A Sub-Additive DC Approach to the Complementarity Problem

Tangi Migot, Lina Abdallah, Mounir Haddou

A Sub-Additive DC Approach to the Complementarity Problem

Migot, T.[*] Abdallah, L.[†] Haddou, M.[‡]

2017

Abstract

In this article, we propose a new merit function based on sub-additive functions for solving a non-linear complementarity problem (NCP). This leads to consider an optimization problem that is equivalent to the NCP. In the case of a concave NCP this optimization problem is a Difference of Convex (DC) program and we can therefore use DC Algorithm to locally solve it. We prove that in the case of a concave monotone NCP, it is sufficient to compute a stationary point of the optimization problem to obtain a solution of the complementarity problem. In the case of a general NCP, assuming that a DC decomposition of the complementarity problem is known, we propose a penalization technique to reformulate the optimization problem as a DC program and prove that local minima of this penalized problem are solutions of the NCP. Numerical results on linear complementarity problems and absolute value equations show that our method is promising.

Keywords: complementarity problem - difference of convex - merit function - dc algorithm

AMS Subject Classification: 90C59 – 90C30 – 90C33 – 65K05 – 49M20

1 Introduction

The non-linear complementarity problem (NCP) consists in finding $x \in \mathbb{R}^n$ satisfying

$$x \geq 0, F(x) \geq 0, x^T F(x) = 0, \tag{1}$$

where $F : \mathbb{R}^n \to \mathbb{R}^n$. In the special case where F is an affine function, (1) is reduced to the linear complementarity problem (LCP). Unless explicitly specified no smoothness assumption on F is made unless continuity.

The NCP has been very popular due to its numerous applications in economy, physics, chemistry,... see the monographs [5, 7] and references therein. This problem is a feasibility and not an optimization problem. However, it is well-known that it is closely related with optimization problems. Indeed, optimality conditions of many optimization problems may be cast as an NCP.

The family of methods to solve this problem can be divided in two: equation reformulation methods and merit function methods. In the former, the complementarity problem (1) is reformulated as an unconstrained (non-smooth) equation solved by a Newton-kind method or an unconstrained minimization problem depending on the assumptions on F, see [3, 4, 6, 13, 14, 20]. In the latter, the complementarity is ensured through a merit function reformulating (1) as an optimization problem, see [8, 9, 11, 16]. An extensive treatment of these methods and their extensions can be found in [7]. In the sequel, we focus on the merit function approach that is the most tractable option if we do not make any strong assumption on the problem.

A generic formulation of a complementarity problem reformulated as an optimization problem is given by

$$\min_{x \in \mathbb{R}^n} \Theta(x) \text{ s.t. } x \geq 0, F(x) \geq 0, \tag{2}$$

[*] IRMAR-Insa, Rennes, France, e-mail: tangi.migot@insa-rennes.fr

[†] LaMa, Tripoli, Lebanon, e-mail: lina_abdallah@hotmail.fr

[‡] IRMAR-Insa, Rennes, France, e-mail: mounir.haddou@insa-rennes.fr

where $\Theta : \mathbb{R}^n \to \mathbb{R}$ is called a merit function if it satisfies the following property: x solves (1) if and only if x is the global minimizer of the previous problem. In this sense, we say that the optimization problem is equivalent to the complementarity problem. If the set of solutions of (1) is empty, then either the global optimal value of the objective function is positive or there is no global minima. It is to be noted that even so $F(x) = Mx + q$ the merit function, Θ, is not convex in general. Thus, it is fundamental to study the numerical methods used to solve the optimization problem (2). Due to the non-convexity, solving means finding a local minimum or at least a stationary point of (2).

Our aim in this article is to use the very efficient local theory designed for difference of convex (DC) functions by considering a new merit function that can be expressed as a sum of convex and concave functions. Local optimization approach for DC optimization and DC Algorithm (DCA) were introduced by Pham Dinh Tao in a preliminary form in the 80's [23] and extensively developed since then, see [21] for a review on this method. This approach has been widely used by many researchers, see a list of references on http://www.lita.univ-lorraine.fr/ lethi/index.php/dca.html. This work is motivated by the fact that DCA have been successfully applied to many (smooth or non-smooth) large-scale non-convex programs in various domains of applied sciences. In a recent article [16], the authors present four formulations of the LCP as a DC optimization problem and show promising numerical results.

Motivated by the recent development of DC approaches to solve the complementarity problems, we propose a new merit function based on a family of smooth sub-additive functions that has been used in the literature in the context of sparse optimization [19] and absolute value equation [1], whose latter can be reformulated as an LCP.

One of our main results considering this new merit function is that it is sufficient to consider stationary points of the related optimization problem instead of global minima to recover the solution of a monotone NCP. In particular, this proves that our approach is not more difficult than equation based approach in advantageous cases.

Besides, we prove that the optimization problem (2) with our new merit function is a DC program if F is concave. This property permits us to use the local theory to solve DC program and in particular DCA. In the case of the general NCP, assuming a DC composition of F is known, the optimization problem with our new merit function can be reformulated as a DC program through penalizations. We prove equivalence between local minimum of this penalized DC program and the solution of the NCP.

Numerical results on linear complementarity problems and absolute value equations show that the new method proposed here is promising compared to some existing methods in the literature.

In Section 2, we introduce the new merit function that is proved to be theoretically sound for the monotone NCP in Section 3. In Section 4, we show that this new formulation leads to a DC program that can be solved using DCA as explained in the Section 5. Finally, in Section 6, we give several numerical experiments to show the validity of our method compared to existing methods in the literature on linear and non-linear problems.

2 A New Merit Function for Complementarity Problems

In this section, we present a new reformulation of the complementarity problem by introducing a new merit function. Let us consider the merit function $\Theta : \mathbb{R}^n_+ \times \mathbb{R}^n_+ \to \mathbb{R}$ defined by

$$\Theta(x, F(x)) := \sum_{i=1}^{n} \theta(x_i) + \theta(F_i(x)) - \theta(x_i + F_i(x)), \tag{3}$$

where functions $\theta : \mathbb{R}^n_+ \to [0,1[$ are non-decreasing concave continuously differentiable functions such that $\theta(0) = 0$. Examples of these functions for $r > 0$ are:

$$\begin{array}{ll} (\theta^1): & \theta(x) = \dfrac{x}{x+r}, \\ (\theta^{w_1}): & \theta(x) = (1 - e^{-(\frac{x}{r})})^k, \text{ for } k \leq 1, \\ (\theta^{log}): & \theta(x) = \dfrac{\log(1+x)}{\log(1+x+r)}. \end{array}$$

For r sufficiently small, the functions θ approximate a step function. This observation already leads to interesting results in sparse optimization in [19] and applied in the context of complementarity problems in [1, 13].

We use the function Θ to reformulate the NCP as follows

$$\Theta(x, F(x)) = 0, \ x \geq 0, \ F(x) \geq 0. \tag{4}$$

The following lemma shows that the functions θ are sub-additive.

Lemma 2.1. *$\forall i$ and $\forall$ $x_i \geq 0$, such that $F(x) \geq 0$, we have*

$$\forall i, \ \theta(x_i) + \theta(F_i(x)) \geq \theta(x_i + F_i(x)),$$

with equality if and only if $x_i = 0$ or $F_i(x) = 0$.

Proof. Since θ is concave, we obtain

$$\forall z \neq y \in \mathbb{R}, \forall t \in (0,1), \ t\theta(z) + (1-t)\theta(y) \leq \theta(tz + (1-t)y),$$

with equality if $t = 0$ or 1 and if $z = y$. For $y = 0$ and $\theta(0) = 0$ yields

$$\theta(tz) = \theta(tz + (1-t)y) \geq t\theta(z), \ \forall t \in (0,1),$$

with equality if $t = 0, 1$ or $z = 0$. Take $i \in \{1, ..., n\}$, $z = x_i$, $y = F_i(x)$ and suppose that $x_i + F_i(x) \neq 0$ (the case $x_i = F_i(x) = 0$ stay true)

$$\begin{aligned} \theta(x_i) + \theta(F_i(x)) &= \theta\left((x_i + F_i(x))\frac{x_i}{x_i + F_i(x)}\right) + \theta\left((x_i + F_i(x))\frac{F_i(x)}{x_i + F_i(x)}\right), \\ &\geq \frac{x_i}{x_i + F_i(x)}\theta((x_i + F_i(x))) + \frac{F_i(x)}{x_i + F_i(x)}\theta((x_i + F_i(x))), \\ &= \theta(x_i + F_i(x)), \end{aligned}$$

with equality if and only if $x_i = 0$ or $F_i(x) = 0$. □

It is now straightforward to conclude that finding a global minimum of (4) is equivalent to solving (1).

Theorem 2.1. *Let $x \in \mathbb{R}^n_+, F(x) : \mathbb{R}^n_+ \to \mathbb{R}^n_+$. It holds that $\Theta(x, F(x)) = 0 \Leftrightarrow x^T F(x) = 0$.*

Proof. By Lemma 2.1, Θ is a sum of non-negative terms, therefore Θ is equal to 0 if and only if all terms are equal to zero. In addition, for a component i there is equality if and only if $x_i = 0$ or $F_i(x) = 0$. □

Using (3) to define a sub-additive merit function, we consider the following optimization problem:

$$\begin{aligned} &\inf_{x \in \mathcal{X}} \ \Theta(x, F(x)) \\ &\text{s.t. } x \geq 0, F(x) \geq 0, \end{aligned} \tag{P}$$

where $\mathcal{X}$ is a bounded convex subset of $\mathbb{R}^n$. A box is a typical choice of $\mathcal{X}$. We denote $\mathcal{F}$ the feasible set of (P) defined as

$$\mathcal{F} := \{x \in \mathcal{X} \mid x \geq 0, F(x) \geq 0\}.$$

It is to be noted that the feasible set of (P) could be non-convex, unless the function F is assumed concave. The following result that is given without proof sums up the possible situations that may arise by solving (P) and their consequences to our initial aim that is solving (1).

