J. Nonlinear Var. Anal. 7 (2023), No. 1, pp. 103-128
Available online at http://jnva.biemdas.com
https://doi.org/10.23952/jnva.7.2023.1.07

OPTIMIZATION CONDITIONS AND DECOMPOSABLE ALGORITHMS FOR CONVERTIBLE NONCONVEX OPTIMIZATION

MIN JIANG[1], RUI SHEN[2,*], ZHIQING MENG[1], CHUANGYIN DANG[3]

[1]*School of Management, Zhejiang University of Technology, Hangzhou 310023, China*
[2]*School of Economics, Zhejiang University of Technology, Hangzhou 310023, China*
[3]*Department of Advanced Design and System Engineering, City University of Hong Kong, Kowloon, Hong Kong*

Abstract. This paper defines a convertible nonconvex function (CN function for short) and a weak (strong) uniform (decomposable, exact) CN function, proves the optimization conditions for their global solutions, and proposes algorithms for solving the unconstrained optimization problems with decomposable CN functions. First, to illustrate the fact that some nonconvex functions, nonsmooth or discontinuous, are actually weak uniform CN functions, examples are given. The operational properties of CN functions are proved, including addition, subtraction, multiplication, division, and compound operations. Second, optimization conditions of the global optimal solution to the unconstrained optimization with weak uniform CN function are proved. Based on the unconstrained optimization problem with decomposable CN functions, a decomposable algorithm is proposed by its augmented Lagrangian penalty function and its convergence is proved. Numerical results demonstrate that an approximate global optimal solution to unconstrained optimization with CN function may be obtained by the decomposable algorithm. The decomposable algorithm can effectively reduce the scale in solving the unconstrained optimization problem with decomposable CN function. This paper provides a new idea for solving unconstrained nonconvex optimization problems.

Keywords. Decomposable algorithm; Optimization conditions; Unconstrained optimization problems; Weak uniform convertible nonconvex function.

1. INTRODUCTION

In this paper, the following unconstrained optimization (convertible nonconvex optimization, CNO) with weak uniform (decomposable) convertible nonconvex (CN) function is considered:

$$\text{(CNO)} \qquad \min \quad f(\boldsymbol{x}) \qquad \text{s.t.} \quad \boldsymbol{x} \in R^n,$$

where $f : R^n \to R$ is neither convex nor smooth. In machine learning, there are many nonconvex, nonsmooth, non-Lipschitz, and discontinuous optimization problems [24, 27, 37, 38]. To solve

*Corresponding author.
E-mail addresses: jiangmin@zjut.edu.cn (M. Jiang), shenrui126@126.com (R. Shen), mengzhiqing@zjut.edu.cn (Z. Meng), mecdang@cityu.edu.hk (C. Dang).
Received January 16, 2022; Accepted September 23, 2022.

these problems, theoretical tools of nonsmooth and nonconvex functions are needed, such as the subdifferentiable, general convex, smoothing, and so on [3, 9, 10, 14, 31].

In this paper, we define a new nonconvex function (Definition 2.2 below), called the (weak or strong uniform) CN function, where the CN function is a nonconvex nonsmooth function form that can be transformed into a convex smooth function with convex equality constraints. The CN function somewhat relates to the upper -UC^k function [5, 11, 17, 28, 31, 34] and the factorable nonconvex function [6, 15, 21, 22, 25, 26, 36, 32].

The lower(upper)-C^k function was suggested by Rockafellar [31]. The class of lower-C^1 functions was first introduced by Spingarn [34]. Spingarn proved that these functions are (Mifflin) semi-smooth and Clarke regular, and are characterized by a generalized monotonicity property of their subgradients, called the submonotonicity. The definition of the lower(upper)-C^k function was given as follows.

Definition 1.1. [11] Let U be an open subset of R^n and $k \in N$. Function $f : U \to R$ is called lower-C^k (for short, LC^k) if, for every $\boldsymbol{x}_0 \in U$, there exist $\delta > 0$, compact topological space S, and a jointly continuous function $F : B(\boldsymbol{x}_0, \delta) \times S \to R$ satisfying $f(\boldsymbol{x}) = \max_{\boldsymbol{s} \in S} F(\boldsymbol{x}, \boldsymbol{s})$ for all $\boldsymbol{x} \in B(\boldsymbol{x}_0, \delta)$ such that all the derivatives of F up to order k with respect to $\boldsymbol{x}$ exist and are jointly continuous. If $-f$ is lower-C^k, then f is said to be upper-C^k.

The lower(upper)-C^k function is nonconvex or nondifferentiable, but it is locally Lipschitz approximately convex function [11]. The Moreau envelopes $erf(\boldsymbol{x}) := \inf_{\boldsymbol{w}} \{f(\boldsymbol{w}) + \frac{r}{2}|\boldsymbol{w} - \boldsymbol{x}|^2\}$ is lower-C^2 [5, 17, 28] such that the subdifferential of the lower(upper)-C^k functions can solve nonconvex optimization by prox-regularity and the proximal mapping(operator) [18]. Chieu et al. proved second-order necessary and sufficient conditions for lower-C^2 functions to be convex and strongly convex in [9]. Some methods for non-smooth non-convex optimization programs with lower(upper)-C^k functions have been studied in [12, 19, 20, 30]. Dao [12] developed a nonconvex bundle method based on the downshift mechanism and a proximity control management technique to solve nonconvex nonsmooth constrained optimization problems. He proved the global convergence in the sense of subsequences for both classes of lower-C^1 and upper-C^1; see [12] for more details. In [19, 20], Hare et al. studied two proximal bundle methods for nonsmooth nonconvex optimization by proximal mapping on lower-C^2 functions. In [30], Noll defined a first-order model of f as an extend case of lower-C^k function and presented a bundle method as follows.

Definition 1.2. A function $\phi : R^n \times R^n \to R$ is called a first-order model of f on $\omega \subset R^n$ if $\phi(\cdot, \boldsymbol{x})$ is convex for every fixed $\boldsymbol{x} \in \omega$, and if the following axioms are satisfied:

(M1) $\phi(\boldsymbol{x}, \boldsymbol{x}) = f(\boldsymbol{x})$ and $\partial_1 \phi(\boldsymbol{x}, \boldsymbol{x}) \subset \partial f(\boldsymbol{x})$.

(M2) For every sequence $\boldsymbol{y}_j \to \boldsymbol{x}$, there exists $\varepsilon_j \to 0^+$ such that $f(\boldsymbol{y}_j) \leq \phi(\boldsymbol{y}_j, \boldsymbol{x}) + \varepsilon_j \|\boldsymbol{y}_j - \boldsymbol{x}\|$ for all $j \in N$.

(M3) For sequences $\boldsymbol{y}_j \to \boldsymbol{y} \in R^n$ and $\boldsymbol{x}_j \to \boldsymbol{x}$ in ω, one has $f(\boldsymbol{y}_j) \leq \limsup_{j \to \infty} \leq \phi(\boldsymbol{y}, \boldsymbol{x})$ for all $j \in N$.

Clearly, if f is a first-order model, then f is not necessarily lower-C^k, and the reverse is not necessarily true.

On the other hand, the branch-and-bound method in conjunction with underestimating convex problems has been proved an effective method to solve global nonconvex optimization problems [1, 4, 35]. Almost all the methods used to solve nonconvex optimization are to construct many convex relaxation subproblems with convex envelopes and convex underestimating; see, e.g., [4, 33, 32, 36]. Based on this idea, the factorable programming technique, one of the most popular approaches for constructing convex relaxations of nonconvex optimization problems, including the problems with convex-transformable functions, was given in [26]. Due to its simplicity, factorable programming technique is included in most global optimization packages, such as BARON, ANTIGONE, and so on [29]. But, Nohra and Sahinidis pointed out in [29] that a main drawback of factorable programming technique is that it often results in large relaxation gaps. In 1976, McCormick [26] first defined the factorable nonconvex function, however the factorable nonconvex function is not necessarily lower-C^1, such as $f(x) = |x|^{0.1} + |x+1|^{0.2}$ on $x \in R$, because $f(x) = |x|^{0.1} + |x+1|^{0.2}$ is not locally Lipschitz [7]. In fact, the factorable nonconvex functions in [22, 25, 26, 36] may be special CN functions (see Definition 2.2). In recent years, research on nonconvex factorable programming further demonstrates its effectiveness in solving the global optimization; see, e.g., [6, 15, 21, 32] and the references therein.

There are many CN functions that are not upper-C^k functions or factorable functions, such as $|x|_0$ because upper-C^k functions are continuous; see [16]. So, a CN function is not necessarily an upper-C^k function or a factorable nonconvex function. There are three differences between factorable functions and CN functions.

(1) They are different in functional decomposition representation. Each function $X^i(x)$ *w.r.t* a single variable in any form of a factorable function is not necessarily a convex or concave function [26]. However, each function $g_i(\boldsymbol{x}, \boldsymbol{y})$ in any CN form of a CN function is convex (see Definition 2.2 below).

(2) To estimate the factorable function, it is necessary to underestimate/overestimate the convex/concave functions of each $X^i(x)$, while CN function does not require the estimation of its convex envelope (underestimating) functions.

(3) The method of solving optimization problems with CN functions differs from the method of solving nonconvex factorable programming. Factorial programming solves its relaxation problem to obtain an approximate global optimal solution. To solve the approximate global optimal solutions to the optimization problems with CN functiosn, their equivalent optimization are needed.

In order to solve (CNO), Jiang et al. [23] discussed optimal conditions, Lagrangian dual, and an algorithm for the unconstrained CN optimization problems. In this paper, a weak uniform CN function is defined, and the weak uniform and decomposable weak uniform of the CN function, optimization conditions, and decomposable algorithms for (CNO) with the weak uniform CN function are studied. The main contributions of this paper are as follows: (1) a weak uniform CN function is proposed, (2) the sufficient conditions of the optimal solution to the optimization problems with such CN function are proved, and (3) a decomposable algorithm for the optimization problem with a decomposable CN function is proposed. This paper provides a new method to solve the difficulties in solving nonconvex or nonsmooth optimization problems.

The remainder of the paper is organized as follows. In Section 2, the CN function and the weak (strongy) uniform (decomposable, exact) CN function are defined, respectively. Some examples are given. The operational properties of the CN functions are demonstrated. In section 3, optimization conditions of the global optimal solution to the unconstrained optimization with the weak uniform CN function are proved. In Section 4, for the decomposable CN function, a decomposable algorithm is proposed by its augmented Lagrangian penalty function, and its convergence is proved. Section 5 the last section, end this paper.

2. The Weak Uniform CN Function

In this section, a (weak, strong uniform, exact) CN function is defined. Some examples are given to demonstrate that some nonconvex or discontinuous functions are differentiable (weak uniform) CN ones.

Definition 2.1. Let function $g : R^n \times R^m \to R$ be differentiable. For all $\boldsymbol{d} \in R^n \times R^m$ and all $(\boldsymbol{x},\boldsymbol{y}) \in R^n \times R^m$, if there is a positive semidefinite matrix $B(\boldsymbol{x},\boldsymbol{y})$ such that

$$g((\boldsymbol{x},\boldsymbol{y})+\boldsymbol{d}) - g(\boldsymbol{x},\boldsymbol{y}) \geq \nabla g(\boldsymbol{x},\boldsymbol{y})^\top \boldsymbol{d} + \frac{1}{2}\boldsymbol{d}^\top B(\boldsymbol{x},\boldsymbol{y})\boldsymbol{d}, \tag{2.1}$$

then g is called a weak uniform convex function which has two cases as follows.

(i) If there is a positive definite matrix $B(\boldsymbol{x},\boldsymbol{y})$ such that (2.1) holds, then g is called a strong uniform convex function.

(ii) If there is an $\bar{\rho} > 0$ such that

$$g((\boldsymbol{x},\boldsymbol{y})+\boldsymbol{d}) - g(\boldsymbol{x},\boldsymbol{y}) \geq \nabla g(\boldsymbol{x},\boldsymbol{y})^\top \boldsymbol{d} + \frac{\bar{\rho}}{2}\|\boldsymbol{d}\|^2, \tag{2.2}$$

then g is called a uniform convex function.

It is clear that a uniform convex function g is not only a weak uniform convex function but also a strong uniform convex function. A strong uniform convex function is a weak uniform convex function. And a weak uniform convex function is a convex function, which is demonstrated in the following examples.

Example 2.1. $g(x,y) = (x+y-1)^2$ for $(x,y) \in R \times R$ is a weak uniform convex function, but $g(x,y)$ is not a strong uniform convex function.

Example 2.2. $g(x,y) = x^4 + y^4$ for $(x,y) \in R \times R$ is a strong uniform convex function for $(x,y) \neq 0$, but $g(x,y)$ is not a uniform convex function.

Example 2.3. $g(\boldsymbol{x},\boldsymbol{y}) = (\boldsymbol{x},\boldsymbol{y})^\top A(\boldsymbol{x},\boldsymbol{y}) + \boldsymbol{c}^\top(\boldsymbol{x},\boldsymbol{y})$ is a weak uniform convex function, where A is a positive semidefinite matrix, $(\boldsymbol{x},\boldsymbol{y}) \in R^n \times R^m$.

We have the following conclusion.

Proposition 2.1. *Let function $g : R^n \times R^m \to R$ be twice continuously differentiable and matrix $B(\boldsymbol{x},\boldsymbol{y})$ be given for $(\boldsymbol{x},\boldsymbol{y}) \in R^n \times R^m$. Then $g(\boldsymbol{x},\boldsymbol{y})$ is a weak (strong) uniform convex function if and only if $\boldsymbol{d}^\top \nabla^2 g(\boldsymbol{x},\boldsymbol{y})\boldsymbol{d} \geq \boldsymbol{d}^\top B(\boldsymbol{x},\boldsymbol{y})\boldsymbol{d} \geq (>) 0$, $\forall(\boldsymbol{x},\boldsymbol{y})$, $\forall \boldsymbol{d} \in R^n \times R^m$.*

Based on the definition of weak uniform convex functions, we define the weak uniform CN function, strong uniform CN function, and uniform CN function.

Definition 2.2. Let $S = S_1 \times S_2 \subset R^n \times R^m$ be a convex set. Let functions $g_i : R^n \times R^m \to R$ ($i = 1,2,\cdots,r$ and $r \geq 1$) be convex on S, and let $g : R^n \times R^m \to R$ be a convex function on S. Let function $f : R^n \to R$ be nonconvex. All functions $g_1(\boldsymbol{x},\boldsymbol{y}), g_2(\boldsymbol{x},\boldsymbol{y}), \cdots, g_r(\boldsymbol{x},\boldsymbol{y})$ are recorded as

$$\boldsymbol{g}(\boldsymbol{x},\boldsymbol{y}) = (g_1(\boldsymbol{x},\boldsymbol{y}), g_2(\boldsymbol{x},\boldsymbol{y}), \cdots, g_r(\boldsymbol{x},\boldsymbol{y}))^\top. \tag{2.3}$$

Let

$$X(\boldsymbol{g}) = \{(\boldsymbol{x},\boldsymbol{y}) \in S \mid g_i(\boldsymbol{x},\boldsymbol{y}) = 0,\ i = 1,2,\cdots,r\}. \tag{2.4}$$

If, for each $\boldsymbol{x} \in S$ and $f(\boldsymbol{x})$, there is a $\boldsymbol{y} \in R^m$ such that $(\boldsymbol{x},\boldsymbol{y}) \in X(\boldsymbol{g})$ and

$$f(\boldsymbol{x}) = g(\boldsymbol{x},\boldsymbol{y}) = \min_{(\boldsymbol{x},\boldsymbol{y}') \in X(\boldsymbol{g})} g(\boldsymbol{x},\boldsymbol{y}'), \tag{2.5}$$

then f is called a convertible nonconvex (CN) function on S (when $S = R^n \times R^m$, the term "on S" is omitted), i.e., if there are $g(\boldsymbol{x},\boldsymbol{y})$ and $g_1(\boldsymbol{x},\boldsymbol{y}), g_2(\boldsymbol{x},\boldsymbol{y}), \cdots, g_r(\boldsymbol{x},\boldsymbol{y})$ such that (2.5) holds, then f is a CN function on S, and $[g : g_1, g_2, \cdots, g_r]$ is called a convertible nonconvex(CN) form of f on S, briefing as $f = [g : g_1, g_2, \cdots, g_r]$. For f, the number of its CN form is more than one. In particular,

(i) if g is a weak uniform convex function on S, then f is called a weak uniform CN function on S;

(ii) if g is a strong uniform convex function on S, then f is called a strong uniform CN function on S;

(iii) if g is a uniform convex function on S, then f is called a uniform CN function on S.

Furthermore, if for each $\boldsymbol{x} \in S$ and $f(\boldsymbol{x})$ there is a $\boldsymbol{y} \in R^m$ such that $(\boldsymbol{x},\boldsymbol{y}) \in X(\boldsymbol{g})$ and

$$f(\boldsymbol{x}) = g(\boldsymbol{x},\boldsymbol{y}) = g(\boldsymbol{x},\boldsymbol{y}'), \quad \forall (\boldsymbol{x},\boldsymbol{y}') \in X(\boldsymbol{g}),$$

then f is called an exact convertible nonconvex(CN) function on S. $[g : g_1, g_2, \cdots, g_r]$ is called an exact convertible nonconvex (CN) form of f on S. Then, f is called a (weak or strong) uniform exact CN function on S if condition (i), (ii), or (iii) holds accordingly.

When $S = R^n \times R^m$, the term "on S" above is all omitted. It is clear that a uniform CN function is a CN function, a weak uniform CN function, and a strong uniform CN function. A weak uniform convertible convex function is a CN function.

Remark 2.1. It is clear that the definition of the CN function differs form that of the upper-C^k function [11] and a first-order model [30]. For example, 0-norm $\|\boldsymbol{x}\|_0$ is not an upper-C^k function [11, 16] or a first-order model because it is not continuous, but $\|\boldsymbol{x}\|_0$ is a CN function in Example 2.7. Hence, the CN function contains a wider range of functions than upper-C^k functions and the first-order model.

