Pure and Applied Functional Analysis
Volume 8, Number 5, 2023, 1357–1376

Yokohama Publishers
ISSN 2189-3764 ONLINE JOURNAL

EXACT MULTIDIMENSIONAL PENALTY DCA FOR CONSTRAINED NONSMOOTH DC OPTIMIZATION IN BANACH SPACES

MAKSIM V. DOLGOPOLIK

Abstract. We present a new adaptive local search method for solving nonsmooth DC (Difference-of-Convex functions) optimization problems with equality and inequality constraints in Banach spaces based on a combination of the DCA (Difference-of-Convex functions Algorithm) and exact penalty techniques. To improve performance of the method, we propose to use an exact penalty function with multidimensional penalty parameter, that is, with individual penalty parameter for each constraint. The method adaptively updates each of these penalty parameters by taking into account the corresponding constraint violation measure. We provide a convergence analysis of this method, present several simple numerical examples illustrating its performance, and discuss several ways to further improve efficiency of the proposed method.

1. Introduction

The approach to both smooth and nonsmooth optimization problems based on the use of DC (Difference-of-Convex functions) decompositions of objective function and constrains is a very popular methodology in numerical optimization. Over the last 40 years, a vast array of optimization methods based on the DC methodology have been developed, with the most well-known and, perhaps, the most popular one of them being the so-called *DCA* (Difference-of-Convex functions Algorithm). This method was first proposed in [35] and further improved and analysed in various particular cases in [13, 23, 24, 26, 31–33] . For a recent survey on the DCA and its applications see [25, 27].

Numerical optimization methods closely related to the DCA were studied in [7, 46, 47], while some modifications of the DCA aimed at improving its performance and accelerating convergence were proposed in the recent papers [1, 8, 50]. In the nonsmooth case, many alternatives to the DCA for minimizing DC functions have been proposed in the last 10 yeas, such as, codifferential methods [3, 45], bundle and double bundle methods [19–21, 30, 48], aggregate subgradient method [2], etc.

All of the aforementioned papers deal with either unconstrained DC optimization problems or inequality constrained DC problems. Only in the recent papers by Strekalovsky [39, 40] and in the preprint [15] numerical methods for equality

2020 *Mathematics Subject Classification.* 65K05, 90C48, 90C26, 90C30.
Key words and phrases. DC optimization, nonsmooth optimization, constrained optimization, DCA, exact penalty function.

constrained DC optimization problems have been studied. Moreover, in all papers cited above only the finite dimensional case have been considered, despite the fact that the DC methodology can be very efficiently applied in the infinite dimensional case, e.g. to optimal control problems, as was shown in the papers by Strekalovsky et al. [**?**, 37, 38, 41, 43, 44] .

To the best of the author's knowledge, optimization methods for general equality and inequality constrained DC optimization problems in Banach spaces have not been studied in the past, although such methods can be applied to many classes of variational and optimal control problems. The main goal of this paper is to fill in the gap and analyse a general local search method for such problems based on a combination of the DCA and the exact penalty technique.

Instead of utilising steering exact penalty methodology as in [15, 39, 40], here we propose to use an exact penalty function with *multidimensional* penalty parameter, that is, a penalty function with individual penalty parameter for each constraint. The use of multidimensional penalty parameter allows one to take into account violation of each individual constraint and flexibly adjust corresponding penalty parameters to ensure balanced progress towards feasibility with respect to each of the constrains. If a certain constraint is satisfied, then the corresponding penalty parameter is not updated. In turn, if another constraint is violated, then the corresponding penalty parameter is increased proportionally to the corresponding constraint violation measure.

An efficient way for updating multidimensional penalty parameter based on a global primal-dual penalty method was proposed by Burachik, Kaya, and Price in the recent paper [4]. The penalty updating strategy from [4] was further analysed in [14]. In this paper we employ it in the context of DC optimization to develop an exact penalty DCA with multidimensional penalty updates for finding locally optimal solution of constrained nonsmooth DC optimization problems in Banach spaces. We present a detailed description of this method and prove its convergence to critical points of DC optimization problems in separable Banach spaces. We also give several illustrative numerical examples that underline potential benefits and drawbacks of the proposed method and indicate several possible ways to improve the efficiency of the method.

The paper is organized as follows. Optimality conditions for nonsmooth DC optimization problems with equality and inequality constraints are discussed in Section 2. A version of the exact penalty DCA utilizing exact penalty functions with multidimensional penalty parameter and primal-dual penalty updates is studied in Sections 3 and 4. Finally, some simple numerical examples illustrating performance of this exact penalty DCA are given in Section 5.

2. Optimality conditions and criticality

Let X be a real Banach space. Consider the following constrained nonsmooth DC optimization problem:

$$\begin{aligned} &\text{minimize} \quad f_0(x) = g_0(x) - h_0(x) \\ &\text{subject to} \quad f_i(x) = g_i(x) - h_i(x) \le 0, \quad i \in \mathcal{I}, \\ &\qquad\qquad\quad f_j(x) = g_j(x) - h_j(x) = 0, \quad j \in \mathcal{E}, \quad x \in A. \end{aligned} \tag{$\mathcal{P}$}$$

We suppose that $g_k, h_k \colon X \to \mathbb{R}$, $k \in \{0\} \cup \mathcal{I} \cup \mathcal{E}$, are lower semicontinuous (l.s.c.) convex functions, $\mathcal{I} = \{1, \ldots, \ell\}$ and $\mathcal{E} = \{\ell + 1, \ldots, m\}$ are finite index sets (one of which can be empty), and $A \subseteq X$ is a closed convex set. Note that since X is a Banach space, the functions g_k and h_k are actually continuous on X by [16, Crlr. I.2.5]. Thoughout this article, we also suppose that the following assumption on the problem $(\mathcal{P})$ holds true.

Assumption 2.1. The feasible set Ω of the problem $(\mathcal{P})$ is not empty, and the objective function f_0 is bounded below on Ω.

Our aim is to present a method for finding locally optimal solutions of the problem $(\mathcal{P})$. Let us start by deriving local optimality condition for this problem. We will derive optimality conditions that are similar to optimality conditions for nonsmooth mathematical programming problems in terms of Demyanov-Rubinov-Polyakova quasidifferentials [11, 12]. To this end, introduce the function

$$\varphi(x) = \sum_{i \in \mathcal{I}} \max\{f_i(x), 0\} + \sum_{j \in \mathcal{E}} |f_j(x)|$$

that measures the violation of the equality and inequality constraints of the problem $(\mathcal{P})$. Below we use this function to express an abstract constraint qualification for the problem $(\mathcal{P})$. Sufficient conditions for the validity of this constraint qualification in terms of subdifferentials of the functions g_k and h_k in the case when $A = X$ can be found in [11] (see also [12]).

Let X^* be the topological dual space of X and $\langle \cdot, \cdot \rangle$ be the corresponding duality pairing. Denote by $N_A(x) = \{v \in X^* \mid \langle v, y - x \rangle \le 0 \; \forall y \in A\}$ the normal cone to the set A at a point $x \in A$, and introduce the index set $\mathcal{I}(x) = \{i \in \mathcal{I} \mid f_i(x) = 0\}$. Let $\operatorname{dist}(x, \Omega) = \inf_{y \in \Omega} \|x - y\|$ for any $x \in X$, where, as above, Ω is the feasible region of the problem $(\mathcal{P})$.

The following theorem contains local optimality conditions for the problem $(\mathcal{P})$, which can be viewed as a generalisation of [11, Thm. 3].

Theorem 2.2. *Let x_* be a locally optimal solution of the problem $(\mathcal{P})$. Suppose that φ admits a local error bound near x_* with respect to the set A, i.e. there exist $\varkappa > 0$ and a neighbourhood U of x_* such that for all $x \in U \cap A$ one has $\varphi(x) \ge \varkappa \operatorname{dist}(x, \Omega)$. Then there exists $c_* > 0$ such that for all $v_k \in \partial h_k(x_*)$, $k \in \{0\} \cup \mathcal{I} \cup \mathcal{E}$, and $w_j \in \partial g_j(x_*)$, $j \in \mathcal{E}$, one can find multipliers $\lambda_i, \underline{\mu}_j, \overline{\mu}_j > 0$,*

$i \in \mathcal{I}$, $j \in \mathcal{E}$, such that

$$(2.1)\quad \begin{aligned} 0 \in \partial g_0(x_*) - v_0 + \sum_{i \in \mathcal{I}} \lambda_i \big(\partial g_i(x_*) - v_i \big) + \sum_{j \in \mathcal{E}} \underline{\mu}_j \big(\partial g_j(x_*) - v_j \big) \\ - \sum_{j \in \mathcal{E}} \overline{\mu}_j \big(w_j - \partial h_j(x_*) \big) + N_A(x_*), \end{aligned}$$

and $\lambda_i f_i(x_*) = 0$, $\max\{\lambda_i, \underline{\mu}_j + \overline{\mu}_j\} \le c_*$ for all $i \in \mathcal{I}$, $j \in \mathcal{E}$.