Theorem 2.2. *Assume that $F : \mathbb{R}^n_+ \to \mathbb{R}^n$ is a continuous function and that $\mathcal{F}$ is non-empty. Then,* (P) *attains its extrema and exactly one of the following cases arises:*

- *The global optimal value of* (P) *is zero and the corresponding point is a solution of* (1)*;*
- *The global optimal value of* (P) *is positive and* (1) *has no solution.*

In general, if no preliminary information is known to define $\mathcal{X}$, we consider $\mathcal{X} = \mathbb{R}^n$ with the risk to compute an infimum instead of a minimum.

3 Sub-Additive Reformulation for the Monotone NCP

In this section, we prove that for a popular class of NCP finding a stationary point of (P) is actually sufficient to solve the initial problem. First, let us introduce the monotone NCP.

3.1 Monotone Complementarity Problems

We focus on the NCPs that are said monotone, i.e. F is P function (see definition below). We give classical definitions and characterization of this family of complementarity problems.

Definition 3.1. *[7] The function $F : \mathbb{R}^n \mapsto \mathbb{R}^n$ is said to be a P_0 function iff*

$$\max_{i:x_i \neq y_i} (x-y)_i(F_i(x) - F_i(y)) \geq 0 \quad \forall x \neq y \in \mathbb{R}^n.$$

If this inequality is strict, F is said to be a P-function.

Assuming differentiability of the function F this definition reduces to a condition on the jacobian matrix. We extend the definition of a P_0 (resp. P) function to a P_0 (resp. P) matrix.

Definition 3.2. *[7] A matrix $M \in \mathbb{R}^{n \times n}$ is said to be a:*

- *P_0 matrix if all of its principal minors are nonnegative;*
- *P matrix if all of its principal minors are positive;*
- *positive semidefinite if M is symmetric and $x^T Mx \geq 0$ for all $x \in \mathbb{R}^n$, and positive definite if $x^T Mx > 0$ for all $x \in \mathbb{R}^n (x \neq 0)$.*

A P matrix is a generalization of a positive definite matrix, while a P_0 matrix is a generalization of a positive semi-definite matrix. In the case of a smooth function F, then Proposition 3.5.10 of [7] gives the following characterization.

Proposition 3.1. *For a continuously differentiable function F, if its transpose jacobian $\nabla F(x)^T$ is a P matrix then F is a P function, whereas the transpose jacobian $\nabla F(x)^T$ is a P_0-matrix if and only if F is a P_0 function.*

The monotone NCPs are particularly useful since they provide a characterization of the set of solutions of the NCP. The following proposition is a sum up of results given in [7].

Proposition 3.2. *If F is a P function, then* (1) *has at most one solution.*

Among other interesting properties, we show in Lemma 3.1 that some regularity may be obtained for the set $\mathcal{F}$ for this family of NCPs.

3.2 Sufficient Optimality Condition for the Montone NCP

We assume here that the function F is a continuously differentiable function. A classical approach in non-linear programming is to study stationary points of (P) using the classical Karush-Kuhn-Tucker conditions. Let $x^* \in \mathcal{F}$ ($\mathcal{X} = \mathbb{R}^n$) be a stationary point of (P) if there exists $(\lambda, \mu) \in \mathbb{R}^n_+ \times \mathbb{R}^n_+$ that satisfies

$$\begin{aligned}&\nabla\Theta(x^*, F(x^*)) - \nabla F(x^*)^T\lambda - \mu = 0,\\ &\lambda^T F(x^*) = 0, F(x^*) \geq 0, \lambda \geq 0,\\ &x^{*,T}\mu = 0, x^* \geq 0, \mu \geq 0.\end{aligned}$$

It holds from non-linear programming theory that a local minimum of (P) that satisfies a constraint qualification is a stationary point of (P). Let us now give two examples of classical constraint qualifications. Beforehand, we use the following notations:

$$\mathcal{I}_F(x) := \{i = 1, \dots, n | F_i(x) = 0\}\} \text{ and } \mathcal{I}(x) := \{i = 1, \dots, n | x_i = 0\}\}.$$

Definition 3.3. *We say that $x^* \in \mathcal{F}$ satisfies*

- *Linear Independence CQ (LICQ) if the gradients*

$$\{\nabla F_i(x^*) \mid \forall i \in \mathcal{I}_F(x^*)\} \cup \{\nabla x^*_i \mid \forall i \in \mathcal{I}(x^*)\}$$

  are linearly independent.

- *Mangasarian-Fromowitz CQ (MFCQ) if the only solution to the equation*

$$0 = \sum_{i=1}^n \mu_i \nabla x^*_i + \sum_{i=1}^n \lambda_i \nabla F_i(x^*),$$

  with for all $i \in \{1, \dots, n\}$ $\min(\lambda_i, F_i(x^*)) = 0$ and $\min(\mu_i, x^*_i) = 0$, is the trivial solution. We say in this case that the gradients

$$\{\nabla F_i(x^*) \mid \forall i \in \mathcal{I}_F(x^*)\} \cup \{\nabla x^*_i \mid \forall i \in \mathcal{I}(x^*)\}$$

  are positively linearly independent.

In the special case of a monotone NCP, the following lemma shows that some constraint qualification holds for the set $\mathcal{F}$.

Lemma 3.1. *Let F be a continuously differentiable function from $\mathbb{R}^n$ to $\mathbb{R}^n$. Assume that F is a P function. Then, MFCQ holds at any point $x^* \in \mathcal{F}$. Furthermore, if x^* satisfies $x^* + F(x^*) > 0$, then LICQ holds at any point $x^* \in \mathcal{F}$.*

The assumption that $x^* + F(x^*) > 0$ is also well-known in the literature of complementarity problems under the name of strict complementarity.

By definition, MFCQ holds at a point x^* if the gradients of the active constraints are positively-linearly independent and LICQ holds at a point x^* if the gradients of the constraints are linearly independent.

Proof. First, let us prove that given $x^* \in \mathcal{F}$ if x^* satisfies $x^* + F(x^*) > 0$, then LICQ holds at x^*. Let $(\lambda, \mu) \in \mathbb{R}^n_+ \times \mathbb{R}^n_+$ be such that

$$0 = \sum_{i=1}^n \mu_i \nabla x^*_i + \sum_{i=1}^n \lambda_i \nabla F_i(x^*) \ , \tag{5}$$

where

$$\mu_i = 0 \text{ for } i \in I := \{i \in \{1, \dots, n\} \mid x^*_i > 0\},$$

$$\lambda_i = 0 \text{ for } i \in J := \{i \in \{1, \dots, n\} \mid F_i(x^*) > 0\}.$$

Since x^* satisfies $x^* + F(x^*) > 0$, then (5) is a combination of n gradients. We arrange the components of x and $F(x)$ such that $x_iF_i(x) = 0$ for $i = 1, \dots, m$ and $x_i > 0, F_i(x) > 0$ for $i = m+1, \dots, n$, so we can rewrite (5) as

$$\begin{pmatrix} I_J & \nabla F_{J\times I}(x^*)^T \\ 0 & \nabla F_{I\times I}(x^*)^T \end{pmatrix} \begin{pmatrix} \mu_J \\ \lambda_I \end{pmatrix} = 0.$$

By continuity of F it holds that the jacobian matrix of F is a P matrix by Proposition 3.1 and in particular all its principal minor are non-singular matrices. Computing the Schur complement on the left-hand side it holds that the matrix is non-singular, therefore $\mu_J = \lambda_I = 0$. So, LICQ holds at x^*.

Consider the case where there exists an index i such that $x_i^* + F_i(x^*) = 0$. Obviously (5) may be a combination of at least $n+1$ gradients and so, LICQ cannot hold. However, we prove that MFCQ holds at x^*.

Let $SC := \{i \in \{1, \dots, m\} \mid x_i^* = F_i(x^*) = 0\}$ and assume that $(\lambda, \mu) \in \mathbb{R}^{2n}_+$. We can rewrite (5) as

$$\begin{pmatrix} I_{J\setminus SC} & \nabla F_{J\setminus SC\times SC}(x^*)^T & \nabla F_{J\setminus SC\times I\setminus SC}(x^*)^T \\ 0 & \nabla F_{SC\times SC}(x^*)^T & \nabla F_{SC\times I\setminus SC}(x^*)^T \\ 0 & \nabla F_{I\setminus SC\times SC}(x^*)^T & \nabla F_{I\setminus SC\times I\setminus SC}(x^*)^T \end{pmatrix} \begin{pmatrix} \mu_{J\setminus SC} \\ \mu_{SC} + \lambda_{SC} \\ \lambda_{I\setminus SC} \end{pmatrix} = 0.$$

The left-hand side is obviously non-singular since $\nabla F_{I+SC\times I+SC}(x^*)^T$ is non-singular. So, it follows that $\mu_{J\setminus SC} = \mu_{SC} + \lambda_{SC} = \lambda_{I\setminus SC} = 0$, which leads to $\mu_J = \lambda_I = 0$ by non-negativity of μ and λ. Thus, the gradients

$$\{\nabla F_i(x^*)\ \forall i \in \{i|F_i(x^*) = 0\}\} \cup \{\nabla x_i^*\ \forall i \in \{i|x_i^* = 0\}\}$$

are positively linearly independent and the result follows. □

The previous lemma guarantees that any local minimum of (P) is a stationary point. The following theorem proves that computing a stationary point of (P) may be sufficient to solve (1).