Definition 2.2 means that a nonconvex function $f(\boldsymbol{x})$ may be converted into a (weak uniform) convertible convex function. By Definition 2.2, a set is defined by

$$X(f) = \{(\boldsymbol{x},\boldsymbol{y}) | f(\boldsymbol{x}) = g(\boldsymbol{x},\boldsymbol{y}), \forall (\boldsymbol{x},\boldsymbol{y}) \in X(\boldsymbol{g})\}. \tag{2.6}$$

For a fixed $(\boldsymbol{x},\boldsymbol{y})$, two sets are defined by

$$Y_g(\boldsymbol{x}) = \{\boldsymbol{y} \in R^m | (\boldsymbol{x},\boldsymbol{y}) \in X(\boldsymbol{g})\}, X_g(\boldsymbol{y}) = \{\boldsymbol{x} \in R^m | (\boldsymbol{x},\boldsymbol{y}) \in X(\boldsymbol{g})\}.$$

The following conclusion is clear.

Proposition 2.2. *Let f be a CN function on S. Then, (1) f is an exact CN function on S if and only if $X(f) = X(\boldsymbol{g})$; (2) $X(f)$ and $X(\boldsymbol{g})$ are closed sets for $S = R^n$; (3) $f(\boldsymbol{x}) = \min_{\boldsymbol{y} \in Y_g(\boldsymbol{x})} g(\boldsymbol{x},\boldsymbol{y})$; and (4) $\min_{\boldsymbol{x} \in X_g(\boldsymbol{y})} f(\boldsymbol{x}) \le \min_{\boldsymbol{x} \in X_g(\boldsymbol{y})} g(\boldsymbol{x},\boldsymbol{y})$.*

Proposition 2.2 demonstrates that if f is a CN function on S, $f(\boldsymbol{x}) = g(\boldsymbol{x},\boldsymbol{y}) \leq g(\boldsymbol{x},\boldsymbol{y}')$ for all $(\boldsymbol{x},\boldsymbol{y}) \in X(f)$ and $(\boldsymbol{x},\boldsymbol{y}') \in X(\boldsymbol{g})\backslash X(f)$. $(\boldsymbol{x},\boldsymbol{y}) \in X(f)$ is called a CN point of f. Next, an example is given to demonstrate that the number of weak uniform (exact) CN forms could be more than one.

Example 2.4. Non-convex function $f(x_1,x_2) = 2x_1x_2$ is a weak uniform exact CN function. One of its weak uniform exact CN forms is $f = [(x_1+x_2)^2 - y_1 - y_2 : x_1^2 - y_1, x_2^2 - y_2]$. A second weak uniform exact CN form of $f(x_1,x_2) = 2x_1x_2$ is $f = [(x_1+x_2)^2 - y_1 : x_1^2 + x_2^2 - y_1]$. Hence, it is understood that there are more than one weak uniform exact CN form.

Next, some operational properties of the weak uniform (exact) CN function are easily proved as follows.

Proposition 2.3. *If $f : R^n \to R$ is an exact CN function on S, then $-f$ is an exact CN function on S.*

Proposition 2.4. *If $f_1, f_2 : R^n \to R$ are (weak uniform or strong uniform) (exact) CN functions on S, then $\alpha_1 f_1 + \alpha_2 f_2$ is a (weak uniform or strong uniform) (exact) CN function on S for any $\alpha_1, \alpha_2 > 0$. Especially, if $f_1, f_2 : R^n \to R$ are exact CN functions on S, then $\alpha_1 f_1 + \alpha_2 f_2$ is an exact CN function on S for any $\alpha_1, \alpha_2 \in R$.*

Proposition 2.5. *If $f_1, f_2 : R^n \to R$ are (weak uniform or strong uniform) exact CN functions on S, then $f_1 f_2$ is a weak uniform exact CN function on S.*

Proposition 2.6. *If $f_1, f_2 : R^n \to R$ are (weak uniform or strong uniform) exact CN functions on S, then $\frac{f_1}{f_2}$ is a weak uniform exact CN function on S.*

Proposition 2.7. *If $f : R^n \to R$ is a (weak uniform or strong uniform) (exact) CN function on S and $\phi : R \to R$ is a monotone increasing convex function, then $\phi(f)$ is a (weak uniform or strong uniform) (exact) CN function on S.*

Example 2.5. Let a weak uniform DC function $f(\boldsymbol{x},\boldsymbol{y}) = d(\boldsymbol{x},\boldsymbol{y}) - c(\boldsymbol{x},\boldsymbol{y})$ for $(\boldsymbol{x},\boldsymbol{y}) \in R^n \times R^m$, where $d(\boldsymbol{x},\boldsymbol{y})$ and $c(\boldsymbol{x},\boldsymbol{y})$ are weak uniform convex functions on $(\boldsymbol{x},\boldsymbol{y}) \in R^n \times R^m$. Let $g(\boldsymbol{x},\boldsymbol{y},z) = d(\boldsymbol{x},\boldsymbol{y}) - z$ and $g_1(\boldsymbol{x},\boldsymbol{y},z) = c(\boldsymbol{x},\boldsymbol{y}) - z$. Thus $f(x,y)$ is a weak uniform CN function.

By Proposition 2.3-2.7, some polynomial functions are CN. For example, multi-convex function $f(\boldsymbol{x}) = x_1x_2\cdots x_n$ is a weak uniform CN by Proposition 2.4. Thus weak uniform CN functions cover a wide range of non-convex functions. To illustrate it, some examples are given as follows.

Example 2.6. ([7, Example 2.1]) A CN form of nonsmooth function $f(x_1,x_2) = (x_1+x_2-1)^2 + \lambda(|x_1|^{\frac{1}{2}} + |x_2|^{\frac{1}{2}})$ is defined as

$$[(x_1+x_2-1)^2 + \lambda(y_1+y_4) : y_1^4 - y_3, x_1^2 - y_3, y_2^2 - y_1, y_4^4 - y_6, x_2^2 - y_6, y_5^2 - y_4],$$

where $\lambda > 0$. So, $f(x)$ is a weak uniform exact CN function.

Example 2.7. A CN form of the nonconvex and discontinuous function $f(x_1,x_2) = (x_1+x_2-1)^2 + \lambda\|(x_1,x_2)\|_0$ is defined as

$$\begin{aligned} f &= [(x_1+x_2-1)^2 + \lambda(y_1^2+y_3^2) : (x_1+y_1-1)^2 - y_2, x_1^2 + (y_1-1)^2 - y_2, \\ &\quad y_1^2 - y_1, (x_2+y_3-1)^2 - y_4, x_2^2 + (y_3-1)^2 - y_4, y_3^2 - y_3], \end{aligned}$$

where $\lambda > 0$. So, $f(x)$ is a weak uniform CN function, but not exact.

The above examples demonstrate that some nonsmooth, nonconvex, or discontinuous functions are CN functions as shown by their twice differentiable and weak uniform CN forms. Many nonconvex functions in the fields of machine learning, image processing, and signal processing are combinations of simple CN functions by Propositions 2.3-2.7. For example, let $\alpha_i > 0 (i = 1, 2, \cdots, m), \alpha > 0, \beta > 0$, $k_0 > 0$, and $k_1 > 0$, then the nonconvex functions:

$$\prod_{i=1}^{m} x_i^{\alpha_i},\ f(\boldsymbol{x})^{\alpha} g(\boldsymbol{x})^{\beta},\ \frac{f(\boldsymbol{x})^{\alpha} exp(g_0(\boldsymbol{x}))}{k_0 + k_1 \sum_{i=1}^{m} exp(g_i(\boldsymbol{x}))},$$

where $f(\boldsymbol{x}), g(\boldsymbol{x}), g_i(\boldsymbol{x}) (i = 0, 1, \cdots, m)$ are convex on $\boldsymbol{x}$, are CN functions; see [29]. The compiled functions demonstrate themselves in a wide variety of scientific and engineering applications. Their exact CN forms are easily obtained by Propositions 2.3-2.7.

3. Optimization Conditions of (CNO)

In this section, it is assumed that f is a weak uniform CN function or a CN function with $f = [g : g_1, g_2, \cdots, g_r]$, where $g, g_1, g_2, \cdots, g_r$ are twice differentiable, but $f(\boldsymbol{x})$ is not necessarily differentiable on $\boldsymbol{x} \in R^n$.

With the (weak uniform) CN form of f, one considers the following constrained optimization problem:

$$\text{(CNP)} \qquad \min_{(\boldsymbol{x},\boldsymbol{y}) \in R^n \times R^m} \ g(\boldsymbol{x},\boldsymbol{y}) \qquad \text{s.t.} \quad g_i(\boldsymbol{x},\boldsymbol{y}) = 0, i = 1, 2, \cdots, r.$$

It is clear $\min_{\boldsymbol{x} \in R^n} \ f(\boldsymbol{x}) = \min_{(\boldsymbol{x},\boldsymbol{y}) \in X(f)} \ g(\boldsymbol{x},\boldsymbol{y}) = \min_{(\boldsymbol{x},\boldsymbol{y}) \in X(\boldsymbol{g})} \ g(\boldsymbol{x},\boldsymbol{y})$.

Let a directional sets at a fixed $(\boldsymbol{x},\boldsymbol{y})$ be defined by

$$T(\boldsymbol{x},\boldsymbol{y}) = \{\boldsymbol{d} \in R^n \times R^m \quad | \quad \nabla g_i(\boldsymbol{x},\boldsymbol{y})^\top \boldsymbol{d} \le 0, i = 1, 2, \cdots, r\}$$

Let us prove the sufficient condition of an optimal solution to (CNO) by solving (CNP).

Theorem 3.1. *Suppose that $(\boldsymbol{x}^*,\boldsymbol{y}^*) \in X(f)$ and f is a weak uniform CN function. Consider the problem*

$$\textit{(WCNP)}(\boldsymbol{x}^*,\boldsymbol{y}^*) \qquad \min \ \nabla g(\boldsymbol{x}^*,\boldsymbol{y}^*)^\top \boldsymbol{d} + \frac{1}{2} \boldsymbol{d}^\top B(\boldsymbol{x}^*,\boldsymbol{y}^*) \boldsymbol{d} \qquad \textit{s.t.} \quad \boldsymbol{d} \in T(\boldsymbol{x}^*,\boldsymbol{y}^*).$$

If $\boldsymbol{d}^*$ is an optimal solution to (WCNP)$(\boldsymbol{x}^*,\boldsymbol{y}^*)$ such that $\nabla g(\boldsymbol{x}^*,\boldsymbol{y}^*)^\top \boldsymbol{d}^* + \frac{1}{2} \boldsymbol{d}^{*T} B(\boldsymbol{x}^*,\boldsymbol{y}^*) \boldsymbol{d}^* \ge 0$, then $(\boldsymbol{x}^*,\boldsymbol{y}^*)$ is an optimal solution to (CNP) and $\boldsymbol{x}^*$ is an optimal solution to (CNO). Furthermore, if there is a $\boldsymbol{d}'$ such that $\nabla g_i(\boldsymbol{x}^*,\boldsymbol{y}^*)^\top \boldsymbol{d}' < 0, i = 1, 2, \cdots, r$, then there are $\alpha_i^* \ge 0, i = 1, 2, \cdots, r$ such that

$$B(\boldsymbol{x}^*,\boldsymbol{y}^*) \boldsymbol{d}^* + \nabla g(\boldsymbol{x}^*,\boldsymbol{y}^*) + \sum_{i=1}^{r} \alpha_i^* \nabla g_i(\boldsymbol{x}^*,\boldsymbol{y}^*) = 0, \tag{3.1}$$

$$\alpha_i^* \nabla g_i(\boldsymbol{x}^*,\boldsymbol{y}^*)^\top \boldsymbol{d}^* = 0, i = 1, 2, \cdots, r. \tag{3.2}$$

Conversely, if there are $\boldsymbol{d}^* \in T(\boldsymbol{x}^*,\boldsymbol{y}^*)$ and $\alpha_i^* \ge 0, i = 1, 2, \cdots, r$ such that (3.1) and (3.2) hold, then $\boldsymbol{d}^*$ is an optimal solution to (WCNP)$(\boldsymbol{x}^*,\boldsymbol{y}^*)$.

Proof. For any $(\boldsymbol{x},\boldsymbol{y}) \in X(\boldsymbol{g})$, let $\boldsymbol{d} = [(\boldsymbol{x},\boldsymbol{y}) - (\boldsymbol{x}^*,\boldsymbol{y}^*)]$. Then

$$0 = g_i(\boldsymbol{x},\boldsymbol{y}) - g_i(\boldsymbol{x}^*,\boldsymbol{y}^*) \quad \geq \quad \nabla g_i(\boldsymbol{x}^*,\boldsymbol{y}^*)^\top [(\boldsymbol{x},\boldsymbol{y}) - (\boldsymbol{x}^*,\boldsymbol{y}^*)], \qquad i = 1,2,\cdots,r,$$

and $(\boldsymbol{x},\boldsymbol{y}) - (\boldsymbol{x}^*,\boldsymbol{y}^*)$ is a feasible solution to (WCNP)$(\boldsymbol{x}^*,\boldsymbol{y}^*)$. It follows that

$$\begin{aligned} g(\boldsymbol{x},\boldsymbol{y}) - g(\boldsymbol{x}^*,\boldsymbol{y}^*) &\geq \nabla g(\boldsymbol{x}^*,\boldsymbol{y}^*)^\top \boldsymbol{d} + \frac{1}{2}\boldsymbol{d}^\top B(\boldsymbol{x}^*,\boldsymbol{y}^*)\boldsymbol{d}, \\ &\geq \nabla g(\boldsymbol{x}^*,\boldsymbol{y}^*)^\top \boldsymbol{d}^* + \frac{1}{2}\boldsymbol{d}^{*\top} B(\boldsymbol{x}^*,\boldsymbol{y}^*)\boldsymbol{d}^* \geq 0. \end{aligned}$$

Hence, $(\boldsymbol{x}^*,\boldsymbol{y}^*)$ is an optimal solution to (CNP), and $\boldsymbol{x}^*$ is an optimal solution to (CNO). Since (WCNP)$(\boldsymbol{x}^*,\boldsymbol{y}^*)$ is a convex programm, by the KKT condition, conclusions (3.1) and (3.2) are true. Conversely, the conclusion is clear. □

By Theorem 3.1, we have the following three corollaries.

Corollary 3.1. *Suppose that $(\boldsymbol{x}^*,\boldsymbol{y}^*) \in X(f)$ and f is a uniform CN function. Consider the problem*

$$\text{(UCNP)}(\boldsymbol{x}^*,\boldsymbol{y}^*) \qquad \begin{aligned} \min \quad & \nabla g(\boldsymbol{x}^*,\boldsymbol{y}^*)^\top \boldsymbol{d} + \frac{\bar{\rho}}{2}\|\boldsymbol{d}\|^2 \\ s.t. \quad & \boldsymbol{d} \in T(\boldsymbol{x}^*,\boldsymbol{y}^*). \end{aligned}$$

If $\boldsymbol{d}^*$ is an optimal solution to (UCNP)$(\boldsymbol{x}^*,\boldsymbol{y}^*)$ such that $\nabla g(\boldsymbol{x}^*,\boldsymbol{y}^*)^\top \boldsymbol{d}^* + \frac{\bar{\rho}}{2}\|\boldsymbol{d}^*\|^2 \geq 0$, then $(\boldsymbol{x}^*,\boldsymbol{y}^*)$ is an optimal solution to (CNP) and $\boldsymbol{x}^*$ is an optimal solution to (CNO). Furthermore, if there is a $\boldsymbol{d}'$ such that $\nabla g_i(\boldsymbol{x}^*,\boldsymbol{y}^*)^\top \boldsymbol{d}' < 0, i = 1,2,\cdots,r$, then there are $\alpha_i^* \geq 0, i = 1,2,\cdots,r$ such that

$$\bar{\rho}\boldsymbol{d}^* + \nabla g(\boldsymbol{x}^*,\boldsymbol{y}^*) + \sum_{i=1}^{r} \alpha_i^* \nabla g_i(\boldsymbol{x}^*,\boldsymbol{y}^*) = 0, \tag{3.3}$$

$$\alpha_i^* \nabla g_i(\boldsymbol{x}^*,\boldsymbol{y}^*)^\top \boldsymbol{d}^* = 0, i = 1,2,\cdots,r. \tag{3.4}$$

Conversely, if there are $\boldsymbol{d}^* \in T(\boldsymbol{x}^*,\boldsymbol{y}^*)$ and $\alpha_i^* \geq 0, i = 1,2,\cdots,r$ such that (3.3) and (3.4) hold, then $\boldsymbol{d}^*$ is an optimal solution to (UCNP)$(\boldsymbol{x}^*,\boldsymbol{y}^*)$.

