Operations Research Letters 40 (2012) 541–545

Contents lists available at SciVerse ScienceDirect

Operations Research Letters

journal homepage: www.elsevier.com/locate/orl

An improved ellipsoid method for solving convex differentiable optimization problems

Amir Beck [a,*], Shoham Sabach [b]

[a] Department of Industrial Engineering and Management, Technion—Israel Institute of Technology, Haifa 32000, Israel

[b] Department of mathematics, Technion—Israel Institute of Technology, Haifa 32000, Israel

ARTICLE INFO

Article history:
Received 15 July 2012
Received in revised form
17 September 2012
Accepted 17 September 2012
Available online 2 October 2012

Keywords:
Ellipsoid method
Convex optimization
Lipschitz gradient

ABSTRACT

We consider the problem of solving convex differentiable problems with simple constraints. We devise an improved ellipsoid method that relies on improved deep cuts exploiting the differentiability property of the objective function as well as the ability to compute an orthogonal projection onto the feasible set. The linear rate of convergence of the objective function values sequence is proven and several numerical results illustrate the potential advantage of this approach over the classical ellipsoid method.

1. Introduction

1.1. Short review of the ellipsoid method

Consider the convex problem

$$(\mathrm{P}):\quad \begin{array}{ll} \min & f(\mathbf{x}) \\ \text{s.t.} & \mathbf{x} \in X, \end{array}$$

where f is a convex function over the closed and convex set X. We assume that the objective function f is subdifferentiable over X and that the problem is solvable with X^* being its optimal set and f^* being its optimal value. One way to tackle the general problem (P) is via the celebrated ellipsoid method, which is one of the most fundamental methods in convex optimization. It was first developed by Yudin and Nemirovski (1976), and Shor (1977) for general convex optimization problems and was then came to awareness with the seminal work of Khachiyan [6] showing – using the ellipsoid method – that linear programming can be solved in a polynomial time. The ellipsoid method does not require the objective function to be differentiable and it assumes that two oracles are available: a separation oracle and a subgradient oracle.

To describe the ellipsoid method in more details, we use the following notation. The ellipsoid with center $\mathbf{c} \in \mathbb{R}^n$ and associated matrix $\mathbf{P} \in \mathbb{R}^{n\times n}$ $(\mathbf{P} \succ \mathbf{0})$ is given by

$$E(\mathbf{c}, \mathbf{P}) \equiv \left\{\mathbf{x} \in \mathbb{R}^n : (\mathbf{x}-\mathbf{c})^T \mathbf{P}^{-1} (\mathbf{x}-\mathbf{c}) \leq 1\right\}.$$

For a given $\mathbf{g}, \mathbf{c} \in \mathbb{R}^n$ such that $\mathbf{g} \neq \mathbf{0}$ and $h \in \mathbb{R}$ we define the hyperplane

$$H(\mathbf{g}, \mathbf{c}, h) \equiv \left\{\mathbf{x} \in \mathbb{R}^n : \mathbf{g}^T (\mathbf{x}-\mathbf{c}) + h = 0\right\}$$

and its associated half-space

$$H^-(\mathbf{g}, \mathbf{c}, h) \equiv \left\{\mathbf{x} \in \mathbb{R}^n : \mathbf{g}^T (\mathbf{x}-\mathbf{c}) + h \leq 0\right\}.$$

A schematic description of the ellipsoid method is as follows: we begin with an ellipsoid $E(\mathbf{c}_0, \mathbf{P}_0)$ that contains the optimal set X^*. At iteration k, an ellipsoid $E(\mathbf{c}_k, \mathbf{P}_k)$ is given for which $X^* \subseteq E(\mathbf{c}_k, \mathbf{P}_k)$. We then find an hyperplane of the form $H(\mathbf{g}_k, \mathbf{c}_k, h_k)$ where $\mathbf{g}_k \in \mathbb{R}^n$ $(\mathbf{g}_k \neq 0)$ and $h_k \geq 0$ such that

$$X^* \subseteq H^-(\mathbf{g}_k, \mathbf{c}_k, h_k) \cap E(\mathbf{c}_k, \mathbf{P}_k).$$

The ellipsoid at the next iteration $E(\mathbf{c}_{k+1}, \mathbf{P}_{k+1})$ is defined to be the minimum volume ellipsoid containing $H^-(\mathbf{g}_k, \mathbf{c}_k, h_k) \cap E(\mathbf{c}_k, \mathbf{P}_k)$. The schematic algorithm, which includes the specific update formulas for the minimum volume ellipsoid is now described in detail.