Theorem 3.1. *Let F be a continuously differentiable P function. Assume that a solution of* (1) *exists. Then, any KKT point of* (P) *that satisfies the strict complementarity condition, is a solution of* (1).

Proof. Through the proof, we denote $\theta'(x)$ for $x \in \mathbb{R}^n$, the vector of componentwise derivatives of θ, i.e. $\theta'(x) := (\theta'(x_i))_{1\leq i\leq n}$.

Let $x \in \mathcal{F}$ be a stationary point of (P). Thus, there exists $\lambda, \mu \in \mathbb{R}^n_+ \times \mathbb{R}^n_+$ such that

$$\begin{aligned} &\theta'(x) + \nabla F(x)^T\theta'(F(x)) - \theta'(x + F(x)) - \nabla F(x)^T\theta'(x + F(x)) - \nabla F(x)^T\lambda - \mu = 0, \\ &\lambda^T F(x) = 0, F(x) \geq 0, \lambda \geq 0, \\ &x^T\mu = 0, x \geq 0, \mu \geq 0. \end{aligned}$$

We show the result by contradiction. Assume that x is not a solution of the NCP, then there exists a set of indices J non-empty such that for all $i \in J$ it holds that $x_i > 0\,\text{and}\,F_i(x) > 0$.

Denote the sets $I_1 = \{j = 1, \dots, n \mid x_j = 0, F_j(x) > 0\}, I_2 = \{k = 1, \dots, n \mid x_k > 0, F_k(x) = 0\}$. By the strict complementarity assumption, it always holds that $I_1 \cup J \cup I_2 = \{1, \dots, n\}$.

Rewriting the previous equation yields

$$\begin{aligned} \begin{pmatrix} 0_{I_1} \\ 0_J \\ 0_{I_2} \end{pmatrix} &= \begin{pmatrix} \theta'(0) \\ \theta'(x_J) \\ \theta'(x_{I_2}) \end{pmatrix} - \begin{pmatrix} \theta'(F_{I_1}(x)) \\ \theta'(x_J + F_J(x)) \\ \theta'(x_{I_2}) \end{pmatrix} + \nabla F(x)^T \begin{pmatrix} \theta'(F_{I_1}(x)) \\ \theta'(F_J(x)) \\ \theta'(0) \end{pmatrix} - \nabla F(x)^T \begin{pmatrix} \theta'(F_{I_1}(x)) \\ \theta'(F_J(x) + x_J) \\ \theta'(x_{I_2}) \end{pmatrix} \\ &\quad -\nabla F(x)^T \begin{pmatrix} 0_{I_1} \\ 0_J \\ \lambda_{I_2} \end{pmatrix} - \begin{pmatrix} \mu_{I_1} \\ 0_J \\ 0_{I_2} \end{pmatrix}, \\ &= \begin{pmatrix} \theta'(0) - \theta'(F_{I_1}(x)) \\ \theta'(x_J) - \theta'(x_J + F_J(x)) \\ 0 \end{pmatrix} + \nabla F(x)^T \begin{pmatrix} 0 \\ \theta'(F_J(x)) - \theta'(x_J + F_J(x)) \\ \theta'(0) - \theta'(x_{I_2}) - \lambda_{I_2} \end{pmatrix} - \begin{pmatrix} \mu_{I_1} \\ 0_J \\ 0_{I_2} \end{pmatrix}. \end{aligned}$$

The equation on I_2 gives

$$\begin{aligned}
0 &= \nabla F_{I_2\times J}(x)(\theta^{'}(F_J(x)) - \theta^{'}(x_J + F_J(x))) + \nabla F_{I_2\times I_2}(x)^T(\theta^{'}(0) - \theta^{'}(x_{I_2}) - \lambda_{I_2}),\\
\nabla F_{I_2\times I_2}(x)^T\lambda_{I_2} &= \nabla F_{I_2\times J}(x)^T(\theta^{'}(F_J(x)) - \theta^{'}(x_J + F_J(x))) + \nabla F_{I_2\times I_2}(x)^T(\theta^{'}(0) - \theta^{'}(x_{I_2})),\\
\lambda_{I_2} &= (\nabla F_{I_2\times I_2}(x)^T)^{-1}\nabla F_{I_2\times J}(x)^T(\theta^{'}(F_J(x)) - \theta^{'}(x_J + F_J(x))) + \theta^{'}(0) - \theta^{'}(x_{I_2}).
\end{aligned}$$

Existence of the inverse of the matrix $\nabla F_{I_2\times I_2}(x)^T$ is given by the jacobian matrix of F being a P matrix.

The equation on J and replacing λ_{I_2} yields

$$\begin{aligned}
0 &= \theta^{'}(x_J) - \theta^{'}(x_J + F_J(x)) + \nabla F_{J\times J}(x)^T(\theta^{'}(F_J(x)) - \theta^{'}(x_J + F_J(x)))\\
&+ \nabla F_{J\times I_2}(x)^T(\theta^{'}(0) - \theta^{'}(x_{I_2}) - \lambda_{I_2})\\
0 &= \theta^{'}(x_J) - \theta^{'}(x_J + F_J(x)),\\
&+ (\nabla F_{J\times J}(x) - \nabla F_{J\times I_2}(x)^T(\nabla F_{I_2\times I_2}(x)^T)^{-1}\nabla F_{I_2\times J}(x)^T)(\theta^{'}(F_J(x)) - \theta^{'}(x_J + F_J(x))).
\end{aligned}$$

The matrix $\nabla F_{J\times J}(x)^T - \nabla F_{J\times I_2}(x)^T(\nabla F_{I_2\times I_2}(x)^T)^{-1}\nabla F_{I_2\times J}(x)^T$ correspond of the Schur complement of the matrix $\nabla F^T_{I_2+J\times I_2+J}$, whose determinant is positive by the P function assumption.

Since the determinant of $\nabla F_{I_2\times I_2}(x)^T$ is also positive, it holds that the Schur complement is also a P matrix. It follows that

$$\left(\nabla F_{J\times J}(x)^T - \nabla F_{J\times I_2}(x)^T(\nabla F_{I_2\times I_2}(x)^T)^{-1}\nabla F_{I_2\times J}(x)^T\right)\left(\theta^{'}(F_J(x)) - \theta^{'}(x_J + F_J(x))\right)$$

is positive componentwise.

Furthermore by concavity of θ it holds that $\theta^{'}(x_J) - \theta^{'}(x_J + F_J(x))$ is also positive. Thus, the last equation cannot hold. This is a contradiction with the existence of the set J. So, the result follows. □

We present here a counter-example to show that we cannot reduce the P function assumption to P_0 function.

Example 3.1. *Let $F(x) = Mx + q$ such that*

$$q = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \text{ and } M = \begin{pmatrix} 0 & 0 \\ 1 & 1 \end{pmatrix}.$$

M is a P_0 matrix. The point $x^* = (1,1)^T$ is clearly not a solution of the NCP, since $F(x) = Mx+q = (0,2)^T$. However it is a stationary point of (P) *with*

$$\lambda = \theta'(2) + \theta'(0) - 2\theta'(3).$$

Following the notations in the previous proof, we have $I_1 = \emptyset, J = 2$ and $I_2 = 1$. The equation on I_2 gives

$$\nabla F_{I_2\times J}(x)(\theta^{'}(F_J(x)) - \theta^{'}(x_J + F_J(x))) + \nabla F_{I_2\times I_2}(x)^T(\theta^{'}(0) - \theta^{'}(x_{I_2}) - \lambda_{I_2}) = 0,$$

since $\nabla F_{I_2\times J}(x) = \nabla F_{I_2\times I_2}(x) = 0$. The equation on J gives

$$\begin{aligned}
0 &= \theta^{'}(x_J) - \theta^{'}(x_J + F_J(x)) + \nabla F_{J\times J}(x)^T(\theta^{'}(F_J(x)) - \theta^{'}(x_J + F_J(x))) + \nabla F_{J\times I_2}(x)^T(\theta^{'}(0) - \theta^{'}(x_{I_2}) - \lambda),\\
&= \theta'(1) - \theta(3) + \theta'(2) - \theta'(3) + \theta'(0) - \theta'(1) - \lambda,
\end{aligned}$$

which holds true by definition of λ.

We now present another example, where the matrix is not a P matrix but a symmetric negative definite.

Example 3.2. *Let $F(x) = Mx + q$ such that*

$$q = \begin{pmatrix} 2 \\ 2 \end{pmatrix} \text{ and } M = \begin{pmatrix} -1 & 0 \\ 0 & -1 \end{pmatrix}.$$

In this case, we have

$$\Theta(x, F(x)) = \theta(x_1) + \theta(x_2) + \theta(-x_1 + 2) + \theta(-x_2 + 2) - \theta(2) - \theta(2).$$

Computing the gradient of the objective function equal to zero yields

$$\begin{aligned} \theta'(x_1) - \theta'(-x_1 + 2) = 0, \\ \theta'(x_2) - \theta'(-x_2 + 2) = 0. \end{aligned}$$

The point $x^* = (1, 1)^T$ satisfies $\nabla\Theta(x^*, F(x^*)) = 0$ and therefore is a stationary point of (P)*. However, this point is not a solution of* (1)*.*

An open question to this fundamental result is the possibility to remove the strict complementarity assumption. We left this question to future research.

