Yokohama Publishers
ISSN 1349-8169 ONLINE JOURNAL

Pacific Journal of Optimization
Vol. 20, No. 3, pp. 429–440, 2024

A DISCRETIZATION METHOD FOR A CLASS OF COPOSITIVE PROGRAMMING PROBLEMS*

Chuanhao Guo†, Pengfei Ma and Kok Lay Teo†

Dedicated to Masao Fukushima on the occasion of his 75th birthday‡

Abstract: In this paper, we consider a class of copositive programming problems with a continuous objective function. First, we show that these copositive programming problems can be transformed equivalently into semi-infinite programming problems through the use of a key lemma. Then, according to the structure characteristics of the problems, a new discretization method is proposed to solve these transformed problems. Moreover, under some mild conditions, we show that the proposed new method will terminated in a finite number of iterations, giving rise to a feasible solution with the corresponding cost function value better than or equal to the predicted optimal cost function value of the original problem, or confirming that there does not admits a feasible solution for the original problem. Finally, two numerical examples are solved using the proposed method. The results obtained demonstrate the applicability of the proposed algorithm.

Key words: *discretization method; finite convergence; copositive programming; semi-infinite programming*

Mathematics Subject Classification: *90C26, 90C34, 90C59, 49M25*

---

1 Introduction

In this paper, we consider the following class of copositive programming problems

$$
\text{(CP)} \quad \begin{array}{ll} \min\limits_{x\in\mathbb{X}} & f(x) \\ \text{s.t.} & \sum\limits_{i=1}^{m} x_i Q_i + B = Z, \\ & Z \in \mathbb{C}_n, \end{array}
$$

*This work is supported by the Ministry of Higher Education (MoHE) Malaysia through the Fundamental Research Grant Scheme (FRGS/1/2021/STG06/SYUC/03/1), China National Scholarship Fund (Grant No. 202208330246), Science and Technology Innovation Special Fund of Guangzhou Science and Technology Bureau in 2020 (Grant No. 202002030119), and National Natural Science Foundation of China (Grant No. 11501350).

†Corresponding author

‡We are pleased to entrust this paper to Prof. Masao Fukushima. I first met Masao in 1999, when I was the Head of the Department of Applied Mathematics and Chair Professor of Applied Mathematics at the Hong Kong Polytechnic University. Masao is a very distinguished and respected scholar with a high international reputation. His work on optimization has inspired the careers of many researchers. On a personal level, we have organized many conferences over the years. He is a good friend. This note was written by Kok Lay Teo.

 DOI: https://doi.org/10.61208/pjo-2023-022

where the objective function $f(x) : \mathbb{X} \subset \mathbb{R}^m \to \mathbb{R}$ is continuous, $\mathbb{X}$ is a convex compact subset of $\mathbb{R}^m$, Q_i $(i = 1, 2, \ldots, m)$ and B belong to $\mathbb{S}^n$ which is the set of all $n \times n$ symmetric matrices, $\mathbb{C}_n$ is the cone of copositive matrices defined as

$$\mathbb{C}_n := \{Z \in \mathbb{S}^n : \ v^T Z v \geq 0, \ \forall v \in \mathbb{R}^n_+\}. \tag{1.1}$$

Copositive programming is a relatively new field in mathematical optimization, it can be seen as a generalization of semidefinite programming because the optimization is carried out over the cone of copositive matrices. It is shown in [3] that any quadratic programming with linear constraints can be formulated as a copositive programming. For further details, see, for example [12, 10, 5], and references cited therein.

Although copositive programming is a convex optimization problem, it is intractable under the standard definition given in [5]. That is because the problem of detecting a given matrix $Z \in \mathbb{C}_n$ is NP-complete [11], and hence copositive programming is not solvable in polynomial time. Therefore, the reformulation as a convex optimization problem does not alter the complexity of the original problem, even though several quadratic problems have been shown to have exact copositive reformulations. However, this new convex reformulation motivates the exploration of better bounds than those previously obtained. Moreover, it seems feasible to find new methods considering some of the approximations introduced in [9, 8, 6, 14, 7].

In [2], it is shown that a class of copositive programming problems can be equivalently expressed as a class of linear SIP problems considered in [13]. Subsequently, an approximation scheme is proposed in [1] to solve such linear copositive programming problems. This approximation scheme is inspired by the discretization method proposed in [15] for solving linear SIP problems. Note that the copositive programming problem considered in [1] involves only a linear objective function, and the proposed approximation scheme is based on the optimality conditions and the duality results of the linear SIP problems.