Corollary 3.2. *Suppose that $(\boldsymbol{x}^*,\boldsymbol{y}^*) \in X(f)$ and f is a CN function. Consider the problem*

$$\text{(LCNP)}(\boldsymbol{x}^*,\boldsymbol{y}^*) \qquad \begin{aligned} \min \quad & \nabla g(\boldsymbol{x}^*,\boldsymbol{y}^*)^\top \boldsymbol{d} \\ s.t. \quad & \boldsymbol{d} \in T(\boldsymbol{x}^*,\boldsymbol{y}^*). \end{aligned}$$

If $\boldsymbol{d}^*$ is an optimal solution to (LCNP)$(\boldsymbol{x}^*,\boldsymbol{y}^*)$ such that $\nabla g(\boldsymbol{x}^*,\boldsymbol{y}^*)^\top \boldsymbol{d}^* \geq 0$, then $(\boldsymbol{x}^*,\boldsymbol{y}^*)$ is an optimal solution to (CNP) and $\boldsymbol{x}^*$ is an optimal solution to (CNO). Furthermore, if there is a $\boldsymbol{d}'$ such that $\nabla g_i(\boldsymbol{x}^*,\boldsymbol{y}^*)^\top \boldsymbol{d}' < 0, i = 1,2,\cdots,r$, then there are $\alpha_i^* \geq 0, i = 1,2,\cdots,r$ such that

$$\nabla g(\boldsymbol{x}^*,\boldsymbol{y}^*) + \sum_{i=1}^{r} \alpha_i^* \nabla g_i(\boldsymbol{x}^*,\boldsymbol{y}^*) = 0. \tag{3.5}$$

$$\alpha_i^* \nabla g_i(\boldsymbol{x}^*,\boldsymbol{y}^*)^\top \boldsymbol{d}^* = 0, i = 1,2,\cdots,r. \tag{3.6}$$

Conversely, if there are $\boldsymbol{d}^* \in T(\boldsymbol{x}^*,\boldsymbol{y}^*)$ and $\alpha_i^* \geq 0, i = 1,2,\cdots,r$ such that (3.5) and (3.6) hold, then $(\boldsymbol{x}^*,\boldsymbol{y}^*)$ is an optimal solution to (CNP) and $\boldsymbol{x}^*$ is an optimal solution to (CNO).

Corollary 3.3. *Suppose that $(\boldsymbol{x}^*,\boldsymbol{y}^*) \in X(f)$ and f is a CN function. If $\nabla g(\boldsymbol{x}^*,\boldsymbol{y}^*) = 0$, then $(\boldsymbol{x}^*,\boldsymbol{y}^*)$ is an optimal solution to (CNP) and $\boldsymbol{x}^*$ is an optimal solution to (CNO).*

Define a set

$$T_1(\boldsymbol{x}^*,\boldsymbol{y}^*) = \{(\boldsymbol{x},\boldsymbol{y}) \mid \nabla g_i(\boldsymbol{x}^*,\boldsymbol{y}^*)^\top[(\boldsymbol{x},\boldsymbol{y})-(\boldsymbol{x}^*,\boldsymbol{y}^*)] \le 0, i=1,2,\cdots,r.\}$$

We have $X(\boldsymbol{g}) \subset T_1(\boldsymbol{x}^*,\boldsymbol{y}^*)$ and $T_1(\boldsymbol{x}^*,\boldsymbol{y}^*)-(\boldsymbol{x}^*,\boldsymbol{y}^*)=T(\boldsymbol{x}^*,\boldsymbol{y}^*)$. By [13], if $G_1(\boldsymbol{d})$ of (WCNP)$(\boldsymbol{x}^*,\boldsymbol{y}^*)$ has a lower bound, then there is a global optimal solution to (WCNP)$(\boldsymbol{x}^*,\boldsymbol{y}^*)$. Then we have the following conclusion.

Theorem 3.2. *Suppose that $(\boldsymbol{x}^*,\boldsymbol{y}^*) \in X(f)$ and f is a weak uniform CN function. Consider the problem*

$$\begin{array}{llll} \text{(WCNP1)}(\boldsymbol{x}^*,\boldsymbol{y}^*) & \min & G_1(\boldsymbol{x},\boldsymbol{y}) = \nabla g(\boldsymbol{x}^*,\boldsymbol{y}^*)^\top[(\boldsymbol{x},\boldsymbol{y})-(\boldsymbol{x}^*,\boldsymbol{y}^*)] \\ & & +\frac{1}{2}[(\boldsymbol{x},\boldsymbol{y})-(\boldsymbol{x}^*,\boldsymbol{y}^*)]^\top B(\boldsymbol{x}^*,\boldsymbol{y}^*)[(\boldsymbol{x},\boldsymbol{y})-(\boldsymbol{x}^*,\boldsymbol{y}^*)] \\ & s.t. & (\boldsymbol{x},\boldsymbol{y}) \in T_1(\boldsymbol{x}^*,\boldsymbol{y}^*). \end{array}$$

If the objective value $G_1(\boldsymbol{x},\boldsymbol{y})$ of (WCNP1)$(\boldsymbol{x}^*,\boldsymbol{y}^*)$ has a lower bound, then there is an optimal solution $(\bar{\boldsymbol{x}},\bar{\boldsymbol{y}})$ to (WCNP1)$(\boldsymbol{x}^*,\boldsymbol{y}^*)$. Furthermore, if $G_1(\bar{\boldsymbol{x}},\bar{\boldsymbol{y}}) \ge 0$, then $(\boldsymbol{x}^*,\boldsymbol{y}^*)$ is an optimal solution to (CNP) and $\boldsymbol{x}^*$ is an optimal solution to (CNO), or if $G_1(\bar{\boldsymbol{x}},\bar{\boldsymbol{y}}) < 0$, then $G_1(\bar{\boldsymbol{x}},\bar{\boldsymbol{y}}) = -\frac{1}{2}\bar{\boldsymbol{d}}^\top B(\boldsymbol{x}^*,\boldsymbol{y}^*)\bar{\boldsymbol{d}}$, where $\bar{\boldsymbol{d}} = (\bar{\boldsymbol{x}},\bar{\boldsymbol{y}})-(\boldsymbol{x}^*,\boldsymbol{y}^*)$. If there is $(\boldsymbol{x}',\boldsymbol{y}')$ such that $\nabla g_i(\boldsymbol{x}^*,\boldsymbol{y}^*)^\top[(\boldsymbol{x},\boldsymbol{y})-(\boldsymbol{x}^*,\boldsymbol{y}^*)] < 0, i=1,2,\cdots,r$ holds, then there are $\alpha_i^* \ge 0, i=1,2,\cdots,r$ such that

$$\bar{\boldsymbol{d}}^\top B(\boldsymbol{x}^*,\boldsymbol{y}^*) + \nabla g(\boldsymbol{x}^*,\boldsymbol{y}^*) + \sum_{i=1}^{r} \alpha_i^* \nabla g_i(\boldsymbol{x}^*,\boldsymbol{y}^*) = 0.$$

These conclusions of Theorem 3.1 and Theorem 3.2 are true when f is a strong uniform CN function. Theorem 3.2 demonstrates that $(\boldsymbol{x}^*,\boldsymbol{y}^*)$ is an optimal solution to (CNP) if $B(\boldsymbol{x}^*,\boldsymbol{y}^*)$ is positive definite or $\boldsymbol{d}^{*T}B(\boldsymbol{x}^*,\boldsymbol{y}^*)\boldsymbol{d}^* > 0$. If $(\boldsymbol{x}^*,\boldsymbol{y}^*)$ is not an optimal solution to (CNP), then $B(\boldsymbol{x}^*,\boldsymbol{y}^*)$ is not positive definite or $\boldsymbol{d}^{*T}B(\boldsymbol{x}^*,\boldsymbol{y}^*)\boldsymbol{d}^* = 0$.

Next, another sufficient condition of the optimal solution to (CNO) is obtained by solving (CNP).

Theorem 3.3. *Suppose that $(\boldsymbol{x}^*,\boldsymbol{y}^*) \in X(f)$ and f is a weak uniform CN function. Define a set*

$$\begin{array}{rcl} K_w(\boldsymbol{x}^*,\boldsymbol{y}^*) & = & \{(\boldsymbol{x},\boldsymbol{y}) \in R^n \times R^m \mid \nabla g(\boldsymbol{x}^*,\boldsymbol{y}^*)^\top[(\boldsymbol{x},\boldsymbol{y})-(\boldsymbol{x}^*,\boldsymbol{y}^*)] \\ & & +\frac{1}{2}[(\boldsymbol{x},\boldsymbol{y})-(\boldsymbol{x}^*,\boldsymbol{y}^*)]^\top B(\boldsymbol{x}^*,\boldsymbol{y}^*)[(\boldsymbol{x},\boldsymbol{y})-(\boldsymbol{x}^*,\boldsymbol{y}^*)] < 0\}. \end{array}$$

If $X(\boldsymbol{g}) \cap K_w(\boldsymbol{x}^*,\boldsymbol{y}^*) = \emptyset$ holds, then $(\boldsymbol{x}^*,\boldsymbol{y}^*)$ is an optimal solution to (CNP) and $\boldsymbol{x}^*$ is an optimal solution to (CNO).

By Theorem 3.3, if $(\boldsymbol{x}^*,\boldsymbol{y}^*)$ is an optimal solution to

$$\begin{array}{llll} \text{(WCNPP)}(\boldsymbol{x}^*,\boldsymbol{y}^*) & \min & \nabla g(\boldsymbol{x}^*,\boldsymbol{y}^*)^\top[(\boldsymbol{x},\boldsymbol{y})-(\boldsymbol{x}^*,\boldsymbol{y}^*)] \\ & & +\frac{1}{2}[(\boldsymbol{x},\boldsymbol{y})-(\boldsymbol{x}^*,\boldsymbol{y}^*)]^\top B(\boldsymbol{x}^*,\boldsymbol{y}^*)[(\boldsymbol{x},\boldsymbol{y})-(\boldsymbol{x}^*,\boldsymbol{y}^*)] \\ & \text{s.t.} & (\boldsymbol{x},\boldsymbol{y}) \in T_1(\boldsymbol{x}^*,\boldsymbol{y}^*) \cap X(\boldsymbol{g}), \end{array}$$

then $(\boldsymbol{x}^*,\boldsymbol{y}^*)$ is an optimal solution to (CNP) and $\boldsymbol{x}^*$ is an optimal solution to (CNO).

Corollary 3.4. *Suppose that $(\boldsymbol{x}^*,\boldsymbol{y}^*)\in X(f)$ and f is a uniform CN function. Define a set*

$$K_u(\boldsymbol{x}^*,\boldsymbol{y}^*)=\{(\boldsymbol{x},\boldsymbol{y})\in R^n\times R^m \mid \nabla g(\boldsymbol{x}^*,\boldsymbol{y}^*)^\top[(\boldsymbol{x},\boldsymbol{y})-(\boldsymbol{x}^*,\boldsymbol{y}^*)] + \frac{\bar{\rho}}{2}\|(\boldsymbol{x}-\boldsymbol{x}^*,\boldsymbol{y}-\boldsymbol{y}^*)\|^2<0\}. \tag{3.7}$$

If $X(\boldsymbol{g})\cap K_u(\boldsymbol{x}^*,\boldsymbol{y}^*)=\emptyset$ holds, then $(\boldsymbol{x}^*,\boldsymbol{y}^*)$ is an optimal solution to (CNP) and $\boldsymbol{x}^*$ is an optimal solution to (CNO).

By Corollary 3.1 and Corollary 3.4, if $(\boldsymbol{x}^*,\boldsymbol{y}^*)$ is an optimal solution to

$$\text{(UCNP)}(\boldsymbol{x}^*,\boldsymbol{y}^*) \qquad \begin{array}{ll} \min & \nabla g(\boldsymbol{x}^*,\boldsymbol{y}^*)^\top[(\boldsymbol{x},\boldsymbol{y})-(\boldsymbol{x}^*,\boldsymbol{y}^*)]+\frac{\bar{\rho}}{2}\|(\boldsymbol{x}-\boldsymbol{x}^*,\boldsymbol{y}-\boldsymbol{y}^*)\|^2 \\ \text{s.t.} & (\boldsymbol{x},\boldsymbol{y})\in T_1(\boldsymbol{x}^*,\boldsymbol{y}^*)\cap X(\boldsymbol{g}), \end{array}$$

then $(\boldsymbol{x}^*,\boldsymbol{y}^*)$ is an optimal solution to (CNP) and $\boldsymbol{x}^*$ is an optimal solution to (CNO).

Corollary 3.5. *Suppose that $(\boldsymbol{x}^*,\boldsymbol{y}^*)\in X(f)$ and f is a CN function. Define a set*

$$K_c(\boldsymbol{x}^*,\boldsymbol{y}^*)=\{(\boldsymbol{x},\boldsymbol{y})\in R^n\times R^m \mid \nabla g(\boldsymbol{x}^*,\boldsymbol{y}^*)^\top[(\boldsymbol{x},\boldsymbol{y})-(\boldsymbol{x}^*,\boldsymbol{y}^*)]<0\}. \tag{3.8}$$

If $X(\boldsymbol{g})\cap K_c(\boldsymbol{x}^*,\boldsymbol{y}^*)=\emptyset$ holds, then $(\boldsymbol{x}^*,\boldsymbol{y}^*)$ is an optimal solution to (CNP) and $\boldsymbol{x}^*$ is an optimal solution to (CNO).

By Corollary 3.2 and Corollary 3.5, if $(\boldsymbol{x}^*,\boldsymbol{y}^*)$ is an optimal solution to

$$\text{(LCNP)}(\boldsymbol{x}^*,\boldsymbol{y}^*) \qquad \begin{array}{ll} \min & \nabla g(\boldsymbol{x}^*,\boldsymbol{y}^*)^\top[(\boldsymbol{x},\boldsymbol{y})-(\boldsymbol{x}^*,\boldsymbol{y}^*)] \\ \text{s.t.} & (\boldsymbol{x},\boldsymbol{y})\in T_1(\boldsymbol{x}^*,\boldsymbol{y}^*)\cap X(\boldsymbol{g}), \end{array}$$

then $(\boldsymbol{x}^*,\boldsymbol{y}^*)$ is an optimal solution to (CNP) and $\boldsymbol{x}^*$ is an optimal solution to (CNO).

It is clear that $K_w(\boldsymbol{x}^*,\boldsymbol{y}^*)\subset K_c(\boldsymbol{x}^*,\boldsymbol{y}^*)$ and $K_u(\boldsymbol{x}^*,\boldsymbol{y}^*)\subset K_c(\boldsymbol{x}^*,\boldsymbol{y}^*)$.

Theorem 3.4. *Suppose that $(\boldsymbol{x}^*,\boldsymbol{y}^*)\in X(f)$ and f is a weak uniform CN function. If there is a neighborhood $O(\boldsymbol{x}^*,\boldsymbol{y}^*)$ of $(\boldsymbol{x}^*,\boldsymbol{y}^*)$ such that*

$$X(\boldsymbol{g})\cap O(\boldsymbol{x}^*,\boldsymbol{y}^*)\cap K_w(\boldsymbol{x}^*,\boldsymbol{y}^*)=\emptyset \tag{3.9}$$

holds, then $(\boldsymbol{x}^*,\boldsymbol{y}^*)$ is a local optimal solution to (CNP) and $\boldsymbol{x}^*$ is a local optimal solution to (CNO).

In particular, if f is a strong uniform CN function or a uniform CN function, the conclusion of Theorem 3.4 still holds.

Theorem 3.5. *Suppose that $(\boldsymbol{x}^*,\boldsymbol{y}^*)\in X(f)$ and f is a CN function. If there is a neighborhood $O(\boldsymbol{x}^*,\boldsymbol{y}^*)$ of $(\boldsymbol{x}^*,\boldsymbol{y}^*)$ such that*

$$X(\boldsymbol{g})\cap O(\boldsymbol{x}^*,\boldsymbol{y}^*)\cap K_c(\boldsymbol{x}^*,\boldsymbol{y}^*)=\emptyset \tag{3.10}$$

holds, then $(\boldsymbol{x}^*,\boldsymbol{y}^*)$ is a local optimal solution to (CNP) and $\boldsymbol{x}^*$ is a local optimal solution to (CNO).

For $(\boldsymbol{x}^*,\boldsymbol{y}^*)\in X(f)$, let

$$T_0(\boldsymbol{x}^*,\boldsymbol{y}^*)=\{\boldsymbol{d}\in R^n\times R^m \mid \nabla g_i(\boldsymbol{x}^*,\boldsymbol{y}^*)^\top\boldsymbol{d}=0, i=1,2,\cdots,r\}. \tag{3.11}$$

For $\boldsymbol{\alpha}\in R^r$, a Lagrange function of (CNP) is defined by

$$L(\boldsymbol{x},\boldsymbol{y},\boldsymbol{\alpha})=g(\boldsymbol{x},\boldsymbol{y})+\boldsymbol{\alpha}^\top\boldsymbol{g}(\boldsymbol{x},\boldsymbol{y}). \tag{3.12}$$

The following necessary conditions of (CNP) are obvious.

Theorem 3.6. *Suppose that $(\boldsymbol{x}^*,\boldsymbol{y}^*)\in X(f)$ is a local optimal solution to (CNP). Then*
(i) $\nabla g(\boldsymbol{x}^*,\boldsymbol{y}^*)^T\boldsymbol{d}\geq 0$ for $\boldsymbol{d}\in T_0(\boldsymbol{x}^*,\boldsymbol{y}^*)$.
(ii) If $\nabla g_i(\boldsymbol{x}^*,\boldsymbol{y}^*)(i=1,2,\cdots,r)$ is linearly independent, then there are $\alpha_1^*,\alpha_2^*,\cdots,\alpha_r^*$ such that

$$\nabla g(\boldsymbol{x}^*,\boldsymbol{y}^*)+\sum_{i=1}^{r}\alpha_i^*\nabla g_i(\boldsymbol{x}^*,\boldsymbol{y}^*)=0. \tag{3.13}$$

Furthermore, if $L(\boldsymbol{x},\boldsymbol{y},\boldsymbol{\alpha}^*)$ is convex on $(\boldsymbol{x},\boldsymbol{y})$, then $(\boldsymbol{x}^*,\boldsymbol{y}^*)$ is a global optimal solution to (CNP) and $\boldsymbol{x}^*$ is a global optimal solution to (CNO).

In Theorem 3.6, if $\alpha_1^*,\alpha_2^*,\cdots,\alpha_r^*\geq 0$, then $(\boldsymbol{x}^*,\boldsymbol{y}^*)$ is a global optimal solution to (CNP) and $\boldsymbol{x}^*$ is a global optimal solution to (CNO). The following examples demonstrate that the above sufficient conditions help determine an optimal solution to (CNP) or (CNO), which further help us to devise a global algorithm for (CNP), which brings a new way to study and solve nonconvex and nonsmooth optimization problems.

