Accelerated Difference of Convex functions Algorithm and its Application to Sparse Binary Logistic Regression

Duy Nhat Phan, Hoai Minh Le, Hoai An Le Thi
Department Computer Science and Application, LGIPM, University of Lorraine
{duy-nhat.phan, minh.le, hoai-an.le-thi}@univ-lorraine.fr

Abstract

In this work, we present a variant of DCA (Difference of Convex function Algorithm) with the aim of improving its performance. The proposed algorithm, named Accelerated DCA (*ADCA*), consists in incorporating the Nesterov's acceleration technique into DCA. We first investigate *ADCA* for solving the standard DC program and rigorously study its convergence properties and the convergence rate. Secondly, we develop *ADCA* for a special case of the standard DC program whose the objective function is the sum of a differentiable function with L-Lipschitz continuous gradient (possibly nonconvex) and a DC function. We exploit the special structure of the problem to propose an efficient DC decomposition for which the corresponding *ADCA* scheme is inexpensive. As an application, we consider the sparse binary logistic regression problem. Numerical experiments on several benchmark datasets illustrate the efficiency of our algorithm and its superiority over well-known methods.

1 Introduction

A standard DC (Difference of Convex functions) program is of the form

$$\min_{x\in\mathbb{R}^n} \left\{F(x) := G(x) - H(x)\right\}, \tag{1}$$

where G and H are lower semi-continuous proper convex functions on $\mathbb{R}^n$. Such a function F is called a DC function while G and H are the DC components of F. Note that a convex constraint $x \in C$ can be incorporated into the objective function F by using the indicator function on C, defined by $\chi_C = 0$ if $x \in C$ and $\chi_C = +\infty$ if $x \notin C$. The DC program (1) plays a key role in optimization since almost nonconvex programs encountered in practice can be formulated/reformulated as a DC program.

DCA (DC Algorithm) for solving the DC program (1) was introduced in 1985 by Pham Dinh Tao and extensively developed by Le Thi Hoai An and Pham Dinh Tao since 1994 to become now classic and increasingly popular ([Le Thi and Pham Dinh, 2005; 2018; Pham Dinh and Le Thi, 1997; 2014] and references therein). The original key idea of DCA relies on the structure DC of the objective function. DCA works with the DC components G and H but not directly with the function F. The main idea of DCA is simple: each iteration l of DCA approximates the concave part $-H$ by its affine majorization (that corresponds to taking $y^k \in \partial H(x^k)$) and computes x^{k+1} by solving the resulting convex problem.

$$\min\{G(x) - \langle y^k, x\rangle : x \in \mathbb{R}^n\} \quad (P_k).$$

DCA has been successfully applied to various nonconvex/nonsmooth programs thanks to its versatility, flexibility, robustness, inexpensiveness and their adaptation to specific structure of considered problems. It has been proved that with an appropriate DC decompositions and a suitably equivalent DC reformulations, DCA permits to recover most of standard methods in convex and nonconvex programming. Several well-known methods in Machine Learning such as Expectation-Maximization (EM) [Dempster *et al.*, 1977], Successive Linear Approximation (SLA) [Bradley and Mangasarian, 1998], Iterative Shrinkage-Thresholding Algorithms (ISTA) [Chambolle *et al.*, 1998], and Convex-Concave Procedure (CCCP) [Yuille and Rangarajan, 2003] are special cases of DCA. The readers are referred to the recent paper [Le Thi and Pham Dinh, 2018] for an extensive overview of thirty years of development of DCA.

Nowadays, especially with the Big Data explosion, it has been becoming more and more important to develop advanced optimization methods able to handle very large-scale problems. Motivated by the success of DCA on several nonconvex/nonsmooth programs to which it was proved to be more robust and more efficient than related standard methods, we aim to investigate a variant of DCA in order to improve its performance.

Paper's contribution: the contributions of this paper is multiples.

Firstly, we introduce Accelerated DCA (*ADCA*) for solving the standard DC program (1). In *ADCA*, we incorporate the Nesterov's acceleration technique into standard DCA in order to improve its performance. The idea of *ADCA* is different to the usual line search acceleration using Armijo type rule which can be computationally costly. The acceleration step in *ADCA* which consists in using an extrapolated point from the current iterate and the previous one, aims to find a point z^k which is better than x^k for the computation of x^{k+1}.

We then provide a rigorous argument to prove the interesting convergence properties of *ADCA* as well as the convergence rate under the Lojasiewicz assumption.

Secondly, we investigate *ADCA* for a special case of the standard DC program, namely the sum of two nonconvex function minimization problem

$$\min_{x\in\mathbb{R}^n} f(x) + r(x), \tag{2}$$

where f is a differentiable function with L-Lipschitz continuous gradient (possibly nonconvex) and r is a DC function. The problem (2) covers several nonconvex and nonsmooth problems arising from various fields such as machine learning, computational biology, signal processing, etc.

The problem (2) has been attracting attention of many researchers. Proximal gradient (PG) methods, which are also known as different names ISTA, fixed point iteration, forward-backward splitting (see e.g. [Daubechies *et al.*, 2004; Beck and Teboulle, 2009b; Combettes and Wajs, 2005; Hale *et al.*, 2008]), have been extensively developed for the convex case of (2), i.e., both f and r are convex. In [Nesterov, 1983], the author introduced the first accelerated proximal gradient (APG) method for solving (2) with f convex and $r = 0$. Later, Beck and Teboulle [Beck and Teboulle, 2009a; 2009b] extended it for the case where both f and r are convex. These algorithms exhibit a global convergence rate $\mathcal{O}(1/k^2)$, where k is the iteration counter. Recently, several extensions of APG method for the convex case of (2) have been proposed for the nonconvex cases. For instance, [Li and Lin, 2015] have extended the method of [Beck and Teboulle, 2009a] for f differentiable with L-Lipschitz gradient and r nonconvex. In [Gu *et al.*, 2018], the authors presented inexact versions of PG and APG for solving (2) with f smooth and r nonsmooth (possibly nonconvex). Another inexact APG for the nonconvex case of (2) which only requires one proximal step at each iteration, were proposed in [Yao *et al.*, 2017]. However, the aforementioned algorithms have to compute the proximal map of nonconvex functions r which do not has closed form in many cases. Usually, this computation can be very expensive or impossible.