The ellipsoid method

- *Initialization.* Set $\mathbf{c}_0 = \mathbf{0}$ and $\mathbf{P}_0 = R^2\mathbf{I}$.
- *General Step* $(k = 0, 1, 2, \ldots)$
  - A. Find $\mathbf{g}_k \in \mathbb{R}^n$ $(\mathbf{g}_k \neq \mathbf{0})$ and $h_k \geq 0$ for which $X^* \subseteq H^-(\mathbf{g}_k, \mathbf{c}_k, h_k) \cap E(\mathbf{c}_k, \mathbf{P}_k)$.
  - B. $\mathbf{c}_{k+1}$ and $\mathbf{P}_{k+1}$ are the center and associated matrix of the minimum volume ellipsoid containing $H^-(\mathbf{g}_k, \mathbf{c}_k, h_k) \cap$

* Corresponding author.
E-mail addresses: becka@ie.technion.ac.il (A. Beck), ssbach@gmail.com (S. Sabach).

0167-6377/$ – see front matter
doi:10.1016/j.orl.2012.09.005

$E(\mathbf{c}_k, \mathbf{P}_k)$ and are explicitly given by the following expressions:

$$\mathbf{c}_{k+1} = \mathbf{c}_k - \frac{1+n\alpha_k}{1+n}\mathbf{P}_k\widetilde{\mathbf{g}}_k$$

$$\mathbf{P}_{k+1} = \frac{n^2\left(1-\alpha_k^2\right)}{n^2-1}\left(\mathbf{P}_k - \frac{2\left(1+n\alpha_k\right)}{(1+n)\left(1+\alpha_k\right)}\mathbf{P}_k\widetilde{\mathbf{g}}_k\widetilde{\mathbf{g}}_k^T\mathbf{P}_k\right)$$

where

$$\widetilde{\mathbf{g}}_k = \frac{\mathbf{g}_k}{\sqrt{\mathbf{g}_k^T\mathbf{P}_k\mathbf{g}_k}} \quad \text{and} \quad \alpha_k = \frac{h_k}{\sqrt{\mathbf{g}_k^T\mathbf{P}_k\mathbf{g}_k}}.$$

Of course, there are two missing elements in the above description. First, the choice of R is not given. In the classical ellipsoid method R is any positive number satisfying $X^* \subseteq E\left(\mathbf{0}, R^2\mathbf{I}\right)$. Second, there is no specification of the way the cutting hyperplane $H(\mathbf{g}_k, \mathbf{c}_k, h_k)$ is constructed. In the classical ellipsoid method the cutting parameter is a "neutral" cut, meaning that $h_k = 0$ whereas $\mathbf{g}_k$ is constructed as follows: we assume that a separation and a subgradient oracles are given. We call the separation oracle with input $\mathbf{c}_k$. The oracle determines whether $\mathbf{c}_k$ belongs or not to X. If $\mathbf{c}_k \notin X$, then the separation oracle generates a separating hyperplane between $\mathbf{c}_k$ and X and generates $\mathbf{g}_k \neq \mathbf{0}$ for which $X^* \subseteq H^-(\mathbf{g}_k, \mathbf{c}_k, 0)$. Otherwise, if $\mathbf{c}_k \in X$, then the subgradient oracle provides a vector $\mathbf{g}_k$ in the subdifferential set $\partial f(\mathbf{c}_k)$ for which it is easy to show that $X^* \subseteq H^-(\mathbf{g}_k, \mathbf{c}_k, 0)$.

Remark 1.1. The classical ellipsoid method generates neutral cuts, but there are variations in which deep cuts ($h_k > 0$) are used [4,5]. For instance, in [5] the authors exploit the fact that the objective function values of the feasible centers are not nonincreasing. It is not difficult to see that when $\mathbf{c}_k \in X$, one can use a cutting hyperplane of the form $H(\mathbf{g}_k, \mathbf{c}_k, h_k)$, where $\mathbf{g}_k \in \partial f(\mathbf{c}_k)$ as before, but with

$$h_k = f(\mathbf{c}_k) - \min\left\{f\left(\mathbf{c}_j\right) : 0 \le j \le k, \mathbf{c}_k \in X\right\}. \tag{1.1}$$

When $f(\mathbf{c}_k)$ is not of the smallest value among the feasible centers at iterations $0, 1, 2, \ldots, k$, the resulting cut is deep ($h_k > 0$).

