

A D.C. OPTIMIZATION ALGORITHM FOR SOLVING THE TRUST-REGION SUBPROBLEM*

PHAM DINH TAO[†] AND LE THI HOAI AN[†]

Abstract. This paper is devoted to difference of convex functions (d.c.) optimization: d.c. duality, local and global optimality conditions in d.c. programming, the d.c. algorithm (DCA), and its application to solving the trust-region problem. The DCA is an iterative method that is quite different from well-known related algorithms. Thanks to the particular structure of the trust-region problem, the DCA is very simple (requiring only matrix-vector products) and, in practice, converges to the global solution. The inexpensive implicitly restarted Lanczos method of Sorensen is used to check the optimality of solutions provided by the DCA. When a nonglobal solution is found, a simple numerical procedure is introduced both to find a feasible point having a smaller objective value and to restart the DCA at this point. It is shown that in the nonconvex case, the DCA converges to the global solution of the trust-region problem, using only matrix-vector products and requiring at most $2m+2$ restarts, where m is the number of distinct negative eigenvalues of the coefficient matrix that defines the problem. Numerical simulations establish the robustness and efficiency of the DCA compared to standard related methods, especially for large-scale problems.

Key words. d.c. optimization, d.c. duality, global and local optimality conditions, regularization techniques, DCA, Lanczos method, trust-region subproblem

AMS subject classifications. 65K05, 65K10, 90C30, 90C35

PII. S1052623494274313

1. Introduction. In this paper we shall be concerned with the following nonconvex quadratic optimization problems with nonempty solution sets:

$$(Q_1) \quad \min \left\{ \frac{1}{2} x^T A x + b^T x : \|x\| \leq r \right\},$$

$$(Q_2) \quad \min \left\{ \frac{1}{2} x^T A x + b^T x : \|x\| = r \right\},$$

where A is an $n \times n$ real symmetric matrix, $b \in \mathbb{R}^n$, r is a positive number and $\|\cdot\|$ denotes the Euclidean norm of \mathbb{R}^n . If A is positive semidefinite, then (Q_1) is a convex quadratic problem; in general, (Q_1) is nonconvex. Problem (Q_2) , whose feasible domain is a sphere, is always nonconvex even if A is positive semidefinite. (Q_1) and (Q_2) are among a few nonconvex optimization problems which possess a complete characterization of their solutions. These problems play an important role in optimization and numerical analysis. For example, for solving the problem of minimizing a twice continuously differentiable function on the whole space \mathbb{R}^n , D. M. Gay [5], J. J. Moré and D. C. Sorensen [16], [17], [36], and T. Pham Dinh and his coworkers [1], [7], [24], [25], [26] have used trust-region methods. It is known that these methods are of Newton type and consist of solving subproblems of the form (Q_1) ; consequently, (Q_1) has been often called a trust-region subproblem. Furthermore, in numerical analysis, several problems, e.g., constrained eigenvalue and quadratically

*Received by the editors August 20, 1994; accepted for publication (in revised form) November 6, 1996.

<http://www.siam.org/journals/siopt/8-2/27431.html>

[†]Mathematical Modelling and Applied Optimization Group, LMI-INSA Rouen - CNRS URA 1378, BP 08, 76131 Mont Saint Aignan, France (pham@insa-rouen.fr, lethi@insa-rouen.fr).

constrained least squares problems, can be formulated in the forms of (Q_1) and (Q_2) [4], [6].

To our knowledge, the first efficient (primal-dual) algorithm for solving (Q_1) is due to J. J. Moré and D. C. Sorensen [17]. These authors used a safeguarding technique combined with an adapted Newton method (due to Hebden) for computing a saddle-point of the Lagrangian function $L(x, \lambda)$. In the same primal-dual approach T. Pham Dinh and L. T. Hoai An [24], [26] proposed a dichotomy process as an alternative for the safeguarding technique of J. J. Moré and D. C. Sorensen. According to many comparative numerical simulations [7], [24], [26], the dichotomy algorithm seems to be less expensive than the safeguarding one. Moreover, it can be also applied to solving (Q_2) .

G. H. Golub, W. Gander, and U. Von Matt [4], [6] studied (Q_2) from both theoretical and computational viewpoints. These authors mentioned algorithmic difficulties of (Q_2) caused by the nonconvexity of the constraint $\{x : \|x\| = r\}$. In [4] the sensitivity of the solutions to the primal problem was discussed. More recently, G. H. Golub and U. Von Matt [6] proposed a new computational approach to the solution of (Q_2) in the normal case. The method again uses the theoretical tools developed in [4]. It seems to be well suited when A is a large sparse matrix.

J. M. Martinez [15] has investigated the nature of local nonglobal solutions of (Q_1) and (Q_2) and has shown the following interesting property (especially for local algorithms): these problems have at most one local nonglobal solution. Moreover, being inspired by G. E. Forsythe and G. H. Golub's work [3], S. Lucidi, L. Palagi, and M. Roma [13] have stated a very nice result: in (Q_1) the objective function can admit at most $2m + 2$ different values at Kuhn-Tucker points, where m is the number of distinct negative eigenvalues of A .

The algorithms in [17], [24] require matrix factorizations which are not feasible in the large-scale setting. Recently, matrix-free methods [32], [35], [38] have been introduced to solve large-scale trust-region subproblems. These methods recast the trust-region subproblem in terms of a parametrized eigenvalue problem and require matrix-vector products only. Sorensen's algorithm [38] provides a superlinearly convergent scheme to adjust the parameter and find the optimal solution from the eigenvector of the parametrized problem, as long as the hard case does not occur. The so-called hard case [17] is characterized by whenever the vector b is orthogonal to the eigenspace of A corresponding to its smallest eigenvalue λ_1 and there is a solution u to the system $(A - \lambda_1 I)u = -b$ such that $\|u\| < r$. This algorithm is linearly convergent in the hard case. The implicitly restarted Lanczos method (IRLM) introduced by D. C. Sorensen [12], [37] which meets the requirements of limited storage and reliance only on matrix-vector products is the recommended technique used in [38] to find the smallest eigenvalue and the corresponding eigenvector of the parameterized problem. A primal-dual semidefinite framework for the trust-region subproblem is proposed by F. Rendl and H. Wolkowicz in [32], where a dual simplex-type method is used in the general iteration and a primal simplex-type method provides steps for the hard case iteration. In [32] a block Lanczos routine is used at each iteration instead of the IRLM. The new matrix-free algorithm of S. A. Santos and D. C. Sorensen [35] improves upon that of D. C. Sorensen by introducing a unified iteration that naturally includes the hard case. It is shown that this algorithm is superlinearly convergent in all cases. Numerical simulations given in [35] proved the superiority of S. A. Santos and D. C. Sorensen's algorithm upon that of D. C. Sorensen. Moreover, these tests indicate that the former has advantage over F. Rendl and H. Wolkowicz's algorithm.

In convex approach to nonconvex nondifferentiable optimization the first author has extensively studied subgradient methods for solving convex maximization problems (see [20] and references therein). The d.c. duality investigated by J. F. Toland [39] and the works on d.c. optimization by T. Pham Dinh and L. T. Hoai An [7], [27], [28], [29], [30] can be considered as a natural generalization of the aforementioned works concerning convex maximization. DCA based on the duality in d.c. optimization had been introduced by T. Pham Dinh and S. Elberoussi in [23]. Important developments and improvements for the DCA from both theoretical and numerical viewpoints have been completed after the works by the authors [7], [8], [27], [28], [29], [30] appeared. These algorithms are actually among a few algorithms which allow to solve large-scale d.c. optimization problems [7], [8]. Due to their local character they cannot guarantee the globality of computed solutions for general d.c. programs. In general the DCAs, converge to a local solution; however, we observe that the DCAs, converge quite often to a global one [7], [8].

The d.c. objective function (of a d.c. program) has infinitely many d.c. decompositions which may have an important influence on the qualities (robustness, stability, rate of convergence, and globability of sought solutions) of the DCA. So, it is particularly interesting to obtain various equivalent d.c. forms for the primal and dual problems. The Lagrangian duality without gap in d.c. optimization [7], [28], [30] and regularization techniques partially answer this concern. In practice, regularization techniques using the kernel $\lambda/2\|\cdot\|^2$ and inf-convolution may provide interesting d.c. decompositions of objective functions for the DCA [7], [27], [29]. Furthermore, it is worth noting that by using conjointly suitable d.c. decompositions of convex functions and proximal regularization techniques [7], [27], [29] we can obtain the proximal point algorithm [14], [34] and the Goldstein–Levitin–Polyak subgradient projection method [31] as particular cases of the DCA. It would be interesting to find conditions on the choice of the d.c. decompositions and the initial point x^o to ensure the convergence of the DCA to a global minimizer. In practice, the DCA have been successfully applied to many large-scale d.c. optimization problems and proved to be more robust and efficient than related standard methods [7], [8], [27], [29].

The main part of the paper is dealing with the d.c. optimization framework (d.c. duality, local and global optimalities for d.c. programming, and the DCA) and its application to the treatment of problem (Q_1) since (Q_2) can be simply transformed into an equivalent problem of the form (Q_1) .

We first briefly present global optimality conditions in (Q_1) and (Q_2) [1], [2], [5], [7], [17], [24], [25], [26], [36]. These results express the same important fact: writing the constraints in (Q_1) and (Q_2) in equivalent forms, we obtain [26], [30] the stability of the Lagrangian duality relative to them, i.e., there is no duality gap and the solution sets of (Q_1) and (Q_2) can be described with the help of their dual solutions exactly as in convex programming. This approach emphasizes main properties of the dual objective function and is particularly useful to the development of various primal-dual algorithms for solving these problems. The algorithms in [17], [24], [32], [35], [38] are originated from this primal-dual approach.

In the third section we study the d.c. optimization framework, the description of the DCA and its convergence. The key point which makes a unified and deep d.c. optimization theory possible relies on the particular structure of the objective function to be minimized on \mathbb{R}^n :

$$f(x) = g(x) - h(x)$$

with g and h being convex on \mathbb{R}^n . One works then actually with the powerful convex

analysis tools applied to the two d.c. components g and h of the d.c. function f . With the help of the functional conjugate notion, the d.c. duality associates a primal d.c. program with a dual d.c. one and states relationships between them. More precisely, the d.c. duality is built on the fundamental feature of proper lower semicontinuous convex functions; such functions $\theta(x)$ are the supremum of the affine functions

$$\sup\{\langle x, y \rangle - \theta^*(y) : y \in \mathbb{R}^n\}.$$

Thanks to a symmetry in the d.c. duality (the bidual d.c. program is exactly the primal one) and the d.c. duality transportation of global minimizers, solving a d.c. program implies solving the dual one and vice versa. Furthermore, it may be very useful when one program is easier to be solved than the other (for example, one can directly solve Problem (Q_1) but not its dual d.c. program). The equality of the optimal value in the primal and dual d.c. programs can be easily translated (with the help of ε -subdifferentials of the d.c. components) in global optimality conditions. These conditions have been stated for the first time by J. B. Hiriart-Urruty [9] in a complicated way and mark the passage from convex optimization to nonconvex optimization. They are nice but difficult to use for devising solution methods to d.c. programs.

Local d.c. optimality conditions constitute the basis of the DCA. In general, it is not easy to state them as in global d.c. optimality and very few properties have been found which are useful in practice. Therefore, we shall present in section 3 the most significative results on d.c. programming; most of them are new. In particular, we give there a new elegant property concerning sufficient local d.c. optimality conditions and their consequences, e.g., the d.c. duality transportation of local minimizers ((ii) of Theorem 6). The latter is very helpful in establishing relationships between local minimizers of primal and dual d.c. programs. All these results are indispensable in the understanding of DCA for locally solving primal and dual d.c. programs. The proof of the convergence of the DCA will be presented in the appendix.

An application of the DCA to solving the trust-region subproblem (Q_1) is given in the fourth section. The DCA for this problem is quite different from the aforementioned algorithms: it is an iterative method based on the local optimality in d.c. programming. For solving (Q_1) we propose a quite appropriate d.c. decomposition. It turns out that the DCA becomes very simple: it indifferently treats both the standard (or normal) and hard cases and requires matrix–vector products only. Theoretically, the DCA generates a sequence of feasible points $\{x^k\}$ which decrease the values of the original quadratic function and converges to a Kuhn–Tucker point of (Q_1) . Main results for d.c. optimization show the convergence of the DCA to a local solution for certain classes of d.c. programs. These do not concern (Q_1) directly; however, the foundations of the d.c. optimization theory enable us to conjecture that the DCA converges to a local solution to (Q_1) and so, in general, to a global solution to this problem according to J. M. Martinez's results [15]. For the trust-region subproblem, it is interesting to note the following paradox: checking a global solution is easier than checking a local nonglobal one [15]. We take advantage of this fact to introduce a simple numerical procedure for checking the global optimality of solutions x^* given by the DCA or finding a feasible point \bar{x} having a smaller objective value in case x^* is not global. This procedure uses the efficient and inexpensive IRLM to compute the smallest eigenvalue of A and a corresponding eigenvector. In case of nonglobal solutions, we restart DCA with the new initial point \bar{x} . Finally, we can affirm that the DCA (with at most $2m + 2$ restarting procedures), which requires only matrix–vector products too, converges to a global solution.

In the last section we present computational results and comments. A lot of numerical experiments have been done to compare the DCA with some known related algorithms, among them the new matrix-free algorithm of S. A. Santos and D. C. Sorensen [35]. They have proved the robustness and the efficiency of the DCA, especially in the large-scale setting, and the fact that in practice the DCA (without restarting procedure) converges to a global solution of the trust-region subproblem.

2. Global optimality conditions for (Q_1) and (Q_2) . We first summarize several well-known global optimality conditions for (Q_1) whose proofs can be found in [1], [16], [17], [25], [26], [30], [36].

THEOREM 2.1. *x^* is a solution to Problem (Q_1) if and only if there exists $\lambda^* \geq 0$ such that*

- (i) $(A + \lambda^* I)x^* = -b$,
- (ii) $\lambda^*(\|x^*\| - r) = 0, \|x^*\| \leq r$,
- (iii) $(A + \lambda^* I)$ is positive semidefinite.

Moreover, λ^* is unique.

The conditions (i) and (ii) are Kuhn–Tucker conditions (necessary local optimality conditions). So, only the last condition is crucial and does express the passage from “convex” to “nonconvex” in (Q_1) .