Inspired by the ideas reported in [1], we consider a class of copositive programming problems. They are collectively referred to as problem (CP). This problem can be seen as an extension of the problem considered in [1], since the objective function of problem (CP) only needs to be continuous. First, we show that problem (CP) can be equipollently expressed as a SIP problem through the use of a key lemma. Then, by an appropriate parameterization of the objective function, a new discretization method is proposed to solve this SIP problem. Moreover, under some mild assumptions, we show that this new method will terminate in a finite number of iterations, generating a feasible solution with the corresponding objective function value being better than or equal to the predicted objective function value of problem (CP), or confirming that there does not admit a feasible solution to problem (CP) with the predicted objective function value. In particular, if the predicted objective function value is equal to the optimal objective function value of problem (CP), then the method will output an exact optimal solution to problem (CP).

The rest of paper is organized as follows. In Section 2, the basic framework of discretization method for solving SIP is presented. Through the use of a key lemma, problem (CP) is reformulated equivalently as a SIP problem in Section 3. The new discretization method and an analysis showing termination for a finite number of iterations are presented in Section 4 and Section 5, respectively. In Section 6, two numerical examples are solved and the results obtained are presented. Finally, we conclude the paper in Section 7.

2 Preliminary

In this section, we review basic results for discretization methods for solving SIP.

First, it is well-known that SIP is an old mathematical programming problem that is well studied in the literature. See, for example, [16] and the references cited therein. The general form of SIP is

$$\text{(SIP)} \qquad \begin{array}{ll} \min\limits_{x\in\mathbb{X}} & f(x) \\ \text{s.t.} & g(x,s)\le 0, \quad s\in\Omega, \end{array}$$

where $f(x):\mathbb{R}^m\to\mathbb{R}$ and $g(x,s):\mathbb{R}^m\times\Omega\to\mathbb{R}$ are continuous functions, and Ω is a nonempty compact subset of $\mathbb{R}^p$. Let this problem be referred to as problem (SIP).

Clearly, for problem (SIP), there are infinitely many constraints. Thus, it cannot be solved in polynomial time. Therefore, there exist no polynomial time algorithms for solving problem (SIP). Discretization method is a popular numerical method. It approximates the set Ω by a finite subset $\Omega'\subset\Omega$ such that problem (SIP) is approximated as the following approximate problem, which is refereed to as problem (2.1).

$$\begin{array}{ll} \min\limits_{x\in\mathbb{X}} & f(x) \\ \text{s.t.} & g(x,s)\le 0, \quad s\in\Omega'. \end{array} \tag{2.1}$$

Since $\Omega'\subset\Omega$, it holds that $\mathrm{Opt(SIP)}\ge\mathrm{Opt}(2.1)$, i.e., the discretization method can produce a lower bound for problem (SIP). The approximate problem (2.1) can be solved either locally or globally to produce an optimal solution $\bar{x}$. To check whether $\bar{x}$ is a feasible point to problem (SIP) or not, it is required to solve the following auxiliary optimization problem

$$\begin{array}{ll} \max & g(\bar{x},s) \\ \text{s.t.} & s\in\Omega \end{array} \tag{2.2}$$

in each iteration. Clearly, if $\mathrm{Opt}(2.2)\le 0$, we can conclude that $\bar{x}$ is an optimal solution of problem (SIP). If $\mathrm{Opt}(2.2)>0$, the discretized set is required to be refined and updated. We then return to solve the corresponding problem (2.1) and problem (2.2).

This process is continued until optimal solution to problem (SIP) is obtained. See, for example, [4] and relevant references cited therein for more details on discretization methods for solving problem (SIP).

3 Problem Reformulation

To establish a SIP equivalent representation of problem (CP), a key lemma is given below.

Lemma 3.1. *Let $A\in\mathbb{S}^n$, and let $\|\cdot\|$ denote the usual 2-norm on $\mathbb{R}^n$. The following properties are equivalent:*

(i) *A is copositive;*

(ii) $y^TAy\ge 0,\ \forall y\in\mathbb{R}^n_+,\ \|y\|\le 1.$

Proof. The (i)⇒(ii) is obviously from the definition of copostive cone (1.1).