The convergence of the ellipsoid method relies on the fact that the volumes of the generated ellipsoids $E_k \equiv E(\mathbf{c}_k, \mathbf{P}_k)$ are decreasing at a linear rate to zero (more on that in the sequel). In particular, it is known that (see *e.g.*, [3])

$$\frac{\mathrm{Vol}(E_{k+1})}{\mathrm{Vol}(E_k)} = \delta_k^{\frac{n}{2}}\sqrt{1-\sigma_k}, \tag{1.2}$$

where

$$\delta_k = \frac{n^2\left(1-\alpha_k^2\right)}{n^2-1}, \tag{1.3}$$

$$\sigma_k = \frac{2\left(1+n\alpha_k\right)}{(n+1)\left(1+\alpha_k\right)}. \tag{1.4}$$

For the neutral cut setting ($\alpha_k = h_k = 0$) the decrease is at least by a factor of $e^{-1/(2n+2)}$.

1.2. The new assumptions

Suppose that in addition to the convexity of the objective function and the existence of the two oracles, we have the following assumption.

Assumption 1. The objective function $f : X \to \mathbb{R}$ is convex, differentiable and has a Lipschitz gradient over X:

$$\|\nabla f(\mathbf{x}) - \nabla f(\mathbf{y})\| \le L(f)\|\mathbf{x}-\mathbf{y}\| \quad \text{for every } \mathbf{x}, \mathbf{y} \in X. \tag{1.5}$$

In addition, we assume that the orthogonal projection operator defined by

$$P_X(\mathbf{x}) := \operatorname*{argmin}_{\mathbf{y}\in X} \|\mathbf{x}-\mathbf{y}\|^2 \tag{1.6}$$

is "easy to compute". Of course, this scenario is more restrictive than the general setting of the ellipsoid method, however there are situations in which such assumptions are met. The basic question is the following:

Main question: is there a way to utilize the additional differentiability assumption of the objective function and computability of the orthogonal projection in order to improve the ellipsoid method?

The answer of this question is affirmative and will be described in the following sections.

2. A new deep cut ellipsoid method

Before describing the deep cuts that will be used in the paper, some important preliminaries on the gradient mapping are required.

2.1. The gradient mapping

We define the following two mappings which are essential in our analysis of the proposed algorithm.

Definition 1 (*Gradient Mapping*)**.** Let X be a nonempty, closed and convex subset of $\mathbb{R}^n$ and let $f : X \to \mathbb{R}$ be a differentiable function. For every $M > 0$, we define

(i) the *proj-grad mapping* by

$$T_M(\mathbf{x}) \equiv P_X\left(\mathbf{x} - \frac{1}{M}\nabla f(\mathbf{x})\right) \quad \text{for all } \mathbf{x} \in \mathbb{R}^n; \tag{2.1}$$

(ii) the *gradient mapping* (see also [7]) by

$$\begin{aligned} G_M(\mathbf{x}) &\equiv M(\mathbf{x} - T_M(\mathbf{x})) \\ &= M\left[\mathbf{x} - P_X\left(\mathbf{x} - \frac{1}{M}\nabla f(\mathbf{x})\right)\right]. \end{aligned} \tag{2.2}$$

Remark 2.1 (*Unconstrained Case*)**.** In the unconstrained setting, that is, when $X = \mathbb{R}^n$, the orthogonal projection is the identity operator and hence

(i) the proj-grad mapping T_M is equal to $I - \frac{1}{M}\nabla f$;
(ii) the gradient mapping G_M is equal to ∇f.

It is well known (see *e.g.*, [7]) that $G_M(\mathbf{x}) = 0$ if and only if $\mathbf{x} \in X^*$.

We now present a useful inequality for convex functions with Lipschitz gradient. The result was established in [1] for the more general prox-grad operator and it is recalled here for the specific case of the proj-grad mapping (see [1, Lemma 1.6]).

Lemma 2.1. *Let X be a nonempty, closed and convex subset of $\mathbb{R}^n$ and let $f : X \to \mathbb{R}$ be a convex differentiable function whose gradient is Lipschitz. Let $\mathbf{z} \in X$ and $\mathbf{w} \in \mathbb{R}^n$. Then if the inequality*

$$\begin{aligned} f(T_M(\mathbf{x})) \le f(\mathbf{w}) &+ \langle \nabla f(\mathbf{w}), T_M(\mathbf{w}) - \mathbf{w}\rangle \\ &+ \frac{M}{2}\|T_M(\mathbf{w}) - \mathbf{w}\|^2 \end{aligned} \tag{2.3}$$

holds for a positive number M, then

$$f(T_M(\mathbf{w})) - f(\mathbf{z}) \le \langle G_M(\mathbf{w}), \mathbf{w}-\mathbf{z}\rangle - \frac{1}{2M}\|G_M(\mathbf{w})\|^2. \tag{2.4}$$

Remark 2.2. By the descent lemma (see [2, Proposition A.24]), property (2.3) is satisfied when $M \ge L(f)$, which implies that in those cases the inequality (2.4) holds true.