In this work, by exploiting the properties of f and r, we propose an efficient DC decomposition for which the corresponding DCA and ADCA are inexpensive.

Finally, as an application, we consider the sparse binary logistic regression and carefully perform numerical experiments of all proposed algorithms.

The remainder of the paper is organized as follows. In Section 2 we introduce *ADCA* for the standard DC program and study its convergence properties. *ADCA* for solving the problem (2) is presented in Section 3. The numerical experiments on the sparse binary logistic regression problem are reported in Section 4 and Section 5 concludes the paper.

2 Accelerated DCA for Standard DC Program

Before presenting the Accelerated DCA, let us recall some basis notations that will be used in the sequel.

The modulus of strong convexity of θ on Ω, denoted by $\mu(\theta, \Omega)$ or $\mu(\theta)$ if $\Omega = \mathbb{R}^n$, is given by

$$\mu(\theta, \Omega) = \sup\{\mu \geq 0 : \theta - (\mu/2)\|.\|^2 \text{ is convex on } \Omega\}.$$

One says that θ is *strongly convex* on Ω if $\mu(\theta, \Omega) > 0$.

For a convex function θ, the subdifferential of θ at $x_0 \in \mathrm{dom}\theta := \{x \in \mathbb{R}^n : \theta(x_0) < +\infty\}$, denoted by $\partial\theta(x_0)$, is defined by

$$\partial\theta(x_0) := \{y \in \mathbb{R}^n : \theta(x) \geq \theta(x_0) + \langle x - x_0, y\rangle, \forall x \in \mathbb{R}^n\}.$$

The subdifferential $\partial\theta(x_0)$ generalizes the derivative in the sense that θ is differentiable at x_0 if and only if $\partial\theta(x_0) \equiv \{\nabla_x\theta(x_0)\}$.

A point x^* is called a *critical point* of $G - H$, or a generalized Karush-Kuhn-Tucker point (KKT) of (P_{dc})) if

$$\partial H(x^*) \cap \partial G(x^*) \neq \emptyset. \tag{3}$$

We now introduce the Accelerated DCA (*ADCA*) for solving the standard DC program (1). According to the DCA scheme, at each iteration, one computes $y^k \in \partial H(x^k)$ then solves the convex sub-problem (1) to get x^{k+1}. The idea of *ADCA*, in order to accelerate DCA, is to find a point z^k which is better than x^k for the computation of x^{k+1}. In this work, we consider z^k as an extrapolated point of the current iterate x^k and the previous iterate x^{k-1}:

$$z^k = x^k + \frac{t_k - 1}{t_{k+1}}\left(x^k - x^{k-1}\right),$$

where $t_{k+1} = \frac{1+\sqrt{1+4t_k^2}}{2}$. If z^k is better than one of last q iterates $\{x^{k-q}, \ldots, x^{k-1}, x^k\}$ in term of objective function, i.e., $F(z^k) \leq \max\limits_{t=\max(0,k-q),\ldots,k} F(x^t)$ then z^k will be used instead of x^k to compute y^k. This condition allows the objective function F to increase and consequently to escape from a potential bad local minimum [Grippo and Sciandrone, 2002; Wright *et al.*, 2009]. Theoretically, a large value of q increases the chance of using the extrapolated points z^k in *ADCA* and consequently increases its chance to accelerate. Note that if $q = 0$ then $F(z^k) \leq F(x^k)$ and *ADCA* is a monotone algorithm like DCA. *ADCA* is described in Algorithm 1.

Algorithm 1 *ADCA* for solving the standard DC program (1)

```
Initialization: Choose an initial point $x^0$, $z^0 = x^0$, $q \in \mathbb{N}$, $t_0 = 1$, and $k \leftarrow 0$.
repeat
    Compute $t_{k+1} = \frac{1+\sqrt{1+4t_k^2}}{2}$.
    Compute $z^k = x^k + \frac{t_k-1}{t_{k+1}}\left(x^k - x^{k-1}\right)$ if $k \geq 1$.
    If $F(z^k) \leq \max_{t=\max(0,k-q),\ldots,k} F(x^t)$ then set $v^k = z^k$, otherwise set $v^k = x^k$.
    Compute $y^k \in \partial H(v^k)$.
    Compute $x^{k+1} = arg\min_{x\in\mathbb{R}^n}\left\{G(x) - \langle y^k, x\rangle\right\}$.
    $k \leftarrow k + 1$.
until Stopping criterion.
```

Remark 1. *ADCA does not require any particular property of the sequence $\{t_k\}$ for extrapolation. The above sequence t was chosen thanks to its interesting convergence rate proved in [Beck and Teboulle, 2009b]. More precisely, the well-known FISTA algorithm [Beck and Teboulle, 2009b] for solving the convex case of* (2)*, i.e. both $f(x)$ and $r(x)$ are convex, is nothing else but a special case of ADCA in which we always use the extrapolated point z^k instead of the last iterate x^k. In FISTA, the extrapolated point v^k is updated with $t_{k+1} = \frac{1+\sqrt{1+4t_k^2}}{2}$ and the authors have proved that FISTA has a interesting convergence rate $O(\frac{1}{k^2})$.*

2.1 Convergence Analysis of ADCA

In this subsection, we study the convergence of *ADCA*. Our first result provides the behavior of the limit points of the sequence $\{x^{\phi(k)}\}$ generated by *ADCA*, where $\phi(k) = \text{argmin}_{t=k+1,...,k+1+q}\|x^t - v^{t-1}\|^2$.