Now, we show the relationship (ii)⇒(i). Let $y\in\mathbb{R}^n_+$ with $\|y\|\le 1$. If $\|y\|=0$, then it follows that $y=0$ and $y^TAy=0$. If $\|y\|>0$, then let $\bar{y}:=\frac{y}{\|y\|}$. Clearly, $\|\bar{y}\|=1$ and $\bar{y}^TA\bar{y}=\frac{1}{\|y\|^2}y^TAy\ge 0$. Thus, it holds that (ii)⇒(i). □

By Lemma 3.1, for any $Z\in\mathbb{S}^n$

$$Z\in\mathbb{C}_n \Leftrightarrow v^TZv\ge 0, \forall v\in V:=\{v\in\mathbb{R}^n_+:\ \|v\|\le 1\}.$$

Thus, it follows from the definition of problem (CP) that

$$\begin{aligned} Z \in \mathbb{C}_n \quad &\Leftrightarrow \sum_{i=1}^{m} x_i Q_i + B \in \mathbb{C}_n \\ &\Leftrightarrow v^T(\sum_{i=1}^{m} x_i Q_i + B)v \geq 0,\ \forall v \in V \\ &\Leftrightarrow a_v^T x - b_v \leq 0,\ \forall v \in V, \end{aligned}$$

where $a_v := (-v^T Q_1 v, -v^T Q_2 v, \ldots, -v^T Q_m v)^T$ and $b_v := v^T B v$. On this basis, problem (CP) can be expressed equivalently as follows:

$$\text{(SIPcp)} \quad \begin{array}{ll} \min\limits_{x \in \mathbb{X}} & f(x) \\ \text{s.t.} & a_v^T x - b_v \leq 0,\ \forall v \in V, \end{array}$$

i.e.,

$$\text{(CP)} \quad \begin{array}{ll} \min\limits_{x \in \mathbb{X}} & f(x) \\ \text{s.t.} & \sum\limits_{i=1}^{m} x_i Q_i + B = Z, \\ & Z \in \mathbb{C}_n, \end{array} \quad \Longleftrightarrow \quad \text{(SIPcp)} \quad \begin{array}{ll} \min\limits_{x \in \mathbb{X}} & f(x) \\ \text{s.t.} & a_v^T x - b_v \leq 0,\ \forall v \in V. \end{array}$$

By virtue of the definition of problem (SIP), it is clear that problem (SIPcp) is a SIP problem.

Now, problem (CP) can be reformulated equivalently as a SIP problem, i.e., problem (SIPcp). Note that the two problems have the same objective function, but they have different constraints. If we could obtain an optimal solution x^* to problem (SIPcp), we can show that x^* is also an optimal solution to problem (CP) with the corresponding matrix $Z^* \in \mathbb{C}_n$.

4 Description of Algorithm

In this section, a new discretization method for solving problem (SIPcp) is presented, and some important properties are analysized. For convenience, denote x^* as an optimal solution to problem (SIPcp), and let $f^* := f(x^*)$ be the corresponding optimal objective function. We have the following lemma.

Lemma 4.1. *For a given objective function value f_0 of problem (SIPcp), the optimal objective function value of*

$$\min_{x \in \mathbb{X}} \max\{f(x) - f_0,\ \max\{a_v^T x - b_v,\ \forall v \in V\}\} \tag{4.1}$$

is less than or equal to 0 if and only if $f^* \leq f_0$.

Proof. The proof is divided into the following two steps.

(i). If Opt(4.1) ≤ 0, then there exists an $\hat{x} \in \mathbb{X}$ such that

$$\max\{f(\hat{x}) - f_0,\ \max\{a_v^T \hat{x} - b_v,\ \forall v \in V\}\} \leq 0,$$

i.e.,

$$f(\hat{x}) - f_0 \leq 0,\ \max\{a_v^T \hat{x} - b_v,\ \forall v \in V\} \leq 0,$$

the last inequality implies that $\hat{x}$ is a feasible solution of problem (SIPcp). Thus,

$$f^* \leq f(\hat{x}) \leq f_0.$$

(ii). Suppose that $f^* \leq f_0$, i.e., $f^* - f_0 \leq 0$. Since x^* is the optimal solution of problem (SIPcp), and that $\max\{a_v^T x^* - b_v,\ \forall v \in V\} \leq 0$, we obtain

$$\max\{f^* - f_0,\ \max\{a_v^T x^* - b_v,\ \forall v \in V\}\} \leq 0.$$

Therefore, it follows that

$$\begin{aligned}\min_{x \in \mathbb{X}} & \max\{f(x) - f_0,\ \max\{a_v^T x - b_v,\ \forall v \in V\}\} \\ & \leq \max\{f^* - f_0,\ \max\{a_v^T x^* - b_v,\ \forall v \in V\}\} \\ & \leq 0.\end{aligned}$$

□

The results of Lemma 4.1 show that the auxiliary problem (4.1) can provide a feasible solution to problem (SIPcp), and its optimal cost function value is better than or equal to the predicted optimal cost function value f_0 of problem (SIPcp). That is, $\mathrm{Opt}(4.1) \leq 0$, and hence $f^* \leq f_0$. The following remark reveal some further results obtained from Lemma 4.1 under some additional conditions.