2.2. The deep cut

The deep cut that will be used relies on the following technical lemma.

Lemma 2.2 (*Deep Cut Property*)**.** *Let M be a positive number and assume that $\mathbf{x} \in \mathbb{R}^n$ is a vector satisfying the inequality*

$$f(T_M(\mathbf{x})) \le f(\mathbf{x}) + \langle \nabla f(\mathbf{x}), T_M(\mathbf{x}) - \mathbf{x} \rangle + \frac{M}{2} \|T_M(\mathbf{x}) - \mathbf{x}\|^2. \tag{2.5}$$

Then, for any $\mathbf{x}^* \in X^*$, the inequality

$$\left\langle G_M(\mathbf{x}), \mathbf{x} - \mathbf{x}^* \right\rangle \ge \frac{1}{2M} \|G_M(\mathbf{x})\|^2 \tag{2.6}$$

holds true, that is

$$X^* \subseteq Q_{M,\mathbf{x}} \equiv \left\{ \mathbf{z} \in \mathbb{R}^n : \langle G_M(\mathbf{x}), \mathbf{x} - \mathbf{z} \rangle \ge \frac{1}{2M} \|G_M(\mathbf{x})\|^2 \right\}.$$

Proof. Invoke Lemma 2.1 with $\mathbf{z} = \mathbf{x}^*$ and $\mathbf{w} = \mathbf{x}$ and obtain that

$$f(T_M(\mathbf{x})) - f\left(\mathbf{x}^*\right) \le \langle G_M(\mathbf{x}), \mathbf{x} - \mathbf{x}^* \rangle - \frac{1}{2M} \|G_M(\mathbf{x})\|^2.$$

The result (2.6) then follows by noting that the optimality of $\mathbf{x}^*$ yields $f(T_M(\mathbf{x})) \ge f(\mathbf{x}^*)$. □

Remark 2.3. As in Remark 2.2, by the descent lemma, the inequality (2.5) is satisfied for all $M \ge L(f)$ and in those cases the inclusion $X^* \subseteq Q_{M,\mathbf{x}}$ holds true.

2.3. The improved deep cut ellipsoid (IDCE) method

The deep cut ellipsoid method that we will consider is using the cutting hyperplane described in Lemma 2.2. In its most basic form, at iteration k, the corresponding half-space would be:

$$\left\{ \mathbf{x} \in \mathbb{R}^n : \left\langle G_{L(f)}(\mathbf{c}_k), \mathbf{x} - \mathbf{c}_k \right\rangle + \frac{1}{2L(f)} \left\| G_{L(f)}(\mathbf{c}_k) \right\|^2 \le 0 \right\}.$$

Note that as opposed to the classical ellipsoid method, there is only one option for a cutting plane, and there is no need to consider different cases (*e.g.*, according to whether the center is feasible or not). If the Lipschitz constant $L(f)$ was known, then we could have defined the deep cut method by employing the ellipsoid method with step A described by:

$$\mathbf{g}_k = G_{L(f)}(\mathbf{c}_k)$$
$$h_k = \frac{1}{2L(f)} \left\| G_{L(f)}(\mathbf{c}_k) \right\|^2.$$

The improved deep cut method (IDCE) that we consider below has two additional features: first, it does not assume any knowledge on the Lipschitz constant by incorporating a backtracking procedure for estimating the constant and in addition, similarly to the deep cut technique described in Remark 1.1, it utilizes the knowledge on the best function value obtained so far.

The IDCE method

Input: L_{-1}-an initial estimate on the Lipschitz constant. $\eta > 1$.

Employ the ellipsoid method with R chosen to satisfy $X^* \subseteq E(\mathbf{0}, (R/2)^2\mathbf{I})$ and with the following step A:

A.1. Find the smallest nonnegative integer such that with $\bar{L} = \eta^{i_k} L_{k-1}$ the inequality

$$f\left(T_{\bar{L}}(\mathbf{c}_k)\right) \le f(\mathbf{c}_k) + \left\langle \nabla f(\mathbf{c}_k), T_{\bar{L}}(\mathbf{c}_k) - \mathbf{c}_k \right\rangle + \frac{\bar{L}}{2} \left\| T_{\bar{L}}(\mathbf{c}_k) - \mathbf{c}_k \right\|^2 \tag{2.7}$$

is satisfied.
Set $L_k = \bar{L}$.

