SCIENCE CHINA
Mathematics

• ARTICLES •

October 2015 Vol. 58 No. 10: 2245–2254
doi: 10.1007/s11425-014-4959-z

Partially positive matrices

ZHOU AnWa & FAN JinYan*

Department of Mathematics, and MOE-LSC, Shanghai Jiao Tong University, Shanghai 200240*, China*
Email: congcongyan@sjtu.edu.cn, jyfan@sjtu.edu.cn

Received January 19, 2014; accepted August 28, 2014; published online December 24, 2014

Abstract A real $n \times n$ symmetric matrix P is partially positive (PP) for a given index set $I \subseteq \{1, \dots, n\}$ if there exists a matrix V such that $V(I,:) \geqslant 0$ and $P = VV^{\mathrm{T}}$. We give a characterization of PP-matrices. A semidefinite algorithm is presented for checking whether a matrix is partially positive or not. Its properties are studied. A PP-decomposition of a matrix can also be obtained if it is partially positive.

Keywords partially positive matrices, completely positive matrices, $\mathcal{A}$-truncated K-moment problem, semidefinite algorithm

MSC(2010) 15A23, 90C22

Citation: Zhou A W, Fan J Y. Partially positive matrices. Sci China Math, 2015, 58: 2245–2254, doi: 10.1007/s11425-014-4959-z

1 Introduction

Let I be a subset of $\{1, \dots, n\}$. A real $n \times n$ symmetric matrix P is partially positive (PP) for I if there exist vectors $v_1, \dots, v_l \in \mathbb{R}^n$ such that

$$P = v_1 v_1^{\mathrm{T}} + \cdots + v_l v_l^{\mathrm{T}}, \quad v_i(I) \geqslant 0, \tag{1.1}$$

where l is called the length of the decomposition (1.1). If P is partially positive, we call (1.1) a PP-decomposition of P. Specially, if $I = \{1, \dots, n\}$, we call P is completely positive (CP) and (1.1) is a CP-decomposition of P. Clearly, a PP-matrix is positive semidefinite and $P(I, I)$ is CP.

CP-matrices are special cases of PP-matrices. They have wide applications in general quadratic programming [2], etc. Zhou and Fan [19] proposed a semidefinite algorithm for the CP-matrix completion problem, which includes the CP-checking as a special case. In this paper, we consider a more general problem: How do we check whether a matrix is partially positive for a given index set? If it is not partially positive, can we get a certificate for this? If it is partially positive, can we get a PP-decomposition for it? To the best of our knowledge, little is known about that. We characterize the PP-matrices and propose a semidefinite algorithm for the problem of checking the partial positivity.

The paper is organized as follows. In Section 2, we give a necessary and sufficient condition to characterize PP-matrices. In Section 3, we formulate the problem of checking the partial positivity of a matrix as an $\mathcal{A}$-truncated K-moment problem. A semidefinite algorithm is proposed for it. Its properties are also studied. In Section 4, we propose another approach to the problem. Some computational results are reported in Section 5. Finally, we conclude the paper in Section 6.

*Corresponding author

2 A characterization of PP-matrices

Denote by $\mathcal{S}_n$ the set of real $n \times n$ symmetric matrices. Suppose $Q \in \mathcal{S}_n$ is in the block form

$$Q = \begin{pmatrix} B & C^{\mathrm{T}} \\ C & A \end{pmatrix}, \tag{2.1}$$

where A and B are square matrices.

The generalized Schur complement of A in Q, denoted by $S(Q, A)$, is the matrix $B - C^{\mathrm{T}} A^{\dagger} C$, where $A^{\dagger}$ is the Moore-Penrose generalized inverse.

Lemma 2.1 (See [1])**.** *Let Q given in* (2.1) *be positive semidefinite. Then the matrices $S(Q, A)$ and*

$$\begin{pmatrix} C^{\mathrm{T}} A^{\dagger} C & C^{\mathrm{T}} \\ C & A \end{pmatrix}$$

are positive semidefinite.

Denote by $R(A)$ the range space of A. If Q is positive semidefinite, then

$$R([C, A]) = R(A). \tag{2.2}$$

So, $R(C) \subseteq R(A)$. Therefore,

$$AA^{\dagger} C = C. \tag{2.3}$$

Based on the above results, we give a necessary and sufficient condition for a matrix to be partially positive.

Theorem 2.2. *Let $I \subseteq \{1, \ldots, n\}$. Then $P \in \mathcal{S}_n$ is partially positive for I if and only if P is positive semidefinite and $P(I, I)$ is completely positive.*

Proof. If P is partially positive for I, by (1.1), P is positive semidefinite and $A = P(I, I)$ is completely positive.