Example 3.1. Consider an optimization problem ([7, Example 2.1]):

$$\begin{array}{lll}\text{(EX8)} & \min & f(x_1,x_2)=(x_1+x_2-1)^2+\lambda(|x_1|^{\frac{1}{2}}+|x_2|^{\frac{1}{2}})\\ & s.t. & x_1,x_2\in R,\end{array}$$

where $f(x)$ is a non-smooth and non-convex function. By Example 2.6, $f(x)$ is a weak uniform exact CN function. Let $(\boldsymbol{x}^*,\boldsymbol{y}^*)\in X(f)$ and $(\boldsymbol{x},\boldsymbol{y})=(x_1,x_2,y_1,\cdots,y_6)$. The weak uniform exact CN optimization of (EX8) is defined by

$$\begin{array}{lll}\text{(MEX8)} & \min & g(\boldsymbol{x},\boldsymbol{y})=(x_1+x_2-1)^2+\lambda(y_1+y_4),\\ & s.t. & (\boldsymbol{x},\boldsymbol{y})\in X(f),\end{array}$$

where $X(f)=\{(\boldsymbol{x},\boldsymbol{y})\mid y_1^4-y_3=0,x_1^2-y_3=0,y_2^2-y_1=0,y_4^4-y_6=0,x_2^2-y_6=0,y_5^2-y_4=0\}$.

Then, the linear programming (LCNP-EX8) $(\boldsymbol{x}^*,\boldsymbol{y}^*)$ of (MEX8) at $(\boldsymbol{x}^*,\boldsymbol{y}^*)$ is defined by Corollary 3.2. Let $(\boldsymbol{x}^*,\boldsymbol{y}^*)=(0,0,0,0,0,0,0,0)\in X(f)$. $\boldsymbol{x}=(0,0)$ is an optimal solution to (EX8) for $\lambda\geq 8^{\frac{1}{4}}$ in [7]. It is clear that there is no optimal solution to (LCNP-EX8) at $(\boldsymbol{x}^*,\boldsymbol{y}^*)=(0,0,0,0,0,0,0,0)\in X(f)$. Now, a programming (WCNP-EX8) $(\boldsymbol{x}^*,\boldsymbol{y}^*)$ of (MEX8) at $(\boldsymbol{x}^*,\boldsymbol{y}^*)$ is defined by Theorem 3.1. Then, by Theorem 3.3, the objective value of (WCNP-EX8)$(\boldsymbol{x}^*,\boldsymbol{y}^*)$ at $(\boldsymbol{x}^*,\boldsymbol{y}^*)=(0,0,0,0,0,0,0,0)$ is computed such that it is not less than zero. Thus we obtain that $\boldsymbol{x}^*=(0,0)$ is an optimal solution to (EX8) for $\lambda\geq\frac{4}{3}\sqrt{\frac{2}{3}}(<8^{\frac{1}{4}})$. Chen et al. [7] pointed out that there is a smaller error bound $\beta^*<8^{\frac{1}{4}}$ that makes $\boldsymbol{x}^*=(0,0)$ an optimal solution to (EX8). Here, the above error bound $\beta^*=\frac{4}{3}\sqrt{\frac{2}{3}}$. This result demonstrates that the sufficient conditions in Theorem 3.3 for determining the global optimal solution are valid.

On the other hand, let $\boldsymbol{d}=(d_1,d_2,\cdots,d_8)^\top\in R^8$ for $(\boldsymbol{x},\boldsymbol{y})\in R^2\times R^6$. By Theorem 3.1, a programming (WCNP-EX8) $(\boldsymbol{x}^*,\boldsymbol{y}^*)$ of (MEX8) at $(\boldsymbol{x}^*,\boldsymbol{y}^*)=(0,0,0,0,0,0,0,0)$ is defined. $(d_1^*,d_2^*,0,0,0,0,0,0)$ is an optimal solution to (WCNP-EX8)$(\boldsymbol{x}^*,\boldsymbol{y}^*)$ when $d_1^*+d_2^*=1$. By (3.1) and (3.2), we have $\nabla g(\boldsymbol{x}^*,\boldsymbol{y}^*)=(-2,-2,\lambda,0,0,\lambda,0,0)^\top$, $\nabla g_1(\boldsymbol{x}^*,\boldsymbol{y}^*)=(0,0,0,0,-1,0,0,0)^\top$, $\nabla g_2(\boldsymbol{x}^*,\boldsymbol{y}^*)=(0,0,0,0,-1,0,0,0)^\top$, $\nabla g_3(\boldsymbol{x}^*,\boldsymbol{y}^*)=(0,0,-1,0,0,0,0,0)^\top$, $\nabla g_4(\boldsymbol{x}^*,\boldsymbol{y}^*)=(0,0,0,0,0,0,0,-1)^\top$, $\nabla g_5(\boldsymbol{x}^*,\boldsymbol{y}^*)=(0,0,0,0,0,0,0,-1)^\top$, and $\nabla g_6(\boldsymbol{x}^*,\boldsymbol{y}^*)=(0,0,0,0,0,-1,0,0)^\top$.

When $(\alpha_1^*, \alpha_2^*, \alpha_3^*, \alpha_4^*, \alpha_5^*, \alpha_6^*) = (0,0,\lambda,0,0,\lambda)$, and $d_1^* + d_2^* = 1$ with $\lambda > 0$, optimization condition (3.1) and (3.2) hold. But, KKT condition (3.5) and (3.6) are not true. So, $(\boldsymbol{x}^*, \boldsymbol{y}^*)$ is not a KKT point, and $\nabla g_i(\boldsymbol{x}^*, \boldsymbol{y}^*)(i = 1,2,\cdots,6)$ is not linear independent.

The above example demonstrates that optimization conditions (3.1) and (3.2) hold if there is an optimal solution to (CNP) when the objective value of (WCNP-EX8) $(\boldsymbol{x}^*, \boldsymbol{y}^*)$ has a lower bound. The optimization condition of the weak uniform CN form is better than that of CN function [23].

Example 3.2. Let the function $f(x_1,x_2) = (x_1 + x_2 - 1)^2 + \lambda \|(x_1,x_2)\|_0$ be nonconvex and discontinuous, where $\lambda > 0$.

$$\text{(EX9)} \quad \min \quad f(x_1,x_2) = (x_1+x_2-1)^2 + \lambda\|(x_1,x_2)\|_0$$
$$s.t. \quad x_1, x_2 \in R,$$

where $f(x)$ is a non-smooth and non-convex function, $f(x)$ is a weak uniform CN function. By Example 2.7, a weak uniform CN optimization of (EX9) is defined by

$$\text{(EX9)} \quad \min \quad g(\boldsymbol{x},\boldsymbol{y}) = (x_1+x_2-1)^2 + \lambda(y_1^2+y_2^2),$$
$$s.t. \quad (\boldsymbol{x},\boldsymbol{y}) \in X(f),$$

where $(\boldsymbol{x},\boldsymbol{y}) = (x_1,x_2,y_1,y_2,y_3,y_4)$ and $X(f)$ is defined by Example 2.7. Let $(\boldsymbol{x}^*,\boldsymbol{y}^*) \in X(f)$. When $(\boldsymbol{x}^*,\boldsymbol{y}^*) = (0,0,0,1,0,1) \in X(f)$, it is clear that $\boldsymbol{x} = (0,0)$ is an optimal solution to (EX9) for $\lambda \geq 2$. So, by Corollary 3.2, it is easily known that there is no optimal solution to (LCNP-EX9) $(\boldsymbol{x}^*,\boldsymbol{y}^*)$ and $X(g) \cap K_c(\boldsymbol{x}^*,\boldsymbol{y}^*) \neq \emptyset$ for $\lambda \geq 1$. But, by Theorem 3.3, it is clear that, for any $(x_1,x_2,y_1,\cdots,y_4) \in X(f)$ and $\lambda \geq 1$,

$$-2(x_1+x_2) + \lambda(2y_1+2y_3) + (x_1+x_2)^2 \geq -2(x_1+x_2) + 1 + (x_1+x_2)^2 \geq 0,$$

i.e., $X(f) \cap K_w(\boldsymbol{x}^*,\boldsymbol{y}^*) = \emptyset$. Hence, $\boldsymbol{x} = (0,0)$ is an optimal solution to (EX9) for $\lambda \geq 1$ by Theorem 3.3. It is easily checked that (3.1) and (3.2) hold.

Example 3.1 and 3.2 demonstrates that optimization conditions (3.1) and (3.2) probably hold if f is a weak uniform CN function when the KKT conditions (3.5) and (3.6) are not true.

4. DECOMPOSABLE ALGORITHM OF (CNP)

The CN function form $[g : g_1, g_2, \cdots, g_r]$ of f has more variables than f. Because CN function optimization $\min_{(\boldsymbol{x},\boldsymbol{y})\in X(\boldsymbol{g})} g(\boldsymbol{x},\boldsymbol{y})$ has more variables than (CNO), it is not easy to solve it, resulting in a scale problem. However, many of decomposable CN functions help to reduce the scale problem of CN function optimization. Now, a decomposable form of $(\boldsymbol{x},\boldsymbol{y}) \in X(\boldsymbol{g})$ of the CN function $f(\boldsymbol{x})$ on S is given. Let $f = [g : g_1, g_2, \cdots, g_r]$ be a CN form on S. $((\boldsymbol{x}_1,\boldsymbol{y}_1), (\boldsymbol{x}_2,\boldsymbol{y}_2), \cdots, (\boldsymbol{x}_p,\boldsymbol{y}_p))$ is called a decomposition of $(\boldsymbol{x},\boldsymbol{y})$ on S if it satisfies the following conditions:

(i) $\boldsymbol{x} = (\boldsymbol{x}_1,\boldsymbol{x}_2,\cdots,\boldsymbol{x}_p)^\top \in S$, where $\boldsymbol{x}_j \in R^{p_j} (j = 1,2,\cdots,p)$ is composed of p_j variables of $\boldsymbol{x}$ with $\sum_{j=1}^p p_j = n$;

(ii) $\boldsymbol{y} = (\boldsymbol{y}_1,\boldsymbol{y}_2,\cdots,\boldsymbol{y}_p)^\top \in R^m$, where $\boldsymbol{y}_j \in R^{q_j} (j = 1,2,\cdots,p)$ is composed of q_j variables of $\boldsymbol{y}$ with $\sum_{j=1}^p q_j = m$;

(iii) $((\boldsymbol{x}_1,\boldsymbol{y}_1), (\boldsymbol{x}_2,\boldsymbol{y}_2), \cdots, (\boldsymbol{x}_p,\boldsymbol{y}_p))$ is a rearrangement of $(\boldsymbol{x},\boldsymbol{y})$;

(iv) there are no identical variables between $(\boldsymbol{x}_k,\boldsymbol{y}_k)$ and $(\boldsymbol{x}_j,\boldsymbol{y}_j)$ for any $j,k=1,2,\cdots,p,k\neq j$.

Note that a decomposition of $(\boldsymbol{x},\boldsymbol{y})\in X(\boldsymbol{g})$ means that $(\boldsymbol{x},\boldsymbol{y})=((\boldsymbol{x}_1,\boldsymbol{y}_1),(\boldsymbol{x}_2,\boldsymbol{y}_2),\cdots,(\boldsymbol{x}_p,\boldsymbol{y}_p))$. p is called the decomposition number. We have $2\leq p\leq\min\{n,m\}$. $p=2$ is the minimum decomposition number, and $p=\min\{n,m\}$ is the maximum decomposition number. Let $(\boldsymbol{x}_j,\boldsymbol{y}_j\mid(\boldsymbol{x},\boldsymbol{y})):=((\boldsymbol{x}_1,\boldsymbol{y}_1),(\boldsymbol{x}_2,\boldsymbol{y}_2),\cdots,(\boldsymbol{x}_p,\boldsymbol{y}_p))$, where $(\boldsymbol{x}_j,\boldsymbol{y}_j)$ is a variable, that is, all $(\boldsymbol{x}_k,\boldsymbol{y}_k)(k=1,2,\cdots,p,k\neq j)$ are fixed except $(\boldsymbol{x}_j,\boldsymbol{y}_j)$. For each $(\boldsymbol{x}_j,\boldsymbol{y}_j)$, $j=1,2,\cdots,p$, define

$$X(\boldsymbol{g}_j)=\{(\boldsymbol{x}_j,\boldsymbol{y}_j)\in R^{p_j}\times R^{q_j}\mid g_i(\boldsymbol{x}_j,\boldsymbol{y}_j\mid(\boldsymbol{x},\boldsymbol{y}))=0,i=1,2,\cdots,r\},$$

where $(\boldsymbol{x},\boldsymbol{y})\in X(\boldsymbol{g})$. It follows that $X(\boldsymbol{g})=X(\boldsymbol{g}_1)\times X(\boldsymbol{g}_2)\cdots\times X(\boldsymbol{g}_p)$.

Definition 4.1. Let the CN form of f be $[g:g_1,g_2,\cdots,g_r]$ on S. If each function $g_1,g_2,\cdots,g_r$ relates only to one of the variables: $\{(\boldsymbol{x}_1,\boldsymbol{y}_1),(\boldsymbol{x}_2,\boldsymbol{y}_2),\ \cdots,(\boldsymbol{x}_p,\boldsymbol{y}_p)\}$ respectively, i.e., the following constraints $g_i(\boldsymbol{x},\boldsymbol{y})$ includes its decision variables that belongs to $(\boldsymbol{x}_j,\boldsymbol{y}_j)$:

$$g_i(\boldsymbol{x},\boldsymbol{y})=g_i(\boldsymbol{x}_j,\boldsymbol{y}_j)=0,i=1,2,\cdots,r.$$

Let $\boldsymbol{g}_j(\boldsymbol{x}_j,\boldsymbol{y}_j)$ be composed of r_j constraints of $g_t(\boldsymbol{x}_j,\boldsymbol{y}_j)(t=j_1,j_2,\cdots,r_j)$ in $\{g_1(\boldsymbol{x},\boldsymbol{y}),g_2(\boldsymbol{x},\boldsymbol{y}),\cdots,g_r(\boldsymbol{x},\boldsymbol{y})\}$. If $\boldsymbol{g}_j(\boldsymbol{x}_j,\boldsymbol{y}_j)=0$ can be expressed equivalently as

$$\boldsymbol{g}_j(\boldsymbol{x}_j,\boldsymbol{y}_j)=(g_{j_1}(\boldsymbol{x}_j,\boldsymbol{y}_j),g_{j_2}(\boldsymbol{x}_j,\boldsymbol{y}_j),\cdots,g_{j_{r_j}}(\boldsymbol{x}_j,\boldsymbol{y}_j))=0,j=1,2,\cdots,p,$$

where $g_t(\boldsymbol{x}_j,\boldsymbol{y}_j)\in\{g_1(\boldsymbol{x},\boldsymbol{y}),g_2(\boldsymbol{x},\boldsymbol{y}),\cdots,g_r(\boldsymbol{x},\boldsymbol{y})\}$, $t=j_1,j_2,\cdots,r_j$ with $\sum\limits_{j=1} r_j=r$, i.e.,

$$\boldsymbol{g}(\boldsymbol{x},\boldsymbol{y})=(\boldsymbol{g}_1(\boldsymbol{x}_1,\boldsymbol{y}_1),\boldsymbol{g}_2(\boldsymbol{x}_2,\boldsymbol{y}_2),\cdots,\boldsymbol{g}_p(\boldsymbol{x}_p,\boldsymbol{y}_p))=0, \tag{4.1}$$

then f is called a decomposable CN function on S, and p is called the decomposable number. If there is not any decomposable CN function form, then f is called an undecomposable CN function. (4.1) shows that each function $g_i(\boldsymbol{x},\boldsymbol{y})$ only relates to some variable $(\boldsymbol{x}_j,\boldsymbol{y}_j)$ in $\{(\boldsymbol{x}_1,\boldsymbol{y}_1),(\boldsymbol{x}_2,\boldsymbol{y}_2),\cdots,(\boldsymbol{x}_p,\boldsymbol{y}_p)\}$.