A.2. Compute $\mathbf{g}_k$ and h_k as follows:

$$\mathbf{g}_k = G_{L_k}(\mathbf{c}_k), \tag{2.8}$$

$$h_k = \frac{1}{2L_k} \left\| G_{L_k}(\mathbf{c}_k) \right\|^2 + f\left(T_{L_k}(\mathbf{c}_k)\right) - \ell_k, \tag{2.9}$$

where $\ell_k = \min\left\{ f\left(T_{L_j}(\mathbf{c}_j)\right) : 0 \le j \le k \right\}$.

Note that if L_{-1} is an upper bound on the Lipschitz constant $L(f)$, then Step A.1 is redundant and we have $L_k \equiv L_{-1}$. Also, as opposed to the classical ellipsoid method, we assume here that $X^* \subseteq E(\mathbf{0}, (R/2)^2\mathbf{I})$ and *not* that $X^* \subseteq E(\mathbf{0}, R^2\mathbf{I})$.

An interesting question that arises here is: *what is the sequence generated by the method?* In the classical ellipsoid method, the generated sequence is essentially the subsequence of centers which are feasible. This is in fact a problematic issue, since it implies the assumption that the feasible set has a nonempty interior, otherwise practically none of the centers will be feasible. In the setting of this paper, we will see that it is quite natural to define the sequence generated by the method as the sequence $\left\{ T_{L_k}(\mathbf{c}_k) \right\}_{k=0}^{\infty}$, which is obviously feasible and its evaluation is not an additional computational burden since these proj-grad expressions are required anyway for employing the IDCE method. In addition, the nonemptiness of the interior of the feasible set is no longer required.

3. Complexity analysis of the IDCE method

3.1. Decrease of volumes

Our ultimate goal is to estimate the convergence of the function values $\{f(T_{L_k}(\mathbf{c}_k))\}_{k\ge 0}$ to the optimal value. As in the classical ellipsoid method, the analysis relies on the volume decrease property (1.2), which by the definitions of δ_m and σ_m (Eqs. (1.3) and (1.4), respectively) yields

$$\frac{\mathrm{Vol}(E_{m+1})}{\mathrm{Vol}(E_m)} = \delta_m^{\frac{n}{2}} \sqrt{1 - \sigma_m} = \left(\frac{n^2}{n^2 - 1} \right)^{\frac{n}{2}} \times \left(1 - \alpha_m^2\right)^{\frac{n}{2}} \sqrt{1 - \frac{2(1 + n\alpha_m)}{(n+1)(1+\alpha_m)}}.$$

After some simple algebra we have that

$$\frac{\mathrm{Vol}(E_{m+1})}{\mathrm{Vol}(E_m)} = \left(1 + \frac{1}{n^2 - 1}\right)^{\frac{n-1}{2}} \left(1 - \frac{1}{n+1}\right) \times \left(1 - \alpha_m^2\right)^{\frac{n}{2}} \sqrt{\frac{1 - \alpha_m}{1 + \alpha_m}}.$$

Since $1 + x \le e^x$ for any $x \in \mathbb{R}$ we obtain

$$\frac{\mathrm{Vol}(E_{m+1})}{\mathrm{Vol}(E_m)} \le e^{\frac{n-1}{2(n^2-1)}} \cdot e^{-\frac{1}{n+1}} \left(1 - \alpha_m^2\right)^{\frac{n}{2}} \sqrt{\frac{1 - \alpha_m}{1 + \alpha_m}} = e^{\frac{-1}{2(n+1)}} \left(1 - \alpha_m^2\right)^{\frac{n}{2}} \sqrt{\frac{1 - \alpha_m}{1 + \alpha_m}},$$

and therefore

$$\mathrm{Vol}(E_m) \le e^{\frac{-1}{2(n+1)}} \left(1 - \alpha_{m-1}^2\right)^{\frac{n}{2}} \mathrm{Vol}(E_{m-1}) \le \cdots \le e^{\frac{-m}{2(n+1)}} \prod_{j=0}^{m-1} \left(1 - \alpha_j^2\right)^{\frac{n}{2}} \mathrm{Vol}(E_0). \tag{3.1}$$

By denoting

$$j_m \in \operatorname{argmin}\{\alpha_k : k = 0, 1, \ldots, m-1\} \quad (3.2)$$

we obtain that (3.1) implies the following soon to be useful result.

Lemma 3.1. *Let $\{E_m\}_{m=0}^{\infty}$ be the sequence of ellipsoids generated by the IDCE method. Then*

$$\mathrm{Vol}\,(E_m) \le e^{\frac{-m}{2(n+1)}} \left(1 - \alpha_{j_m}^2\right)^{\frac{nm}{2}} \mathrm{Vol}\,(E_0),$$

where the index j_m is defined in (3.2).