Proof. By [51, Crlr. 2.2.13] the objective function f_0 is locally Lipschitz continuous as the difference of finite l.s.c. convex functions. Therefore by [10, Thm. 2.4 and Prp. 2.7] under the assumptions of the theorem there exists $c_* > 0$ such that x_* is a local minimizer of the penalty function $\Phi_{c_*} = f_0 + c_* \varphi$ on the set A.

Applying the calculus rules for directional derivatives [9, Sect. I.3] and the fact that any l.s.c. convex function is directionally differentiable (see, e.g. [18, Prp. 4.1.3]) one obtains that the penalty function Φ_{c_*} is directionally differentiable and its directional derivative at x_* has the form

$$\begin{aligned} \Phi'_{c_*}(x_*, \cdot) = f'_0(x_*, \cdot) + c_* \sum_{i \in \mathcal{I}(x_*)} \max\{f'_i(x_*, \cdot), 0\} \\ + c_* \sum_{j \in \mathcal{E}} \max\{f'_j(x_*, \cdot), -f'_j(x_*, \cdot)\}. \end{aligned}$$

By the standard necessary optimality conditions in terms of directional derivatives [9, Lemma V.1.2] one has $\Phi'_{c_*}(x_*, z - x_*) \ge 0$ for all $z \in A$.

Fix any $v_k \in \partial h_k(x_*)$, $k \in \{0\} \cup \mathcal{I} \cup \mathcal{E}$, and $w_j \in \partial g_j(x_*)$, $j \in \mathcal{E}$, and introduce the convex functions

$$\begin{aligned} \eta(z) &= p_0(z) + c_* \sum_{i \in \mathcal{I}(x_*)} \max\{0, p_i(z)\} + c_* \sum_{j \in \mathcal{E}} \max\{p_j(z), q_j(z)\}, \\ p_k(z) &= \max_{v \in \partial g_k(x_*) - v_k} \langle v, z \rangle, \quad q_j(z) = \max_{w \in \partial h_j(x_*) - w_j} \langle w, z \rangle, \quad i \in \mathcal{I}, \quad j \in \mathcal{E}. \end{aligned}$$

Taking into account the fact that the directional derivative of a convex function is the support function of its subdifferential [51, Thm. 2.4.9] one obtains that $\eta(\cdot) \ge \Phi'_{c_*}(x_*, \cdot)$. Therefore 0 is a point of global minimum of the convex function η on the set $A - x_*$, which implies that $0 \in \partial \eta(0) + N_A(x_*)$ (see, e.g. [18, Thm. 1.1.2']). Hence with the use of the subdifferential calculus one gets that

$$\begin{aligned} 0 \in \partial g_0(x_*) - v_0 + c_* \sum_{i \in \mathcal{I}(x_*)} \operatorname{co} \big\{ \partial g_i(x_*) - v_i, 0 \big\} \\ + c_* \sum_{j \in \mathcal{E}} \operatorname{co} \big\{ \partial g_j(x_*) - v_j, w_j - \partial h_j(x_*) \big\} + N_A(x_*). \end{aligned}$$

Consequently, by the definition of convex hull for any $i \in \mathcal{I}(x_*)$ there exists $\alpha_i \in [0,1]$ and for any $j \in \mathcal{E}$ there exists $\beta_j \in [0,1]$ such that

$$0 \in \partial g_0(x_*) - v_0 + c_* \sum_{i \in \mathcal{I}(x_*)} \alpha_i \big(\partial g_i(x_*) - v_i\big) \\ + c_* \sum_{j \in \mathcal{E}} \big(\beta_j(\partial g_j(x_*) - v_j) + (1-\beta_j)(w_j - \partial h_j(x_*))\big) + N_A(x_*).$$

Hence putting $\lambda_i = c_* \alpha_*$ for $i \in \mathcal{I}(x_*)$, $\lambda_i = 0$ for $i \in \mathcal{I} \setminus \mathcal{I}(x_*)$, $\underline{\mu}_j = c_* \beta_j$ and $\overline{\mu}_j = c_*(1-\beta_j)$ for all $j \in \mathcal{E}$ one obtains the required result. □

Remark 2.3. Multipliers λ_i, $\underline{\mu}_j$, and $\overline{\mu}_j$ obviously depend on the choice of subgradients v_k and w_j of the functions h_k and g_j, respectively, and *cannot* be chosen independently of those subgradients (cf. [28, 29]). Moreover, note that in optimality conditions (2.1) there are *two* multipliers $\underline{\mu}_j$ and $\overline{\mu}_j$ corresponding to each equality constraint $f_j(x) = 0$. Both these facts are specific features of optimality conditions for nonsmooth mathematical programming problems in terms of quasidifferentials and corresponding optimality conditions for nonsmooth DC optimization problems (see the discussion in [11, 12]). Let us also point out that the inequality $\max_{i \in \mathcal{I}, j \in \mathcal{E}} \{\lambda_i, \underline{\mu}_j + \overline{\mu}_j\} \le c_*$ simply means that the multipliers are uniformly bounded for all subgradients v_k and w_j of the functions h_k and g_j.

Although optimality conditions from Thm. 2.2 are very sharp and capable of discerning nonoptimality of points at which many other optimality conditions are satisfied (see examples in [11, 12]), these conditions are unsuitable for practical applications and analysis of numerical methods. In particular, to apply these conditions one needs to compute the entire subdifferentials of the functions g_k and h_k, $k \in \{0\} \cup \mathcal{I} \cup \mathcal{E}$, which is often either impossible or too computationally expensive. Therefore, similarly to the case of optimality conditions for unconstrained nonsmooth DC optimization problems (see the discussions in [19, 25, 46]), we introduce the following notion of *criticality* for the problem $(\mathcal{P})$, which is much more convenient for applications than optimality conditions from Thm. 2.2 and can be viewed as a significantly weakened form of these optimality conditions.

Definition 2.4. A point x_* is said to be critical for the problem $(\mathcal{P})$, if there exist $v_k \in \partial h_k(x_*)$, $k \in \{0\} \cup \mathcal{I} \cup \mathcal{E}$, and $w_j \in \partial g_j(x_*)$, $j \in \mathcal{E}$ such that inclusion (2.1) holds true for some $\lambda_i, \underline{\mu}_j, \overline{\mu}_j \ge 0$, $i \in \mathcal{I}$, $j \in \mathcal{E}$, satisfying the complementarity condition $\lambda_i f_i(x_*) = 0$ for all $i \in \mathcal{I}$.

Let us point out an almost obvious, yet useful reformulation of the notion of criticality.

Lemma 2.5. *A feasible point x_* is critical for the problem $(\mathcal{P})$ if and only if there exist $c > 0$, $v_k \in \partial h_k(x_*)$, $k \in \{0\} \cup \mathcal{I} \cup \mathcal{E}$, and $w_j \in \partial g_j(x_*)$, $j \in \mathcal{E}$, such that x_**

is a point of global minimum of the convex function

$$Q_c(x) = g_0(x) - \langle v_0, x - x_* \rangle + c \sum_{i \in \mathcal{I}} \max\big\{ g_i(x) - h_i(x_*) - \langle v_i, x - x_* \rangle, 0 \big\} \\ + c \sum_{j \in \mathcal{E}} \max\big\{ g_j(x) - h_j(x_*) - \langle v_j, x - x_* \rangle, h_j(x) - g_j(x_*) - \langle w_j, x - x_* \rangle \big\} \tag{2.2}$$

on the set A.

Proof. By the standard optimality conditions, x_* is a point of global minimum of the convex function Q_c on the set A iff $0 \in \partial Q_c(x_*) + N_A(x_*)$. In turn, by the standard rules of the subdifferential calculus this inclusion is satisfied for some $c > 0$ iff inclusion (2.1) holds true for some $\lambda_i, \underline{\mu}_j, \overline{\mu}_j > 0$, $i \in \mathcal{I}$, $j \in \mathcal{E}$, satisfying the complementarity condition $\lambda_i f_i(x_*) = 0$ for all $i \in \mathcal{I}$. □

Remark 2.6. Note that since the function Q_c is nondecreasing in c and $Q_c(x_*) = g_0(x_*)$, the point x_* is a global minimizer of Q_c on A if and only if x_* is a global minimizer of Q_t on A for any $t \geq c$.

3. Exact penalty DCA with multidimensional penalty parameter

We propose to find locally optimal solutions (more precisely, critical points) of the problem $(\mathcal{P})$ with the use of a modification of the famous DCA using exact penalty functions. Exact penalty DCAs with the simplest penalty updates (e.g., the penalty parameter is increased by a given factor after every iteration till a feasible point is found) were studied for inequality constrained problem in [23, 33], and for general cone constrained DC optimization problems in [13, 27]. However, as is well known, carefully chosen penalty updates might significantly improve performance of exact penalty methods (see, e.g. the discussion and examples in [5, 6]). Therefore, our main goal is to present and analyse an exact penalty DCA, in which the penalty parameter is updated adaptively.