Conversely, suppose P is positive semidefinite and $P(I, I)$ is completely positive. Then there exists a permutation matrix E such that

$$\tilde{P} = E^{\mathrm{T}} P E = \begin{pmatrix} B & C^{\mathrm{T}} \\ C & A \end{pmatrix}, \tag{2.4}$$

where $A = P(I, I)$. So, there exists a $D \geqslant 0$ such that

$$A = DD^{\mathrm{T}}. \tag{2.5}$$

Note that $R(A) = R(D)$. By (2.3),

$$DD^{\dagger} C = C.$$

Let

$$X = D^{\dagger} C. \tag{2.6}$$

Then

$$DX = C. \tag{2.7}$$

By Lemma 2.1,

$$B - X^{\mathrm{T}} X = B - C^{\mathrm{T}} (D^{\dagger})^{\mathrm{T}} D^{\dagger} C = B - C^{\mathrm{T}} A^{\dagger} C \succeq 0.$$

Thus, there exists an L such that

$$B - X^{\mathrm{T}} X = LL^{\mathrm{T}}. \tag{2.8}$$

So, by (2.5)–(2.8),

$$\tilde{P} = \begin{pmatrix} B & C^{\mathrm{T}} \\ C & A \end{pmatrix} = \begin{pmatrix} B - X^{\mathrm{T}} X & 0 \\ 0 & 0 \end{pmatrix} + \begin{pmatrix} X^{\mathrm{T}} X & C^{\mathrm{T}} \\ C & A \end{pmatrix}$$

$$= \begin{pmatrix} L \\ 0 \end{pmatrix} \begin{pmatrix} L \\ 0 \end{pmatrix}^{\mathrm{T}} + \begin{pmatrix} X^{\mathrm{T}} \\ D \end{pmatrix} \begin{pmatrix} X^{\mathrm{T}} \\ D \end{pmatrix}^{\mathrm{T}}. \tag{2.9}$$

Therefore,

$$P = E \begin{pmatrix} L \\ 0 \end{pmatrix} \begin{pmatrix} L \\ 0 \end{pmatrix}^{\mathrm{T}} E^{\mathrm{T}} + E \begin{pmatrix} X^{\mathrm{T}} \\ D \end{pmatrix} \begin{pmatrix} X^{\mathrm{T}} \\ D \end{pmatrix}^{\mathrm{T}} E^{\mathrm{T}} \tag{2.10}$$

is a PP-decomposition of P for I. The proof is completed. □

Theorem 2.2 gives a way to check whether a matrix is PP.

Proposition 2.3. *Suppose $I_1 \subseteq I_2 \subseteq \{1,\dots,n\}$ and $P \in \mathcal{S}_n$ is not partially positive for I_1. Then P is not partially positive for I_2 either.*

Proof. We prove by contradiction. Suppose P is partially positive for I_2. Then there exists a matrix V such that

$$P = VV^{\mathrm{T}}, \quad V(I_2,:) \geqslant 0.$$

Since $I_1 \subseteq I_2 \subseteq \{1,\dots,n\}$, we have

$$P = VV^{\mathrm{T}}, \quad V(I_1,:) \geqslant 0.$$

Hence, P is partially positive for I_1, which is a contradiction. The proof is completed. □

Let $I \subseteq \{1,\dots,n\}$ be given. Denote by

$$\mathcal{E} = \{v \in \mathbb{R}^n : v(I) \geqslant 0\}$$

the set of partially positive vectors and

$$\mathcal{P}_n = \left\{ \sum_i v_i v_i^{\mathrm{T}} : v_i \in \mathcal{E} \right\}$$

the partially positive cone. We have the following result.

Proposition 2.4. *The partially positive cone $\mathcal{P}_n$ is proper* (*i.e., closed, convex, point and full-dimensional*).

Proof. Obviously, $\mathcal{E} \subseteq \mathbb{R}^n$ is a closed cone and has nonempty interior. By [6, Proposition 5], $\mathcal{P}_n$ is proper. □

3 A semidefinite algorithm for checking PP

It is shown in [4] that checking a CP matrix is NP-hard. Thus checking a PP matrix is also NP-hard. In recent years, semidefinite programming relaxation and approximation algorithms have been presented for some NP-hard problems, for example, see [11, 12, 18]. In this section, we show how to formulate the problem of checking a PP matrix as an $\mathcal{A}$-truncated K-moment problem ($\mathcal{A}$-TKMP) and propose a semidefinite algorithm for it. The convergence of the algorithm is also discussed. A PP-decomposition of a matrix can also be obtained if it is PP.

3.1 Formulation as an $\mathcal{A}$-TKMP

Let $\mathbb{N}$ be the set of nonnegative integers. For $\alpha = (\alpha_1,\dots,\alpha_n) \in \mathbb{N}^n$, denote $|\alpha| := \alpha_1 + \cdots + \alpha_n$. Let

$$\mathcal{A} := \{\alpha \in \mathbb{N}^n : |\alpha| = 2\}. \tag{3.1}$$

Note that a symmetric matrix can be identified by a vector consisting of its upper triangular entries. Then, $P \in \mathcal{S}_n$ can be identified as a vector as

$$\boldsymbol{p} = (\boldsymbol{p}_\alpha)_{\alpha\in\mathcal{A}} \in \mathbb{R}^{\mathcal{A}}, \quad \boldsymbol{p}_\alpha = P_{ij} \quad \text{if} \quad \alpha = e_i + e_j, \quad i \leqslant j,$$

where $\mathbb{R}^{\mathcal{A}}$ denotes the space of real vectors indexed by $\alpha \in \mathcal{A}$ and e_i is the i-th unit vector in $\mathbb{R}^n$. We call $\boldsymbol{p}$ an $\mathcal{A}$-truncated moment sequence ($\mathcal{A}$-tms) (see [8, 14]).