We note that (Q_2) is equivalent to a problem of the form (Q_1) . Namely, it is equivalent to

$$(1) \quad \min \left\{ \frac{1}{2}x^T(A + \gamma I)x + b^T x \quad : \quad \|x\| \leq r \right\},$$

where γ is a real number such that $A + \gamma I$ is not positive semidefinite. Indeed, (Q_2) is equivalent to

$$(2) \quad \min \left\{ \frac{1}{2}x^T(A + \gamma I)x + b^T x \quad : \quad \|x\| = r \right\},$$

and in case $A + \gamma I$ is not positive semidefinite problems (1) and (2), by Theorem 2.1, have the same sets of solutions. Thus, solving (Q_2) is reduced to solving problem (1) which is of the form (Q_1) .

In [36] D. C. Sorensen gave a sufficient condition for optimality to problem (Q_2) . In [24] T. Pham Dinh proved that this condition is not only sufficient, but also necessary. This result was also given partially in R. Fletcher [2] using the optimality conditions of Theorem 2.1. It looks as follows.

THEOREM 2.2. *x^* is a solution to (Q_2) if and only if there exists λ^* such that*

- (i) $(A + \lambda^* I)$ is positive semidefinite,
- (ii) $(A + \lambda^* I)x^* = -b$,
- (iii) $\|x^*\| = r$.

This number λ^* is unique.

Denote by $\lambda_1 \leq \dots \leq \lambda_n$ the eigenvalues of A . Let ϕ be a function defined on $\mathbb{R} \setminus \{-\lambda_i : i = 1, \dots, n\}$ by

$$\phi(\lambda) = \|x(\lambda)\|,$$

where $x(\lambda)$ is the solution of $(A + \lambda I)x = -b$.

The following nice result which has been stated very recently by J. Martinez [15] strengthens the ability for the DCA to reach a solution (global minimizer) of (Q_1) .

THEOREM 2.3. (i) *If x^* is a local nonglobal minimizer for (Q_1) or (Q_2) , then $(A + \lambda^*I)x^* = -b$ with $\lambda^* \in] - \lambda_2, -\lambda_1[$ and $\phi'(\lambda^*) \geq 0$. If x^* is a local nonglobal minimizer for (Q_1) , then $\lambda^* \geq 0$.*

(ii) *There exists at most one local nonglobal minimizer for (Q_1) or (Q_2) .*

(iii) *If $\|x^*\| = r$, $(A + \lambda^*I)x^* = -b$ for some $\lambda^* \in] - \lambda_2, -\lambda_1[$ and $\phi'(\lambda^*) > 0$, then x^* is a strict local minimizer for (Q_2) . If, in addition, $\lambda^* > 0$, then x^* is also a strict local minimizer for (Q_1) .*

(iv) *If b is orthogonal to some eigenvector associated with λ_1 , then there is no local nonglobal minimizer for (Q_1) and (Q_2) .*

Finally, a result in S. Lucidi, L. Palagi, and M. Roma [13] guarantees that in the nonconvex case the restarting procedure can be repeated at most $2m + 2$ times, where m is the number of the distinct negative eigenvalues of A .

THEOREM 2.4. *In (Q_1) the objective function can admit at most $2m + 2$ different values at Kuhn–Tucker points, where m is the number of the distinct negative eigenvalues of A .*

3. D.c. optimization. Let the space $X = \mathbb{R}^n$ be equipped with the canonical inner product $\langle \cdot, \cdot \rangle$. Thus the dual space Y of X can be identified with X itself. The Euclidean norm of X is denoted by $\| \cdot \| = \langle \cdot, \cdot \rangle^{1/2}$. Denote by $\Gamma_o(X)$ the set of all proper lower semicontinuous convex functions on X . The conjugate function g^* of $g \in \Gamma_o(X)$ is a function belonging to $\Gamma_o(Y)$ and defined by

$$g^*(y) = \sup\{\langle x, y \rangle - g(x) : x \in X\}.$$

For a convex set C in X the indicator function of C is denoted by $\chi_C(x) = 0$ if $x \in C$, $+\infty$ otherwise. We shall use the following usual notations of [33]

$$\text{dom } g = \{x \in X : g(x) < +\infty\}.$$

For $\epsilon > 0$ and $x^o \in \text{dom } g$, the symbol $\partial_\epsilon g(x^o)$ denotes ϵ -subdifferential of g at x^o , i.e.,

$$\partial_\epsilon g(x^o) = \{y \in Y : g(x) \geq g(x^o) + \langle x - x^o, y \rangle - \epsilon \quad \forall x \in X\},$$

while $\partial g(x^o)$ stands for the usual (or exact) subdifferential of g at x^o . Also, $\text{dom } \partial g = \{x \in X : \partial g(x) \neq \emptyset\}$ and $\text{range } \partial g = \cup\{\partial g(x) : x \in \text{dom } \partial g\}$. We adopt in the sequel the convention $+\infty - (+\infty) = +\infty$.

A d.c. program is that of the form

$$(P) \quad \alpha = \inf\{f(x) := g(x) - h(x) : x \in X\},$$

where g and h belong to $\Gamma_o(X)$. Such a function f is called a d.c. function on X while g and h are called its d.c. components.

Using the definition of conjugate functions, we have

$$\begin{aligned} \alpha &= \inf\{g(x) - h(x) : x \in X\} = \inf\{g(x) - \sup\{\langle x, y \rangle - h^*(y) : y \in Y\} : x \in X\} \\ &= \inf\{\beta(y) : y \in Y\} \end{aligned}$$

with

$$(P_y) \quad \beta(y) = \inf\{g(x) - (\langle x, y \rangle - h^*(y)) : x \in X\}.$$

It is clear that $\beta(y) = h^*(y) - g^*(y)$ if $y \in \text{dom } h^*$, $+\infty$ otherwise. Finally, we state the dual problem

$$\alpha = \inf\{h^*(y) - g^*(y) : y \in \text{dom } h^*\}$$

which is written according to the above convention as

$$(D) \quad \alpha = \inf\{h^*(y) - g^*(y) : y \in Y\}.$$

We observe the perfect symmetry between primal and dual programs (P) and (D): the dual program to (D) is exactly (P).

Note that the finiteness of α merely implies that

$$(3) \quad \text{dom } g \subset \text{dom } h \quad \text{and} \quad \text{dom } h^* \subset \text{dom } g^*.$$

Such inclusions will be assumed throughout the paper.

This d.c. duality was first studied by J. F. Toland [39] in a more general framework. It can be considered as a logical generalization of our earlier works concerning convex maximization ([20] and references therein).

It is worth noting the richness of the set of d.c. functions on X , denoted by $\mathcal{DC}(X)$ [7], [9], [10], [27]:

(i) $\mathcal{DC}(X)$ is a subspace containing the class of lower- \mathcal{C}^2 functions (f is said to be lower- \mathcal{C}^2 if f is locally a supremum of a family of \mathcal{C}^2 functions). In particular, $\mathcal{DC}(X)$ contains the space $\mathcal{C}^{1,1}(X)$ of functions whose gradient is locally Lipschitzian on X .

(ii) $\mathcal{DC}(X)$ is closed under all the operations usually considered in optimization. In particular, a linear combination of $f_i \in \mathcal{DC}(X)$ belongs to $\mathcal{DC}(X)$, a finite supremum of d.c. functions is d.c. Moreover, the product of two d.c. functions remains d.c. [7], [9], [10], [27].

(iii) Under some caution we can say that $\mathcal{DC}(X)$ is the subspace generated by the convex cone $\Gamma_o(X) : \mathcal{DC}(X) = \Gamma_o(X) - \Gamma_o(X)$. This relation marks the passage from convex optimization to nonconvex optimization and also indicates that $\mathcal{DC}(X)$ constitutes a minimal realistic extension of $\Gamma_o(X)$.

A point x^* is said to be a *local minimizer* of $g - h$ if $g(x^*) - h(x^*)$ is finite (i.e., $x^* \in \text{dom } g \cap \text{dom } h$) and there exists a neighborhood U of x^* such that

$$(4) \quad g(x^*) - h(x^*) \leq g(x) - h(x) \quad \forall x \in U.$$

Under the convention $+\infty - (+\infty) = +\infty$, the property (4) is equivalent to $g(x^*) - h(x^*) \leq g(x) - h(x)$, $\forall x \in U \cap \text{dom } g$.

A point x^* is said to be a *critical point* of $g - h$ if $\partial g(x^*) \cap \partial h(x^*) \neq \emptyset$.

The interior of the set S in X is denoted by $\text{int } S$. Moreover, if S is convex then $\text{ri } S$ stands for the relative interior of S .

A convex function f on X is said to be *essentially differentiable* if it satisfies the following three conditions [33]:

- (i) $C = \text{int } (\text{dom } f) \neq \emptyset$,
- (ii) f is differentiable on C ,
- (iii) $\lim_{k \rightarrow \infty} \|\nabla f(x^k)\| = +\infty$ for every sequence $\{x^k\}$ which converges to a point at the boundary of C .

Let $\rho \geq 0$ and C be a convex subset of X . One says that the function $\theta : C \rightarrow \mathbb{R} \cup \{+\infty\}$ is ρ -convex if

$$\theta[\lambda x + (1 - \lambda)x'] \leq \lambda\theta(x) + (1 - \lambda)\theta(x') - \frac{\lambda(1 - \lambda)}{2}\rho\|x - x'\|^2 \quad \forall \lambda \in]0, 1[\quad \forall x, x' \in C.$$

It amounts to say that $\theta - (\rho/2)\|\cdot\|^2$ is convex on C . The modulus of strong convexity of θ on C , denoted by $\rho(\theta, C)$ or $\rho(\theta)$ if $C = X$, is given by

$$(5) \quad \rho(\theta, C) = \sup\{\rho \geq 0 : \theta - (\rho/2)\|\cdot\|^2 \text{ is convex on } C\}.$$

Clearly, the ρ -convexity implies the convexity. One says that θ is *strongly convex* on C if $\rho(\theta, C) > 0$.

For $f \in \Gamma_o(X)$ and $\lambda > 0$ the Moreau–Yosida regularization of f with parameter λ , denoted by f_λ , is the inf-convolution of f and $1/2\lambda\|\cdot\|^2$. The function f_λ is continuously differentiable, underapproximates f without changing the set of minimizers and $(f_\lambda)_\mu = f_{\lambda+\mu}$. More precisely, $\nabla f_\lambda = 1/\lambda[I - (I + \lambda\partial f)^{-1}]$ is Lipschitzian with the ratio $1/\lambda$. The operator $(I + \lambda\partial f)^{-1}$ is called the proximal mapping associated to λf [34].

Let \mathcal{P} and \mathcal{D} denote the solution sets of problems (P) and (D), respectively, and let

$$\mathcal{P}_l = \{x^* \in X : \partial h(x^*) \subset \partial g(x^*)\}, \quad \mathcal{D}_l = \{y^* \in Y : \partial g^*(y^*) \subset \partial h^*(y^*)\}.$$

We present below some fundamental results of d.c. optimization which constitute the basis of the DCA presented in subsection 3.3.

3.1. Duality and global optimality for d.c. optimization.

THEOREM 3.1. *Let \mathcal{P} and \mathcal{D} be the solution sets of problems (P) and (D), respectively. Then*

- (i) $x \in \mathcal{P}$ if and only if $\partial_\epsilon h(x) \subset \partial_\epsilon g(x) \quad \forall \epsilon > 0$.
- (ii) Dually, $y \in \mathcal{D}$ if and only if $\partial_\epsilon g^*(y) \subset \partial_\epsilon h^*(y) \quad \forall \epsilon > 0$.
- (iii) $\cup\{\partial h(x) : x \in \mathcal{P}\} \subset \mathcal{D} \subset \text{dom } h^*$.

The first inclusion becomes equality if g^ is subdifferentiable in \mathcal{D} (in particular, if $\mathcal{D} \subset \text{ri}(\text{dom } g^*)$ or if g^* is subdifferentiable in $\text{dom } h^*$). In this case $\mathcal{D} \subset (\text{dom } \partial g^* \cap \text{dom } \partial h^*)$.*

- (iv) $\cup\{\partial g^*(y) : y \in \mathcal{D}\} \subset \mathcal{P} \subset \text{dom } g$.

The first inclusion becomes equality if h is subdifferentiable in \mathcal{P} (in particular, if $\mathcal{P} \subset \text{ri}(\text{dom } h)$ or if h is subdifferentiable in $\text{dom } g$). In this case $\mathcal{P} \subset (\text{dom } \partial g \cap \text{dom } \partial h)$.

The relationship between primal and dual solutions: $\cup\{\partial h(x) : x \in \mathcal{P}\} \subset \mathcal{D}$ and $\cup\{\partial g^*(y) : y \in \mathcal{D}\} \subset \mathcal{P}$ is due to J. F. Toland [39] in the general context of the duality principle dealing with linear vector spaces in separating duality. A direct proof of the results (except Properties (i) and (ii)), based on the theory of subdifferential for convex functions is given in [7], [23], [27]. The properties (i) and (ii) have been first established by J. B. Hiriart-Urruty [9]. His proof (based on his earlier work concerning the behavior of the ϵ -directional derivative of a convex function as a function of the parameters ϵ) is quite complicated. The following proof of these properties is very simple and well suited to our d.c. duality framework [7], [23], [27]. In fact, they are nothing but a geometrical translation of the equality of the optimal value in the primal and dual d.c. programs (P) and (D).

Indeed, in virtue of the d.c. duality, $x^* \in \mathcal{P}$ if and only if $x^* \in \text{dom } g$ and

$$(6) \quad g(x^*) - h(x^*) \leq h^*(y) - g^*(y) \quad \forall y \in \text{dom } h^*,$$

i.e.,

$$(7) \quad g(x^*) + g^*(y) \leq h(x^*) + h^*(y) \quad \forall y \in \text{dom } h^*.$$

On the other hand, for $x^* \in \text{dom } h$, the property $\partial_\epsilon h(x^*) \subset \partial_\epsilon g(x^*) \quad \forall \epsilon > 0$ is, by definition, equivalent to

$$(8) \quad \forall \epsilon > 0, \langle x^*, y \rangle + \epsilon \geq h(x^*) + h^*(y) \Rightarrow \langle x^*, y \rangle + \epsilon \geq g(x^*) + g^*(y).$$

It is easy to see the equivalence between (7) and (8), and property (i) is thus proved. The global optimality condition in (i) is difficult to use for deriving solution methods to problem (P). The DCA which will be described in subsection 3.3 is based on local optimality conditions. The relations (ii) and (iv) indicate that solving the primal d.c. program (P) implies solving the dual d.c. program (D) and vice versa. It may be useful if one of them is easier to solve than the other.