4 A DC Formulation of the Sub-Additive Merit Function for the NCP

A function f is called a DC (difference of convex) function if it can be expressed as a difference of convex functions $g : \mathbb{R}^n \to \mathbb{R}$ and $h : \mathbb{R}^n \to \mathbb{R}$ such that

$$f = g - h.$$

The definition remains true if g and h are concave functions. A so-called DC program is that of minimizing a DC function over a convex set. A DC function has infinitely many decomposition g,h but the numerical resolution may be sensitive to this choice.

We split this section in two parts. First, we consider the formulation (P) in the case where F is a concave function. Second, we focus on the general case where we only assume that a DC decomposition of the function F is known.

4.1 The Concave NCP

In this section, we show that if F is a concave function the sub-additive function (3) is a DC function and the optimization problem (P) is a DC program. First, let us prove a simple lemma that shows that the composition of concave functions does not alter the concavity.

Lemma 4.1. *Let f be a concave function on $\mathbb{R}^n$, and let ϕ be a concave function on $\mathbb{R}^n$, which is non-decreasing. Then, $h(x) = \phi(f(x))$ is a concave function.*

Proof. For x and y in $\mathbb{R}^n$ and $0 < \lambda < 1$, we have

$$f((1 - \lambda)x + \lambda y) \geq (1 - \lambda)f(x) + \lambda f(y).$$

Applying ϕ to both sides of the inequality, we get

$$h((1 - \lambda)x + \lambda y) \geq \phi\left((1 - \lambda)f(x) + \lambda f(y)\right).$$

By concavity of the function ϕ it follows

$$h((1 - \lambda)x + \lambda y) \geq \phi\left((1 - \lambda)f(x) + \lambda f(y)\right) \geq (1 - \lambda)h(x) + \lambda h(y).$$

Thus, h is concave. □

We now prove that the objective function of (P) is a DC function.

Theorem 4.1. *Let F be a concave function on $\mathbb{R}^n_+$. Then,* (P) *is a DC program.*

Proof. It is clear by concavity hypothesis on F that the set of constraints of (P) is a convex set. Now, let us verify that the function $\Theta(x, F(x))$ is a DC function with

$$g(x) = -\sum_{i=1}^{n} \theta(x_i + F_i(x)) \text{ and } h(x) = -\sum_{i=1}^{n} \theta(x_i) + \theta(F_i(x)).$$

Since F is a concave function and θ is a non-decreasing concave function we can apply Lemma 4.1 to conclude that g and h are both convex. □

It is to be noted that the hypothesis to take F concave is not so restrictive, since it includes the large family of LCPs.

4.2 The General NCP

In the special case where a DC decomposition of the function F is known we can reformulate (P) as a DC program. Given two convex functions $g, h : \mathbb{R}^n \to \mathbb{R}$ such that

$$F(x) = g(x) - h(x).$$

We can reformulate (P) as

$$\begin{aligned} \inf_{x,y \in \mathbb{R}^n} \; & \Theta(x, y) \\ \text{s.t. } & x \geq 0, y \geq 0, \\ & y = F(x). \end{aligned}$$

Using the fact that a DC decomposition of F is known the previous problem becomes

$$\begin{aligned} \inf_{x,y \in \mathbb{R}^n} \; & \Theta(x, y) \\ \text{s.t. } & x \geq 0, y \geq 0, \\ & y = g(x) - h(x). \end{aligned}$$

Even so we use a DC decomposition of F and the objective function is a DC function, the set of constraints in the previous formulation remains non-convex. To tackle this problem, we try to remove the non-convexity in the constraints using slack variables and then penalizing the remaining non-convex constraints. Therefore, we consider the following penalized problem

$$\begin{aligned} \inf_{x,y,t \in \mathbb{R}^n \times \mathbb{R}^n \times \mathbb{R}^n} \; & \Psi(x, y, t) := \Theta(x, y) + \tau \mathbf{e}^T (t - y - h(x)) + \tau \mathbf{e}^T (t - g(x)) \\ \text{s.t. } & (x, y, t) \in C, \\ & t \geq g(x), \; t - h(x) \geq y, \end{aligned} \qquad (Pen_\tau)$$

where $C := \mathbb{R}^n_+ \times \mathbb{R}^n_+ \times \mathbb{R}^n$. The link between the initial problem (P) and the penalized problem (Pen_τ) will be studied in Theorem 4.3 and 4.4. First, it is straightforward to see that the penalized problem is a DC program.

Theorem 4.2. *Given convex functions $g, h : \mathbb{R}^n \to \mathbb{R}^n$. Then, (Pen_τ) is a DC program.*

Proof. We already shown in Theorem 4.1 that the function $\Theta(x, F(x))$ is a DC function if F is a concave function. However, it is to be noted here that we consider $\Theta(x, y)$, which is therefore a DC function.

Besides by convexity of the functions g and h it follows that the penalty terms in the objective function are concave. Thus, the objective function of (Pen_τ) is a DC function.

A similar argument gives that the constraints of (Pen_τ) are convex constraints. This completes the proof. □

We now investigate the link between the penalized DC program (Pen_τ) and the original problem (P).

Lemma 4.2. *Assume that $\mathcal{F}$ is non-empty. Let (x_τ, y_τ, t_τ) be a solution of (Pen_τ) and $\bar{x}$ be a feasible point of* (P)*, i.e. $\bar{x} \in \mathcal{F}$. Then, for all non-negative τ it holds that*

$$0 \leq \Psi(x_\tau, y_\tau, t_\tau) \leq \Theta(\bar{x}, F(\bar{x})).$$

Proof. Since $\bar{x} \in \mathcal{F}$, it holds that $(\bar{x}, F(\bar{x}), g(\bar{x}))$ is a feasible point for (Pen_τ). Thus, by definition of the triple (x_τ, y_τ, t_τ) we have

$$\Psi(x_\tau, y_\tau, t_\tau) \leq \Theta(\bar{x}, F(\bar{x})) + \tau \mathbf{e}^T(g(\bar{x}) - F(\bar{x}) - h(\bar{x})) + \tau \mathbf{e}^T(g(\bar{x}) - g(\bar{x})),$$

and since $F(\bar{x}) = g(\bar{x}) - h(\bar{x})$ we get

$$\Psi(x_\tau, y_\tau, t_\tau) \leq \Theta(\bar{x}, F(\bar{x})).$$

The non-negativity of the left hand-side is a consequence of Lemma 2.1 and that (x_τ, y_τ, t_τ) is feasible for (Pen_τ). This completes the proof. □

A direct consequence of this lemma is equivalence of global minimum between the penalized DC program (Pen_τ) and the original problem (P).

Theorem 4.3. *Assume that* (1) *has a bounded solution. Then, x^* is a global solution of* (P) *if and only if x^* is a global solution of (Pen_τ).*

Proof. Let x^* be a solution of (P). Such solution exists, since we assume that (1) has at least one bounded solution. Then, the triple $(x^*, F(x^*), g(x^*))$ is feasible for (Pen_τ) and satisfies

$$\Theta(x^*, F(x^*)) + \tau \mathbf{e}^T(g(x^*) - F(x^*) - h(x^*)) + \tau \mathbf{e}^T(g(x^*) - g(x^*)) = 0,$$

by Theorem 2.1. This proves one side of the equivalence.

Now, let x_τ be a global solution (Pen_τ). Then, by Lemma 4.2 we have

$$0 \leq \Theta(x_\tau, y_\tau) + \tau \mathbf{e}^T(t_\tau - y_\tau - h(x_\tau)) + \tau \mathbf{e}^T(t_\tau - g(x_\tau)) \leq \Theta(\bar{x}, F(\bar{x})),$$

for all $\bar{x} \in \mathcal{F}$. In particular, consider $\bar{x}$ to be a solution of (1), then $\bar{x} \in \mathcal{F}$ and by Theorem 2.1 we get

$$0 = \Theta(x_\tau, y_\tau) + \tau \mathbf{e}^T(t_\tau - y_\tau - h(x_\tau)) + \tau \mathbf{e}^T(t_\tau - g(x_\tau)).$$

Noticing that the right hand-side is a sum of positive terms yields $0 = \Theta(x_\tau, y_\tau)$,$y_\tau = F(x_\tau)$ and $t_\tau = g(x_\tau)$. By Theorem 2.1 it follows that x_τ is a solution of (1) and also a solution of (P). This completes the proof. □

We now focus on local minima of the penalized formulation. We notice, as proved in the following lemma, that the function Φ is non-increasing with respect to the second variable.

Lemma 4.3. *Let $\tau > \theta'(0)$ and $x \geq 0$. Then, the function $y \in \mathbb{R} \mapsto \theta(y) - \theta(x + y) + \tau y$ is a non-increasing function.*

Proof. Computing the derivative with respect to y of the function $y \in \mathbb{R}_+ \mapsto \theta(y) - \theta(x+y) + \tau y$ gives

$$\theta'(y) - \theta'(x+y) - \tau.$$

However, this expression is negative since the functions θ are non-increasing and by concavity $\theta'(y) \leq \theta'(0) < \tau$. □

The following result considers the case of local optimality and shows that computing a local minimum of the penalized formulation is sufficient to find a local minimum of the initial formulation.