Let

$$K = \{x \in \mathbb{R}^n : x^{\mathrm{T}}x - 1 = 0, x(I) \geqslant 0\}. \tag{3.2}$$

Every vector satisfying (1.1) is a multiple of a vector in K. So, by (1.1), P is partially positive if and only if there exist vectors $v_1, \dots, v_l \in K$ and $\rho_1, \dots, \rho_l > 0$ such that

$$P = \rho_1 v_1 v_1^{\mathrm{T}} + \cdots + \rho_l v_l v_l^{\mathrm{T}}. \tag{3.3}$$

The $\mathcal{A}$-truncated K-moment problem ($\mathcal{A}$-TKMP) studies whether or not a given $\mathcal{A}$-tms $\boldsymbol{p}$ admits a K-measure μ, i.e., a nonnegative Borel measure μ supported in K such that

$$\boldsymbol{p}_\alpha = \int_K x^\alpha d\mu, \quad \forall\, \alpha \in \mathcal{A},$$

where $x^\alpha := x_1^{\alpha_1} \cdots x_n^{\alpha_n}$. A measure μ satisfying the above is called a K-representing measure for $\boldsymbol{p}$. A measure is called finitely atomic if its support is a finite set, and is called l-atomic if its support consists of at most l distinct points. We refer to [14] for representing measures of truncated moments sequences.

Hence, by (1.1), a symmetric matrix P, with the identifying vector $\boldsymbol{p} \in \mathbb{R}^{\mathcal{A}}$, is partially positive if and only if $\boldsymbol{p}$ admits an l-atomic K-measure, i.e.,

$$\boldsymbol{p} = \rho_1 [v_1]_{\mathcal{A}} + \cdots + \rho_l [v_l]_{\mathcal{A}}, \tag{3.4}$$

where each $v_i \in K$, $\rho_i > 0$, and

$$[v]_{\mathcal{A}} := (v^\alpha)_{\alpha \in \mathcal{A}}.$$

In other words, checking the partial positivity of a matrix is equivalent to an $\mathcal{A}$-TKMP with $\mathcal{A}$ and K given in (3.1) and (3.2), respectively.

3.2 A semidefinite algorithm

We start with some basics about localizing matrices. Denote

$$\mathbb{R}[x]_{\mathcal{A}} := \mathrm{span}\{x^\alpha : \alpha \in \mathcal{A}\}.$$

We say $\mathbb{R}[x]_{\mathcal{A}}$ is K-full if there exists a polynomial $r \in \mathbb{R}[x]_{\mathcal{A}}$ such that $r|_K > 0$ (see [7]). Choose $r = \sum_{i=1}^n x_i^2 \in \mathbb{R}[x]_{\mathcal{A}}$, then $r|_K > 0$. So $\mathbb{R}[x]_{\mathcal{A}}$ is K-full for $\mathcal{A}$ and K given in (3.1) and (3.2), respectively. An $\mathcal{A}$-tms $y \in \mathbb{R}^{\mathcal{A}}$ defines an $\mathcal{A}$-Riesz function $\mathscr{L}_y$ acting on $\mathbb{R}[x]_{\mathcal{A}}$ as

$$\mathscr{L}_y\left(\sum_{\alpha \in \mathcal{A}} r_\alpha x^\alpha\right) := \sum_{\alpha \in \mathcal{A}} r_\alpha y_\alpha. \tag{3.5}$$

We also denote $\langle r, y\rangle := \mathscr{L}_y(r)$ for convenience. Let $\mathbb{N}_d^n := \{\alpha \in \mathbb{N}^n : |\alpha| \leqslant d\}$ and $\mathbb{R}[x]_d := \mathrm{span}\{x^\alpha : \alpha \in \mathbb{N}_d^n\}$. For $s \in \mathbb{R}^{\mathbb{N}_{2k}^n}$ and $q \in \mathbb{R}[x]_{2k}$, the k-th localizing matrix of q generated by s is the symmetric matrix $L_q^{(k)}(s)$ satisfying

$$\mathscr{L}_s(qr^2) = \mathrm{vec}(r)^{\mathrm{T}} L_q^{(k)}(s)\, \mathrm{vec}(r), \quad \forall\, r \in \mathbb{R}[x]_{k - \lceil \deg(q)/2 \rceil}. \tag{3.6}$$

In the above, $\mathrm{vec}(r)$ denotes the coefficient vector of r in the graded lexicographical ordering, and $\lceil t \rceil$ denotes the smallest integer that is not smaller than t. In particular, when $q = 1$, $L_1^{(k)}(s)$ is called a k-th order moment matrix and denoted by $M_k(s)$. We refer to [7, 8, 14] for more details about localizing and moment matrices.

Let $I = \{i_1, \dots, i_m\} \subseteq \{1, \dots, n\}$, where $i_1 < i_2 < \cdots < i_m$. For convenience, denote the polynomials

$$h(x) := x^{\mathrm{T}}x - 1, \quad g_0(x) := 1, \quad g_1(x) := x_{i_1}, \quad \dots, \quad g_m(x) := x_{i_m}.$$

Then, K can be equivalently described as

$$K = \{x \in \mathbb{R}^n : h(x) = 0, g(x) \geqslant 0\}, \tag{3.7}$$

where $g(x) = (g_0(x), g_1(x), \ldots, g_m(x))$. Obviously, K is nonempty compact.