Theorem 1. *Let $\mu(G)$ and $\mu(H)$ be the convex modulus of G and H, respectively. If $\alpha = \inf_{x\in\mathbb{R}^n} F(x) > -\infty$ and $\min\{\mu(G), \mu(H)\} > 0$, then for any subsequence $\{x^{\phi(k_j)}\}$ of $\{x^{\phi(k)}\}$, converging to x^* such that $\{y^{\phi(k_j)-1}\}$ is bounded, the limit point x^* is a critical point of* (1)*.*

To prove Theorem 1, we will use the following lemma. Denote by $\{\Gamma^k\}$ the sequence defined as

$$\Gamma^k \quad = \max_{t=\max(0,k-q),...,k} F(x^t).$$

Lemma 1. *Let $\{x^k\}$ and $\{v^k\}$ be sequences generated by ADCA. The following statements hold.*

(i) For any $k = 0, 1, ...,$

$$\Gamma^k - \Gamma^{k+1+q} \geq \frac{\mu(G) + \mu(H)}{2}\|x^{\phi(k)} - v^{\phi(k)-1}\|^2. \quad (4)$$

As a result, by choosing $q = 0$, we get the monotone property of $\{F(x^k)\}$, i.e., $F(x^k) - F(x^{k+1}) \geq \frac{\mu(G)+\mu(H)}{2}\|x^{k+1} - v^k\|^2$.

(ii) If $\alpha = \inf_{x\in\mathbb{R}^n} F(x) > -\infty$ and $\min\{\mu(G), \mu(H)\} > 0$, then $\sum_{k=0}^{+\infty} \|x^{\phi(k)} - v^{\phi(k)-1}\|^2 < +\infty$, and therefore $\lim_{k\to+\infty} \|x^{\phi(k)} - v^{\phi(k)-1}\| = 0$.

Proof. First let us justify (i) by noting from the μ-convexity of G and $y^k \in \partial G(x^{k+1})$ that

$$G(v^k) \geq G(x^{k+1}) + \langle y^k, v^k - x^{k+1}\rangle + \frac{\mu(G)}{2}\|v^k - x^{k+1}\|^2.$$

It follows from the μ-convexity of H and $y^k \in \partial H(v^k)$ that

$$H(x^{k+1}) \geq H(v^k) + \langle y^k, x^{k+1} - v^k\rangle + \frac{\mu(H)}{2}\|x^{k+1} - v^k\|^2.$$

Summing two above inequalities gives us

$$F(v^k) - F(x^{k+1}) \geq \frac{\mu(G) + \mu(H)}{2}\|x^{k+1} - v^k\|^2. \quad (5)$$

Observe that $F(v^k) \leq \max_{t=\max(0,k-d),...,k} F(x^t) = \Gamma^k$. It follows from this and (5) that

$$F(x^{k+1}) \leq \Gamma^k - \frac{\mu(G) + \mu(H)}{2}\|x^{k+1} - v^k\|^2. \quad (6)$$

This implies that $F(x^{k+1}) \leq \Gamma^k$. We prove by induction that for all $t = 0, ..., q$

$$F(x^{k+1+t}) \leq \Gamma^k - \frac{\mu(G) + \mu(H)}{2}\|x^{k+1+t} - v^{k+t}\|^2. \quad (7)$$

Indeed, it follows from (6) that the claim holds for $t = 0$. We suppose that it also holds for $t = 0, ..., p-1$ with $1 \leq p \leq q$. Thus, we have

$$\begin{aligned} F(x^{k+1+p}) \quad &\leq \Gamma^{k+p} - \tfrac{\mu(G)+\mu(H)}{2}\|x^{k+1+p} - v^{k+p}\|^2 \\ &\leq \max(\Gamma^k, F(x^{k+1}), ..., F(x^{k+p})) \\ &\quad - \tfrac{\mu(G)+\mu(H)}{2}\|x^{k+1+p} - v^{k+p}\|^2 \\ &\leq \Gamma^k - \tfrac{\mu(G)+\mu(H)}{2}\|x^{k+1+p} - v^{k+p}\|^2, \end{aligned}$$

where last inequality follows from $F(x^{k+1+t}) \leq \Gamma^k$ for $t = 0, ..., p-1$. Therefore, we obtain

$$\begin{aligned} \Gamma^{k+q+1} \quad &= \max_{t=k+1,...,k+q+1} F(x^t) \\ &\leq \Gamma^k - \min_{t=k+1,...,k+1+q} \tfrac{\mu(G)+\mu(H)}{2}\|x^t - v^{t-1}\|^2 \\ &= \Gamma^k - \tfrac{\mu(G)+\mu(H)}{2}\|x^{\phi(k)} - v^{\phi(k)-1}\|^2. \end{aligned}$$

Next let us prove (ii) by noting that $\Gamma^k \geq \alpha$ for all k. Summing (4) over $k = 0, ..., N$ gives us that

$$\begin{aligned} \tfrac{\mu(G)+\mu(H)}{2} \quad &\textstyle\sum_{k=0}^N \|x^{\phi(k)} - v^{\phi(k)-1}\|^2 \\ &\leq \textstyle\sum_{t=0}^q (\Gamma^t - \Gamma^{N+t+1}) \\ &\leq (q+1)(\max_{t=0,...,q} F(x^t) - \alpha). \end{aligned}$$

Since $\min\{\mu(G), \mu(H)\} > 0$, we have

$$\textstyle\sum_{k=0}^N \|x^{\phi(k)} - v^{\phi(k)-1}\|^2 \leq \frac{2(q+1)(\max\limits_{t=0,...,q} F(x^t) - \alpha)}{\mu(G)+\mu(H)}.$$