The f in Example 2.6 is a decomposable CN function. Another example of decomposable CN functions is given as follows:

$$f(x_1,x_2,x_3,x_4,x_5)=(x_1+x_2+x_3+x_4+x_5-1)^2+\lambda\|(x_1,x_2,x_3,x_4,x_5)\|_0,$$

where $\lambda>0$. Let $\boldsymbol{x}=(x_1,x_2,x_3,x_4,x_5)^\top$ and $\boldsymbol{y}=(y_1,y_2\cdots,y_{10})^\top\in R^{10}$. A CN function form of f is defined by

$$\begin{aligned}
g(\boldsymbol{x},\boldsymbol{y}) &= (x_1+x_2+x_3+x_4+x_5-1)^2+\beta(y_1^2+y_3^2+y_5^2+y_7^2+y_9^2):\\
g_i(\boldsymbol{x},\boldsymbol{y}) &= (x_i+y_i-1)^2-y_{i+5}=0,\ i=1,2,3,4,5,\\
g_{i+5}(\boldsymbol{x},\boldsymbol{y}) &= x_i^2+(y_i-1)^2-y_{i+5}=0,\ i=1,2,3,4,5,\\
g_{i+10}(\boldsymbol{x},\boldsymbol{y}) &= y_i^2-y_i=0, \qquad\qquad i=1,2,3,4,5.
\end{aligned}$$

There are many decomposable CN function forms for the above function f. If let $p=2$, $\boldsymbol{x}_1=(x_1,x_2),\boldsymbol{x}_2=(x_3,x_4,x_5)$, $\boldsymbol{y}_1=(y_1,y_2,y_6,y_7)$, and $\boldsymbol{y}_2=(y_3,y_4,y_5,y_8,y_9,y_{10})$, then a decomposable CN function form of f is defined by

$$\begin{aligned}
\boldsymbol{g}_1(\boldsymbol{x}_1,\boldsymbol{y}_1) &= (g_1(\boldsymbol{x},\boldsymbol{y}),g_2(\boldsymbol{x},\boldsymbol{y}),g_6(\boldsymbol{x},\boldsymbol{y}),g_7(\boldsymbol{x},\boldsymbol{y}),g_{11}(\boldsymbol{x},\boldsymbol{y}),g_{12}(\boldsymbol{x},\boldsymbol{y}))=0,\\
\boldsymbol{g}_2(\boldsymbol{x}_2,\boldsymbol{y}_2) &= (g_3(\boldsymbol{x},\boldsymbol{y}),g_4(\boldsymbol{x},\boldsymbol{y}),g_5(\boldsymbol{x},\boldsymbol{y}),g_8(\boldsymbol{x},\boldsymbol{y}),g_9(\boldsymbol{x},\boldsymbol{y}),g_{10}(\boldsymbol{x},\boldsymbol{y}),g_{13}(\boldsymbol{x},\boldsymbol{y}),\\
&\quad g_{14}(\boldsymbol{x},\boldsymbol{y}),g_{15}(\boldsymbol{x},\boldsymbol{y}))=0,
\end{aligned}$$

where

$$\begin{aligned} X(\boldsymbol{g}_1) = \{(\boldsymbol{x}_1, \boldsymbol{y}_1) \quad &| \quad (g_1(\boldsymbol{x},\boldsymbol{y}), g_2(\boldsymbol{x},\boldsymbol{y}), g_6(\boldsymbol{x},\boldsymbol{y}), g_7(\boldsymbol{x},\boldsymbol{y}), g_{11}(\boldsymbol{x},\boldsymbol{y}), g_{12}(\boldsymbol{x},\boldsymbol{y})) = 0\}, \\ X(\boldsymbol{g}_2) = \{(\boldsymbol{x}_2, \boldsymbol{y}_2) \quad &| \quad (g_3(\boldsymbol{x},\boldsymbol{y}), g_4(\boldsymbol{x},\boldsymbol{y}), g_5(\boldsymbol{x},\boldsymbol{y}), g_8(\boldsymbol{x},\boldsymbol{y}), g_9(\boldsymbol{x},\boldsymbol{y}), g_{10}(\boldsymbol{x},\boldsymbol{y}), g_{13}(\boldsymbol{x},\boldsymbol{y}), \\ & \quad\; g_{14}(\boldsymbol{x},\boldsymbol{y}), g_{15}(\boldsymbol{x},\boldsymbol{y})) = 0\}. \end{aligned}$$

For each $j = 1, 2, \cdots, p$ and $i = 1, 2, \cdots, r$, let a gradient of $g(\boldsymbol{x},\boldsymbol{y})$ and $g_i(\boldsymbol{x},\boldsymbol{y})$ on $(\boldsymbol{x}_j, \boldsymbol{y}_j)$ be defined respectively by $\nabla_j g(\boldsymbol{x},\boldsymbol{y}) := \nabla_{(\boldsymbol{x}_i,\boldsymbol{y}_j)} g(\boldsymbol{x}_i, \boldsymbol{y}_j \mid (\boldsymbol{x},\boldsymbol{y}))$ and $\nabla_j g_i(\boldsymbol{x},\boldsymbol{y}) := \nabla_{(\boldsymbol{x}_i,\boldsymbol{y}_j)} g_i(\boldsymbol{x}_i, \boldsymbol{y}_j \mid (\boldsymbol{x},\boldsymbol{y}))$, where all $(\boldsymbol{x}_k, \boldsymbol{y}_k)(k = 1, 2, \cdots, p, k \neq j)$ are fixed except for $(\boldsymbol{x}_j, \boldsymbol{y}_j)$. We have

$$\nabla g(\boldsymbol{x},\boldsymbol{y}) = (\nabla_1 g(\boldsymbol{x},\boldsymbol{y}), \nabla_2 g(\boldsymbol{x},\boldsymbol{y}), \nabla_p g(\boldsymbol{x},\boldsymbol{y}))^\top, \tag{4.2}$$

$$\nabla g_i(\boldsymbol{x},\boldsymbol{y}) = (\nabla_1 g_i(\boldsymbol{x},\boldsymbol{y}), \nabla_2 g_i(\boldsymbol{x},\boldsymbol{y}), \nabla_p g_i(\boldsymbol{x},\boldsymbol{y}))^\top. \tag{4.3}$$

For $\boldsymbol{d}_j \in R^{p_j} \times R^{q_j}, j = 1, 2, \cdots, p$, let

$$T_j(\boldsymbol{x}_j, \boldsymbol{y}_j) = \{\boldsymbol{d}_j \in R^{p_j} \times R^{q_j} \mid \nabla_j g_i(\boldsymbol{x},\boldsymbol{y})^\top \boldsymbol{d}_j \leq 0, i = 1, 2 \cdots, r\}. \tag{4.4}$$

By (4.4), we have

$$T_1(\boldsymbol{x}_1, \boldsymbol{y}_1) \times T_j(\boldsymbol{x}_2, \boldsymbol{y}_2) \cdots \times T_p(\boldsymbol{x}_p, \boldsymbol{y}_p) \subset T(\boldsymbol{x},\boldsymbol{y}).$$

By Definition 4.1, we have following propositions.

Proposition 4.1. *Let f be a decomposable CN function and $f = [g : g_1, g_2, \cdots, g_r]$, where $((\boldsymbol{x}_1, \boldsymbol{y}_1), (\boldsymbol{x}_2, \boldsymbol{y}_2), \cdots, (\boldsymbol{x}_p, \boldsymbol{y}_p))$ is a decomposition of $(\boldsymbol{x},\boldsymbol{y})$. Then*

$$T_1(\boldsymbol{x}_1, \boldsymbol{y}_1) \times T_2(\boldsymbol{x}_2, \boldsymbol{y}_2) \cdots \times T_p(\boldsymbol{x}_p, \boldsymbol{y}_p) = T(\boldsymbol{x},\boldsymbol{y}). \tag{4.5}$$

Proposition 4.2. *Let $f = [g : g_1, g_2, \cdots, g_r]$ be a decomposable CN function and $((\boldsymbol{x}_1, \boldsymbol{y}_1), (\boldsymbol{x}_2, \boldsymbol{y}_2), \cdots, (\boldsymbol{x}_p, \boldsymbol{y}_p))$ be a decomposition of $(\boldsymbol{x},\boldsymbol{y})$. If $g(\boldsymbol{x},\boldsymbol{y})$ is a (weak or strong) uniform convex function on $(\boldsymbol{x},\boldsymbol{y})$, then $g(\boldsymbol{x}_i, \boldsymbol{y}_j \mid (\boldsymbol{x},\boldsymbol{y}))$ is a (weak or strong) uniform convex function on $(\boldsymbol{x}_j, \boldsymbol{y}_j)(j = 1, 2, \cdots, p)$.*

An example of decomposable CN function is given as follows.

Example 4.1. [8] The function in sparse optimization is

$$f(\boldsymbol{x}) = \|A\boldsymbol{x} - \boldsymbol{b}\|^2 + \lambda \|\boldsymbol{x}\|_0, \tag{4.6}$$

where $A \in R^m \times R^n$, $\boldsymbol{b} \in R^m$, $\lambda > 0$, and $\|\boldsymbol{x}\|_0$ is 0-norm. Then, a weak uniform CN function form of $f(\boldsymbol{x})$ is obtained by

$$\begin{aligned} g(\boldsymbol{x},\boldsymbol{y}) &= \|A\boldsymbol{x} - \boldsymbol{b}\|^2 + \lambda \sum_{i=1}^{n} y_i^2 : \\ g_i(\boldsymbol{x},\boldsymbol{y}) &= (x_i + y_i - 1)^2 - y_{i+n} = 0, \;\; i = 1, 2, \cdots, n, \\ g_{i+n}(\boldsymbol{x},\boldsymbol{y}) &= x_i^2 + (y_i - 1)^2 - y_{i+n} = 0, \;\; i = 1, 2, \cdots, n, \\ g_{i+2n}(\boldsymbol{x},\boldsymbol{y}) &= y_i^2 - y_i = 0, \;\; i = 1, 2, \cdots, n, \end{aligned}$$

where $\boldsymbol{y} \in R^{2n}$. Let $\boldsymbol{w}_i = (x_i, y_i, y_{i+n})$, $i = 1, 2, \cdots, n$. Then, a decomposable CN form of $f(\boldsymbol{x})$ at the maximum decomposition number $p = n$ is defined by

$$\begin{aligned} g(\boldsymbol{w}_1, \boldsymbol{w}_2, \cdots, \boldsymbol{w}_n) &= \|A\boldsymbol{x} - \boldsymbol{b}\|^2 + \lambda \sum_{i=1}^{n} y_i^2 : \\ \boldsymbol{g}_i(\boldsymbol{w}_i) &= ((x_i + y_i - 1)^2 - y_{i+n}, x_i^2 + (y_i - 1)^2 - y_{i+n}, y_i^2 - y_i) \\ &= \boldsymbol{0},\ i = 1, 2, \cdots, n. \end{aligned}$$

If $((\boldsymbol{x}_1, \boldsymbol{y}_1), (\boldsymbol{x}_2, \boldsymbol{y}_2), \cdots, (\boldsymbol{x}_p, \boldsymbol{y}_p))$ is a decomposition of $(\boldsymbol{x}, \boldsymbol{y})$), (CNP) is redefined as

$$\begin{aligned} \text{(CNP)} \qquad & \min \quad g(\boldsymbol{x}, \boldsymbol{y}) = g((\boldsymbol{x}_1, \boldsymbol{y}_1), (\boldsymbol{x}_2, \boldsymbol{y}_2), \cdots, (\boldsymbol{x}_p, \boldsymbol{y}_p)) \\ & \text{s.t.} \quad \boldsymbol{g}_j(\boldsymbol{x}_j, \boldsymbol{y}_j | (\boldsymbol{x}, \boldsymbol{y})) = 0, j = 1, 2, \cdots, p, \\ & \qquad (\boldsymbol{x}_j, \boldsymbol{y}_j) \in R^{p_j} \times R^{q_j}, j = 1, 2, \cdots, p, \end{aligned}$$

where f is not necessarily a decomposable CN function. For each $j = 1, 2, \cdots, p$, the jth subproblem of (CNP) is defined by

$$\begin{aligned} \text{(CNP)}_j \qquad & \min \quad g(\boldsymbol{x}_i, \boldsymbol{y}_j | (\boldsymbol{x}, \boldsymbol{y})) \\ & \text{s.t.} \quad (\boldsymbol{x}_i, \boldsymbol{y}_j | (\boldsymbol{x}, \boldsymbol{y})) \in X(\boldsymbol{g}_j), \end{aligned}$$

where $(\boldsymbol{x}_j, \boldsymbol{y}_j)$ is the variable, i.e., all $(\boldsymbol{x}_k, \boldsymbol{y}_k)(k = 1, 2, \cdots, p, k \neq j)$ are fixed except for $(\boldsymbol{x}_j, \boldsymbol{y}_j)$ in problem (CNP)$_j$. Then, the optimal solution to (CNP) is expected to be obtained by solving p subproblems (CNP)$_j$, $j = 1, 2, \cdots, p$. If f is a decomposable CN function and there is an optimal solution to (CNP) or (CNO), there are optimal solutions to all subproblems (CNP)$_1$,(CNP)$_2$,$\cdots$, (CNP)$_p$ of (CNP). But, if there is not an optimal solution to anyone of all subproblems (CNP)$_1$,(CNP)$_2$,$\cdots$, (CNP)$_p$ of (CNP), there is not an optimal solution to (CNP) or (CNO).

The following theorems are true.

Theorem 4.1. *Suppose that* $(\boldsymbol{x}^*, \boldsymbol{y}^*) = ((\boldsymbol{x}_1^*, \boldsymbol{y}_1^*), (\boldsymbol{x}_2^*, \boldsymbol{y}_2^*), \cdots, (\boldsymbol{x}_p^*, \boldsymbol{y}_p^*)) \in X(f)$ *and* f *is a decomposable CN function. For each* $j = 1, 2, \cdots, p$*, let*

$$\begin{aligned} \textit{(LCNP)}_j(\boldsymbol{x}_j^*, \boldsymbol{y}_j^*) \qquad & \min \quad \nabla_j g(\boldsymbol{x}^*, \boldsymbol{y}^*)^\top \boldsymbol{d}_j \\ & \textit{s.t.} \quad \boldsymbol{d}_j \in T_j(\boldsymbol{x}_j^*, \boldsymbol{y}_j^*). \end{aligned}$$

For all $j = 1, 2, \cdots, p$*, if* $\boldsymbol{d}_j^*$ *is an optimal solution to (LCNP)*$_j(\boldsymbol{x}_j^*, \boldsymbol{y}_j^*)$ *such that* $\nabla_j g(\boldsymbol{x}^*, \boldsymbol{y}^*)^\top \boldsymbol{d}_j^* \geq 0$*, then* $(\boldsymbol{x}^*, \boldsymbol{y}^*)$ *is an optimal solution to (CNP) and* $\boldsymbol{x}^*$ *is an optimal solution to (CNO). Furthermore, if there is a* $\boldsymbol{d}'$ *such that* $\nabla g_i(\boldsymbol{x}^*, \boldsymbol{y}^*)^\top \boldsymbol{d}' < 0, i = 1, 2, \cdots, r$*, then there are* $\alpha_i^* \geq 0, i = 1, 2, \cdots, r$ *such that*

$$\nabla g(\boldsymbol{x}^*, \boldsymbol{y}^*) + \sum_{i=1}^{r} \alpha_i^* \nabla g_i(\boldsymbol{x}^*, \boldsymbol{y}^*) = 0. \tag{4.7}$$

Theorem 4.1 demonstrates that the optimal solution to (CNP) is expected to be obtained by solving p subproblems (CNP)$_j$, $j = 1, 2, \cdots, p$. These subproblems (CNP)$_j(j = 1, 2, \cdots, p)$ have smaller scale although the problem (CNP) has more variables.

Theorem 4.2. *Suppose that* $(\boldsymbol{x}^*,\boldsymbol{y}^*)=((\boldsymbol{x}_1^*,\boldsymbol{y}_1^*),(\boldsymbol{x}_2^*,\boldsymbol{y}_2^*),\cdots,(\boldsymbol{x}_p^*,\boldsymbol{y}_p^*))\in X(f)$ *and* f *is a decomposable uniform CN function. For each* $j=1,2,\cdots,p$, *let*

$$\text{(UCNP)}_j(\boldsymbol{x}_j^*,\boldsymbol{y}_j^*) \qquad \min \quad \nabla_j g(\boldsymbol{x}^*,\boldsymbol{y}^*)^\top \boldsymbol{d}_j+\frac{\bar{\rho}}{2}\|\boldsymbol{d}_j\|^2 \qquad \text{s.t.} \quad \boldsymbol{d}_j\in T_j(\boldsymbol{x}_j^*,\boldsymbol{y}_j^*).$$

For all $j=1,2,\cdots,p$, *if* $\boldsymbol{d}_j^*$ *is an optimal solution to* $\text{(UCNP)}_j(\boldsymbol{x}_j^*,\boldsymbol{y}_j^*)$ *such that* $\nabla_j g(\boldsymbol{x}^*,\boldsymbol{y}^*)^\top\boldsymbol{d}_j^*+\frac{\bar{\rho}}{2}\|\boldsymbol{d}_j^*\|^2\geq 0$, *then* $(\boldsymbol{x}^*,\boldsymbol{y}^*)$ *is an optimal solution to (CNP) and* $\boldsymbol{x}^*$ *is an optimal solution to (CNO). Furthermore, if there is a* $\boldsymbol{d}'$ *such that* $\nabla g_i(\boldsymbol{x}^*,\boldsymbol{y}^*)^\top\boldsymbol{d}'<0, i=1,2,\cdots,r$, *then there are* $\alpha_i^*\geq 0, i=1,2,\cdots,r$ *such that*

$$\frac{\bar{\rho}}{2}\boldsymbol{d}_j^*+\nabla g(\boldsymbol{x}^*,\boldsymbol{y}^*)+\sum_{i=1}^r\alpha_i^*\nabla g_i(\boldsymbol{x}^*,\boldsymbol{y}^*)=0. \tag{4.8}$$

Theorem 4.3. *Suppose that* $(\boldsymbol{x}^*,\boldsymbol{y}^*)\in X(f)$ *and* f *is an uniform decomposable CN function. For each* $j=1,2,\cdots,p$, *define a set*

$$K_{uj}(\boldsymbol{x}_j^*,\boldsymbol{y}_j^*)=\{(\boldsymbol{x}_j,\boldsymbol{y}_j)\in R^{p_j}\times R^{q_j}\mid \nabla_j g(\boldsymbol{x}^*,\boldsymbol{y}^*)^\top(\boldsymbol{x}_j-\boldsymbol{x}_j^*,\boldsymbol{y}_j-\boldsymbol{y}_j^*) +\frac{\bar{\rho}}{2}\|(\boldsymbol{x}_j-\boldsymbol{x}_j^*,\boldsymbol{y}_j-\boldsymbol{y}_j^*)\|^2<0\}. \tag{4.9}$$

If $X(\boldsymbol{g}_j)\cap K_{uj}(\boldsymbol{x}_j^*,\boldsymbol{y}_j^*)=\emptyset$ *holds, then* $(\boldsymbol{x}^*,\boldsymbol{y}^*)$ *is an optimal solution to (CNP) and* $\boldsymbol{x}^*$ *is an optimal solution to (CNO).*

Theorem 4.4. *Suppose that* $(\boldsymbol{x}^*,\boldsymbol{y}^*)\in X(f)$ *and* f *is a decomposable CN function. For each* $j=1,2,\cdots,p$, *define a set*

$$K_{cj}(\boldsymbol{x}_j^*,\boldsymbol{y}_j^*)=\{(\boldsymbol{x}_j,\boldsymbol{y}_j)\in R^{p_j}\times R^{q_j}\mid \nabla_j g(\boldsymbol{x}^*,\boldsymbol{y}^*)^\top(\boldsymbol{x}_j-\boldsymbol{x}_j^*,\boldsymbol{y}_j-\boldsymbol{y}_j^*)<0\}. \tag{4.10}$$

If $X(\boldsymbol{g}_j)\cap K_{cj}(\boldsymbol{x}_j^*,\boldsymbol{y}_j^*)=\emptyset$ *holds, then* $(\boldsymbol{x}^*,\boldsymbol{y}^*)$ *is an optimal solution to (CNP) and* $\boldsymbol{x}^*$ *is an optimal solution to (CNO).*

Now, an augmented Lagrange penalty function for (CNP) with decomposable variable is defined. Next, an algorithm is proposed by the augmented Lagrange penalty function for (CNP) and their convergence is proved.