As shown in [14], a necessary condition for $s \in \mathbb{R}^{\mathbb{N}^n_{2k}}$ to admit a K-measure is

$$L_h^{(k)}(s) = 0, \quad \text{and} \quad L_{g_j}^{(k)}(s) \succeq 0, \quad j = 0, 1, \ldots, m. \tag{3.8}$$

If, in addition to (3.8), s satisfies the rank condition

$$\operatorname{rank} M_{k-1}(s) = \operatorname{rank} M_k(s), \tag{3.9}$$

then s admits a unique K-measure, which is $\operatorname{rank} M_k(s)$-atomic (see Curto and Fialkow [3]). We say that s is flat if both (3.8) and (3.9) are satisfied.

Given two $\mathcal{A}$-truncated moment sequences $y \in \mathbb{R}^{\mathbb{N}^n_d}$ and $z \in \mathbb{R}^{\mathbb{N}^n_e}$, we say z is an extension of y, if $d \leqslant e$ and $y_\alpha = z_\alpha$ for all $\alpha \in \mathbb{N}^n_d$. We denote by $z|_{\mathcal{A}}$ the subvector of z, whose entries are indexed by $\alpha \in \mathcal{A}$. For convenience, we denote by $z|_d$ the subvector $z|_{\mathbb{N}^n_d}$. If z is flat and extends y, we say z is a flat extension of y. Note that an $\mathcal{A}$-tms $y \in \mathbb{R}^{\mathcal{A}}$ admits a K-measure if and only if it is extendable to a flat tms $z \in \mathbb{R}^{\mathbb{N}^n_{2k}}$ for some k (see [14]). Therefore, by (3.4), a matrix P is partially positive if and only if its identifying vector $\boldsymbol{p}$ has a flat extension.

Let $d > 2$ be an even integer. Choose a polynomial $R \in \mathbb{R}[x]_d$ and write it as

$$R(x) = \sum_{\alpha \in \mathbb{N}^n_d} R_\alpha x^\alpha.$$

Consider the linear optimization problem

$$\begin{aligned} \eta = \min_z \ & \sum_{\alpha \in \mathbb{N}^n_d} R_\alpha z_\alpha \\ \text{subject to} \quad & z|_{\mathcal{A}} = \boldsymbol{p}, \quad z \in \Upsilon_d(K), \end{aligned} \tag{3.10}$$

where

$$\Upsilon_d(K) = \{z \in \mathbb{R}^{\mathbb{N}^n_d} : z \text{ admits a } K\text{-measure}\}.$$

Note that since K is a compact set and $\mathbb{R}[x]_{\mathcal{A}}$ is K-full, the feasible set of (3.10) is compact convex and (3.10) has a minimizer for any generic polynomial R. Usually, we choose a generic positive definite $R \in \Sigma_{n,d}$, where $\Sigma_{n,d}$ is the set of all the sum of squares polynomials in n variables with degree d. Since $\Upsilon_d(K)$ is difficult to be described, we relax it by the cone

$$\Gamma_k(h, g) := \{z \in \mathbb{R}^{\mathbb{N}^n_{2k}} \mid L_h^{(k)}(z) = 0, L_{g_j}^{(k)}(z) \succeq 0, j = 0, 1, \ldots, m\}, \tag{3.11}$$

with $k \geqslant d/2$ an integer. The k-th order semidefinite relaxation of (3.10) is

$$\begin{aligned} \eta^k = \min_z \ & \sum_{\alpha \in \mathbb{N}^n_d} R_\alpha z_\alpha \\ \text{subject to} \quad & z|_{\mathcal{A}} = \boldsymbol{p}, \quad z \in \Gamma_k(h, g). \end{aligned} \tag{3.12}$$

Clearly, $\eta^k \leqslant \eta$ for all k. Suppose $z^{*,k}$ is a minimizer of (3.12). If $z^{*,k} \in \Upsilon_d(K)$, then $\eta^k = \eta$ and $z^{*,k}$ is a minimizer of (3.10), i.e., the relaxation (3.12) is exact for solving (3.10). If the relaxation (3.12) is infeasible, then (3.10) is also infeasible.

Based on solving the hierarchy of (3.12), we present a semidefinite algorithm for checking whether a symmetric matrix is partially positive as follows.

Algorithm 3.1. A semidefinite algorithm for checking PP.

Step 0. Choose a generic $R \in \Sigma_{n,d}$, and let $k := d/2$.

Step 1. Solve (3.12). If (3.12) is infeasible, then $\boldsymbol{p}$ does not admit a K-measure, i.e., P is not partially positive, and stop. Otherwise, compute a minimizer $z^{*,k}$. Let $t := 1$.

Step 2. Let $w := z^{*,k}|_{2t}$. If the rank condition (3.9) is not satisfied, go to Step 4.

Step 3. Compute the finitely atomic measure μ admitted by w,

$$\mu = \rho_1 \delta(v_1) + \cdots + \rho_l \delta(v_l),$$

where $l = \mathrm{rank} M_t(w)$, $v_i \in K$, $\rho_i > 0$, and $\delta(v_i)$ is the Dirac measure supported on the point v_i $(i = 1, \ldots, l)$. Stop.