Remark 4.2. Suppose that $f_0 = f^*$. Let $\tilde{x}$ denote the optimal solution of problem (4.1). Then, it holds that

$$f(\tilde{x}) - f_0 \leq 0,\ \max\{a_v^T \hat{x} - b_v,\ \forall v \in V\} \leq 0.$$

This conclusion is obtained from Lemma 4.1, where the second inequality implies that $\tilde{x}$ is a feasible solution to problem (SIPcp). Since x^* is the optimal solution of problem (SIPcp), it is clear from the first inequality that

$$f^* = f(x^*) \leq f(\tilde{x}) \leq f_0 = f^*.$$

Therefore, $\tilde{x}$ is also an optimal solution of problem (SIPcp). Moreover, by virtue of Lemma 3.1, $\tilde{x}$ is also an optimal solution of problem (CP), and the corresponding constrained matrix Z lies in the cone $\mathbb{C}_n$ of copositive matrices.

Now, a new algorithm for solving problem (SIPcp) is given below, which is referred to as Algorithm 1.

Algorithm 1.

Step 0. Choose a predicted objective function value f_0 and a finite set V_0 such that $V_0 \subset V$, and set $k := 0$.

Step 1. Solve problem (4.1) given below

$$\min_{x \in \mathbb{X}} \max\{f(x) - f_0,\ \max\{a_v^T x - b_v,\ \forall v \in V_k\}\}$$

to obtain an optimal solution x_k with the corresponding optimal objective function value Ψ_k.

Step 2. If $\Psi_k > 0$, then the algorithm stops. There does not admits a feasible solutions of problem (SIPcp) such that the corresponding objective function value is less than or equal to f_0. Or else, go to Step 3.

Step 3. If $\Psi_k \leq 0$, then solve the following problem

$$\max\{a_v^T x_k - b_v,\ \forall v \in V\}$$

to obtain an optimal solution v_k. If $a_{v_k}^T x_k - b_{v_k} \leq 0$, then stop. x_k is an achievable optimal solution with an improved optimal objective function value $f(x_k)$. Or else, go to next step.

Step 4. Let

$$V_{k+1} := V_k \cup \{v_k\},\ k := k+1,$$

and go to Step 1.

Remark 4.3. The solution of problem (4.1) is the key step of Algorithm 1. However, problem (4.1) is not an easy problem to solve. More specifically, the difficulty of solving problem (4.1) lies in solving the subproblem $\max\{a_v^T x - b_v,\ \forall v \in V_k\}$ when the dimension of V_k is high, that is, how to efficiently deal with the continuous constraint $v \in V_k$ when the dimension of V_k is high. This is the major challenge in the development of solution methods for semi-infinite programming problems. A popular approach is to discretize the continuous constraint $v \in V_k$ using a grid method. For example, we discretize the constraint V_k with unit vector at each step of solving subproblem $\max\{a_v^T x - b_v,\ \forall v \in V_k\}$ for which the dimension of V_k is low. Problem (4.1) with higher dimensional index set constraints V_k can be efficiently solved only if an efficient solution method to subproblem $\max\{a_v^T x - b_v,\ \forall v \in V_k\}$ with higher dimensional index set constraints V_k is developed. For our paper, the main contribution is to show that for a feasible solution of the problem (SIPcp) with a corresponding objective function value f_0, Algorithm 1 can output a solution whose corresponding objective function value is less than or equal to the given approximate value f_0. Else, there exists no feasible solution of the problem (SIPcp) such that its objective function value is less than or equal to f_0. The development of effective solution methods for solving subproblem $\max\{a_v^T x - b_v,\ \forall v \in V_k\}$ with higher dimensional index set is a challenging future research topic.

Remark 4.4. Note that in Step 3 of Algorithm 1, it follows from the definition of problem (4.1) that there exists an iteration point $x_k \in \mathbb{X}$ such that

$$\max\{f(x_k) - f_0,\ \max\{a_v^T x_k - b_v,\ \forall v \in V_k\}\} \leq 0,$$

which combined with $a_{v_k}^T x_k - b_{v_k} \leq 0$ gives

$$f(x_k) - f_0 \leq 0,\ a_v^T x_k - b_v \leq 0,\ \forall v \in V,$$

where the second inequality indicates that x_k is a feasible solution of problem (SIPcp). Furthermore, problem (4.1) also provides a tighter bound for problem (SIPcp) because $f^* \leq f(x_k) \leq f_0$. In particular, if $f^* = f(x_k)$, then x_k is an optimal solution of problem (SIPcp).