3.2. Complexity bound

The following result establishes the linear convergence rate of the function values of the sequence generated by the IDCE method.

Theorem 3.1. *Let $\{\mathbf{c}_k\}_{k=0}^{\infty}$ be the sequence of ellipsoid centers generated by the IDCE method and let $\varepsilon < L(f)\frac{R^2}{2}$. Then for any m satisfying*

$$m > n(n+1)\ln\left(\frac{2L(f)R^2}{\varepsilon}\right) \quad (3.3)$$

we have

$$\min_{k=0,\ldots,m} f\left(T_{L_k}(\mathbf{c}_k)\right) - f^* \le \varepsilon.$$

Proof. Assume that m satisfies (3.3). By (2.7), we have that

$$f\left(T_{L_k}(\mathbf{c}_k)\right) \le f(\mathbf{c}_k) + \left\langle \nabla f(\mathbf{c}_k), T_{L_k}(\mathbf{c}_k) - \mathbf{c}_k \right\rangle + \frac{L_k}{2}\left\|T_{L_k}(\mathbf{c}_k) - \mathbf{c}_k\right\|^2. \quad (3.4)$$

For any $\mathbf{x} \in \mathbb{R}^n$, using the fact that $T_{L_k}(\mathbf{c}_k) \in X$, we obtain from Lemma 2.1 that

$$f\left(T_{L_k}(\mathbf{c}_k)\right) \le f\left(T_{L(f)}(\mathbf{x})\right) + \left\langle \mathbf{g}_k, \mathbf{c}_k - T_{L(f)}(\mathbf{x})\right\rangle - \frac{1}{2L_k}\|\mathbf{g}_k\|^2. \quad (3.5)$$

For our deep-cut it is known that at iteration k we only discard points $\mathbf{y} \in \mathbb{R}^n$ which satisfy

$$\langle \mathbf{g}_k, \mathbf{c}_k - \mathbf{y}\rangle < \frac{1}{2L_k}\|\mathbf{g}_k\|^2 + f\left(T_{L_k}(\mathbf{c}_k)\right) - \ell_k.$$

Therefore, we discard in particular all points $\mathbf{x} \in \mathbb{R}^n$ which satisfy

$$\left\langle \mathbf{g}_k, \mathbf{c}_k - T_{L(f)}(\mathbf{x})\right\rangle < \frac{1}{2L_k}\|\mathbf{g}_k\|^2 + f\left(T_{L_k}(\mathbf{c}_k)\right) - \ell_k.$$

Combining the latter with (3.5) we get that at iteration k we discard only points $\mathbf{x} \in \mathbb{R}^n$ which satisfy $f\left(T_{L(f)}(\mathbf{x})\right) > \ell_k$. Suppose now that for a given $\varepsilon > 0$ we have $\ell_m > f^* + \varepsilon$. Therefore, under this assumption, it follows that during the IDCE method we only discard points satisfying $f\left(T_{L(f)}(\mathbf{x})\right) > f^* + \varepsilon$, which combined with the fact that the initial ellipsoid is the sphere $E(\mathbf{0}, R^2\mathbf{I})$ implies that

$$\left\{\mathbf{x} \in \mathbb{R}^n : f\left(T_{L(f)}(\mathbf{x})\right) \le f^* + \varepsilon\right\} \cap E(\mathbf{0}, R^2\mathbf{I}) \subseteq E_m. \quad (3.6)$$

Let $\mathbf{x}^* \in X^*$. Taking $\mathbf{w} = \mathbf{x} \in \mathbb{R}^n$ and $\mathbf{z} = \mathbf{x}^* \in X^*$ in (2.4) we have

$$\begin{aligned} f\left(T_{L(f)}(\mathbf{x})\right) &\le f\left(\mathbf{x}^*\right) + \left\langle G_{L(f)}(\mathbf{x}), \mathbf{x} - \mathbf{x}^*\right\rangle - \frac{1}{2L(f)}\left\|G_{L(f)}(\mathbf{x})\right\|^2 \\ &\le f\left(\mathbf{x}^*\right) + \left\|G_{L(f)}(\mathbf{x})\right\|\left\|\mathbf{x} - \mathbf{x}^*\right\|. \end{aligned} \quad (3.7)$$

From Lemma 2.2 we obtain that

$$\begin{aligned} \left\|G_{L(f)}(\mathbf{x})\right\|^2 &\le 2L(f)\left\langle G_{L(f)}(\mathbf{x}), \mathbf{x} - \mathbf{x}^*\right\rangle \\ &\le 2L(f)\left\|G_{L(f)}(\mathbf{x})\right\|\left\|\mathbf{x} - \mathbf{x}^*\right\|. \end{aligned}$$