Passing to the limit over the sequence $\{N\}_{N\in\mathbb{N}}$ tells us that

$$\textstyle\sum_{k=0}^{+\infty} \|x^{\phi(k)} - v^{\phi(k)-1}\|^2 < +\infty, \quad (8)$$

and therefore $\lim_{k\to+\infty} \|x^{\phi(k)} - v^{\phi(k)-1}\| = 0$. □

Proof of Theorem 1. Let $\{x^{\phi(k_j)}\}$ be a subsequence of $\{x^{\phi(k)}\}$ that converges to x^*. It follows from (ii) of Lemma 1 that $\lim_{j\to+\infty} v^{\phi(k_j)-1} = x^*$. Without loss of generality, we can suppose that the sequence $\{y^{\phi(k_j)-1}\}$ converges to y^*. By the closed property of the subdifferential mapping ∂H, we have $y^* \in \partial H(x^*)$. We note that

$$x^{\phi(k_j)} \in \text{argmin}\{G(x) - \langle y^{\phi(k_j)-1}, x\rangle\}.$$

This implies that $y^{\phi(k_j)-1} \in \partial G(x^{\phi(k_j)})$. By the closedness of ∂G, we obtain $y^* \in \partial G(x^*)$. Therefore, $y^* \in \partial G(x^*) \cap \partial H(x^*)$. It follows from this that x^* is a critical point of the DC program (1). □

Recall that a lower semicontinuous function F has the Lojasiewicz property [Attouch and Bolte, 2009] if for any limiting-critical point x^*, that is $0 \in \partial^L F(x^*)$, there exist $C, \epsilon > 0$ and $\theta \in [0, 1)$ such that

$$|F(x) - F(x^*)|^\theta \leq C\|\hat{x}\|, \; \forall x \in B(x^*, \epsilon), \; \forall \hat{x} \in \partial^L F(x).$$

Here $\partial^L F(x)$ denotes the limiting-subdifferential of F at x. The class of functions having the Lojasiewicz property is very ample, for example, semi-algebraic, subanalytic, and log-exp functions. For studying the convergence rate of ADCA, we state the results concerning the following lemma.

Lemma 2. *Consider the settings of Theorem 1. Let $\{x^k\}$ be sequence generated by ADCA with $q = 0$. Denote by Ω the set of limit points of $\{x^k\}$. Suppose further that $\{x^k\}$ and $\{y^k\}$ are bounded, and F is lower semicontinuous. The following statements hold.*

(i) Ω is a compact set and $\lim_{k\to\infty} F(x^k) = F(x^*)$ for some $x^* \in \Omega$. Thus, F has the same value on Ω, which is denoted by F^*.

(ii) If F has the Lojasiewicz property and H is differentiable, then there exist $C, \epsilon > 0$ and $\theta \in [0, 1)$ such that $\forall x \in \{x \in \mathbb{R}^n$: $dist(x, \Omega) \le \epsilon\}$, one has

$$|F(x) - F^*|^\theta \le C\|\hat{x}\|, \tag{9}$$

$\forall \hat{x} \in \partial^L F(x)$.

(iii) If F has the Lojasiewicz property and G is differentiable, then there exist $C, \epsilon > 0$ and $\theta \in [0, 1)$ such that $\forall x \in \{x \in \mathbb{R}^n$: $dist(x, \Omega) \le \epsilon\}$, one has

$$|F(x) - F^*|^\theta \le C\|\hat{x}\|, \tag{10}$$

$\forall \hat{x} \in \partial^L(-F)(x)$.

Proof. (i) Since $\{x^k\}$ is bounded and Ω is the set of its limit points, Ω is a compact set. It follows from $\alpha = \inf_{x\in\mathbb{R}^n} F(x) > -\infty$ and (i) of Lemma 1 that the sequence $\{F(x^k)\}$ is non-increasing and bounded below. Thus, there exists $F^* = \lim_{k\to\infty} F(x^k)$. Let $x^* \in \Omega$. There exists a subsequence $\{x^{k_j}\}$ that converges to x^*. Without loss of generality, we can assume that $\{y^{k_j-1}\}$ converges to y^*. We note that

$$x^{k_j} \in \text{argmin}\{G(x) - \langle y^{k_j-1}, x - v^{k_j-1}\rangle\}.$$

This implies that for all x,

$$\begin{aligned} G(x^{k_j}) \quad & -\langle y^{k_j-1}, x^{k_j} - v^{k_j-1}\rangle \\ & \le G(x^*) - \langle y^{k_j-1}, x^* - v^{k_j-1}\rangle. \end{aligned}$$

Taking $j \to +\infty$ gives us that

$$\limsup_{j\to+\infty} G(x^{k_j}) \le G(x^*). \tag{11}$$

Therefore, we have

$$\begin{aligned} \limsup_{j\to\infty} F(x^{k_j}) & = \limsup_{j\to\infty}[G(x^{k_j}) - H(x^{k_j})] \\ & \le \limsup_{j\to\infty} G(x^{k_j}) - \liminf_{j\to\infty} H(x^{k_j}) \\ & \le G(x^*) - \liminf_{j\to\infty} H(x^{k_j}) \\ & \le G(x^*) - H(x^*) = F(x^*), \end{aligned}$$

where the second inequality follows from (11) and the last inequality holds by the lower semicontinuity of H. On the other hand, from the lower semicontinuity of F, we obtain $\liminf_{j\to\infty} F(x^{k_j}) \ge F(x^*)$. Hence, by the uniqueness of limit, we have $F^* = F(x^*)$.