3.2. Duality and local optimality conditions for d.c. optimization.

THEOREM 3.2. (i) *If x^* is a local minimizer of $g - h$, then $x^* \in \mathcal{P}_l$.*

(ii) *Let x^* be a critical point of $g - h$ and $y^* \in \partial g(x^*) \cap \partial h(x^*)$. Let U be a neighborhood of x^* such that $U \cap \text{dom } g \subset \text{dom } \partial h$. If for any $x \in U \cap \text{dom } g$ there is $y \in \partial h(x)$ such that $h^*(y) - g^*(y) \geq h^*(y^*) - g^*(y^*)$, then x^* is a local minimizer of $g - h$. More precisely,*

$$g(x) - h(x) \geq g(x^*) - h(x^*) \quad \forall x \in U \cap \text{dom } g.$$

Property (i) is well known [7], [9], [23], [39]. For completeness we give below a short proof for it. Property (ii) is new; it establishes an interesting sufficient condition (dealing with the d.c. duality) for local d.c. optimality.

Proof. (i) If x^* is a local minimizer of $g - h$, then there exists a neighborhood of x^* such that

$$g(x) - g(x^*) \geq h(x) - h(x^*) \quad \forall x \in U \cap \text{dom } g.$$

Hence for $y^* \in \partial h(x^*)$ we have $g(x) - g(x^*) \geq \langle x - x^*, y^* \rangle \quad \forall x \in U \cap \text{dom } g$. The convexity of g then implies that $y^* \in \partial g(x^*)$.

(ii) The condition $y^* \in \partial g(x^*) \cap \partial h(x^*)$ implies $g(x^*) + g^*(y^*) = \langle x^*, y^* \rangle = h(x^*) + h^*(y^*)$. Hence

$$(9) \quad g(x^*) - h(x^*) = h^*(y^*) - g^*(y^*).$$

For any $x \in U \cap \text{dom } g$, by assumption, there is $y \in \partial h(x)$ such that

$$(10) \quad h^*(y) - g^*(y) \geq h^*(y^*) - g^*(y^*).$$

On the other hand, we have $h(x) + h^*(y) = \langle x, y \rangle \leq g(x) + g^*(y)$. Hence,

$$(11) \quad g(x) - h(x) \geq h^*(y) - g^*(y).$$

Combining (9), (10), (11), we get $g(x) - h(x) \geq g(x^*) - h(x^*) \quad \forall x \in U \cap \text{dom } g$. □

A function $\theta \in \Gamma_o(X)$ is said to be polyhedral convex if [33]

$$\theta(x) = \max\{\langle a_i, x \rangle - \alpha_i : i = 1, \dots, m\} + \chi_C(x) \quad \forall x \in X,$$

where C is a nonempty polyhedral convex set in X .

Polyhedral d.c. optimization occurs when either g or h is polyhedral convex. This class of d.c. optimization problems, which is frequently encountered in practice, enjoys interesting properties (from both theoretical and practical viewpoints) concerning

local optimality and the convergence of the DCA, and has been extensively developed in [7], [29].

COROLLARY 3.3 (sufficient local optimality). *Let x^* be a point that admits a neighborhood U such that $\partial h(x) \cap \partial g(x^*) \neq \emptyset \quad \forall x \in U \cap \text{dom } g$. Then x^* is a local minimizer of $g - h$. More precisely, $g(x) - h(x) \geq g(x^*) - h(x^*) \quad \forall x \in U \cap \text{dom } g$.*

Proof. Let $x \in U \cap \text{dom } g$ and let $y \in \partial h(x) \cap \partial g(x^*)$. Since $y \in \partial h(x)$ we have $h(x) + h^*(y) = \langle x, y \rangle \leq g(x) + g^*(y)$. So $g(x) - h(x) \geq h^*(y) - g^*(y)$. Similarly, $y \in \partial g(x^*)$ implies that $g(x^*) + g^*(y) = \langle x^*, y \rangle \leq h(x^*) + h^*(y)$. Hence $h^*(y) - g^*(y) \geq g(x^*) - h(x^*)$. If $y^* \in \partial h(x^*) \cap \partial g(x^*)$, then $g(x^*) + g^*(y^*) = \langle x^*, y^* \rangle = h(x^*) + h^*(y^*)$. Hence $g(x^*) - h(x^*) = h^*(y^*) - g^*(y^*)$. The assumptions of Theorem 3.2 (ii) are fulfilled. Thus, the proof is complete. \square

Corollary 3.3 can be used to prove the following well-known result due to C. Michelot (in the case where g, h belonging to $\Gamma_o(X)$ are finite on the whole X) and generalized by the authors [7], [29] to the case of arbitrary g and h belonging to $\Gamma_o(X)$: the converse of property (i) of Theorem 3.2 in case h is polyhedral convex.

COROLLARY 3.4 (sufficient strict local optimality). *If $x^* \in \text{int}(\text{dom } h)$ verifies $\partial h(x^*) \subset \text{int}(\partial g(x^*))$, then x^* is a strict local minimizer of $g - h$.*

Proof. From the upper semicontinuity of the operator ∂h at $x^* \in \text{int}(\text{dom } h)$ [33] it follows that for any open set O containing $\partial h(x^*)$ there is a neighborhood U of x^* such that $\partial h(x) \subset O \quad \forall x \in U$.

Hence by letting $O = \text{int}(\partial g(x^*))$ and taking Corollary 3.3 into account, we have x^* is a local minimizer of $g - h$. But x^* is actually a strict local minimizer of $g - h$. Indeed, since $\partial h(x)$ is compact for $x \in V = U \cap \text{int}(\text{dom } h)$, we have $\forall x \in V \quad \exists \epsilon(x) > 0$ such that $\partial h(x) + \epsilon(x)B \subset O$ (B being the closed unit ball of the Euclidean norm).

Now let $x \in V \setminus \{x^*\}$ and $y \in \partial h(x)$. Then

$$\begin{aligned} g(x) - g(x^*) &\geq \left\langle x - x^*, y + \frac{\epsilon(x)}{\|x - x^*\|} (x - x^*) \right\rangle \\ &= \epsilon(x)\|x - x^*\| + \langle x - x^*, y \rangle \\ &\geq \epsilon(x)\|x - x^*\| + h(x) - h(x^*). \end{aligned}$$

The proof is complete. \square

It may happen that it is easier to solve locally the dual d.c. program (D) than the primal d.c. program (P). So it is useful to state results relative to the d.c. duality transportation of local minimizers. Paradoxically, such a result is more complicated than the d.c. duality transportation of global minimizers in Theorem 3.1.

COROLLARY 3.5 (d.c. duality transportation of a local minimizer). *Let $x^* \in \text{dom } \partial h$ be a local minimizer of $g - h$, and let $y^* \in \partial h(x^*)$ (i.e., $\partial h(x^*)$ is nonempty and x^* admit a neighborhood U such that $g(x) - h(x) \geq g(x^*) - h(x^*) \quad \forall x \in U \cap \text{dom } g$). If*

$$(12) \quad y^* \in \text{int}(\text{dom } g^*) \quad \text{and} \quad \partial g^*(y^*) \subset U$$

((12) holds if g^ is differentiable at y^*), then y^* is a local minimizer of $h^* - g^*$.*

Proof. According to (i) of Theorem 3.2 we have $y^* \in \partial h(x^*) \subset \partial g(x^*)$. So $x^* \in \partial g^*(y^*) \cap \partial h^*(y^*)$. Under the assumption (12) and the upper semicontinuity of ∂g^* , y^* admits a neighborhood $V \subset \text{int}(\text{dom } g^*)$ such that ([33]) $\partial g^*(V) \subset U$. More precisely, $\partial g^*(V) \subset (U \cap \text{dom } g)$ since we have [33] $\text{range } \partial g^* = \text{dom } \partial g$ and $\text{dom } \partial g \subset \text{dom } g$. Using the dual property (in the d.c. duality) in (ii) of Theorem 3.2

we deduce that y^* is a local minimizer of $h^* - g^*$. If g^* is differentiable at y^* , then $x^* = \partial g^*(y^*)$ and we have (12) [33]. \square

By the symmetry of the d.c. duality, Corollary 3.5 has its corresponding dual part.

Remark. This result improves an earlier result of J. F. Toland [39] where he assumed that g^* is differentiable on the whole dual space Y . In [7], [29] we have proved that this result remains true if g^* is only essentially differentiable.

3.3. Description of the DCA for general d.c. programs. For each fixed $x^* \in X$ we consider the problem

$$(S(x^*)) \quad \inf\{h^*(y) - g^*(y) : y \in \partial h(x^*)\},$$

which is equivalent to the convex maximization one

$$\inf\{\langle x^*, y \rangle - g^*(y) : y \in \partial h(x^*)\}.$$

Similarly, for each fixed $y^* \in Y$, for duality, we define the problem

$$(T(y^*)) \quad \inf\{g(x) - h(x) : x \in \partial g^*(y^*)\}.$$

This problem is equivalent to

$$\inf\{\langle x, y^* \rangle - h(x) : x \in \partial g^*(y^*)\}.$$

Let $S(x^*)$, $T(y^*)$ denote the solution sets of Problems $(S(x^*))$ and $(T(y^*))$, respectively.

The complete form of the DCA is based upon duality of d.c. optimization defined by (P) and (D). It allows approximating a point $(x^*, y^*) \in \mathcal{P}_l \times \mathcal{D}_l$. Given a point $x^o \in \text{dom } g$, the algorithm constructs two sequences $\{x^k\}$ and $\{y^k\}$ defined by

$$(13) \quad y^k \in S(x^k); \quad x^{k+1} \in T(y^k).$$

The complete DCA can be viewed as a sort of a decomposition approach of the primal and dual problems (P), (D). From a practical point of view, although problems $(S(x^k))$ and $(T(y^k))$ are simpler than (P), (D) (we work in $\partial h(x^k)$ and $\partial g^*(y^k)$ with convex maximization problems), they remain nonconvex programs and thus are still difficult (see subsection 3.5). In practice the following simplified form of the DCA is used:

• **Simplified form of the DCA:** The idea of the simplified DCA is quite natural: it constructs two sequences $\{x^k\}$ and $\{y^k\}$ (candidates to primal and dual solutions) which are easy to calculate and satisfy the following conditions:

- (i) The sequences $(g - h)(x^k)$ and $(h^* - g^*)(y^k)$ are decreasing.
- (ii) Every limit point x^* (resp. y^*) of the sequence $\{x^k\}$ (resp. $\{y^k\}$) is a critical point of $g - h$ (resp. $h^* - g^*$).

These conditions suggest constructing two sequences $\{x^k\}$ and $\{y^k\}$, starting from a given point $x^o \in \text{dom } g$, by setting

$$y^k \in \partial h(x^k); \quad x^{k+1} \in \partial g^*(y^k).$$

Interpretation of the simplified DCA: At each iteration k we do the following:

$$\begin{aligned} x^k \in \partial g^*(y^{k-1}) &\rightarrow y^k \in \partial h(x^k) \\ &= \operatorname{argmin}\{h^*(y) - [g^*(y^{k-1}) + \langle x^k, y - y^{k-1} \rangle] : y \in Y\}, \quad (D_k) \end{aligned}$$

$$y^k \in \partial h(x^k) \rightarrow x^{k+1} \in \partial g^*(y^k) = \operatorname{argmin}\{g(x) - [h(x^k) + \langle x - x^k, y^k \rangle] : x \in X\}. \quad (P_k)$$

Problem (P_k) is a convex program obtained from (P) by replacing h with its affine minorization defined by $y^k \in \partial h(x^k)$. Similarly, the convex problem (D_k) is obtained from (D) by using the affine minorization of g^* defined by $x^k \in \partial g^*(y^{k-1})$. Here we can see a complete symmetry between problems (P_k) and (D_k) , and between the sequences $\{x^k\}$ and $\{y^k\}$ relative to the duality of d.c. optimization. The two forms of the DCA are identical if g^* and h are essentially differentiable.

• **Well definiteness of the DCA:** The DCA is well defined if one can construct two sequences $\{x^k\}$ and $\{y^k\}$ as above from an arbitrary initial point $x^o \in \operatorname{dom} g$. We have $x^{k+1} \in \partial g^*(y^k)$ and $y^k \in \partial h(x^k) \forall k \geq 0$. So $\{x^k\} \subset \operatorname{range} \partial g^* = \operatorname{dom} \partial g$ and $\{y^k\} \subset \operatorname{range} \partial h = \operatorname{dom} \partial h^*$. Then the following lemma is clear.

LEMMA 3.6. *Sequences $\{x^k\}$, $\{y^k\}$ in the DCA are well defined if and only if*

$$\operatorname{dom} \partial g \subset \operatorname{dom} \partial h \text{ and } \operatorname{dom} \partial h^* \subset \operatorname{dom} \partial g^*.$$

Since for $\varphi \in \Gamma_o(X)$ we have $\operatorname{ri}(\operatorname{dom} \varphi) \subset \operatorname{dom} \partial \varphi \subset \operatorname{dom} \varphi$ ([33]) ($\operatorname{ri}(\operatorname{dom} \varphi)$ stands for the relative interior of $\operatorname{dom} \varphi$) we can say, under the essential assumption (3), that the DCA is, in general, well defined.

Remark. A d.c. function f has infinitely many d.c. decompositions. For example, if $f = g - h$, then $f = (g + \theta) - (h + \theta)$ for every $\theta \in \Gamma_o(X)$ finite on the whole X . It is clear that the primal d.c. programs (P) corresponding to the two d.c. decompositions of the objective function f are identical. But their dual programs are quite different and so is the DCA relative to these d.c. decompositions. In other words, there are as many DCAs as there are d.c. decompositions of the objective function f . It is useful to find a suitable d.c. decomposition of f since it may have an important influence on the efficiency of the DCA for its solution. This question is intimately related to the regularization techniques in d.c. programming [7], [23], [27].

3.4. Convergence of the DCA for general d.c. programs. Let ρ_i and ρ_i^* , ($i = 1, 2$) be real nonnegative numbers such that $0 \leq \rho_i < \rho(f_i)$ (resp. $0 \leq \rho_i^* < \rho(f_i^*)$) where $\rho_i = 0$ (resp. $\rho_i^* = 0$) if $\rho(f_i) = 0$ (resp. $\rho(f_i^*) = 0$) and ρ_i (resp. ρ_i^*) may take the value $\rho(f_i)$ (resp. $\rho(f_i^*)$) if it is attained. We next set $f_1 = g$ and $f_2 = h$. Also let $dx^k := x^{k+1} - x^k$ and $dy^k := y^{k+1} - y^k$.

The basic convergence theorem of the DCA for general d.c. programming will be stated below. Its proof is very technical and long. We will present it in the appendix and refer the reader to our previous works [7], [27] for more details.