Being inspired by the recent papers [4, 14], we study an exact penalty DCA based on an exact penalty function with a *multidimensional* penalty parameter, that is, a penalty function with individual penalty parameter for each constraint. This method utilises a natural adaptive penalty updating strategy, which can be derived from a primal-dual approach to exact penalty functions (see [4]). According to this strategy, an increase of the penalty parameter corresponding to a given constraint must be proportional to the violation of this constraint, so that penalty parameters corresponding to constraints that are "almost" satisfied are changed only slightly, while penalty parameters corresponding to constraints with large violation measure are increased by a substantial amount. An exact penalty method with such penalty updates tracks the violation of each constraint after every iteration and adaptively updates penalty parameters in accordance with this information.

Introduce the penalty function

$$\Psi_\tau(x) = f_0(x) + \sum_{i \in \mathcal{I}} \tau^{(i)} \max\{f_i(x), 0\} + \sum_{j \in \mathcal{E}} \tau^{(j)} |f_j(x)|$$

with multidimensional penalty parameter $\tau = (\tau^{(1)}, \ldots, \tau^{(m)}) \in \mathbb{R}^m_{++}$, where by definition $\mathbb{R}_{++} = (0, +\infty)$. Choose any $y \in X$, and any subgradients $v_k \in \partial h_k(y)$, $k \in$

$\{0\} \cup \mathcal{I} \cup \mathcal{E}$, and $w_j \in \partial g_j(y)$, $j \in \mathcal{E}$. Introduce the vector $V = (v_0, v_1, \ldots, v_m, w_{\ell+1}, \ldots, w_m)$ and define

$$M_\tau(x, y, V) = g_0(x) - \langle v_0, x - y \rangle + \sum_{i \in \mathcal{I}} \tau^{(i)} \max \big\{ g_i(x) - h_i(y) - \langle v_i, x - y \rangle, 0 \big\} + \sum_{j \in \mathcal{E}} \tau^{(j)} \max \big\{ g_j(x) - h_j(y) - \langle v_j, x - y \rangle, h_j(x) - g_j(y) - \langle w_j, x - y \rangle \big\}. \tag{3.1}$$

The function $M_\tau(\cdot, y, V)$ is obviously convex. Furthermore, applying the definition of subgradient one can easily check that for any $x \in X$ one has

$$M_\tau(x, y, V) - h_0(y) \geq \Psi_\tau(x), \quad M_\tau(y, y, V) - h_0(y) = \Psi_\tau(y), \tag{3.2}$$

that is, $M_\tau(\cdot, y, V) - h_0(y)$ is a global convex majorant of the penalty function $\Psi_\tau(\cdot)$. Following the DCA methodology, we will use the function $M_\tau(\cdot, x_n, V)$ to define the next iterate x_{n+1}, given the current point x_n. Namely, the point x_{n+1} is defined as an ε_n-optimal solution of the problem

$$\underset{x}{\text{minimize}} \;\; M_{\tau_n}(x, x_n, V_n) \quad \text{subject to} \quad x \in A$$

for some small $\varepsilon_n > 0$, that is, $x_{n+1} \in A$ is any point satisfying the inequality

$$M_{\tau_n}(x_{n+1}, x_n, V_n) \leq \inf_{x \in A} M_{\tau_n}(x, x_n, V_n) + \varepsilon_n.$$

Note that such point x_{n+1} always exists, provided the penalty function Ψ_{τ_n} is bounded below on the set A (see (3.2)).

A theoretical scheme of the exact multidimensional penalty DCA is given in Algorithmic Pattern 1 below.

Algorithmic Pattern 1: Exact Multidimensional Penalty DCA

Initialization. Choose an initial point $x_0 \in A$, an initial value of the penalty parameter $\tau_0 \in \mathbb{R}^m_{++}$, the maximal value of penalty parameters $\tau_{\max} > 0$, a sequence $\{\varepsilon_n\} \subset \mathbb{R}_{++}$, and set $n := 0$.

Main Step. For all $k \in \{0\} \cup \mathcal{I} \cup \mathcal{E}$ compute $v_{nk} \in \partial h_k(x_n)$, for all $j \in \mathcal{E}$ compute $w_{nj} \in \partial g_j(x_n)$, and define

$$V_n = (v_{n0}, v_{n1}, \ldots, v_{nm}, w_{n(\ell+1)}, \ldots, w_{nm}).$$

Set the value of x_{n+1} to an ε_n-optimal solution of the convex problem

$$\underset{x}{\text{minimize}} \;\; M_{\tau_n}(x, x_n, V_n) \quad \text{subject to} \quad x \in A \tag{3.3}$$

such that $M_{\tau_n}(x_{n+1}, x_n, V_n) \leq M_{\tau_n}(x_n, x_n, V_n)$.

Penalty Update. Choose scaling coefficient $\gamma_n > 0$ and define

$$\tau^{(k)}_{n+1} = \begin{cases} \tau^{(k)}_n + \gamma_n \max\{f_k(x_{n+1}), 0\}, & \text{if } k \in \mathcal{I} \text{ and } \tau^{(k)}_n < \tau_{\max}, \\ \tau^{(k)}_n + \gamma_n |f_k(x_{n+1})|, & \text{if } k \in \mathcal{E} \text{ and } \tau^{(k)}_n < \tau_{\max}, \\ \tau_{\max}, & \text{otherwise.} \end{cases}$$

Check a **stopping criterion**. If it is satisfied, **Stop**. Otherwise, put $n \leftarrow n + 1$ and repeat the **Main Step**.

Remark 3.1. (i) We use the term *algorithmic pattern*, since the method presented in this article is not an algorithm per se, but rather a pattern, whose practical implementation requires the use of convex optimization methods. This pattern defines a whole family of local search method for the problem $(\mathcal{P})$ depending on a method for solving convex optimization subproblems, stopping criteria, and rules for choosing scaling coefficients γ_n.
(ii) Let us note that one can use the following inequalities

$$\big|\Psi_{\tau_n}(x_{n+1}) - \Psi_{\tau_n}(x_n)\big| < \varepsilon_f \quad \Big(\text{and/or } \|x_{n+1} - x_n\| < \varepsilon_x\Big), \quad \varphi(x_{n+1}) < \varepsilon_\varphi$$

with some prespecified $\varepsilon_f > 0$ (and/or $\varepsilon_x > 0$) and $\varepsilon_\varphi > 0$ as a stopping criterion for Algorithmic Pattern 1. Theorem 4.5 below provides a justification for this stopping criterion. It is worth mentioning that the first inequality above can be replaced with

$$\big|M_{\tau_n}(x_{n+1}, x_n, V_n) - M_{\tau_n}(x_{n+1}, x_n, V_n)\big| < \varepsilon_f$$

in the case when the computation of $\Psi_{\tau_n}(x_{n+1})$ and $\Psi_{\tau_n}(x_n)$ is time consuming (see Corollary 4.7).
(iii) Let us also note that the scaling coefficients γ_n in Algorithmic Pattern 1 can be chosen as $\gamma_n \equiv \varkappa$ for some $\varkappa > 0$ or $\gamma_n = \varkappa / \|\psi(x_{n+1})\|$, provided $\psi(x_{n+1}) \neq 0$, where $\|\cdot\|$ is some norm in $\mathbb{R}^m$, and

$$\psi(x) = \Big(\max\{f_1(x), 0\}, \ldots, \max\{f_\ell(x), 0\}, |f_{\ell+1}(x)|, \ldots, |f_m(x)|\Big)^T.$$

Apparently, the choice $\gamma_n = \varkappa / \|\psi(x_{n+1})\|$ or

$$\gamma_n = \begin{cases} \varkappa / \|\psi(x_{n+1})\|, & \text{if } \|\psi(x_{n+1})\| \geq \delta, \\ \varkappa, & \text{otherwise} \end{cases} \tag{3.4}$$

for some small $\delta > 0$ is more reasonable, since it allows one to avoid an excessive increase of penalty parameters $\tau_n^{(k)}$ during first few iterations of the method, when the violation of constraints can be sufficiently large. Furthermore, the combined strategy (3.4) might also help to avoid an excessive increase of the penalty parameter at later stages, when the constraint violation is sufficiently small and the iterates gradually approach a critical point.

Observe that by introducing additional variables $\gamma^{(i)}$ and $\xi^{(j)}$ one can rewrite the convex penalty subproblem on the main step of Algorithmic Pattern 1 as the following equivalent convex programming problem:

$$\begin{aligned} &\underset{x \in X, \gamma \in \mathbb{R}^\ell, \xi \in \mathbb{R}^{m-\ell}}{\text{minimize}} \quad g_0(x) - \langle v_{n0}, x \rangle + \sum_{i \in \mathcal{I}} \tau_n^{(i)} \gamma^{(i)} + \sum_{j \in \mathcal{E}} \tau_n^{(j)} \xi^{(j)}, \\ &\text{subject to } \; g_i(x) - h_i(x_n) - \langle v_{ni}, x - x_n \rangle \leq \gamma^{(i)}, \quad \gamma^{(i)} \geq 0, \quad i \in \mathcal{I}, \\ &g_j(x) - h_j(x_n) - \langle v_{nj}, x - x_n \rangle \leq \xi^{(j)}, \quad j \in \mathcal{E}, \\ &h_j(x) - g_j(x_n) - \langle w_{nj}, x - x_n \rangle \leq \xi^{(j)}, \quad j \in \mathcal{E}, \qquad x \in A. \end{aligned} \tag{3.5}$$

In the case when the functions g_k, $k \in \{0\} \cup \mathcal{I} \cup \mathcal{E}$, and h_j, $j \in \mathcal{E}$, are smooth, one can solve this subproblem with the use of interior point methods, thus avoiding the minimization of the *nonsmooth* function $M_{\tau_n}(\cdot, x_n, V_n)$. Alternatively, one can use a smoothing approximation of the penalty function $M_{\tau_n}(\cdot, x_n, V_n)$ preserving its convexity and approximately solve the corresponding penalty subproblem by minimizing this smoothing approximation (cf. [4]). It is possible to extend the convergence analysis presented below to such exact penalty DCA based on a smoothing approximation. For the sake of shortness, we do not present such extension here and leave it as an open problem for future research.