Theorem 4.4. *Let $\tau > \theta'(0)$. Any local minimum of (Pen_τ) is a local minimum of* (P).

Proof. Let (x_τ, y_τ, t_τ) be a local minimum of (Pen_τ), i.e. there exists $\delta > 0$ such that for all (x, y, t) in the ball of radius δ centered in (x_τ, y_τ, t_τ), denoted $\mathcal{B}_\delta(x_\tau, y_\tau, t_\tau)$, it holds that

$$\Psi(x_\tau, y_\tau, t_\tau) \leq \Psi(x, y, t).$$

By construction, it is clear that any feasible point of the problem (Pen_τ) is also feasible for the problem (P). Thus, it follows that if (x_τ, y_τ, t_τ) satisfies

$$t_\tau = g(x_\tau) \text{ and } y_\tau = g(x_\tau) - h(x_\tau), \tag{6}$$

then it is also a local minimum of (P).

Assume by contradiction that expression in (6) is wrong. In other words, there exists sets $T_t \subset \{1, \dots, n\}$ and $T_y \subset \{1, \dots, n\}$ such that for all $i \in T_t$ and all $j \in T_y$ we have

$$t_{\tau,i} > g_i(x_\tau) \text{ and } y_{\tau,j} < t_{\tau,j} - h_j(x_\tau). \tag{7}$$

Let us consider a point $(x_\tau, \bar{y}, \bar{t})$ such that

$$\bar{t} := \begin{cases} t_{\tau,i} - \bar{\delta}/2, & \text{if } i \in T_t, \\ t_{\tau,i}, & \text{otherwise,} \end{cases}$$

and

$$\bar{y} := \begin{cases} y_{\tau,j} + \bar{\delta}/2, & \text{if } j \in T_y, \\ y_{\tau,j}, & \text{otherwise,} \end{cases}$$

where $\bar{\delta}$ is chosen such that (7) still holds at $(x_\tau, \bar{y}, \bar{t})$.

Obviously, this new point satisfies $(x_\tau, \bar{y}, \bar{t}) \in \mathcal{B}_\delta(x_\tau, y_\tau, t_\tau)$ and so the local optimality of (x_τ, y_τ, t_τ) yields

$$\begin{aligned}
&\Psi(x_\tau, y_\tau, t_\tau) \leq \Psi(x_\tau, \bar{y}, \bar{t}), \\
&\Theta(x_\tau, y_\tau) + \tau \mathbf{e}^T(t_\tau - y_\tau - h(x_\tau)) + \tau \mathbf{e}^T(t_\tau - g(x_\tau)) \leq \Theta(x_\tau, \bar{y}) + \tau \mathbf{e}^T(\bar{t} - \bar{y} - h(x_\tau)) + \tau \mathbf{e}^T(\bar{t} - g(x_\tau)), \\
&\Theta(x_\tau, y_\tau) - \Theta(x_\tau, \bar{y}) + \tau \mathbf{e}^T(t_\tau - y_\tau) - \tau \mathbf{e}^T(\bar{t} - \bar{y}) + \tau \mathbf{e}^T t_\tau - \tau \mathbf{e}^T \bar{t} \leq 0, \\
&\sum_{j \in T_j} \theta(y_{\tau,j}) - \theta(\bar{y}_j) - \theta(x_{\tau,j} + y_{\tau,j}) + \theta(x_{\tau,j} + \bar{y}_j) + \tau(\bar{y}_j - y_{\tau,j}) + 2\tau \sum_{i \in T_t} t_{\tau,i} - \bar{t}_i \leq 0, \\
&\sum_{j \in T_j} \theta(y_{\tau,j}) - \theta(\bar{y}_j) - \theta(x_{\tau,j} + y_{\tau,j}) + \theta(x_{\tau,j} + \bar{y}_j) + \tau(\bar{y}_j - y_{\tau,j}) + 2\tau \sum_{i \in T_t} \tau\bar{\delta}/2 \leq 0\ .
\end{aligned}$$

This, however, leads to a contradiction since $\tau, \delta > 0$ and that for all $j \in T_j$

$$\theta(y_{\tau,j}) - \theta(\bar{y}_j) - \theta(x_{\tau,j} + y_{\tau,j}) + \theta(x_{\tau,j} + \bar{y}_j) + \tau(\bar{y}_j - y_{\tau,j}) > 0,$$

since the function $y \mapsto \theta(y) - \theta(x+y) + \tau y$ is non-increasing for all $x, y \geq 0$ and $\tau > \theta'(0)$ by Lemma 4.3. It follows that $T_t = \emptyset$ and $T_y = \emptyset$. Thus, (x_τ, y_τ, t_τ) satisfies (6) and so it is a local minimum of (P). □

5 Difference of Convex Programming

In this section, we introduce some fundamental properties of a DC program and an algorithm to tackle this problem so-called DC Algorithm (DCA). For a complete study on this subject, the readers are referred to [2, 21, 22] and references herein.

5.1 DC Algorithm

Given two proper lower semi-continuous convex functions $\bar{g}, h$, a DC program consists in finding $x \in \mathbb{R}^n$ such that

$$\inf_{x\in\mathbb{R}^n} g(x) - h(x). \tag{8}$$

In general this unconstrained formulation encompasses convex constraints by considering $g(x) = g_0(x) + \mathcal{X}_C(x)$, where g_0 is a proper lower semi-continuous convex function, C a closed convex set and $\mathcal{X}_C(x)$ is the convex indicator function $\mathcal{X}_C(x) = \{0 \text{ if } x \in C,\ +\infty \text{ if } x \notin C\}$.

Recall that for a lower semi-continuous convex function ψ on $\mathbb{R}^n$ and $\bar{x} \in dom(\psi) := \{x \mid \psi(x) < \infty\}$, the subdifferential of ψ at $\bar{x}$ is given by

$$\partial\psi(\bar{x}) = \{y \in \mathbb{R}^n \mid \psi(x) \geq \psi(\bar{x}) + (x - \bar{x})^T y,\ \forall x \in \mathbb{R}^n\}.$$

In the special case, where ψ is differentiable function, the subdifferential is reduced to a singleton and $\partial\psi(\bar{x}) = \nabla\psi(\bar{x})$. As an example the subdifferential of the convex indicator function for $x \in C$ is given as

$$\begin{aligned}\partial\mathcal{X}_C(x) &= \begin{cases}\{0\},\ \forall x \in int(C),\\ \{d \in \mathbb{R}^n \mid d^T x \leq 0\},\ \text{if } x \in bd(C)\end{cases}\\ &= \{d \in \mathbb{R}^n \mid d^T s \leq 0, \forall s \in C\} := \mathcal{N}_C(x).\end{aligned}$$

The notation $\mathcal{N}_C(x)$ stands for the normal cone of the set C at x.

The necessary local optimality condition for a DC program is

$$\partial h(x^*) \subset \partial g(x^*)\ .$$

We say that x^* satisfies the generalized Kuhn-Tucker condition, if it satisfies

$$\partial h(x^*) \cap \partial g(x^*) \neq \emptyset\ .$$

We also call such a point a critical point.

DCA is an algorithm that is based on a primal-dual formulation of (8). Let the conjugate function of g be defined as

$$g^*(y) := \sup_{x\in\mathbb{R}^n} x^T y - g(x).$$

Then, the dual of (8) is defined as

$$\inf_{y\in\mathbb{R}^n} h^*(y) - g^*(y). \tag{9}$$

There is no duality gap as proven in [2]. Furthermore, the following result from [2, 21, 22] gives a characterization of the solutions of both problems. Let P_{dc} (resp. D_{dc}) denotes the set of solutions of the primal DC program (8) (resp. the dual DC program (9))

Proposition 5.1. *Given g, h two proper lower semi-continuous convex functions. It holds true that*

$$[\cup_{y^*\in D_{dc}} \partial g^*(y^*)] \subset P_{dc}$$

and

$$[\cup_{x^*\in P_{dc}} \partial h(x^*)] \subset D_{dc}.$$

```
$x^0 \in \mathbb{R}^n$ an initial point;
$k := 0$ ;
while $k$==0 or $\|x^k - x^{k-1}\|_\infty > \epsilon$ do
    $y^k \in \partial h(x^k)$ ;
    $x^{k+1} \in \arg\min\{g(x) - h(x^k) - (x - x^k)^T y^k \ : \ x \in \mathbb{R}^n\}$ ;
    $k := k + 1$;
```

Algorithm 1: Generic DCA

Based on local optimality conditions and duality in DCA, the idea of DCA is quite simple: each iteration k of DCA approximates the concave part $-h$ by its affine majorization (that corresponds to taking $y^k \in \partial h(x^k)$) and minimizes the resulting convex function (that is equivalent to compute $x^{k+1} \in \partial g^*(y^k)$). The algorithm is stated explicitly in Algorithm 1.

Convergence properties and its theoretical basis has been analyzed in [2, 21, 22]. We remind here the fundamental facts. DCA is a descent method without line search, thus the sequence $\{g(x^k) - h(x^k)\}$ is a decreasing sequence. So, if the optimal value of (8) is finite, then the sequence $\{x^k\}$ converges. If $g(x^k) - h(x^k) = g(x^{k+1}) - h(x^{k+1})$ or equivalently $x^k = x^{k+1}$ then x^k is a critical point of $g - h$ and the algorithm terminates at iteration k.

Based on the results from Section 4.1 and Section 4.2, DCA can be used to solve the concave NCP and the general NCP.