Thus,

$$\left\|G_{L(f)}(\mathbf{x})\right\| \le 2L(f)\left\|\mathbf{x} - \mathbf{x}^*\right\|,$$

which combined with (3.7) yields

$$f\left(T_{L(f)}(\mathbf{x})\right) \le f\left(\mathbf{x}^*\right) + 2L(f)\left\|\mathbf{x} - \mathbf{x}^*\right\|^2.$$

If $\|\mathbf{x} - \mathbf{x}^*\| \le \sqrt{\frac{\varepsilon}{2L(f)}}$, then we get that

$$f\left(T_{L(f)}(\mathbf{x})\right) \le f\left(\mathbf{x}^*\right) + 2L(f)\left\|\mathbf{x} - \mathbf{x}^*\right\|^2 \le f\left(\mathbf{x}^*\right) + \varepsilon.$$

Hence, by (3.6),

$$B \cap E(\mathbf{0}, R^2\mathbf{I}) \subseteq E_m, \quad (3.8)$$

where

$$B := \left\{\mathbf{x} \in \mathbb{R}^n : \left\|\mathbf{x} - \mathbf{x}^*\right\| \le \sqrt{\frac{\varepsilon}{2L(f)}}\right\}.$$

By the definition of R, we have $\|\mathbf{x}^*\| \le \frac{R}{2}$. In addition, it follows that any $\mathbf{x} \in B$ satisfies $\|\mathbf{x}\| \le \|\mathbf{x}^*\| + \|\mathbf{x} - \mathbf{x}^*\| \le \frac{R}{2} + \frac{R}{2} = R$, so that $B \subseteq E(\mathbf{0}, R^2\mathbf{I})$ and therefore the inclusion (3.8) reads as

$$B \subseteq E_m.$$

Therefore $\mathrm{Vol}\,(B) \le \mathrm{Vol}\,(E_m)$, and we have from Lemma 3.1 (υ_n is the volume of the unit-ball in $\mathbb{R}^n$):

$$\begin{aligned} \mathrm{Vol}\,(B) = \upsilon_n\left(\sqrt{\frac{\varepsilon}{2L(f)}}\right)^n &\le \mathrm{Vol}\,(E_m) \\ &\le e^{-\frac{m}{2(n+1)}}\left(1 - \alpha_{j_m}\right)^{\frac{mn}{2}} \upsilon_n R^n. \end{aligned}$$

Thence

$$\left(\sqrt{\frac{\varepsilon}{2L(f)}}\right)^n \le e^{\frac{-m}{2(n+1)}}\left(1 - \alpha_{j_m}\right)^{\frac{mn}{2}} R^n.$$

Thus,

$$\varepsilon \le 2L(f)R^2\left[e^{\frac{-1}{(n(n+1))}}\right]^m\left(1 - \alpha_{j_m}\right)^m,$$

which combined with the fact that $0 \le \alpha_{j_m} \le 1$ yields

$$\varepsilon \le 2L(f)R^2\left[e^{\frac{-1}{(n(n+1))}}\right]^m.$$

The latter inequality is equivalent to $m \le n(n+1)\ln\left(\frac{2L(f)R^2}{\varepsilon}\right)$, which is a contradiction to (3.3), thus showing the desired result $\ell_m - f^* \le \varepsilon$. □

4. Numerical results

In this section we present several numerical results which illustrate the advantage of the IDCE method over the following three methods:

- The classical ellipsoid method (CE).
- The deep-cut ellipsoid method developed in [5] (DCE).
- The projected gradient method (PG) with a constant stepsize (chosen to be $1/L(f)$) [2].

We will test the four methods on the bound-constrained quadratic problem:

$$(\mathrm{P}): \quad \begin{array}{ll} \min & \mathbf{x}^T\mathbf{A}\mathbf{x} \\ \text{s.t.} & \mathbf{x} \in X, \end{array}$$

where $\mathbf{A}$ is a positive definite $n \times n$ matrix and $X = [-1, 1]^n$. The unique optimal solution is obviously $\mathbf{x} = \mathbf{0}$. Fig. 1 describes the progress of the minimal objective function value for $n = 7$ with $\mathbf{A}$ being randomly chosen. It is clear from the figure that in this

Table 1
Number of problems (out of 100) for which an ε-optimal solution is reached after 100 iterations.