Let us briefly explain why problems (3.3) and (3.5) are interchangable.

Proposition 3.2. *Problems* (3.3) *and* (3.5) *are equivalent in the following sense:*

(1) *they have the same optimal value;*
(2) *if* $(x_{n+1}, \gamma_{n+1}, \xi_{n+1})$ *is an* ε_n*-optimal solution of problem* (3.5)*, then* x_{n+1} *is an* ε_n*-optimal solution of problem* (3.3)*;*
(3) *if* x_{n+1} *is an* ε_n*-optimal solution of problem* (3.3)*, then the triplet* $(x_{n+1}, \gamma(x_{n+1}), \xi(x_{n+1}))$ *is an* ε_n*-optimal solution of problem* (3.5)*.*

Here

$$\gamma^{(i)}(x) = \max\big\{g_i(x) - h_i(x_n) - \langle v_i, x - x_n\rangle, 0\big\},$$
$$\xi^{(j)}(x) = \max\big\{g_j(x) - h_j(x_n) - \langle v_j, x - x_n\rangle, h_j(x) - g_j(x_n) - \langle w_j, x - x_n\rangle\big\}$$

for any $x \in X$ *and all* $i \in \mathcal{I}$ *and* $j \in \mathcal{E}$*.*

Proof. Denote the objective function of problem (3.5) by $T_n(x, \gamma, \xi)$. As is easily seen, for any feasible point (x, γ, ξ) of this problem the point $(x, \gamma(x), \xi(x))$ is also feasible and $\gamma^{(i)} \geq \gamma^{(i)}(x)$, $i \in \mathcal{I}$, $\xi^{(j)} \geq \xi^{(j)}(x)$, $j \in \mathcal{E}$. Therefore, for any feasible point (x, γ, ξ) of problem (3.5) one has

$$T_n(x, \gamma, \xi) \geq T_n(x, \gamma(x), \xi(x)) = M_n(x, x_n, V_n)$$

and the first inequality is strict if either $\gamma \neq \gamma(x)$ or $\xi \neq \xi(x)$, since $\tau_n \in \mathbb{R}^m_{++}$. From the relations above it obviously follows that problems (3.3) and (3.5) have the same optimal value.

If x_{n+1} is an ε_n-optimal solution of problem (3.3), then for any feasible point (x, γ, ξ) of problem (3.5) one has

$$\begin{aligned} T_n(x, \gamma, \xi) \geq M_n(x, x_n, V_n) &\geq M_n(x_{n+1}, x_n, V_n) - \varepsilon_n \\ &= T_n(x_{n+1}, \gamma(x_{n+1}), \xi(x_{n+1})) - \varepsilon_n, \end{aligned}$$

that is, $(x_{n+1}, \gamma(x_{n+1}), \xi(x_{n+1}))$ is an ε_n-optimal solution of problem (3.5).

Conversely, if $(x_{n+1}, \gamma_{n+1}, \xi_{n+1})$ is an ε_n-optimal solution of problem (3.5), then for any $x \in A$ one has

$$\begin{aligned} M_n(x, x_n, V_n) &= T_n(x, \gamma(x), \xi(x)) \geq T_n(x_{n+1}, \gamma_{n+1}, \xi_{n+1}) - \varepsilon_n \\ &\geq T_n(x_{n+1}, \gamma(x_{n+1}), \xi(x_{n+1})) - \varepsilon_n = M_n(x_{n+1}, x_n, V_n) - \varepsilon_n, \end{aligned}$$

that is x_{n+1} is an ε_n-optimal solution of problem (3.3). □

4. Convergence Analysis

Let us now turn to the convergence analysis of Algorithmic Pattern 1. To this end, we need to introduce an auxiliary definition, which extends the notion of criticality from Definition 2.4 to the case of infeasible points.

Definition 4.1. A point $x_* \in A$ is called *a generalised critical point* for a given value $\tau \in \mathbb{R}^m_{++}$ of the penalty parameter, if there exist $v_k \in \partial h_k(x_*)$, $k \in \{0\} \cup \mathcal{I} \cup \mathcal{E}$, and $w_j \in \partial g_j(x_*)$, $j \in \mathcal{E}$, such that x_* is a globally optimal solution of the problem

$$\underset{x}{\text{minimize}}\, M_\tau(x, x_*, V) \quad \text{subject to} \quad x \in A.$$

where $V = (v_0, v_1, \ldots, v_m, w_{\ell+1}, \ldots, w_m)$.

Let us point out two useful properties of generalised critical points. In particular, let us prove that for feasible points the notions of criticality and generalised criticality, in essence, coincide.

Lemma 4.2. *Let x_* be a feasible point of the problem $(\mathcal{P})$. If x_* is a generalised critical point for some $\tau \in \mathbb{R}^m_{++}$, then this point is critical for the problem $(\mathcal{P})$. Conversely, if the point x_* is critical for the problem $(\mathcal{P})$, then there exists $\tau_* \in \mathbb{R}^m_{++}$ such that x_* is a generalised critical point for any $\tau \geq \tau_*$, where the inequality is understood coordinate-wise.*

Proof. Suppose that x_* is a generalised critical point for some $\tau \in \mathbb{R}^m_{++}$. Then by definition there exist $v_k \in \partial h_k(x_*)$, $k \in \{0\} \cup \mathcal{I} \cup \mathcal{E}$, and $w_j \in \partial g_j(x_*)$, $j \in \mathcal{E}$, such that $M_\tau(x, x_*, V) \geq M_\tau(x_*, x_*, V)$ for all $x \in A$. Define $c = \max_k \tau^{(k)}$. Then by definitions $Q_c(x) \geq M_\tau(x, x_* V)$ for all $x \in X$ (see (2.2)). Moreover, the equalities $Q_c(x_*) = M_\tau(x_*, x_*, V) = g_0(x_*)$ hold true by virtue of the fact that the point x_* is feasible. Therefore, $Q_c(x) \geq Q_c(x_*)$ for all $x \in A$, which by Lemma 2.5 implies that x_* is a critical point for the problem $(\mathcal{P})$.

Conversely, if x_* is a critical point for the problem $(\mathcal{P})$, then by Lemma 2.5 there exist $c > 0$, $v_k \in \partial h_k(x_*)$, $k \in \{0\} \cup \mathcal{I} \cup \mathcal{E}$, and $w_j \in \partial g_j(x_*)$, $j \in \mathcal{E}$, such that $Q_c(x) \geq Q_c(x_*)$ for all $x \in A$. Hence, as is easily seen, for any $\tau \in \mathbb{R}^m_{++}$ such that $\tau^{(k)} \geq c$ for all $k \in \mathcal{I} \cup \mathcal{E}$ one has $M_\tau(x, x_*, V) \geq M_\tau(x_*, x_*, V)$, which implies the desired result. □

Remark 4.3. It should be noted that the generalised criticality depends on the choice of penalty parameter τ. In some cases one can escape a generalised critical point by simply changing this parameter (e.g. by increasing those $\tau^{(k)}$, which correspond to infeasible constraints). See [13, Remark 9] for a more detailed discussion.

Lemma 4.4. *Let $\{x_n\}$ be the sequence generated by Algorithmic Pattern 1. Then $\Psi_{\tau_n}(x_{n+1}) \leq \Psi_{\tau_n}(x_n)$ for all $n \in \mathbb{N}$. Furthermore, if for some $n \in \mathbb{N}$ one has $\varepsilon_n = 0$ and the point x_n is not a generalised critical point for τ_n, then $\Psi_{\tau_n}(x_{n+1}) < \Psi_{\tau_n}(x_n)$.*

Proof. By the definition of x_{n+1} (see Algorithmic Pattern 1) one has

$$M_{\tau_n}(x_{n+1}, x_n, V_n) \leq M_{\tau_n}(x_n, x_n, V_n) \quad \forall n \in \mathbb{N}, \tag{4.1}$$

which with the use of inequalities (3.2) implies that $\Psi_{\tau_n}(x_{n+1}) \leq \Psi_{\tau_n}(x_n)$ for all $n \in \mathbb{N}$.

If for some $n \in \mathbb{N}$ one has $\varepsilon_n = 0$ and the point x_n is not a generalised critical point for τ_n, then by definition x_n is not a point of global minimum of the function $M_{\tau_n}(\cdot, x_n, V_n)$ on the set A, that is, inequality (4.1) is strict. Therefore, $\Psi_{\tau_n}(x_{n+1}) < \Psi_{\tau_n}(x_n)$. □

Now we are ready to prove a theorem on convergence of Algorithmic Pattern 1.