5.2 DCA for the Concave NCP

We now make the link between DCA and Theorem 3.1. We remind that this theorem proves that a stationary point of (P) with F a concave differentiable P function is the solution of the NCP assuming such a solution exists. The following sequence of results states that under classical assumptions DCA converges to a stationary point of (P).

Lemma 5.1. *Let x^* be a critical point of* (8)*, where we assume that $\bar{g}(x) = g(x) + \xi_C$ with g, h are differentiable lower semi-continuous convex functions and C is a closed convex set. It holds true that*

$$0 \in \nabla g(x^*) + \mathcal{N}_C(x^*) - \nabla h(x^*),$$

where $\mathcal{N}$ is the normal cone of C at x^*.

Proof. x^* is assumed to be a critical point of (8), i.e.

$$0 \in \partial \bar{g}(x^*) - \partial h(x^*).$$

Now, by construction of $\bar{g}$ we have

$$0 \in \partial g(x^*) + \mathcal{N}_C(x^*) - \partial h(x^*).$$

The result follows by the differentiability assumption on g and h. □

Lemma 5.2. *Let x^* be a critical point of* (8)*, where we assume that $\bar{g}(x) = g(x) + \xi_C$ with g, h are differentiable lower semi-continuous convex functions and C is a closed convex set. Furthermore, assume that*

$$\mathcal{N}_C(x^*) = \mathscr{L}^\circ(x^*), \tag{10}$$

where

$$\mathscr{L}^\circ(x^*) := \{d \in \mathbb{R}^n \mid d = -\nabla F(x^*)^T \lambda - \mu \ \textit{with} \ \min(F(x^*), \lambda) = 0, \min(x^*, \mu) = 0\}.$$

Then, x^* is stationary point of (P)*.*

The cone $\mathscr{L}^{\circ}(x^*)$ is sometimes called the polar cone of the linearized cone in the literature related to Karush-Kuhn Tucker conditions, while the condition (10) refers to the Guignard constraint qualification that is known to be the weakest constraint qualification in non-linear programming.

Proof. In Theorem 3.1, we define a stationary point of (P) as a point $\bar{x}$ such that there exists Lagrange multipliers $(\lambda, \mu) \in \mathbb{R}^{n \times n}$ that satisfies

$$\begin{array}{l} \nabla g(x^*) - \nabla h(x^*) - \nabla F(x^*)^T \lambda - \mu = 0 \\ \lambda^T F(x^*) = 0, F(x^*) \geq 0, \lambda \geq 0 \\ (x^*)^T \mu = 0, x^* \geq 0, \mu \geq 0 \ . \end{array}$$

This can be equivalently written as

$$0 \in \nabla g(x^*) - \nabla h(x^*) + \mathscr{L}^{\circ}(x^*).$$

Now, using assumption (10) we get

$$0 \in \nabla g(x^*) - \nabla h(x^*) + \mathcal{N}_C(x^*).$$

However, this condition is satisfied by any critical point x^* according to Lemma 5.1. □

We can now conclude by the following strong result for DC Algorithm applied to P, which has been shown to be a DC program in Theorem 4.1.

Theorem 5.1. *Let F be a continuously differentiable concave P-function. Assume that a solution of* (1) *exists. Let $\{x^k\}$ be a sequence generated by Algorithm 1 on P. Then, any limit point, up to a subsequence, of $\{x^k\}$ that satisfies the strict complementarity condition is a solution of* (1).

Proof. Assuming that F is a P function and that there exists a solution to (1) yields that this solution is unique. Thus, P attains its minimum, whose optimal value is 0.

The optimal value is bounded and the sequence $\{g(x^k) - h(x^k)\}$ is a decreasing sequence, so Algorithm 1 converges to a critical point. Applying Lemma 5.2, we get that the limit point is also a stationary point of P. Indeed, we have proved in Lemma 3.1 that some constraint qualification holds at any feasible point of the problem, which is known to imply (10).

The result follows by applying Theorem 3.1. □

6 Numerics

Through this article, we studied two DC approaches to tackle the NCP. In this section, we present three different experiments applying DCA on the DC formulations that have been introduced in the previous sections, equation (P) for the concave NCP and the penalized formulation (Pen_τ) for the general NCP.

First, we present a comparison on a list of LCP problems with other DC approaches that have been suggested recently in [16]. Then, we study an adaptation of our approach to the absolute value equation and present a comparison of existing methods as in [1]. These two families of problems belong to the class of concave NCPs. Finally, we give an example of the penalization technique from Section 4.2 on a non-concave NCP.

The algorithms are coded in *Matlab 2014b* on a standard laptop. The convex sub-problems of DCA, Algorithm 1, are solved using *CVX* [12].

6.1 Comparisons of DC Methods for LCP

We compare our method denoted TDC with other DC approaches developed in [16] denoted as in the paper DCA1, DCA2, DCA3 and DCA4 with respective initial points $x = 0$ for DCA1 and DCA2, and $x = (0, b)$ for DCA3 and DCA4. DCA1 and DCA4 are iterative methods with quadratic convex sub-problems that we solve using *CVX*. DCA2 is an iterative method with linear sub-problems that we solve using *linprog*. DCA3 is an iterative method with standard operations. The initial point for TDC is set here at $x = 0$. We use the function $\theta^1 = t/(t+1)$ to define the merit function in TDC.

We run our comparison on a set of 25 problems that are described in Appendix A. This set includes the original problems presented in [16] and some given in [10]. Instances LCP10 to LCP14 are studied for dimensions 300, 500 and 1000. The precision is set as $\epsilon = 10^{-5}$.

The comparative results are given in Table 1 and Table 2. We are interested in the following aspects: the residuals, computed as $\max(x^T(Mx+q), \|\min(x,0)\|_\infty, \|\min(Mx+q,0)\|_\infty)$, the number of iterations and the time. No attempt has been made to optimize the performance of the algorithm, since our aim is to validate our approach and run a preliminary comparison with other methods. Thus, our main concern is focus on the residuals and in particular on the number of solved problems.

Pb.	TDC	DCA1	DCA2	DCA3	DCA4
LCP0	0.000000	0.000000	0.000000	0.000000	0.000000
LCP1	0.000000	0.000000	0.000000	0.010928	0.000000
LCP2	0.000000	3.000000	1.191969	0.183023	0.000000
LCP3	0.000000	2.000000	14056.152537	1.000000	0.000000
LCP4	0.000000	0.000000	0.000000	0.000000	0.000000
LCP5	0.000000	0.000000	30.422537	0.005219	0.000000
LCP6	0.000000	0.000000	453.572844	0.205609	0.000014
LCP7	0.000000	0.000000	0.000000	0.000000	0.000000
LCP8	0.000000	0.000000	0.000000	0.000000	0.000173
LCP9	0.000000	0.000000	0.000000	0.000000	0.000000
LCP10	0.000000	0.000000	0.000000	0.996569	0.000000
LCP10	0.000000	0.000000	0.000000	0.997942	0.000000
LCP10	0.000000	0.000000	0.000000	0.998971	0.000004
LCP11	0.000000	0.000000	2136417.272448	0.000655	0.000000
LCP11	0.000000	0.000001	3567004.210504	0.000489	0.000000
LCP11	0.000000	0.000000	7143507.495324	0.000339	0.000000
LCP12	0.000000	0.000000	3620054.331657	0.001033	0.000000
LCP12	0.000000	0.000000	6014566.787304	0.000791	0.000000
LCP12	0.000000	0.000000	12000859.009675	0.000571	0.000000
LCP13	0.000000	0.000000	41406.616317	0.742913	0.000000
LCP13	0.000000	0.000000	51818.790221	0.857770	0.000000
LCP13	0.000000	0.000000	53590.477924	0.938882	0.000002
LCP14	0.000000	0.000000	3457018.895431	0.997783	0.007906
LCP14	0.000000	0.000009	5431724.049730	0.999193	0.012113
LCP14	0.000000	0.000004	11314010.430384	0.999793	0.017590

Table 1: Comparison on the residuals $\max(x^T(Mx+q), \|\min(x,0)\|_\infty, \|\min(Mx+q,0)\|_\infty)$ of five DC methods.

Table 1 shows that the method TDC is the only one to solve all the problems. DCA1 and DCA4 only fail on respectively 2 (LCP2, LCP3) and 5 problems (LCP6,LCP8, LCP14), and LCP6 is almost solved using DCA4. These experiments already confirm the validity of our approach.

In Table 2, we expect DCA1 and DCA4 to be faster than TDC, since they only solved convex quadratic sub-problems compared to convex problems. This is confirmed by the results. However, for a large part of the problems, TDC remains in the same order of time as the other methods. For 4 instances (LCP6 and the three LCP10), TDC spend significantly less time than DCA4.

Besides, it is interesting to note that for most of the instances (22/25), TDC requires only one iteration.