Let us justify (ii) by noting that $\partial^L F(x^*) = \partial G(x^*) - \nabla H(x^*)$ for all $x^* \in \Omega$. Hence, Ω is a subset of the limiting-critical points of F. According to Lemma 1 in [Attouch and Bolte, 2009], applied to the function F, there exist $C, \epsilon > 0$ and $\theta \in [0, 1)$ such that $\forall x \in \mathbb{R}^n$, $\text{dist}(x, \Omega) \le \epsilon$, $\forall \hat{x} \in \partial^L F(x)$, one has

$$|F(x) - F^*|^\theta \le C\|\hat{x}\|.$$

By using a similar argument, we have the result (iii). □

We now provide the asymptotic convergence rate of ADCA under the Lojasiewicz assumption.

Theorem 2. *Consider the settings of Theorem 1. Suppose further that either G or H is differentiable with locally Lipschitz derivative. Let $\{x^k\}$ be sequence generated by ADCA with $q = 0$. Assume that F is is lower semicontinuous and has the Lojasiewicz property, and $\{x^k\}, \{y^k\}$ are bounded. Denote by θ the parameter, which is defined as in Lemma 2. The following estimations hold*

(i) If $\theta = 0$, then the sequence $\{F(x^k)\}$ converges to F^* in a finite number of steps.

(ii) If $\theta \in (0, 1/2]$, then the sequence $\{F(x^k)\}$ converges linearly to F^*.

(iii) If $\theta \in (1/2, 1)$, then there exist positive constants η and N_0 such that $F(x^k) - F^* \le \eta k^{-\frac{1}{2\theta-1}}$, for all $k \ge N_0$.

Proof. Let us consider the following cases.

Case 1. G is differentiable and its derivative is locally Lipschitz. For each $x \in \Omega$, there exist $L_x, \epsilon_x > 0$ such that

$$\|\nabla G(u) - \nabla G(v)\| \le L_x\|u - v\| \; \forall u, v \in B(x, \epsilon_x). \tag{12}$$

From the compactness of Ω, there exist $w^1, ..., w^m \in \Omega$ such that $\Omega \subset \cup_{i=1}^m B(w^i, \epsilon_{w^i}/4)$, where $B(w, \epsilon)$ is the open ball with the center w and radius ϵ. Set $L = \max\{L_{w^i} : i = 1, ..., m\}$ and $\epsilon = \min\{\epsilon_{w^i}/2 : i = 1, ..., m\}$. It follows from the (ii) of Lemma 1 that $\{v^k\}$ and $\{x^k\}$ share the same the set of limit points Ω. Hence, there exists $N_1 > 0$ such that $v^k \in \cup_{i=1}^m B(w^i, \epsilon_{w^i}/2)$ and $\|x^{k+1} - v^k\| \le \epsilon$ whenever $k \ge N_1$. Thus, for any $k \ge N_1$, there is w^i such that $x^{k+1}, v^k \in B(w^i, \epsilon_{w^i})$. This implies that

$$\begin{aligned} \|\nabla G(x^{k+1}) - \nabla G(v^k)\| & \le L_{w^i}\|x^{k+1} - v^k\| \\ & \le L\|x^{k+1} - v^k\|. \end{aligned} \tag{13}$$

On the other hand, since G is differentiable and by the definition of x^{k+1}, we have

$$\begin{aligned} \nabla G(x^{k+1}) - \nabla G(v^k) & = y^k - \nabla G(v^k) \\ & \in \partial H(v^k) - \nabla G(v^k) \\ & = \partial^L(-F)(v^k). \end{aligned}$$

Therefore, from the (iii) of Lemma 2 and $\text{dist}(v^k, \Omega) \to 0$, by increasing N_1 if necessary, we have for all $k \ge N_1$

$$|F(v^k) - F^*|^\theta \le C\|G(x^{k+1}) - \nabla G(v^k)\|.$$

Combining this and $F(v^k) - F^* \ge F(x^{k+1}) - F^* \ge 0$ gives us that

$$\begin{aligned} |F(x^{k+1}) - F^*|^{2\theta} & \le C^2\|G(x^{k+1}) - \nabla G(v^k)\|^2 \\ & \le (CL)^2\|x^{k+1} - v^k\|^2 \\ & \le \tfrac{2(CL)^2}{\mu(G)+\mu(H)}[F(x^k) - F(x^{k+1}], \end{aligned}$$

where the second inequality follows from (13) and the last inequality follows from (i) of Lemma 1. Hence, by setting $r_k = F(x^k) - F^*$, we obtain

$$r_{k+1}^{2\theta} \le \frac{2(CL)^2}{\mu(G) + \mu(H)}[r_k - r_{k+1}]. \tag{14}$$

Case 2. H is differentiable and its derivative is locally Lipschitz. Similar to Case 1, we can find $L, N_2 > 0$ such that for any $k \ge N_2$,

$$\|\nabla H(v^k) - \nabla H(x^{k+1})\| \le L\|v^k - x^{k+1}\|. \tag{15}$$

Since H is differentiable and by the definition of x^{k+1}, we have

$$\begin{aligned}\nabla H(v^k) - \nabla H(x^{k+1}) &= y^k - \nabla H(x^{k+1}) \\ &\in \partial G(x^{k+1}) - \nabla H(x^{k+1}) \\ &= \partial^L F(x^{k+1}).\end{aligned} \tag{16}$$

Therefore, from the (ii) of Lemma 2 and $\text{dist}(x^{k+1}, \Omega) \to 0$, by increasing N_2 if necessary, we have for all $k \geq N_2$,

$$\begin{aligned}|F(x^{k+1}) - F^*|^{2\theta} &\leq C^2 \|\nabla H(v^k) - \nabla H(x^{k+1})\|^2 \\ &\leq (CL)^2 \|v^k - x^{k+1}\|^2 \\ &\leq \tfrac{2(CL)^2}{\mu(G)+\mu(H)} [F(x^k) - F(x^{k+1})],\end{aligned}$$

where the second inequality follows from (15) and the last inequality follows from the (i) of Lemma 1. Hence, we obtain

$$r_{k+1}^{2\theta} \leq \frac{2(CL)^2}{\mu(G) + \mu(H)} [r_k - r_{k+1}]. \tag{17}$$