Theorem 4.5. *Let $\{x_n\}$ be the sequence generated by Algorithmic Pattern 1. Suppose that the sequence $\{\psi(x_n)\}$ is bounded, the penalty function Ψ_{τ_n} is bounded below on A for some $n \in \mathbb{N}$, and $\varepsilon_n \to 0$ as $n \to \infty$. Then the following statements hold true:*

(1) *the corresponding sequence $\{\tau_n\}$ of penalty parameters is bounded and converges to some τ_*;*
(2) $|\Psi_{\tau_n}(x_{n+1}) - \Psi_{\tau_n}(x_n)| \to 0$ *as* $n \to \infty$;
(3) *if X is a Hilbert space and the function h_0 is strongly convex, then one has $\|x_{n+1} - x_n\| \to 0$ as $n \to \infty$;*
(4) *if X is separable, then all limit points of the sequence $\{x_n\}$ are generalised critical points for τ_*;*
(5) *if X is separable and a limit point x_* of the sequence $\{x_n\}$ is feasible for the problem $(\mathcal{P})$, then it is also critical for this problem.*

Proof. 1. By the definition of penalty update (see Algorithmic Pattern 1) the sequences $\{\tau_n^{(k)}\}$, $k \in \mathcal{I} \cup \mathcal{E}$ are non-decreasing and bounded above by $\tau_{\max}$. Consequently, these sequence converge, which obviously implies that the sequence $\{\tau_n\}$ also converges to some $\tau_* \in \mathbb{R}^m_{++}$.

2. Suppose that the sequence $\{|\Psi_{\tau_n}(x_{n+1}) - \Psi_{\tau_n}(x_n)|\}$ does not converge to zero. Then bearing in mind Lemma 4.4 one can conclude that there exist $\varepsilon > 0$ and a subsequence $\{x_{n_s}\}$ such that

$$\Psi_{\tau_{n_s}}(x_{n_s+1}) \le \Psi_{\tau_{n_s}}(x_{n_s}) - \varepsilon \quad \forall s \in \mathbb{N}. \tag{4.2}$$

Fix any $s \in \mathbb{N}$. Observe that

$$\Psi_{\tau_{n_s}}(x_{n_s+1}) = \Psi_{\tau_{n_s+1}}(x_{n_s+1}) + \langle \tau_{n_s} - \tau_{n_s+1}, \psi(x_{n_s+1}) \rangle$$

By our assumption there exists $C > 0$ such that $\|\psi(x_n)\|_\infty \le C$ for all $n \in \mathbb{N}$, where $\|\cdot\|_\infty$ is the ℓ_∞ norm. Hence taking into account the fact that by definitions all components of the vector $\psi(x_{n_s+1})$ are nonnegative, while all components of the vector $\tau_{n_s} - \tau_{n_s+1}$ are nonpositive, one obtains that

$$\Psi_{\tau_{n_s}}(x_{n_s+1}) \ge \Psi_{\tau_{n_s+1}}(x_{n_s+1}) + C\langle \tau_{n_s} - \tau_{n_s+1}, \vec{\mathbf{1}} \rangle, \tag{4.3}$$

where $\vec{\mathbf{1}} = (1, \ldots, 1) \in \mathbb{R}^m$. Now applying Lemma 4.4 and repeating the same argument once again one gets that

$$\Psi_{\tau_{n_s+1}}(x_{n_s+1}) \ge \Psi_{\tau_{n_s+1}}(x_{n_s+2}) \ge \Psi_{\tau_{n_s+2}}(x_{n_s+2}) + C\langle \tau_{n_s+1} - \tau_{n_s+2}, \vec{\mathbf{1}} \rangle. \tag{4.4}$$

Summing up (4.3) and (4.4) one obtains

$$\Psi_{\tau_{n_s}}(x_{n_s+1}) \ge \Psi_{\tau_{n_s+2}}(x_{n_s+2}) + C\langle \tau_{n_s} - \tau_{n_s+2}, \vec{\mathbf{1}} \rangle.$$

Hence arguing by induction one can easily verify that

$$\Psi_{\tau_{n_s}}(x_{n_s+1}) \ge \Psi_{\tau_{n_{s+1}}}(x_{n_{s+1}}) + C\langle \tau_{n_s} - \tau_{n_{s+1}}, \vec{\mathbf{1}} \rangle \quad \forall s \in \mathbb{N},$$

which with the use of (4.2) implies that

$$\Psi_{\tau_{n_{s+1}}}(x_{n_{s+1}}) \le \Psi_{\tau_{n_s}}(x_{n_s}) - \varepsilon + C\langle \tau_{n_{s+1}} - \tau_{n_s}, \vec{\mathbf{1}} \rangle \quad \forall s \in \mathbb{N}.$$

Recall that the sequence $\{\tau_n\}$ converges to τ_*. Therefore $\{\tau_n\}$ is a Cauchy sequence, which implies that there exists $N \in \mathbb{N}$ such that for all $n, p \ge N$ the inequality

$\|\tau_n - \tau_p\|_\infty \le \varepsilon/2mC$ holds true. Clearly, there exists $s_0 \in \mathbb{N}$ such that $n_{s_0} \ge N$. Then

$$\Psi_{\tau_{n_{s+1}}}(x_{n_{s+1}}) \le \Psi_{\tau_{n_s}}(x_{n_s}) - \frac{\varepsilon}{2} \quad \forall s \ge s_0.$$

Consequently, $\Psi_{\tau_{n_s}}(x_{n_s}) \to -\infty$ as $s \to \infty$, which contradicts the facts that the penalty function Ψ_{τ_n} is bounded below on A for some $n \in \mathbb{N}$ and $\Psi_{\tau_p}(\cdot) \ge \Psi_{\tau_n}(\cdot)$ for all $p \ge n$ due to the penalty updating rule from Algorithmic Pattern 1.

3. Let X be a Hilbert space and h_0 be strongly convex. Then there exists $\mu > 0$ such that

$$h_0(x_{n+1}) - h_0(x_n) \ge \langle v_{0n}, x_{n+1} - x_n \rangle + \frac{\mu}{2}\|x_{n+1} - x_n\|^2 \quad \forall n \in \mathbb{N}.$$

By the definition of x_{n+1} and inequalities (3.2) one has

$$\Psi_{\tau_n}(x_n) = M_{\tau_n}(x_n, x_n, V_n) - h_0(x_n) \ge M_{\tau_n}(x_{n+1}, x_n, V_n) - h_0(x_n) \quad \forall n \in \mathbb{N}.$$

Hence applying the inequality for h_0 and the definitions of subgradients v_k and w_j one obtains that

$$\begin{aligned}
\Psi_{\tau_n}(x_n) &\ge M_{\tau_n}(x_{n+1}, x_n, V_n) - h_0(x_n) \\
&\ge M_{\tau_n}(x_{n+1}, x_n, V_n) - h_0(x_{n+1}) + \langle v_{0n}, x_{n+1} - x_n \rangle + \frac{\mu}{2}\|x_{n+1} - x_n\|^2 \\
&= f_0(x_{n+1}) + \sum_{i \in \mathcal{I}} \tau_n^{(i)} \max\big\{ g_i(x_{n+1}) - h_i(x_n) - \langle v_{ni}, x_{n+1} - x_n \rangle, 0 \big\} \\
&+ \sum_{j \in \mathcal{E}} \tau_n^{(j)} \max\big\{ g_j(x_{n+1}) - h_j(x_n) - \langle v_{nj}, x_{n+1} - x_n \rangle, \\
&h_j(x_{n+1}) - g_j(x_n) - \langle w_{nj}, x_{n+1} - x_n \rangle \big\} \\
&\ge \Psi_{\tau_n}(x_{n+1}) + \frac{\mu}{2}\|x_{n+1} - x_n\|^2
\end{aligned}$$

for all $n \in \mathbb{N}$ (see (3.1)). Therefore by the previous part of the theorem $\|x_{n+1} - x_n\| \to 0$ as $n \to \infty$.

4. Arguing by reductio ad absurdum, suppose that there is a limit point x_* of the sequence $\{x_n\}$ that is not a generalised critical point for τ_*. By the definition of the limit point there exists a subsequence $\{x_{n_s}\}$ converging to x_*. By [51, Thm. 2.4.13] the subdifferential mapping of a l.s.c. convex function is locally bounded, which implies that the corresponding sequences of subgradients $\{v_{n_s k}\}$ and $\{w_{n_s j}\}$ are bounded.

Recall that by our assumption the space X is separable. Therefore by the sequential version of the Banach-Alaoglu theorem any ball in the dual space X^* is sequentially weak* compact. Consequently, replacing, if necessary, the sequence $\{x_{n_s}\}$ with its subsequence one can suppose that the sequences $\{v_{n_s k}\}$ and $\{w_{n_s j}\}$ converge to some $\{v_k^*\}$ and $\{w_j^*\}$ in the weak* topology. Note that $v_k^* \in \partial h_k(x_*)$ for all $k \in \{0\} \cup \mathcal{I} \cup \mathcal{E}$ and $w_j^* \in \partial g_j(x_*)$ for all $j \in \mathcal{E}$ (see, e.g. [51, Thm. 2.4.2, part (ix)].