Let $\boldsymbol{\alpha}_j\in R^{r_j}$, $j=1,2,\cdots,p$, be Lagrange parameters and $\sigma>0$ be a penalty parameter, and $\boldsymbol{\alpha}=(\boldsymbol{\alpha}_1,\boldsymbol{\alpha}_2,\cdots,\boldsymbol{\alpha}_p)$. Suppose that $((\boldsymbol{x}_1,\boldsymbol{y}_1),(\boldsymbol{x}_2,\boldsymbol{y}_2),\cdots,(\boldsymbol{x}_p,\boldsymbol{y}_p))$ is a decomposable variable of $(\boldsymbol{x},\boldsymbol{y})$, and f is not necessarily a decomposable CN function. In order to solve (CNP), the augmented Lagrange penalty functions for all subproblems $\text{(CNP)}_j(j=1,2,\cdots,p)$ are defined by

$$A_j(\boldsymbol{x}_i,\boldsymbol{y}_j\mid(\boldsymbol{x},\boldsymbol{y});\boldsymbol{\alpha}_j,\sigma) = g(\boldsymbol{x}_i,\boldsymbol{y}_j|(\boldsymbol{x},\boldsymbol{y}))+\boldsymbol{\alpha}_j^\top\boldsymbol{g}_j(\boldsymbol{x}_j,\boldsymbol{y}_j|(\boldsymbol{x},\boldsymbol{y})) +\frac{1}{2}\sigma\|\boldsymbol{g}_j(\boldsymbol{x}_j,\boldsymbol{y}_j|(\boldsymbol{x},\boldsymbol{y}))\|^2, \tag{4.11}$$

where $(\boldsymbol{x}_j,\boldsymbol{y}_j)$ is variable, i.e., all $(\boldsymbol{x}_k,\boldsymbol{y}_k)(k=1,2,\cdots,p,k\neq j)$ are fixed except for $(\boldsymbol{x}_j,\boldsymbol{y}_j)$. By (4.11), for $j=1,2,\cdots,p$, define an unconstraint optimization problem

$$\text{(CNP)}_j(\boldsymbol{\alpha}_j,\sigma) \qquad \min \quad A_j(\boldsymbol{x}_j,\boldsymbol{y}_j|(\boldsymbol{x},\boldsymbol{y});\boldsymbol{\alpha}_j,\sigma) \qquad \text{s.t.} \quad (\boldsymbol{x}_j,\boldsymbol{y}_j)\in R^{p_j}\times R^{q_j}.$$

To solve the problem $(\text{CNP})_j(\boldsymbol{\alpha}_j,\sigma)$, an algorithm involving an augmented Lagrange penalty function for (CNP)(which is called Algorithm 1) is proposed.

Algorithm 1.

Step 1: Let $\varepsilon>0,\sigma_1>0,N>1,\boldsymbol{w}_j^0=(\boldsymbol{x}_j^0,\boldsymbol{y}_j^0)\in R^{p_j}\times R^{q_j}(j=1,2,\cdots,p)$, and $\boldsymbol{\alpha}_j^1\in R^{r_j}(j=1,2,\cdots,p)$, $k=1$.

Step 2.1: Let $j=1$. If a point $(\boldsymbol{x}^{k-1},\boldsymbol{y}^{k-1})_1=((\boldsymbol{x}_1^{k-1},\boldsymbol{y}_1^{k-1}),(\boldsymbol{x}_2^{k-1},\boldsymbol{y}_2^{k-1}),\cdots,(\boldsymbol{x}_p^{k-1},\boldsymbol{y}_p^{k-1}))$ is obtained, then find $(\boldsymbol{x}_1^k,\boldsymbol{y}_1^k)\in R^{p_1}\times R^{q_1}$ to the subproblem

$$\min_{(\boldsymbol{x}_1,\boldsymbol{y}_1)} A_1^k(\boldsymbol{x}_1,\boldsymbol{y}_1|(\boldsymbol{x}^{k-1},\boldsymbol{y}^{k-1})_1;\boldsymbol{\alpha}_1^k,\sigma_k)$$

such that $\nabla_1 A_1^k(\boldsymbol{x}_1^k,\boldsymbol{y}_1^k|(\boldsymbol{x}^{k-1},\boldsymbol{y}^{k-1})_1;\boldsymbol{\alpha}_1^k,\sigma_k)=0$, where $(\boldsymbol{x}_1,\boldsymbol{y}_1)$ of function

$$A_1^k(\boldsymbol{x}_1,\boldsymbol{y}_1|(\boldsymbol{x}^{k-1},\boldsymbol{y}^{k-1})_1;\boldsymbol{\alpha}_1^k,\sigma_k)$$

is variable, that is, all $(\boldsymbol{x}_s^{k-1},\boldsymbol{y}_s^{k-1})(s=2,\cdots,p)$ are fixed except for $(\boldsymbol{x}_1,\boldsymbol{y}_1)$. Let $j=2$ and go to Step 2.2.

Step 2.2: Let $j>1$. If a point

$$(\boldsymbol{x}^{k-1},\boldsymbol{y}^{k-1})_j=((\boldsymbol{x}_1^k,\boldsymbol{y}_1^k),\cdots,(\boldsymbol{x}_{j-1}^k,\boldsymbol{y}_{j-1}^k),(\boldsymbol{x}_j^{k-1},\boldsymbol{y}_j^{k-1}),\cdots,(\boldsymbol{x}_p^{k-1},\boldsymbol{y}_p^{k-1}))$$

is obtained, then find $(\boldsymbol{x}_j^k,\boldsymbol{y}_j^k)\in R^{p_j}\times R^{q_j}$ to the subproblem

$$\min_{(\boldsymbol{x}_j,\boldsymbol{y}_j)} A_j^k(\boldsymbol{x}_j,\boldsymbol{y}_j|(\boldsymbol{x}^{k-1},\boldsymbol{y}^{k-1})_j;\boldsymbol{\alpha}_j^k,\sigma_k)$$

such that $\nabla_j A_j^k(\boldsymbol{x}_j^k,\boldsymbol{y}_j^k|(\boldsymbol{x}^{k-1},\boldsymbol{y}^{k-1})_j;\boldsymbol{\alpha}_j^k,\sigma_k)=0$, where

$$(\boldsymbol{x}_j,\boldsymbol{y}_j|(\boldsymbol{x}^{k-1},\boldsymbol{y}^{k-1})_j=((\boldsymbol{x}_1^k,\boldsymbol{y}_1^k),\cdots,(\boldsymbol{x}_{j-1}^k,\boldsymbol{y}_{j-1}^k),(\boldsymbol{x}_j,\boldsymbol{y}_j),(\boldsymbol{x}_{j+1}^{k-1},\boldsymbol{y}_{j+1}^{k-1}),\cdots,(\boldsymbol{x}_p^{k-1},\boldsymbol{y}_p^{k-1})),$$

i.e., $(\boldsymbol{x}_j,\boldsymbol{y}_j)$ of $A_j^k(\boldsymbol{x}_j,\boldsymbol{y}_j|(\boldsymbol{x}^{k-1},\boldsymbol{y}^{k-1})_j;\boldsymbol{\alpha}_j^k,\sigma_k)$ is variable, i.e., all $(\boldsymbol{x}_s^k,\boldsymbol{y}_s^k)(s=1,2,\cdots,j-1)$ and $(\boldsymbol{x}_s^{k-1},\boldsymbol{y}_s^{k-1})(s=j+1,j+2,\cdots,p)$ are fixed except for $(\boldsymbol{x}_j,\boldsymbol{y}_j)$. Go to Step 2.3

Step 2.3: If $j=p$, go to Step 3. Otherwise, set $j:=j+1$ and go to Step 2.2.

Step 3: If $(\boldsymbol{x}^k,\boldsymbol{y}^k)=(\boldsymbol{x}^{k-1},\boldsymbol{y}^{k-1})\in X(f)$ and $L(\boldsymbol{x},\boldsymbol{y},\boldsymbol{\alpha}^k)$ is convex on $(\boldsymbol{x},\boldsymbol{y})$ (see (3.12)), then stop and $\boldsymbol{x}^k$ is an optimal solution to (CNO). Otherwise, go to Step 4.

Step 4: If $\|\boldsymbol{g}(\boldsymbol{x}^k,\boldsymbol{y}^k)\|=\sum_{j=1}^p\|\boldsymbol{g}_j(\boldsymbol{x}_j^k,\boldsymbol{y}_j^k)\|<\varepsilon$, then stop and $\boldsymbol{x}^k$ is an approximate solution to (CNO). Otherwise, for $j=1,2,\cdots,p$, let $\boldsymbol{\alpha}_j^{k+1}=\boldsymbol{\alpha}_j^k+\sigma_k\boldsymbol{g}_j(\boldsymbol{x}_j^k,\boldsymbol{y}_j^k)$, $\sigma_{k+1}=N\sigma_k$, $k:=k+1$ and go to Step 2.1.

Note that, in Step 3, if $(\boldsymbol{x}^k,\boldsymbol{y}^k)=(\boldsymbol{x}^{k-1},\boldsymbol{y}^{k-1})\in X(f)$ holds, then$(\boldsymbol{x}_j^k,\boldsymbol{y}_j^k)\in R^{p_j}\times R^{q_j}(j=1,2,\cdots,p)$. So, for all $(j=1,2,\cdots,p$,

$$\nabla_j A_j^k(\boldsymbol{x}_j^k,\boldsymbol{y}_j^k|(\boldsymbol{x}^{k-1},\boldsymbol{y}^{k-1})_j;\boldsymbol{\alpha}_j^k,\sigma_k)=\nabla_j A_j^k(\boldsymbol{x}_j^k,\boldsymbol{y}_j^k|(\boldsymbol{x}^k,\boldsymbol{y}^k)_j;\boldsymbol{\alpha}_j^k,\sigma_k)=0$$

holds.

From the algorithm, if $\|\boldsymbol{g}(\boldsymbol{x}^k,\boldsymbol{y}^k)\|<\varepsilon$ holds, we find an approximate global optimal solution to (CNO). Under some conditions, it is proved that the algorithm can converge to a KKT point for $\varepsilon=0$. Let

$$S(\pi,g)=\{(\boldsymbol{x},\boldsymbol{y})\mid\pi\geq g(\boldsymbol{x},\boldsymbol{y})\},$$

which is called a level set. If $S(\pi,g)$ is bounded for any given $\pi>0$, then $S(\pi,g)$ is also bounded.

Theorem 4.5. *Let $\varepsilon = 0$ and f be a decomposable CN function. Suppose that a sequence of $\{(\boldsymbol{x}^k, \boldsymbol{y}^k) := ((\boldsymbol{x}_1^k, \boldsymbol{y}_1^k), (\boldsymbol{x}_2^k, \boldsymbol{y}_2^k), \cdots, (\boldsymbol{x}_p^k, \boldsymbol{y}_p^k))\}$, $k = 1, 2, \cdots$, is obtained by Algorithm 1. Let the sequence of $\{H_k(\boldsymbol{x}^k, \boldsymbol{y}^k, \sigma_k)\}$, $k = 1, 2, \cdots$, be bounded and the level set $S(\pi, g)$ be bounded, where*

$$H_k(\boldsymbol{x}^k, \boldsymbol{y}^k, \rho_k) = g(\boldsymbol{x}^k, \boldsymbol{y}^k) + \sigma_k \sum_{i=1}^{r} g_i(\boldsymbol{x}^k, \boldsymbol{y}^k)^2.$$

(i) If the algorithm stops at a finite number of step k, then $\boldsymbol{x}^k$ is a global optimal solution to (CNO).
(ii) If $\{(\boldsymbol{x}^k, \boldsymbol{y}^k)\}$ is an infinite sequence, then $\{(\boldsymbol{x}^k, \boldsymbol{y}^k)\}$ is bounded and any limit point $(\boldsymbol{x}^*, \boldsymbol{y}^*)$ of the sequence belongs to $X(\boldsymbol{g})$, and there exist $\eta > 0$ and λ_i, $i = 1, 2, \cdots, r$, such that

$$\eta \nabla g(\boldsymbol{x}^*, \boldsymbol{y}^*) + \sum_{i=1}^{r} \lambda_i \nabla g_i(\boldsymbol{x}^*, \boldsymbol{y}^*) = 0. \tag{4.12}$$

If $(\boldsymbol{x}^*, \boldsymbol{y}^*) \in X(f)$ and $\eta g(\boldsymbol{x}, \boldsymbol{y}) + \sum_{i=1}^{r} \lambda_i g_i(\boldsymbol{x}, \boldsymbol{y})$ is convex on $(\boldsymbol{x}, \boldsymbol{y})$ or $\lambda_i \geq 0$, $i = 1, 2, \cdots, r$, then $\boldsymbol{x}^*$ is an optimal solution to (CNO).

Proof. (i) From Step 2 and Step 3 of the algorithm, we have

$$\begin{aligned} \nabla_j A_j^k(\boldsymbol{x}_j^k, \boldsymbol{y}_j^k | (\boldsymbol{x}^{k-1}, \boldsymbol{y}^{k-1})_j; \boldsymbol{\alpha}_j^k, \sigma_k) &= \nabla_j A_j^k(\boldsymbol{x}_j^k, \boldsymbol{y}_j^k | (\boldsymbol{x}^k, \boldsymbol{y}^k)_j; \boldsymbol{\alpha}_j^k, \sigma_k) \\ &= \nabla_j g_j(\boldsymbol{x}^k, \boldsymbol{y}^k) + \boldsymbol{\alpha}_j^{k\top} \nabla_j \boldsymbol{g}_j(\boldsymbol{x}_j^k, \boldsymbol{y}_j{}^k) \\ &= 0, \quad j = 1, 2, \cdots, p, \end{aligned}$$

where $\boldsymbol{\alpha}^k = (\boldsymbol{\alpha}_1^k, \boldsymbol{\alpha}_2^k, \cdots, \boldsymbol{\alpha}_p^k)^\top$. Thus $\nabla g(\boldsymbol{x}^k, \boldsymbol{y}^k) + \boldsymbol{\alpha}^{k\top} \nabla \boldsymbol{g}(\boldsymbol{x}^k, \boldsymbol{y}^k) = 0$. By Theorem 4.1, $\boldsymbol{x}^k$ is a global optimal solution to (CNO).