Step 4. If $t < k$, set $t := t + 1$ and go to Step 2; otherwise, set $k := k + 1$ and go to Step 1.

Remark 3.2. Denote $[x]_d := (x^\alpha)_{\alpha \in \mathbb{N}^n_d}$. We choose $R = [x]^{\mathrm{T}}_{d/2} J^{\mathrm{T}} J [x]_{d/2}$ in (3.12), where J is a random square matrix obeying Gaussian distribution. We check the rank condition (3.9) numerically with the help of singular value decompositions [5]. The rank of a matrix is evaluated as the number of its singular values that are greater than or equal to 10^{-6}. We use Henrion and Lasserre's method in [9] to get an r-atomic K-measure for w.

Remark 3.3. If $I = \{1, \ldots, n\}$, the partially positive matrix reduces to the completely positive matrix. So, Algorithm 3.1 can also check whether a symmetric matrix is completely positive or not.

We show some properties of Algorithm 3.1, which can be deduced from [14, 16].

Theorem 3.4. *Algorithm* 3.1 *has the following properties*:

(1) *If* (3.12) *is infeasible for some k, then $\boldsymbol{p}$ admits no K-measures and the corresponding matrix P is not partially positive.*

(2) *If the matrix P is not partially positive, then* (3.12) *is infeasible for all k big enough.*

(3) *If the matrix P is partially positive, then for almost all generated R, we can asymptotically get a flat extension of $\boldsymbol{p}$ by solving the hierarchy of* (3.12)*. This gives a PP-decomposition of P.*

Remark 3.5. If the matrix P with the identifying vector $\boldsymbol{p} \in \mathbb{R}^{\mathcal{A}}$ is partially positive, then, under some general conditions, which are almost necessary and sufficient, we can get a flat extension of $\boldsymbol{p}$ by solving the hierarchy of (3.12), within finitely many steps (see [13, 15]). This always happens in our numerical experiments. After getting a flat extension of $\boldsymbol{p}$, we can get an l-atomic K-measure for $\boldsymbol{p}$, which then produces a PP-decomposition of P.

4 An alternative approach to checking PP

In this section, we propose an alternative approach to the problem of checking whether a symmetric matrix is partially positive or not.

Theorem 2.2 shows that a matrix $P \in \mathcal{S}_n$ is partially positive for a given index set $I \subseteq \{1, \ldots, n\}$ if and only if P is positive semidefinite and $P(I, I)$ is completely positive. So, to check the partial positivity of P, it is natural to first check whether P is positive semidefinite. If P is not positive semidefinite, then P is not PP for I; otherwise, we check whether $P(I, I)$ is CP.

Considering the above analysis, we present another algorithm for checking PP.

Algorithm 4.1.

Step 1. Check whether P is positive semidefinite. If not, output P is not PP, and stop.

Step 2. Check whether $P(I, I)$ is CP. Stop.

Remark 4.2. In Step 2, we can use Algorithm 3.1 to check whether $P(I, I)$ is CP as CP is a special case of PP. If $P(I, I)$ is CP, a CP-decomposition for it can also be obtained by Algorithm 3.1. Then, by (2.6), (2.8) and (2.10), we can get a PP-decomposition of P. In fact, we can also use the algorithms given in [14, 19] to check whether $P(I, I)$ is CP.

To check the partial positivity of P, Algorithm 3.1 uses the whole matrix P, while Algorithm 4.1 first checks whether P is positive semidefinite, then checks whether the submatrix $P(I, I)$ is CP. As the size of $P(I,I)$ is generally smaller than P, it is more efficient for Algorithm 3.1 to get a CP-decomposition of $P(I,I)$ than to get a PP-decomposition of P. In this sense, Algorithm 4.1 has the advantage of solving bigger problems.

The convergence of Algorithm 4.1 is similar to that of Algorithm 3.1, so we omit it here.

5 Numerical experiments

In this section, we present some numerical experiments to show how to check the partial positivity of a matrix by Algorithms 3.1 and 4.1. A PP-decomposition of a matrix is also given if it is PP. We use software GloptiPoly 3 [10] and SeDuMi [17] to solve the relaxation problem (3.12). We choose $d = 4$ and $k = 2$ in Step 0 of Algorithm 3.1.

Example 5.1. Consider the matrix P given as

$$P = \begin{pmatrix} 1 & 1 & 1 & 0 & 0 & 0 & 1 \\ 1 & 3 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 2 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 2 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 3 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 2 & 1 \\ 1 & 0 & 0 & 0 & 0 & 1 & 6 \end{pmatrix}, \tag{5.1}$$

and $I = \{1, 3, 4, 7\}$.

We apply Algorithms 3.1 and 4.1 to check whether P is PP or not. Algorithm 3.1 terminates at Step 1 with $k = 2$, i.e., (3.12) is infeasible. So, P is not partially positive for the given index set I. Algorithm 4.1 terminates at Step 1, hence P is not positive semidefinite, which also implies that P is not PP for I.