From the fact that x_* is not a generalised critical point for τ_* it follows that there exist $y \in A$ and $\varepsilon > 0$ such that

$$M_{\tau_*}(y, x_*, V^*) < M_{\tau_*}(x_*, x_*, V^*) - \varepsilon, \quad V^* = (v_0^*, v_1^*, \ldots, v_m^*, w_{\ell+1}^*, \ldots, w_m^*)$$

(see Def. 4.1). Since the sequence $\{V_{n_s}\}$ converes to V^* in the weak* topology, there exists $s_1 \in \mathbb{N}$ such that

$$M_{\tau_*}(y, x_*, V_{n_s}) < M_{\tau_*}(x_*, x_*, V_{n_s}) - \frac{3\varepsilon}{4} \quad \forall s \geq s_1.$$

Therefore, taking into account the definition of M_τ (see (3.1)), the main step of Algorithmic Pattern 1, and the facts that $x_{n_s} \to x_*$, $\varepsilon_n \to 0$, and the sequence $\{V_{n_s}\}$ is bounded one obtains that there exists $s_2 \geq s_1$ such that

$$\begin{aligned} M_{\tau_{n_s}}(x_{n_s+1}, x_{n_s}, V_{n_s}) &\leq M_{\tau_{n_s}}(y, x_{n_s}, V_{n_s}) + \varepsilon_n \\ &\leq M_{\tau_{n_s}}(x_{n_s}, x_{n_s}, V_{n_s}) - \frac{\varepsilon}{4} \quad \forall s \geq s_2. \end{aligned}$$

Hence with the use of (3.2) one gets that

$$\Psi_{\tau_{n_s}}(x_{n_s+1}) \leq \Psi_{\tau_{n_s}}(x_{n_s}) - \frac{\varepsilon}{2} \quad \forall s \geq s_0,$$

Now arguing in the same way as in the proof of the second statement of the theorem one can readily verify that the inequality above implies that $\Psi_{\tau_{n_s}}(x_{n_s}) \to -\infty$ as $s \to \infty$, which contradicts the assumption that the function Ψ_{τ_n} is bounded below on A for some $n \in \mathbb{N}$. Therefore, all limit points of the sequence $\{x_n\}$ are generalised critical point for τ_*.

The validity of statement 5 follows directly from the previous statement and Lemma 4.2. □

Remark 4.6. Note that in the case when X is a Hilbert space, the assumption on the strong convexity of the function h_0 is obviously not restrictive, since one can always replace the DC decomposition $f_0(x) = g_0(x) - h_0(x)$ of the objective function with the following one:

$$f_0(x) = \big(g_0(x) + \mu\|x\|^2\big) - \big(h_0(x) + \mu\|x\|^2\big)$$

for some $\mu > 0$.

Corollary 4.7. *Under the assumptions of the previous theorem for any $n \in \mathbb{N}$ one has*

$$0 \leq M_{\tau_n}(x_n, x_n, V_n) - M_{\tau_n}(x_{n+1}, x_n, V_n) \leq \Psi_{\tau_n}(x_n) - \Psi_{\tau_n}(x_{n+1}),$$

and $M_{\tau_n}(x_n, x_n, V_n) - M_{\tau_n}(x_{n+1}, x_n, V_n) \to 0$ as $n \to \infty$.

Proof. By the definition of x_{n+1} (see Algorithmic Pattern 1) one has

$$M_{\tau_n}(x_{n+1}, x_n, V_n) - M_{\tau_n}(x_n, x_n, V_n) \leq 0 \quad \forall n \in \mathbb{N},$$

Adding and subtracting $h_0(x_n)$ in the left-hand side of this inequality and applying inequalities (3.2) one gets that

$$\Psi_{\tau_n}(x_{n+1}) - \Psi(x_n) \leq M_{\tau_n}(x_{n+1}, x_n, V_n) - M_{\tau_n}(x_n, x_n, V_n) \leq 0,$$

which with the use of the second statement of the previous theorem implies the required result. □

5. Illustrative numerical examples

Let us give several simple illustrative finite dimensional numerical examples to help the reader better understand the way the exact multidimensional penalty DCA works, as well as to highligh some essential differences between this method and the steering exact penalty DCA from the recent paper [15].

In all examples, the initial value of the penalty parameter in Algorithmic Pattern 1 was chosen as $\tau_0 = (1, \ldots, 1)$, while for the steering exact penalty DCA it was chosen as $c_0 = 1$. The scaling coefficients γ_n in Algorithmic Pattern 1 were chosen as

$$\gamma_n = \begin{cases} 10/\|\psi(x_n)\|, & \text{if } \|\psi(x_n)\| \geq 0.1, \\ 10, & \text{if } 10^{-6} \leq \|\psi(x_n)\| < 0.1, \\ 0, & \text{otherwise,} \end{cases}$$

while the penalty parameter c_n in the steering exact penalty DCA was increased by the factor $\rho = 10$, each time one of the corresponding inequality was not satisfied (see [15]). We also chose parameters of this method as follows: $\eta_1 = \eta_2 = 0.1$ and $\varepsilon_{feas} = 0.01$.

The following inequalities

$$\varphi(x_n) < 10^{-6}, \quad \Psi_{\tau_n}(x_{n+1}) - \Psi_{\tau_n}(x_n) < 10^{-6}.$$

were used as the termination criterion. Finally, all convex optimization subproblems in both methods were solved with the use of `cvx`, a package for specifying and solving convex programs [17, 34].

Example 5.1. As the first example, we consider the following simple optimization problem with one nonsmooth nonconvex equality constraint:

$$\text{minimize} \;\; 20(x^{(1)} - 2)^2 + 20(x^{(2)})^2 \quad \text{subject to} \quad |x^{(1)}| - |x^{(2)}| = 0. \tag{5.1}$$

One has $X = \mathbb{R}^2$, $\mathcal{I} = \emptyset$, $\mathcal{E} = \{1\}$, and

$$g_0(x) = 20(x^{(1)} - 2)^2 + 20(x^{(2)})^2, \quad h_0(x) = 0$$
$$g_1(x) = |x^{(1)}|, \quad h_1(x) = |x^{(2)}|.$$

Problem (5.1) has two globally optimal solutions: $x_* = (1, \pm 1)$. We chose the point $x_0 = (-2, 0)$ as initial guess for both methods. Computation results for Algorithmic Pattern 1 are given in Table 1 and for the steering exact penalty DCA in Table 2.

Table 1. Output of the exact multidimensional penalty DCA for Example 5.1.

n	x_n	$f_0(x_n)$	$\varphi(x_n)$	τ_n
0	(-2, 0)	320	2	1
1	(1.975, 0)	0.0125	1.975	11
2	(1.725, 0.275)	3.0252	1.45	21
3	(1.475, 0.5251)	11.0254	0.95	31
4	(1.225, 0.775)	24.0254	0.45	41
5	(1, 1)	40	0	41

Table 2. Output of the steering exact penalty DCA for Example 5.1.

n	x_n	$f_0(x_n)$	$\varphi(x_n)$	c_n
0	(-2, 0)	320	2	1
1	(1.75, 0)	1.25	1.75	10
2	(1.75, -0.25)	2.5	1.5	10
3	(1, -1)	40	0	100

Computation results show that once the penalty parameter exceeds a certain threshold (namely, the least exact penalty parameter for problem (5.1); see e.g. [10, 36]), both methods find one of the globally optimal solutions in just one step. However, both methods find a sufficiently large value of the penalty parameter in different ways. The steering exact penalty DCA increases the penalty parameter to ensure a reasonable rate of decay of the infeasibility measure, while the exact multidimensional penalty DCA increases the penalty parameter each iteration till a feasible point of found.

More conservative penalty updates of the exact multidimensional penalty DCA (instead of multiplying the penalty parameter by the factor $\rho = 10$ as in the steering exact penalty DCA, one simply adds 10 to the current value of the penalty parameter, if the current iterate is infeasible) leads to the fact that this method requires more iterations to find a correct value of the penalty parameter than the steering exact penalty DCA. Note, however, that Algorithmic Pattern 1 is designed for problem with multiple constraints, while in the case of problems with a single constraint it is reduced to the simples penalty updating scheme: increase the penalty parameter by a constant value till a feasible point is found. The benefits of this method can be clearly seen only for problems with many constraints.