Thus, from (14) and (17), we have shown that, in both cases, there exists $\tau > 0$ such that

$$r_{k+1}^{2\theta} \leq \tau [r_k - r_{k+1}]. \tag{18}$$

By using similar arguments in [Frankel *et al.*, 2015], it is easy to show that sequence $\{r_k\}$ satisfying the above inductive property converges to zero at different rates according to θ as stated in the theorem. □

3 ADCA for the Sum of Two Nonconvex Functions Minimization Problem

In this section, we consider the sum of two nonconvex functions minimization problem (2), i.e.,

$$\min_{x \in \mathbb{R}^n} f(x) + r(x),$$

where f is a differentiable function with L-Lipschitz continuous gradient (possibly nonconvex) and r is a DC function. Clearly, f can be expressed as a DC function: $f(x) = \frac{\rho}{2}\|x\|^2 - \left[\frac{\rho}{2}\|x\|^2 - f(x)\right]$, where $\rho \geq L$. Let $r(x) := g(x) - h(x)$, then a DC decomposition of $F(x) = f(x) + r(x)$ is given by

$$F(x) = G(x) - H(x), \tag{19}$$

where $G(x) = \frac{\rho}{2}\|x\|^2 + g(x)$ and $H(x) = \frac{\rho}{2}\|x\|^2 - f(x) + h(x)$. According to Algorithm 1, *ADCA* for solving the problem (2) is described in Algorithm 2. We note that $\mu(G) > \rho > 0$ and the convergence results of Algorithm 2 are guaranteed by Theorem 1 and 2. Furthermore, with most of existing nonconvex regularizers r in the literature, the resolution of the subproblem (20) is inexpensive or can be explicitly computed.

Algorithm 2 ADCA for solving (2)

```
Initialization: Choose an initial point $x^0$, $z^0 = x^0$, $q \in \mathbb{N}$, $t_0 = 1$, $\rho > L$, and $k \leftarrow 0$.
repeat
    Compute $t_{k+1} = \frac{1+\sqrt{1+4t_k^2}}{2}$.
    Compute $z^k = x^k + \frac{t_k - 1}{t_{k+1}}\left(x^k - x^{k-1}\right)$ if $k \geq 1$.
    If $F(z^k) \leq \max_{t=\max(0,k-q),\dots,k} F(x^t)$ then set $v^k = z^k$, otherwise set $v^k = x^k$.
    Compute $y^k = \rho v^k - \nabla f(v^k) + \xi^k$, where $\xi^k \in \partial h(v^k)$.
    Compute $x^{k+1}$ by solving strongly convex problem
        $$\min_{x \in \mathbb{R}^n} \left\{ \frac{\rho}{2}\|x\|^2 + g(x) - \langle y^k, x \rangle \right\}. \tag{20}$$
    $k \leftarrow k + 1$.
until Stopping criterion.
```

4 Numerical Experiment

To study the performance of the *ADCA*, we consider the sparse binary logistic regression problem. The problem can be described as follows. Let $\{(x_1, y_1), \dots, (x_n, y_n)\}$ be a training set with observation vectors $x_i \in \mathbb{R}^d$ and labels $y_i \in \{-1, 1\}$. We aim to find a hyperplane $f = \langle w, x \rangle + b$ that separates the two classes. To find w and b, we maximize the log-likelihood function $-\frac{1}{n}\sum_{i=1}^n \log(1 + \exp(-y_i(x_i^T w + b)))$. On the other hand, to deal with irrelevant and redundant features in high-dimensional data, we use features selection method which consists in minimizing the zero-norm of w. Hence the sparse binary logistic regression is formulated by

$$\min_{w \in \mathbb{R}^d, b \in \mathbb{R}} \frac{1}{n} \sum_{i=1}^n \log(1 + \exp(-y_i(x_i^T w + b))) + \lambda \|w\|_0, \tag{21}$$

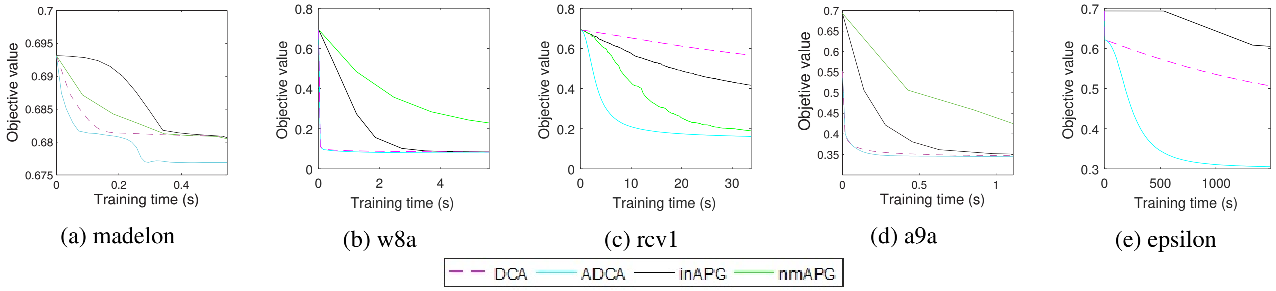
where $\lambda > 0$ is the trade-off parameter between the two terms. The minimization of zero-norm is known to be NP-hard for which several methods have been developed in the literature. The readers are referred to [Le Thi *et al.*, 2015] for an extensive overview of existing methods for the minimization of zero-norm. In this work, we replace $\|w\|_0$ by a nonconvex approximation, namely the exponential concave function defined by $r_{exp}(w) = \sum_{i=1}^d (1 - \exp(-\alpha|w_i|))$. Thus, the problem (21) becomes