Lemma 4.5. *If $f^* < f_0$ and $\max\{a_v^T x^* - b_v,\ \forall v \in V\} < 0$, then there exists an $\varepsilon > 0$ such that the objective function value of problem (4.1) equal to $-\varepsilon$, i.e.,*

$$-\varepsilon = \min_{x \in \Lambda} \max\{f(x) - f_0,\ \max\{a_v^T x - b_v,\ \forall v \in V\}\}, \tag{4.2}$$

where $\Lambda := \mathbb{X} \cap \mathbb{N}(x^*,\ \lambda)$, and $\mathbb{N}(x^*,\ \lambda)$ denotes the x^*-neighbourhood with radius λ.

Proof. Since $f^* < f_0$ and $\max\{a_v^T x^* - b_v,\ \forall v \in V\} < 0$, and due to the continuity of the functions involved, there exists a direction d such that for sufficiently small $\lambda > 0$, it holds that

$$-\varepsilon_1 := f(x^* + \lambda d) - f_0 < 0,\ -\varepsilon_2 := \max\{a_v^T (x^* + \lambda d) - b_v,\ \forall v \in V\} < 0$$

for sufficiently small positive parameters ε_1 and ε_2. Moreover, since the set $\mathbb{X}$ is compact, the finiteness of the optimal objective function value of problem (4.2) over the set $\mathbb{X} \cap \mathbb{N}(x^*,\ \lambda)$ is assured. It is denoted by a positive parameter ε as follows:

$$\begin{aligned} -\varepsilon: \quad &= \min_{x\in\Lambda} \max\{f(x) - f_0,\ \max\{a_v^T x - b_v,\ \forall v \in V\}\} \\ &\leq \max\{f(x^* + \lambda d) - f_0,\ \max\{a_v^T (x^* + \lambda d) - b_v,\ \forall v \in V\}\} \\ &= \max\{-\varepsilon_1,\ -\varepsilon_2\} < 0. \end{aligned}$$

Thus, the validity of the conclusion of Lemma 4.5 follows readily. □

5 Finite Convergence

In this section, we analyze the termination of Algorithm 1 for a finite number of iterations. First, recall the important role played by the condition $f_0 > f^*$ on the predicted objective function value f_0 of problem (SIPcp) in the analysis in the sections above. The following theorem shows that Algorithm 1 terminates after a finite number of iterations under this condition.

Theorem 5.1. *If $f^* < f_0$ and $\max\{a_v^T x^* - b_v,\ \forall v \in V\} < 0$, then Algorithm 1 terminates in a finite number of iterations. Furthermore, Algorithm 1 outputs a solution with the corresponding objective function value being less than or equal to f_0 for problem (SIPcp).*

Proof. In view of the structure of Algorithm 1 and the results obtained in Lemma 4.5, it holds that $\Psi_k \leq -\varepsilon < 0$ at each iteration. Thus, Step 3 of Algorithm 1 will be executed. Hence, Algorithm 1 either terminates in a finite number of iterations, or else generates two infinite iteration sequences $\{x_k\}$ and $\{v_k\}$.

Without loss of generality, we assume that Algorithm 1 generates two infinite sequences of points $\{x_k\}$ and $\{v_k\}$ such that

$$a_{v_k}^T x_k - b_{v_k} > 0,\ a_{v_i}^T x_k - b_{v_i} \leq -\varepsilon < 0,\ \forall i < k, \tag{5.1}$$

where the first inequality is obtained from Step 3 of Algorithm 1, the second inequality is from the results obtained in Lemma 4.5, and ε is defined in (4.1). Furthermore, it follows from (5.1) that

$$|\ (a_{v_k}^T x_k - b_{v_k}) - (a_{v_i}^T x_k - b_{v_i})\ | > \varepsilon,\ \forall i < k. \tag{5.2}$$

On the other hand, since the constraint function $a_v^T x - b_v$ is continuously differentiable with respect to $v \in V$, its gradient $\nabla(a_v^T x - b_v)$ is also bounded above, denoted by $\mathbf{M}$, because the sets $\mathbb{X}$ and V are compact. Thus,

$$|\ (a_{v_k}^T x_k - b_{v_k}) - (a_{v_i}^T x_k - b_{v_i})\ | \leq \mathbf{M} \parallel v_k - v_i \parallel,\ \forall i < k \tag{5.3}$$

holds. Combining (5.2) with (5.3), it follows that

$$\varepsilon/\mathbf{M} < \parallel v_k - v_i \parallel,\ \forall i < k. \tag{5.4}$$

Since the set V is compact, there exists a convergence subsequence of $\{v_k\}$, denoted by the original sequence $\{v_k\}$, such that

$$\parallel v_k - v_i \parallel \leq \epsilon \tag{5.5}$$

holds for $k,\ i$ sufficiently large, where ϵ is an arbitrary small positive parameter. In particular, let $\epsilon := \varepsilon/\mathbf{M}$. Clearly, (5.5) is a contradiction to (5.4).