n	10^{-3}				10^{-5}				10^{-6}			
	CE	DCE	IDCE	PG	CE	DCE	IDCE	PG	CE	DCE	IDCE	PG
5	100	100	**100**	87	62	51	**100**	72	9	7	**100**	68
6	99	99	**99**	91	30	36	**99**	79	1	5	**95**	73
7	**100**	**100**	99	96	12	15	**99**	85	0	0	73	**84**
8	97	95	96	**100**	12	20	**96**	95	2	3	44	**92**
9	88	89	91	**98**	9	9	83	**92**	0	2	26	**92**
10	86	90	93	**100**	9	3	74	**98**	0	1	20	**98**

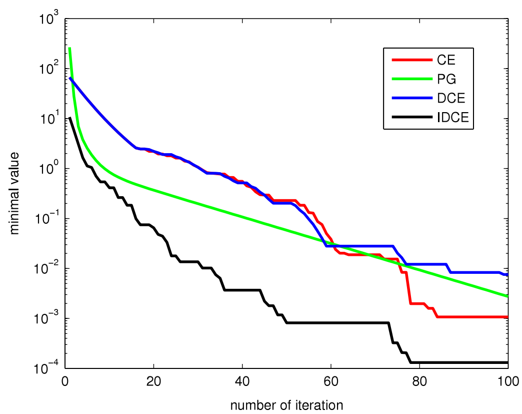

Fig. 1. The minimal objective function value for the first 100 iterations of the four methods (7×7 example).

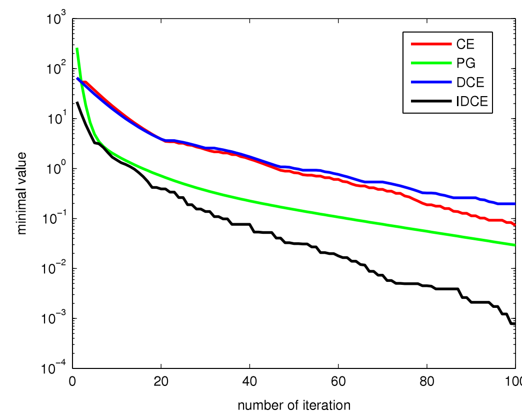

Fig. 2. The minimal objective function value for the first 100 iterations of the four methods (9×9 example).

example after 100 iterations, all the three other methods reach the unique optimal solution with an accuracy of no more than 10^{-3} and the IDCE method reaches the unique solution approximation of 10^{-4}. The IDCE method reaches an 10^{-3}-optimal solution after only 45 iterations.

We repeated this experiment with $n = 9$ and a randomly generated $\mathbf{A}$ and the results are shown in Fig. 2. The results here are similar to the previous example, but here the three other methods behave even more poorly—they do not reach much more than approximately 10^{-1} accuracy.

We made a more extensive set of tests in which we solved 100 problems—each corresponding to a realization of the positive definite $n \times n$ matrix $\mathbf{A}$ for $n = 5, \ldots, 10$ matrix. In Table 1, For each n, and for each choice of an algorithm, we indicate how many problems (out of the 100) reached an ε-optimal solution for $\varepsilon = 10^{-3}$, 10^{-5}, 10^{-6} after 100 iterations.

For the moderate accuracy $\varepsilon = 10^{-3}$, the IDCE is usually slightly better than the other ellipsoid variants CE and DCE, but the PG method is a bit better for $n = 8, 9, 10$. For the greater accuracies $\varepsilon = 10^{-5}, 10^{-6}$ it is clear that the IDCE method significantly outperforms CE and DCE and is better than PG for the smaller dimensions.

References

[1] A. Beck, M. Teboulle, Gradient-based algorithms with applications to signal recovery problems, in: D. Palomar, Y. Eldar (Eds.), Convex Optimization in Signal Processing and Communications, Cambridge University Press, 2009, pp. 139–162.
[2] D.P. Bertsekas, Nonlinear Programming, second ed., Athena Scientific, Belmont MA, 1999.
[3] R.G. Bland, D. Goldfarb, M.J. Todd, The ellipsoid method: a survey, Oper. Res. 29 (1981) 1039–1091.
[4] J.G. Ecker, M. Kupferschmid, An ellipsoid algorithm for nonlinear programming, Math. Program. 27 (1983) 83–106.
[5] J.B.G. Frenk, J. Gromicho, S. Zhang, A deep cut ellipsoid algorithm for convex programming: theory and applications, Math. Program. 63 (1994) 83–108.
[6] L.G. Khachiyan, A polynomial algorithm in linear programming, Doklady Akademiia Nauk SSSR 244 (1979) 1093–1096; translated in Soviet Mathematics Doklady 20 (1979) 191–194.
[7] Y. Nesterov, Introductory Lectures on Convex Optimization, Kluwer, Boston, 2004.