Pb.	TDC	DCA1	DCA2	DCA3	DCA4
LCP0	1 (4.310494)	0 (0.000636)	1 (2.255981)	1 (0.000052)	1 (0.655963)
LCP1	1 (0.087655)	1 (0.077509)	1 (0.016695)	116 (0.001562)	1 (0.071924)
LCP2	9 (0.981270)	2 (0.150380)	2 (0.009943)	1057 (0.012527)	4 (0.366921)
LCP3	18 (1.828030)	2 (0.162421)	2 (0.010436)	1002 (0.012531)	11 (1.041105)
LCP4	1 (0.104623)	0 (0.000037)	1 (0.012372)	1 (0.000063)	1 (0.093485)
LCP5	1 (0.098558)	1 (0.090425)	2 (0.073538)	46 (0.000611)	1 (0.076413)
LCP6	2 (0.188896)	1 (0.079031)	2 (0.010363)	18498 (0.223561)	24 (2.416563)
LCP7	1 (0.129127)	0 (0.000028)	1 (0.006391)	1 (0.000036)	1 (0.142373)
LCP8	1 (0.098097)	0 (0.000024)	1 (0.005906)	1 (0.000038)	3 (0.375995)
LCP9	1 (0.118442)	0 (0.000027)	1 (0.008533)	1 (0.000061)	1 (0.121109)
LCP10	1 (2.032424)	1 (0.782662)	1 (0.658506)	3 (0.000803)	1 (6.102593)
LCP10	1 (6.511099)	1 (1.754187)	1 (2.801324)	3 (0.004826)	1 (31.286924)
LCP10	1 (49.815355)	1 (8.815220)	1 (24.645624)	3 (0.020579)	1 (426.055509)
LCP11	1 (0.127772)	1 (0.151737)	2 (0.036948)	101 (0.024243)	1 (0.121978)
LCP11	1 (0.150208)	1 (0.109892)	2 (0.064879)	105 (0.158833)	1 (0.140689)
LCP11	1 (0.216863)	1 (0.162223)	2 (1.620359)	110 (0.721783)	1 (0.222982)
LCP12	1 (0.125809)	1 (0.104318)	2 (0.026417)	232 (0.042913)	1 (0.119623)
LCP12	1 (0.130139)	1 (0.117876)	2 (0.061030)	241 (0.396976)	1 (0.128972)
LCP12	1 (0.202222)	1 (0.158284)	2 (1.618803)	252 (1.711475)	1 (0.209815)
LCP13	1 (2.881841)	1 (0.249691)	2 (0.489375)	12456 (2.689871)	1 (1.258522)
LCP13	1 (15.979708)	1 (0.744162)	2 (2.049274)	8376 (12.453357)	1 (7.303410)
LCP13	1 (211.015326)	1 (3.918635)	2 (5.199419)	5346 (34.129436)	1 (74.023011)
LCP14	1 (0.137329)	1 (0.136163)	2 (0.028150)	20000 (5.148445)	3 (0.471519)
LCP14	1 (0.192593)	1 (0.150660)	2 (0.092064)	20000 (32.703992)	3 (0.583343)
LCP14	1 (0.235936)	1 (0.217084)	2 (1.619427)	20000 (126.621461)	3 (0.889027)

Table 2: Comparison on the number of iterations and time (in parentheses) in seconds of five DC methods.

The three remaining problems seem to be difficult, since it is LCP2 and LCP3, the two failures of DCA1, and LCP6, which is only approximately solved by DCA4 with 24 iterations.

All in all, these preliminary results are very encouraging. Further research should explore optimization of the performance and consider instances with very large dimension.

6.2 Application to Absolute Value Equation and Comparison

In [1], the authors propose a smoothing technique based on a complementarity reformulation of the absolute value equation, defined as

$$Ax - |x| = b, \tag{11}$$

with $A \in \mathbb{R}^{n\times n}$ and $b \in \mathbb{R}^n$.

In particular, their motivation was to solve the absolute value equation without any assumption on the data except existence of at least one solution. This is the same motivation as in this article. Using the same technique as in [1], (11) can be cast as the following complementarity problem

$$A(x^+ - x^-) - (x^+ + x^-) = b, \; 0 \leq x^- \perp x^+ \geq 0. \tag{12}$$

Now, applying the technique proposed in this article yields to consider the following sub-additive regularized formulation

$$\begin{aligned}\min_{(x^-,x^+)\in\mathbb{R}^n\times\mathbb{R}^n} \quad & \Theta(x^-,x^+)\\ \text{s.t.}\ & A(x^+-x^-)-(x^++x^-)=b,\\ & (x^-,x^+)\geq 0.\end{aligned} \tag{13}$$

Then, we can solve (11) using DC approach discussed earlier on the problem (13).

The following heuristic mentioned in [1] can be rather useful to accelerate convergence and assure a good precision when we are close to the solution. After finding the current point x^k we solve if possible the linear system

$$(A-\text{diag}(\delta(x^k)))z=b. \tag{14}$$

If x solves (11), then we solved (11) with the same precision as we solved the linear system. However, if x does not solve (11), we continue the iteration in r with x^k. This idea is similar to compute a Newton iteration.

We generate for several n and several values of the parameters one hundred instances of the problem following [17]:

"Choose a random A from a uniform distribution on $[-10,10]$, then choose a random x from a uniform distribution on $[-1,1]$ and set $b=Ax-|x|$."

The difficulty here relies on the fact that the problem may not be uniquely solvable and thus quite hard to solve. The precision is set as 10^{-6}. We compare 4 methods tailored for general absolute value equations:

- TDC-AVE, which is the DCA on (13) with $\theta(t)=t/(t+0.1)$;
- TAVE method from [1];
- concave minimization method CMM from [17];
- successive linear programming method LPM from [18].

The initial point used for TDC-AVE is obtained by solving the following linear program:

$$\begin{aligned}\min_{(x^+,x^-)\in\mathbb{R}^n\times\mathbb{R}^n} \quad & x^++x^-\\ \text{s.t.}\ & x^+\geq 0, x^-\geq 0, A(x^+-x^-)-(x^++x^-)=b.\end{aligned}$$

This initial point was already proposed in [1].

Due to the potential difficulty of the equation, we focus on the number of unsolved problems. We do not include any Newton-kind method in our comparison, since they are not applicable without additional assumptions on A.

n	TDC-AVE	TAVE	CMM	LPM (LAVE)
32	0	0	9	7
64	0	3	8	13
128	1	8	10	13
256	0	8	11	11
512	0	11	8	10
1024	0	16	7	9

Table 3: Comparison on the number of unsolved problems for 100 hundred randomly generated AVE of size n.

n	TDC-AVE ite.(time)	TAVE ite.(time)	CMM ite.(time)	LPM ite.(time)
32	204 (36.9269)	306(0.5783)	485(1.8362)	248 (0.7955)
64	134(32.9252)	491(2.7602)	458(2.9125)	342(2.5223)
128	207 (108.1083)	841(20.9437)	568(19.0521)	409 (17.0269)
256	143 (358.7079)	1129 (361.3884)	595 (182.7998)	445 (182.0419)
512	127 (2025.1)	1469 (3712.2)	499 (1462.6)	400 (1214.2)
1024	111 (18587)	1993 (37356)	479 (11513)	452 (13565)

Table 4: Comparison on the number of iterations and time in seconds for 100 hundred randomly generated AVE of size n.

Table 3 shows the number of unsolved problems among 100 problems for each value of n. Table 4 presents the number of iterations and the time required for each method to solve the 100 problems for each value of n.

Overall, these results confirm the validity of our algorithm, since TDC-AVE solves most of the problems in each case (1 failure over 600 instances) and improves significantly the existing methods.

For now, the obvious drawback of this approach is that it is more computationally difficult, since we solve at each step a convex program instead of a linear program. However, it is to be noted that when the dimension grows the gap is reduced. Although, it was not our main goal in this article and we left this for further research.

6.3 A Numerical Example of DC Penalization on a General NCP

In this section, we use the theoretical analysis derived in Section 4.2 to solve the NCP. Let $F : \mathbb{R}^n \to \mathbb{R}^n$.

We proposed in Theorem 4.2 a DC program, which can be interpreted as a penalization of the formulation of the NCP using the sub-additive merit function, (P). The technique used here is to apply DC Algorithm presented in the previous section to the problem (Pen_τ). This approach is only a heuristic, although Theorem 3.1 and Theorem 4.4 tend to give the intuition that it should be a good one.

In order to validate our approach, we report the results of this approach applied to an example from [10] and initially proposed in [15], which consider the NCP with

$$F(x) = \begin{pmatrix} 3x_1^2 + 2x_1x_2 + 2x_2^2 + x_3 + 3x_4 - 6 \\ 2x_1^2 + x_2^2 + x_1 + 10x_3 + 2x_4 - 2 \\ 3x_1^2 + x_1x_2 + 2x_2^2 + 2x_3 + 9x_4 - 9 \\ x_1^2 + 3x_2^2 + 2x_3 + 3x_4 - 3 \end{pmatrix}.$$

Two solutions are known: $x^* = (\sqrt{6}/2, 0, 0, 0.5)$ and $\bar{x} = (1, 0, 3, 0)$. The difficulty in solving this problem arises when a Newton-type method is used, since the LCP formed by linearizing F around $x = 0$ has no solution.

Note that all the components of F are convex functions, and so a trivial DC decomposition of F is given by

$$F(x) = g(x) - h(x), \text{ with } g(x) = F(x) \text{ and } h(x) = 0.$$

We choose five random initial points x_0 from a uniform distribution on $[-1, 1]$, we take $\tau_0 = \tau_1 = 1/(r + 1), r = 0.1$ and the precision $\epsilon = 1e - 5$. Results are given in Table 5, where we reported the complementarity $(x^T F(x))$, the solution obtained (sol), the smallest component of F at the solution, the number of iterations (nb-iter), the distance between y and $F(x)$ and the distance between t and $g(x)$.