THEOREM 3.7. *Suppose that the sequences $\{x^k\}$ and $\{y^k\}$ are defined by the simplified DCA. Then we have*

$$(i) \quad (g - h)(x^{k+1}) \leq (h^* - g^*)(y^k) - \max \left\{ \frac{\rho_2}{2} \|dx^k\|^2, \frac{\rho_2^*}{2} \|dy^k\|^2 \right\} \leq (g - h)(x^k) \\ - \max \left\{ \frac{\rho_1 + \rho_2}{2} \|dx^k\|^2, \frac{\rho_1^*}{2} \|dy^{k-1}\|^2 + \frac{\rho_2}{2} \|dx^k\|^2, \frac{\rho_1^*}{2} \|dy^{k-1}\|^2 + \frac{\rho_2^*}{2} \|dy^k\|^2 \right\}.$$

The equality $(g - h)(x^{k+1}) = (g - h)(x^k)$ holds if and only if $x^k \in \partial g^*(y^k)$, $y^k \in \partial h(x^{k+1})$ and $(\rho_1 + \rho_2)dx^k = \rho_1^*dy^{k-1} = \rho_2^*dy^k = 0$. In this case

• $(g - h)(x^{k+1}) = (h^* - g^*)(y^k)$ and x^k, x^{k+1} are the critical points of $g - h$ satisfying $y^k \in (\partial g(x^k) \cap \partial h(x^k))$ and $y^k \in (\partial g(x^{k+1}) \cap \partial h(x^{k+1}))$,

- y^k is a critical point of $h^* - g^*$ satisfying $[x^k, x^{k+1}] \subset ((\partial g^*(y^k) \cap \partial h^*(y^k)))$,
 - $x^{k+1} = x^k$ if $\rho(g) + \rho(h) > 0$, $y^k = y^{k-1}$ if $\rho(g^*) > 0$ and $y^k = y^{k+1}$ if $\rho(h^*) > 0$.
- (ii) Similarly, for the dual problem we have

$$(h^* - g^*)(y^{k+1}) \leq (g - h)(x^{k+1}) - \max \left\{ \frac{\rho_1}{2} \|dx^{k+1}\|^2, \frac{\rho_1^*}{2} \|dy^k\|^2 \right\} \leq (h^* - g^*)(y^k) - \max \left\{ \frac{\rho_1}{2} \|dx^{k+1}\|^2 + \frac{\rho_2}{2} \|dx^k\|^2, \frac{\rho_1^*}{2} \|dy^k\|^2 + \frac{\rho_2}{2} \|dx^k\|^2, \frac{\rho_1^* + \rho_2^*}{2} \|dy^k\|^2 \right\}.$$

The equality $(h^* - g^*)(y^{k+1}) = (h^* - g^*)(y^k)$ holds if and only if $x^{k+1} \in \partial g^*(y^{k+1})$, $y^k \in \partial h(x^{k+1})$ and $(\rho_1^* + \rho_2^*)dy^k = \rho_2 dx^k = \rho_1 dx^{k+1} = 0$. In this case

- $(h^* - g^*)(y^{k+1}) = (g - h)(x^{k+1})$ and y^k, y^{k+1} are the critical points of $h^* - g^*$ satisfying $x^{k+1} \in (\partial g^*(y^k) \cap \partial h^*(y^k))$ and $x^{k+1} \in (\partial g^*(y^{k+1}) \cap \partial h^*(y^{k+1}))$,
- x^{k+1} is a critical point of $g - h$ satisfying $[y^k, y^{k+1}] \subset ((\partial g(x^{k+1}) \cap \partial h(x^{k+1})))$,
- $y^{k+1} = y^k$ if $\rho(g^*) + \rho(h^*) > 0$, $x^{k+1} = x^k$ if $\rho(h) > 0$ and $x^{k+1} = x^{k+2}$ if $\rho(g) > 0$.

(iii) If α is finite, then the decreasing sequences $\{(g - h)(x^k)\}$ and $\{(h^* - g^*)(y^k)\}$ converge to the same limit $\beta \geq \alpha$, i.e., $\lim_{k \rightarrow +\infty} (g - h)(x^k) = \lim_{k \rightarrow +\infty} (h^* - g^*)(y^k) = \beta$. If $\rho(g) + \rho(h) > 0$ (resp. $\rho(g^*) + \rho(h^*) > 0$), then $\lim_{k \rightarrow +\infty} \{x^{k+1} - x^k\} = 0$ (resp. $\lim_{k \rightarrow +\infty} \{y^{k+1} - y^k\} = 0$). Moreover, $\lim_{k \rightarrow +\infty} \{g(x^k) + g^*(y^k) - \langle x^k, y^k \rangle\} = 0 = \lim_{k \rightarrow +\infty} \{h(x^{k+1}) + h^*(y^k) - \langle x^{k+1}, y^k \rangle\}$.

(iv) If α is finite and the sequences $\{x^k\}$ and $\{y^k\}$ are bounded, then for every limit x^* of $\{x^k\}$ (resp. y^* of $\{y^k\}$) there exists a cluster point y^* of $\{y^k\}$ (resp. x^* of $\{x^k\}$) such that

- $(x^*, y^*) \in [\partial g^*(y^*) \cap \partial h^*(y^*)] \times [\partial g(x^*) \cap \partial h(x^*)]$ and $(g - h)(x^*) = (h^* - g^*)(y^*) = \beta$,
- $\lim_{k \rightarrow +\infty} \{g(x^k) + g^*(y^k)\} = \lim_{k \rightarrow +\infty} \langle x^k, y^k \rangle$.

Comments on Theorem 3.7. (i) Properties (i) and (ii) prove that the DCA is a descent method for both primal and dual programs. The DCA provides critical points for (P) and (D) after finitely many operations if there is no strict decrease of the primal (or dual) objective function.

(ii) If C and D are convex sets such that $\{x^k\} \subset C$ and $\{y^k\} \subset D$, then Theorem 3.7 remains valid if we replace $\rho(f_i)$ by $\rho(f_i, C)$ and $\rho(f_i^*)$ by $\rho(f_i^*, D)$ for $i = 1, 2$. By this way we may improve the results in the theorem.

(iii) In (ii) of Theorem 3.7, the convergence of the whole sequence $\{x^k\}$ (resp. $\{y^k\}$) can be ensured under the following conditions [18], [20]:

- $\{x^k\}$ is bounded;
- The set of limit points of $\{x^k\}$ is finite;
- $\lim_{k \rightarrow +\infty} \|x^{k+1} - x^k\| = 0$.

(iv) The only difference between the simplified DCA and the complete DCA lies on the choice of y^k in $\partial h(x^k)$ and x^{k+1} in $\partial g^*(y^k)$. The convergence result of the complete DCA is thus improved: in Theorem 3.7, the nonemptiness of a subdifferential intersection is replaced by a subdifferential inclusion [7], [23], [27]. In other words, the complete DCA permits to obtain a couple of elements $(x^*, y^*) \in \mathcal{P}_l \times \mathcal{D}_l$. In practice the simplified DCA usually yields a local minimizer which is also global [7].

(v) In general, the qualities (robustness, stability, rate of convergence, and globality of sought solutions) of the DCA, in both complete and simplified forms, depend upon the d.c. decomposition of the function f . Theorem 3.7 shows how strong convexity of d.c. components in primal and dual problems can influence the DCA. To

make the d.c. components (of the primal objective function $f = g - h$) strongly convex we usually apply the following process

$$f = g - h = \left(g + \frac{\lambda}{2} \|\cdot\|^2 \right) - \left(h + \frac{\lambda}{2} \|\cdot\|^2 \right).$$

In this case the d.c. components in the dual problem will be continuously differentiable. Similarly inf-convolution of g and h with $\lambda/2 \|\cdot\|^2$ will make the d.c. components (in dual problems) strongly convex and the d.c. components of the primal objective function continuously differentiable. For a detailed study of regularization techniques in the d.c. optimization, see [7], [23], [27].

3.5. How to restart the simplified DCA for obtaining x^* such that $\partial h(x^*) \subset \partial g(x^*)$. As mentioned above, the complete DCA theoretically provides a x^* such that $\partial h(x^*) \subset \partial g(x^*)$. In practice, except for the cases where the convex maximization problems $(S(x^k))$ and $(T(y^k))$ are easy to solve, one generally uses the simplified DCA. It is worth noting that if the simplified DCA terminates at some point x^* for which $\partial h(x^*)$ is not contained in $\partial g(x^*)$, then one can reduce the objective function value by restarting it from a new initial point $x^o = x^*$ with $y^o \in \partial h(x^o)$ such that $y^o \notin \partial g(x^o)$. In fact, since

$$g(x^1) + g^*(y^o) = \langle x^1, y^o \rangle \leq h(x^1) - h(x^o) + \langle x^o, y^o \rangle$$

and $\langle x^o, y^o \rangle < g(x^o) + g^*(y^o)$ because $y^o \notin \partial g(x^o)$, we have

$$g(x^1) + g^*(y^o) < h(x^1) - h(x^o) + g(x^o) + g^*(y^o).$$

Hence,

$$g(x^1) - h(x^1) < g(x^o) - h(x^o).$$

We have given the main concept of the d.c. programming and the DCA. In the next section we shall apply this technique to the trust-region subproblem.

4. Solving trust-region subproblem (Q_1) by the DCA.

4.1. DCA for solving Problem (Q_1) . We describe below the DCA for solving (Q_1) . For this we must point out d.c. decompositions of the objective function in (Q_1) .

$$(Q_1) \quad \alpha = \inf \left\{ \frac{1}{2} x^T A x + b^T x + \chi_E(x) : x \in \mathbb{R}^n \right\},$$

where $E = \{x \in \mathbb{R}^n : \|x\| \leq r\}$ and χ_E stands for its indicator function. The following d.c. decomposition seems to be the most natural

$$f(x) = \frac{1}{2} x^T A x + b^T x + \chi_E(x) = g_1(x) - h_1(x)$$

with

$$(14) \quad g_1(x) = \frac{1}{2} x^T (A + \rho I) x + x^T b + \chi_E(x), \quad h_1(x) = \frac{1}{2} \rho x^T x,$$

and ρ is a positive real number such that $A + \rho I$ is positive semidefinite.

For this case in the simplified DCA $y^k = \rho x^k$ and x^{k+1} is a solution of the problem

$$(15) \quad \min \left\{ \frac{1}{2} x^T (A + \rho I) x + x^T (b - y^k) : \|x\| \leq r \right\}.$$

Thus, at each iteration, the algorithm requires solving a convex quadratic program and so is expensive, especially in large-scale problems. In practice the algorithm with this d.c. decomposition is unstable ([7]). Note that if A is positive semidefinite then DCA with this decomposition reduces to the well-known proximal point algorithm [14], [34].

From the computational viewpoint the following decomposition, in our opinion, is the most efficient: $f(x) = g(x) - h(x)$ with

$$(16) \quad g(x) = \frac{1}{2} \rho \|x\|^2 + b^T x + \chi_E(x), \quad h(x) = \frac{1}{2} x^T (\rho I - A) x,$$

and ρ is a positive number such that $\rho I - A$ is positive semidefinite.

Clearly, $g, h \in \Gamma_o(\mathbb{R}^n)$ and thus (Q_1) takes the form

$$(17) \quad (P) \quad \min \{ g(x) - h(x) : x \in \mathbb{R} \}.$$

The simplified DCA applied to this case can be formulated as follows. Let $x^o \in \mathbb{R}^n$ and $k \geq 0$. Further, let $y^k = (\rho I - A)x^k$ and x^{k+1} be a solution of the problem

$$(18) \quad \min \left\{ \frac{\rho}{2} \|x\|^2 + x^T (b - y^k) + \chi_E(x) : x \in \mathbb{R}^n \right\}.$$

Clearly,

$$(18) \Leftrightarrow \min_{x \in E} \left\| x - \frac{y^k - b}{\rho} \right\|^2.$$

Thus, x^{k+1} is, in fact, the projection of $(y^k - b)/\rho$ onto E , i.e.,

$$(19) \quad x^{k+1} = P_E \left(x^k - \frac{1}{\rho} (Ax^k + b) \right).$$

Summing up, we state the chosen DCA as follows:

DCA :

Let $x^o \in \mathbb{R}^n$ and $k \geq 0$

a) If $\|(\rho I - A)x^k - b\| \leq \rho r$ take $x^{k+1} = \frac{(\rho I - A)x^k - b}{\rho}$.

b) Otherwise, take $x^{k+1} = r \frac{(\rho I - A)x^k - b}{\|(\rho I - A)x^k - b\|}$.

If $\|x^{k+1} - x^k\| \leq \epsilon$, terminate. Otherwise, increase k by 1 and return to a).

Remark. (i) In case $b = 0$, except if A is positive semidefinite (for which zero is a solution), we choose x^o such that $y^o = (\rho I - A)x^o \neq 0$, since $y^o = 0$ implies $x^k = 0 \forall k \geq 1$.

(ii) We have presented the two d.c. decompositions (14), (16) (of the objective function f) which give rise to quite different DCA. These algorithms treat indifferently the normal and hard cases of (Q_1) . However, the DCA corresponding to the decomposition (16) requires only matrix–vector products since the solution (19) is explicit. This DCA seems to be more stable and more efficient than the former. The following convergence results are a consequence of Theorem 3.7.

THEOREM 4.1. (i) $f(x^{k+1}) \leq f(x^k) - 1/2(\rho + \lambda)\|x^{k+1} - x^k\|^2$, where λ is the smallest eigenvalue of $(\rho I - A)$.

(ii) $f(x^k) \searrow \alpha' \geq \alpha$ and $\lim_{k \rightarrow \infty} \|x^{k+1} - x^k\| = 0$.

(iii) Every limit point x^* of the sequence $\{x^k\}$ is a Kuhn–Tucker point; i.e., there exists $\lambda^* \geq 0$ such that $(A + \lambda^* I)x^* = -b$, $\lambda^*(\|x^*\| - r) = 0$ and $\|x^*\| \leq r$.

Proof. For the d.c. decomposition (16) of f , we have $\rho(g, E) = \rho$, $\rho(h) = \rho - \lambda_n$. Since the sequence $\{x^k\}$ is contained in E and the supremum $\rho(g, E)$ in (5) is attained, according to (i) of Theorem 3.7 and (i) of comments on Theorem 3.7, we get

$$f(x^{k+1}) \leq f(x^k) - \frac{1}{2}(2\rho - \lambda_n)\|x^{k+1} - x^k\|^2.$$

Property (ii) then follows from (iii) of the above theorem.