Example 5.2. Let us consider an example of a mathematical program with complementarity constraints (MPCC) given in [5, Example 3]. Namely, consider the following problem:

$$\begin{aligned} &\text{minimize} \enspace x^{(1)} + x^{(2)} \\ &\text{subject to} \enspace 1 - (x^{(2)})^2 \le 0, \quad x^{(1)}x^{(2)} \le 0, \quad x^{(1)} \ge 0, \quad x^{(2)} \ge 0. \end{aligned} \tag{5.2}$$

In this case $X = \mathbb{R}^2$ and one can define $\mathcal{I} = \{1, 2, 3, 4\}$, $\mathcal{E} = \emptyset$, and

$$g_0(x) = x^{(1)} + x^{(2)}, \quad h_0(x) = 0, \quad g_1(x) = 0, \quad h_1(x) = (x^{(2)})^2 - 1,$$
$$g_2(x) = 0.5(x^{(1)} + x^{(2)})^2, \quad h_2(x) = 0.5(x^{(1)})^2 + 0.5(x^{(2)})^2,$$
$$g_3(x) = -x^{(1)}, \quad h_3(x) = 0, \quad g_3(x) = -x^{(2)}, \quad h_3(x) = 0.$$

The unique globally optimal solution of problem (5.2) is $x_* = (0, 1)$. Note that the Mangasarian-Fromovitz constraint qualification is not satisfied at this point.

We take the point $x_0 = (0.1, 0.9)$, suggested in [5, Example 3], as initial guess for both methods. Computation results are given in Tables 3 and 4.

Although in this example both methods terminate after the same number of iterations, the exact multidimensional penalty DCA computes in two iteration a solution with higher accuracy than the steering exact penalty DCA computes in

TABLE 3. Output of the exact multidimensional penalty DCA for Example 5.2.

n	x_n	$f_0(x_n)$	$\varphi(x_n)$	τ_n
0	(0.1, 0.9)	1	0.28	(1, 1, 1, 1)
1	(-1.0238, 1.0056)	-0.02	1.0238	(1, 1, 11, 1)
2	(8.8E-11, 1)	1	0	(1, 1, 11, 1)
3	(-2.46E-12, 1)	1	0	(1, 1, 11, 1)

TABLE 4. Output of the steering exact penalty DCA for Example 5.2.

n	x_n	$f_0(x_n)$	$\varphi(x_n)$	c_n
0	(0.1, 0.9)	1	0.28	1
1	(-0.0056, 1.0056)	0.99	0.0055	10
2	(-3E-5, 1)	0.99	0	10
3	(3E-9, 1)	1	0	10

three iterations. What is more noteworthy, however, is the fact that the exact multidimensional penalty DCA increases only the penalty parameter corresponding to the constraint $x_1 \geq 0$. All other penalty parameters, including the ones corresponding to the nonlinear constraints, remain unchanged and equal to 1. This example clearly demonstrate potential benefits of using multidimensional penalty parameter.

Let us also note that both algorithms use a much smaller value of the penalty parameter than the line search exact penalty method using steering rules from [5] (see [5, Example 3]).

Example 5.3. Let us finally consider an example highlighting some drawbacks of the exact multidimensional penalty DCA. Namely, consider the following optimization problem

$$\begin{aligned} &\text{minimize} \quad x^{(1)} \quad \text{subject to} \quad (x^{(1)})^2 + 1 - x^{(2)} = 0, \\ &x^{(1)} - x^{(3)} - 1 = 0, \quad x^{(2)} \geq 0, \quad x^{(3)} \geq 0. \end{aligned} \tag{5.3}$$

introduced in [?] (see also [5, Example 1]). In this case $X = \mathbb{R}^2$ and one can define $\mathcal{I} = \{1, 2\}$, $\mathcal{E} = \{3, 4\}$, and

$$g_0(x) = x^{(1)}, \quad h_0(x) = 0, \quad g_1(x) = -x^{(2)}, \quad h_1(x) = 0,$$
$$g_2(x) = -x^{(3)}, \quad h_2(x) = 0, \quad g_3(x) = (x^{(1)})^2 + 1 - x^{(2)}, \quad h_3(x) = 0,$$
$$g_4(x) = x^{(1)} - x^{(3)} - 1, \quad h_4(x) = 0$$

The unique globally optimal solution is $x_* = (1, 2, 0)$. We use the starting point $x_0 = (-3, 11, 1)$ given in [5, Example 1]. Computation results are presented in Tables 5 and 6. Values of the objective function are not included into the tables, since they coincide with the first coordinate of x_n.

In this example the exact multidimensional penalty DCA terminates one iteration earlier than the steering exact penalty DCA. However, one can see the effect of overcorrection of the penalty parameter. Algorithmic Pattern 1 keeps increasing the multidimensional penalty parameter till a feasible point is found, even though there

Table 5. Output of the steering exact penalty DCA for Example 5.3.

n	x_n	$\varphi(x_n)$	τ_n
0	(- 3, 1, 1)	14	(1, 1, 1, 1)
1	(-3, 10, -3.1833)	4	(1, 10.6863, 1, 3.4851)
2	(-1, 0, 0)	4	(1, 10.6863, 8.0711, 10.5561)
3	(0.184, 0.3329, 0)	1.517	(1, 10.6863, 14.5869, 18.1419)
4	(0.848, 1.4987, -0.152)	0.372	(1, 16,3619, 22.8202, 18.1419)
5	(1, 1.9885, 0)	0.012	(1, 16.3619, 22.9357, 18.1419)
6	(1, 2, 0)	0	(1, 16.3619, 22.9357, 18.1419)

Table 6. Output of the exact multidimensional penalty DCA for Example 5.3.

n	x_n	$\varphi(x_n)$	c_n
0	(- 3, 1, 1)	14	1
1	(-3, 10, -3.1833)	4	1
2	(-2.1, 5.052, -2.289)	3.505	10
3	(-1.2, 2.035, -1.7368)	2.605	10
4	(-0.3, 0.685, -0.9834)	1.705	10
5	(0.6, 0.955, -0.2957)	0.805	10
6	(1, 1.92, 0)	0.08	10
7	(1, 2, 0)	0	10

is no real need for penalty increase, as is demonstrated by the computation results with fixed penalty parameter $\tau_2 = (1, 10.6863, 8.0711, 10.5561)$ given in Table 7.

Table 7. Output of the exact multidimensional penalty DCA with fixed penalty parameter for Example 5.3.

n	x_n	$\varphi(x_n)$
0	(- 3, 1, 1)	14
1	(-1.816, 3.597, 0)	3.517
2	(-0.632, 0.6985, 0)	2.333
3	(0.552, 0.6038, 0)	1.149
4	(1, 1.8996, 0)	0.1
5	(1, 2, 0)	0

Thus, one can conclude that a modification of Algorithmic Pattern 1 is needed, which would take into account progress towards feasibility, when a feasible point cannot be rapidly found. This is especially relevant for equality constrained problems. One possible way to modify Algorithmic Pattern 1 consists in incorporating multidimensional penalty updates from this method into the steering exact penalty DCA from [15]. Another approach can be based on behaviour of the infeasibility measure $\varphi(x_n)$ and/or the value of the penalty function $\Psi_{\tau_n}(x_n)$. The inequalities

$$\varphi(x_{n+1}) < (1-\eta)\varphi(x_n) \quad \text{and/or}$$
$$\Psi_{\tau_n}(x_{n+1}) - \Psi_{\tau_n}(x_n) \le (1-\eta)\Big(\Psi_{\tau_{n-1}}(x_n) - \Psi_{\tau_{n-1}}(x_{n-1})\Big)$$

with some small $\eta > 0$ might be considered as a reasonable criterion for whether a penalty update is needed, since under mild assumptions DCA converges with linear rate [25].

As another way of improving efficiency of Algorithmic Pattern 1, one can consider replacing the DCA, as a method for minimizing the penalty function $\Psi_{\tau_n}(\cdot)$ on each step of the algorithmic pattern, with boosted [1, 50] or inertial [8] versions of the DCA. A detailed analysis of such modifications of Algorithmic Pattern 1 is an interesting open problem for future research.

Acknowledgements

The author is sincerely grateful to the anonymous referee for taking the time to carefully read the paper and valuable comments and suggestions that helped to improve the quality of the paper.

References

[1] F. J. Aragón Artacho and P. T. Vuong, *The boosted difference of convex functions algorithm for nonsmooth functions*, SIAM J. Optim. **30** (2020), 980–1006.

[2] A. M. Bagirov, S. Taheri, K. Joki, N. Karmitsa and M. M. Mäkelä, *Aggregate subgradient method for nonsmooth DC optimization*, Optim. Lett. **15** (2021), 83–96.

[3] A. M. Bagirov and J. Ugon, *Codifferential method for minimizing nonsmooth DC functions*, J. Glob. Optim. **50** (2011), 3–22.

[4] R. S. Burachik, C. Y. Kaya and C. J. Price, *A primal-dual penalty method via rounded weighted-ℓ_1 Lagrangian duality*, Optim., 2021, Doi: 10.1080/02331934.2021.1934680.

[5] R. H. Byrd, G. Lopez-Calva and J. Nocedal, *A line search exact penalty method using steering rules*, Math. Program. **133** (2012), 39–73.

[6] R. H. Byrd, J. Nocedal and R. A. Waltz, *Steering exact penalty methods for nonlinear programming*, Optim. Meth. Softw. **23** (2008), 197–213.

[7] W. de Oliveira, *Proximal bundle methods for nonsmooth DC programming*, J. Glob. Optim. **75** (2019), 523–563.

[8] W. de Oliveira and M. P. Tcheou, *An inertial algorithm for DC programming*, Set-Valued Var. Anal. **27** (2019), 895–919.