Example 5.2. Consider the matrix P given as

$$P = \begin{pmatrix} 1 & 0 & 1 & 0 & 0 & 0 & 1 \\ 0 & 3 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 2 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 2 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 3 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 2 & 1 \\ 1 & 0 & 0 & 0 & 0 & 1 & 6 \end{pmatrix}, \tag{5.2}$$

and $I = \{1, 3, 4, 6, 7\}$.

It can be checked that P is positive semidefinite. However, $P(I,I)$ is not completely positive (see [1, Example 2.9]). So, by Theorem 2.2, P is not PP for I. We use Algorithms 3.1 and 4.1 to verify this fact. Algorithm 3.1 terminates at Step 1 with $k = 3$, which implies that P is not PP for I. Algorithm 4.1 terminates at Step 2 as the relaxation problem of checking the complete positivity of $P(I,I)$ is infeasible. This also implies that P is not PP for I.

Now we let $\tilde{I} = \{3, 4, 6, 7\}$. Algorithm 3.1 terminates at Step 3 with $k = 4$. So P is PP for $\tilde{I}$. We obtain the PP-decomposition $P = \sum_{i=1}^{9} \rho_i v_i v_i^{\mathrm{T}}$, where ρ_i and v_i are

$$\begin{aligned}
\rho_1 &= 1.2444, \quad v_1 = (-0.1424, -0.2062, 0.0000, 0.3439, -0.3330, 0.8414, 0.0000)^{\mathrm{T}}, \\
\rho_2 &= 2.0634, \quad v_2 = (0.3140, -0.7968, 0.5014, 0.0000, -0.1230, 0.0000, 0.0000)^{\mathrm{T}}, \\
\rho_3 &= 1.3333, \quad v_3 = (-0.1528, -0.7787, 0.0000, 0.5822, 0.1771, 0.0000, 0.0000)^{\mathrm{T}}, \\
\rho_4 &= 3.3274, \quad v_4 = (0.0364, 0.1658, 0.0000, 0.0000, -0.5338, 0.0000, 0.8284)^{\mathrm{T}},
\end{aligned}$$

$$\begin{aligned}
\rho_5 &= 2.9337, \quad v_5 = (0.3239, 0.3954, 0.7106, 0.4797, 0.0610, 0.0000, 0.0000)^{\mathrm{T}},\\
\rho_6 &= 1.2861, \quad v_6 = (0.1889, -0.3929, 0.0000, 0.0000, 0.1649, 0.0033, 0.8847)^{\mathrm{T}},\\
\rho_7 &= 2.4879, \quad v_7 = (-0.2236, 0.0873, 0.0000, 0.5378, 0.6722, 0.4486, 0.0000)^{\mathrm{T}},\\
\rho_8 &= 0.4664, \quad v_8 = (0.4262, 0.3612, 0.0000, 0.1156, 0.3668, 0.7348, 0.0000)^{\mathrm{T}},\\
\rho_9 &= 3.8573, \quad v_9 = (0.2118, 0.0030, 0.0000, 0.0000, 0.3971, 0.3082, 0.8381)^{\mathrm{T}}.
\end{aligned} \tag{5.3}$$

We also apply Algorithm 4.1. It gives a CP-decomposition $P(\tilde{I},\tilde{I}) = DD^{\mathrm{T}}$, where

$$D = \begin{pmatrix} 1.4142 & 0.0000 & 0.0000 & 0.0000\\ 0.7071 & 1.2247 & 0.0000 & 0.0000\\ 0.0000 & 0.8165 & 1.1547 & 0.0000\\ 0.0000 & 0.0000 & 0.8660 & 2.2913 \end{pmatrix}.$$

Furthermore, we obtain a PP-decomposition $P = VV^{\mathrm{T}}$, where

$$V = \begin{pmatrix} 0 & 0.1910 & 0.3262 & 0.7071 & -0.4082 & 0.2887 & 0.3273\\ 1.7321 & 0 & 0 & 0 & 0 & 0 & 0\\ 0 & 0 & 0 & 1.4142 & 0.0000 & 0.0000 & 0.0000\\ 0 & 0 & 0 & 0.7071 & 1.2247 & 0.0000 & 0.0000\\ 0 & 1.4960 & 0.0000 & 0.0000 & 0.8165 & 0.2887 & -0.1091\\ 0 & 0 & 0 & 0.0000 & 0.8165 & 1.1547 & 0.0000\\ 0 & 0 & 0 & 0.0000 & 0.0000 & 0.8660 & 2.2913 \end{pmatrix}.$$

Algorithm 4.1 gets a shorter PP-decomposition than Algorithm 3.1.

Example 5.3. Consider the matrix P given as

$$P = \begin{pmatrix} 16 & 5 & 4 & 3 & 2 & 1\\ 5 & 20 & 5 & 4 & 3 & 2\\ 4 & 5 & 22 & 5 & 4 & 3\\ 3 & 4 & 5 & 22 & 5 & 4\\ 2 & 3 & 4 & 5 & 20 & 5\\ 1 & 2 & 3 & 4 & 5 & 16 \end{pmatrix}, \tag{5.4}$$

and $I = \{2, 5, 6\}$.