Now let x^* be a limit point of $\{x^k\}$. Since the sequences $\{x^k\}$ and $\{y^k\}$ are bounded, Property (iv) of Theorem 3.7 then implies

$$(20) \quad \partial h(x^*) = \nabla h(x^*) \subset \partial g(x^*).$$

We have $\nabla h(x^*) = (\rho I - A)x^*$, $\partial g(x^*) = \rho x^* + b + \partial \chi_E(x^*)$ with $\partial \chi_E(x^*) = \{0\}$ if $\|x^*\| < r$, $\{\lambda x^* : \lambda \geq 0\}$, otherwise. So the relation (20) is equivalent to

$$(A + \lambda^* I)x^* = -b, \lambda^*(\|x^*\| - r) = 0, \text{ and } \|x^*\| \leq r$$

for some nonnegative real λ^* . The proof is complete. □

4.2. Using the DCA for globally solving (Q_1) . Let x^* be a point produced by the DCA (Theorem 4.1). It is clear that if A is positive semidefinite, then (P) in (17) is a “false” d.c. optimization problem (i.e., it is a convex one in reality) and the condition (iii) is actually a characterization of (Q_1) . In other words, the DCA converges to a solution of (Q_1) in this case. In general, using Theorem 2.1 we can calculate λ^* and check the global optimality. Indeed, from Theorem 2.1 it follows that

$$\lambda^* = \frac{-\langle x^*, Ax^* \rangle - \langle x^*, b \rangle}{r^2}.$$

So, checking the global optimality of x^* can be done, for example, by applying the inexpensive algorithm IRLM to compute the smallest eigenvalue λ_1 of A and a corresponding eigenvector u :

$$Au = \lambda_1 u.$$

If $\lambda^* + \lambda_1 \geq 0$, then x^* is a global solution to (Q_1) . Otherwise,

$$(21) \quad u^T(A + \lambda^* I)u = (\lambda^* + \lambda_1)u^T u < 0.$$

In this case we will indicate below how to restart the DCA, say, how to find \bar{x} such that $\|\bar{x}\| \leq r$ and $f(\bar{x}) < f(x^*)$. First let us mention the following property.

LEMMA 4.2. *If $f(x) = 1/2\langle x, Ax \rangle + \langle b, x \rangle$, with A being symmetric, and $(A + \mu I)y = -b$, then*

$$f(x) = f(y) - \frac{\mu}{2}(\|x\|^2 - \|y\|^2) + \frac{1}{2}\langle x - y, (A + \mu I)(x - y) \rangle \forall x.$$

Proof. We have

$$\begin{aligned} f(x) &= f(y) + \langle x - y, Ay + b \rangle + \frac{1}{2}\langle x - y, A(x - y) \rangle \\ &= f(y) - \mu\langle x - y, y \rangle + \frac{1}{2}\langle x - y, (A + \mu I)(x - y) \rangle - \frac{\mu}{2}\langle x - y, x - y \rangle. \end{aligned}$$

Since $\mu/2\langle x - y, x - y \rangle = \mu/2(\|x\|^2 + \|y\|^2) - \mu\langle x, y \rangle$, we get

$$-\frac{\mu}{2}\langle x - y, x - y \rangle - \mu\langle x - y, y \rangle = -\frac{\mu}{2}(\|x\|^2 + \|y\|^2) + \mu\|y\|^2 = -\frac{\mu}{2}(\|x\|^2 - \|y\|^2).$$

The proof is complete. \square

How to compute the initial point for restarting the DCA. a) If $\langle b, x^* \rangle > 0$, then taking $\bar{x} = -x^*$ we have $f(\bar{x}) < f(x^*)$. Indeed, in virtue of Lemma 4.2,

$$(22) \quad f(x) = f(x^*) - \frac{\lambda^*}{2}(\|x\|^2 - \|x^*\|^2) + \frac{1}{2}\langle x - x^*, (A + \lambda^* I)(x - x^*) \rangle.$$

This implies

$$f(\bar{x}) = f(x^*) + 2\langle (A + \lambda^* I)x^*, x^* \rangle = f(x^*) - 2\langle b, x^* \rangle < f(x^*).$$

So we can restart the DCA from the initial point \bar{x} .

b) If $\langle b, x^* \rangle \leq 0$, then we distinguish two cases:

(b.1). If either $\|x^*\| < r$ or $\|x^*\| = r$ and $u^T x^* \neq 0$, then we take $\bar{x} = x^* + \gamma u$, where $\gamma \neq 0$ is chosen such that $\|\bar{x}\| = r$. More precisely, γ is a nontrivial solution of the following equation:

$$\|u\|^2 \gamma^2 + 2u^T x^* \gamma + \|x^*\|^2 - r^2 = 0,$$

which is given explicitly by

$$(23) \quad \gamma = \begin{cases} -2(u^T x^*)/\|u\|^2 & \text{if } \|x^*\| = r, \\ \left(-(u^T x^*) \pm \sqrt{\Delta} \right) / \|u\|^2 & \text{if } \|x^*\| < r, \end{cases}$$

where $\Delta = (u^T x^*)^2 - \|u\|^2(\|x^*\|^2 - r^2)$.

So we can choose \bar{x} as the initial point for restarting the DCA, since

$$f(\bar{x}) = f(x^*) + \frac{\gamma^2}{2}\langle u, (A + \lambda^* I)u \rangle < f(x^*).$$

Note that in (23) the greater γ^2 is, the smaller $f(\bar{x})$ becomes. Then the better value of γ is given by $\gamma = (-(u^T x^*) + \sqrt{\Delta})/\|u\|^2$ if $u^T x^* < 0$, $(-(u^T x^*) - \sqrt{\Delta})/\|u\|^2$ otherwise.

(b.2). If $\|x^*\| = r$, and $u^T x^* = 0$, then we find a new vector $v = u + \tau x^*$ with $\tau < 0$ such that $v^T (A + \lambda^* I)v < 0$ and $v^T x^* \neq 0$. Return to (b.1) with v instead of u .

To prove the existence of such a vector we will proceed as follows: we have $v^T x^* = u^T x^* + \tau \|x^*\|^2 = \tau \|x^*\|^2$. So $v^T x^* < 0$ if $\tau < 0$. Since

$$\begin{aligned} \langle v, (A + \lambda^* I)v \rangle &= \tau^2 \langle x^*, (A + \lambda^* I)x^* \rangle + 2\tau \langle u, (A + \lambda^* I)x^* \rangle + \langle u, (A + \lambda^* I)u \rangle \\ &= -\tau^2 \langle b, x^* \rangle - 2\tau \langle b, u \rangle + \langle u, (A + \lambda^* I)u \rangle \end{aligned}$$

we easily deduce the following properties:

- If $b^T x^* = 0$ and $b^T u \leq 0$, then $v^T (A + \lambda^* I)v < 0$ for all $\tau < 0$. If $b^T x^* = 0$ and $b^T u > 0$, then $v^T (A + \lambda^* I)v < 0$ for $0 > \tau > u^T (A + \lambda^* I)u / b^T u$.

- If $b^T x^* < 0$, then $v^T (A + \lambda^* I)v < 0$ for $\tau_1 < \tau < 0$, where τ_1 is the smallest root of the second degree polynomial of τ , $-b^T x^* \tau^2 - 2b^T u \tau + u^T (A + \lambda^* I)u$, i.e., $\tau_1 = [b^T u + ((b^T u)^2 + b^T x^* u^T (A + \lambda^* I)u)^{1/2}] / b^T x^*$.

Summing up, we conclude that if a Kuhn–Tucker point x^* produced by the DCA is not a global solution to (Q_1) then another point \bar{x} (such that $\|\bar{x}\| \leq r$ and $f(\bar{x}) < f(x^*)$) is already available for restarting the DCA.

Remark. In case (b.2) the optimal choice for τ is a bit complicated because it deals with roots of a third degree polynomial of τ . Finally, we state here our global algorithm for solving (Q_1) in the nonconvex case.

Algorithm GDCA: (globally solving (Q_1)).

Initialization. Compute λ_1 and u by the IRLM.

 Compute an approximation $\tilde{\lambda}_n$ to λ_n by the IRLM.

 Let $\rho := \tilde{\lambda}_n + 0.1$, $\bar{x} \in C$, *stop* := *false*.

While *stop* = *false* **do**

 1. Apply the DCA from starting point $x^o := \bar{x}$ to obtain x^* .

 2. Let $\lambda^* = (-\langle x^*, Ax^* \rangle - \langle x^*, b \rangle) / r^2$.

If $\lambda^* \geq -\lambda_1$ **then** *stop* := *true*, x^* is a global solution of (Q_1)

else compute \bar{x} such that $f(\bar{x}) < f(x^*)$ and return to 1.

end while

4.3. Relation between the DCA and the Goldstein–Levitin–Polyak gradient projection algorithm for solving (Q_1) in the convex case. In case the matrix A in (Q_1) is positive semidefinite, the Goldstein–Levitin–Polyak gradient projection algorithm applied to (Q_1) looks as follows:

$$(24) \quad x^{k+1} = P_E(x^k - \rho^{-1}(Ax^k + b)) = F(x^k).$$

The following well-known convergence result can be proved by using classical arguments of fixed point theory (B. Polyak [31]).

PROPOSITION 4.3. *Let A be positive semidefinite and $\rho > 0$. Then*

(i) $\|F(x) - F(x')\| \leq \max\{|1 - \rho^{-1}\lambda_n|, |1 - \rho^{-1}\lambda_1|\} \|x - x'\|$.

(ii) *If $\rho > \lambda_n/2$, then every limit point of $\{x^k\}$ is a solution of (Q_1) . Moreover, if the solution set of (Q_1) is finite, then the whole sequence $\{x^k\}$ converges to a solution of (Q_1) .*

(iii) *If A is positive definite and $0 < \rho^{-1} < 2\lambda_n^{-1}$, then F is a contraction with constant $q(\rho) < 1$. Moreover, $\bar{\rho} = (\lambda_n + \lambda_1)/2$ is the best choice which minimizes $q(\rho)$ over $]\lambda_n/2, +\infty[$:*

$$q(\bar{\rho}) = \frac{\lambda_n - \lambda_1}{\lambda_n + \lambda_1}.$$

Remark. (i) The Goldstein–Levitin–Polyak gradient projection can be considered as a special case of the DCA applied to (Q_1) .

(ii) In the DCA ρ must be greater or equal to λ_n , while Proposition 4.3 requires only that $\rho > \lambda_n/2$.

(iii) The assertion (ii) of Proposition 4.3 is a direct consequence of Theorem 4.1.

(iv) Actually, the whole sequence $\{x^k\}$ converges to a solution of (Q_1) if this problem admits a solution. The proof of this result is based on the theory of maximal monotone operators.

4.4. Optimal choice of the regularization parameter ρ . In practice, the convergence rate of the DCA depends on the value ρ . Except for the case where the matrix A is positive definite for which the “theoretically optimal” value $(\lambda_1 + \lambda_n)/2$ (that is not “practically optimal,” however, as shown in Table 5) of ρ has been pointed out in Proposition 4.3, the general optimal choice of ρ is not easy. Numerical experiments proved that, in general, the closer to λ_n the positive parameter ρ is, the better the DCA converges. Such good parameters ρ belong to $\{\lambda > 0 : \lambda \in]\rho^*, \rho^* + \varepsilon[\}$, where $\varepsilon > 0$ is sufficiently small and $\rho^* = \lambda_n$ if $\lambda_n > 0$, 0 otherwise. In practice, such a ρ can be computed by using the quite inexpensive IRLM due to D.C. Sorensen [37]. We need only a low accuracy for an approximation $\tilde{\lambda}_n$ given by the IRLM of the largest eigenvalue λ_n of A . In other words, we are satisfied with an approximation $\tilde{\lambda}_n$ such that $\tilde{\lambda}_n \leq \lambda_n \leq \tilde{\lambda}_n + \Delta\tilde{\lambda}_n$, where the variation $\Delta\tilde{\lambda}_n$ is given by: if $|\tilde{\lambda}_n| = \alpha 10^k$ with $\alpha \in [0, 10[$ and k integer, then $\Delta\tilde{\lambda}_n = 2 \cdot 10^{k-1}$ when $\alpha \geq 5$, 10^{k-1} when $0 < \alpha < 5$, 0 when $\alpha = 0$.

Such an approximation $\tilde{\lambda}_n$ can be provided by the IRLM with the tolerance equal to 0.1 [37] at little cost. A good practical choice of ρ is then $\max\{\tilde{\lambda}_n + \Delta\tilde{\lambda}_n, 10^{-3}\}$.

5. Numerical experiments. In this section, we present some computational tests on the performance of the DCA for different sets of test problems. Our experiments are composed of five sets of tests. In the first set we compare the DCA with safeguarding [17] and dichotomy [7], [24] algorithms. The algorithms have been coded in PASCAL under Unix system and run on Sun SPARC-2 station with double precision. In the second set, we provide a comparison between the DCA and the approach proposed by S. A. Santos and D. C. Sorensen [35]. The third set reports the performance of the DCA and the gradient projection algorithm in the convex case. Finally, in the fourth and fifth sets we study the sensitivity of the DCA, respectively, for different choices of ρ and several sizes of the radius. For the last four sets the algorithms have been coded in FORTRAN 77 and run on Sun SPARC-10 station with double precision.

The stopping criterion of the DCA was actually $er \leq \epsilon$, where

$$(25) \quad er = \begin{cases} \|x^{k+1} - x^k\|^2 / \|x^k\|^2 & \text{if } \|x^k\| > 1, \\ \|x^{k+1} - x^k\|^2 & \text{otherwise,} \end{cases}$$

with $\epsilon = 10^{-6}$. The initial point of the DCA was always chosen as $x_i^0 = r/\sqrt{n}$, $i = 1, \dots, n$.

5.1. Comparison with safeguarding and dichotomy algorithms. In the first experiment we solved 150 problems (five problems per dimension n) whose data was randomly generated in a similar way as described in [17]. The matrix A is of the form $A = UDU^T$ for some orthogonal matrix U and a diagonal matrix D . Let $b = Ug$ for some vector g . The orthogonal matrix U was of the form $Q_1Q_2Q_3$ where $Q_j = I - 2w_jw_j^T/\|w_j\|^2$, $j = 1, 2, 3$ and the components w_j were random numbers in $(-1, 1)$. The elements of g were chosen as random numbers in $(-1, 1)$ while the diagonal elements of matrix D are random numbers in $(-5, 5)$. This choice

leads to the normal case. The hard case is generated by setting to zero all the components of g corresponding to the smallest element of D (that is λ_1) and then by taking r greater than $\|(A - \lambda_1 I)^+ b\|$ (here “+” denotes the Moore–Penrose generalized inverse), i.e., $r > \|d\|$ where $d_i = g_i / (D_{ii} - \lambda_1)$ if $D_{ii} \neq \lambda_1$, 0 otherwise. A positive definite case (resp. positive semidefinite) was obtained by replacing $D_{ii} < 0$ by $|D_{ii}|$ (resp. 0). In the normal case, r is randomly chosen in $(1, 100)$. The values of ρ (in the DCA) were chosen as $0.25\|A\|_1$ or $0.35\|A\|_1$. For the safeguarding algorithm (Sfg) we took $\sigma_1 = 10^{-4}$, $\sigma_2 = 10^{-2}$. For the dichotomy algorithm (Dct) we chose $\varepsilon_3 = 0.001(\lambda_L^k + \lambda_U^k)$. The optimal values provided by the three algorithms are, to within 10^{-3} , equal.