Finally, in view of the structure of Algorithm 1, it follows that Algorithm 1 outputs a solution with the corresponding objective function value being less than or equal to f_0 when Algorithm 1 stops at Step 3 after a finite number of iterations. Thus, the conclusions of Theorem 5.1 follow readily. □

The following theorem presents the result on the termination of Algorithm 1 after a finite number of iterations under the condition $f^* > f_0$.

Theorem 5.2. *If $f^* > f_0$, then Algorithm 1 terminates at Step 2 in a finite number of iterations. That is, there does not admit a feasible solution of problem (SIPcp) such that its objective function value is less than or equal to f_0.*

Proof. First, let us define an auxiliary problem as given below:

$$\Theta := \min_{x \in \mathbb{X}} f(x). \tag{5.6}$$

If $\Theta > f_0$, i.e., $f(x) - f_0 > 0$, $\forall x \in \mathbb{X}$, then by virtue of the structure of Algorithm 1 and the definition of problem (4.1), Algorithm 1 terminates at Step 2, showing that f_0 is not achievable since $\Psi_k > 0$ in each iteration.

On the other hand, if $\Theta \le f_0 < f^*$, i.e., $\exists \bar{x} \in \mathbb{X}$, then it holds that $f(\bar{x}) - f_0 \le 0$. By virtue of the structure of Algorithm 1, we assume that Algorithm 1 does not terminate at Step 2 in each iteration, and it generates two infinite iteration point sets $\{x_k\}$ and $\{v_k\}$. Note that $\Psi_k \le 0$ from the assumption. Then, it follows from the definition of Ψ_k that

$$\min_{x \in \mathbb{X}} \max\{f(x) - f_0,\ \max\{a_v^T x - b_v :\ \forall v \in V_k\}\} \le 0,$$

which implies that

$$\max\{f(x_k) - f_0,\ \max\{a_v^T x_k - b_v :\ \forall v \in V_k\}\} \le 0,$$

i.e.,

$$f(x_k) - f_0 \le 0,\ a_{v_i}^T x_k - b_{v_i} \le 0,\ \forall i < k. \tag{5.7}$$

On the one hand,

$$\begin{aligned} a_{v_k}^T x_k - b_{v_k} &= \max\{a_v^T x_k - b_v :\ \forall v \in V_k\} \\ &= \max\{a_v^T x_k - b_v :\ \forall v \in V_k,\ f(x_k) - f_0 \le 0\} \\ &\ge \min_{x \in \mathbb{X}} \max\{a_v^T x - b_v :\ \forall v \in V,\ f(x) - f_0 \le 0\}\}, \end{aligned} \tag{5.8}$$

where the second relationship "=" holds due to the fact that the constraint $f(x_k) - f_0 \le 0$ is redundant for $\max\{a_v^T x_k - b_v :\ \forall v \in V\}$. On the other hand, in view of the condition $f^* > f_0$, it follows from the result obtained in Lemma 4.1 that

$$\begin{aligned} 0 < \theta :\ &= \min_{x \in \mathbb{X}} \max\{f(x) - f_0,\ \max\{a_v^T x - b_v :\ \forall v \in V\}\} \\ &\le \max\{f(\bar{x}) - f_0,\ \max\{a_v^T \bar{x} - b_v :\ \forall v \in V\}\} \\ &= \max\{a_v^T \bar{x} - b_v :\ \forall v \in V,\ f(\bar{x}) - f_0 \le 0\}, \end{aligned} \tag{5.9}$$

where θ is the positive optimal objective function value of problem (4.1), the last relationship "=" holds from the fact that $f(\bar{x}) - f_0 \le 0$. Moreover, it is clear from (5.9) that

$$\begin{aligned} &\theta \le \max\{a_v^T \bar{x} - b_v :\ \forall v \in V,\ f(\bar{x}) - f_0 \le 0\} \\ &\Rightarrow \\ &\theta \le \min_{x \in \mathbb{X}} \max\{a_v^T x - b_v :\ \forall v \in V,\ f(x) - f_0 \le 0\}, \end{aligned}$$

which combined with (5.8) gives

$$a_{v_k}^T x_k - b_{v_k} \geq \theta > 0. \tag{5.10}$$

By virtue of (5.10) and the second inequality of (5.7), we have

$$(a_{v_k}^T x_k - b_{v_k}) - (a_{v_i}^T x_k - b_{v_i}) \geq \theta > 0, \ \forall i < k. \tag{5.11}$$