We first apply Algorithm 3.1 to check whether P is PP or not. It terminates at Step 3 with $k = 4$. So, P is PP for I. The PP-decomposition of P is $P = \sum_{i=1}^{11} \rho_i v_i v_i^{\mathrm{T}}$, where ρ_i and v_i are

$$\begin{aligned}
\rho_1 &= 3.4149, \quad v_1 = (-0.5049, 0.0000, -0.6365, 0.3539, 0.4634, 0.0000)^{\mathrm{T}},\\
\rho_2 &= 8.2685, \quad v_2 = (0.5086, 0.0000, -0.3375, -0.4051, 0.0000, 0.6806)^{\mathrm{T}},\\
\rho_3 &= 6.2683, \quad v_3 = (0.3977, 0.0000, -0.2598, 0.6151, 0.0000, 0.6292)^{\mathrm{T}},\\
\rho_4 &= 17.2625, \quad v_4 = (0.4610, 0.0000, -0.0600, 0.5212, 0.7157, 0.0000)^{\mathrm{T}},\\
\rho_5 &= 8.0393, \quad v_5 = (-0.5525, 0.0000, 0.1023, 0.5084, 0.4752, 0.4472)^{\mathrm{T}},\\
\rho_6 &= 18.8776, \quad v_6 = (0.3690, 0.8929, 0.1835, 0.1813, 0.0000, 0.0000)^{\mathrm{T}},\\
\rho_7 &= 9.7859, \quad v_7 = (-0.2277, 0.0000, 0.3458, 0.6714, 0.0000, 0.6146)^{\mathrm{T}},\\
\rho_8 &= 18.2278, \quad v_8 = (0.3654, 0.0000, 0.8535, 0.3715, 0.0000, 0.0000)^{\mathrm{T}},\\
\rho_9 &= 5.3541, \quad v_9 = (-0.1413, 0.7734, 0.1594, 0.3510, 0.0000, 0.4830)^{\mathrm{T}},\\
\rho_{10} &= 8.1673, \quad v_{10} = (-0.1681, 0.4625, 0.3300, -0.1349, 0.7942, 0.0000)^{\mathrm{T}},
\end{aligned} \tag{5.5}$$

$$\rho_{11} = 12.3336, \quad v_{11} = (0.0466, 0.0000, 0.4928, -0.4696, 0.5294, 0.5041)^{\mathrm{T}}.$$

We also apply Algorithm 4.1 to P for $I = \{2, 5, 6\}$. It gives a CP-decomposition $P(I, I) = DD^{\mathrm{T}}$, where

$$D = \begin{pmatrix} 0.0314 & 4.4720 & 0.0000 \\ 0.0000 & 0.6708 & 4.4215 \\ 3.8321 & 0.4204 & 1.0671 \end{pmatrix}.$$

A PP-decomposition is further obtained as $P = VV^{\mathrm{T}}$, where

$$V = \begin{pmatrix} 0.5631 & 0.2980 & 3.7764 & 0.0596 & 1.1176 & 0.2828 \\ 0 & 0 & 0 & 0.0314 & 4.4720 & 0.0000 \\ 4.4731 & 0 & 0.0000 & 0.4558 & 1.1149 & 0.7355 \\ 0.6641 & 4.3964 & 0.0000 & 0.6689 & 0.8898 & 0.9958 \\ 0 & 0 & 0 & 0.0000 & 0.6708 & 4.4215 \\ 0 & 0 & 0 & 3.8321 & 0.4204 & 1.0671 \end{pmatrix}.$$

Actually, it is a CP-decomposition for P as $V \geqslant 0$, moreover, the length of the decomposition above is shorter than that given by Algorithm 3.1.

In fact, P is a nonnegative symmetric strictly diagonally dominant matrix. So, it is completely positive (see [20]). Therefore, P is partially positive for all $I \subseteq \{1, \ldots, 6\}$. Let $I = \{1, \ldots, 6\}$. Algorithm 3.1 gives the following CP-decomposition $P = \sum_{i=1}^{16} \rho_i v_i v_i^{\mathrm{T}}$, where ρ_i and v_i are