$$\min_{(w,b)} \frac{1}{n} \sum_{i=1}^n \log(1 + \exp(-y_i(x_i^T w + b))) + \lambda r_{exp}(w). \tag{22}$$

r_{exp} can be expressed as a DC function $r_{exp}(w) = g_{exp}(w) - h_{exp}(w)$ where $g_{exp}(w) = \alpha\|w\|_1$ and $h_{exp}(w) = \sum_{i=1}^d (\alpha|w_i| - 1 + \exp(-\alpha|w_i|))$. Thus, the problem (22) takes the form of (2) and can be solved by *DCA* and *ADCA*. Note that with this DC decomposition, the corresponding *DCA* and *ADCA* are very inexpensive: it only required the soft thresholding operator (proximal operator of ℓ_1-norm), which is explicitly computed.

The experiments are performed on 4 algorithms: DCA, *ADCA*, inexact APG (*inAGP*) [Yao *et al.*, 2017] and nonmonotone APG (*nmAGP*) [Li and Lin, 2015]. For DCA and *ADCA*, we estimated a Lipschitz constant L by computing a bound of Hessian matrix of logistic loss. Note that *inAPG* and *nmAPG* require to compute the proximal mapping of the DC function r. However, this proximal mapping do not have a closed form. We therefore use DCA to compute the proximal mapping of r in *inAPG* and *nmAPG*. All the algorithms are terminated when the change of two consecutive objective function values is less than 10^{-5}. We also stop algorithms after 5h (18.000 seconds) of CPU time.

The detailed information of used datasets is summarized in the first column of Table 1. n_{train} (resp. n_{test}) represents



(a) madelon (b) w8a (c) rcv1 (d) a9a (e) epsilon

Figure 1: Objective value versus training time (in seconds)

the number of points in training set (resp. test set) while d is the number of features. All data sets can be downloaded from well-known data repertory LibSVM.

Dataset	Method	Time (s)	Acc (%)	Sparsity (%)
madelon	DCA	1.14	**62.17**	**0.4**
n_{train}=2 000	ADCA	**0.54**	**62.17**	**0.4**
n_{test}=600	inAPG	0.86	**62.17**	**0.4**
d=500	nmAPG	1.23	**62.17**	**0.4**
w8a	DCA	32.8	98.43	16
n_{train}=49 749	ADCA	**5.58**	**98.51**	**15.67**
n_{test}=14 951	inAPG	36.42	98.45	17
d=300	nmAPG	54.81	98.4	19.33
rcv1	DCA	113.03	91.8	0.87
n_{train}=20 242	ADCA	**33.74**	**94.23**	0.79
n_{test}=677 399	inAPG	39.65	91.1	0.85
d=47 236	nmAPG	112.37	93.9	**0.72**
a9a	DCA	7.11	84.95	**32.52**
n_{train}=32 561	ADCA	**1.11**	**84.98**	33.33
n_{test}=16 281	inAPG	6.38	84.97	**32.52**
d=123	nmAPG	14.01	84.97	**32.52**
real-sim	DCA	33.25	94.49	2.71
n_{train}=57 847	ADCA	**7.74**	**94.5**	**2.59**
n_{test}=14 462	inAPG	33.49	94.42	2.62
d=20 958	nmAPG	53.11	94.39	2.82
epsilon	DCA	18000	87.53	7.77
n_{train}=400 000	ADCA	1488	**88.22**	**7.75**
n_{test}=100 000	inAPG	18000	73.14	12.6
d=2 000	nmAPG	NA	NA	NA

Table 1: Comparative results. Bold values correspond to the best results for each dataset

All experiments are performed on a PC Intel i7 CPU3770, 3.40 GHz of 8GB RAM and the codes were written in MATLAB. We fix $\alpha = 5$ as proposed in [Bradley and Mangasarian, 1998]. The parameter q is set to 5. The trade-off parameter λ is fixed to 10^{-4} on rcv1, epsilon and 10^{-3} on the other data sets. For the *epsilon*, the nmAPG algorithm did not furnish any result due to a out of memory problem.

We reported in Table 1 the running time in second, the classification accuracy on test set and the sparsity (percentage of selected features) of solution . We also plot the curves of objective function values versus training time in Figure 1.

Concerning the classification accuracy, *ADCA* gives the best results for all 5 datasets (all four algorithms give the same result on *madelon*). As for the sparsity of solution, the four algorithms are comparable. All four algorithms give the same results on *madelon* and the best sparsity of solution on *a9a* is obtained by *DCA*, *inAPG*, *nmAPG*. *nmAPG* suppresses more features than the others on *rcv1* while *ADCA* gives the best sparsity of solution on *w8a,epsilon*. In term of running time, *ADCA* is clearly the fastest one among the four compared algorithms. *ADCA* improve considerably the running time comparing to *DCA*: *ADCA* is up to 12.09 times faster than *DCA* (*gisette*). *ADCA* is faster than *inAPG* and *nmAPG* which also use acceleration technique. We can observe from Figure 1 that the objective functions of *ADCA* decrease drastically in few first iterations comparing to the others three algorithms.

Overall, among the four compared algorithms, *ADCA* is the fastest one while giving better classification accuracy and sparsity of solution.