Tables 1 and 2 indicate, respectively, the average iteration (IT) and the average CPU time in seconds (on Sun SPARC-2) of three algorithms in nonconvex and convex cases.

TABLE 1

Comparison with dichotomy and safeguarding algorithms, A is indefinite, the normal case (left) and the hard case (right).

n	DCA		Dct		Sfg	
	IT	time	IT	time	IT	time
100	53.2	1.3	6	2.5	7	3.2
200	113	11.3	6.2	28.4	7.4	32.6
250	102	13.5	5	42.1	7	60.6
300	75	15.5	6	85.6	7.4	95.3
350	73	20.4	5.2	122.2	7	161.6
400	59	23.6	5.2	172.0	6	189.7
450	77	80.6	4.4	250.2	6	270.2
500	67	90	6	497	6.4	502

n	DCA		Dct		Sfg	
	IT	time	IT	time	IT	time
100	83	2.2	8.6	4.7	10	5.5
200	222	22.2	9.2	30.3	11.2	37.9
250	180	37.5	9	62.1	11	75.4
300	178	45.0	7.2	75.0	12	130.4
350	190	74.0	6.4	129.2	12	244.9
400	279	122	5.6	181.4	11.4	355.1
450	187	139	6.6	316.6	11.2	539.6
500	234	144	7.6	475.7	11	684.9

TABLE 2

Comparison with dichotomy and safeguarding algorithms, A is positive definite (left) and A is positive semidefinite (right).

n	DCA		Dct		Sfg	
	IT	time	IT	time	IT	time
100	43	1.2	2.4	1.45	2.4	1.45
150	32	2.0	3	7.0	4.4	8.5
200	21	2.25	3.4	11.9	4	14.5
250	23	3.8	3.4	21.5	4	24.7
300	22	4.2	3	38.7	4	53.42
350	22	6.8	3	50.2	4	84.7
400	16	7.5	4.2	130	4	126.5

n	DCA		Dct		Sfg	
	IT	time	IT	time	IT	time
100	10	0.32	3	2.0	3	2.0
150	35	2.5	3	7.6	3	8.0
200	23	2.3	3.5	12.9	4	15.5
250	22	3.7	3	25.0	3	25.0
300	45	13.0	3	52.2	3.4	73.4
350	165	48.8	2	56.8	5	85.2
400	17	7.2	3	118	3	120

5.2. Comparison with the new matrix-free algorithm. In the second experiment we solved two different families of 112 problems by the DCA and the new matrix-free algorithm (NMFA). We used the code in MATLAB of S. A. Santos and D. C. Sorensen [35] to generate the data. First, the matrix A takes the form $L - 5I$, where L is the standard 2-D discrete Laplacian on the unit square based upon a 5-point stencil with equally-spaced meshpoints. The shift of $-5I$ was introduced to make A indefinite. The vector b was randomly generated with entries uniformly distributed on $(0, 1)$. The hard case is generated by performing the operation $b \leftarrow b - q_1(q_1^T b)$ to orthogonalize the vector b against q_1 , the eigenvector of A corresponding to the smallest eigenvalue λ_1 . The radius was always equal to 100.

In the second family of problems, the matrix A is of the form UDU^T and generated as in subsection 5.6. of [35]. D was a diagonal matrix and $U = I - 2uu^T$, $u^T u = 1$. The elements of $D = \text{diag}(d_1, \dots, d_n)$ were randomly generated and uniformly distributed

on $(-5, 5)$. The entries of u and b were randomly selected from a uniform distribution on $(-0.5, 0.5)$. The vector u was normalized to have unit length. As in the first family, an orthogonalization of b against q_1 generated a hard case. A noise vector of norm 10^{-8} was added to b . To ensure the hard case the radius was chosen as $r = 2r_{min}$ where $r_{min} = \|(A - d_1 I)^+ b\|$. In the normal case the radius was fixed at $r = 100$ when $n = 256, n = 576, n = 1024$ and $r = 20$ when $n = 100$.

We used the IRLM implemented in the package ARPACK [12], [37] for computing the two smallest eigenpairs at each iteration in the NMFA. For the IRLM we choose nine Lanczos basis vectors, six shifts on each implicit restart, and different tolerances at each case of the two families. In the first family ($A = L - 5I$) the initial tolerance was $\varepsilon_R = 0.5$ for all dimensions in both cases, except $\varepsilon_R = 0.1$ when $n = 1024$ in the normal case. Subsequently, in the normal case $\varepsilon_R = \max\{\min\{\varepsilon_R, |r - \|x^k\|/r|\}, 0.075\}$ when $n = 100, n = 256, n = 576$; $\varepsilon_R = \max\{\min\{\varepsilon_R, |r - \|x^k\|/r|\}, 0.01\}$ when $n = 1024$ and in the hard case; $\varepsilon_R = \max\{\min\{\varepsilon_R, |r - \|x^k\|/1000r|, |\tau^2(z^T Az - \lambda_k)/(1000(b^T x^k + \lambda_k r^2))|\}, 0.01\}$ when $n = 100, n = 256, n = 576$, $\varepsilon_R = \max\{\min\{\varepsilon_R, |r - \|x^k\|/1000r|, |\tau^2(z^T Az - \lambda_k)/1000(b^T x^k + \lambda_k r^2)|\}, 0.001\}$ when $n = 1024$. In the second family ($A = UDU^T$) the initial tolerance was $\varepsilon_R = 10^{-4}$ (resp., $\varepsilon_R = 10^{-2}$) when $n = 100$, $\varepsilon_R = 10^{-6}$ (resp., $\varepsilon_R = 10^{-3}$) when $n = 256, n = 576$, and $\varepsilon_R = 10^{-7}$ (resp., $\varepsilon_R = 10^{-4}$) when $n = 1024$ in the hard case (resp., the normal case). Subsequently, $\varepsilon_R = \min\{\varepsilon_R, |r - \|x^k\|/1000r|, |\tau^2(z^T Az - \lambda_k)/(1000(b^T x^k + \lambda_k r^2))|\}$ (resp., $\varepsilon_R = \min\{\varepsilon_R, \sqrt{|r - \|x^k\|/r|}, |\tau^2(z^T Az - \lambda_k)/(b^T x^k + \lambda_k r^2)|\}$). For a fixed tolerance ε_R , the stopping condition in the IRLM is $(\|f\| \|e_j^T y\|)^2 \leq \varepsilon_R |\mu| G(j)$ (cf. [19, pp. 206, 222]). In our experiments $j = 3$. The IRLM was started with $v = (1, \dots, 1)^T / \sqrt{n+1}$ and then the previously calculated eigenvector corresponding to the smallest eigenvalue was used as the initial vector. The parameters for the NMFA were the following: $\varepsilon_\delta = \varepsilon_{hc} = 10^{-6}$, $\varepsilon_\nu = 10^{-2}$, $\alpha_0 = \min\{0, \alpha_U\}$ and the initial of δ_S was $\delta_S = \min\{A_{ii} : i = 1, \dots, n\} + 0.5$ in the hard case with $A = UDU^T$ and $\delta_S = \min\{A_{ii} : i = 1, \dots, n\}$ in other cases. (See [35] for these notations.)

For computing ρ in the DCA we also applied the IRLM to the matrix A with the tolerance $\varepsilon_R = 0.5$ in the first family ($A = L - 5I$) and $\varepsilon_R = 0.1$ in the second family ($A = UDU^T$). So we obtained an approximation $\tilde{\lambda}_n$ of the largest eigenvalue λ_n of A and took $\rho = \tilde{\lambda}_n + 0.1$ (resp. $\rho = \tilde{\lambda}_n + 0.5$) when $A = L - 5I$ (resp. $A = UDU^T$).

For each case (normal and hard cases) seven problems per dimension were generated with different seeds for each family and solved by both Santos and Sorensen's algorithm (NMFA) and our code (DCA). The average results are indicated in Tables 3 and 4. In these tables we used the following notation: IT stands for the number of iterations; MVP stands for the number of matrix–vector products; KTC = $\|b + (A + \lambda^* I)x^*\| / \|b\|$ (this presents the Kuhn–Tucker condition); time: CPU time in seconds (on Sun SPARC-10). The value λ^* is the *average* of the obtained multipliers provided by each algorithm. Note that for the DCA the number of required matrix–vector products is exactly the sum of the number of iterations in the DCA and the number of matrix–vector products in the IRLM. In our experiments we observe from numerical results that the last quantity was always equal to nine. The quantity KTC in the DCA was equal to zero because λ^* was deduced from the expression $(A + \lambda^* I)x^* = -b$. CPU time of the DCA was presented in two parts: the first (resp. second) part indicated CPU time of the IRLM (resp. DCA).

5.3. Comparison with the gradient projection algorithm in the convex case. In the third experiment we solved 56 problems whose data was generated as

TABLE 3
Comparison with the new matrix-free method, $A = L - 5I$.

n	Type	Algorithm	IT	MVP	$\frac{r-\ x^*\ }{r}$	KTC	λ^*	time
100	normal case	NMFA	7.4	89.4	-3.10^{-8}	7.10^{-5}	4.8822	0.3
		DCA	47.6	56.6	-3.10^{-17}	0.0	4.7939	0.03+0.02
100	hard case	NMFA	12.6	298.2	-9.10^{-17}	3.10^{-3}	4.8340	0.9
		DCA	45.2	54.2	6.10^{-17}	0.0	4.8381	0.03+0.02
256	normal case	NMFA	7.8	131.4	-9.10^{-8}	2.10^{-6}	5.0013	0.7
		DCA	108.6	117.6	-2.10^{-16}	0.0	4.8631	0.04+ 0.1
256	hard case	NMFA	9.4	383.4	2.10^{-17}	9.10^{-4}	4.9319	1.7
		DCA	86.8	95.6	5.10^{-33}	0.0	4.9325	0.04+0.1
576	normal case	NMFA	8.2	135	10^{-7}	8.10^{-6}	5.0709	1.5
		DCA	471.8	480.8	-2.10^{-16}	0.0	5.0701	0.07+1.3
576	hard case	NMFA	10.8	678	2.10^{-16}	10^{-2}	4.9684	5.2
		DCA	166.6	175.6	-4.10^{-16}	0.0	4.9711	0.07 + 0.5
1024	normal case	NMFA	7.4	187.8	4.10^{-8}	4.10^{-7}	5.1212	4.9
		DCA	346.8	355.8	2.10^{-16}	0.0	5.1188	0.1+1.8
1024	hard case	NMFA	11.6	1490	-2.10^{-7}	2.10^{-2}	4.9818	20.1
		DCA	315.2	324.2	-2.10^{-16}	0.0	4.9907	0.1+1.6

TABLE 4
Comparison with the new matrix-free method, $A = UDU^T$.

n	Type	Algorithm	IT	MVP	$\frac{r-\ x^*\ }{r}$	KTC	λ^*	time
100	normal case	NMFA	6.6	313.8	4.10^{-8}	6.10^{-8}	4.970	3.2
		DCA	249	258	-7.10^{-17}	0.0	4.959	0.1+2.2
100	hard case	NMFA	8.8	909.6	-3.10^{-17}	10^{-2}	4.9492	9.1
		DCA	307.6	316.6	-2.10^{-16}	0.0	4.9469	0.1+2.7
256	normal case	NMFA	8	962.4	-7.10^{-9}	5.10^{-4}	4.9748	48.3
		DCA	316.6	325.6	5.10^{-16}	0.0	4.9698	0.5+18.2
256	hard case	NMFA	6.6	1447.8	2.10^{-16}	8.10^{-3}	4.9730	144.3
		DCA	316.8	325.8	7.10^{-17}	0.0	4.9717	0.5+18.4
576	normal case	NMFA	9.3	1658.3	-2.10^{-7}	10^{-8}	4.9852	391.9
		DCA	348	357	-4.10^{-16}	0.0	4.9812	3.2+124.9
576	hard case	NMFA	8.4	2242.8	-6.10^{-17}	4.10^{-3}	4.9843	1347.3
		DCA	343.4	352.3	-8.10^{-16}	0.0	4.9809	3.3+123.9
1024	normal case	NMFA	9.8	2318.3	2.10^{-7}	4.10^{-9}	4.9876	2209.8
		DCA	431	440	-6.10^{-16}	0.0	4.9806	13.2+476.4
1024	hard case	NMFA	10.3	3279	-8.10^{-16}	10^{-2}	4.9919	4381.6
		DCA	439.6	448.6	9.10^{-16}	0.0	4.9861	13.2+576.4

in subsection 5.1. We considered two types of matrices for A : positive semidefinite (p.s.d.) and positive definite (p.d.) with $n = 250$ and $n = 500$. The diagonal elements of D were in $(-5, 5)$ when $n = 250$ and in $(-50, 50)$ when $n = 500$. The elements of g were random numbers in $(-1, 1)$. As in subsection 5.1, a p.d. (resp., p.s.d.) matrix $A = UDU^T$ was obtained by replacing $D_{ii} < 0$ by $|D_{ii}|$ (resp., 0). We considered two cases (normal and hard case) when A is a p.s.d. matrix. For each type of A seven problems per dimension and per case were solved by the gradient projection algorithm (GPA) and by the DCA. In the GPA we took $\rho = 1/2\lambda_n + 0.1$ if A is p.s.d. and the theoretically optimal $\rho = 1/2(\lambda_1 + \lambda_n)$ if A is p.d. In the DCA we took $\rho = \lambda_n$. The average result is reported in Table 5.

TABLE 5
Comparison with the gradient projection method in the convex case.

n	Type of A	Algorithm	IT	$\frac{r-\ x^*\ }{r}$	λ^*	time
250	A p.d	GPA	640.4	-2.10^{-16}	0.5539	50.8
		DCA	250.2	0.06	0.5533	20.0
250	A p.s.d normal case	GPA	48	-10^{-16}	1.0230	4.1
		DCA	79	2.10^{-16}	1.0230	6.4
250	A p.s.d hard case	GPA	1360.4	0.01	0.0009	108.0
		DCA	250.8	0.02	0.0011	20.3
500	A p.d	GPA	2573.2	0.2841	0.0008	877.2
		DCA	294.6	0.3866	0.00172	101.4
500	A p.s.d normal case	GPA	225.6	-3.10^{-16}	0.4827	80.4
		DCA	305	-2.10^{-16}	0.4800	108.0
500	A p.s.d hard case	GPA	742.6	0.05	0.0074	254.4
		DCA	349.4	0.1	0.0095	120.1

5.4. Different choices for ρ . In the fourth experiment we solved 20 problems (ten per dimension) of the second family generated exactly as in subsection 5.2. We considered the normal case when $n = 256$ and the hard case when $n = 1024$. In the first case the radius was equal to 100 and in the last case it varied in the interval (110, 701). Each of these problems was solved five times, with $\rho = \|A\|_1$, $\rho = 1/2\|A\|_1$, $\rho = 1/3\|A\|_1$, $\rho = 5$, and $\rho = \tilde{\lambda}_n + 0.5$. The value $\tilde{\lambda}_n$ was an approximation to the largest eigenvalue of A calculated by the IRLM with the tolerance $tol = 0.1$. In these problems, $\|A\|_1$ were in the interval (18, 19.5). The average result is indicated in Table 6. CPU time (on Sun SPARC-10) in case $\rho = \tilde{\lambda}_n + 0.5$ was presented in two parts as in Tables 3 and 4. The quantity avOPT presented the average of optimal values.