(ii) As $k \to +\infty$, since $\{H_k(\boldsymbol{x}^k, \boldsymbol{y}^k, \rho_k)\}$ is bounded, there must be some $\pi > 0$ such that $\pi > H_k(\boldsymbol{x}^k, \boldsymbol{y}^k, \sigma_k) \geq g(\boldsymbol{x}^k, \boldsymbol{y}^k)$. $\{(\boldsymbol{x}^k, \boldsymbol{y}^k)\}$ is bounded because the level set $S(\pi, f)$ is bounded. Without loss of generality, suppose $(\boldsymbol{x}^k, \boldsymbol{y}^k) \to (\boldsymbol{x}^*, \boldsymbol{y}^*)$, i.e., $(\boldsymbol{x}_j^k, \boldsymbol{y}_j^k) \to (\boldsymbol{x}_j^*, \boldsymbol{y}_j^*) (j = 1, 2, \cdots, p)$. Since g is continuous, one has that $S(\pi, g)$ is closed. So, $g(\boldsymbol{x}^k, \boldsymbol{y}^k)$ is bounded and there is a $\sigma' > 0$ such that $g(\boldsymbol{x}^k, \boldsymbol{y}^k) > -\sigma'$. From the above inequality, we have that

$$\sum_{i=1}^{r} g_i(\boldsymbol{x}^k, \boldsymbol{y}^k)^2 \leq \frac{1}{\sigma_k}(\pi - g(\boldsymbol{x}^k, \boldsymbol{y}^k)) < \frac{\pi + \sigma'}{\sigma_k}.$$

Thus $\sum_{i=1}^{r} (g_i(\boldsymbol{x}^k, \boldsymbol{y}^k))^2 \to 0$ as $\sigma_k \to +\infty$. Hence, $(\boldsymbol{x}^*, \boldsymbol{y}^*) \in X(\boldsymbol{g})$. Note that there is an infinite sequence $\{(\boldsymbol{x}^k, \boldsymbol{y}^k, \boldsymbol{\alpha}^k, \rho_k)\}$ such that $\nabla_j A_j^k(\boldsymbol{x}_j^k, \boldsymbol{y}_j^k | (\boldsymbol{x}^{k-1}, \boldsymbol{y}^{k-1})_j; \boldsymbol{\alpha}_j^k, \sigma_k) = 0$, $j = 1, 2, \cdots, p$. For all $j = 1, 2, \cdots, p$, we have

$$\nabla_j g(\boldsymbol{x}_j^k, \boldsymbol{y}_j^k | (\boldsymbol{x}^{k-1}, \boldsymbol{y}^{k-1})_j) + \sum_{i=1}^{r_j} \alpha_{ji}^{k+1} \nabla_j g_{ji}(\boldsymbol{x}_j^k, \boldsymbol{y}_j^k) = 0, \tag{4.13}$$

where $\boldsymbol{\alpha}_j^k = (\alpha_{j1}^k, \alpha_{j2}^k, \cdots, \alpha_{jr_j}^k)$, $(\boldsymbol{x}_j^k, \boldsymbol{y}_j^k | (\boldsymbol{x}^{k-1}, \boldsymbol{y}^{k-1})_j) = ((\boldsymbol{x}_1^k, \boldsymbol{y}_1^k), \cdots, (\boldsymbol{x}_{j-1}^k, \boldsymbol{y}_{j-1}^k), (\boldsymbol{x}_j^k, \boldsymbol{y}_j^k), (\boldsymbol{x}_{j+1}^{k-1}, \boldsymbol{y}_{j+1}^{k-1}), \cdots, (\boldsymbol{x}_p^{k-1}, \boldsymbol{y}_p^{k-1}))$ and $\alpha_{ji}^{k+1} = \alpha_{ji}^k + \sigma_k g_{ji}(\boldsymbol{x}_{ji}^k, \boldsymbol{y}_{ji}^k)$ $(i = 1, 2, \cdots, r_j)$. For all $j = 1, 2, \cdots, p$, let $\gamma_j^k = 1 + \sum_{i=1}^{r_j} (\max\{\alpha_{ji}^{k+1}, 0\} + \max\{-\alpha_{ji}^{k+1}, 0\}) > 0$. Let $\eta_j^k = \frac{1}{\gamma_j^k} > 0$, $\mu_{ji}^k = \frac{\max\{\alpha_{ji}^{k+1}, 0\}}{\gamma_j^k} \geq$

$0, i = 1,2,\cdots,r_j$, and $\nu_{ji}^k = \frac{\max\{-\alpha_{ji}^{k+1},0\}}{\gamma_j^k} \geq 0, i = 1,2,\cdots,r_j$. Then,

$$\eta_j^k + \sum_{i=1}^{r_j}(\mu_{ji}^k + \nu_{ji}^k) = 1. \tag{4.14}$$

Clearly, for all $j = 1,2,\cdots,p$, as $k \to \infty$, we have $\eta_j^k \to \eta_j > 0, \mu_{ji}^k \to \mu_{ji}, \nu_{ji}^k \to \nu_{ji}, \forall i = 1,2,\cdots,r_j$. By (4.13) and (4.14), we have

$$\eta \nabla g(\boldsymbol{x}^*,\boldsymbol{y}^*) + \sum_{i=1}^{r}(\mu_i - \nu_i)\nabla g_i(\boldsymbol{x}^*,\boldsymbol{y}^*) = 0. \tag{4.15}$$

Let $\lambda^k = \mu_i^k - \nu_i^k \to \lambda$ as $k \to +\infty$, we conclude from (4.15) that (4.12). □

Finally, four examples are given to demonstrate that our algorithm can solve an approximate optimal solution to (CNO). All codes are written with Matlab2016a and numerical experiments are carried out with Thinkpad S3.

Example 4.2. Consider the optimization problem which is the same as Example 3.1. $(x_1^*,x_2^*) = (0,0)$ is an optimal solution to (Ex4.2) and $f(0,0) = 1$ for $\lambda > 2$. The CN function $f(x)$ is decomposable. By Example 3.1 and our algorithm, two subproblems are solved:

$$\begin{aligned}
(\text{Ex4.2})_1 \quad \min \quad & g((x_1,y_1,y_2,y_3),(x_2,y_4,y_5,y_6)) = (x_1+x_2-1)^2 + \lambda(y_2^2+y_5^2) \\
s.t. \quad & \boldsymbol{g}_1(x_1,\boldsymbol{y}_1) = (y_1^4 - y_3, x_1^2 - y_3, y_2^2 - y_1) = 0.
\end{aligned}$$

$$\begin{aligned}
(\text{Ex4.2})_2 \quad \min \quad & g((x_1,y_1,y_2,y_3),(x_2,y_4,y_5,y_6)) = (x_1+x_2-1)^2 + \lambda(y_2^2+y_5^2) \\
s.t. \quad & \boldsymbol{g}_2(x_2,\boldsymbol{y}_2) = (y_4^4 - y_6, x_2^2 - y_6, y_5^2 - y_4)) = 0.
\end{aligned}$$

Now, stating parameters $\lambda = 20000, \varepsilon = 10^{-4}, \theta = 0.1, \sigma_1 = 1000, N = 1000, \alpha_1^1 = (2,2,2)^\top$, $\alpha_1^2 = (2,2,2)^\top$, and $((x_1^0,y_1^0,y_2^0,y_3^0),(x_2^0,y_4^0,y_5^0,y_6^0)) = ((2,2,2,2),(2,2,2,2))$. At iteration 3, an approximate optimal solution is obtained $((x_1^k,y_1^k,y_2^k,y_3^k), (x_2^k,y_4^k,y_5^k,y_6^k)) = ((0.0009, 0.0000, 0.0000, 0.0000), (0.0006, 0.0000, -0.0000, 0.0000))$. Furthermore, when random values of starting points are chosen from $[-500,500]$, the same approximate optimal solution is obtained at iteration 6.

Example 4.3. Consider the special sparse optimization problem in [8]:

$$\begin{aligned}
(\text{Ex4.3}) \quad \min \quad & f_n(\boldsymbol{x}) = (\sum_{i=1}^{n} ix_i - 2n)^2 + \lambda \sum_{i=1}^{n} |x_i|_0 \\
s.t. \quad & \boldsymbol{x} \in R^n.
\end{aligned}$$

For $(x_i,y_i,y_{i+n}), i = 1,2,\cdots,n$, a decomposable PCN form of $f_n(\boldsymbol{x})$ at the maximum decomposition number $p = n$ is defined by

$$\begin{aligned}
&[(\sum_{i=1}^{n} ix_i - 2n)^2 + \lambda \sum_{i=1}^{n} y_i^2 : ((x_i+y_i-1)^2 - y_{i+n}, \\
&x_i^2 + (y_i-1)^2 - y_{i+n}, y_i^2 - y_i) = 0, i = 1,2,\cdots,n].
\end{aligned}$$

Let $n = ep$, $\boldsymbol{w}_j = (x_{ej+1}, x_{ej+2}, \cdots, x_{ej+e}, y_{ej+1}, y_{ej+2}, \cdots, y_{ej+e}, y_{ej+1+n}, y_{ej+2+n}, \cdots, y_{ej+e+n})$, $j = 0,1,2,\cdots,p-1$. So, for $j = 0,1,2,\cdots,p-1$, p subproblems are solved by:

$$\begin{aligned}
(\text{Ex4.3})_j \quad \min \quad & g(\boldsymbol{w}_0, \boldsymbol{w}_1, \cdots, \boldsymbol{w}_{p-1}) = (\sum_{j=0}^{p-1}\sum_{t=1}^{e}(ej+t)x_{ej+t} - 2n)^2 + \lambda \sum_{t=1}^{e} y_{ej+t}^2 \\
s.t. \quad & \boldsymbol{g}_j(\boldsymbol{w}_j) = ((x_{ej+t} + y_{ej+t} - 1)^2 - y_{ej+t+n}, \\
& x_{ej+t}^2 + (y_{ej+t} - 1)^2 - y_{ej+t+n}, y_{ej+t}^2 - y_{ej+t}) = 0,\ t = 1,2,\cdots,e.
\end{aligned}$$

For $j = 0,1,2,\cdots,p-1$, the augmented Lagrange penalty optimization of $(\text{Ex4.3})_j$ is defined by

$$\begin{aligned}
(\text{Ex4.3})_j \quad \min \quad & A_j(\boldsymbol{w}_j, \boldsymbol{\alpha}_j, \sigma) = (\sum_{j=0}^{p-1}\sum_{t=1}^{e}(ej+t)x_{ej+t} - 2n)^2 + \lambda \sum_{t=1}^{e} y_{ej+t}^2 + \\
& \sum_{t=1}^{e}[\alpha_{1t}((x_{ej+t} + y_{ej+t} - 1)^2 - y_{ej+t+n}) + \alpha_{2t}(x_{ej+t}^2 + (y_{ej+t} - 1)^2 \\
& - y_{ej+t+n}) + \alpha_3(y_{ej+t}^2 - y_{ej+t}) + \sigma(((x_{ej+t} + y_{ej+t} - 1)^2 - y_{ej+t+n})^2 \\
& + (x_{ej+t}^2 + (y_{ej+t} - 1)^2 - y_{ej+t+n})^2 + \alpha_3(y_{ej+t}^2 - y_{ej+t})^2)] \\
s.t. \quad & (x_{ej+t}, y_{ej+t}, y_{ej+t+n}) \in R^3,\ t = 1,2,\cdots,e.
\end{aligned}$$

By Algorithm 1, let the starting parameters $\varepsilon = 10^{-4}, \sigma_1 = 5, N = 10, \boldsymbol{\alpha}_j = (0,0,\cdots,0)$, and $\boldsymbol{w}_j^0 = (0,0,0,\cdots,0)$ be taken. When $e = 5, p = 1,2,6,10,20,100,200$, $\lambda = 1,10,100,500,1000$, the value of 0-norm $\|\boldsymbol{x}^k\|_0$ is obtained by Algorithm 1 as shown in Table 1. Numerical results demonstrate that an approximate sparse optimal solution is obtained. When λ is larger, the value of 0-norm $\|\boldsymbol{x}^k\|_0$ is smaller at the approximate sparse optimal solution.

TABLE 1. Value of $\|\boldsymbol{x}^k\|_0$ obtained by Algorithm 1 when $e = 5$.

$n = 5p$	$\lambda = 1$	$\lambda = 10$	$\lambda = 100$	$\lambda = 500$	$\lambda = 1000$
5	1	1	0	0	0
10	2	2	0	0	0
30	15	10	10	3	1
50	22	17	18	16	10
100	54	37	39	42	42
500	325	192	208	227	208
1000	611	382	427	471	421

TABLE 2. Numerical results obtained by Algorithm 1 when $e = 5$ and $n = 100$.

λ	10	100	1000	2000	3000	5000	8000	10000	15000	20000	21000
$\|\boldsymbol{x}^k\|_0$	37	39	42	40	31	11	7	5	2	1	0

Numerical experiments demonstrate if problem (CNP) is not decomposable, we cannot obtain an approximate sparse optimal solution by Algorithm 1 for $p = 1$, $\lambda \in [1,10]$ and $n > 15$. As the scale of the problem increases, the approximate spear solution may be obtained by using the decomposition algorithm (CNP). When λ is greater, the value of 0-norm $\|\boldsymbol{x}^k\|_0$ is smaller as shown in Table 2 when $n = e \times p = 5 \times 20 = 100$.

Example 4.4. Consider the nonconvex optimization problem ([2, Problem 5] and [3, Problem 73]):

$$\begin{aligned}
\text{(Ex4.4)} \quad \min \quad & f_n(\boldsymbol{x}) = n \max\{|x_i| : i = 1,2,\cdots,n\} - \sum_{i=1}^{n} |x_i| \\
s.t. \quad & \boldsymbol{x} \in R^n.
\end{aligned}$$

An optimal solution to (Ex4.4) is $\boldsymbol{x}^* = (\pm\alpha, \pm\alpha, \cdots, \pm\alpha)^\top$ with $f(\boldsymbol{x}^*) = 0$ for $\alpha \in R$ [3]. Let $\boldsymbol{x} \in R^n, \boldsymbol{y} \in R^{2n+1}$. A convertible nonconvex form of f is defined by

$$[ny_{2n+1} - \sum_{i=1}^{n} y_i : y_i^2 - y_{i+n}, x_i^2 - y_{i+n},\ i = 1,2,\cdots,n],$$

where $\boldsymbol{x} \in S_1 = R^n, \boldsymbol{y} \in S_2 = \{\boldsymbol{y} \mid -y_i \le 0, y_i - y_{2n+1} \le 0,\ i = 1,2,\cdots,n\}$. Let $n = ep$, $\boldsymbol{w}_0 = (x_1, x_2, \cdots, x_e, y_1, y_2, \cdots, y_e, y_{1+n}, y_{2+n}, \cdots, y_{e+n}, y_{2n+1})$, $\boldsymbol{w}_{j-1} = (x_{ej+1}, x_{ej+2}, \cdots, x_{ej+e}, y_{ej+1}, y_{ej+2}, \cdots, y_{ej+e}, y_{ej+1+n}, y_{ej+2+n}, \cdots, y_{ej+e+n})$, $j = 1,2,\cdots,p-1$. So, for $j = 0,1,2,\cdots,p-1$, p subproblems are solved by:

$$\begin{aligned}
\text{(Ex4.4)}_j \quad \min \quad & g(\boldsymbol{w}_0, \boldsymbol{w}_1, \cdots, \boldsymbol{w}_{p-1}) = ny_{2n+1} - \sum_{t=1}^{e} y_{et+j} \\
s.t. \quad & \boldsymbol{g}_j(\boldsymbol{w}_j) = (y_{ej+t}^2 - y_{ej+t+n}, x_{ej+t}^2 - y_{2n+1}) = 0,\ t = 1,2,\cdots,e, \\
& -y_{ej+t} \le 0, y_{ej+t}^2 - y_{ej+t+n} \le 0,\ t = 1,2,\cdots,e.
\end{aligned}$$

For $j = 0,1,2,\cdots,p-1$, the augmented Lagrange penalty optimization of (Ex4.4)_j is defined by

$$\begin{aligned}
\text{(Ex4.4)}_j \quad \min \ A_j(\boldsymbol{w}_j, \boldsymbol{\alpha}_j, \sigma) = \quad & ny_{2n+1} - \sum_{t=1}^{e} y_{et+j} + \sum_{t=1}^{e} [\alpha_{1t}(y_{ej+t}^2 - y_{ej+t+n}) \\
& + \alpha_{2t}(x_{ej+t}^2 - y_{2n+1}) + \sigma((y_{ej+t}^2 - y_{ej+t+n})^2 \\
& + (x_{ej+t}^2 - y_{2n+1})^2 + \max\{y_{ej+t}^2 - y_{ej+t+n}, 0\}^2)] \\
s.t. \quad & -y_{ej+t} \le 0,\ t = 1,2,\cdots,e.
\end{aligned}$$

By Algorithm 1, let the starting parameters $\varepsilon = 10^{-4}, \sigma_1 = 5, N = 10, \boldsymbol{\alpha}_j = (0,0,\cdots,0)$, and $\boldsymbol{w}_j^0 = (1,2,3,\cdots,3e+1)$ be taken. So, numerical results in Table 3 and Table 4 are obtained for $e = 3,5$. Through Algorithm 1, an approximate solution can be obtained at iteration steps 2-4 . The numerical results demonstrate that the running time is a multiple of the decomposition scale p. Through numerical experiments, it is appropriate to take e from 2 to 6 in this example. When e exceeds 6, the larger the n, the worse the approximate solution. However, the smaller the e, the longer the execution time. The scale of (Ex4.4) here is much larger than those of (Ex4.4) in [2]. If fact, the example demonstrates that the solution $\boldsymbol{x}^*$ to (Ex4.4) is composed of $\boldsymbol{x}^k$ repeated e times. For example, because $\boldsymbol{x}^k = (3.1448, 3.1448, 3.1448, 3.1448, 3.1448)$ is an optimal solution to (Ex4.4) at $n = 5$ by Algorithm 1. Then, $(3.1448, 3.1448, 3.1448, 3.1448, 3.1448)$ extends to $\boldsymbol{x}^k = (3.1448, 3.1448, 3.1448, 3.1448, 3.1448, 3.1448, 3.1448, 3.1448, 3.1448, 3.1448)$ which is an optimal solution to (Ex4.4) at $n = 10$. When n is very large, it is easy to obtain its optimal solution.