The remaining part of the proof is similar to the roof given for the last part of Theorem 5.1 and is therefore omitted. By using a similar argument as that given for the proof of Theorem 5.1, we can also draw a contradictory conclusion. Thus we can conclude that the assumption does not hold, and Algorithm 1 terminates at Step 2 after a finite number of iterations. At the same time, the results also show that problem (SIPcp) does not admit a feasible solution with the objective function value being less than or equal to f_0. □

Note that the above two theorems show the termination of Algorithm 1 in a finite number of iterations under the conditions of $f_0 > f^*$ and $f_0 < f^*$, respectively. In particular, when $f_0 = f^*$, we have $\Psi_k \leq 0$ for all k from the results obtained in Lemma 4.1. In view of the structure of Algorithm 1, it is clear that Step 3 is performed. By the definition of problem (4.1) and the condition $f_0 = f^*$, the result of $a_v^T x_k - b_v \leq 0$ holds for all k and $v \in V$. Thus, Algorithm 1 terminates at Step 3 in the first iteration, and x^* is also an optimal solution of problem (SIPcp).

6 Numerical Experiments

In this section, two numerical examples are considered and solved to show the feasibility of Algorithm 1. These problems are solved by Algorithm 1, which is implemented by using MATLAB R2018a on the Windows 10 platform.

Example 6.1 Consider problem (CP) with the objective function $f(x) = x$ and the coefficient matrixes

$$Q = \begin{pmatrix} 4 & -1 & 0 \\ -1 & 1 & 0 \\ 0 & 0 & 4 \end{pmatrix}, \ B = \begin{pmatrix} 0 & 0 & 5 \\ 0 & 0 & -2 \\ 5 & -2 & 0 \end{pmatrix},$$

as well as $-5 \leq x \leq 5$.

In the initial process of running Algorithm 1, the predicted objective function value was set as $f_0 = -5$, the initial finite index set was chosen as $V_0 = \{[0, 0, 0]^T\}$. Then, Algorithm 1 terminates at $k = 2$, with $f^* = 5$ and $x^* = 5$. Moreover, the current corresponding constraint matrix is equal to

$$5Q + B = 5 \begin{pmatrix} 4 & -1 & 0 \\ -1 & 1 & 0 \\ 0 & 0 & 4 \end{pmatrix} + \begin{pmatrix} 0 & 0 & 5 \\ 0 & 0 & -2 \\ 5 & -2 & 0 \end{pmatrix} = \begin{pmatrix} 20 & -5 & 5 \\ -5 & 5 & -2 \\ 5 & -2 & 20 \end{pmatrix},$$

whose eigenvalues are approximately equal to $(3, 15, 26)$ that are all strictly greater than 0. At the same time, in view of the definition of copositive matrix (1.1), we can conclude that $5Q + B \in \mathbb{C}_3$, that is, there does admits a feasible solution $x^* = 5$ of problem (CP).

Example 6.2 Consider problem (CP) with the objective function $f(x) = x_1^2 + x_2$ and the coefficient matrixes

$$Q_1 = \begin{pmatrix} 1 & -1 & 1 \\ -1 & 1 & -1 \\ 1 & -1 & 1 \end{pmatrix}, \ Q_2 = \begin{pmatrix} 1 & 2 & 0 \\ 2 & 0 & 3 \\ 0 & 3 & 5 \end{pmatrix}, \ B = \begin{pmatrix} 5 & 3 & 4 \\ 3 & -2 & 0 \\ 4 & 0 & -2 \end{pmatrix},$$

as well as $0 \le x_1 \le 1,\ 0 \le x_2 \le 1$.

In the initial step of Algorithm 1, the predicted objective function value was chosen as $f_0 = 2$, and the initial finite set $V_0 = \{[0, 0, 1]^T\}$. Algorithm 1 terminates at Step 2 at $k = 2$ iteration, with $x_1 = x_2 = 0$. At this point, $\Psi_2 = 5.5 > 0$ and it follows that there does not admits a feasible solutions of problem (SIPcp) such that the corresponding objective function value is less than or equal to $f_0 = 2$. Moreover, the current corresponding constraint matrix $x_1Q_1 + x_2Q_2 + B$ is equal to B, whose eigenvalues are equal to $(-4.6, -2.0, 7.6)$ that are not all strictly greater than 0. It is also easy to verify that $B \notin \mathbb{C}_3$ from the definition of copositive matrix (1.1), that is, there does not admits a feasible solution of problem (CP) such that the corresponding objective function value is less than or equal to $f_0 = 2$.