TABLE 6
Average behavior for different choices of ρ .

n	ρ	$\frac{r-\ x^*\ }{r}$	avOPT	λ^*	IT	time
256 normal case	$\ A\ _1$	4.10^{-16}	-24823.7	4.9626	411	23.3
	$0.5\ A\ _1$	-3.10^{-16}	-24851.1	4.9682	362.4	20.6
	$0.3\ A\ _1$	3.10^{-16}	-24858.5	4.9697	319.4	18.4
	5.0	-1.10^{-16}	-24859.4	4.9699	311.6	18.4
	$\tilde{\lambda}_n + 0.5$	5.10^{-16}	-24858.9	4.96988	316.6	0.5+18.2
1024 hard case	$\ A\ _1$	-2.10^{-16}	-471484.3	4.9691	503	551.4
	$0.5\ A\ _1$	8.10^{-15}	-496199.1	4.9751	455.2	515.2
	$0.3\ A\ _1$	-10^{-15}	-496438.5	4.9795	438	487.3
	5.0	3.10^{-16}	-496481.4	4.9797	433.7	472.7
	$\tilde{\lambda}_n + 0.5$	3.10^{-16}	-496481.4	4.9797	433.7	13.2 + 502.7

5.5. Different sizes for the radius. In the last experiment (Table 7) we solved 10 problems of the second family mentioned above. We considered here the normal case with $n = 500$. The radius varied by a factor of 10 through the values 100, ..., 0.001. ρ was always equal to 5.

5.6. Comments on computational results. (i) For every test problem the DCA (without restarting procedure) always provided a global solution.

TABLE 7
Average behavior for different sizes of the radius.

$r \rightarrow$	100	10	1	0.1	0.01	0.001
IT	354	314	18	5	3	2.2
time	120.6	107.1	6.7	2.5	1.8	1.5

(ii) The parameter ρ as well as the d.c. representation of the objective function have an important influence on the convergence rate of the DCA. The practical choice (subsection 4.6) of ρ seems to be well suited (Table 6). We observe (see Table 5) the following:

- If A is positive definite, the choice $\rho = \lambda_n$ is much better than the theoretically optimal choice $\rho = (\lambda_1 + \lambda_n)/2$ given in Proposition 1. The ratio of execution times is 2.5 for $n = 250$ and 8.7 for $n = 500$.

- If A is positive semidefinite, the choice $\rho = \lambda_n$ is better than the choice $\rho = 0.5\lambda_n + 0.1$ in the hard case. The ratio of execution times is 5.4 when $n = 250$ and 2 when $n = 500$. The situation reverses in the normal case (1.6 when $n = 250$ and 1.35 when $n = 500$).

(iii) The DCA is faster than the Sfg. Table 1 indicates that in the normal case (resp., hard case) with A being indefinite the ratio of execution times varies between 2.4 and 8 (resp., 1.7 and 4.7). In the convex case Table 2 shows that the ratio of execution times varies between 1.2 and 16.8 (resp., 1.7 and 16.6) when A is positive definite (resp., positive semidefinite).

(iv) The DCA is faster than the NMFA. In the case where $A = L - 5I$, Table 3 indicates that the ratio of execution times belongs to the interval $]1, 18[$. It is more significant in the hard case (equal to 18 when $n = 100$, 12 when $n = 256$, 10 when $n = 576$ and 11.8 when $n = 1024$). In the normal case the ratio is smaller (equal to 6 when $n = 100$, 5 when $n = 256$ and 2.6 when $n = 1024$).

In the case where $A = UDU^T$, Table 4 shows similar results. In the normal case the ratio of execution times is 1.4 when $n = 100$, 2.6 when $n = 256$, 3 when $n = 576$, and 4.5 when $n = 1024$. It becomes (in the hard case) 3.2 when $n = 100$, 7.6 when $n = 256$, 10.6 when $n = 576$, and 7.4 when $n = 1024$.

(v) One of the referees suggested the example: $A = \text{Diag}(1, -1)$, $b = (1, 1)$, $r = 2$, and $\rho = 1.1$, where the pure DCA (i.e., without the restarting procedure) with the choice $x^o = (r/\sqrt{n}, r/\sqrt{n}) = (\sqrt{2}, \sqrt{2})$ does not lead to a global solution. However, we observe that the random choice of $x^o : x_i^o = 2[\text{random}(100) - 0.5]$, $i = 1, 2$ yields a global one.

6. Conclusion. We have presented a thorough study of the d.c. optimization, its algorithm DCA, and the application of the DCA to solving the trust-region subproblems.

By its very approach the DCA is completely different from other available methods for the trust-region problem [17], [24], [32], [35], [38]. The DCA is quite simple and requires matrix–vector products only. Moreover, it indifferently treats the normal and hard cases of the trust-region subproblem. Theoretically, it has been stated that the DCA converges to a Kuhn–Tucker point of the trust-region subproblem. We observe, however, that in all numerical examples we tried, convergence occurred to a global minimizer. The foundations of the d.c. optimization theory enable us to conjecture that the DCA converges to a local minimizer (of the trust-region subproblem) and so, most times, to a global one according to Martinez’s results. In the absence of a

theoretical proof of the conjecture, we have pointed out a simple numerical procedure to restart the DCA with a new initial point-in-case when the solution provided by the DCA is not global. This restarting procedure needs computing the smallest eigenvalue of the matrix A (in the trust-region subproblem) and a corresponding eigenvector and uses for that the quite inexpensive *Implicitly Restarted Lanczos Method* introduced by D. C. Sorensen [37]. Since the IRLM meets the requirements of limited storage and reliance only on matrix–vector products, the DCA with at most $2m + 2$ restartings (Theorem 4) requires only matrix–vector products too and converges to a global solution. In practice the DCA rarely has recourse to the restarting procedure. From the computational viewpoint, a lot of our numerical experiments proved the robustness and efficiency of the DCA with respect to other well-known algorithms, especially in the large scale trust-region subproblem.

Appendix. Proof of the basic convergence theorem of the DCA for general d.c. programming. First, we need the following results.

PROPOSITION A.1. *Suppose that the sequences $\{x^k\}$ and $\{y^k\}$ are generated by the simplified DCA. Then we have*

$$(i) \quad (g - h)(x^{k+1}) \leq (h^* - g^*)(y^k) - \frac{\rho_2}{2} \|dx^k\|^2 \leq (g - h)(x^k) - \frac{\rho_1 + \rho_2}{2} \|dx^k\|^2.$$

The equality $(g - h)(x^{k+1}) = (g - h)(x^k)$ holds if and only if

$$x^k \in \partial g^*(y^k), y^k \in \partial h(x^{k+1}) \text{ and } (\rho_1 + \rho_2) \|dx^k\| = 0.$$

(ii) *Similarly, by duality we have*

$$(h^* - g^*)(y^{k+1}) \leq (g - h)(x^{k+1}) - \frac{\rho_1^*}{2} \|dy^k\|^2 \leq (h^* - g^*)(y^k) - \frac{\rho_1^* + \rho_2^*}{2} \|dy^k\|^2.$$

The equality $(h^* - g^*)(y^{k+1}) = (h^* - g^*)(y^k)$ holds if and only if

$$x^{k+1} \in \partial g^*(y^{k+1}), y^k \in \partial h(x^{k+1}) \text{ and } (\rho_1^* + \rho_2^*) \|dy^k\| = 0.$$

Proof of Proposition A.1. (i) Since $y^k \in \partial h(x^k)$ we have $h(x^{k+1}) \geq h(x^k) + \langle x^{k+1} - x^k, y^k \rangle + \rho_2/2 \|dx^k\|^2$. Hence,

$$(26) \quad (g - h)(x^{k+1}) \leq g(x^{k+1}) - \langle x^{k+1} - x^k, y^k \rangle - h(x^k) - \frac{\rho_2}{2} \|dx^k\|^2.$$

Likewise, $x^{k+1} \in \partial g^*(y^k)$ implies

$$g(x^k) \geq g(x^{k+1}) + \langle x^k - x^{k+1}, y^k \rangle + \frac{\rho_1}{2} \|dx^k\|^2.$$

So,

$$(27) \quad g(x^{k+1}) - \langle x^{k+1} - x^k, y^k \rangle - h(x^k) \leq (g - h)(x^k) - \frac{\rho_1}{2} \|dx^k\|^2.$$

On the other hand,

$$(28) \quad x^{k+1} \in \partial g^*(y^k) \Leftrightarrow \langle x^{k+1}, y^k \rangle = g(x^{k+1}) + g^*(y^k),$$

$$(29) \quad y^k \in \partial h(x^k) \Leftrightarrow \langle x^k, y^k \rangle = h(x^k) + h^*(y^k).$$

Thus,

$$(30) \quad g(x^{k+1}) - \langle x^{k+1} - x^k, y^k \rangle - h(x^k) = h^*(y^k) - g^*(y^k).$$

Finally, combining (26), (27), and (30), we get

$$(31) \quad (g-h)(x^{k+1}) \leq (h^* - g^*)(y^k) - \frac{\rho_2}{2} \|dx^k\|^2 \leq (g-h)(x^k) - \frac{\rho_1 + \rho_2}{2} \|dx^k\|^2.$$

If $\rho_1 + \rho_2 > 0$, the last statement of (i) is an immediate consequence of the equality $dx^k = 0$ and the construction of the sequences $\{x^k\}$ and $\{y^k\}$.

It is clear that there exist ρ_1 and ρ_2 such that $\rho_1 + \rho_2 > 0$ if and only if $\rho(h) + \rho(g) > 0$. In case $\rho(h) = \rho(g) = 0$, (29) implies the equivalence between $(g-h)(x^{k+1}) = (g-h)(x^k)$ and the combination of (32) and (33)

$$(32) \quad (g-h)(x^{k+1}) = (h^* - g^*)(y^k),$$

$$(33) \quad (h^* - g^*)(y^k) = (g-h)(x^k).$$

We then deduce from (28) and (32) that $h(x^{k+1}) + h^*(y^k) = \langle x^{k+1}, y^k \rangle$, i.e., $y^k \in \partial h(x^{k+1})$. Similarly, (29) and (33) give $g(x^k) + g^*(y^k) = \langle x^k, y^k \rangle$, i.e., $x^k \in \partial g^*(y^k)$. Property (ii) is proved analogously. \square

The following result is an important consequence of Proposition A.1.

COROLLARY A.2.

$$(i) \quad (g-h)(x^{k+1}) \leq (h^* - g^*)(y^k) - \frac{\rho_2}{2} \|dx^k\|^2 \leq (g-h)(x^k) - \left[\frac{\rho_1^*}{2} \|dy^{k-1}\|^2 + \frac{\rho_2}{2} \|dx^k\|^2 \right].$$

$$(ii) \quad (g-h)(x^{k+1}) \leq (h^* - g^*)(y^k) - \frac{\rho_2^*}{2} \|dy^k\|^2 \leq (g-h)(x^k) - \left[\frac{\rho_1^*}{2} \|dy^{k-1}\|^2 + \frac{\rho_2^*}{2} \|dy^k\|^2 \right].$$

The equality $(g-h)(x^{k+1}) = (g-h)(x^k)$ holds if and only if

$$x^k \in \partial g^*(y^k), \quad y^k \in \partial h(x^{k+1}) \quad \text{and} \quad (\rho_1 + \rho_2) dx^k = \rho_1^* dy^{k-1} = \rho_2^* dy^k = 0.$$

Similarly, by duality we have

$$(iii) \quad (h^* - g^*)(y^{k+1}) \leq (g-h)(x^{k+1}) - \frac{\rho_1^*}{2} \|dy^k\|^2 \leq (h^* - g^*)(y^k) - \left[\frac{\rho_1^*}{2} \|dy^k\|^2 + \frac{\rho_2}{2} \|dx^k\|^2 \right].$$

$$(iv) \quad (h^* - g^*)(y^{k+1}) \leq (g-h)(x^{k+1}) - \frac{\rho_1}{2} \|dx^{k+1}\|^2 \leq (h^* - g^*)(y^k) - \left[\frac{\rho_1}{2} \|dx^{k+1}\|^2 + \frac{\rho_2}{2} \|dx^k\|^2 \right].$$

The equality $(h^* - g^*)(y^{k+1}) = (h^* - g^*)(y^k)$ holds if and only if

$$x^{k+1} \in \partial g^*(y^{k+1}), \quad y^k \in \partial h(x^{k+1}) \quad \text{and} \quad (\rho_1^* + \rho_2^*) dy^k = \rho_2 dx^k = \rho_1 dx^{k+1} = 0.$$

Proof. The inequalities in (i) and (ii) are easily deduced from Properties (i) and (ii) of Proposition A.1. The inequalities in (iii) and (iv) can be shown by the same arguments as in the proof of Proposition A.1. \square

We are now in a position to demonstrate Theorem 3.7.

Proof of Theorem 3.7. Properties (i) and (ii) are proved analogously; therefore, we give here the proof for (i) only.

The first inequality of (i) is an immediate consequence of (i) and (ii) of Corollary A.2.

- If $\rho_2 \|dx^k\|^2 \leq \rho_2^* \|dy^k\|^2$, then it follows from (ii) of Corollary A.2 that

$$\begin{aligned} (h^* - g^*)(y^k) - \max \left\{ \frac{\rho_2}{2} \|dx^k\|^2, \frac{\rho_2^*}{2} \|dy^k\|^2 \right\} &= (h^* - g^*)(y^k) - \frac{\rho_2^*}{2} \|dy^k\|^2 \\ &\leq (g - h)(x^k) - \left\{ \frac{\rho_1^*}{2} \|dy^{k-1}\|^2 + \frac{\rho_2^*}{2} \|dy^k\|^2 \right\}. \end{aligned}$$

Similarly, Property (i) of Corollary A.2 implies

$$\begin{aligned} (h^* - g^*)(y^k) - \frac{\rho_2^*}{2} \|dy^k\|^2 &\leq (h^* - g^*)(y^k) - \frac{\rho_2}{2} \|dx^k\|^2 \\ &\leq (g - h)(x^k) - \left[\frac{\rho_1^*}{2} \|dy^{k-1}\|^2 + \frac{\rho_2}{2} \|dx^k\|^2 \right]. \end{aligned}$$

On the other hand, by (i) of Proposition A.1,

$$(h^* - g^*)(y^k) - \frac{\rho_2^*}{2} \|dy^k\|^2 \leq (h^* - g^*)(y^k) - \frac{\rho_2}{2} \|dx^k\|^2 \leq (g - h)(x^k) - \frac{\rho_1 + \rho_2}{2} \|dx^k\|^2.$$

Combining these inequalities, we get the second inequality of (i).