TABLE 3. Numerical results obtained by Algorithm 1 when $e = 5$.

n	k	$(x_1^k, x_2^k, \cdots, x_n^k)$	Running time
5	2	(3.1448,3.1448,3.1448, 3.1448,3.1448)	2.9155s
10	2	$(3.1448, 3.1448, 3.1448, \cdots, 3.1448)$	4.3817s
50	2	$(3.1448, 3.1448, 3.1448, \cdots, 3.1448)$	18.1406s
250	3	$(3.3481, 3.3481, 3.3481, \cdots, 3.3481)$	152.7852s
500	4	$(3.3484, 3.3484, 3.3484, \cdots, -3.3484)$	401.494226s
1000	4	$(3.3484, 3.3484, 3.3484, \cdots, 3.3484)$	855.879704s

TABLE 4. Numerical results obtained by Algorithm 1 when $e = 3$.

n	k	$(x_1^k, x_2^k, \cdots, x_n^k)$	Running time
6	2	$(3.4322, 3.4322, 3.4322, \cdots, 3.4322)$	3.1981s
30	2	$(3.4322, 3.4322, 3.4322, \cdots, 3.4322)$	12.5217s
90	2	$(3.4322, 3.4322, 3.4322, \cdots, 3.4322)$	45.0358s
300	3	$(3.4262, 3.4262, 3.4262, \cdots, 3.4262)$	215.3782s
600	3	$(3.4262, 3.4262, 3.4262, \cdots, 3.4262)$	457.0393s
1500	3	$(3.4262, 3.4262, 3.4262, \cdots, 3.4262)$	1102.5697s

Example 4.5. Consider the nonconvex optimization problem (Problem 64 on Page 280 in [3]):

$$\text{(Ex4.5)} \quad \min \quad f_n(\boldsymbol{x}) = \sum_{i=1}^{n-1} (-x_i + 2(x_i^2 + x_{i+1}^2 - 1) + 1.75|x_i^2 + x_{i+1}^2 - 1|)$$
$$s.t. \quad \boldsymbol{x} \in R^n.$$

Let $\boldsymbol{x} \in R^n, \boldsymbol{y} \in R^{2n+1}$. A convertible nonconvex form of f is defined by

$$[\sum_{i=1}^{n-1} (-x_i + 2y_i + 1.75y_{i+n}) \quad : \quad x_i^2 + x_{i+1}^2 - 1 - y_i = 0, y_{i+n}^2 - y_{i+2n} = 0,$$
$$y_i^2 - y_{i+2n} = 0 \; i = 1, 2, \cdots, n-1],$$

where $\boldsymbol{x} \in S_1 = R^n, \boldsymbol{y} \in S_2 = \{\boldsymbol{y} \mid y_i \geq -1, y_{i+n}, y_{i+2n} \geq 0, \; i = 1, 2, \cdots, n\}$. Let $n = ep$, $\boldsymbol{w}_j = (x_{ej+1}, x_{ej+2}, \cdots, x_{ej+e}, y_{ej+1}, y_{ej+2}, \cdots, y_{ej+e}, y_{ej+1+n}, y_{ej+2+n}, \cdots, y_{ej+e+n}, y_{ej+1+2n}, y_{ej+2+2n}, \cdots, y_{ej+e+2n})$, $j = 0, 1, 2, \cdots, p-1$. So, for $j = 0, 1, 2, \cdots, p-1$, p subproblems are solved by:

$$\text{(Ex4.5)}_j \quad \min \quad g(\boldsymbol{w}_0, \boldsymbol{w}_1, \cdots, \boldsymbol{w}_{p-1}) = \sum_{t=1}^{e} (-x_{ej+t} + 2y_{ej+t} + 1.75y_{ej+t+n})$$
$$s.t. \quad \boldsymbol{g}_j(\boldsymbol{w}_j) = (x_{ej+t}^2 + x_{ej+t+1}^2 - 1 - y_{ej+t}, y_{ej+t+n}^2 - y_{ej+t+2n},$$
$$y_{ej+t}^2 - y_{ej+t+2n}) = 0, \; t = 1, 2, \cdots, e,$$
$$-y_{ej+t} \leq 1, -y_{ej+t+n}, -y_{ej+t+2n} \leq 0, \; t = 1, 2, \cdots, e.$$

Since $\{\boldsymbol{w}_{j-1}\}\cap\{\boldsymbol{w}_j\}=\{x_{ej+1}\}$, $f_n(\boldsymbol{x})$ is not decomposable. If $\boldsymbol{w}^*_{j-1}(j>0)$ is an optimal solution to $(\text{EX4.5})_{j-1}$, a term "$\sigma(x^*_{ej+1}-x_{ej+1})^2$" is added to $(\text{EX4.5})_j$, where, for $j=0,1,2,\cdots,p-1$, the augmented Lagrange penalty optimization of $(\text{EX4.5})_j$ is defined by

$$
\begin{aligned}
(\text{Ex4.5})_j \quad \min \ A_j(\boldsymbol{w}_j,\boldsymbol{\alpha}_j,\sigma) = \ & \sum_{t=1}^{e}(-x_{ej+t}+2y_{ej+t}+1.75y_{ej+t+n})+ \\
& \sum_{t=1}^{e}[\alpha_{1t}(x^2_{ej+t}+x^2_{ej+t+1}-1-y_{ej+t})+ \\
& \alpha_{2t}(y^2_{ej+t+n}-y_{ej+t+2n})+\alpha_{3t}(y^2_{ej+t}-y_{ej+t+2n})) \\
& +\sigma((x^2_{ej+t}+x^2_{ej+t+1}-1-y_{ej+t})^2+ \\
& (y^2_{ej+t+n}-y_{ej+t+2n})^2+(y^2_{ej+t}-y_{ej+t+2n})^2)] \\
& +\sigma(x^*_{ej+1}-x_{ej+1})^2 \\
s.t. \qquad\qquad & -y_{ej+t}\le 1, -y_{ej+t+n}, -y_{ej+t+2n}\le 0,\ t=1,2,\cdots,e.
\end{aligned}
$$

Especially, when $j=0$, "$\sigma(x^*_1-x_1)^2$" is deleted from the above $(\text{EX4.5})_j$. When $n=50,200$, and 1000, their approximate objective values $f(\boldsymbol{x}^k)$ are $-34.795,-140.86$, and -706.55 respectively at starting point $x^1=(1,1,\cdots,1)^\top$ in [3]. By Algorithm 1, let the starting parameters $\varepsilon=10^{-4},\sigma_1=5,N=100,\boldsymbol{\alpha}_j=(0,0,\cdots,0)^\top$, and $\boldsymbol{w}^0_j=(1,1,1,\cdots,1)^\top$ be taken. So, at iteration step 1, numerical results are obtained by Algorithm 1 as shown in Tables 5, 6, and 7 respectively when $n=50,200$ and 1000. In Tables 5, 6, and 7, the best objective value of (Ex4.5) is obtained when $n=2p$. It is worth noting that the solutions are effective in Tables 5-7 because (Ex4.5) is an unconstrained optimization problem.

TABLE 5. Numerical results obtained by Algorithm 1 when $n=50$.

e	p	$f_n(\boldsymbol{x}^k)$	$\\|\boldsymbol{g}(\boldsymbol{x}^k,\boldsymbol{y}^k)\\|$	Running time(s)
50	1	-38.2547	0.0593	4.2525
25	2	-36.8531	0.0472	5.6183
10	5	-39.4672	0.0001	10.8266
5	10	-35.8447	0.0000	24.6390
2	25	-40.8154	0.0000	47.2351

TABLE 6. Numerical results obtained by Algorithm 1 when $n=200$.

e	p	$f_n(\boldsymbol{x}^k)$	$\\|\boldsymbol{g}(\boldsymbol{x}^k,\boldsymbol{y}^k)\\|$	Running time(s)
20	10	-159.4185	0.04810	25.5085
10	20	-159.4124	0.00101	37.1376
5	40	-145.9025	0.00002	82.4633
4	50	-156.7485	0.00000	93.6619
2	100	-166.0306	0.00000	151.7478

By Algorithm 1, a solution $\tilde{\boldsymbol{x}}=(0.6897,0.4153,0.4965,0.5386,0.4707)$ with its objective value $f_5(\tilde{\boldsymbol{x}})=-3.0766$ is obtained at $n=5$ and $p=1$. And another solution $\hat{\boldsymbol{x}}=(0.6772,0.49930)$

TABLE 7. Numerical results obtained by Algorithm 1 when $n = 1000$.

e	p	$f_n(\boldsymbol{x}^k)$	$\Vert\boldsymbol{g}(\boldsymbol{x}^k,\boldsymbol{y}^k)\Vert$	Running time(s)
10	100	-805.0714	0.00108	271.2822
5	200	-726.9435	0.00002	566.6910
4	250	-783.3616	0.00001	589.5969
2	500	-833.6333	0.00002	717.9175

with its objective value $f_2(\hat{\boldsymbol{x}}) = -0.8749$ is obtained at $n = 2$ and $p = 1$. In Table 8, the solution $\boldsymbol{x}^k$ in the first three columns is composed of $\tilde{\boldsymbol{x}}$ repeated p times, such as $\boldsymbol{x}^k = (\tilde{\boldsymbol{x}},\tilde{\boldsymbol{x}},\tilde{\boldsymbol{x}},\tilde{\boldsymbol{x}},\tilde{\boldsymbol{x}},\tilde{\boldsymbol{x}},\tilde{\boldsymbol{x}},\tilde{\boldsymbol{x}},\tilde{\boldsymbol{x}},\tilde{\boldsymbol{x}})$ at $e = 5$ and $p = 10$ in line 1, where $f_n(\boldsymbol{x}^k)$ is the objective value of $\boldsymbol{x}^k$. In Table 8, the solution $\boldsymbol{x}^k$ in columns 4 to 6 is composed of $\hat{\boldsymbol{x}}$ repeated p times. In Table 8, the values in the last six columns are obtained by Algorithm 1 (See Tables 5, 6, and 7). Numerical results show that for large-scale unconstrained optimization problems, a better solution can be obtained directly by Algorithm 1 by solving small-scale subproblems of it when the structure of all the subproblems are similar, i.e. the constraint structure of all the equality subproblems is the same as those in Examples 4.3, 4.4, and 4.5. This decomposable method is effective in examples 4.4 and 4.5.

TABLE 8. Numerical results obtained by Algorithm 1.

e	p	$f_n(\boldsymbol{x}^k)$	e	p	$f_n(\boldsymbol{x}^k)$	e	p	$f_n(\boldsymbol{x}^k)$	e	p	$f_n(\boldsymbol{x}^k)$
5	10	-37.8238	2	25	-41.1109	5	10	-35.8447	2	25	-40.8154
5	40	-153.6478	2	100	-166.8484	5	40	-145.9025	2	100	-166.0306
5	200	-771.3758	2	500	-837.4484	5	200	-726.9435	2	500	-833.6333
5	1000	-3860.0158	2	2500	-4190.4484	5	1000	-3647.9851	2	2500	-4171.0713
5	2000	-7720.8158	2	5000	-8381.6984	5	2000	-7295.0801	2	5000	-8343.5399

Algorithm 1 demonstrates that when these parameters $\varepsilon > 0, \sigma_1 > 0, N > 1, (\boldsymbol{x}_j^0, \boldsymbol{y}_j^0)$ $(j = 1,2,\cdots,p)$, $\boldsymbol{\alpha}_j^1 (j = 1,2,\cdots,p)$ are properly selected. We obtain an approximate optimal solution to an unconstrained optimization problem with CN function, the above examples demonstrate that the scale problem of (CNP) can be avoided and reduced by solving the subproblems in Algorithm 1. If (CNP) is a small-scale problem for $p = 1$ or the number of variables in (CNP) is very small, Algorithm 1 may obtain an approximate solution to (CNP).

5. CONCLUSION

This paper discusses three difficulties relating to unconstrained nonconvex or nonsmooth optimization problems. (1) An unconstrained, nonconvex, and nonsmooth optimization problem is transformed into a constrained optimization problem, where its objective function and constrained functions are convex and smooth. A new concept - weak uniform or decomposable CN optimization - is proposed, which covers many nonconvex nonsmooth functions, even discontinuous nonconvex functions. (2) The optimality conditions of the global optimal solution to unconstrained nonconvex or nonsmooth problems are obtained. The sufficient conditions of the global optimal solution are proved for the weak uniform or decomposable CN optimization problems. (3) A decomposable algorithm for unconstrained nonconvex or nonsmooth optimization problem is proposed based on the augmented Lagrange penalty function of (CNP). The results of numerical examples demonstrate that Algorithm 1 may obtain an approximate global optimal solution after properly selecting the initial parameters. Another advantage of the algorithm is that it does not need to use subgradient and smoothing techniques such that Matlab, the optimization software, is directly usable, which makes this method easy for engineers to use. This paper provides a new method for solving nonconvex unconstrained optimization problems, which shows its potential importance in many application fields. There are at least three directions worthy of further study in terms of weak uniform CN optimization problems (CNP):

(1) decomposable Newton algorithms or decomposable SQP algorithms for (CNP),

(2) Lagrangian multiplier alternating algorithms for (CNP),

(3) some special structures of (CNP), for example when f is a weak uniform CN function with $f = [g, g_1, g_2, \cdots, g_r]$, where $g, g_1, g_2, \cdots, g_r$ are quadratic functions.

Acknowledgments
This work was supported by the National Natural Science Foundation of China (No. 11871434). The authors would like to express their gratitude to anonymous referees' detailed comments and remarks that helped us improve the presentation of this paper considerably.

REFERENCES

[1] F.A. Al-Khayyaltt, J.E. Falk, jointly constrained biconvex programming, Math. Oper. Res. 8 (1983), 273-286.
[2] A.M. Bagirov, A method for minimizing of quasidifferentiable functions, Optim. Meth. Softw. 17 (2002), 31-60.
[3] A.M. Bagirov, N. Karmitsa, N.M. Mäkelä, Introduction to Nonsmooth Optimization: Theory, Practice and Software. Springer, New York, 2014.
[4] X. Bao, A. Khajavirad, N.V. Sahinidis, M. Tawarmalani, Global optimization of nonconvex problems with multilinear intermediates, Math. Program. Comput. 7 (2015), 1-37.
[5] M.L. Bougeard, Morse theory for some lower-C^2 functions in finite dimension, Math. Program. 41 (1988), 141-159.
[6] G.A. Bunin, Extended reverse-convex programming: An approximate enumeration approach to global optimization, J. Global Optim. 65 (2016), 191-229.
[7] Chen, X.J., Ge, D.D., Wang, Z.Z., Ye Y.Y.: Complexity of unconstrained $L_2 - L_p$ minimization,Mathematical Programming, Ser. A 143:371-383(2014)
[8] X. Chen, F. Xu, Y. Ye, Lower bound theory of nonzero entries in solutions of L_2-L_p minimization, SIAM J. Sci. Comput. 32 (2010), 2832-2852.
[9] N.H. Chieu, G.M. Lee, B.S. Mordukhovich, T.T.A. Nghia, Coderivative characterizations of maximal monotonicity for set-valued mappings, J. Convex Anal. 23 (2016), 461-480.
[10] F.H. Clarke, Opimization and Nonsmooth Analysis. John Wiley & Sons, 1983.
[11] A. Daniilidis, J. Malick, Filling the gap between lower-C^1 and lower-C^2 functions, J. Convex Anal. 12 (2005), 315-329.
[12] M.N. Dao, Bundle method for nonconvex nonsmooth constrained optimization, J. Convex Anal. 22 (2015), 1061-1090.
[13] M. Frank, P. Wolfe, An algorithm for quadratic programming, Naval Res. Logisics, 3 (1956), 95-110.
[14] M. Grant, S. Boyd, Y. Ye, Disciplined convex programming, In: L. Liberti and N. Maculan (eds.), Global Optimization: From Theory to Implementation, Nonconvex Optimization and its Applications, pp. 155-210, Springer, 2006.
[15] L. Granvilliers Filtering domains of factorable functions using interval contractors, In: H.A.L. Thi, H.M. Le, T.P. Dinh (eds.) Springer Cham, 2020. https://doi.org/10.1007/978-3-030-21803-4.
[16] W.L. Hare, Functions and sets of smooth substructure: Relationships and examples, Comput. Optim. Appl. 33 (2006), 249-270.

[17] W.L. Hare, R.A. Poliquin, Prox-Regularity and stability of the proximal mapping, J. Convex Anal. 14 (2007), 589-606.

[18] W. Hare, C. Sagastizába, Computing proximal points of nonconvex functions, Math. Program. Ser. B 116 (2009) 221-258.

[19] W. Hare, C. Sagastizábal, A redistributed proximal bundle method for nonconvex optimization, SIAM J. Optim. 20 (2010), 2442-2473.

[20] W. Hare, C. Sagastizábal, M. Solodov, A proximal bundle method for nonsmooth nonconvex functions with inexact information, Comput. Optim. Appl. 63 (2016), 1-28.

[21] T. He, M. Tawarmalani, A new framework to relax composite functions in nonlinear programs, Math. Program. 190 (2021),427-466.

[22] R.H.F. Jackson, G.P. Mccormick, Second-order sensitivity analysis in factorable programming: Theory and applications, Math. Program. 41 (1988), 1-27.

[23] M. Jiang, Z. Meng, C. Dang, R. Shen, Unconstrained optimization of convertible nonconvex function, submitted to Appl. Anal. under review.

[24] W. Li, W. Bian,Projection neural network for a class of sparse regression problems with cardinality penalty, Neurocomting, 431 (2021), 188-200.

[25] A. Lundell, J. Westerlund, T. Westerlund, Some transformation techniques with applications in global optimization, J. Glob. Optim. 43 (2009), 391-405.

[26] G.P. McCormick, Computability of global solutions to factorable nonconvex programs: part I-convex underestimating problems, Math. Program. 10 (1976), 147-175.

[27] M. Mohri, A. Rostamizadeh, A. Talwalkar, Foundations of Machine Learning, The MIT Press Cambridge, Massachusetts, 2012.

[28] H.V. Ngai, J.P. Penot, Subdifferentiation of regularized functions, Set-Valued Var. Anal 24 (2016), 167-189.

[29] J.C. Nohra, N.V. Sahinidis, Global optimization of nonconvex problems with convex-transformable intermediates, J Glob. Optim. 72 (2018), 255-276.

[30] D. Noll, Cutting plane oracles to minimize non-smooth non-convex functions, Set-Valued Anal. 18 (2010), 531-568.

[31] R.T. Rockafellar, R.J.-B. Wets, Variational Analysis, Springer, Berlin, 1998.

[32] F. Serrano, Intersection Cuts for Factorable MINLP. In: Lodi, A., Nagarajan, V. (eds) Integer Programming and Combinatorial Optimization. IPCO 2019. Lecture Notes in Computer Science, vol 11480, Springer, Cham, 2019.

[33] H.D. Sherali, H. Wang, Global optimization of nonconvex factorable programming problems, Math. Program. 89 (2001), 459-478.

[34] J.E. Spingarn, Submonotone subdifferentials of Lipschitz functions, Trans. Amer. Math. Soc. 264 (1981), 77-89.

[35] M. Tawarmalani, N.V. Sahinidis, Global optimization of mixed-integer nonlinear programs: A theoretical and computational study, Math. Program. 99 (2004), 563-591.

[36] M. Tawarmalani, J.P. Richard, C. Xiong, Explicit convex and concave envelopes through polyhedral subdivisions, Math. Program. 138 (2013), 531-577.

[37] H. Wang, et al., Support vector machine classifier via $L_{0/1}$ soft-margin loss, IEEE Trans. Pattern Anal. Mach. Intelligence doi: 10.1109/TPAMI.2021.3092177.

[38] H. Zhang, B. Dong, A review on deep learning in medical image reconstruction, J. Oper. Res. Soc. China 8 (2020), 311-340.