5 Conclusion

We have rigorously studied *ADCA*, a variant of DCA with the aim of improving its performance. *ADCA* consists in incorporating the Nesterov's acceleration technique into DCA. We have proved that *ADCA* converges to a critical point of the standard DC program. Furthermore, we proved the convergence rate of *ADCA* under the Lojasiewics assumption. *ADCA* is then developed for the minimization of sum of two nonconvex functions problem, a special case of the standard DC program. Exploiting the fact that $f(x)$ is differentiable with L-Lipschitz gradient, we propose, an efficient DC decomposition for which the corresponding *ADCA* scheme is inexpensive. To evaluate the performance of proposed algorithm, we consider the sparse binary logistic regression problem. *ADCA* for solving the latter is very inexpensive: it only required the soft thresholding operator which is explicitly computed. Numerical results showed that *ADCA* improves considerably the running time of DCA (up to 12.09 times faster than DCA) while giving similar or better classification accuracy and sparsity of solution. Furthermore, *ADCA* outperformed related accelerated proximal gradient methods such as non-monotone APG and inexact APG.

References

[Attouch and Bolte, 2009] Hedy Attouch and Jerome Bolte. On the convergence of the proximal algorithm for nonsmooth functions involving analytic features. *Mathematical Programming*, 116(1):5–16, 2009.

[Beck and Teboulle, 2009a] Amir Beck and Marc Teboulle. Fast gradient-based algorithms for constrained total variation image denoising and deblurring problems. *IEEE Transactions on Image Processing*, 18(11):2419–2434, 2009.

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

[Bradley and Mangasarian, 1998] Paul Bradley and Olvi Leon Mangasarian. Feature selection via concave minimization and support vector machines. In *Machine Learning Proceedings of the Fifteenth International Conference (ICML '98)*, pages 82–90, 1998.

[Chambolle *et al.*, 1998] Antonin Chambolle, Ronald DeVore, Nam-Yong Lee, and Bradley Lucier. Nonlinear wavelet image processing: variational problems, compression, and noise removal through wavelet shrinkage. *IEEE Transactions on Image Processing*, 7(3):319–335, 1998.

[Combettes and Wajs, 2005] Patrick Combettes and Valerie Wajs. Signal recovery by proximal forward-backward splitting. *Multiscale Modeling & Simulation*, 4(4):1168–1200, 2005.

[Daubechies *et al.*, 2004] Ingrid Daubechies, Michel Defrise, and Christine De Mol. An iterative thresholding algorithm for linear inverse problems with a sparsity constraint. *Communications on Pure and Applied Mathematics*, 57(11):1413–1457, 2004.

[Dempster *et al.*, 1977] Arthur Dempster, Natalie Laird, and Donald Rubin. Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society, Serie B*, 39(1):1–38, 1977.

[Frankel *et al.*, 2015] Pierre Frankel, Guillaume Garrigos, and Peypouquet Peypouquet. Splitting methods with variable metric for kurdyka–łojasiewicz functions and general convergence rates. *Journal of Optimization Theory and Applications*, 165(3):874–900, 2015.

[Grippo and Sciandrone, 2002] Luigi Grippo and Marco Sciandrone. Nonmonotone globalization techniques for the barzilai-borwein gradient method. *Computational Optimization and Applications*, 23(2):143–169, 2002.

[Gu *et al.*, 2018] Bin Gu, Zhouyuan Huo, and Heng Huang. Inexact proximal gradient methods for non-convexand non-smooth optimization. In *32th AAAI Conference on Artificial Intelligence AAAI-18*, 2018.

[Hale *et al.*, 2008] Elaine Hale, Wotao Yin, and Yin Zhang. Fixed-point continuation for ℓ_1-minimization: Methodology and convergence. *SIAM Journal on Optimization*, 19(3):1107–1130, 2008.

[Le Thi and Pham Dinh, 2005] Hoai An Le Thi and Tao Pham Dinh. The DC (difference of convex functions) programming and DCA revisited with DC models of real world nonconvex optimization problems. *Annals of Operations Research*, 133:23–46, 2005.

[Le Thi and Pham Dinh, 2018] Hoai An Le Thi and Tao Pham Dinh. DC programming and DCA: thirty years of developments. *Mathematical Programming, Special Issue: DC Programming - Theory, Algorithms and Applications*, 169(1):5–64, 2018.

[Le Thi *et al.*, 2015] Hoai An Le Thi, Tao Pham Dinh, Hoai Minh Le, and Xuan Thanh Vo. DC approximation approaches for sparse optimization. *European Journal of Operational Research*, 244(1):26–46, 2015.

[Li and Lin, 2015] Huan Li and Zhouchen Lin. Accelerated proximal gradient methods for nonconvex programming. In *Advances in Neural Information Processing Systems*, pages 377–387, 2015.

[Nesterov, 1983] Yurii Nesterov. A method of solving a convex programming problem with convergence rate $\mathcal{O}(1/k^2)$. *Soviet Mathematics Doklady*, 27:372–376, 1983.

[Pham Dinh and Le Thi, 1997] Tao Pham Dinh and Hoai An Le Thi. Convex analysis approach to D.C. programming: Theory, algorithms and applications. *Acta Mathematica Vietnamica*, 22(1):289–355, 1997.

[Pham Dinh and Le Thi, 2014] T. Pham Dinh and H. A. Le Thi. Recent advances in dc programming and dca. In *Transactions on Computational Intelligence XIII*, pages 1–37, 2014.

[Wright *et al.*, 2009] Stephen Wright, Robert Nowak, and Mario Figueiredo. Sparse reconstruction by separable approximation. *IEEE Transactions on Signal Processing*, 57(7):2479–2493, 2009.

[Yao *et al.*, 2017] Quanming Yao, James. Kwok, Fei Gao, Wei Chen, and Tie-Yan Liu. Efficient inexact proximal gradient algorithm for nonconvex problems. In *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence*, pages 3308–3314, 2017.

[Yuille and Rangarajan, 2003] A. L. Yuille and A. Rangarajan. The concave-convex procedure. *Neural Computation*, 15(4):915–936, 2003.