[9] V. F. Demyanov and A. M. Rubinov, *Constructive Nonsmooth Analysis*, Peter Lang, Frankfurt am Main, 1995.

[10] M. V. Dolgopolik, *A unifying theory of exactness of linear penalty functions*, Optim. **65** (2016), 1167–1202.

[11] M. V. Dolgopolik, *Metric regularity of quasidifferentiable mappings and optimality conditions for nonsmooth mathematical programming problems*, Set-Valued Var. Anal. **28** (2020), 427–449.

[12] M. V. Dolgopolik, *A new constraint qualification and sharp optimality conditions for nonsmooth mathematical programming problems in terms of quasidifferentials*, SIAM J. Optim. **30** (2020), 2603–2627.

[13] M. V. Dolgopolik, *DC semidefinite programming and cone constrained DC optimization*, arXiv: 2102.01481, 2021.

[14] M. V. Dolgopolik, *Exact penalty functions with multidimensional penalty parameter and adaptive penalty updates*, Optim. Lett. **16** (2022), 1281–1300.

[15] M. V. Dolgopolik, *K Steering exact penalty DCA for nonsmooth DC optimization problems with equality and inequality constraints*, arXiv: 2107.01433, 2022.
[16] I. Ekeland and R. Temam, *Convex Analysis and Variational Problems*, SIAM, Philadelphia, 1999.
[17] M. Grand and S. Boyd, *Graph implementations for nonsmooth convex programs*, in: Recent Advances in Learning and Control, Lecture Notes in Control and Information Sciences, V. Blondel, S. Boyd, and H. Kimura (eds), Springer, London, 2008, pp. 95–110.
[18] A. D. Ioffe and V. M. Tihomirov, *Theory of Extremal Problems*, North-Holland, Amsterdam, 1979.
[19] K. Joki and A. M. Bagirov, *Bundle methods for nonsmooth DC optimization, in: Numerical Nonsmooth Optimization*, A. Bagirov, M. Gaudioso, N. Karmitsa, M. Mäkelä, S. Taheri (eds), Springer, Cham, 2020, pp. 263–296.
[20] K. Joki, A. M. Bagirov, N. Karmitsa, M. Mäkelä and S. Taheri, *Double bundle method for finding Clarke stationary points in nonsmooth DC programming*, SIAM J. Optim. **28** (2018), 1892–1919.
[21] K. Joki, A. M. Bagirov, N. Karmitsa, and M. M. Mäkelä, *A proximal bundle method for nonsmooth DC optimization utilizing nonconvex cutting planes*, J. Glob. Optim. **68** (2017), 501–535.
[22] N. Karmitsa, *Test Problems for Large-Scale Nonsmooth Minimization. Reports of the Department of Mathematical Information Technology. Series B. Scientific Computing. No. B.4/2007*, University of Jyväskylä, 2007.
[23] H. A. Le Thi, V. N. Nuynh and T. Pham Dinh, *DC programming and DCA for general DC programs*, in: Advanced Computational Methods for Knowledge Engineering, T. van Do, H. A. L. Thi, N. T. Nguyen (eds), Springer, Berlin, Heidelberg, 2014, pp. 15–35.
[24] H. A. Le Thi and T. Pham Dinh, *The DC (difference of convex functions) programming and DCA revisited with DC models of real world nonconvex optimization problems*, Ann. Oper. Res. **133** (2005), 23–46.
[25] H. A. Le Thi and T. Pham Dinh, *k DC programming and DCA: thirty years of developments*, Math. Program. **169** (2018), 5–68.
[26] H. A. Le Thi, T. Pham Dinh and L. D. Muu, *Numerical solution for optimization over the efficient set by D.C. optimization algorithm*, Oper. Res. Lett. **19** (1996), 117–128.
[27] T. Lipp and S. Boyd, *Variations and extension of the convex-concave procedure*, Optim. Eng. **17** (2016), 263–287.
[28] B. Luderer, *Does the special choice of quasidifferentials influence necessary minimum conditions?*, in: Advances in Optimization, W. Oettli, D. Pallaschke (eds), Springer, Berling, 1992, pp. 256–266.
[29] B. Luderer, R. Rösiger and U. Würker, On necessary conditions in quasidifferential calculus: Independence of the specific choice of quasidifferentials. Optim. **22** (1991), 643–660.
[30] O. Montonen and K. Joki, *Bundle-based descent method for nonsmooth multiobjective DC optimization with inequality constraints*, J. Glob. Optim. **72** (2018), 403–429.
[31] T. Pham Dinh and H. A. Le Thi, *Convex analysis approach to D.C. programming: theory, algorithms and applications*, Acta Math. Vietnamica **22** (1997), 289–355.
[32] T. Pham Dinh and H. A. Le Thi, *D.C. optimization algorithms for solving the trust region subproblem*, SIAM J. Optim. **8** (1998), 476–505.
[33] T. Pham Dinh and H. A. Le Thi, *Recent advances in DC programming and DCA*, in: Transactions on Computational Intelligence XIII, N. T. Nguyen and H. A. L. Thi (eds), Springer, Berlin, Heidelberg, 2014, pp. 1–37.
[34] CVX Research, Inc., CVX: Matlab Software for Disciplined Convex Programming, version 2.2, http://cvxr.com/cvx, 2020.
[35] T. Pham Dinh and E. B. Souad, *Algorithms for solving a class of nonconvex optimization problems. Methods of subgradients*, in: Fermat Days 85: Mathematics for Optimization. North-Holland Mathematics Studies, J.-B. Hiriart-Urruty (ed.), vol. 129, Norht-Holland, msterdam, 1986, pp. 249–271.

[36] A. M. Rubinov and X. Yang, *Lagrange-Type Functions in Constrained Non-Convex Optimization*, Kluwer Academic Publishers, Dordrecht, 2003.
[37] A. S. Strekalovsky, *Maximizing a state convex lagrange functional in optimal control*, Autom. Remote Control **73** (2012), 949–961.
[38] A. S. Strekalovsky, *Global optimality conditions for optimal control problems with function of A.D Alexandrov*, J. Optim. Theory Appl. **159** (2013), 297–321.
[39] A. S. Strekalovsky, *On Nonconvex Optimization Problems with D.C. Equality and Inequality Constraints*, IFAC-PapersOnLine **51** (2018), 895–900.
[40] A. S. Strekalovsky, *Local search for nonsmooth DC optimization with DC equality and inequality constraints*, in: Numerical Nonsmooth Optimization. State of the Art Algorithms, A. M. Bagirov, M. Gaudioso, N. Karmitsa, M. M. Mäkelä, S. Taheri (eds), Springer, Cham, 2020, pp. 229–262.
[41] A. S. Strekalovsky, *A local search scheme for the inequality-constrained optimal control problem*, in: Mathematical Optimization Theory and Operations Research. MOTOR 2021, P. Pardalos, M. Khachay, A. Kazakov (eds), Springer, Cham, 2021, pp. 17–31.
[42] A. S. Strekalovsky and M. V. Yanulevich, *Global search in the optimal control problem with a terminal objective functional represented as the difference of two convex functions*, Comput. Math. Math. Phys. **48** (2008), 1119–1132.
[43] A. S. Strekalovsky and M. V. Yanulevich, *Global search in a nonconvex optimal control problem*, J. Comput. Syst. Sci. Int. **52** (2013), 893–908.
[44] A. S. Strekalovsky and M. V. Yanulevich, *Global search in the optimal control problem with a terminal objective functional represented as the difference of two convex functions*, Comput. Math. Math. Phys. **48** (2008), 1119–1132.
[45] A. H. Tor, A. Bagirov and B. Karasözen, *Aggregate codifferential method for nonsmooth DC optimization*, J. Comput. Appl. Math. **259** (2014), 851–867.
[46] W. van Ackooij and W. de Oliveira, *Non-smooth DC-constrained optimization: constraint qualification and minimizing methodologies*, Optim. Methods Softw. **34** (2019), 890–920.
[47] W. van Ackooij and W. de Oliveira, *Nonsmooth and nonconvex optimization via approximate difference-of-convex decompositions*, J. Optim. Theory Appl. **182** (2019), 49–80.
[48] W. van Ackooij and S. Demassey and P. Javal and H. Morais and W. de Oliveira and B. Swaminathan, *A bundle method for nonsmooth DC programming with application to chance-constrained problems*, Comput. Optim. Appl. **78** (2021), 451–490.
[49] A. Wächter and L. T. Biegler, *Failure of global convergence for a class of interior point methods for nonlinear programming*, Math. Program. **88** (2000), 565–574.
[50] T. Wu and X. Gu and Z. Li and J. Niu And T. Zeng, *Efficient boosted DC algorithm for nonconvex image restoration with Rician noise*, SIAM J. Imaging Sci. **15** (2022), 424–454.
[51] C. Zălinescu, *Convex Analysis in General Vector Spaces*, World Scientific, Singapore, 2002.

Manuscript received September 6 2022
revised December 6 2022

M. V. Dolgopolik
Institute for Problems in Mechanical Engineering of the Russian Academy of Sciences, Saint Petersburg, Russia
E-mail address: dmv@ipme.ru