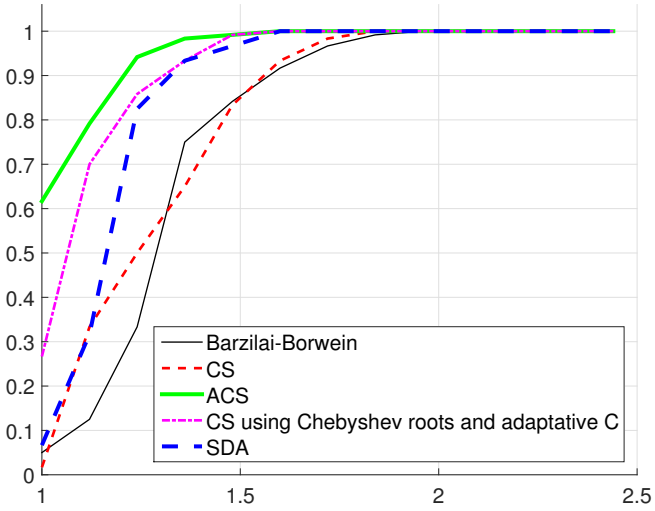

Figure 7: Performance profile for the algorithms Barzilai-Borwein, Algorithm (**??**) (CS), Algorithm (**??**) (ACS), steepest descent with alignment (SDA) and CS with adaptive computation of Chebyshev steps.

Conclusion. We believe that a good understanding of algorithms for quadratic problems is fundamental for the design of methods for more general problems. The steepest descent method, mainly using approximated Cauchy steps, is the most well known of all methods. Its poor performance, which contradicts the simple intuition of greedily computing the maximum function reduction at each iteration, has always been frustrating. It was explained by Akaike, and only recently a simple cure has been found: just take a short step once in a while. In this paper we hope to have summarized the classical proofs of this behavior, and proposed new procedures for quadratic problems. We did not, for the time being, try to apply similar schemes to non-quadratic functions, but we hope that these ideas may lead to general methods without the need of strategies for dealing with non-monotonicity.

References

[1] H. Akaike. On a successive transformation of probability distribution and its application to the analysis of the optimum gradient method. *Ann. Inst. Statist. Math. Tokyo*, 11:1–17, 1959.

[2] J. Barzilai and J. M. Borwein. Two-point step size gradient methods. *IMA J. Numer. Anal.*, 8:141–148, 1988.

[3] E. G. Birgin, J. M. Martínez, and M. Raydan. Spectral Projected Gradient Methods. In C. A. Floudas and P. M. Pardalos, editors, *Encyclopedia of Optimization*, pages 3652–3659. Springer, 2009.

[4] A. Cauchy. Méthode générale pour la résolution des systèmes d'équations simultanées. *Comp. Rend. Acad. Sci. Paris*, 25:536–538, 1847.

[5] Y. H. Dai. Alternate step gradient method. *Optimization*, 52(4-5):395–415, 2003.

[6] R. de Asmundis, D. di Serafino, W. Hager, G. Toraldo, and H. Zhang. An efficient gradient method using the Yuan steplength. Technical report, Sapienza University of Rome, Italy, 2014.

[7] R. de Asmundis, D. di Serafino, R. Riccio, and G. Toraldo. On spectral properties of steepest descent mehtods. *IMA J. Numer. Anal.*, 33:1416–1435, 2013.

[8] E. D. Dolan and J. J. Moré. Benchmarking optimization software with performance profiles. *Mathematical Programming*, 91(2):201–213, 2002.

[9] G. E. Forsythe. On the asymptotic directions of the s-dimensional optimum gradient method. *Numerische Mathematik*, 11:57–76, 1968.

[10] C. C. Gonzaga. Optimal performance of the steepest descent algorithm for quadratic functions. Technical report, Federal University of Santa Catarina, Florianopolis, Brazil, 2014.

[11] J. Nocedal, A. Sartenaer, and C. Zhu. On the behavior of the gradient norm in the steepest descent method. *Computational Optimization and Applications*, 22:5–35, 2002.

[12] M. Raydan. The Barzilai and Borwein gradient method for large scale unconstrained minimization problem. *SIAM Journal on Optimization*, 7:26–33, 1997.

[13] M. Raydan and B. Svaiter. Relaxed steepest descent and Cauchy-Barzilai-Borwein method. *Computatinal Optimization and Applications*, 21:155–167, 2002.