$$\begin{aligned}
&\rho_1 = 12.0753, \quad v_1 = (0.0000, 0.0000, 0.0000, 0.3128, 0.0000, 0.9498)^{\mathrm{T}}, \\
&\rho_2 = 0.2638, \quad v_2 = (0.0000, 0.0000, 0.9998, 0.0207, 0.0000, 0.0000)^{\mathrm{T}}, \\
&\rho_3 = 7.5303, \quad v_3 = (0.0000, 0.0000, 0.7669, 0.0000, 0.3768, 0.5195)^{\mathrm{T}}, \\
&\rho_4 = 8.6983, \quad v_4 = (0.8242, 0.0000, 0.5580, 0.0971, 0.0000, 0.0000)^{\mathrm{T}}, \\
&\rho_5 = 13.2307, \quad v_5 = (0.0000, 0.0000, 0.0000, 1.0000, 0.0000, 0.0000)^{\mathrm{T}}, \\
&\rho_6 = 5.3722, \quad v_6 = (0.0000, 0.0000, 0.9243, 0.1038, 0.3673, 0.0000)^{\mathrm{T}}, \\
&\rho_7 = 8.1539, \quad v_7 = (0.0000, 0.0000, 0.6392, 0.7690, 0.0000, 0.0000)^{\mathrm{T}}, \\
&\rho_8 = 7.8696, \quad v_8 = (0.0000, 0.0000, 0.0000, 0.0629, 0.8458, 0.5297)^{\mathrm{T}}, \\
&\rho_9 = 3.6693, \quad v_9 = (0.7306, 0.5719, 0.0000, 0.0000, 0.0000, 0.3730)^{\mathrm{T}}, \\
&\rho_{10} = 10.4216, \quad v_{10} = (0.0000, 0.5994, 0.8005, 0.0000, 0.0000, 0.0000)^{\mathrm{T}}, \\
&\rho_{11} = 4.5920, \quad v_{11} = (0.0000, 0.9533, 0.0000, 0.1182, 0.0000, 0.2780)^{\mathrm{T}}, \\
&\rho_{12} = 7.6981, \quad v_{12} = (0.8216, 0.5481, 0.0000, 0.1566, 0.0000, 0.0000)^{\mathrm{T}}, \\
&\rho_{13} = 9.5664, \quad v_{13} = (0.0000, 0.0000, 0.0000, 0.2903, 0.9569, 0.0000)^{\mathrm{T}}, \\
&\rho_{14} = 4.8866, \quad v_{14} = (0.7753, 0.0000, 0.0000, 0.3467, 0.5279, 0.0000)^{\mathrm{T}}, \\
&\rho_{15} = 5.5739, \quad v_{15} = (0.0000, 0.9379, 0.0000, 0.3469, 0.0000, 0.0000)^{\mathrm{T}}, \\
&\rho_{16} = 6.3986, \quad v_{16} = (0.0000, 0.7571, 0.0000, 0.2081, 0.6193, 0.0000)^{\mathrm{T}}.
\end{aligned} \tag{5.6}$$

6 Conclusions

In this paper, we introduce the partially positive matrices and investigate whether a symmetric matrix P is partially positive for a given index set I. A semidefinite algorithm (see Algorithm 3.1) is presented. If P is not partially positive, a certificate for this can be obtained; if P is partially positive, a PP-decomposition can be obtained. An alternative approach (see Algorithm 4.1) is also proposed to check the partial positivity. It first checks whether P is positive semidefinite, then checks whether $P(I, I)$ is completely positive by Algorithm 3.1. Numerical results show that Algorithm 4.1 may give a shorter PP-decomposition of a matrix if it is partially positive.

Acknowledgements This work was supported by National Natural Science Foundation of China (Grant No. 11171217). The authors thank the associate editor and anonymous referees for their valuable comments and suggestions.

References

1 Berman A, Shaked-Monderer N. Completely Positive Matrices. Singapore: World Scientific, 2003
2 Bomze I M, Dür M, de Klerk E, et al. On copositive programming and standard quadratic optimization problems. J Global Optim, 2000, 18: 301–320
3 Curto R, Fialkow L. Truncated K-moment problems in several variables. J Operator Theory, 2005, 54: 189–226
4 Dickinson P J, Gijben L. On the computational complexity of membership problems for the completely positive cone and its dual. Comput Optim Appl, 2014, 57: 403–415
5 Golub G H, Van Loan C F. Matrix Computations, 3rd ed. Baltimore: The Johns Hopkins University Press, 1996
6 Gowda M S, Sznajder R, Tao J. The automorphism group of a completely positive cone and its Lie algebra. Linear Algebra Appl, 2013, 438: 3862–3871
7 Fialkow L, Nie J. The truncated moment problem via homogenization and flat extensions. J Funct Anal, 2012, 263: 1682–1700
8 Helton J W, Nie J. A semidefinite approach for truncated K-moment problems. Found Comput Math, 2012, 12: 851–881
9 Henrion D, Lasserre J. Detecting global optimality and extracting solutions in GloptiPoly. Lect Notes Control Inform Sci, 2005, 312: 293–310
10 Henrion D, Lasserre J, Löfberg J. GloptiPoly 3: Moments, optimization and semidefinite programming. Optim Methods Softw, 2009, 24: 761–779
11 Huang Y, Zhang S. Approximation algorithms for indefinite complex quadratic maximization problems. Sci China Math, 2010, 53: 2697–2708
12 Lasserre J B. Moments, Positive Polynomials and Their Applications. London: Imperial College Press, 2009
13 Nie J. Certifying convergence of Lasserre's hierarchy via flat truncation. Math Program Ser A, 2013, 142: 485–510
14 Nie J. The $\mathcal{A}$-truncated K-moment problem. Found Comput Math, 2014, 14: 1243–1276
15 Nie J. Optimality conditions and finite convergence of Lasserre's hierarchy. Math Program Ser A, 2014, 146: 97–121
16 Nie J. Linear optimization with cones of moments and nonnegative polynomials. Math Program Ser B, 2014, doi: 10.1007/s10107-014-0797-6
17 Sturm J F. SeDuMi 1.02: A MATLAB toolbox for optimization over symmetric cones. Optim Methods Softw, 1999, 11: 625–653
18 Xia Y. New semidefinite programming relaxations for box constrained quadratic program. Sci China Math, 2013, 56: 877–886
19 Zhou A, Fan J. The CP-matrix completion problem. SIAM J Matrix Anal Appl, 2014, 35: 127–142
20 Zhou A, Fan J. Interiors of completely positive cones. ArXiv:1401.1255, 2014