In these five tests, we have a convergence to one of the two solutions x^* and $\bar{x}$ of the problem. It is interesting to notice that as expected by the theoretical study, when the problem is solved $t = g(x)$ and $y = F(x)$. The results obtained validate the theoretical part and confirm our approach.

x^0	$(x^k)^T(g(x^k)-h(x^k))$	sol	$\min_i F_i(x)$	nb-iter	$\|y-F(Sol)\|$	$\|t-g(Sol)\|$
(0.4039;0.0965; 0.1320 ;0.9421)	0	x^*	0	36	0	0
(0.3532;0.8212;0.0154;0.0430)	0	x^*	0	98	0	0
(0.0497;0.9027;0.9448;0.4909)	0	$\bar{x}$	0	734	0	0
(0.4893;0.3377;0.9001;0.3692)	0	x^*	0	90	0	0
(0.6256;0.7802;0.0811;0.9294)	0	x^*	0	61	0	0

Table 5: Sub-additive DC penalization approach to solve the general NCP for five initial points.

7 Conclusion

In this paper, we have proposed a new formulation based on sub-additive merit functions for solving nonlinear complementarity problems. We have proved that the reformulated problem is DC program, whenever F is a concave or a DC decomposition of F is known. Thus, we can use the local algorithm DCA to solve the problem.

We also proved that in order to solve the monotone concave NCP, it is sufficient to computing a stationary point of the problem. Besides, in the general case, without any assumption on F except that a DC decomposition is known, we propose a penalization formulation, whose local minima are solutions of the NCP.

Numerical experiments on several LCP problems and a comparison with other DC approaches prove the efficiency of our method. We have presented an application of absolute value equation and shows that our method proposed is promising. Finally, we also give some numerical results for an example when F is non-linear to validate our penalization approach.

References

[1] Lina Abdallah, Mounir Haddou, and Tangi Migot. Solving absolute value equation using complementarity and smoothing functions. *Journal of Computational and Applied Mathematics*, 327:196 – 207, 2018.

[2] Le Thi Hoai An and Pham Dinh Tao. The dc (difference of convex functions) programming and dca revisited with dc models of real world nonconvex optimization problems. *Annals of operations research*, 133(1):23–46, 2005.

[3] Chunhui Chen and Olvi L. Mangasarian. Smoothing methods for convex inequalities and linear complementarity problems. *Mathematical Programming*, 71(1):51–69, Nov 1995.

[4] Chunhui Chen and Olvi L. Mangasarian. A class of smoothing functions for nonlinear and mixed complementarity problems. *Computational Optimization and Applications*, 5(2):97–138, Mar 1996.

[5] Richard W. Cottle, Jong-Shi Pang, and Richard E. Stone. *The linear complementarity problem.* SIAM, 2009.

[6] Tecla De Luca, Francisco Facchinei, and Christian Kanzow. A semismooth equation approach to the solution of nonlinear complementarity problems. *Mathematical Programming*, 75(3):407–439, Dec 1996.

[7] Francisco Facchinei and Jong-Shi Pang. *Finite-dimensional variational inequalities and complementarity problems.* Springer Science & Business Media, 2007.

[8] Francisco Facchinei and Joo Soares. A new merit function for nonlinear complementarity problems and a related algorithm. *SIAM Journal on Optimization*, 7(1):225–247, 1997.

[9] Andreas Fischer. Solution of monotone complementarity problems with locally lipschitzian functions. *Mathematical Programming*, 76(3):513–532, Mar 1997.

[10] Christodoulos A Floudas, Panos M Pardalos, Claire Adjiman, William R Esposito, Zeynep H Gümüs, Stephen T Harding, John L Klepeis, Clifford A Meyer, and Carl A Schweiger. *Handbook of test problems in local and global optimization*, volume 33. Springer Science & Business Media, 2013.

[11] Masao Fukushima. *Merit Functions for Variational Inequality and Complementarity Problems*, pages 155–170. Springer US, Boston, MA, 1996.

[12] Michael Grant, Stephen Boyd, and Yinyu Ye. Cvx: Matlab software for disciplined convex programming, 2008.

[13] Mounir Haddou and Patrick Maheux. Smoothing methods for nonlinear complementarity problems. *Journal of Optimization Theory and Applications*, 160(3):711–729, 2014.

[14] Christian Kanzow. Some noninterior continuation methods for linear complementarity problems. *SIAM Journal on Matrix Analysis and Applications*, 17(4):851–868, 1996.

[15] Masakazu Kojima and Susumu Shindoh. *Extensions of Newton and quasi-Newton methods to systems of PC 1 equations.* Inst. of Technology, Department of Information Sciences, 1986.

[16] Hoai An Le Thi and Tao Pham Dinh. On solving linear complementarity problems by dc programming and dca. *Computational Optimization and Applications*, 50(3):507–524, 2011.

[17] Olvi L. Mangasarian. Absolute value equation solution via concave minimization. *Optimization Letters*, 1(1):3–8, 2007.

[18] Olvi L. Mangasarian. Absolute value equation solution via linear programming. *Journal of Optimization Theory and Applications*, 161(3):870–876, 2014.

[19] Tangi Migot and Mounir Haddou. A smoothing method for sparse optimization over polyhedral sets. In *Modelling, computation and optimization in information systems and management sciences*, pages 369–379. Springer, 2015.

[20] Liqun Qi, Defeng Sun, and Guanglu Zhou. A new look at smoothing newton methods for nonlinear complementarity problems and box constrained variational inequalities. *Mathematical Programming*, 87(1):1–35, Jan 2000.

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

[22] Pham Dinh Tao and Le Thi Hoai An. A dc optimization algorithm for solving the trust-region subproblem. *SIAM Journal on Optimization*, 8(2):476–505, 1998.

[23] Pham Dinh Tao and El Bernoussi Souad. *Duality in D.C. (Difference of Convex functions) Optimization. Subgradient Methods*, pages 277–293. Birkhäuser Basel, Basel, 1988.

A Description of LCP Test Problems from Section 6.1

In this appendix, we describe the datas of the LCP used in the numerical tests in Section 6.1.

LCP0

$$M = \begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix} \text{ and } q = \begin{pmatrix} 1 & 1 \end{pmatrix}^T .$$

LCP1

$$M\text{as in LCP0 and } q = \begin{pmatrix} -1 & -1 \end{pmatrix}^T.$$

LCP2

$$M = \begin{pmatrix} 0 & -1 & 2 \\ 2 & 0 & -2 \\ -1 & 1 & 0 \end{pmatrix} \text{ and } q = \begin{pmatrix} 3 & -6 & 1 \end{pmatrix}^T.$$

LCP3

$$M \text{ as in LCP2 and } q = \begin{pmatrix} -3 & 6 & -1 \end{pmatrix}^T.$$

LCP4

$$M = \begin{pmatrix} 0 & 0 & 10 & 20 \\ 0 & 0 & 30 & 15 \\ 10 & 20 & 0 & 0 \\ 30 & 15 & 0 & 0 \end{pmatrix} \text{ and } q = \begin{pmatrix} 1 & 1 & 1 & 1 \end{pmatrix}^T.$$

LCP5

$$M \text{ as in LCP4 and } q = \begin{pmatrix} -1 & -1 & -1 & -1 \end{pmatrix}^T.$$

LCP6

$$M = \begin{pmatrix} 11 & 0 & 10 & -1 \\ 0 & 11 & 10 & -1 \\ 10 & 10 & 21 & -1 \\ 1 & 1 & 1 & 1 \end{pmatrix} \text{ and } q = \begin{pmatrix} 50 & 50 & 23 & -6 \end{pmatrix}^T.$$

LCP7

$$M = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & -1 & 1 \end{pmatrix} \text{ and } q = \begin{pmatrix} 0 & 0 & 1 \end{pmatrix}^T.$$

LCP8

$$M \text{ as in LCP6 and } q = \begin{pmatrix} 50 & 50 & 1 & 6 \end{pmatrix}^T.$$

LCP9

$$M = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & -2 \\ 0 & 2 & -1 \end{pmatrix} \text{ and } q = \begin{pmatrix} 0 & 0 & 1 \end{pmatrix}^T.$$

LCP10

$$M = \begin{pmatrix} 1 & 2 & 2 & \dots & 2 \\ 2 & 5 & 6 & \dots & 6 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 2 & 6 & 10 & \dots & 4n-3 \end{pmatrix} \text{ and } q = \begin{pmatrix} -1 & -1 & \dots & -1 \end{pmatrix}^T.$$

LCP11

$$M = \begin{pmatrix} 4 & -2 & 0 & \dots & 0 \\ 1 & 4 & -2 & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & 4 \end{pmatrix} \text{ and } q = \begin{pmatrix} -1 & -1 & \dots & -1 \end{pmatrix}^T.$$

LCP12

$$M = \begin{pmatrix} 4 & -1 & 0 & \dots & 0 & 0 \\ -1 & 4 & -1 & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & -1 & 4 \end{pmatrix} \text{ and } q = \begin{pmatrix} -1 & -1 & \dots & -1 \end{pmatrix}.$$

LCP13

$$M = \begin{pmatrix} 1 & 2 & 2 & \dots & 2 \\ 0 & 1 & 2 & \dots & 2 \\ 0 & 0 & 1 & \dots & 2 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & 1 \end{pmatrix} \text{ and } q = \begin{pmatrix} -1 & -1 & \dots & -1 \end{pmatrix}.$$

LCP14

$$M = diag(1/n, 2/n, \dots, 1) \text{ and } q = \begin{pmatrix} -1 & -1 & \dots & -1 \end{pmatrix}^T$$