7 Conclusions

In this paper, a class of copositive programming problem (CP) with a continuous objective function is considered. We show that problem (CP) can be reformulated equivalently as a semi-infinite programming problem through the use of a key lemma. Then, a new discretization method is proposed to solve the transformed problem. Under some mild conditions, we show that the proposed new method will terminated in a finite number of iterations, giving rise to a feasible solution with the corresponding cost function value better than or equal to the predicted optimal cost function value of the original problem, or confirming that there does not admit a feasible solution for the original problem. Two numerical examples are considered and solved, showing the feasibility of the proposed algorithm.

Acknowledgements

All authors are gratefull to the chief-editor Professor Li-qun Qi and the two anonymous reviewers for their valuable comments and suggestions, which have greatly improved the earlier version of our paper.

References

[1] F. Ahmed, M. Dür and G. Still, Copositive programming via semi-infinite optimization, *J. Optimiz. Theory App.* 159 (2013) 322–340.

[2] F. Ahmed, Relations between semidefinite, copositive, semi-infinite and integer programming, Universuty of Twente, 2010.

[3] S. Burer, On the copositive representation of binary and continuous nonconvex quadratic programs, *Math. Program.* 120 (2009) 479–495.

[4] J.W. Blankenship and J.E. Falk, Infinitely constrained optimization problems, *J. Optimiz. Theory App.* 19 (1976) 261–281.

[5] M. Dür, Copositive programming-a survey, in: *Recent Advances in Optimization and Its Applications in Engineering*, 2010, pp. 3–20.

[6] C. Dobre and J. Vera, Exploiting symmetry in copositive programs via semidefinite hierarchies, *Math. Program.* 151 (2015) 659–680.

[7] M. Dür and F. Rendl, Conic optimization: a survey with special focus on copositive optimization and binary quadratic problems, *EURO Journal on Computational Optimization* 9 (2021) 1–12.

[8] O. Kostyukova and T. Tchemisova, Regularization algorithms for linear copositive programming problems: an approach based on the concept of immobile indicese, *Algorithms* 15 (2022) 1–17.

[9] E. de Klerk and D.V. Pasechnik, Approximation of the Stability Number of a Graph via Copositive Programming, *SIAM J. Optim.* 12 (2002) 875–892.

[10] A. Mittal and G.A. Hanasusanto, Finding minimum volume circumscribing ellipsoids using generalized copositive programming, *Oper. Res.* **70** (2021) 1–16.

[11] K.G. Murty and S.N. Kabadi, Some NP-complete problems in quadratic and nonlinear programming, *Math. Program.* 39 (1987) 117–129.

[12] D.V. Nguyen, Reformulation of the quadratic multidimensional knapsack problem as copositive/completely positive prorams, *Advances in Intelligent Systems and Computing*, 629 (2018) 13-22.

[13] R. Reemtsen and J.J. Rückmann, *Semi-Infinite Programming*, Springer Science & Business Media, 1998.

[14] G. Sagol and E.A. Yildirim, Analysis of copositive optimization based linear programming bounds on standard quadratic optimization, *J. Global Optim.* 63 (2015) 37–59.

[15] G. Still, Discretization in semi-infinite programming: the rate of convergence, *Math. Program.* 91 (2001) 53–69.

[16] L. Zhang, S.Y. Wu and M.A. López, A new exchange method for convex semi-infinite programming, *SIAM J. Optimiz.* 20 (2010) 2959–2977.

Manuscript received 31 January 2023
revised 13 June 2023
accepted for publication 26 July 2023

Chuanhao Guo
School of Economics and Management
Zhejiang Sci-Tech University
Hangzhou 310018, Zhejiang, China;
School of Mathematical Sciences
Sunway University
Selangor Darul Ehsan 47500, Malaysia
E-mail address: guo-ch@163.com

Pengfei Ma
Department of Mathematics
Zhejiang University of Science and Technology
Hangzhou 310023, Zhejiang, China
E-mail address: mathpengfeima@126.com

Kok Lay Teo
School of Mathematical Sciences
Sunway University
Selangor Darul Ehsan 47500, Malaysia
E-mail address: K.L.Teo@curtin.edu.au