- If $\rho_2^* \|dy^k\|^2 \leq \rho_2 \|dx^k\|^2$, then by using the same arguments we can easily show the second inequality of (i). The first property of (iii) is evident. We will prove the last one.

Taking (30) and (i) into account, we have

$$\lim_{k \rightarrow +\infty} (g - h)(x^{k+1}) = \lim_{k \rightarrow +\infty} \{g(x^{k+1}) - \langle x^{k+1} - x^k, y^k \rangle - h(x^k)\} = \lim_{k \rightarrow +\infty} (g - h)(x^k).$$

The second equality implies $\lim_{k \rightarrow +\infty} \{g(x^{k+1}) - \langle x^{k+1} - x^k, y^k \rangle - g(x^k)\} = 0$, i.e., $\lim_{k \rightarrow +\infty} \{g(x^k) + g^*(y^k) - \langle x^k, y^k \rangle\} = 0$, since $x^{k+1} \in \partial g^*(y^k)$.

Likewise, it results from the first equality that $\lim_{k \rightarrow +\infty} \{h(x^{k+1}) - \langle x^{k+1} - x^k, y^k \rangle - h(x^k)\} = 0$, i.e., $\lim_{k \rightarrow +\infty} \{h(x^{k+1}) + h^*(y^k) - \langle x^{k+1}, y^k \rangle\} = 0$ since $y^k \in \partial h(x^k)$.

(iv) We assume α is finite and the sequences $\{x^k\}$ and $\{y^k\}$ are bounded. Let x^* be a limit point of $\{x^k\}$. For the sake of simplicity we write $\lim_{k \rightarrow +\infty} x^k = x^*$. We can suppose (by extracting a subsequence if necessary) that the sequence $\{y^k\}$ converges to $y^* \in \partial h(x^*)$. Property (iii) then implies

$$\lim_{k \rightarrow +\infty} \{g(x^k) + g^*(y^k)\} = \lim_{k \rightarrow +\infty} \langle x^k, y^k \rangle = \langle x^*, y^* \rangle.$$

Let $\theta(x, y) = g(x) + g^*(y)$ for $(x, y) \in X \times Y$. It is clear that $\theta \in \Gamma_o(X \times Y)$. Then the lower semicontinuity of θ implies

$$\theta(x^*, y^*) \leq \lim_{k \rightarrow +\infty} \inf \theta(x^k, y^k) = \lim_{k \rightarrow +\infty} \theta(x^k, y^k) = \langle x^*, y^* \rangle,$$

i.e., $\theta(x^*, y^*) = g(x^*) + g^*(y^*) = \langle x^*, y^* \rangle$.

In other words, $y^* \in \partial g(x^*)$. According to Lemma A.3 stated below we have

$$\lim_{k \rightarrow +\infty} h(x^k) = h(x^*) \text{ and } \lim_{k \rightarrow +\infty} h^*(y^k) = h^*(y^*) \text{ since } y^k \in \partial h(x^k), x^k \rightarrow x^* \text{ and } y^k \rightarrow y^*.$$

Hence, in virtue of (iii),

$$\lim_{k \rightarrow +\infty} (g - h)(x^k) = \lim_{k \rightarrow +\infty} g(x^k) - \lim_{k \rightarrow +\infty} h(x^k) = \lim_{k \rightarrow +\infty} g(x^k) - h(x^*) = \beta,$$

$$\lim_{k \rightarrow +\infty} (h^* - g^*)(y^k) = \lim_{k \rightarrow +\infty} h^*(y^k) - \lim_{k \rightarrow +\infty} g^*(y^k) = h^*(y^*) - \lim_{k \rightarrow +\infty} g^*(y^k) = \beta.$$

It then suffices to show that

$$\lim_{k \rightarrow +\infty} g(x^k) = g(x^*); \quad \lim_{k \rightarrow +\infty} g^*(y^k) = g^*(y^*).$$

Since $\lim_{k \rightarrow +\infty} g(x^k)$ and $\lim_{k \rightarrow +\infty} g^*(y^k)$ exist, (iii) implies

$$g(x^*) + g^*(y^*) = \lim_{k \rightarrow +\infty} \{g(x^k) + g^*(y^k)\} = \lim_{k \rightarrow +\infty} g(x^k) + \lim_{k \rightarrow +\infty} g^*(y^k).$$

Further, because of the lower semicontinuity of g and g^* ,

$$\lim_{k \rightarrow +\infty} g(x^k) = \lim_{k \rightarrow +\infty} \inf g(x^k) \geq g(x^*), \quad \lim_{k \rightarrow +\infty} g^*(y^k) = \lim_{k \rightarrow +\infty} \inf g^*(y^k) \geq g^*(y^*).$$

The former equalities imply that these last inequalities are in fact equalities. The proof of Theorem 3.7 is complete. \square

LEMMA A.3. *Let $h \in \Gamma_o(X)$ and $\{x^k\}$ be a sequence of elements in X such that*

- (i) $x^k \rightarrow x^*$;
- (ii) *there exists a bounded sequence $\{y^k\}$ with $y^k \in \partial h(x^k)$;*
- (iii) $\partial h(x^*)$ *is nonempty.*

Then $\lim_{k \rightarrow +\infty} h(x^k) = h(x^)$.*

Proof. Indeed, let $y^* \in \partial h(x^*)$, then $h(x^k) \geq h(x^*) + \langle x^k - x^*, y^* \rangle \quad \forall k$. Since $y^k \in \partial h(x^k)$ we have $h(x^*) \geq h(x^k) + \langle x^* - x^k, y^k \rangle \quad \forall k$. Hence, $h(x^k) \leq h(x^*) + \langle x^k - x^*, y^k \rangle \quad \forall k$. As $x^k \rightarrow x^*$, we have $\lim_{k \rightarrow +\infty} \langle x^k - x^*, y^* \rangle = 0$. Moreover, $\lim_{k \rightarrow +\infty} \langle x^k - x^*, y^k \rangle = 0$, since the sequence $\{y^k\}$ is bounded. Consequently, $\lim_{k \rightarrow +\infty} h(x^k) = h(x^*)$. \square

Acknowledgments. The authors are grateful to the referees and to Professor J. M. Martinez for their helpful comments and suggestions as well as for their example in 5.6 (v) which have improved the presentation of the revised paper. Thanks also go to Dr. S. A. Santos and Professor D. C. Sorensen for making their MATLAB code available to us for experimentation.

REFERENCES

- [1] J. R. CLERMONT, M. E. DE LA LANDE, AND T. PHAM DINH, *Analysis of plane and axisymmetric flows of incompressible fluids with the stream tube method: Numerical simulation by trust region algorithm*, Internat. J. Numer. Methods Fluids, 13 (1991), pp. 371–339.
- [2] R. FLETCHER, *Practical Methods of Optimization*, 2nd ed., Wiley-Interscience, New York, 1991.
- [3] G. E. FORSYTHE AND G. H. GOLUB, *On the stationary values of a second-degree polynomial on the unit sphere*, J. Soc. Indust. Appl. Math., 13 (1965), pp. 1050–1068.
- [4] W. GANDER, G. H. GOLUB, AND U. VON MATT, *A constrained eigenvalue problem*, Linear Algebra Appl., 114/115 (1989), pp. 815–839.
- [5] D. M. GAY, *Computing optimal locally constrained steps*, SIAM J. Sci. Statist. Comput., 2 (1981), pp. 186–197.
- [6] G. H. GOLUB AND U. VON MATT, *Quadratically constrained least squares and quadratic problems*, Numer. Math., 59 (1991), pp. 561–580.

- [7] L. T. HOAI AN, *Analyse numérique des algorithmes de l'Optimisation d.c. Approches locales et globales. Code et simulations numériques en grande dimension. Applications*, Thèse de Doctorat de l'Université de Rouen, France, 1994.
- [8] L. T. HOAI AN AND T. PHAM DINH, *Solving a class of linearly constrained indefinite quadratic problems by D.c. algorithms*, J. Global Optim., 11 (1997), pp. 253–285.
- [9] J. B. HIRIART-URRUTY, *From convex optimization to non convex optimization. Part I: Necessary and sufficient conditions for global optimality*, in Nonsmooth Optimization and Related Topics, Ettore Majorana International Sci. Ser. Phys. Sci. 43, Plenum Press, New York, 1988.
- [10] R. HORST AND H. TUY, *Global Optimization (Deterministic Approaches)*, Springer-Verlag, Berlin, 1993.
- [11] P. J. LAURENT, *Approximation et Optimisation*, Hermann, Paris, 1972.
- [12] R. LEHOUCQ, D. C. SORENSEN, AND P. A. VU, *ARPACK: An implementation of the implicitly restarted Arnoldi iteration that computes some of the eigenvalues and eigenvectors of a large sparse matrix*, available online from netlib@ornl.gov under the directory scalapack.
- [13] S. LUCIDI, L. PALAGI, AND M. ROMA, *On some properties of quadratic programs with a convex quadratic constraint*, SIAM J. Optim., 8 (1998), pp. 105–122.
- [14] P. MAHEY AND T. PHAM DINH, *Proximal decomposition of the graph of maximal monotone operator*, SIAM J. Optim., 5 (1995), pp. 454–468.
- [15] J. M. MARTINEZ, *Local minimizers of quadratic functions on Euclidean balls and spheres*, SIAM J. Optim., 4 (1994), pp. 159–176.
- [16] J. J. MORE, *Recent developments in algorithm and software for trust region methods*, in Mathematical Programming, The State of the Art, Springer-Verlag, Berlin, 1983, pp. 258–287.
- [17] J. J. MORE AND D. C. SORENSEN, *Computing a trust region step*, SIAM J. Sci. Statist. Comput., 4 (1983), pp. 553–572.
- [18] A. M. OSTROWSKI, *Solutions of Equations and Systems of Equations*, Academic Press, New York, 1966.
- [19] B. N. PARLETT, *The Symmetric Eigenvalue Problem*, Prentice-Hall, Englewood Cliffs, NJ, 1980.
- [20] T. PHAM DINH, *Contribution à la théorie de normes and ses applications à l'analyse numérique*, Thèse de Doctorat d'Etat Es Science, Université Joseph-Fourier, Grenoble, 1981.
- [21] T. PHAM DINH, *Convergence of subgradient method for computing the bound norm of matrices*, Linear Algebra Appl., 62 (1984), pp. 163–182.
- [22] T. PHAM DINH, *Algorithmes de calcul du maximum d'une forme quadratique sur la boule unité de la norme du maximum*, Numer. Math., 45 (1984), pp. 377–401.
- [23] T. PHAM DINH AND S. ELBERNOUSSI, *Duality in d.c. (difference of convex functions) optimization. Subgradient methods*, in Trends in Mathematical Optimization, Internat. Ser. Numer. Math. 84, Birkhäuser, Basel, 1988, pp. 276–294.
- [24] T. PHAM DINH, *Méthodes numériques pour la minimisation globale d'une forme quadratique (convexe ou non convexe) sur une boule et une sphère euclidiennes*, Rapport de Recherche, Université Joseph-Fourier, Grenoble, 1989.
- [25] T. PHAM DINH AND S. WANG, *Training multi-layered neural network with a trust region based algorithm*, Math. Modelling Numer. Anal., 24 (1990), pp. 523–553.
- [26] T. PHAM DINH AND L. T. HOAI AN, *Minimisation globale d'une forme quadratique sur une boule et une sphère euclidiennes. Stabilité de la dualité lagrangienne. Optimalité globale. Méthodes numériques*, Rapport de Recherche, L.M.I, CNRS URA 1378, INSA-Rouen, 1992.
- [27] T. PHAM DINH AND L. T. HOAI AN, *Optimisation d.c. (différence de deux fonctions convexes). Dualité et Stabilité. Optimalités locale et globale. Algorithmes de l'optimisation d.c. (DCA)*, Rapport de Recherche, LMI, CNRS URA 1378, INSA-Rouen, 1994.
- [28] T. PHAM DINH AND L. T. HOAI AN, *Stabilité de la dualité lagrangienne en optimisation d.c. (différence de deux fonctions convexes)*, C.R. Acad. Sci. Paris, Ser. I Math., t.318 (1994), pp. 379–384.
- [29] T. PHAM DINH AND L. T. HOAI AN, *Polyhedral d.c. (Difference of Convex Functions) Programming: Theory, Polyhedral d.c. Algorithm (DCA) and Applications*, preprint, LMI, CNRS URA 1378, INSA-Rouen, 1995.
- [30] T. PHAM DINH AND L. T. HOAI AN, *Lagrangian stability and global optimality on nonconvex quadratic minimization over Euclidean balls and spheres*, J. Convex Anal., 2 (1995), pp. 263–276.
- [31] B. POLYAK, *Introduction to Optimization*, Optimization Software, Inc., Publication Division, New York, 1987.

- [32] F. RENDL AND H. WOLKOWICZ, *A Semidefinite Framework to Trust Region Subproblems with Application to Large Scale Minimization*, CORR Report 94-32, Department of Combinatorics and Optimization, University of Waterloo, Waterloo, ON, Canada, 1994.
- [33] R. T. ROCKAFELLAR, *Convex Analysis*, Princeton University Press, Princeton, NJ, 1970.
- [34] R. T. ROCKAFELLAR, *Monotone operators and the proximal point algorithm in convex programming*, SIAM J. Control Optim., 14 (1976), pp. 877–898.
- [35] S. A. SANTOS AND D. C. SORENSEN, *A new matrix-free algorithm for the large-scale trust-region subproblem*, SIAM J. Optim., submitted.
- [36] D. C. SORENSEN, *Newton's method with a model trust region modification*, SIAM J. Numer. Anal., 19 (1982), pp. 409–426.
- [37] D. C. SORENSEN, *Implicit application of polynomial filters in a K-step Arnoldi method*, SIAM J. Matrix Anal. Appl., 13 (1992), pp. 357–385.
- [38] D. C. SORENSEN, *Minimization of a large scale quadratic function subject to a spherical constraint*, SIAM J. Optim., 7 (1997), pp. 141–161.
- [39] J. F. TOLAND, *On subdifferential calculus and duality in nonconvex optimization*, Bull. Soc. Math. France, Mémoire, 60 (1979), pp. 177–183.